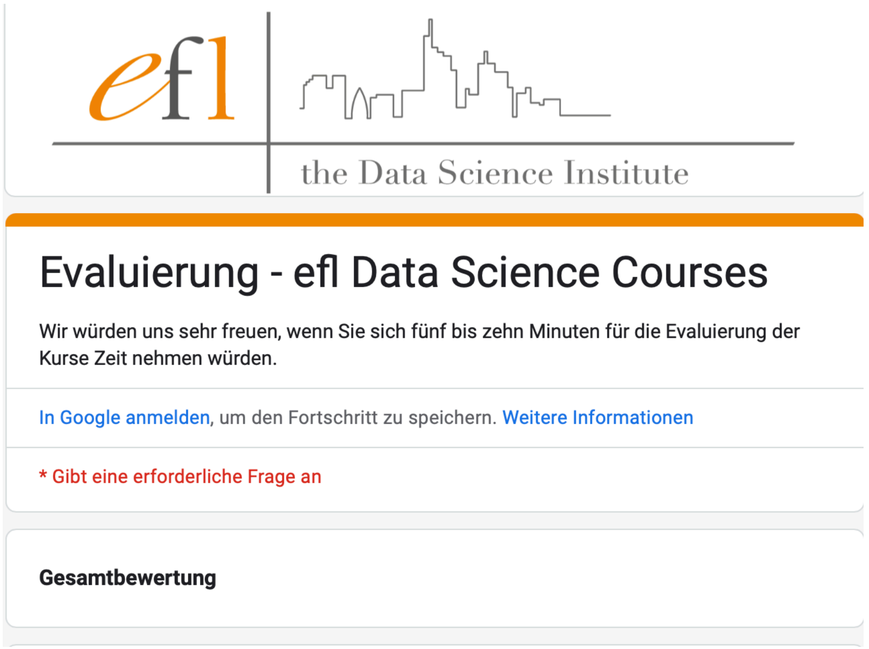
Lecture:
Data Visualization

Elf Data Science Course

Dr. Benjamin M. Henrich

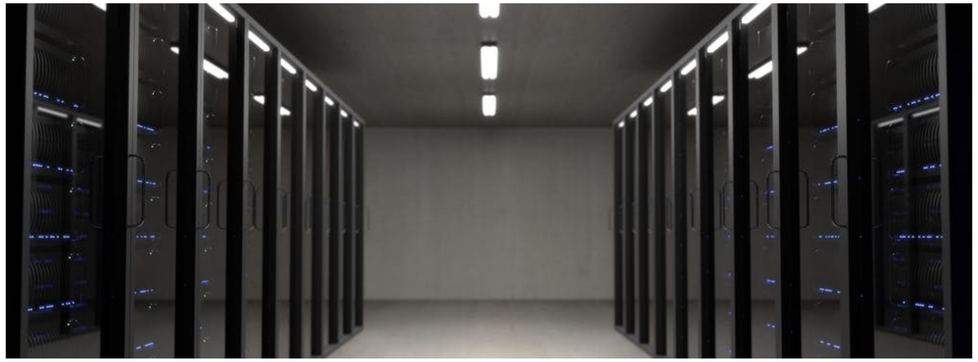
Survey



https://docs.google.com/forms/d/e/1FAIpQLSdH4RcSBN20rMX3bQjzfIt1JX_RjTijpiRebA_c062j83RFig/viewform

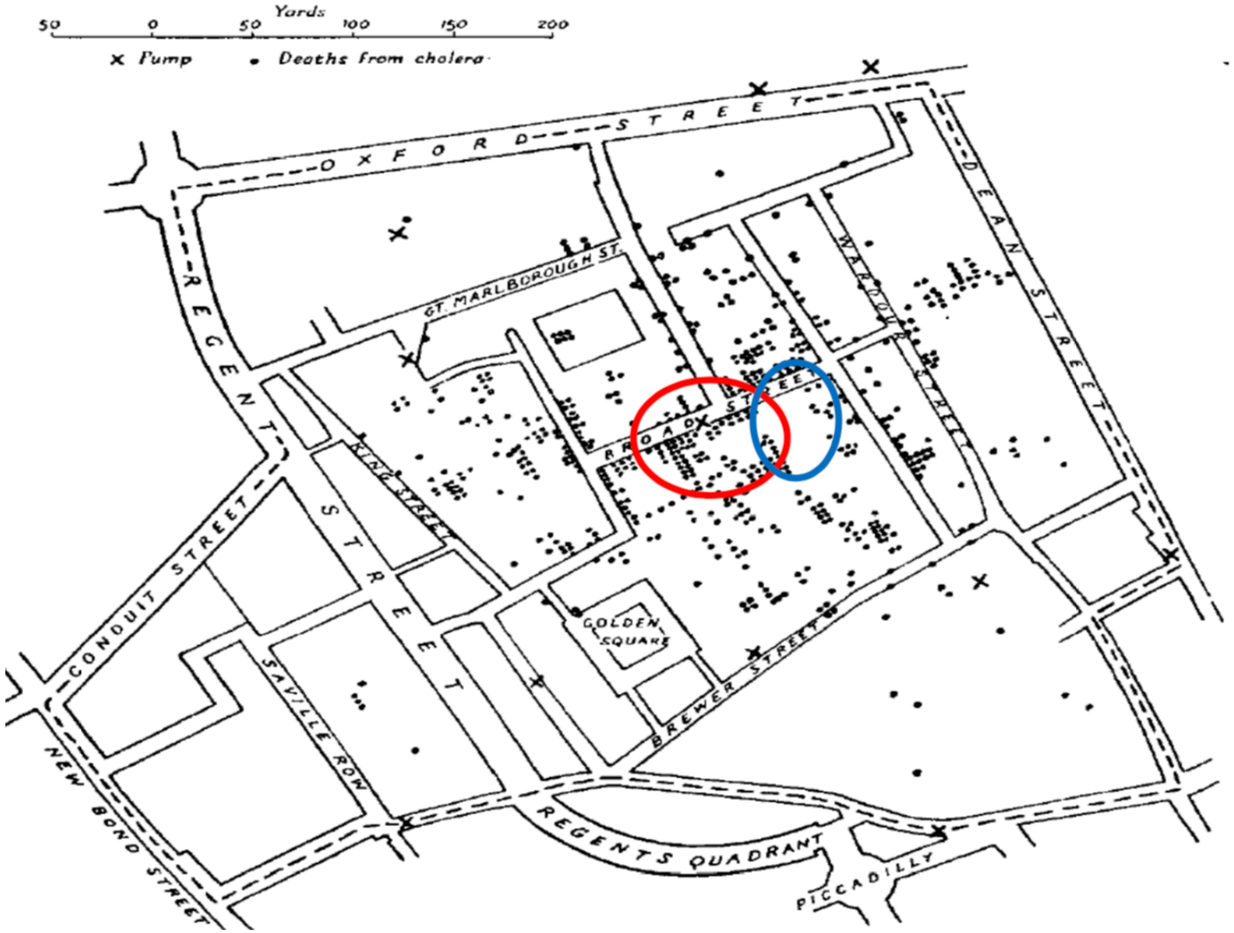
Motivation

- Big data can also mean data chaos
- An image says more than 1000 words
- The goal from data to knowledge



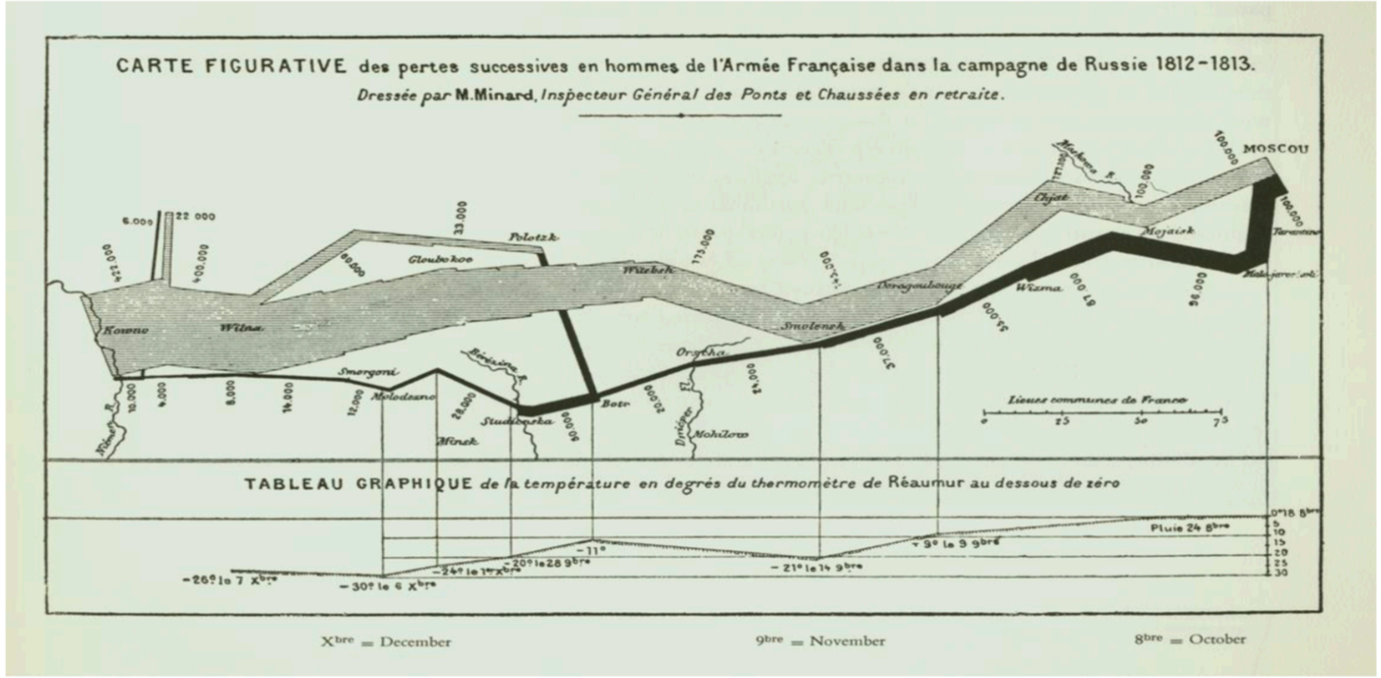
Motivation

- Dr. John Snow (1854): Map of the Cholera Epidemic of London (1853)
- Shows cases of illness sorted by street district
- Contaminated pump might caused the local cholera cases
- Water pump affected has been shut down



Motivation

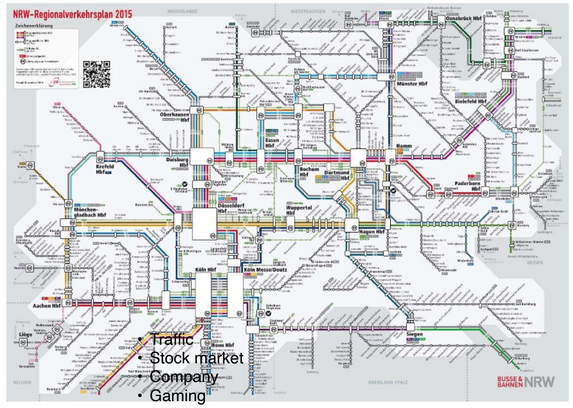
- Minard (1861): Map of Napoleon's campaign in Russia (1812/13)"the best statistical drawing ever made..." (Tufts)army strength
- Troop movements
- Temperature during the retreat conditions

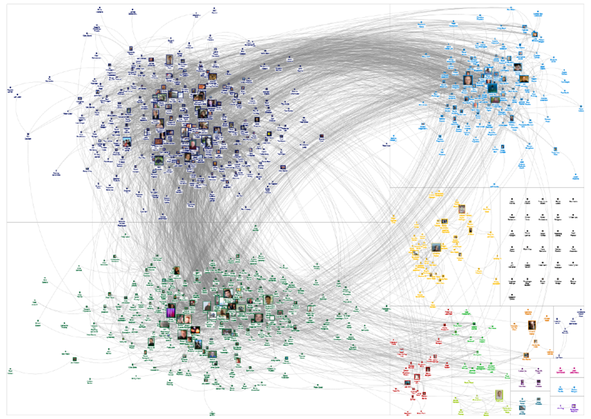


Today visualizations are everywhere

Examples: …

- Traffic
- Stock market
- Company
- Gaming



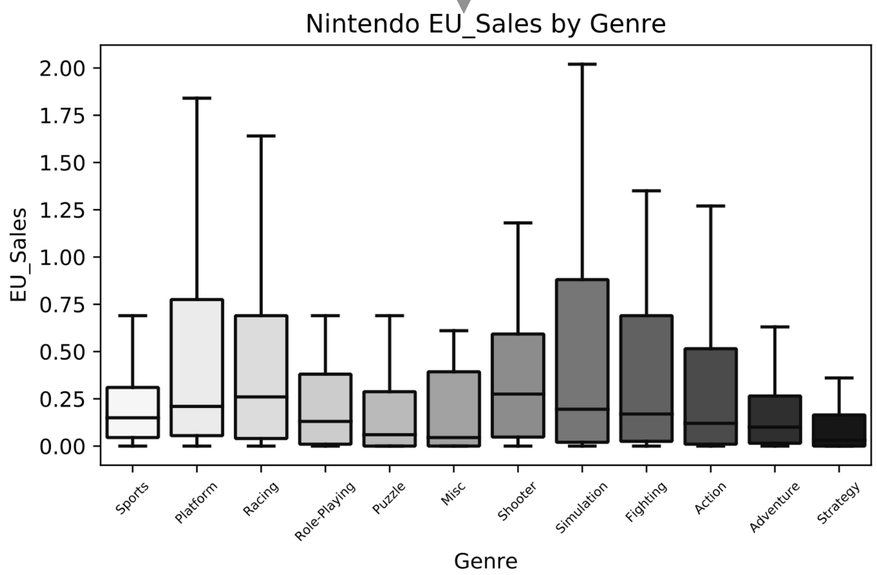


Data/ Information Visualization

- The use of computer-supported, interactive, visual representations of abstract data to amplify cognition. [Card et al 1999]

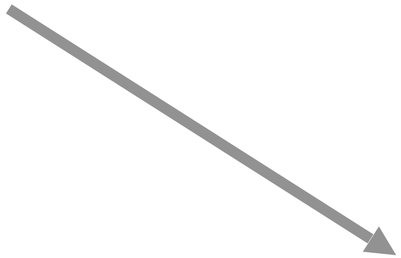
- Is the communication of abstract data through the use of interactive visual interfaces.
[Keim et al 2006]

	Rank	Name	Platform	Year	Genre	Publisher	NA_Sales	EU_Sales	JP_Sales	Other_Sales	Global_Sales
0	1	Wii Sports	Wii	2006.00000	Sports	Nintendo	41.49000	29.02000	3.77000	8.46000	82.74000
1	2	Super Mario Bros.	NES	1985.00000	Platform	Nintendo	29.08000	3.58000	6.81000	0.77000	40.24000
2	3	Mario Kart Wii	Wii	2008.00000	Racing	Nintendo	15.85000	12.88000	3.79000	3.31000	35.82000
3	4	Wii Sports Resort	Wii	2009.00000	Sports	Nintendo	15.75000	11.01000	3.28000	2.96000	33.00000
4	5	Pokemon Red/Pokemon Blue	GB	1996.00000	Role-Playing	Nintendo	11.27000	8.89000	10.22000	1.00000	31.37000
5	6	Tetris	GB	1989.00000	Puzzle	Nintendo	23.20000	2.26000	4.22000	0.58000	30.26000
6	7	New Super Mario Bros.	DS	2006.00000	Platform	Nintendo	11.38000	9.23000	6.50000	2.90000	30.01000
7	8	Wii Play	Wii	2006.00000	Misc	Nintendo	14.03000	9.20000	2.93000	2.85000	29.02000
8	9	New Super Mario Bros. Wii	Wii	2009.00000	Platform	Nintendo	14.59000	7.06000	4.70000	2.26000	28.62000
9	10	Duck Hunt	NES	1984.00000	Shooter	Nintendo	26.93000	0.63000	0.28000	0.47000	28.31000
10	11	Nintendogs	DS	2005.00000	Simulation	Nintendo	9.07000	11.00000	1.93000	2.75000	24.76000
11	12	Mario Kart DS	DS	2005.00000	Racing	Nintendo	9.81000	7.57000	4.13000	1.92000	23.42000
12	13	Pokemon Gold/Pokemon Si...	GB	1999.00000	Role-Playing	Nintendo	9.00000	6.18000	7.20000	0.71000	23.10000
13	14	Wii Fit	Wii	2007.00000	Sports	Nintendo	8.94000	8.03000	3.60000	2.15000	22.72000
14	15	Wii Fit Plus	Wii	2009.00000	Sports	Nintendo	9.09000	8.59000	2.53000	1.79000	22.00000



From Data to Visualization

V_Time	V_Name	V_Size
2019-05-01	Julius	1.85
2019-05-02	Laura	1.55
2019-05-05	Benny	1.87



V_Size
1.85
1.55
1.87

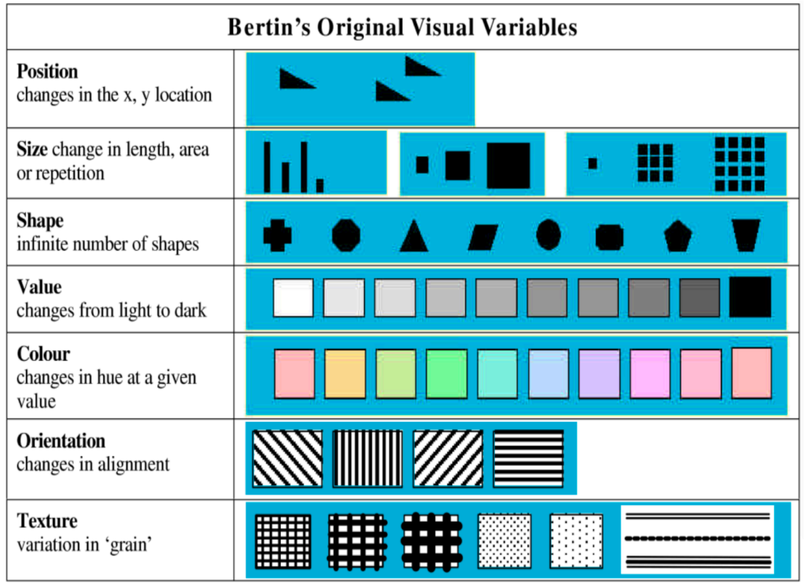
Basic Types of Variables for Visualizations (statistics)

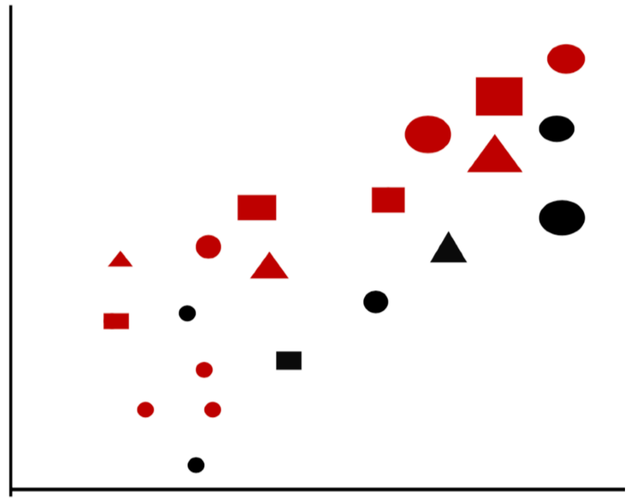
- **Categorical variable** - Disordered set, e.g. names {Ben, Max, Laura} Only defined relation:
  - Equality relation (=)
  - String
- **Ordinal variable** - have natural, ordered categories and the distances between the categories is not known, e.g. Ranking {1,2,3…}
  - Defined order <.
  - Relationen: =, >,<
- **Numerical Variable** - Numeric range, e.g body size [1.85, 1.55, 1.78]
  - Arithmetic operations possible
  - Relationen: =, >,<, und Arithmetische Operationen
  - Discrete variable (Integer)
  - Continuous variable (Float)

Visualization and the Question of right Mapping?

- Visualization is any technique for creating images, diagrams, or animations to communicate a message.

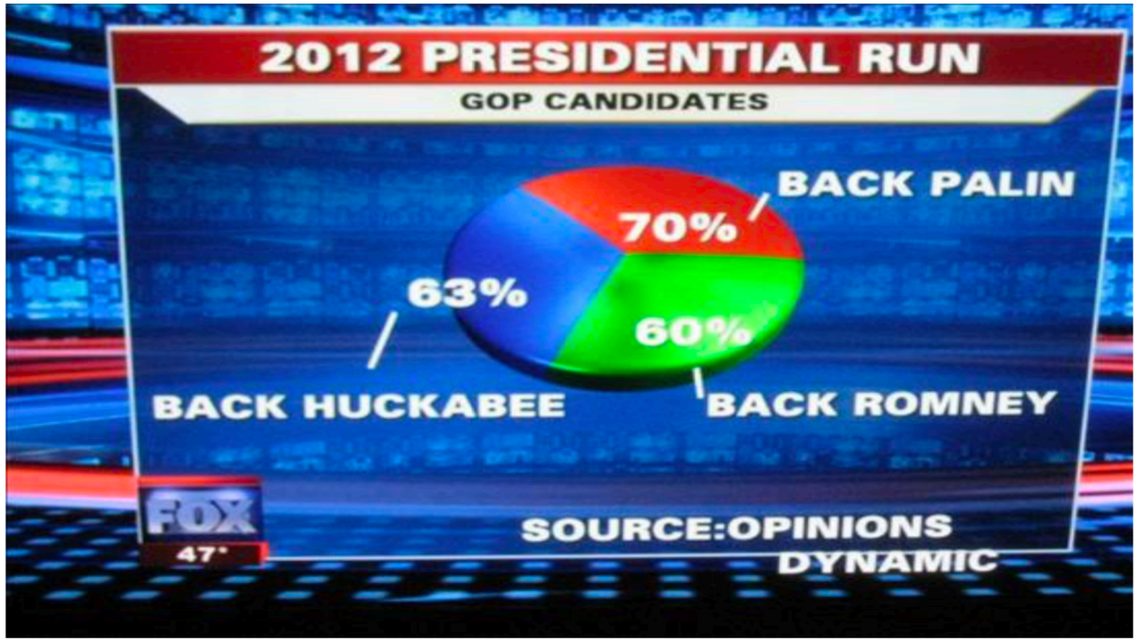
- Visualization through visual imagery has been an effective way to communicate both abstract and concrete ideas since the dawn of humanity.



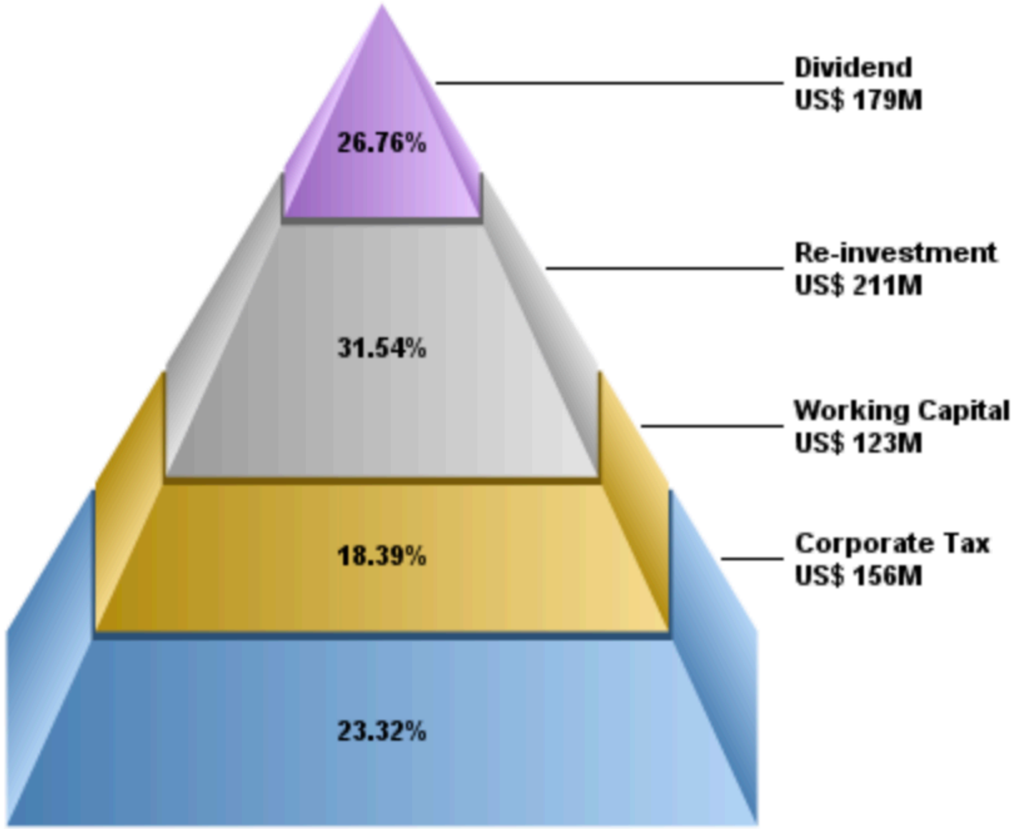


Good Visualizations?

- Let's get some ideas together...



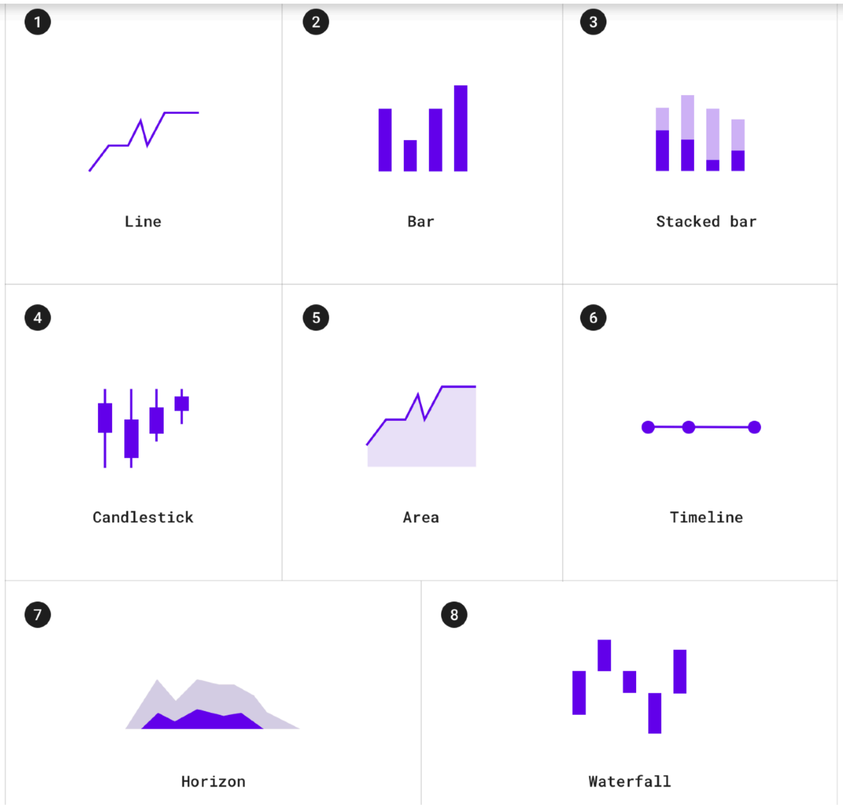
Fox News 2012 Presidential Run Quelle: wonkette.com



http://www.advsofteng.com/gallery_pyramid.html

Data Visualization Style Guidelines

Communication > Data visualization > Types

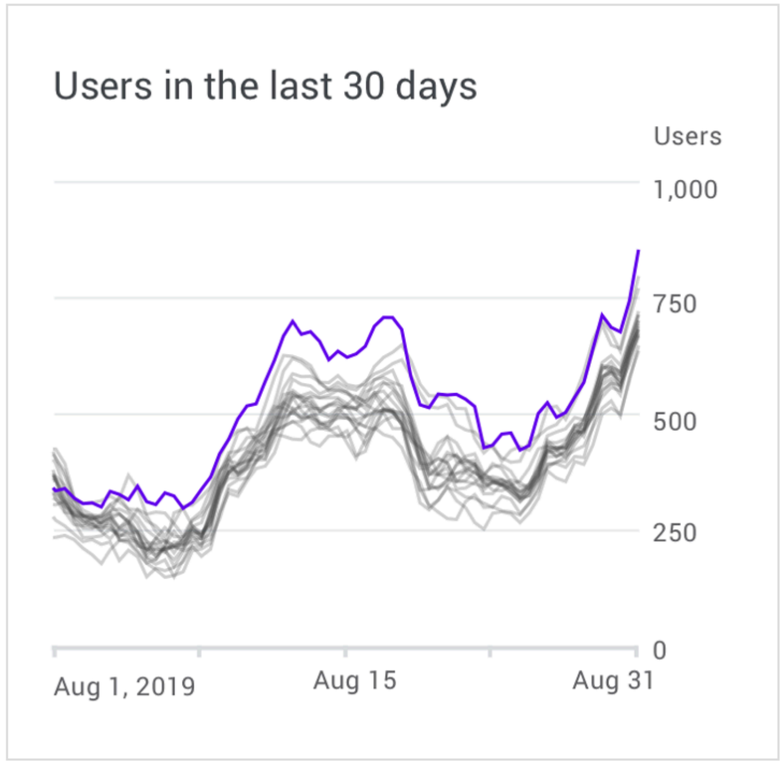


MATERIAL DESIGN

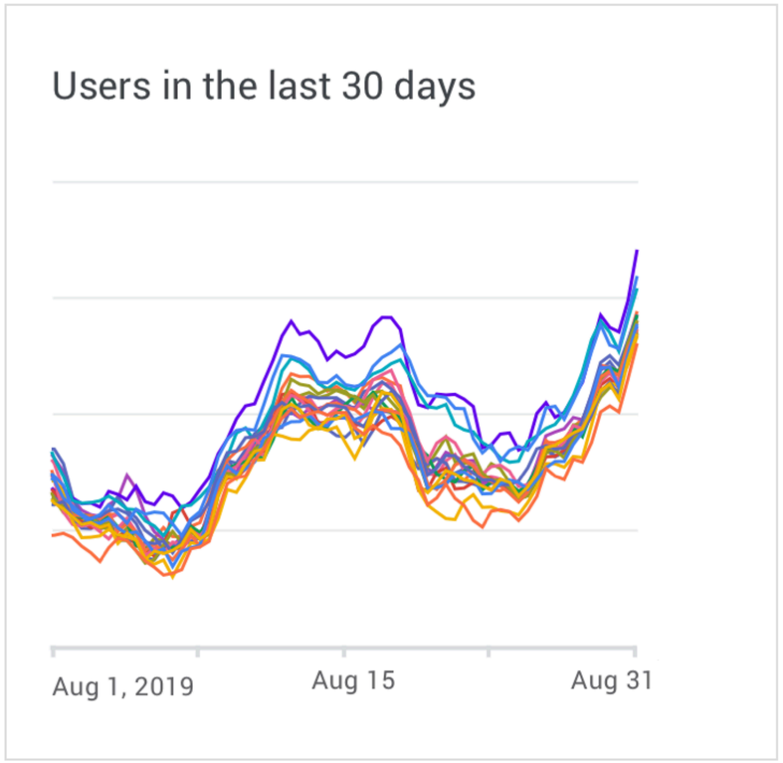
- https://material.io/design/communication/data-visualization.html#

Right Guidelines for Good Visualizations

- Title
- Labels on axis
- Keep it simple
- Only a few colours

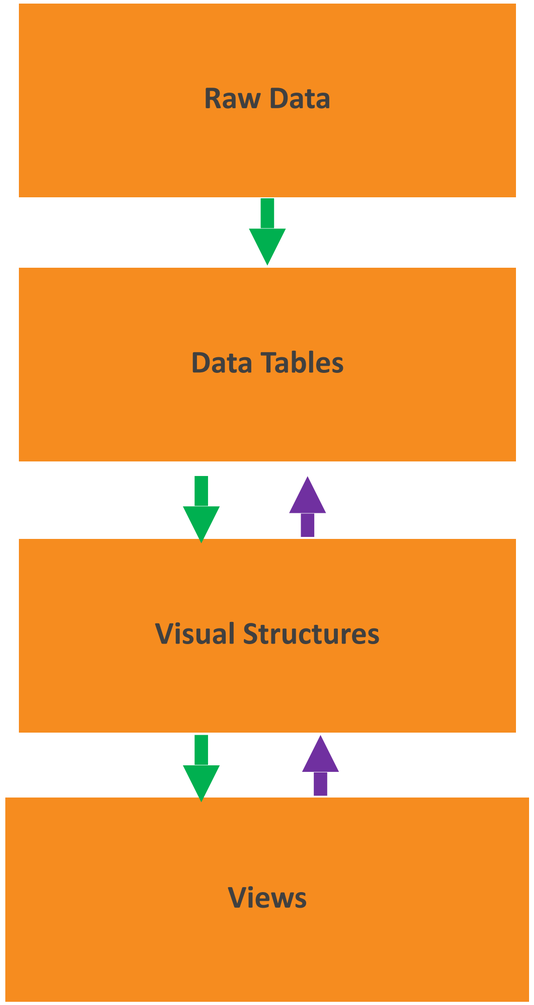


Do



Don't

Visualization Process - The Baseline



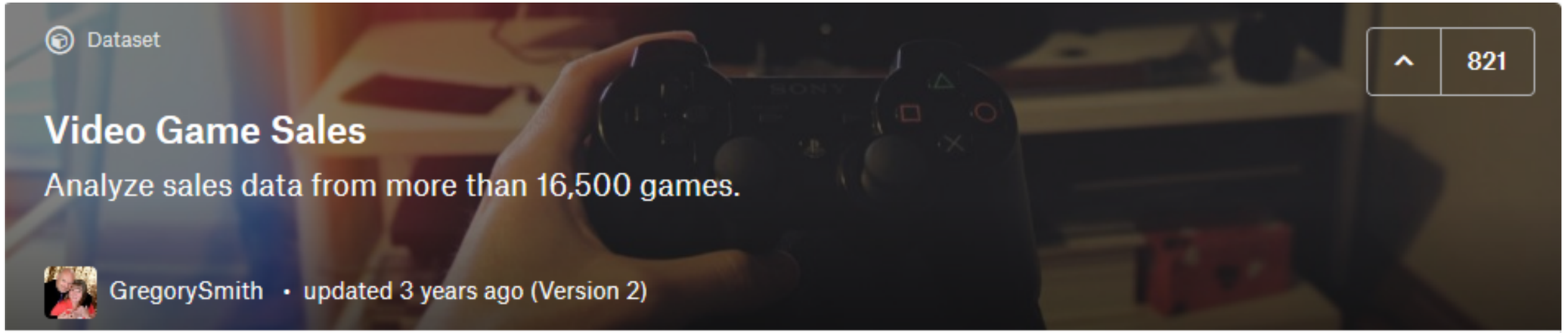
What is the **goal** of visualization?
What is the point of interest?
What do you want to **understand**?

How is the data **sampled**?
Which data is **relevant**?

Plot the data.
Are there **anomalies**?
Are there **patterns**?

Build a Visualization.
Do the results make **sense?**

Using large Datasets for Data Analysis



Cheat Sheet for Today

Imports and Overview

Read a comma-separated values (csv) file into DataFrame

```
df = pd.read_csv(,../data/FILENAME.csv', sep=',')
```

Information about a DataFrame

```
df_info = df.info()
```

Drop all rows with NaN

```
df = df.dropna(axis=0)
```

DataFrame Operations

Information about a DataFrame

```
df[,Column'].method() # General Syntax
```

Example: Get the mean for the column

```
float_mean_global_sales = round(df['Global_Sales'].mean(), 2)
```

Example: Get unique values in a column

```
list_brands = df['Publisher'].unique().tolist()
```

Select rows based on a conditional expression

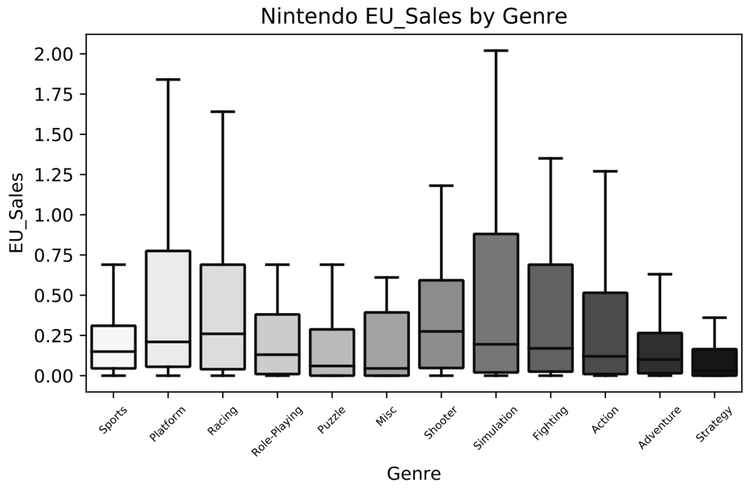
```
df = df[df['A'] == 0]
df = df[(df['A'] == 0) | (df['B'] == 1)]
```

Example: Get the data for the given brand

```
df_nintendo = df[df['Publisher'] == 'Nintendo']
```

Create a Plot

```
plt.figure()
sns.boxplot(x='Genre', y='EU_Sales',
data=df_nintendo, showfliers=False,
            palette='Greys')
plt.title('Nintendo EU_Sales by Genre')
plt.xticks(rotation=45, fontsize=6)
plt.tight_layout()
str_path = str_folder_output + '/' +
'fig_BoxPlotUpdat3.pdf'
plt.savefig(str_path)
plt.close()
```



Examples: Lets have a look into the code

Version control

ClassDataExploring.py

```python
        self.iCurrent_year = self.dT_now.year

    def method_test_plot(self, df):
        string_export_path_findings = self.string_export_path_findings
        # Generate Plot
        fig, ax = plt.subplots()
        ax.spines['right'].set_visible(False)
        ax.spines['top'].set_visible(False)
        ax.yaxis.set_ticks_position('left')
        ax.xaxis.set_ticks_position('bottom')

        df = df[df['category'] == 'depot']

        sns.lineplot(data=df, x='date_valuta', y='amount',
                     color='gold', linewidth=1.0,
                     marker='o', label='Portfolio Value')
        plt.xticks(rotation=80)

        plt.fill_between(df['date_valuta'].values, df['amount'], facecolor='gold', alpha=0.30)

        plt.xlabel('Jahr')
        plt.ylabel('Dividende in Euro')
        plt.xticks(rotation=45)

        fig.set_size_inches(15.0, 12.0, forward=True)
        ax.legend(prop={'size': 6}, frameon=False, loc='upper right')
        plt.savefig('fig_test_pdf')
        plt.close()
```

Import the Data

- The data frame object has some powerful operations
- This makes our life easier to reach the goals

```
import pandas as pd
df = pd.read_csv('...csv')
df = df.dropna()
```

Create a Figure: General Syntax

```
# Access to the library
import matplotlib.pyplot as plt
plt.figure()
plt.plotname(parameters)
plt. . . .
Plt.savefig()
plt.close()
```

Create a new figure

Choose plot

Additional adjustments

Save plot as .png or .pdf

Close plot

General Syntax: DataFrame Data Access

Create a new variable

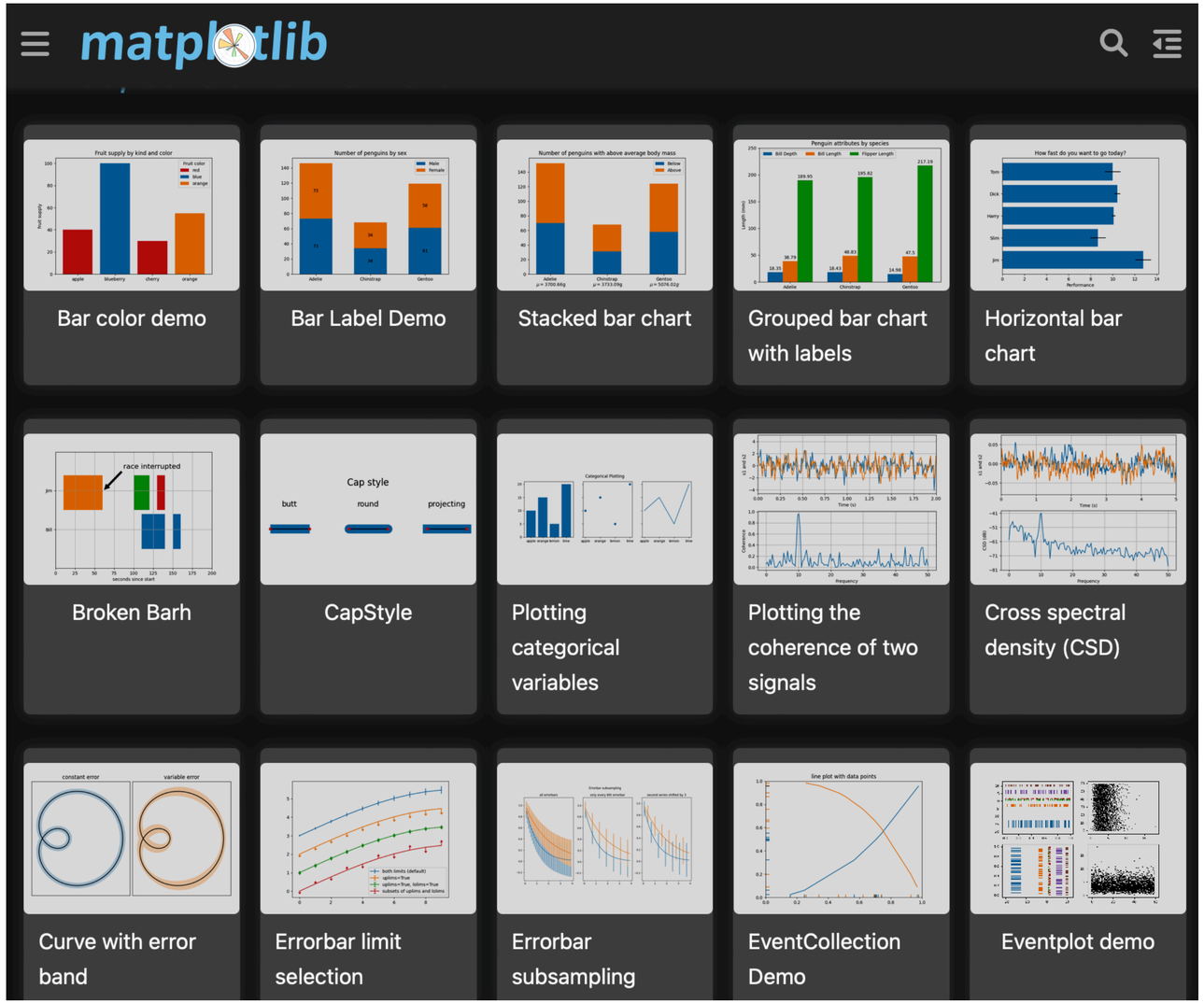
Useful method

```
import pandas as pd
df = pd.read_csv('.csv')
variable = df['Column name'].method()
```

Data frame object with column referencing

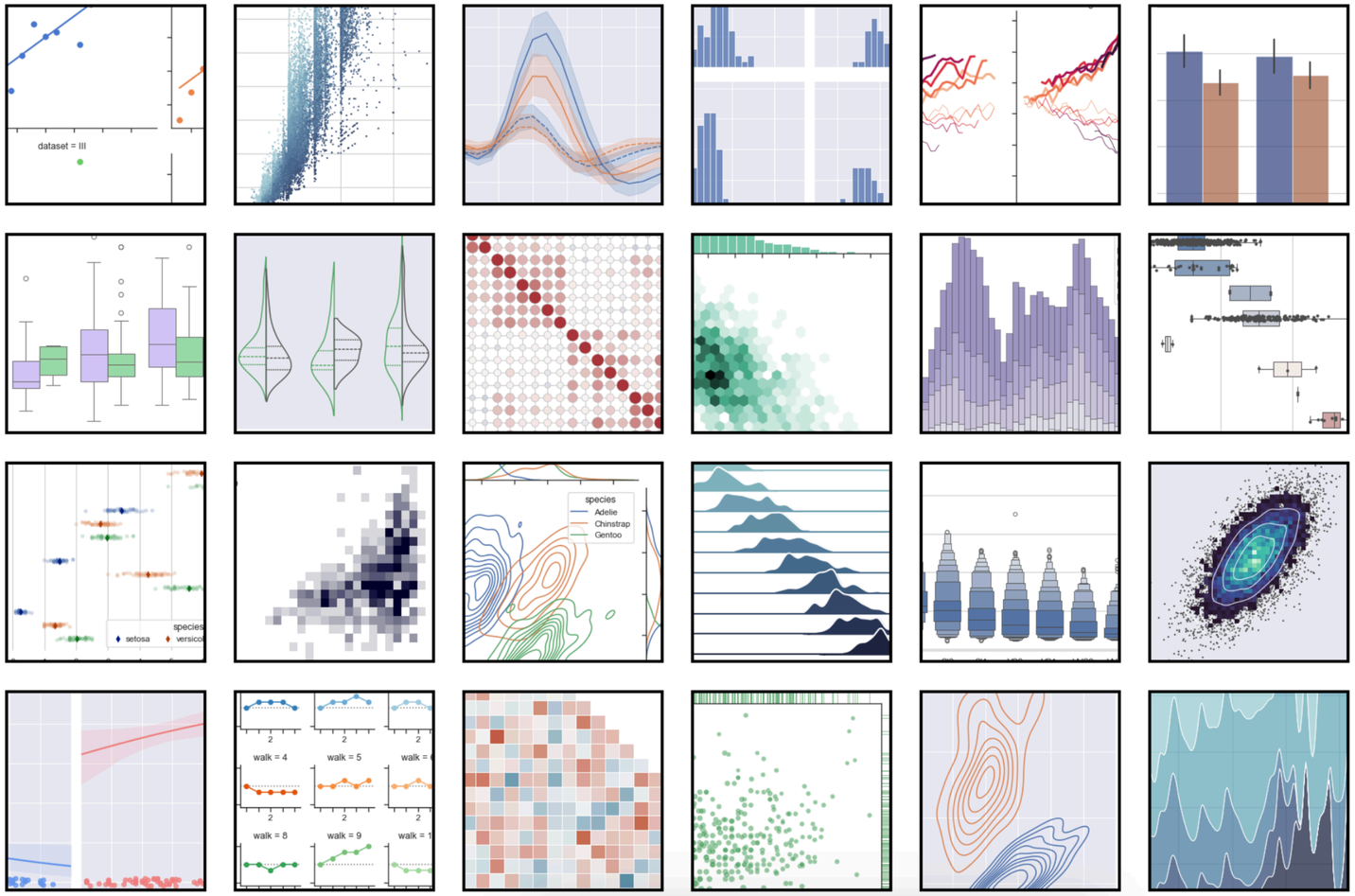
Matplotlib: Visualization with Python

- Matplotlib: Visualization with Python
- Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible. https://matplotlib.org/
- Baseline for Python, easy to use for:
  - Pairwise data, such as scatter
  - Statistical distributions, such as box plot
  - 3D and volumetric data, such as surface



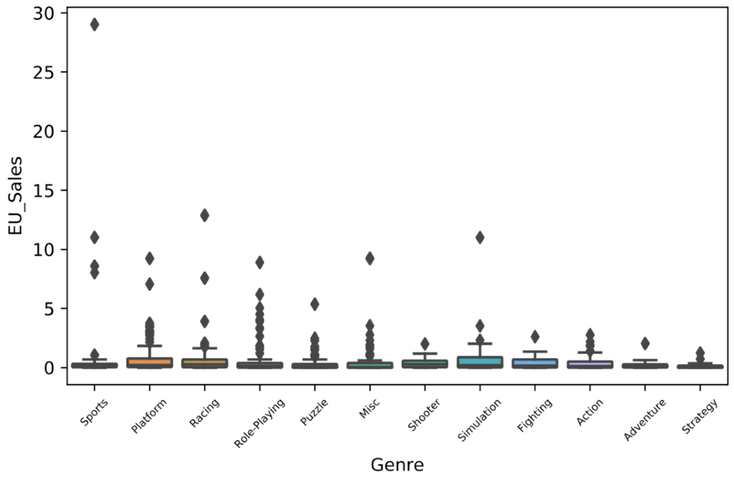
Seaborn: Statistical Data Visualization

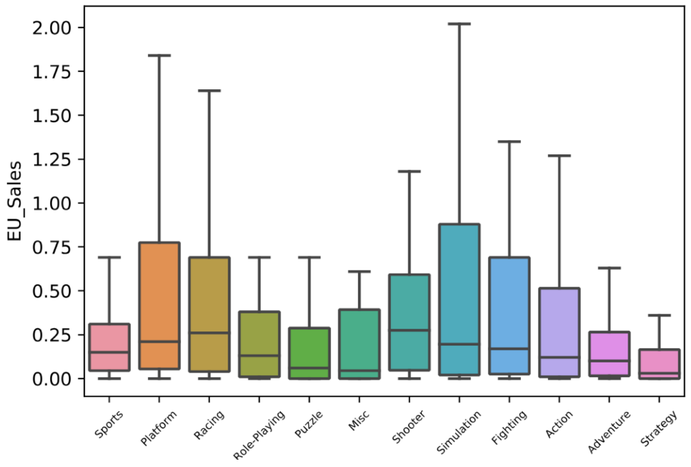
- Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics. https://seaborn.pydata.org/
- An extension of the matplotlib is Seaborn.
- This library can operate directly with data frame objects. For data
science, a useful tool!
- It provides a high-level interface for drawing attractive and informative statistical graphics.
- Easy to use with DataFrames

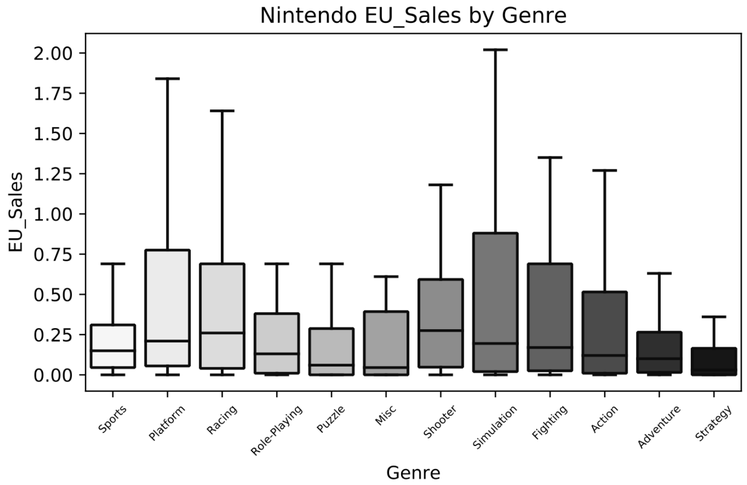


Visualization: Lessons learned

- There is a large number of visualizations.
- However, creating a good visualization is not a trivial undertaking.
- Concentration on the central message is important.







Iterative process

References

- https://www.datacamp.com/community/blog/seaborn-cheat-sheet-python
- https://material.io/design/communication/data-visualization.html#

Cheat Sheets:

- https://github.com/pandas-dev/pandas/blob/master/doc/cheatsheet/Pandas_Cheat_Sheet.pdf
- https://drive.google.com/drive/folders/0ByIrJAE4KMTtaGhRcXkxNHhmY2M
- https://python-graph-gallery.com/cheat-sheets/

Thank You


YouTube
Suchen
Data Science in a Nutshell
Dr. Benjamin M. Abdel-Karim
Data Science in a Nutshell - Einleitung
Keine Aufrufe · 17.08.2021
TEILEN
SPEICHERN


Benjamin M. Abdel-Karim (Hrsg.)
Data Science
Best practices mit Python
EBOOK INSIDE
Springer Vieweg