

# Tools for EDA & visualisation

Jan Hučín

2022

# Outline

PROFINIT

1. EDA – reminder
2. Why we make graphs & examples
3. How to – imperative vs. declarative plotting
4. How to – grammar of graphics principle
5. How to – Python tools

# EDA = exploratory data analysis

PROFINIT

The path to understanding the reality behind data.

- › data understanding
- › statistical inference

Use of:

- › data source
- › statistical tools
- › visualisation tools
- › reporting

| HR Information                |             | Contact       |       |
|-------------------------------|-------------|---------------|-------|
| Position                      | Salary      | Office        | Extn. |
| Accountant                    | \$162,700   | Tokyo         | 5407  |
| Chief Executive Officer (CEO) | \$1,200,000 | London        | 5797  |
| Junior Technical Author       | \$86,000    | San Francisco | 1562  |
| Software Engineer             | \$132,000   | London        | 2558  |
| Software Engineer             | \$206,850   | San Francisco | 1314  |
| Integration Specialist        | \$372,000   | New York      | 4804  |
| Software Engineer             | \$163,500   | London        | 6222  |
| Pre-Sales Support             | \$106,450   | New York      | 8330  |
| Sales Assistant               | \$145,600   | New York      | 3990  |
| Senior Javascript Developer   | \$433,060   | Edinburgh     | 6224  |



# Typical use of EDA

PROFINIT

Goal:

- *exploration of dataset XY (regarding problems P1, P2)*

Data:

- *dataset XY, obtained from source Z,*
- *limited to cases ABC, from 2020 to 2021*

...

Summary:

- *regarding P1, there is no useful data in dataset XY because of reasons 1,2,3*
- *regarding P2, it is related to variables K, L, M; their distributions and relationships are described in the report*

Part of **Data Understanding**  
(see CRISP)

- › Key dataset properties
- › Tables structure and values
- › Data origin and quality
- › *Descriptive statistics*
- › *Data visualisation*

# Purpose – technical and impressing

PROFINIT

Perception of the (good) image – much faster than of the table.

- › to show observed **distributions** of individual variables
- › to show observed **relationships** between multiple variables
- › to build (or support) a **story**

# Categorical variable distribution

PROFINIT

## Frequency table

- › absolute
- › relative
- › cumulative (for ordered)

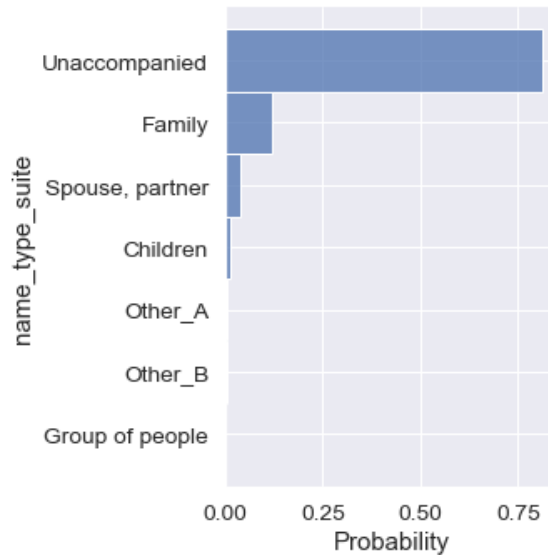
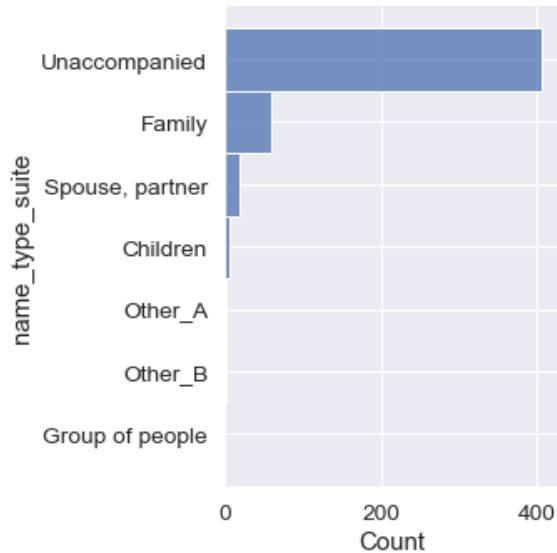
| name_type_suite | count | count_rel |
|-----------------|-------|-----------|
| Children        | 7     | 0.014028  |
| Family          | 60    | 0.120240  |
| Group of people | 1     | 0.002004  |
| Other_A         | 3     | 0.006012  |
| Other_B         | 2     | 0.004008  |
| Spouse, partner | 20    | 0.040080  |
| Unaccompanied   | 406   | 0.813627  |

| education                     | count | count_cum | count_rel | count_relcum |
|-------------------------------|-------|-----------|-----------|--------------|
| Lower secondary               | 1     | 1         | 0.002     | 0.002        |
| Secondary / secondary special | 358   | 359       | 0.716     | 0.718        |
| Incomplete higher             | 14    | 373       | 0.028     | 0.746        |
| Higher education              | 127   | 500       | 0.254     | 1.000        |

# Categorical variable distribution

PROFITIT

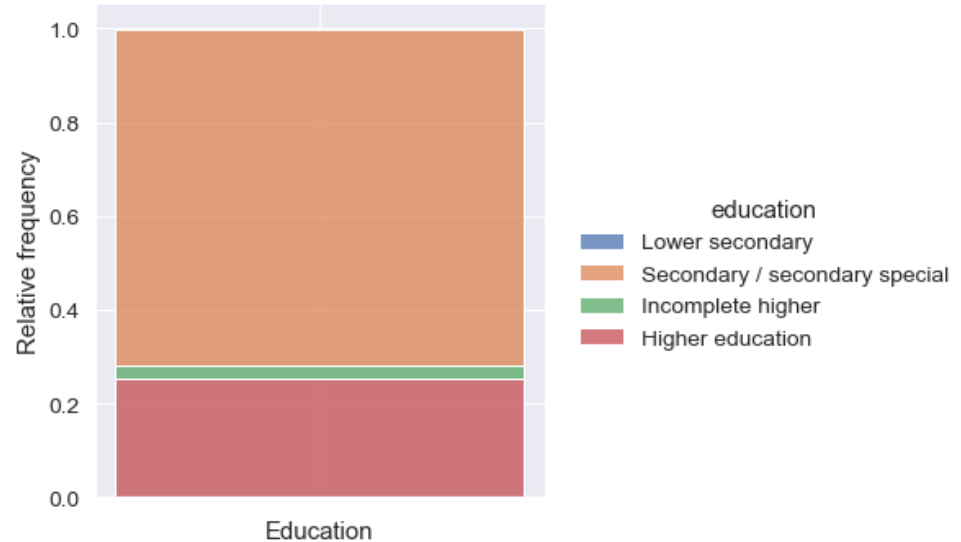
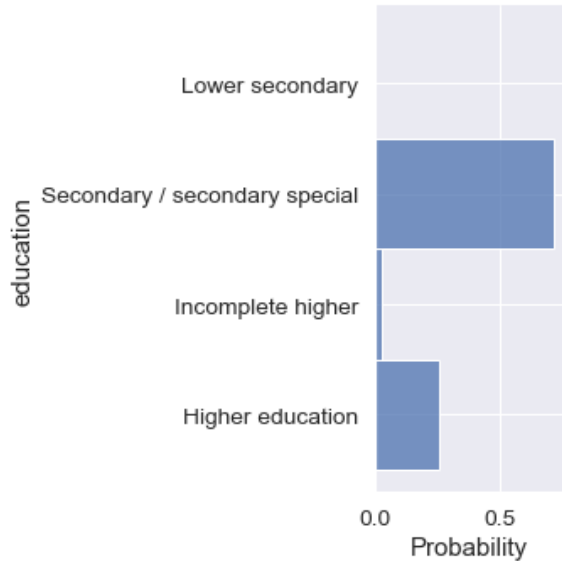
## Frequency graphs



# Categorical variable distribution

PROFITIT

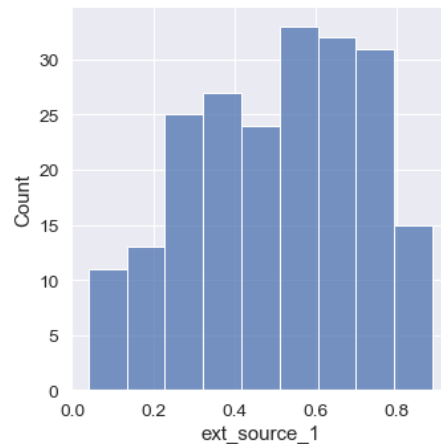
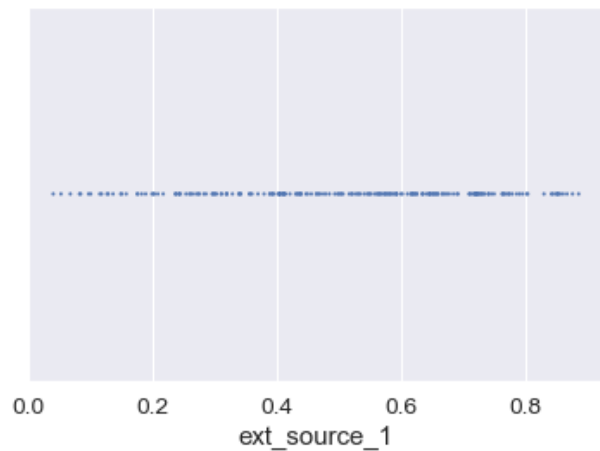
## Frequency graphs stacked (ordinal variables)



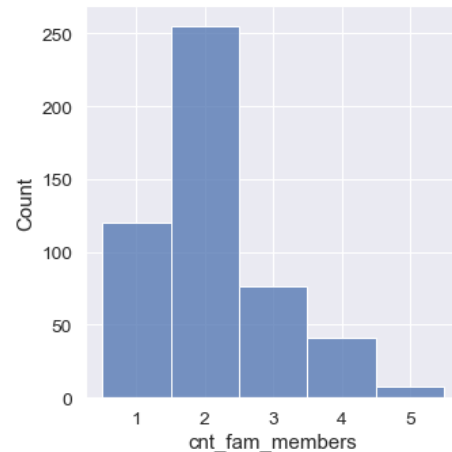


# Numeric variable distribution

- › A little of unique values → treat as categorical
- › A lot of unique values:
  - full information: ECDF, rug/strip
  - balanced: histogram, density estimation
  - compressed: boxplot, numerical statistics



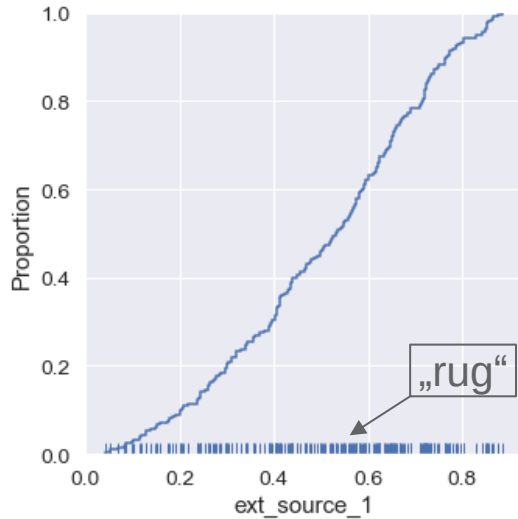
PROFITIT



# Numeric variable distribution

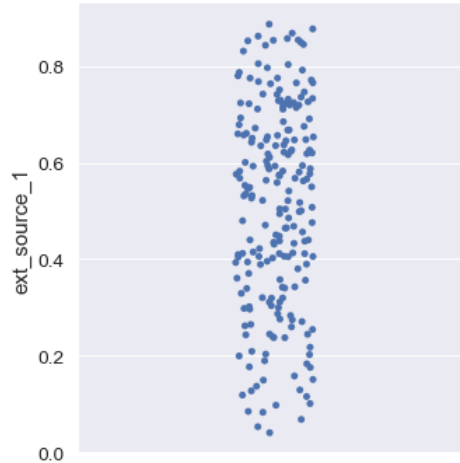
PROFINIT

full information:

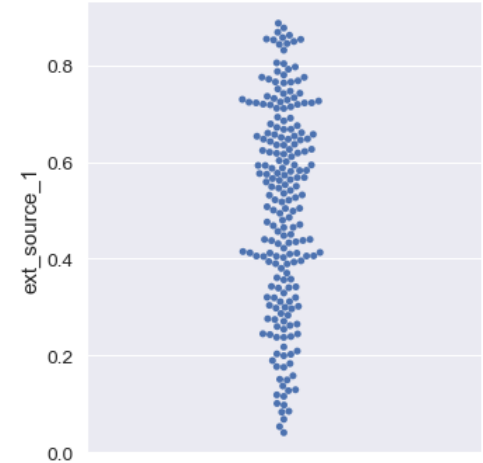


ECDF: empirical cumulative  
distribution function

$$F(z) = \text{prop. cases } (X < z)$$



stripplot

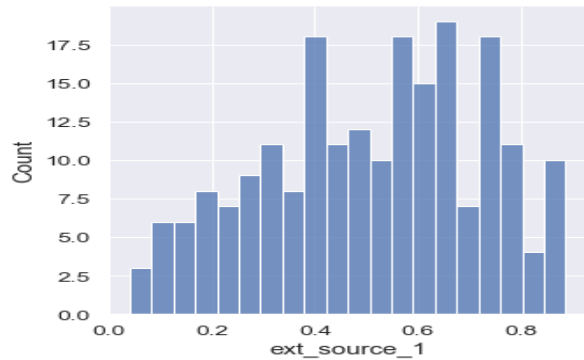
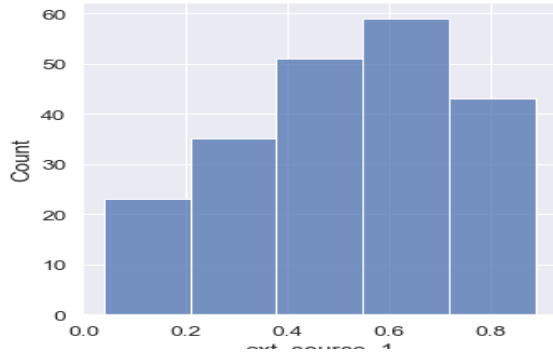


swarmplot

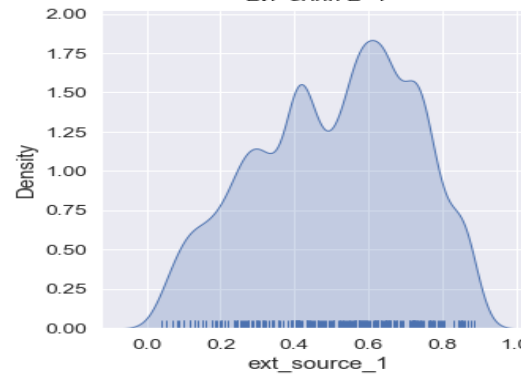
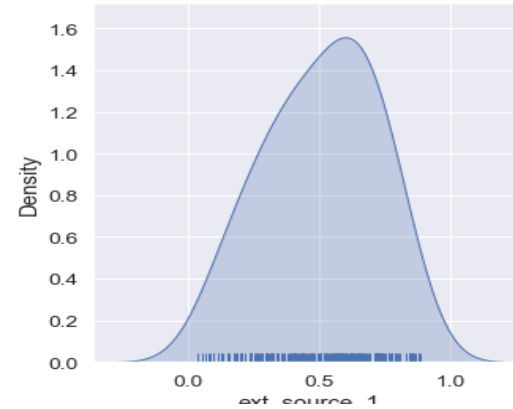
# Numeric variable distribution

PROFITIT

balanced:



histogram



KDE (kernel density est.)

# Numeric variable distribution

PROFINIT

compressed: (essential info in few numbers)

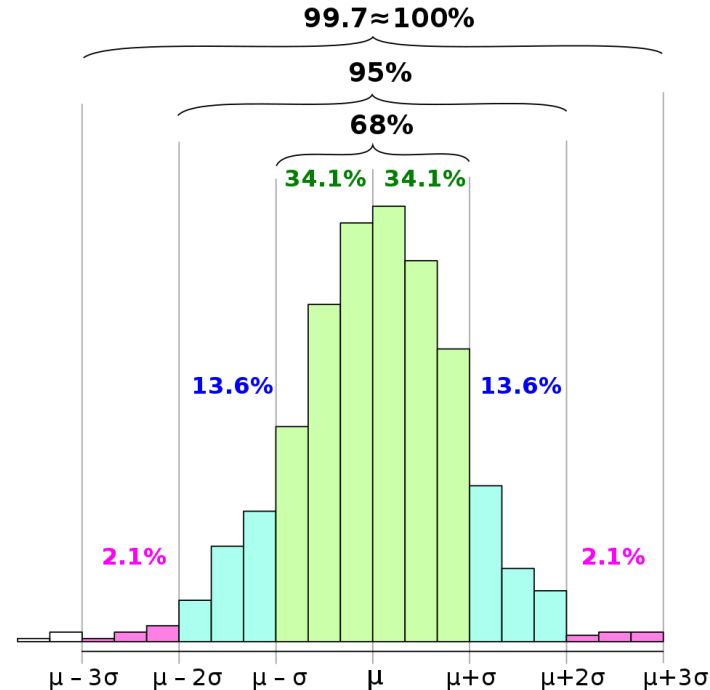
- › What range are values in? → min, max
- › Which value is the „center“? → (trimmed) mean/average, median, mode
- › What is the dispersion of values? → standard deviation, interquartile range
- › What other values or thresholds are important? → quantiles, second mode
- › What is the shape of distribution? → approximating by a standard distribution

# Numeric variable distribution

PROFINIT

## Gaussian-like distributions

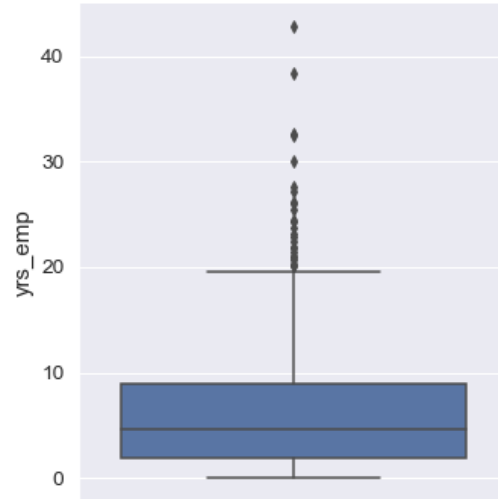
- › quasi-symmetric, unimodal
- › mean ~ median ~ mode
- › standard deviation (SD, sigma,  $\sigma$ ) is meaningful
- ›  $1\sigma$ ,  $2\sigma$ ,  $3\sigma$  rules applies



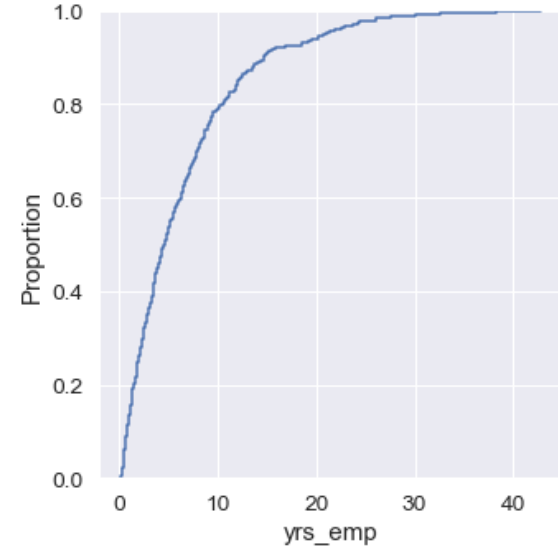
# Numeric variable distribution

## skewed distributions

- › non-symmetric, unimodal
- ›  $\text{mean} \neq \text{median}$ ,  $\text{mean} \neq \text{mode}$
- › SD hardly interpretable
- › needs robust statistics: quantiles (median, quartiles, deciles)
- › boxplot, ECDF



boxplot  
median, quartiles, „outliers“



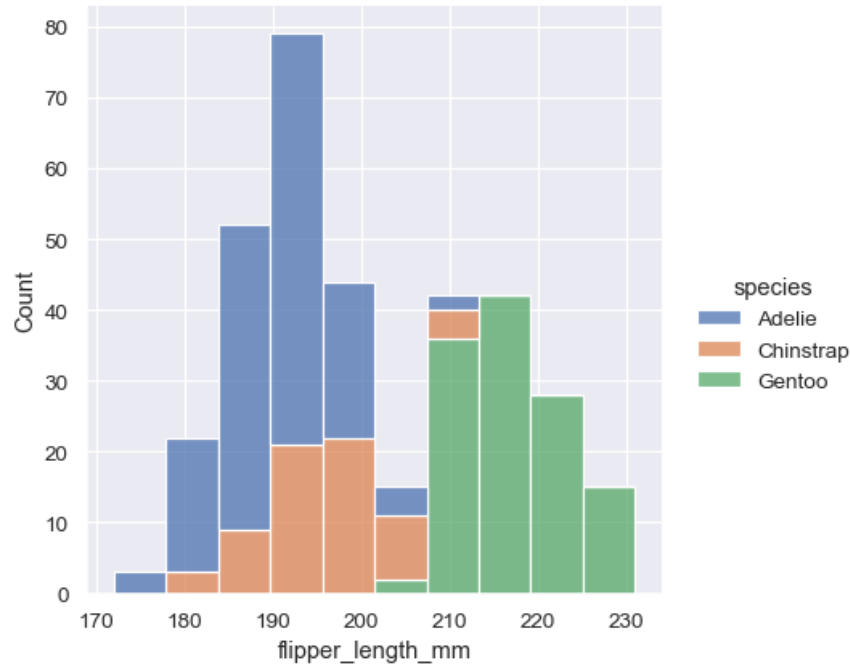
ECDF

# Numeric variable distribution

PROFITIT

## weird distributions

- › mix of multiple distributions
- › when split by another variable, seems reasonable
- › → relationships between (among) variables



# Relationships between variables

PROFINIT

1. categorical vs. categorical
2. categorical vs. numeric
3. numeric vs. numeric

Multiple relationships: basic pair + split by other (categorical) variables



# Categorical vs. categorical

PROFITIT

## contingency table

- › absolute
- › relative by rows / columns
- › relative completely

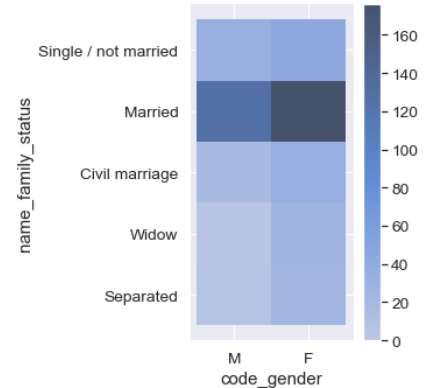
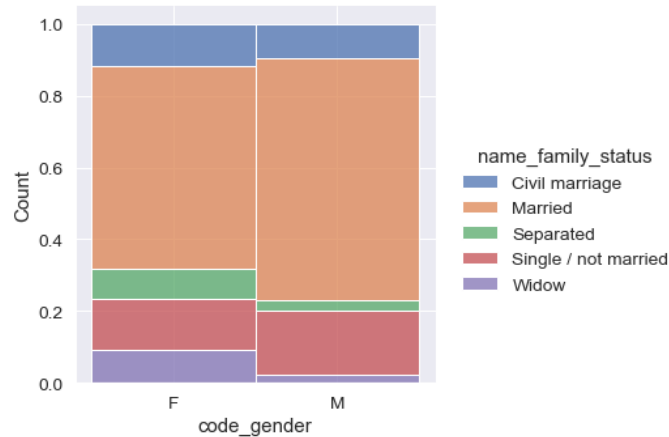
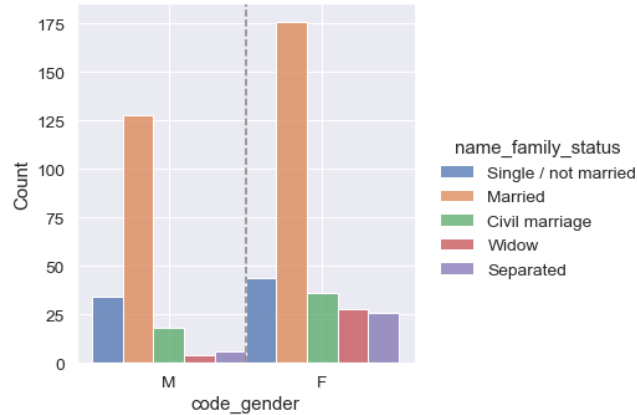
| Family_status        | female | male |
|----------------------|--------|------|
| Civil marriage       | 36     | 18   |
| Married              | 176    | 128  |
| Separated            | 26     | 6    |
| Single / not married | 44     | 34   |
| Widow                | 28     | 4    |

| Family_status        | female | male  |
|----------------------|--------|-------|
| Civil marriage       | 0.116  | 0.095 |
| Married              | 0.568  | 0.673 |
| Separated            | 0.084  | 0.032 |
| Single / not married | 0.142  | 0.179 |
| Widow                | 0.090  | 0.021 |

# Categorical vs. categorical graphs

PROFINIT

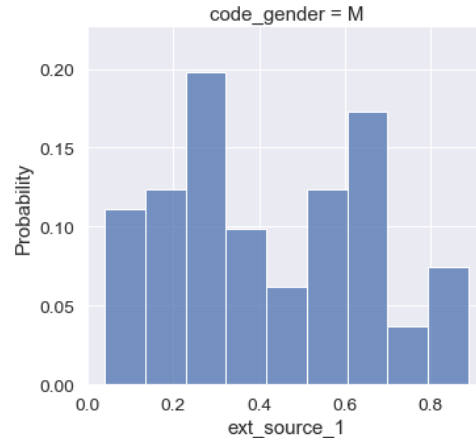
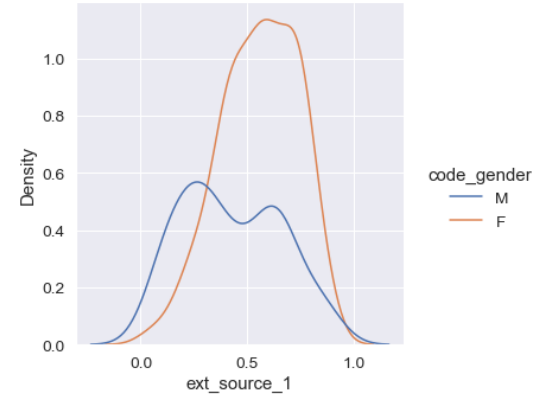
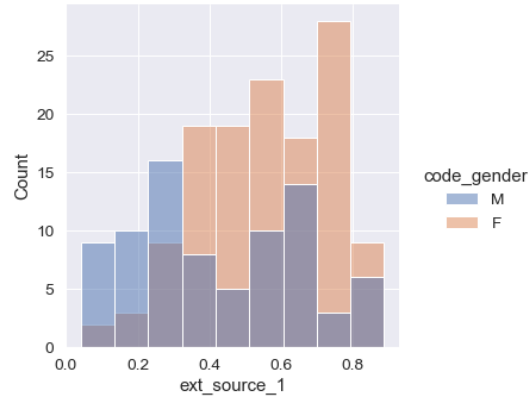
- › barplot
- › discrete heatmap



# Numeric vs. categorical graphs

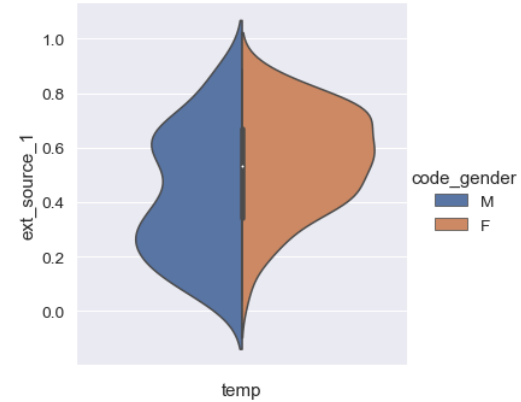
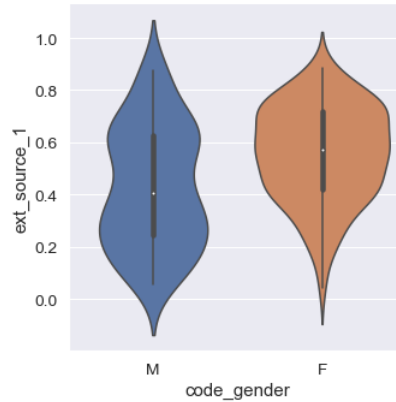
PROFITIT

- › split by categories
- › look at numeric distribution by category



# Numeric vs. categorical graphs

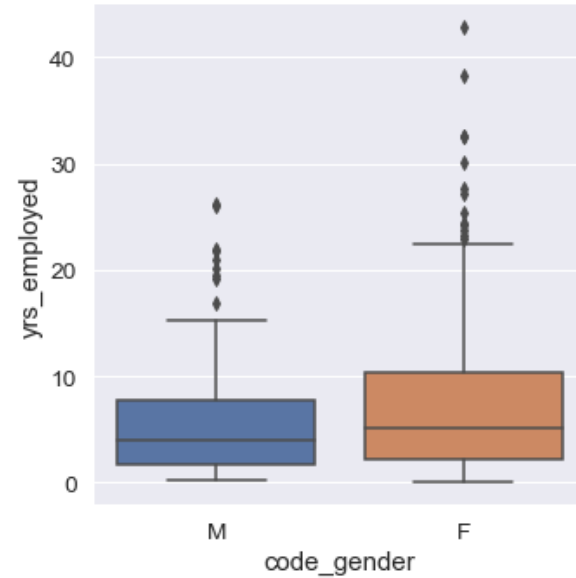
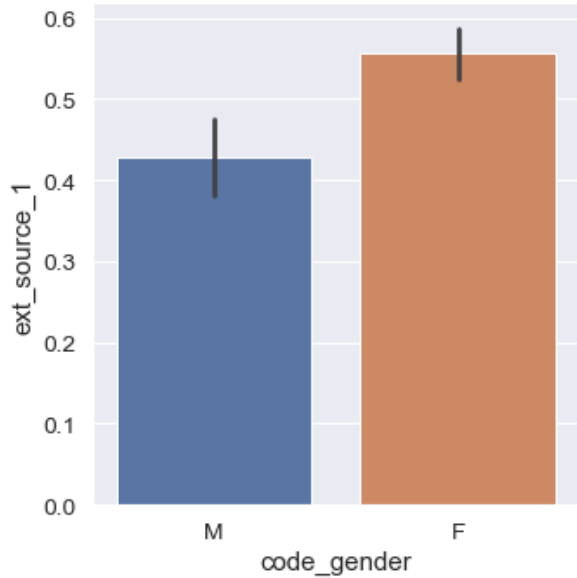
PROFINIT



violinplot

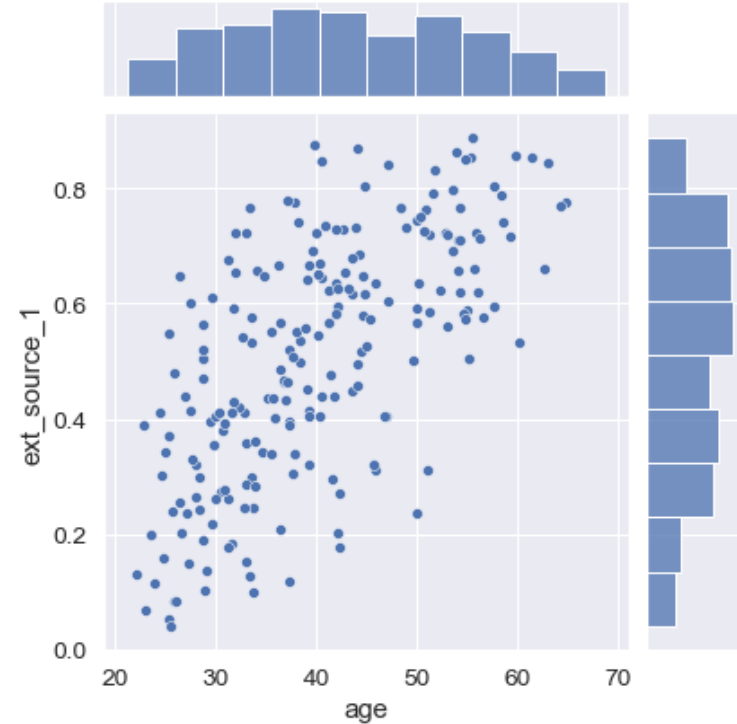
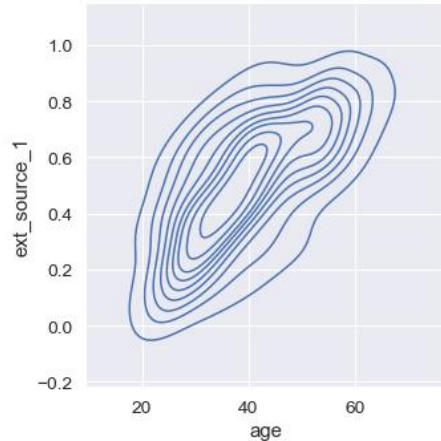
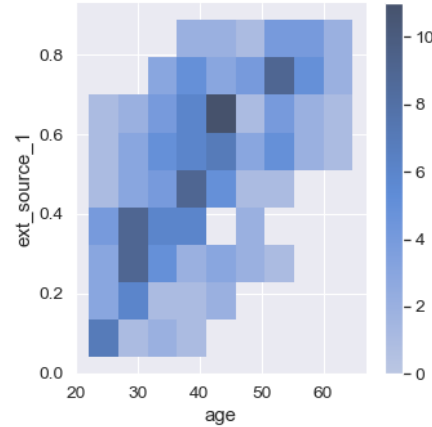
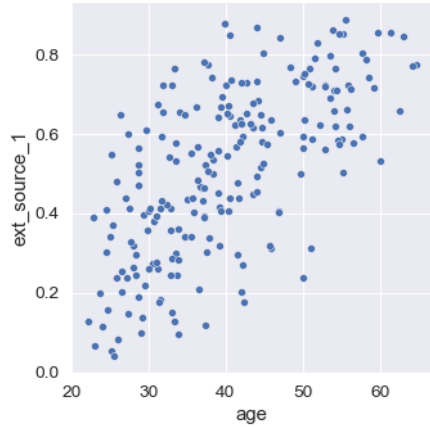
# Numeric vs. categorical graphs

PROFITIT



# Numeric vs. numeric graphs

PROFINIT

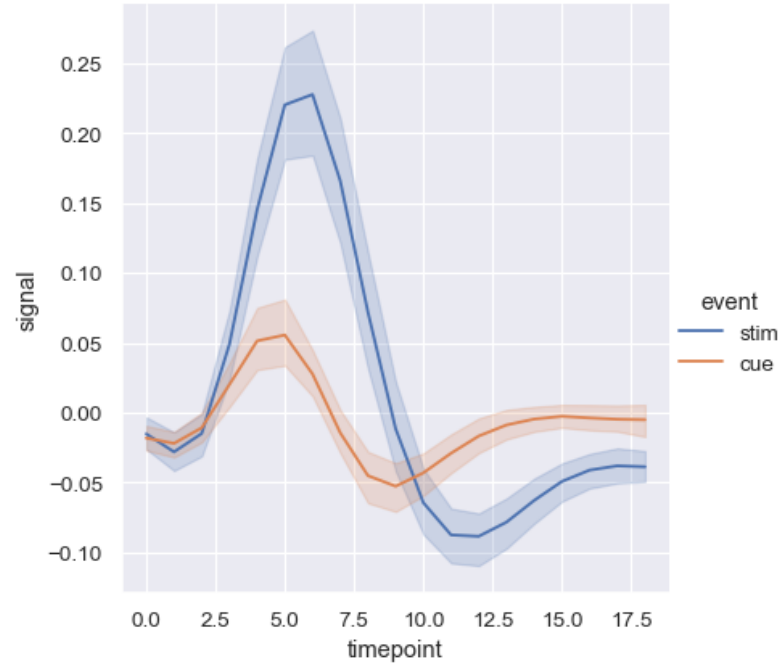
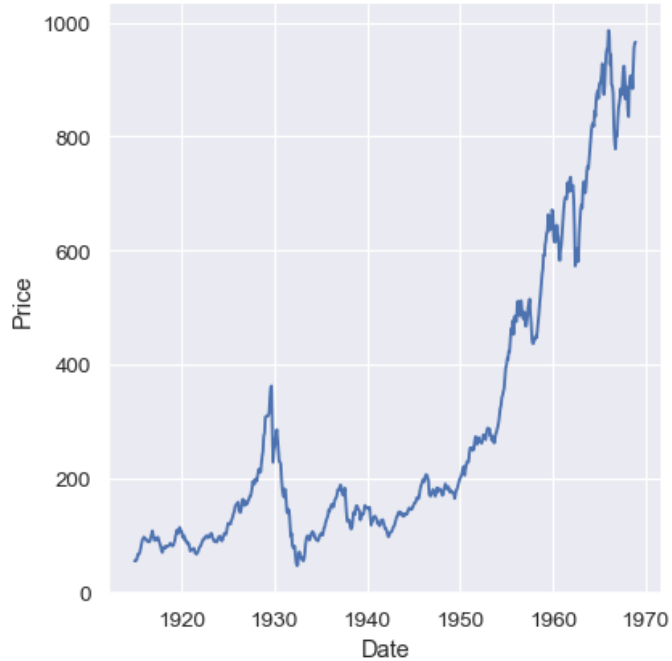


- › scatterplot
- › contourplot
- › heatmap

# Numeric vs. numeric graphs

PROFINIT

## › lineplot



# Imperative and declarative plotting

PROFINIT

## Imperative plotting

Detailed instructions step by step.

- › Draw a yellow circle in the middle.
- › Draw two small black circles side by side.
- › (etc.)
- › **+** full control, get whatever you want
- › **-** tedious, Leonardos are rare





# Imperative and declarative plotting

PROFINIT

## Declarative plotting

Asking a friend/artist to draw a picture according to your needs.

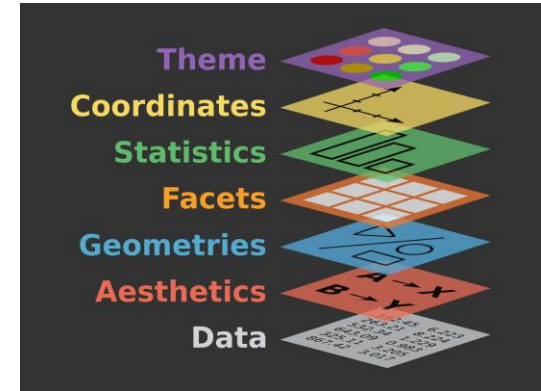
- › Draw young lady with dark hair sitting alone and having a mysterious smile on her face.
- › + nicely looking results
- › - lower level of control, need to know how to express your needs



# Grammar of graphics

## Seven components

1. Theme = Adds all non-data ink
2. Coordinates = How do we position the visual?
3. Statistics = How we preprocess the data?
4. Facets = Do we split the visual into subplots?
5. Geometric objects = What marks are we using?
6. Aesthetics = How do we show it?
7. Data = What do we want to show?



# Plotting in Python

PROFINIT

## › **matplotlib**

- standard package
- „sweat and toil“

## › **seaborn**

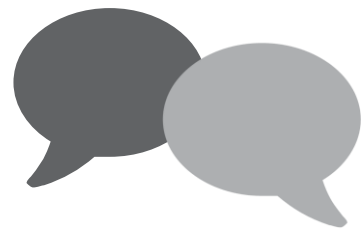
- package for high-level plotting
- more intuitive
- enables low-level fine-tuning by matplotlib

## › **plotnine**

- grammar of graphics

## › **profplots**

- ?



**Questions?**