

Lecture #4: Exploratory data analysis (EDA)

Previously on

- Python for atomistic modeling
- ASE's Atoms and Pymatgen's Structure
 - Neighbor list
 - Voronoi partitioning
- Data in materials informatics
- Computational data
 - The Materials project API

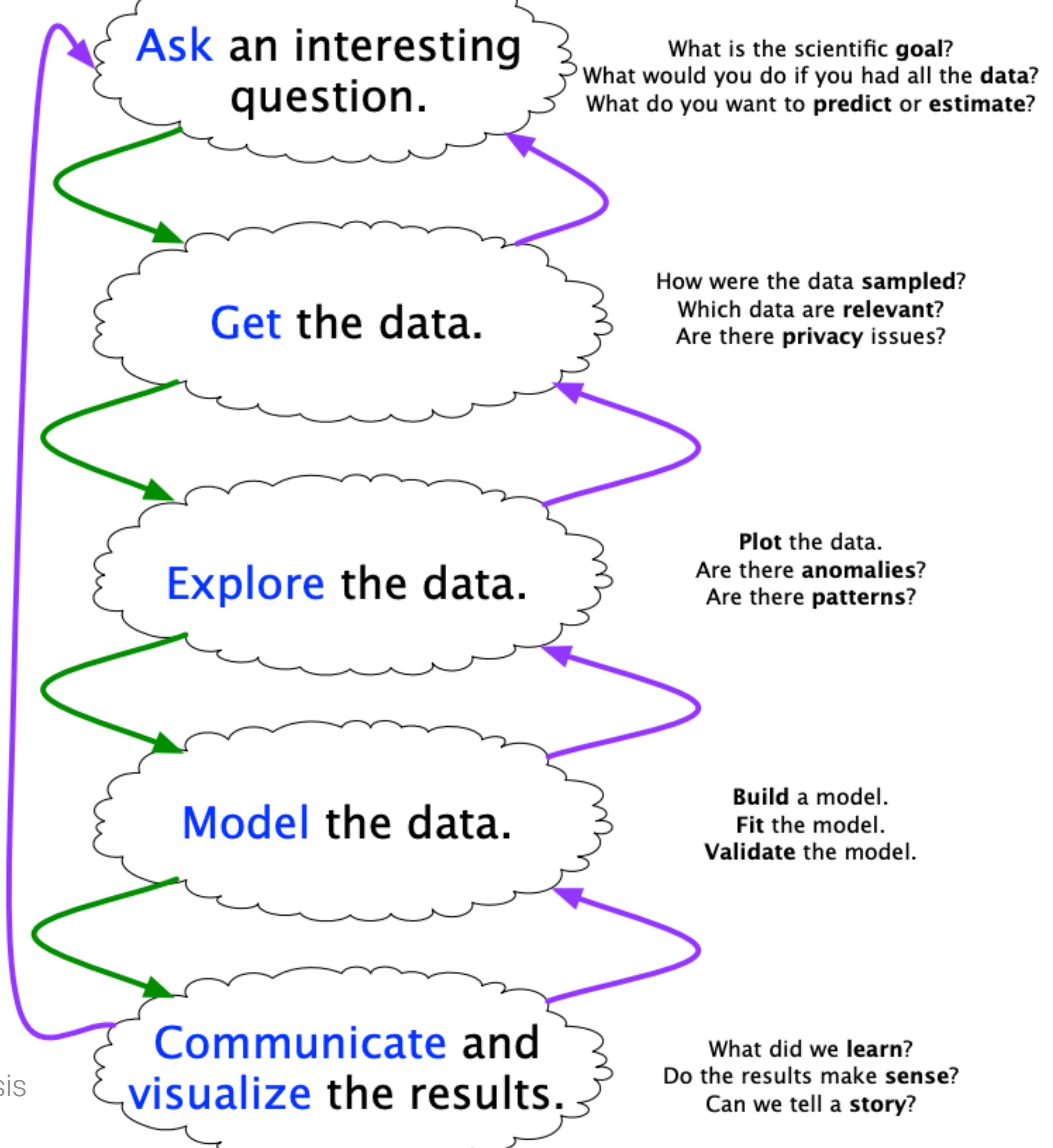
Goals/Agenda

- Explain why visualizing data is important when analyzing data
- Provide tips on how to use visualization to explore data

Attribution

- Parts of these slides are adapted from the excellent lecture on exploratory data analysis from the course CS 109A: Introduction to Data Science by Pavlos Protopapas & Kevin Rader shared under MIT licence
 - <https://harvard-iacs.github.io/2018-CS109A/lectures/lecture-3/presentation/lecture3.pdf>
- Consider the following materials your reading homework

The data science workflow



Sample size

Number of observations in a dataset (study)

```
len(data)
```

Mean

```
np.mean(data)
```

$$\bar{x} = \frac{1}{n} \left(\sum_{i=1}^n x_i \right) = \frac{x_1 + x_2 + \cdots + x_n}{n}$$

1, 3, 3, **6**, 7, 8, 9

Median

"The median of a set of numbers is the value separating the higher half from the lower half of a data sample, a population, or a probability distribution."

```
np.median(data)
```

$$\text{Median} = \underline{\underline{6}}$$

1, 2, 3, **4**, **5**, 6, 8, 9

$$\begin{aligned}\text{Median} &= (4 + 5) \div 2 \\ &= \underline{\underline{4.5}}\end{aligned}$$

Standard deviation

"...is a measure of the amount of variation of the values of a variable about its mean."

```
np.std(data)
```

$$\sigma = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \mu)^2}, \text{ where } \mu = \frac{1}{N} \sum_{i=1}^N x_i.$$

Correlation coefficient

"The Pearson correlation coefficient measures the linear relationship between two datasets. Like other correlation coefficients, this one varies between -1 and +1 with 0 implying no correlation."

[scipy docs](#)

The correlation coefficient is calculated as follows:

$$r = \frac{\sum(x - m_x)(y - m_y)}{\sqrt{\sum(x - m_x)^2 \sum(y - m_y)^2}}$$

where m_x is the mean of the vector x and m_y is the mean of the vector y.

Descriptive statistics of band gap (E_g) distribution in the Materials Project

Sample size

- 103,217

Mean of E_g

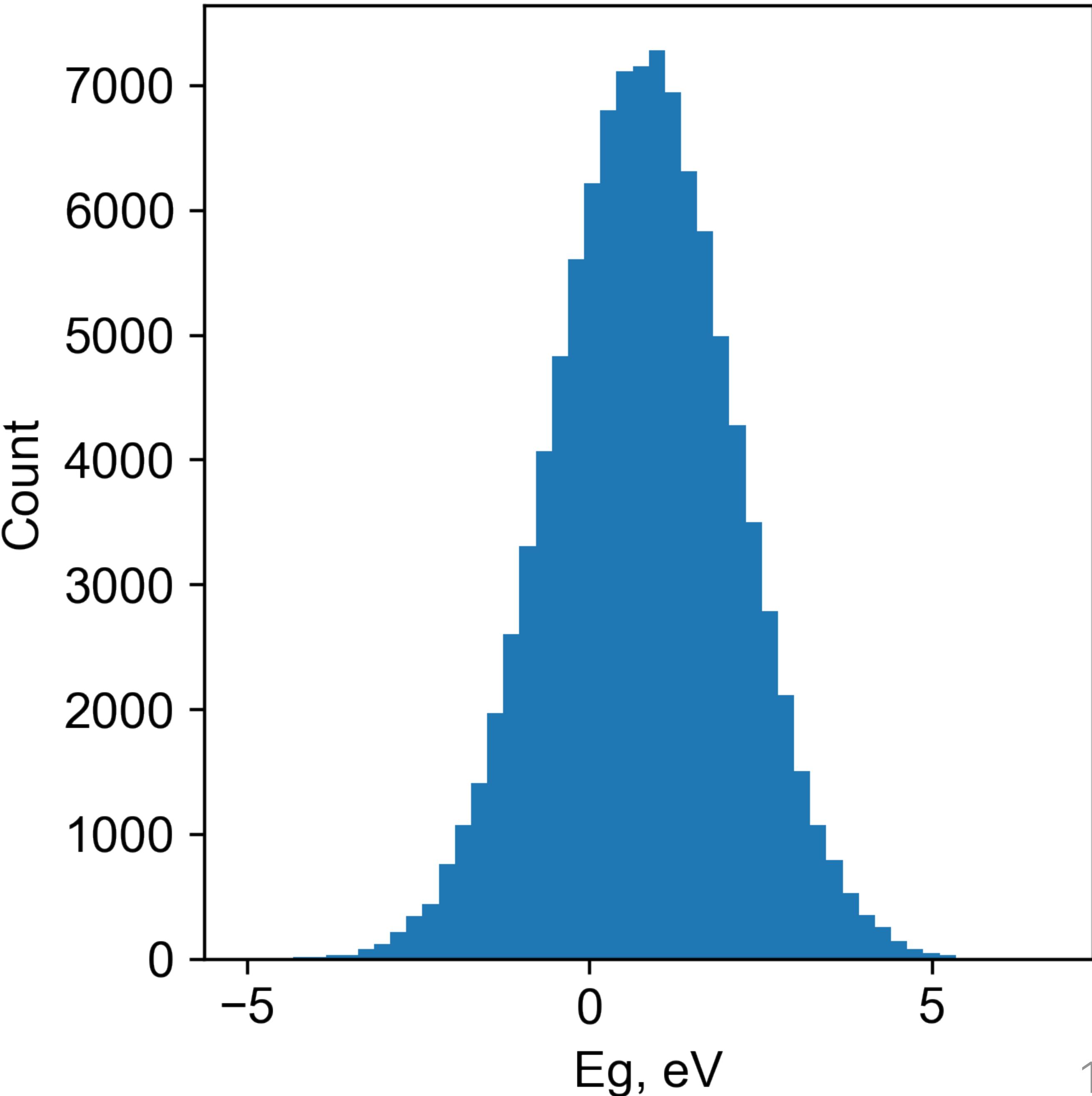
- 0.79 eV

Standard deviation of E_g :

- 1.37 eV

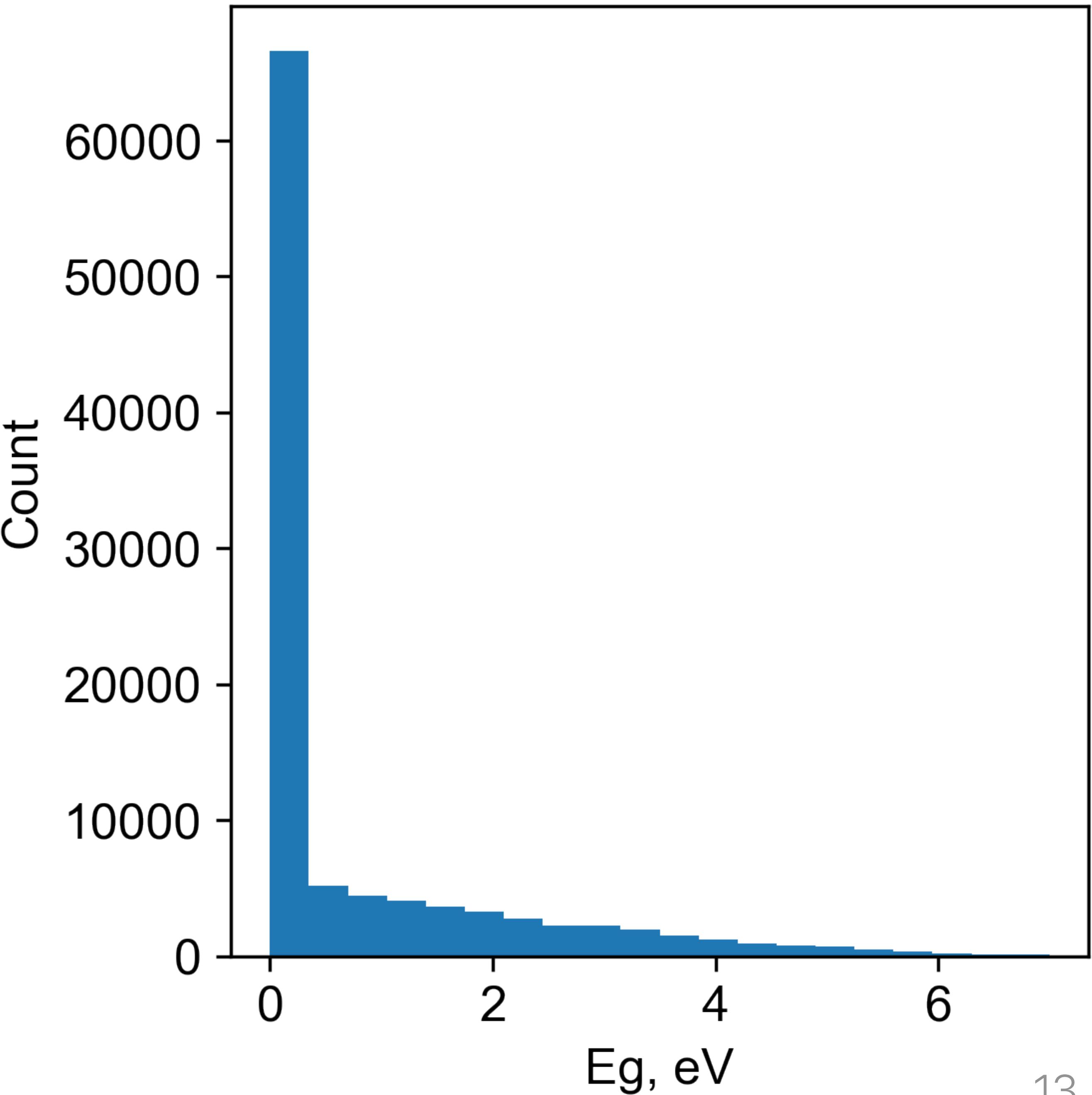
Is it what you expected?

- What wrong with this distribution?



This is the real distribution

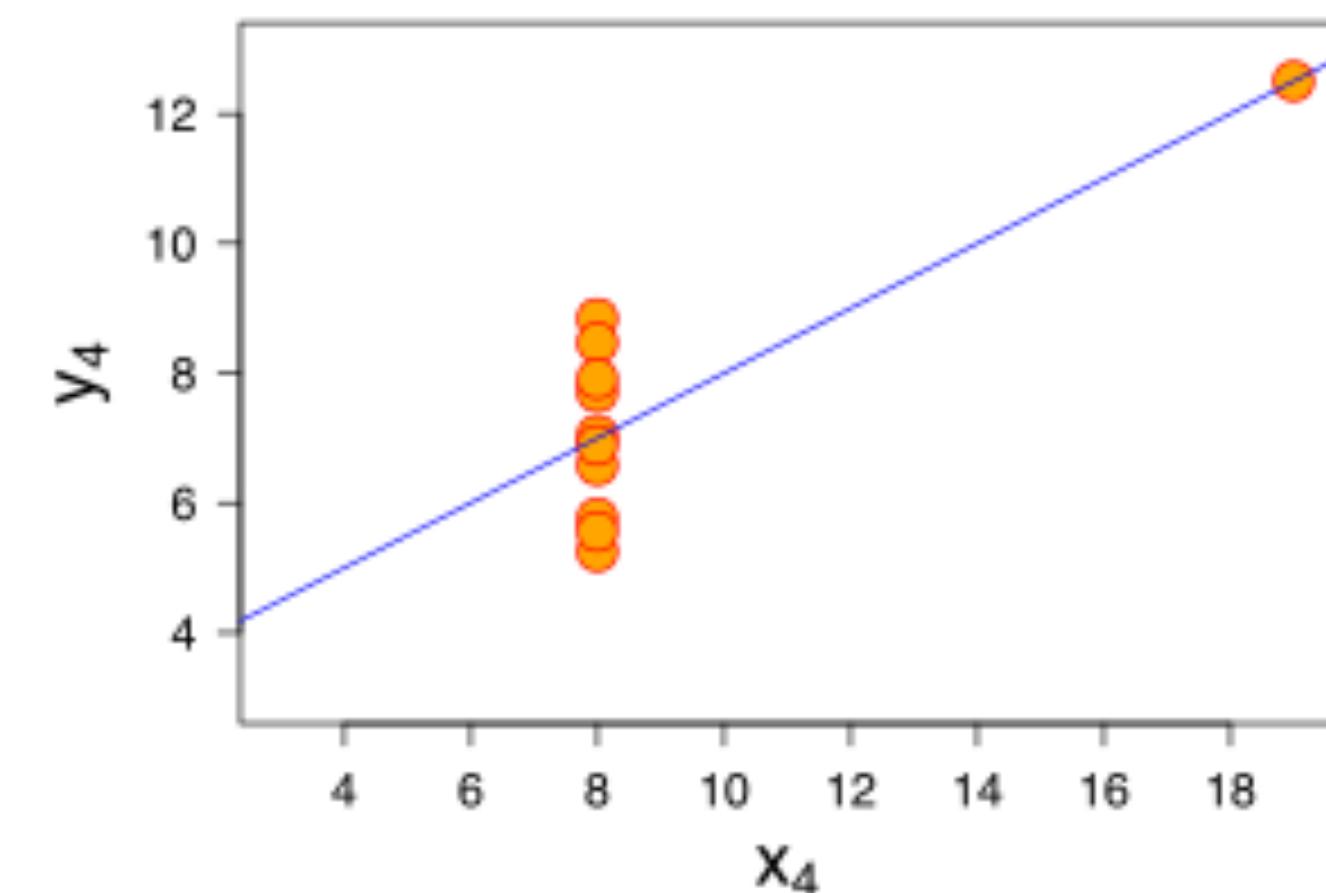
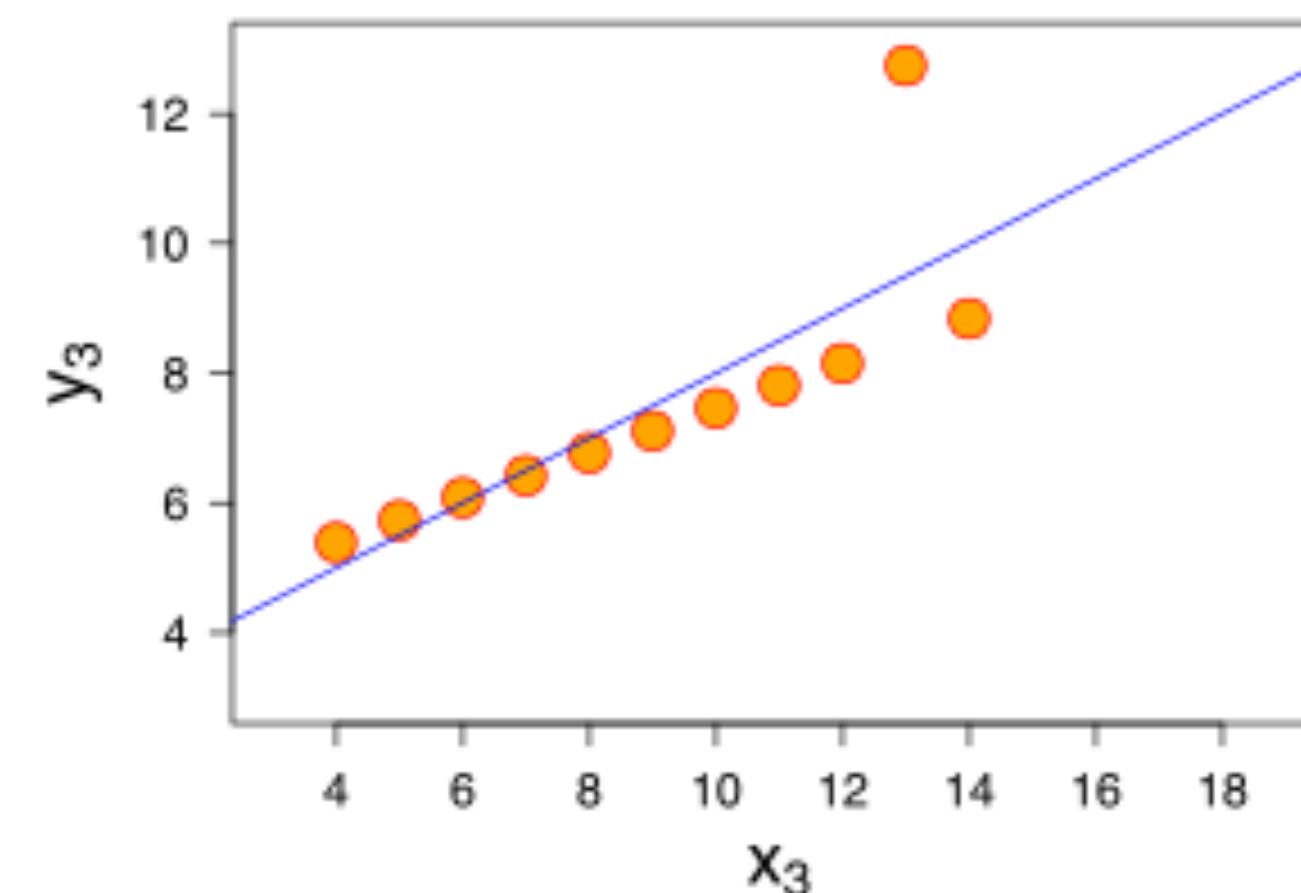
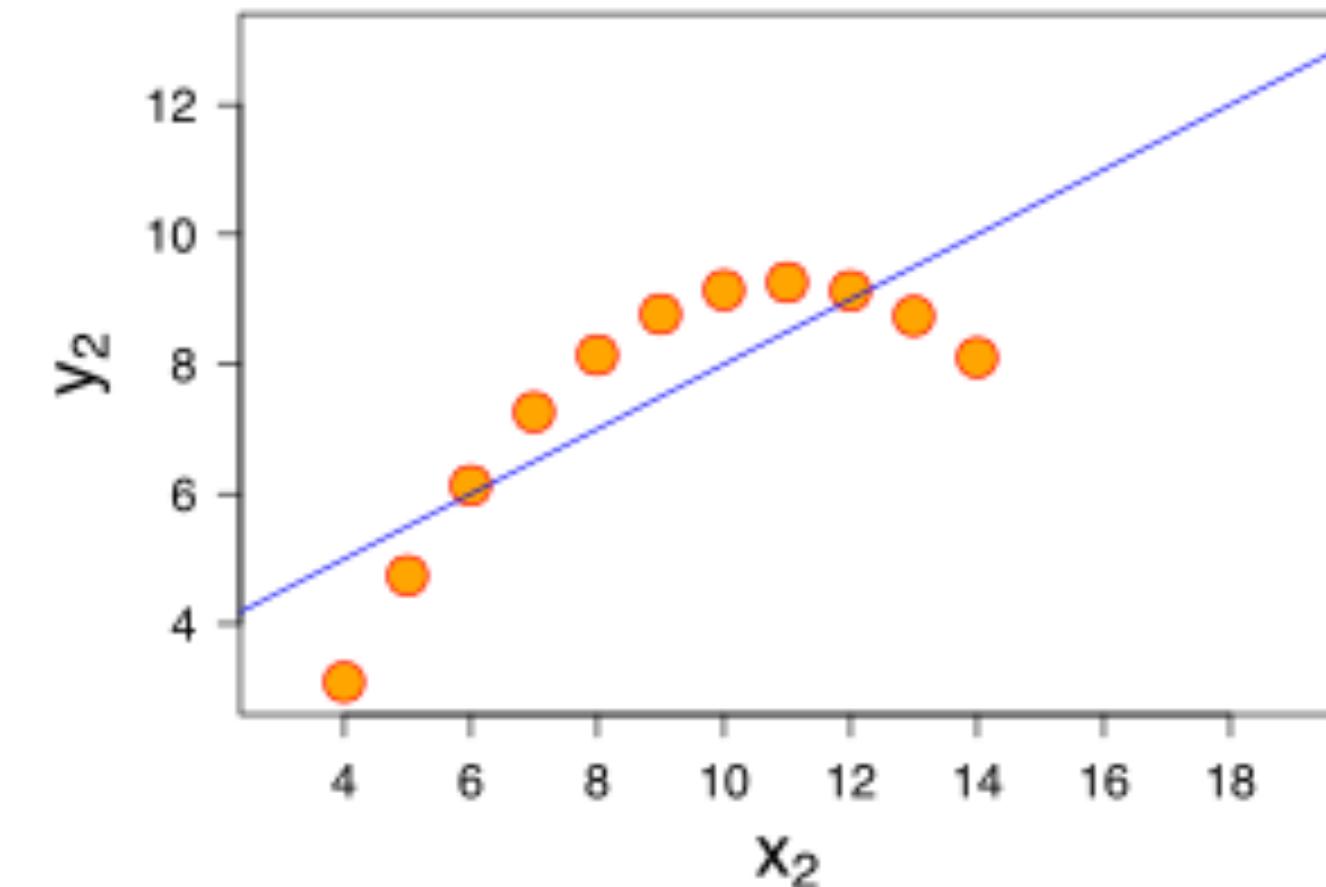
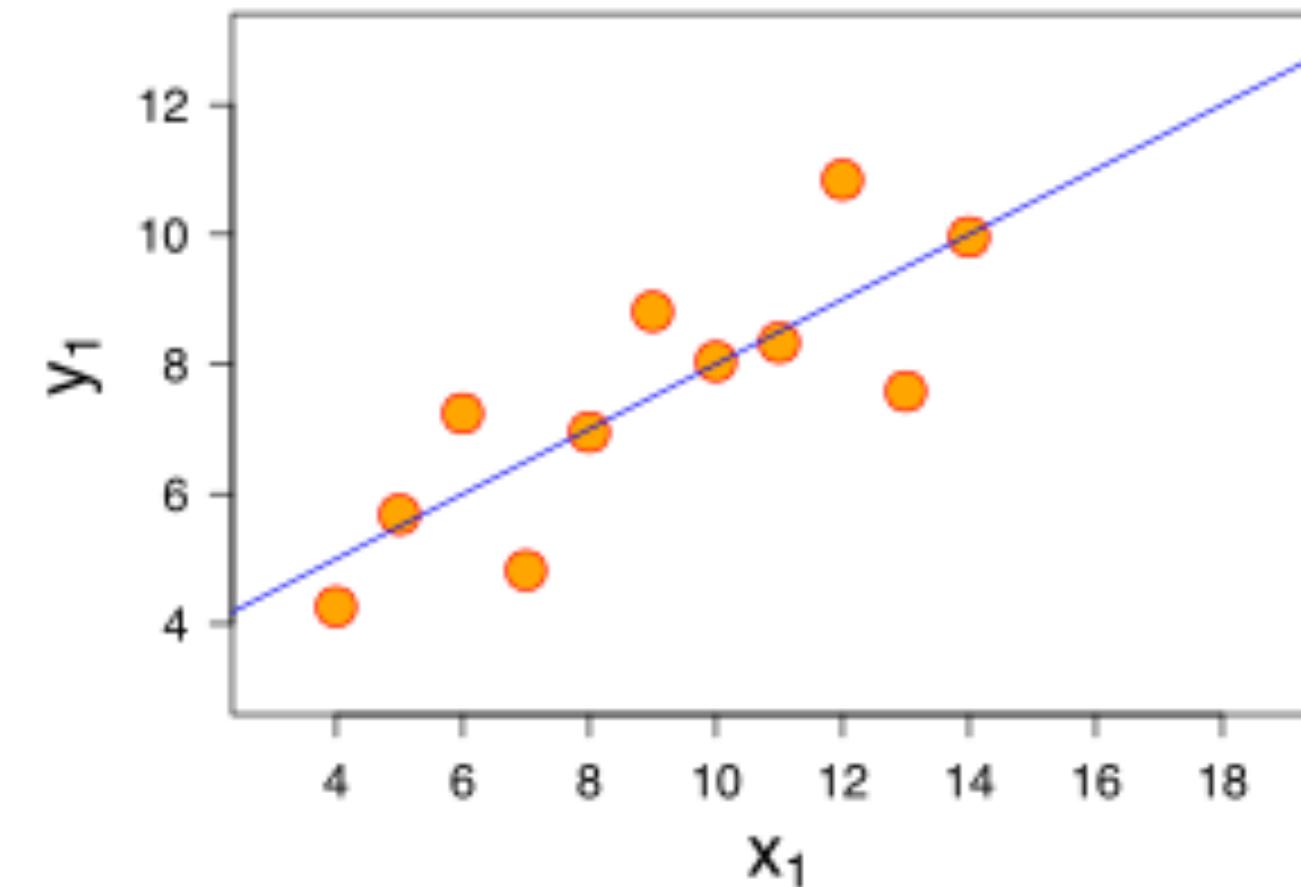
- Metals have a zero E_g
- Median(E_g) = 0.0 says that metals represent at least half of the sample



Why is visual inspection of data important?

- Same descriptive statistics
- Very different distributions

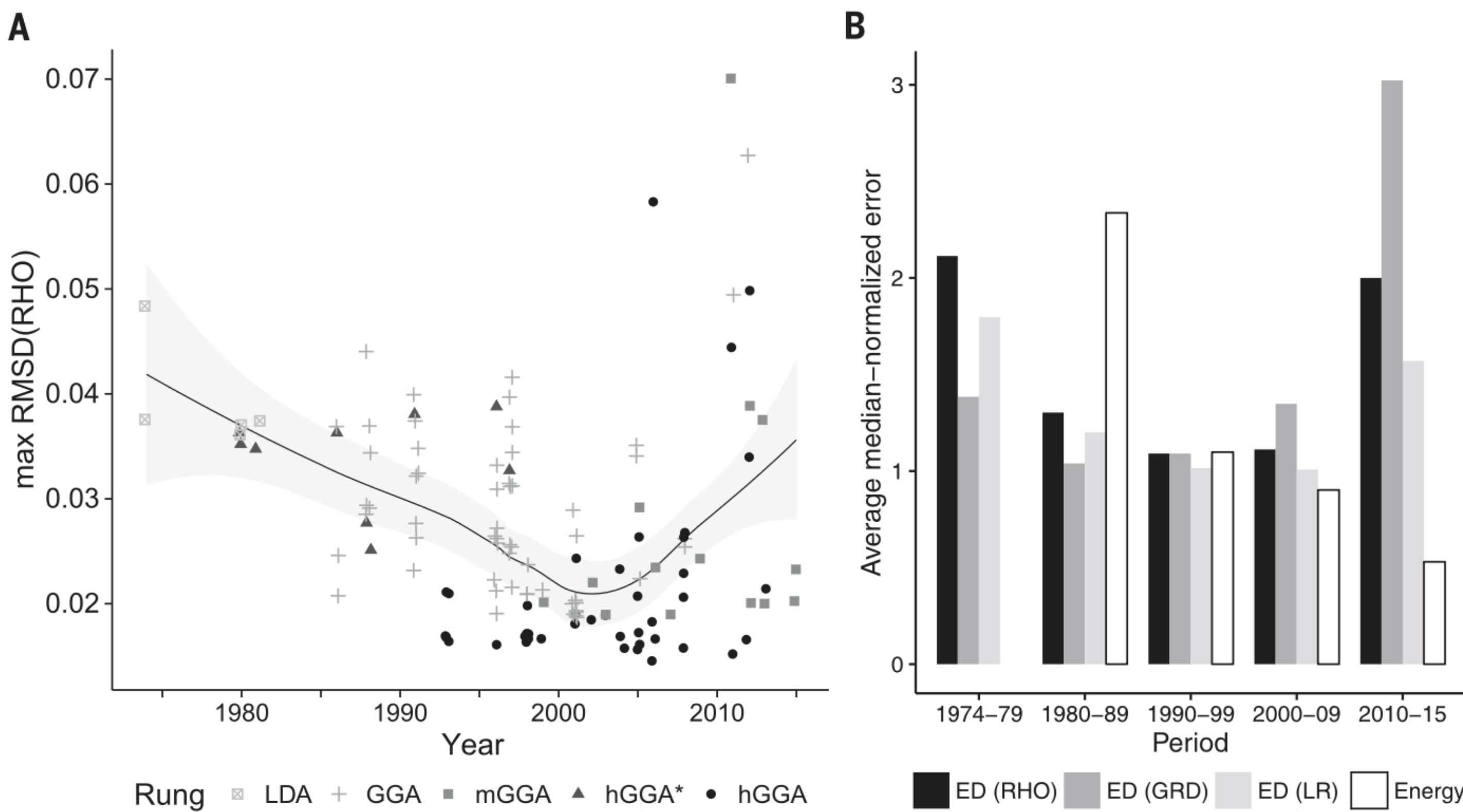
https://en.wikipedia.org/wiki/Anscombe%27s_quartet



Visulaization goals

- Communicate (Explanatory)
- Present data and ideas
- Explain and inform
- Provide evidence and support
- Influence and persuade
- Analyze (Exploratory)
- Explore the data
- Assess a situation
- Determine how to proceed
- Decide what to do

Communicate

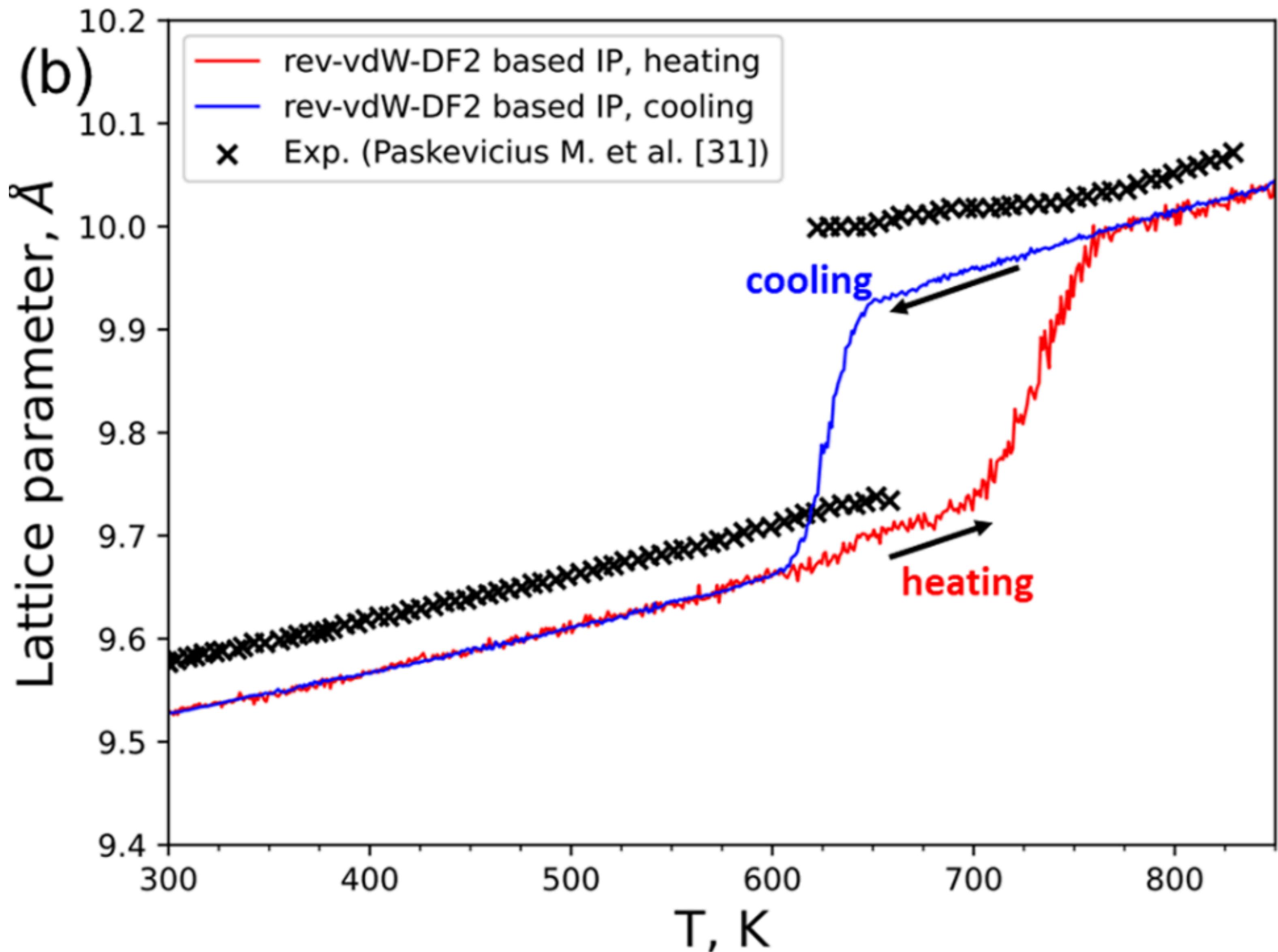


Medvedev *et al.*, *Science* **355**, 49–52 (2017) 6 January 2017

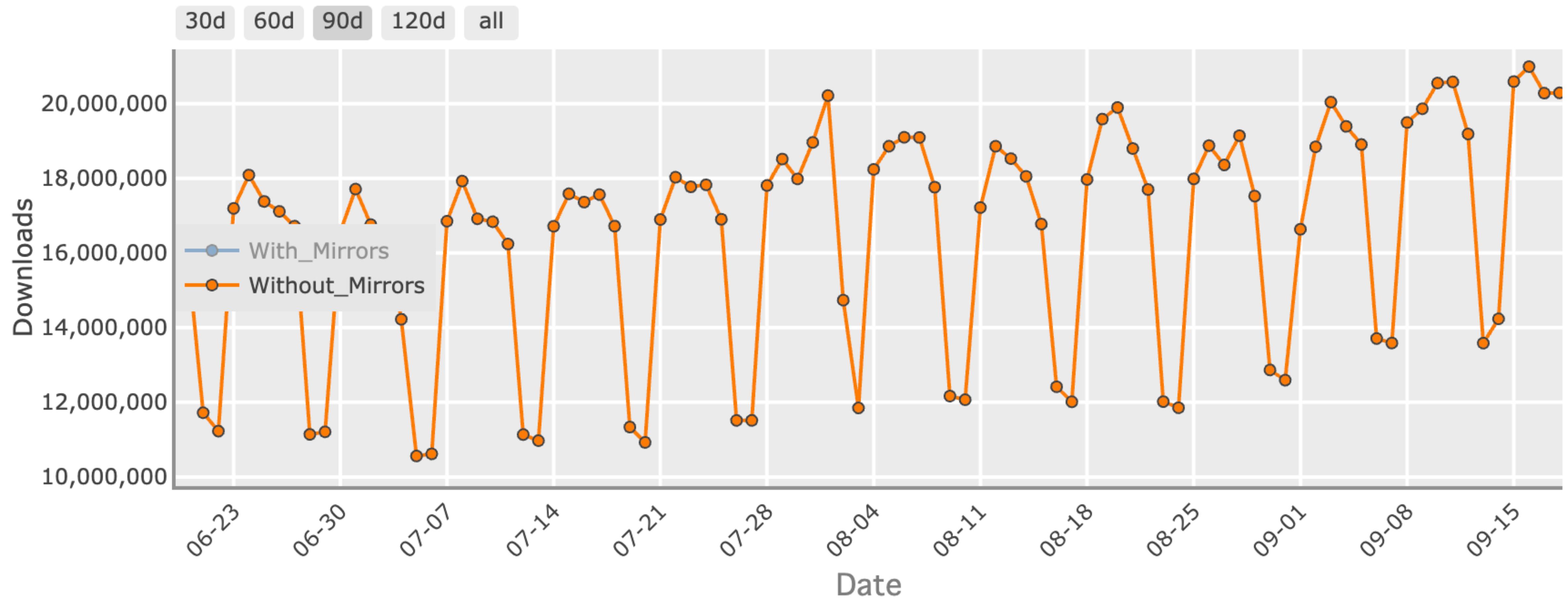
1 of 4

Fig. 1. The historical trends in maximal deviation of the density produced by various DFT methods from the exact one. (A) The line shows the average deviation, with the light gray area denoting its 95% confidence interval; hGGA* denotes 100% exact exchange-based methods. **(B)** The bars denote averages of DFT functionals' median-normalized absolute error for energy [open bars, Truhlar's data (4)] and electron density with its derivatives (solid bars, this work) per publication decade.

Explore



Numpy daily downloads statistics: What are these periods?



Exploratory data analysis pipeline

- Build data
- Clean data
- Explore global features
- Explore group features

Build (read) data in a structured format

- Pandas DataFrame
- One row per variable

```
df = pd.read_csv("eg_data.csv")
```

Clean the data

- Outliers
- NaNs (missing values)
- Constant rows
- Duplicates

```
df.dropna()
```

- Plus visual support: histograms, scatter plots, box plots, etc

Study the global summary statistics

```
df.describe()
```

```
df.aggregate(  
    {  
        "column_name": ["min", "max", "median", "skew"]  
    }  
)
```

Plus visual support: histograms, scatter plots, box plots, etc

Study the summary statistics of the subgroups

```
df[["bandgap", "chemsys"]].groupby("chemsys").mean()
```

Plus visual support: histograms, scatter plots, box plots, bar plots, etc

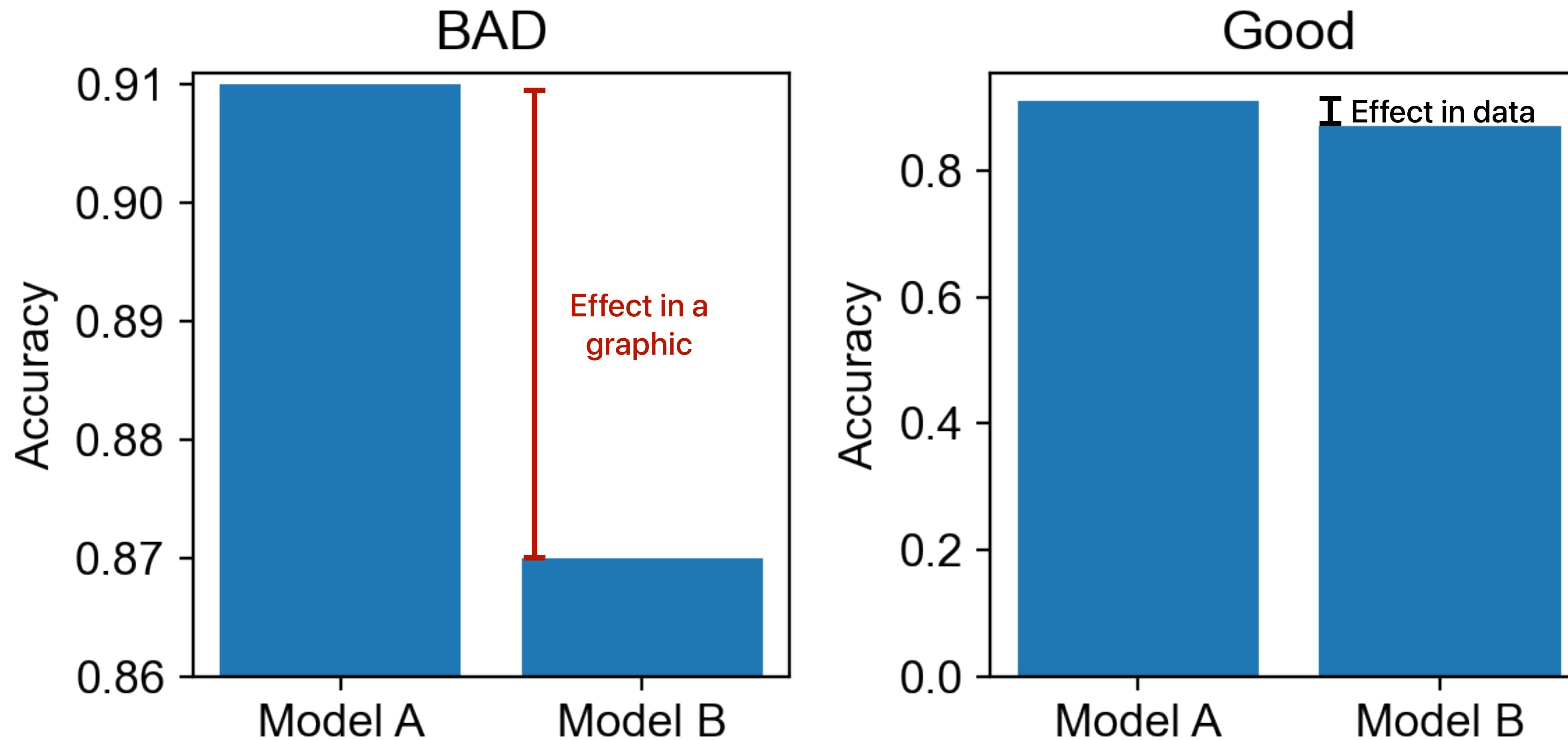
Some principles for effective EDA

Avoid misleading graphs

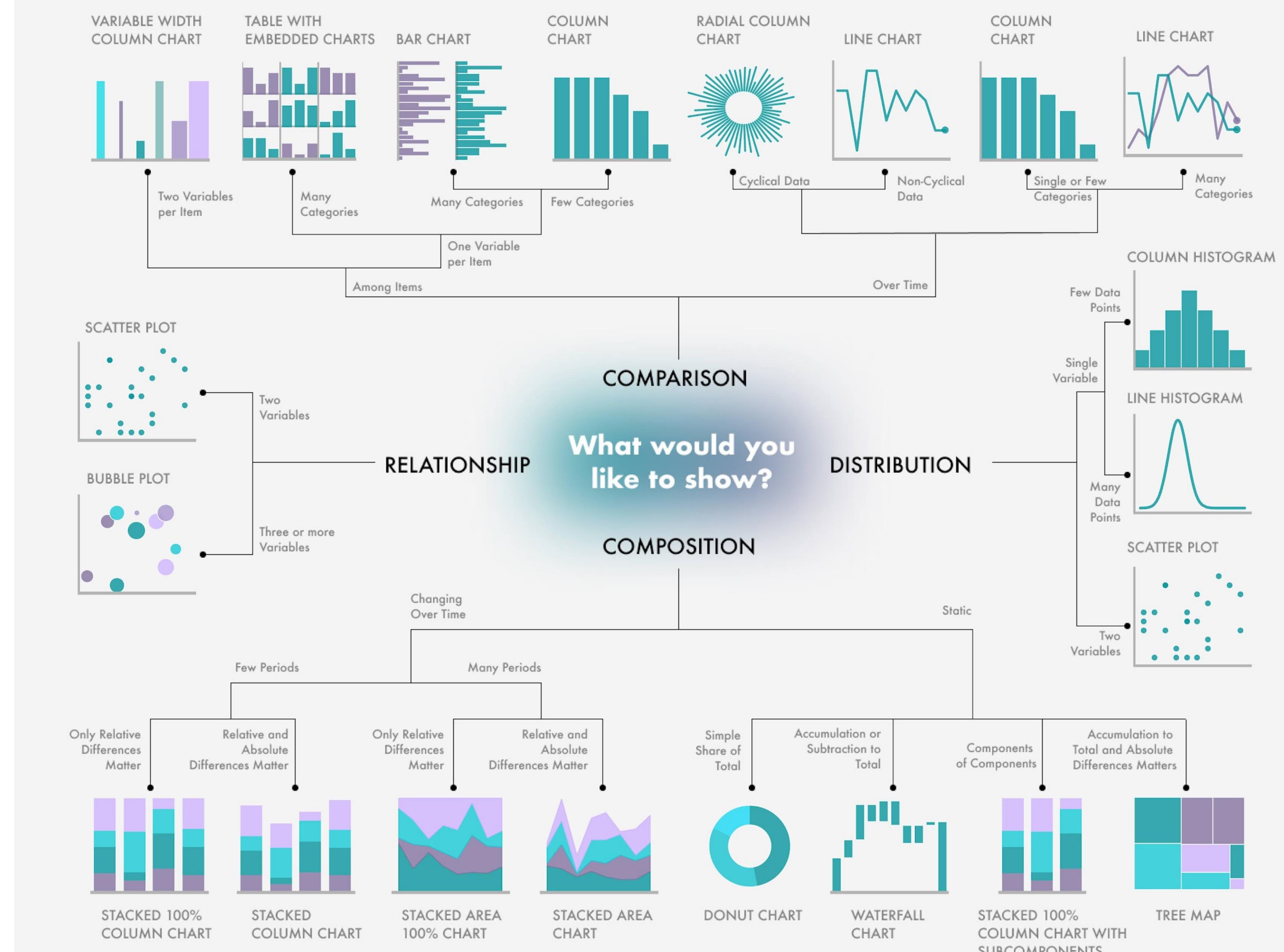
- Do not distort scales
- Do not truncate graph when comparing the data
 - or indicate the truncation
- Avoid 3D charts
- Do not change y(or x)-axis maximum
- Aspect ratio determines the perception of steepness in slope
 - be proportional

Have a look at this page: https://en.wikipedia.org/wiki/Misleading_graph

$$\text{Lie factor} = \frac{\text{Effect in a graphic}}{\text{Effect in data}}$$



Use the right display

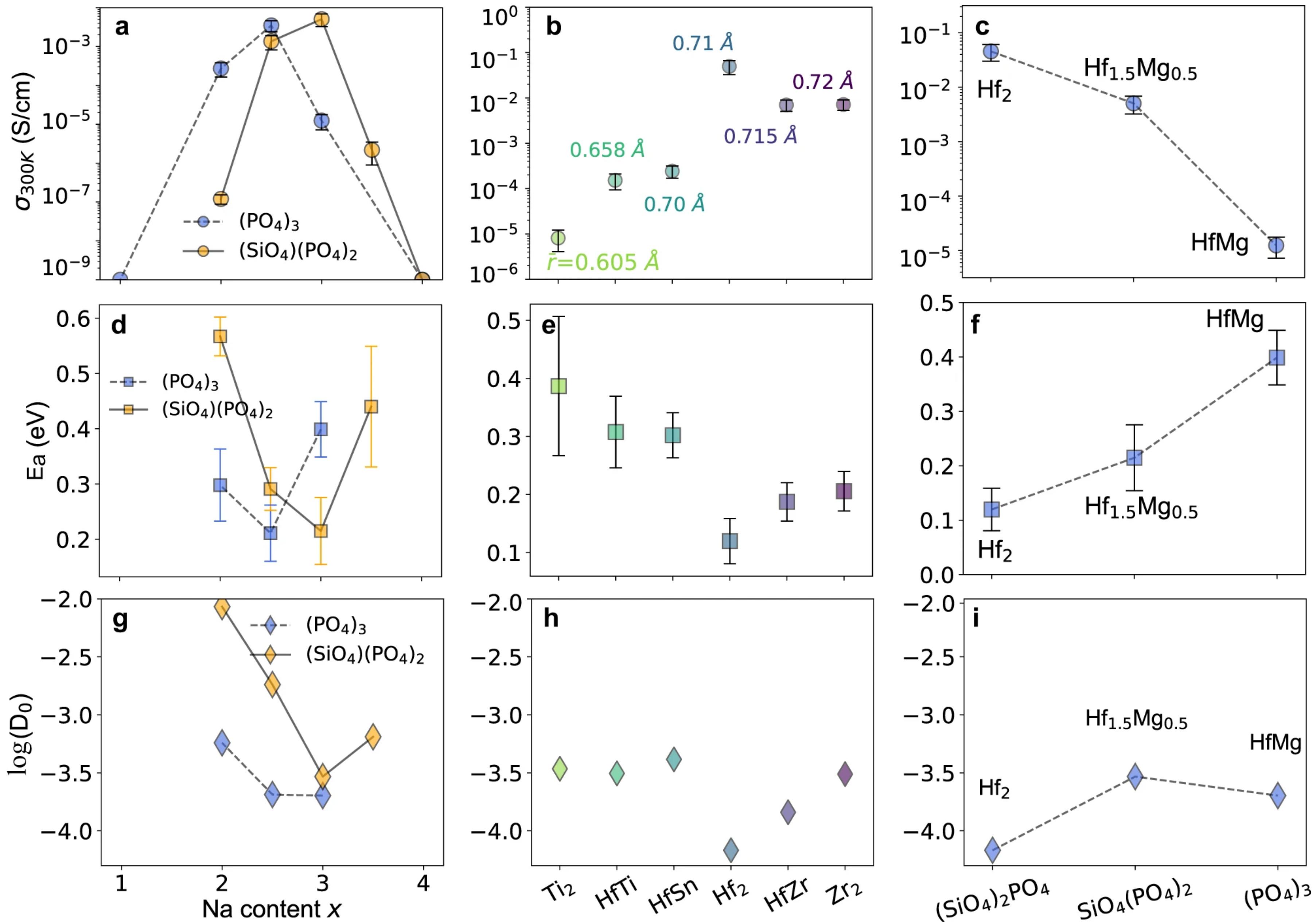


Correlations

- Scatter plot,
correlation matrix

Is it a good graph?

Why?

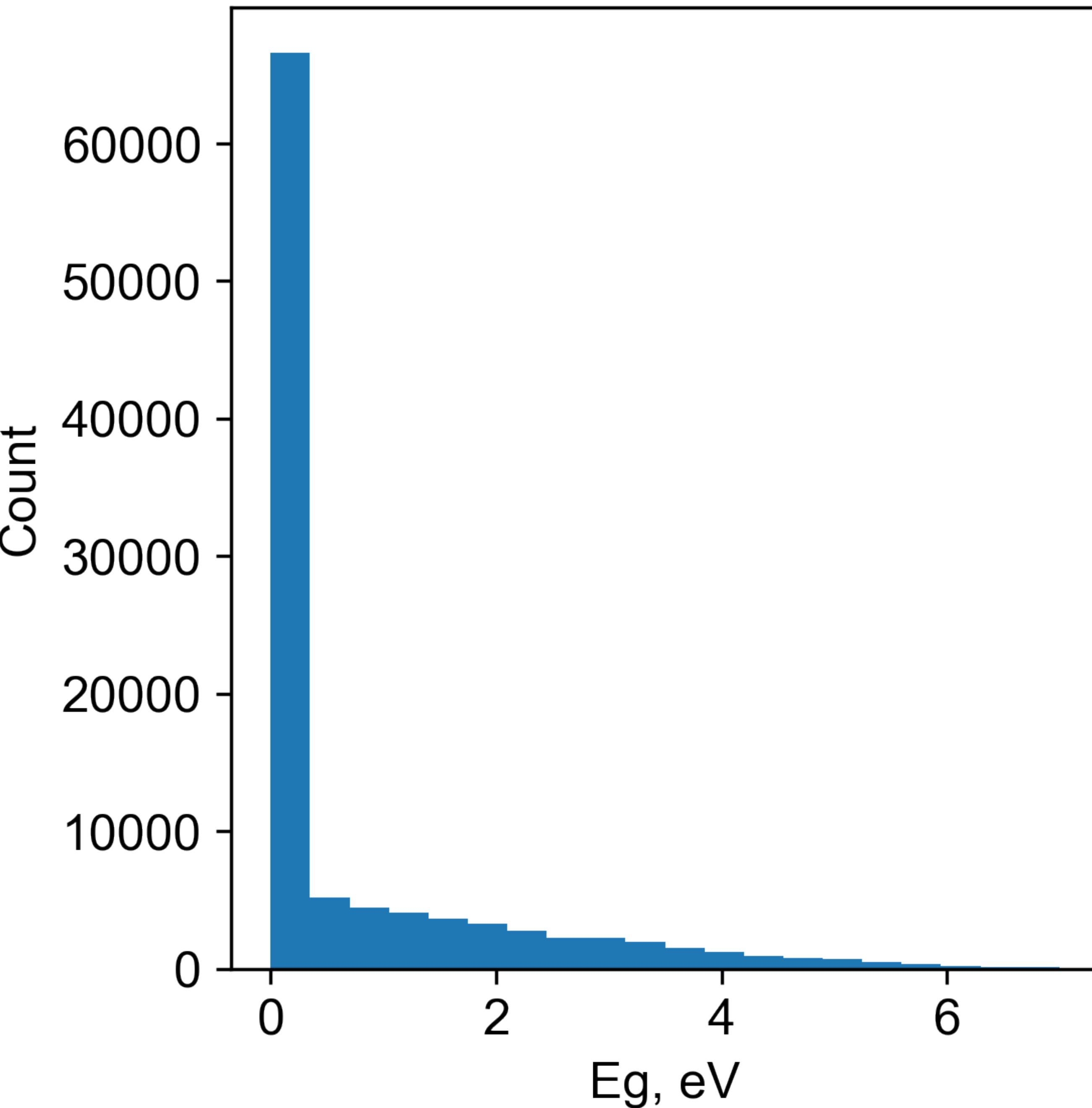


Distribution

- Histogram,
density plot

Is it a good graph?

Why?

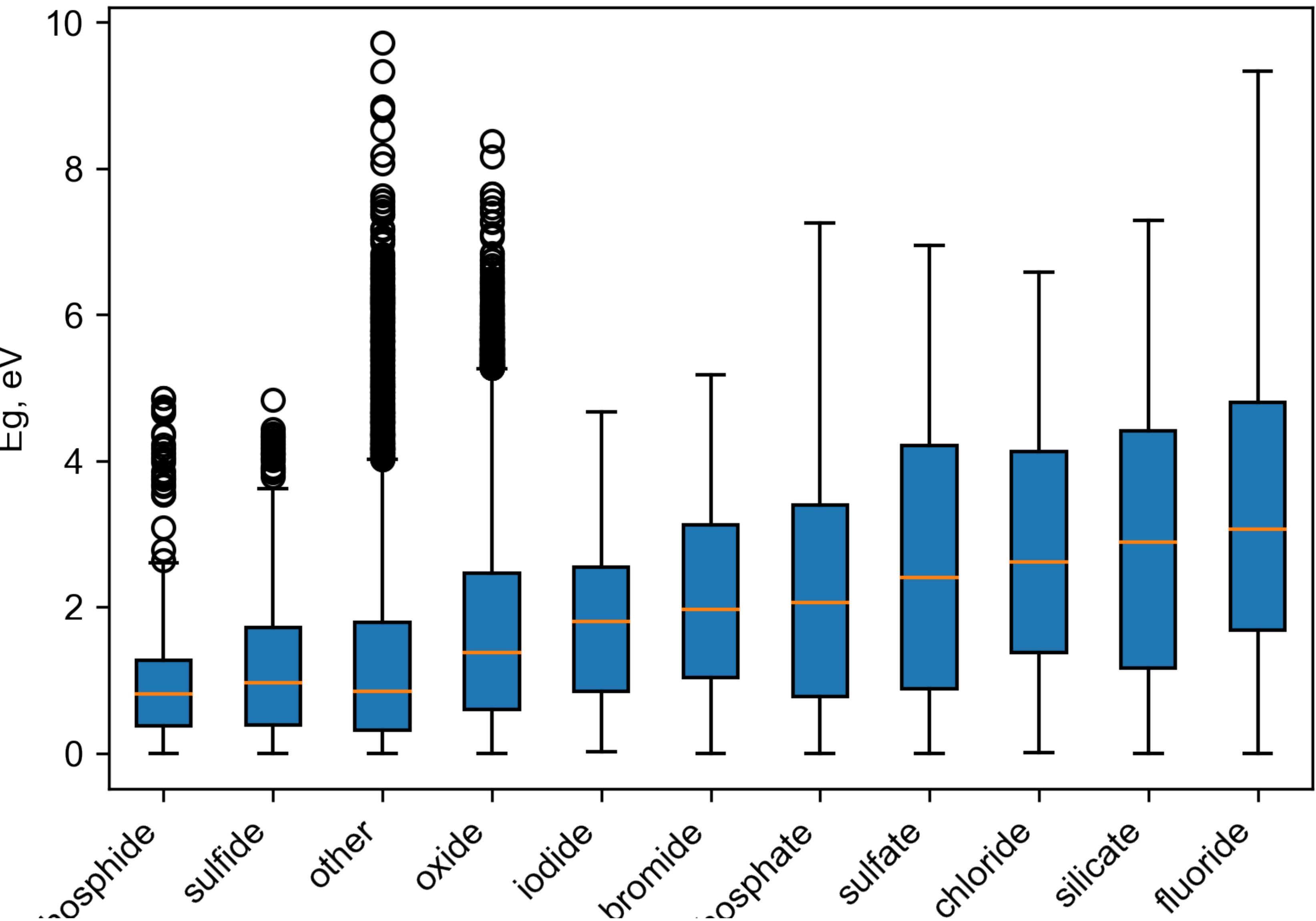


Comparison

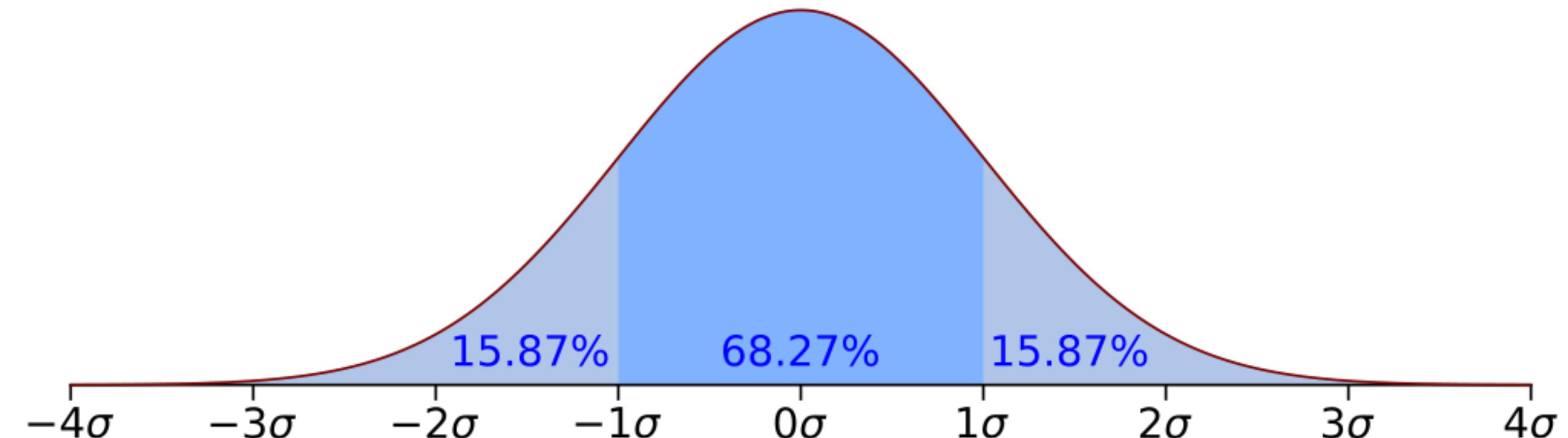
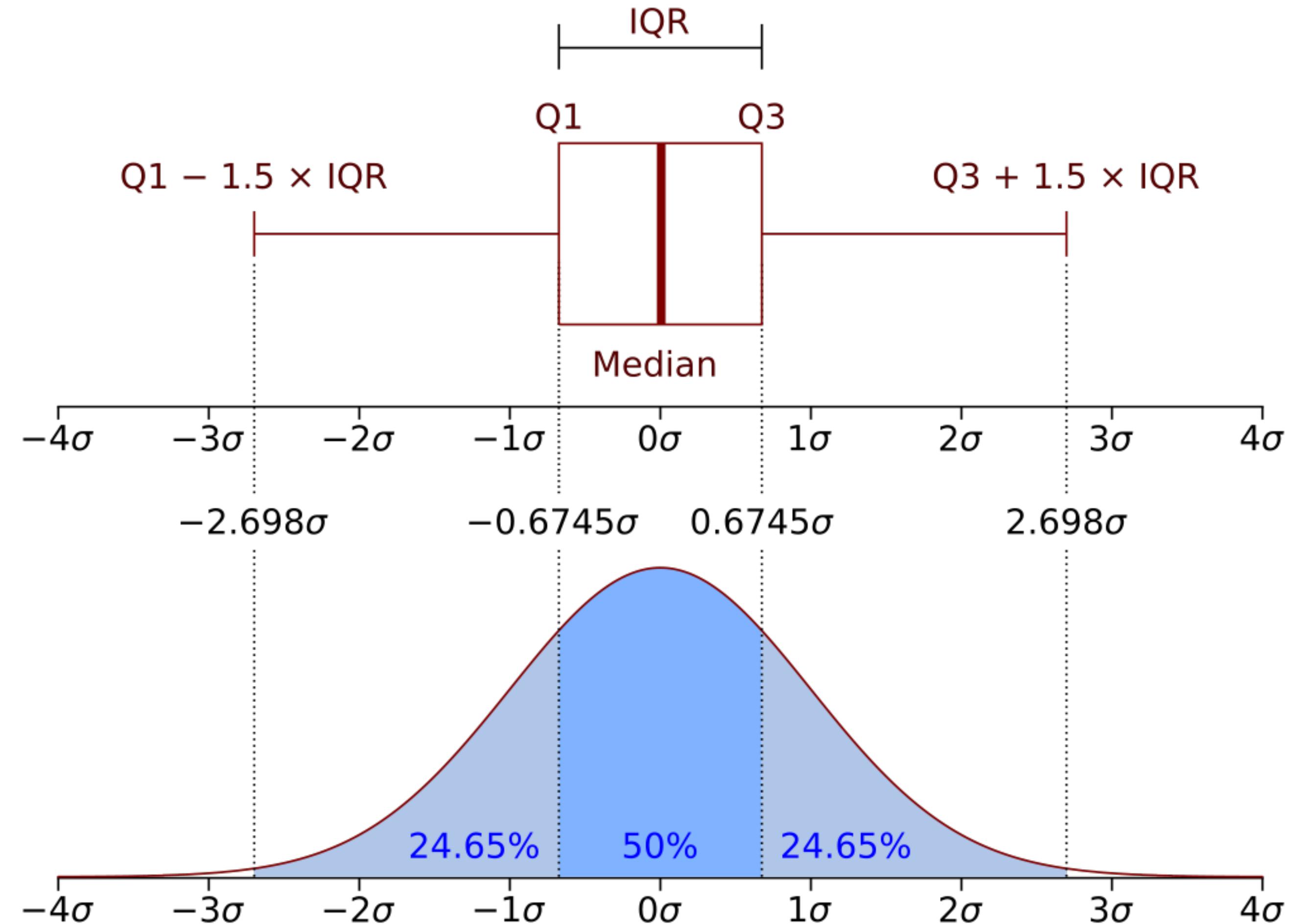
- Bar plot, box plot

Is it a good graph?

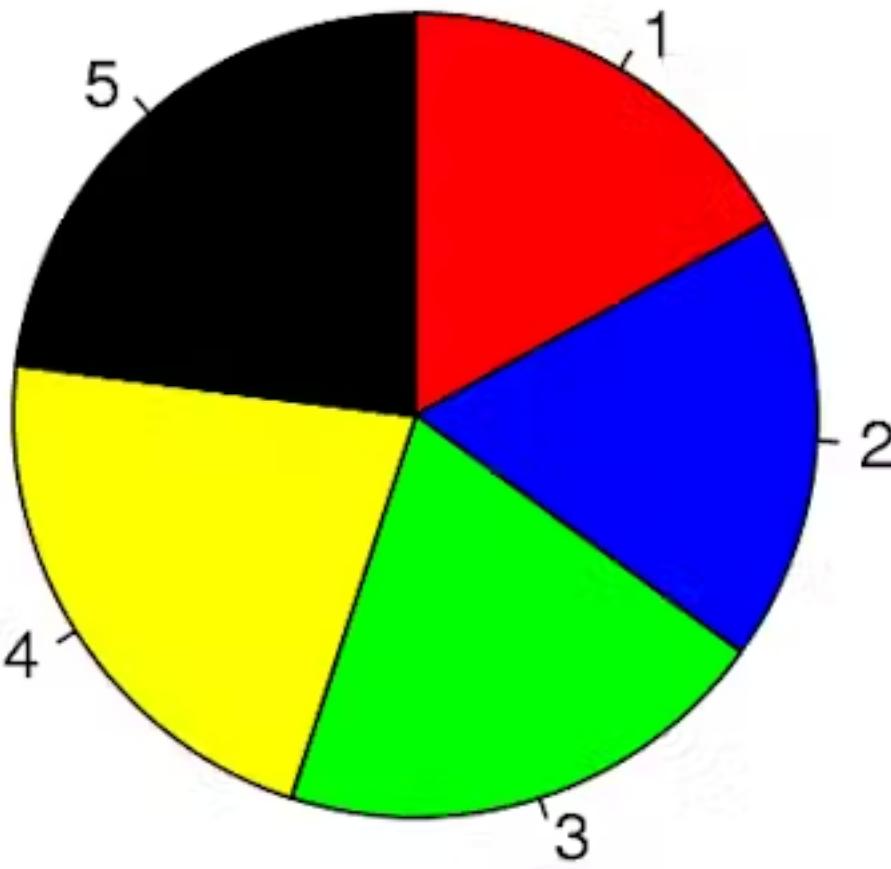
Why?



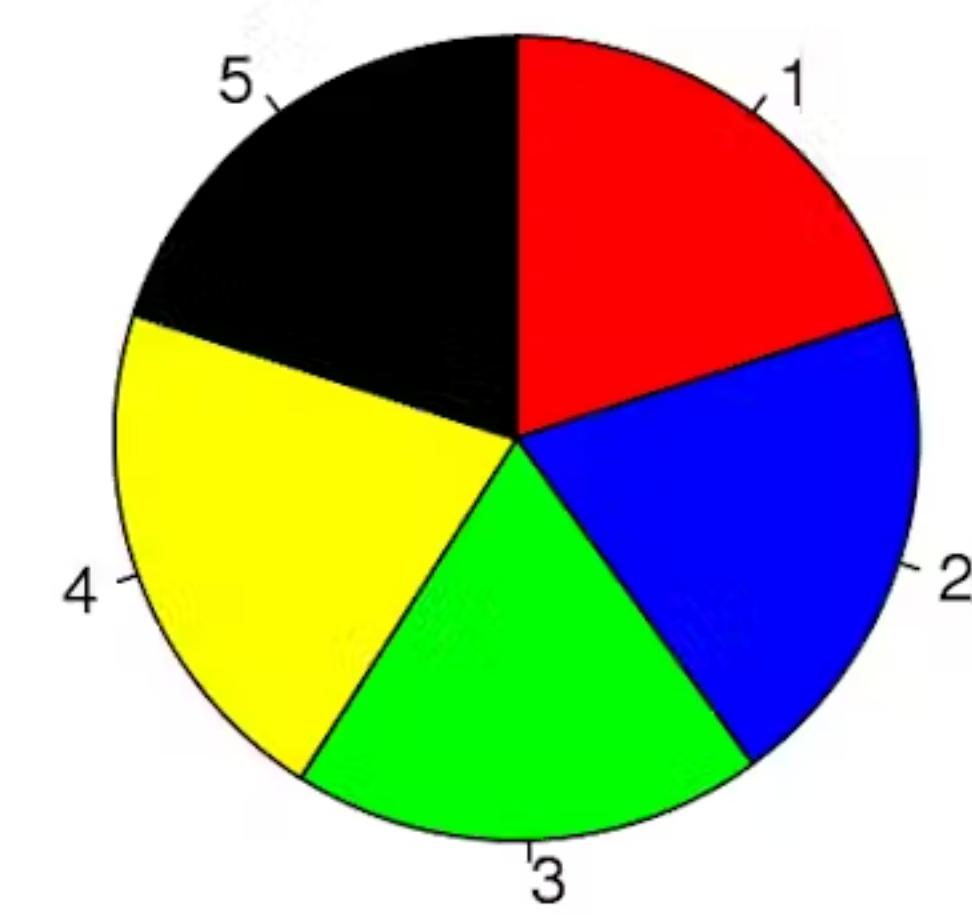
Box plot



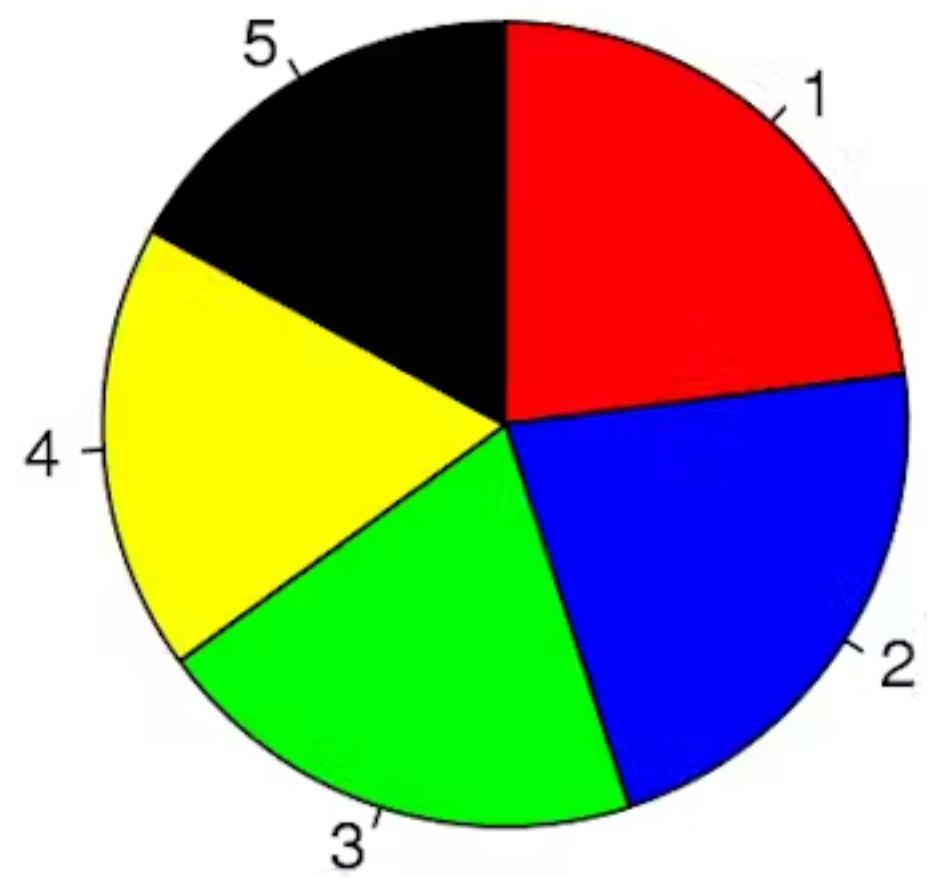
A



B

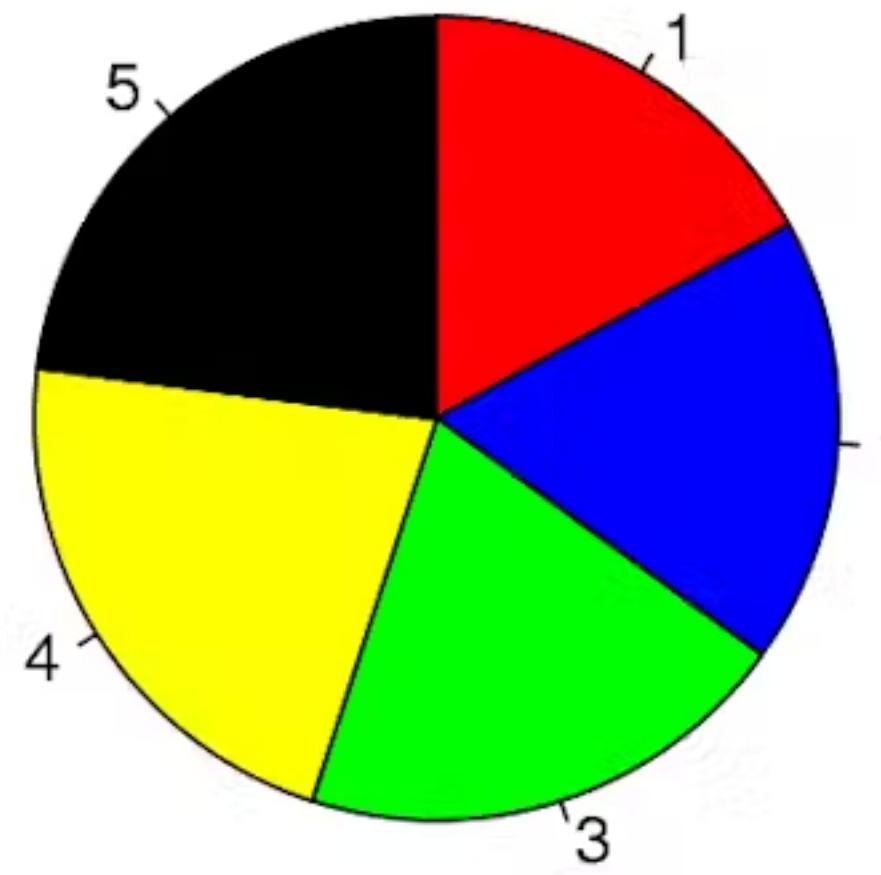


C

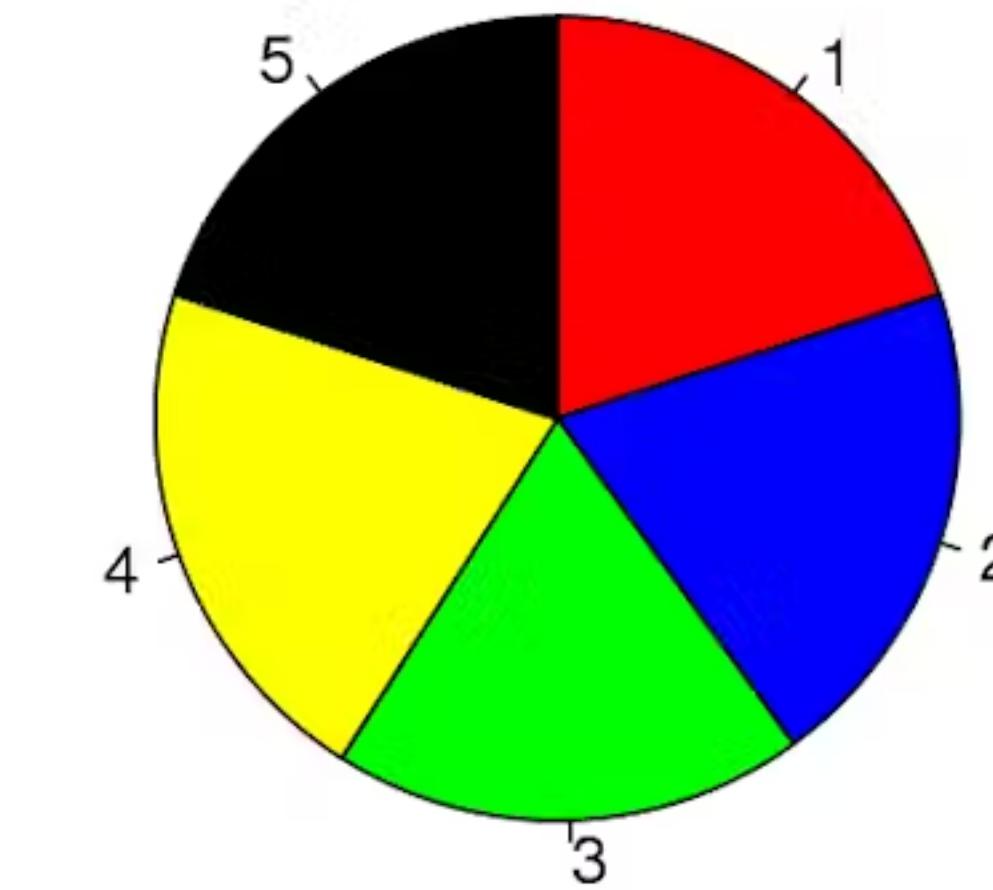


Don't use pie charts!

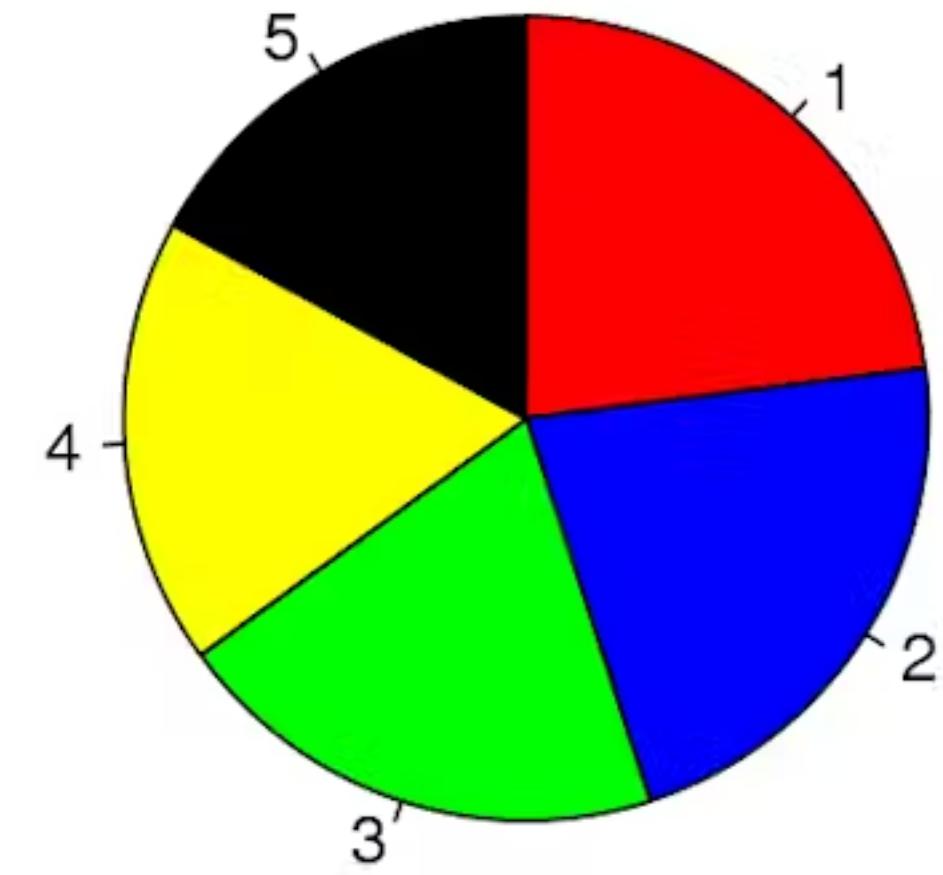
A



B

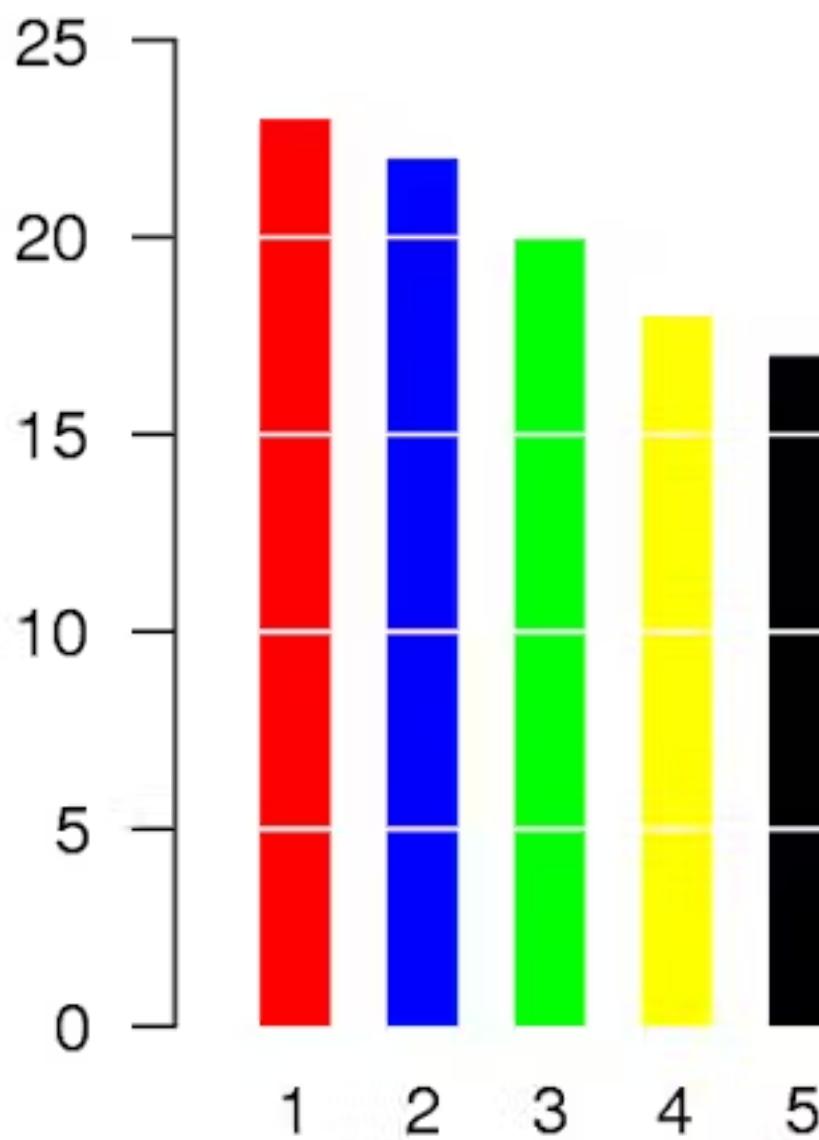
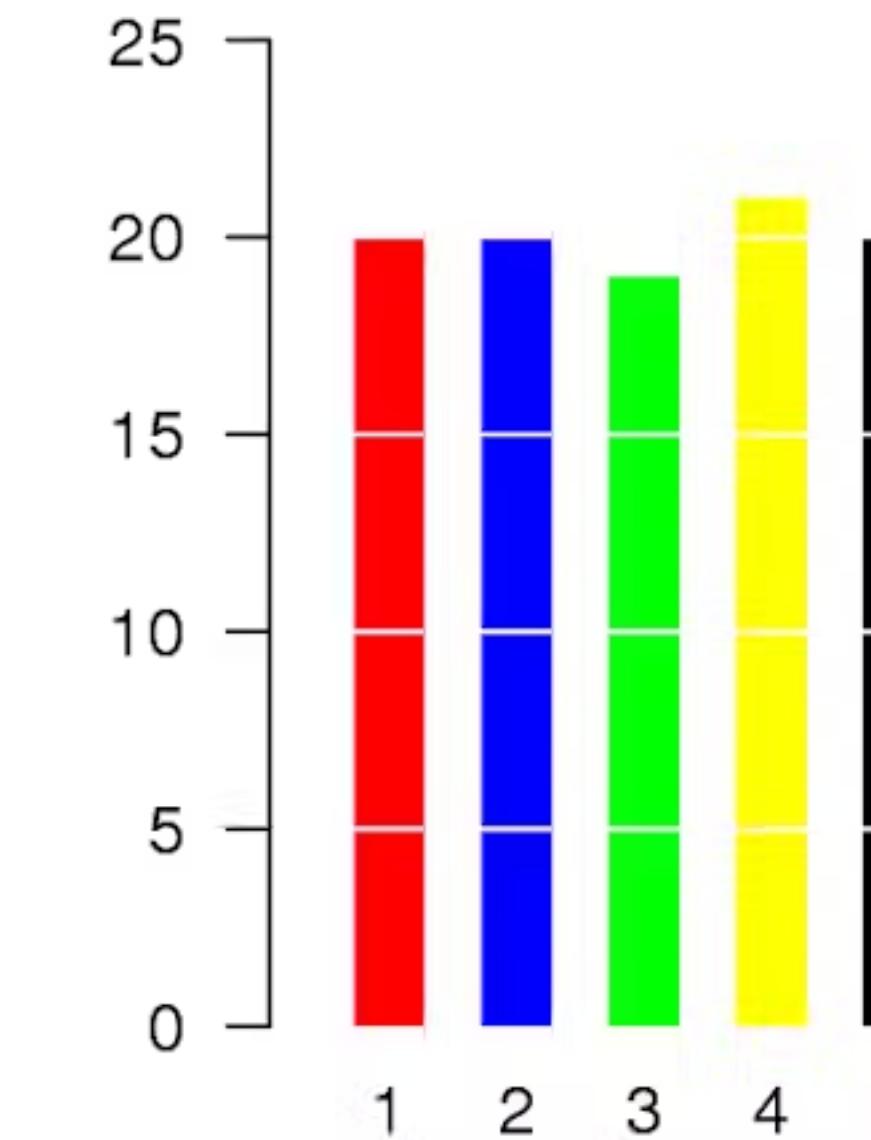
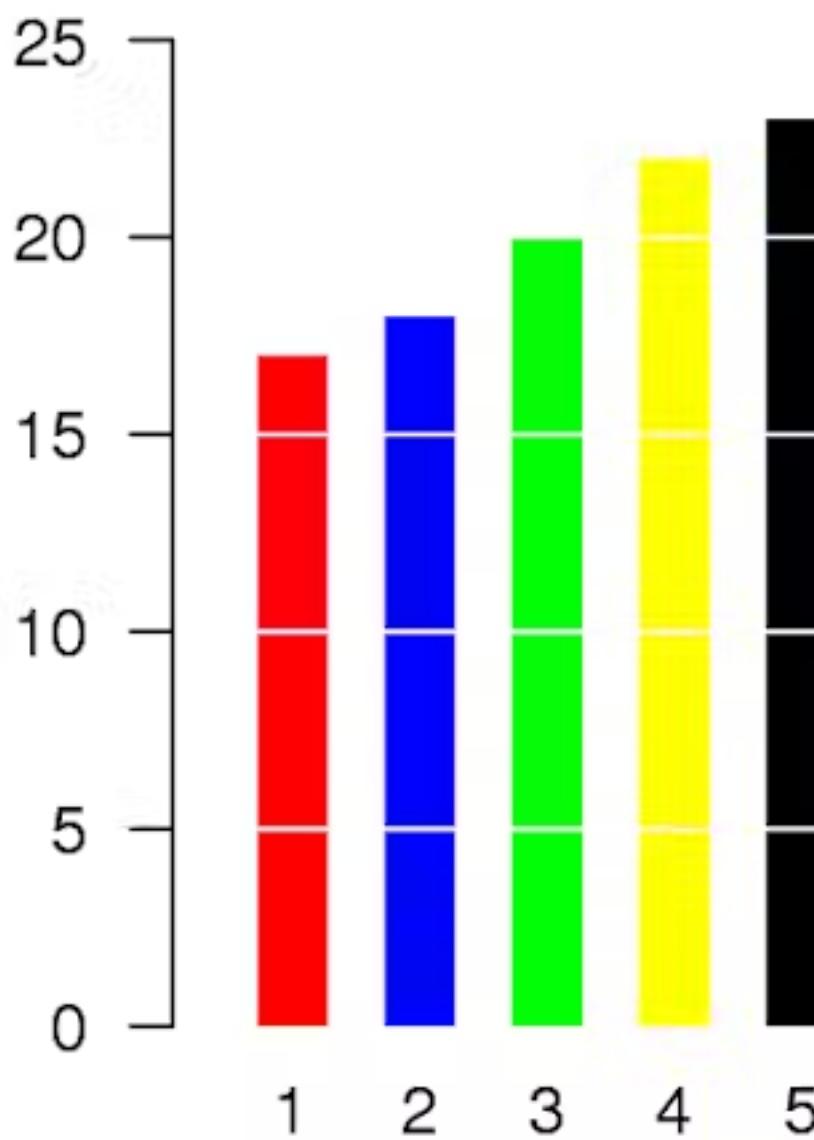


C



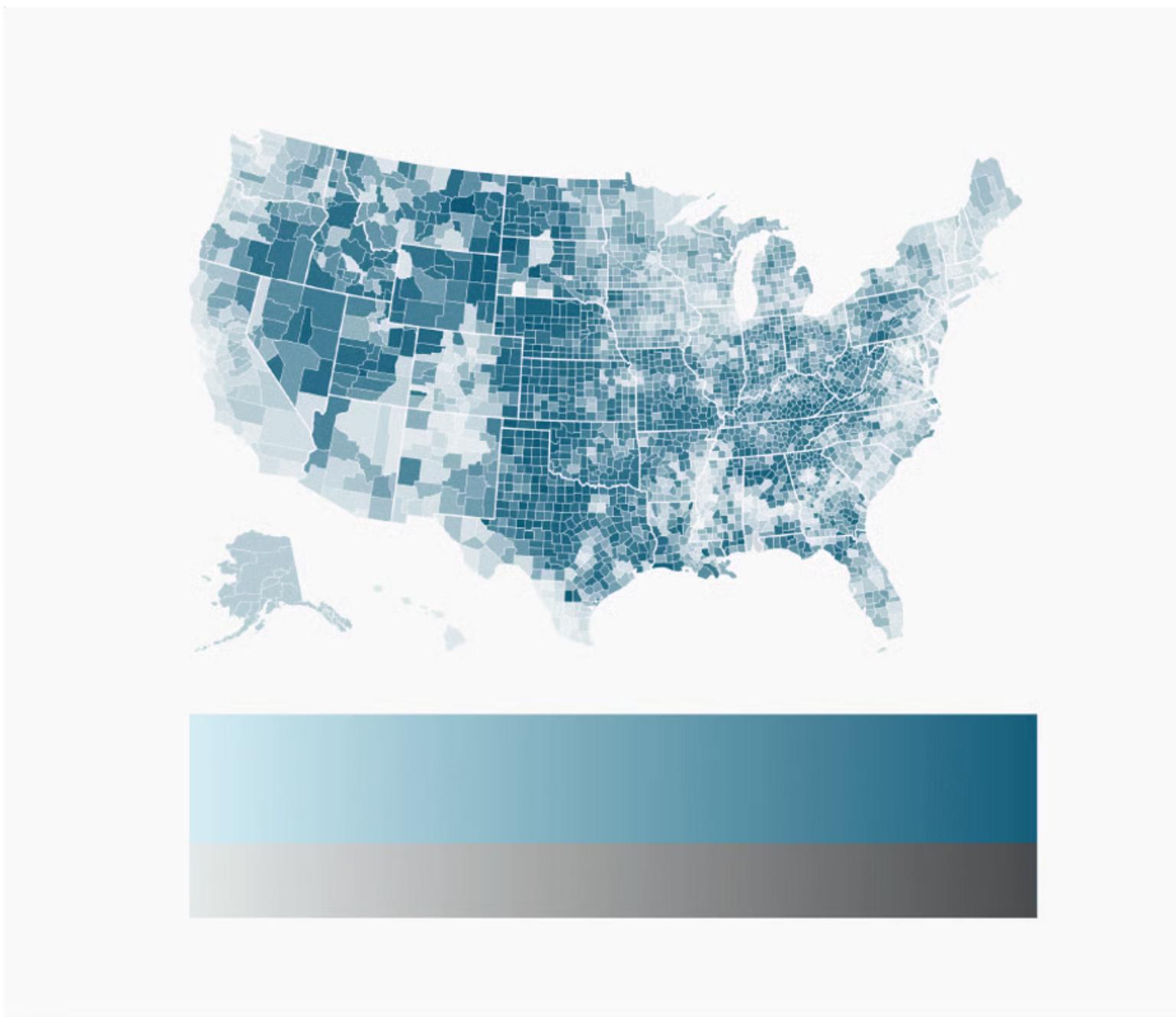
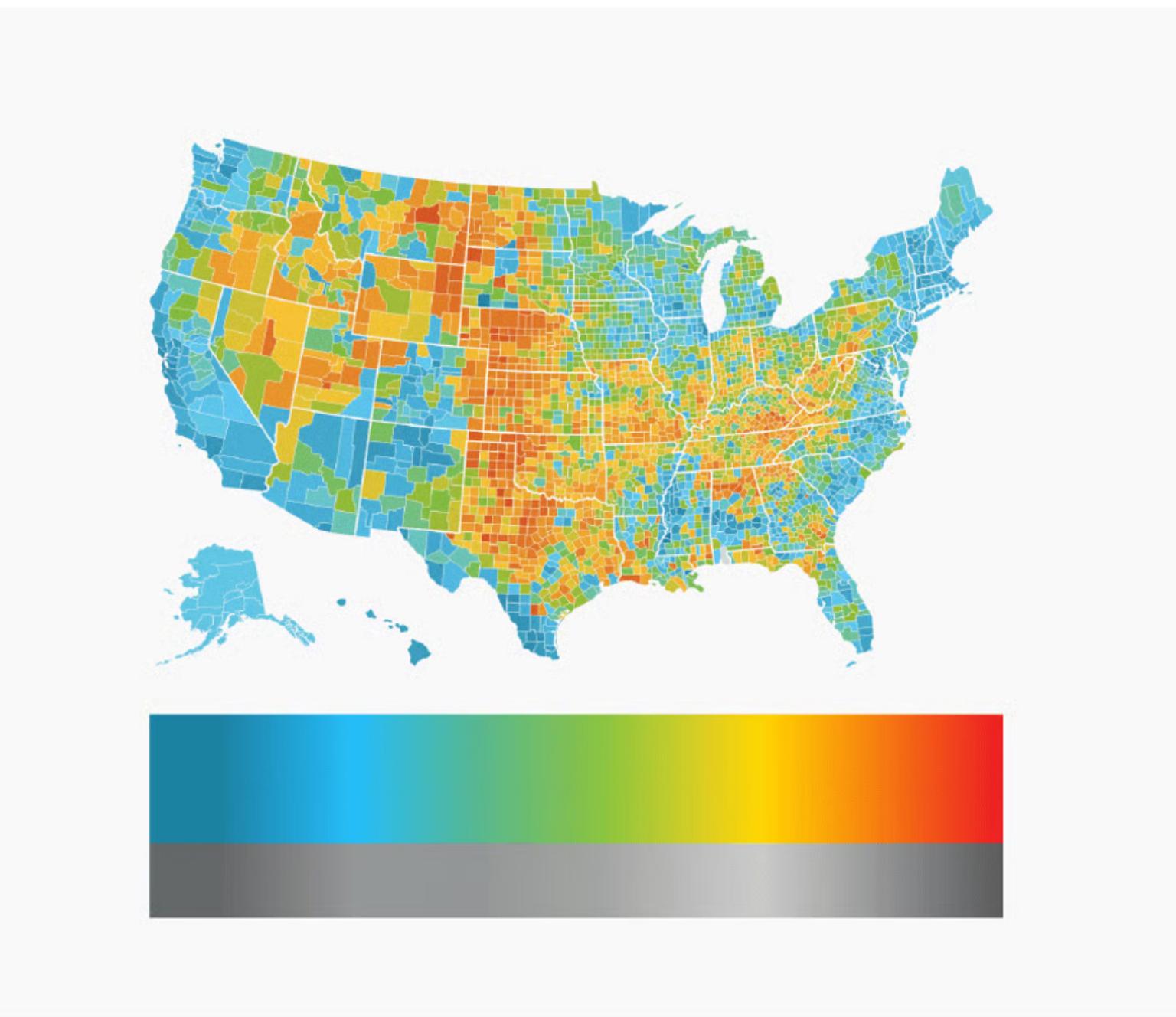
Don't use pie charts!

Bar plots are easier to compare



Use color

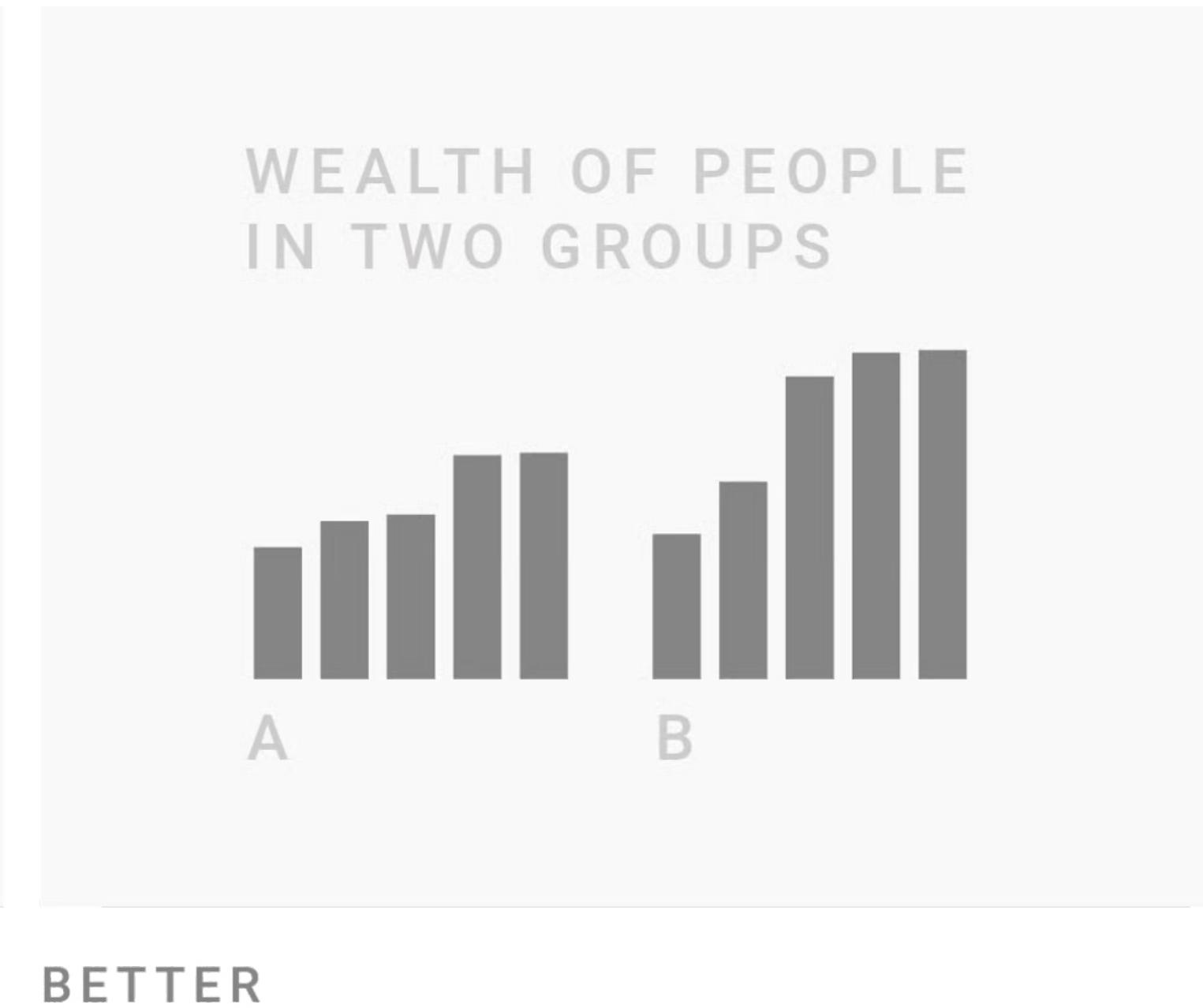
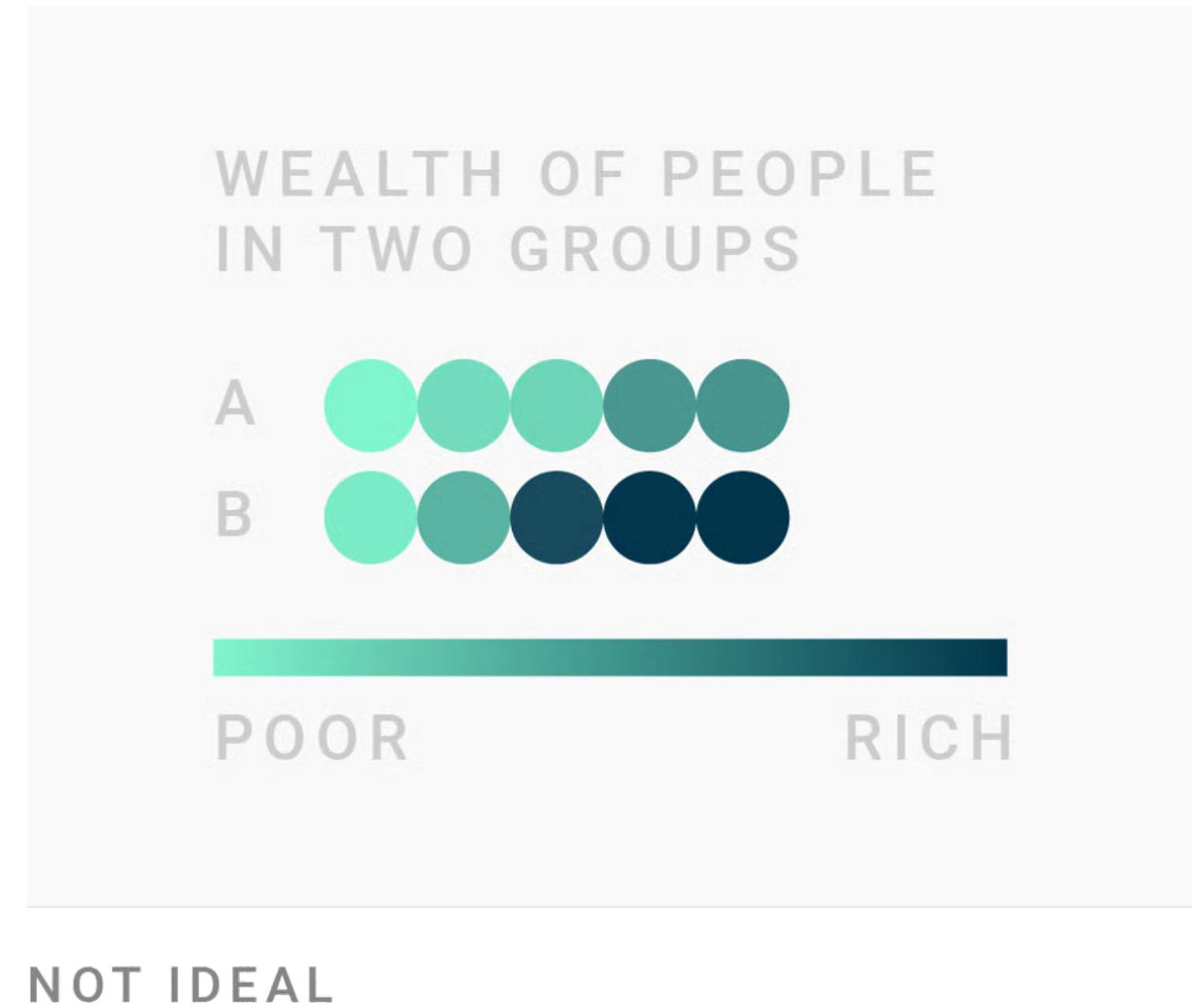
Have a look at this page: [https://
www.datawrapper.de/blog/colors](https://www.datawrapper.de/blog/colors)



**But consider a better alternative
if possible**

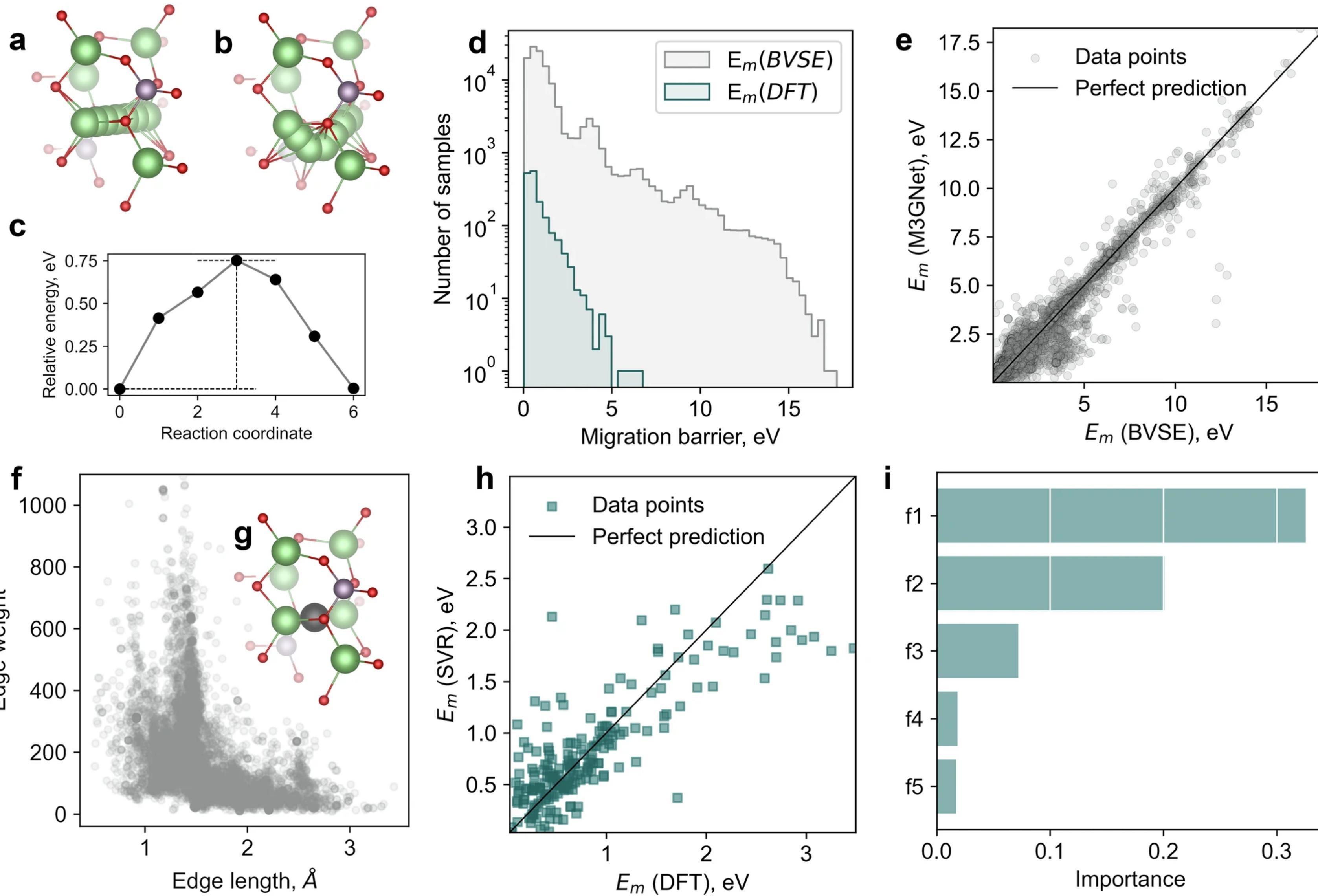
- The simpler is better

Have a look at this page: [https://
www.datawrapper.de/blog/colors](https://www.datawrapper.de/blog/colors)



**Don't use too many panels
(if possible)**

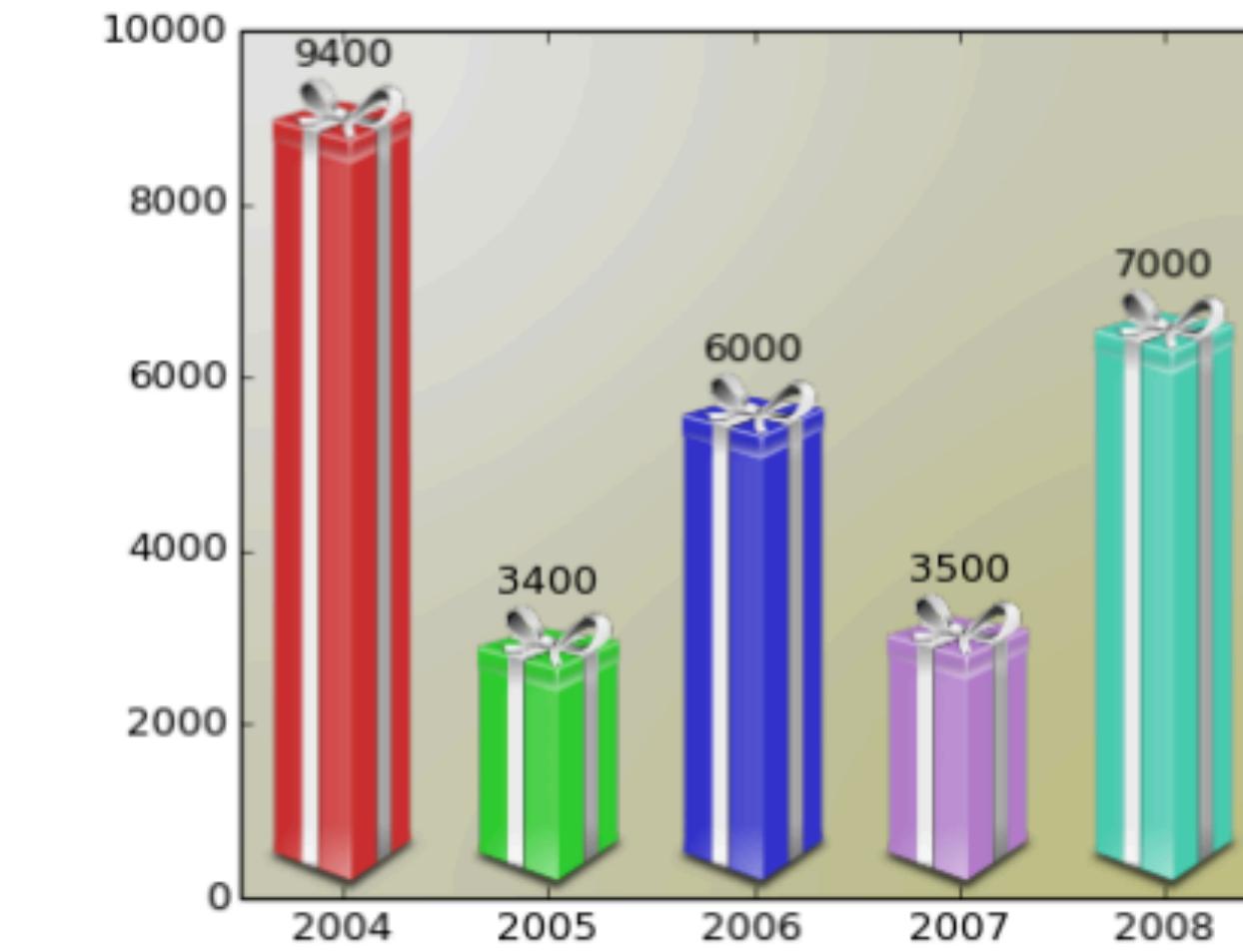
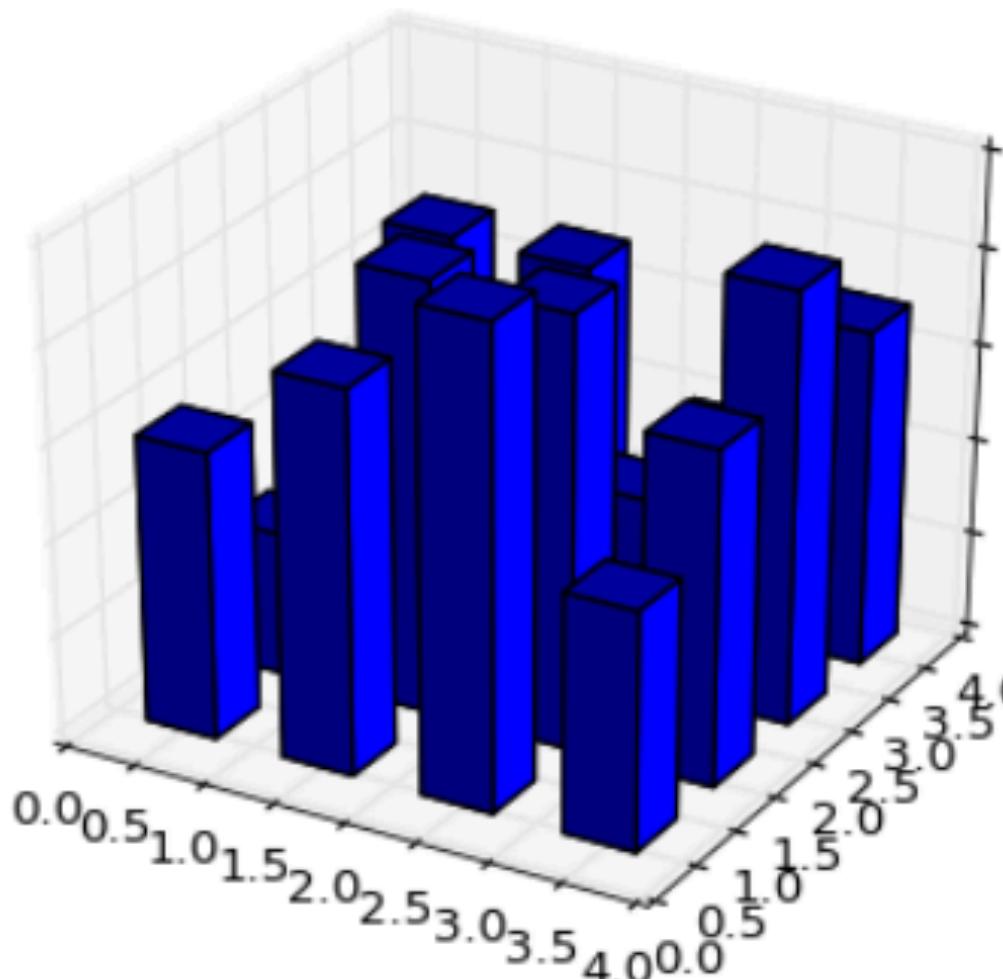
- It is difficult to follow



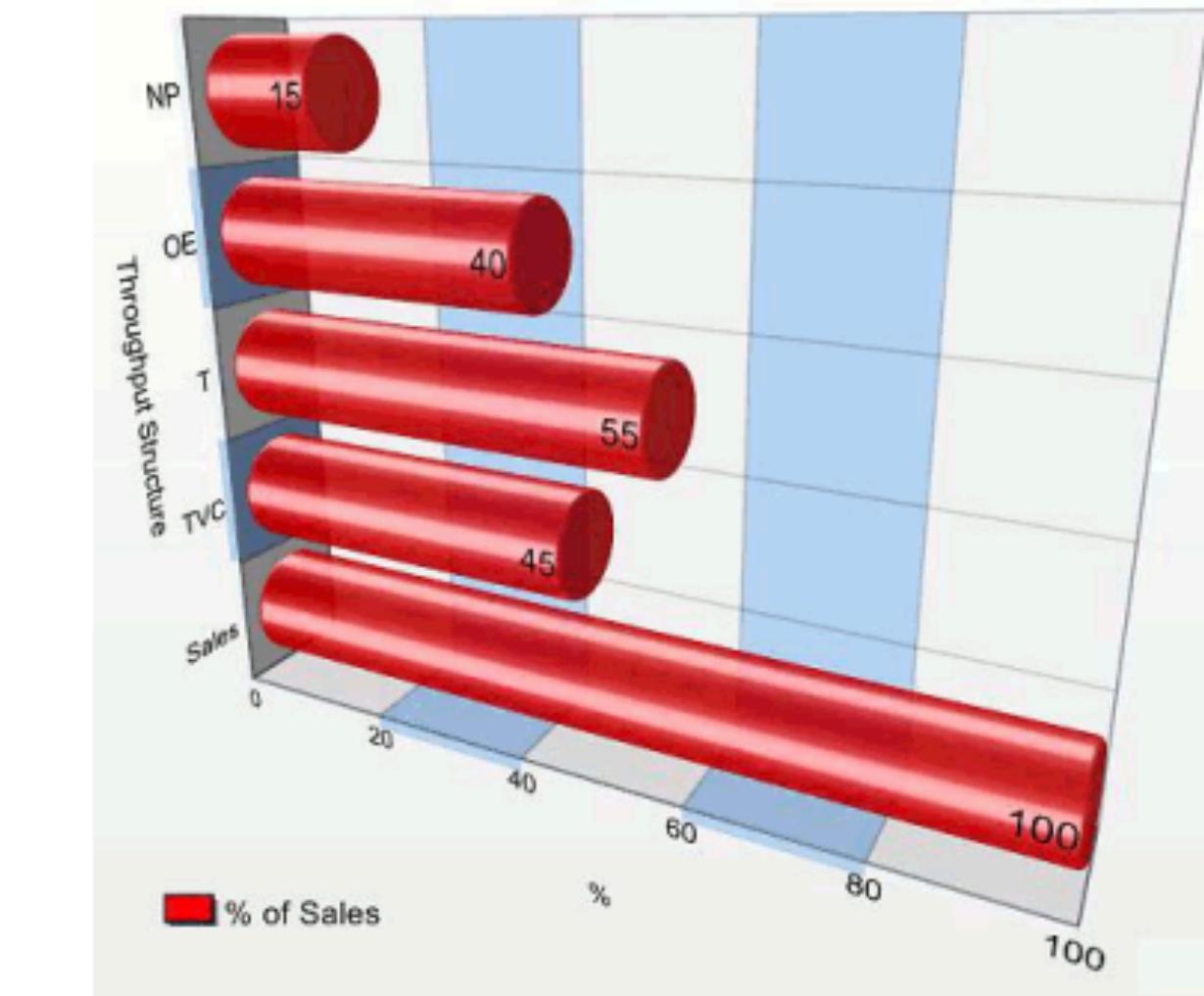
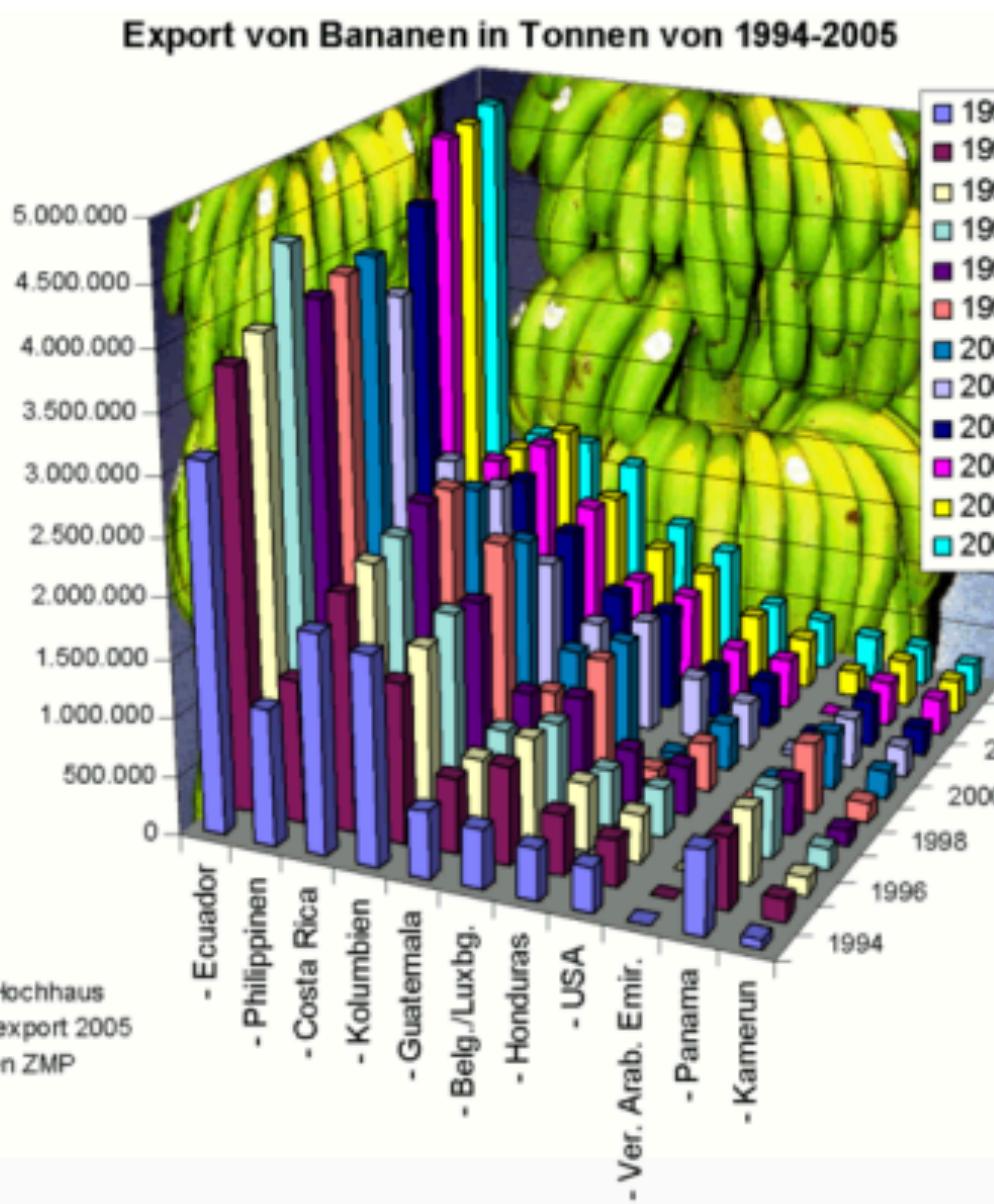
Don't!

My favourite

- Don't use 3D plots
- It's difficult to compare



matplotlib gallery



Excel Charts Blog

Take home message

Visualizing data helps you

- Present data and ideas
- Analyze results
- Define future steps

The data is more important than the design

- Represent the data in a right way
- Avoid misleading graphs

Resources

<https://harvard-iacs.github.io/2018-CS109A/lectures/lecture-3/presentation/lecture3.pdf>

https://en.wikipedia.org/wiki/Misleading_graph

https://en.wikipedia.org/wiki/Anscombe%27s_quartet

<https://blog.datawrapper.de/colors/>