Data Science II

- Introduction to Data Visualization -

Visualizing Amounts and Distributions

Prof. Dr. Eduard Kromer
Summer Semester 2024
University of Applied Sciences Landshut

Visualizing Amounts

Visualizing Amounts

- we have a set of categories and a quantitative value for each category
- visualizations focus on the magnitude of the quantitative values
- standard visualization: bar plot (simple bars, grouped and stacked bars)
  - alternatives: dot plot and heatmap

Categories and Quantitative Values

US Box Office for February 19, 2022

Data as of Feb 19, 9:50 PST

Key: New This Day Estimated

TD	YD	Release	Daily	%± YD	%± LW	Theaters	Avg	To Date	Days	Distributor
1	-	Uncharted	$15,400,000	-	-	4,275	$3,602	$15,400,000	1	Sony Pictures Entertainment (SPE)
2	-	Dog	$5,044,000	-	-	3,677	$1,371	$5,044,000	1	United Artists Releasing
3	1	Death on the Nile	$1,768,000	+144.2%	-65.1%	3,280	$539	$20,495,032	8	20th Century Studios
4	3	Spider-Man: No Way Home	$1,715,000	+197.8%	-10.6%	2,956	$580	$764,635,686	64	Sony Pictures Entertainment (SPE)
5	2	Jackass Forever	$1,470,000	+114%	-48.9%	3,071	$478	$43,013,627	15	Paramount Pictures
6	4	Marry Me	$1,080,000	+120.3%	-63.5%	3,643	$296	$14,202,665	8	Universal Pictures
7	-	The Cursed	$594,000	-	-	1,687	$352	$594,000	1	LD Entertainment
8	7	Sing 2	$590,000	+269%	-12.7%	2,476	$238	$145,108,985	59	Universal Pictures
9	5	Scream	$515,000	+124%	-37.4%	1,907	$270	$75,570,539	36	Paramount Pictures
10	6	Blacklight	$470,000	+106.2%	-61.9%	2,772	$169	$5,771,030	8	-
11	19	Encanto	$77,000	+729.5%	+229.2%	1,330	$58	$94,469,067	87	Walt Disney Studios Motion Pictures
12	13	West Side Story	$77,000	+159%	+15.1%	955	$80	$37,448,749	71	20th Century Studios
13	10	Belfast	$45,000	+8.7%	-47.5%	734	$61	$8,183,185	99	Focus Features
14	14	Redeeming Love	$31,000	+28.1%	-71.6%	369	$84	$9,036,825	29	Universal Pictures
15	11	The King's Man	$30,000	-12.3%	-75.3%	265	$113	$37,010,867	59	20th Century Studios
16	17	Nightmare Alley	$13,000	+2.3%	-49.6%	250	$52	$11,087,332	64	Searchlight Pictures
17	23	The 355	$4,000	-34.6%	-87%	229	$17	$14,548,430	43	Universal Pictures

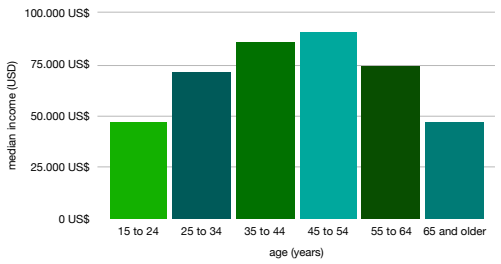
Source: https://www.boxofficemojo.com/date/2022-02-19/

Categories and Quantitative Values

Characteristic	2019			2020			Percent change in real median income (2020 less 2019)*	
	Number (thousands)	Median income (dollars)		Number (thousands)	Median income (dollars)			
		Estimate	Margin of error¹ (±)		Estimate	Margin of error¹ (±)	Estimate	Margin of error¹ (±)
Age of Householder								
15 to 24	5.406	48.532	2.158	5.485	46.886	1.540	-3,4	5,05
25 to 34	20.424	71.161	1.424	20.654	71.566	1.154	0,6	2,15
35 to 44	21.432	89.968	2.563	22.105	85.694	1.712	*-4,8	2,93
45 to 54	21.659	93.372	2.008	21.663	90.359	1.958	*-3,2	2,50
55 to 64	24.603	76.631	1.501	24.336	74.270	2.105	*-3,1	2,45
65 and older	34.927	47.949	923	35.688	46.360	934	*-3,3	2,23

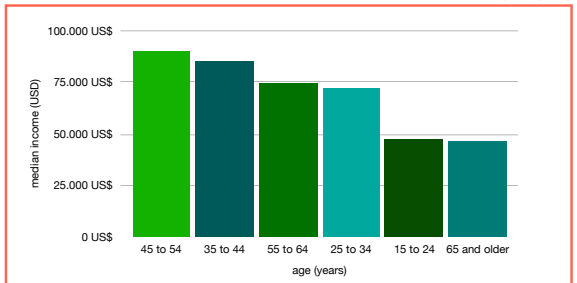
US Census Bureau; 2020 median US annual household income vs age group

Bar Plot



Bar Plot

BAD — DON'T DO THIS

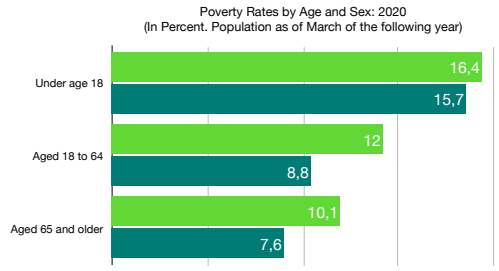


Multiple Categories and Quantitative Values

Poverty Rates by Age and Sex 2020
(In percent. Population as of March of the following year)

	Female	Male
Under age 18	16,4	15,7
Aged 18 to 64	12,0	8,8
Aged 65 and older	10,1	7,6

Grouped Bars

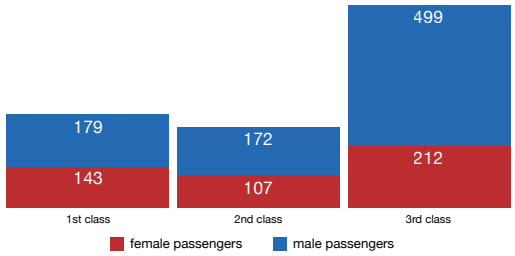


Multiple Categories and Quantitative Values

Female and Male passengers on the Titanic

	female passengers	male passengers
1st class	143	179
2nd class	107	172
3rd class	212	499

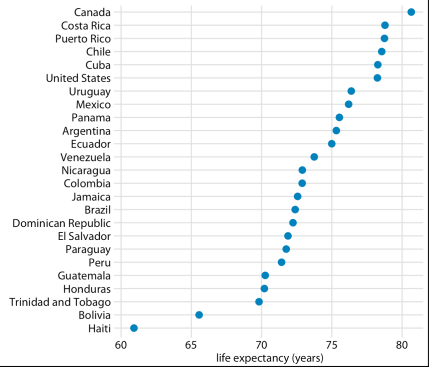
Stacked Bars



Dot Plots

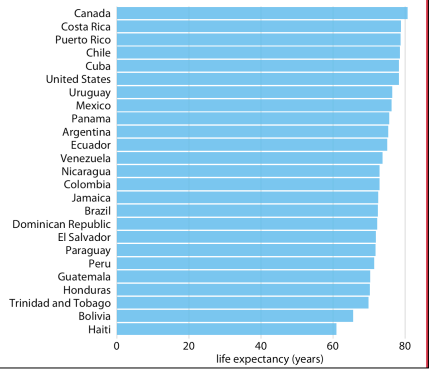
- important limitation of bar plots:
  - they need to start at zero, so that the bar length is proportional to the amount shown
- with dot plots we can indicate amounts by placing dots at the approprate location along the x or y axis

Life expectancies of countries in the Americas for the year 2007



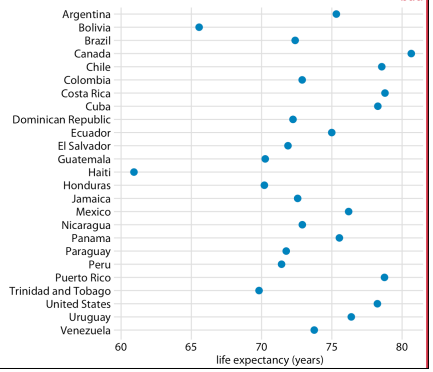
Life expectancies of countries in the Americas for the year 2007

bad



Life expectancies of countries in the Americas for the year 2007

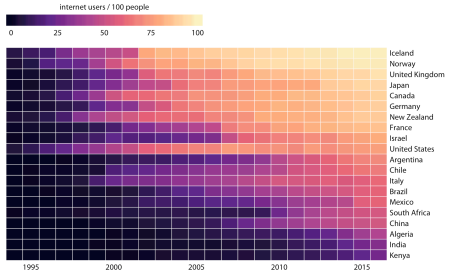
bad



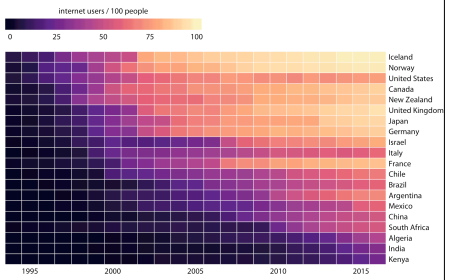
Heatmaps

- as an alternative to mapping data values onto positions via bars or dots, we can map data values onto colors
- while such a visualization makes it harder to determine the exact data values shown, it does an excellent job of highlighting broader trends

Internet adoption over time, for select countries



Internet adoption over time, for select countries



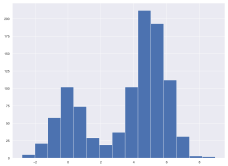
Exercise

1. the following data sets are provided to you in our Moodle course:
   - 2020 median US annual household income vs age group
   - Poverty Rates by Age and Sex 2020
   - Female and Male passengers on the Titanic
   - use those data sets and recreate the bar plots, grouped and stacked bar plots from the previous slides using the *matplotlib* library
2. use the data set in *gender_earnings_disparity.csv* to generate a corresponding dot plot — you can use *matplotlib* or *plotly*

Visualizing Distributions

Histograms

- Wikipedia: "A histogram is an approximate representation of the distribution of numerical data."
- How to construct a histogram?
  - divide the data into discrete bins (series of intervals) and count the number of points that fall in each bin
  - bins are usually specified as consecutive, adjacent, non-overlapping intervals of equal size (they are not required to be of equal size)
  - for bins of equal size: rectangle with height proportional to the frequency is erected over the bin
    - normalization: use relative frequencies; sum of heights equals 1
  - for bins of unequal width: area of erected rectangle is proportional to the frequency of cases in bin



Example: Titanic Data Set

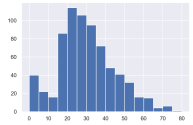
	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
...	...	...	...	...	...	...	...	...	...	...	...	...
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.0000	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.0000	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.4500	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.0000	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.7500	NaN	Q

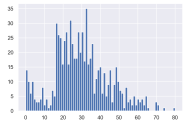
```
import pandas as pd

titanic = pd.read_csv('titanic.csv')
titanic
```

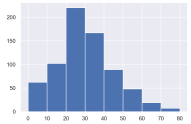
Drawing Histograms

- Use column 'Age' in dataframe `titanic`, divide the data into bins of width 5 (years) and count the number of points that fall in each bin.
- You may use `pandas` and / or `numpy` for counting the frequencies
  - there are many different ways how you can obtain the required counts, e.g. you could use `np.where(…)` or `np.histogram(…)` [read the documentation]
- Draw the corresponding histogram [with pen and paper].
- Confirm the correctness of your drawing by using the `histogram` function of `matplotlib.pyplot`; change the bin width to 1 year and then to 10 years and compare the histograms. What do you notice?
- How does your histogram change if you normalize it?

Drawing Histograms with matplotlib





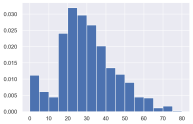


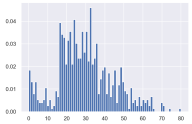
```
import matplotlib.pyplot as plt
import seaborn as sns; sns.set()

age = titanic['Age'].dropna().astype(int)

hist = plt.hist(age, bins=16)
```

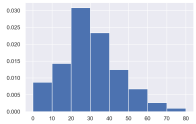
```
hist = plt.hist(age, bins=80)
```

```
hist = plt.hist(age, bins=8)
```

Drawing Histograms with matplotlib







```
import matplotlib.pyplot as plt
import seaborn as sns; sns.set()

age = titanic['Age'].dropna().astype(int)

hist = plt.hist(age, bins=16, density=True)
```

```
hist = plt.hist(age, bins=80, density=True)
```

```
hist = plt.hist(age, bins=8, density=True)
```

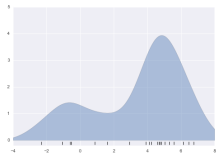
Can you verify (using numpy) the sum of heights of the rectangles equals 1?

Visual Appearance of Histograms

- the visual appearance of histograms depends on the choice of bin width
- the default choice for the bin width (of a visualization program) is probably not the most appropriate one for your data
  - always try different widths and verify that the histogram reflects the underlying data accurately

Kernel Density Estimation

- in a density plot we try to visualize the underlying probability distribution of the data by drawing an appropriate continuous curve
  - we estimate this curve from our data
  - most commonly used estimation method is kernel density estimation
- in our case, a kernel is a non-negative real-valued integrable function $K(x; h)$ which is controlled by the bandwith parameter h



Kernel Density Estimation

- given this kernel form, the density estimate at a point y within a group of points x_i, $i = 1,\ldots,N$ is given by

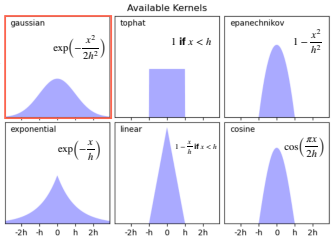
$$\rho_K(y) = \sum_{i=1}^{N} K(y - x_i; h)$$

- the bandwidth parameter acts as a smoothing parameter, controlling the tradeoff between bias and variance in the result
  - a large bandwidth leads to a very smooth (high bias) density distribution
  - a small bandwidth leads to an unsmooth (high variance) density distribution

Source: Scikit-Learn Documentation

Kernel Forms in scikit-learn

most widely used



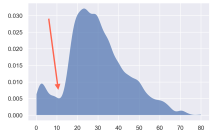
Visual Appearance of a Density Plot

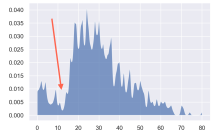
- the density plot depends on the choice of kernel and bandwidth
- the bandwith parameter behaves similarly to the bin width in histograms
  - bandwidth large: smaller features in the distribution of the data may dissapear
  - bandwidth small: main trends may be obscured

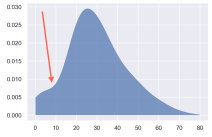
Visual Appearance of a Density Plot

- use `KernelDensity` from `sklearn.neighbors` with the Gaussian kernel to estimate the density of the Titanic age data you used to generate the histograms
  - use different bandwidths (e.g. 0.5, 2, and 5) and plot the estimated densities using `matplotlib.pyplot.fill_between`
- experiment with different kernels and visualize your results

Gaussian Kernel with Different Bandwidths







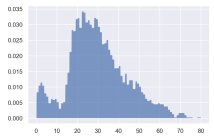
```
from sklearn.neighbors import KernelDensity
```

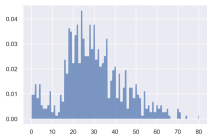
```
kde = KernelDensity(bandwidth=0.5, kernel='gaussian')
```

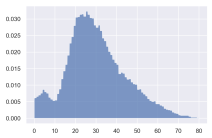
```
kde = KernelDensity(bandwidth=5, kernel='gaussian')
```

```
kde = KernelDensity(bandwidth=2.0, kernel='gaussian')
kde.fit(age.values.reshape(-1,1))

x = np.linspace(0, 80, 1000)
logprob = kde.score_samples(x[:, None])

plt.fill_between(x, np.exp(logprob), alpha=0.7);
```

Tophat Kernel with Different Bandwidths





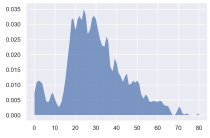


```
kde = KernelDensity(bandwidth=2, kernel='tophat')
```

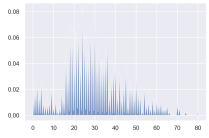
```
kde = KernelDensity(bandwidth=0.5, kernel='tophat')
```

```
kde = KernelDensity(bandwidth=5, kernel='tophat')
```

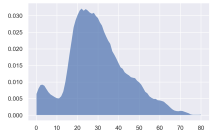
Linear Kernel with Different Bandwidths




```
kde = KernelDensity(bandwidth=3, kernel='linear')
```

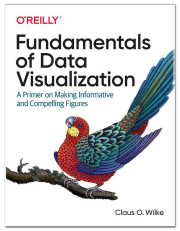



```
kde = KernelDensity(bandwidth=0.5, kernel='linear')
```




```
kde = KernelDensity(bandwidth=5, kernel='linear')
```

Literature



References

- Slide 13-15, 17,18; Image Source: Claus O. Wilke - Fundamentals of Data Visualization, O'Reilly
- Slide 29; Image Source: https://scikit-learn.org/stable/modules/density.html