Data & Things
(Spring 25)

Wednesday February 5

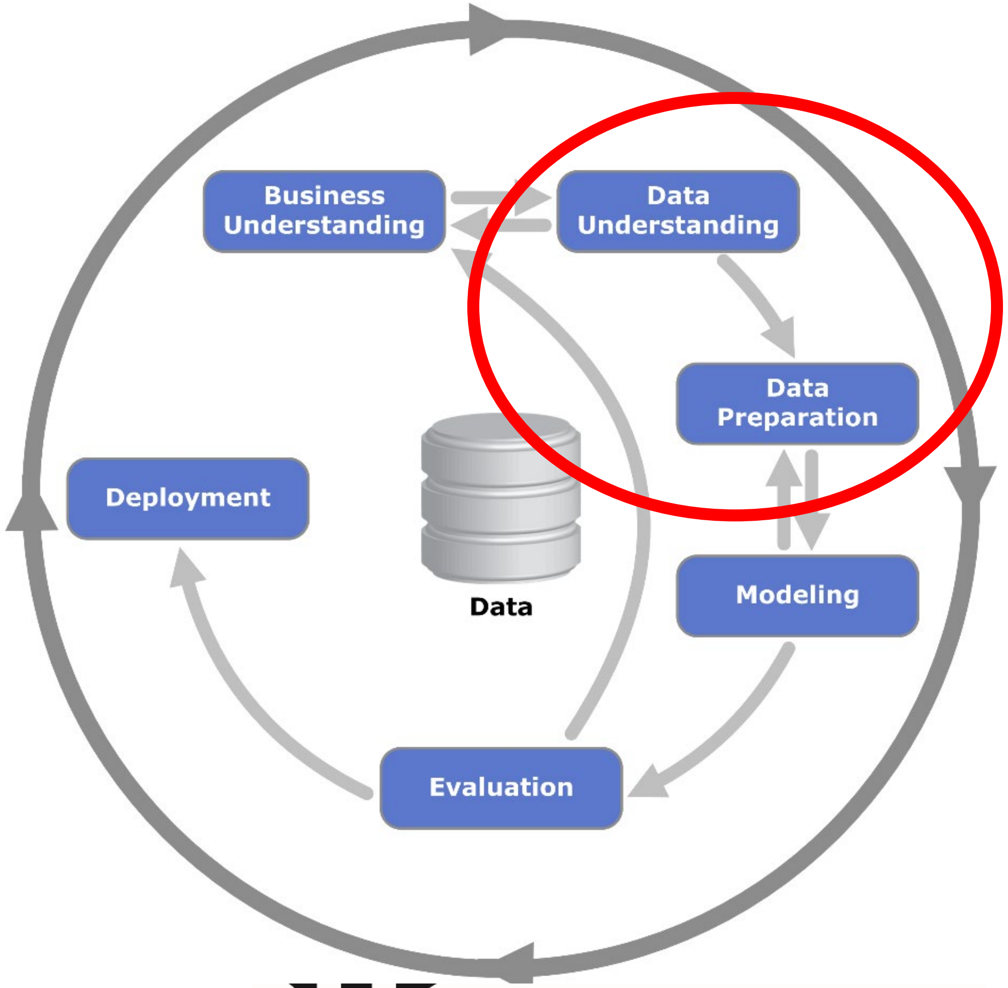
Lecture 2: Data transformation and exploratory data analysis

Jens Ulrik Hansen

Roskilde Universitet

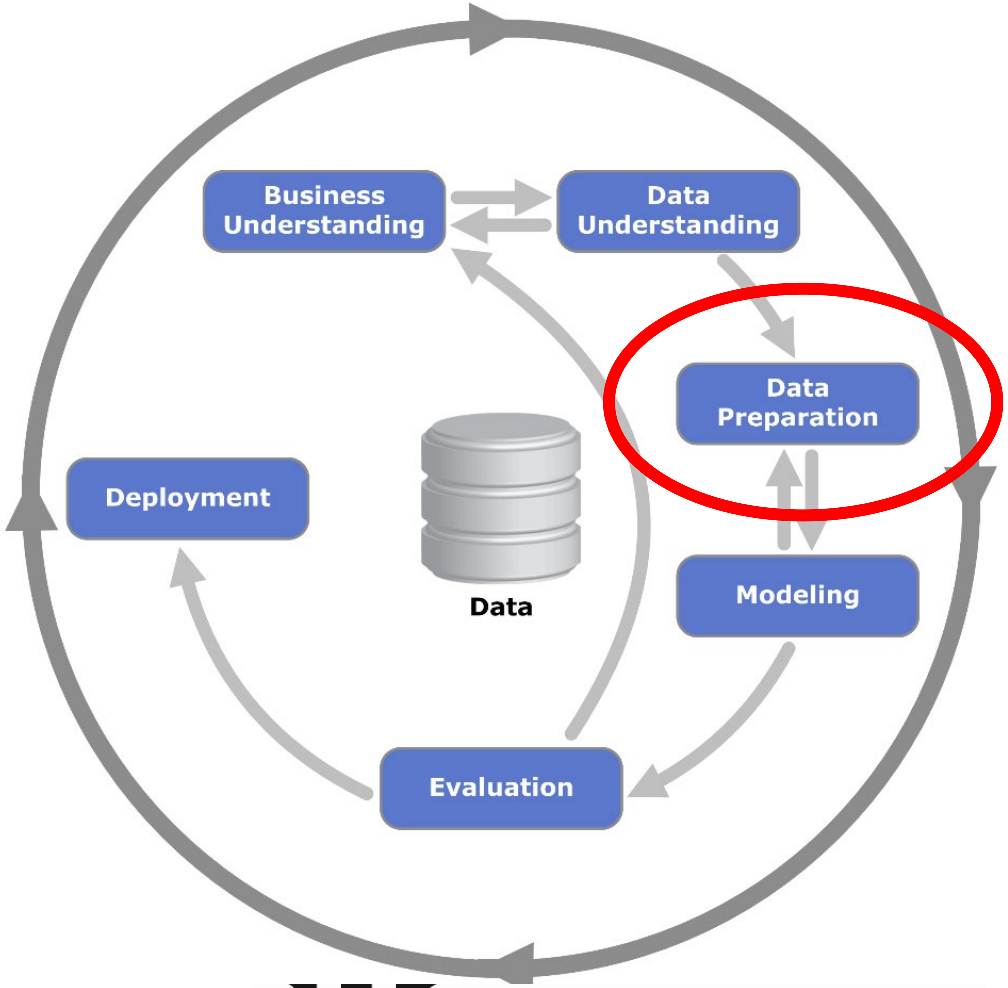
Data transformation and exploratory data analysis

- **<u>The Data Science process</u>**
  - CRISP-DM: Cross-industry standard process for data mining (https://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining)
  - Today we are going to focus on the steps “Data Understanding” and “Data Preparation”
  - More specifically, we are going to focus on Data Transformation and Exploratory Data Analysis (EDA)



Data transformation and exploratory data analysis

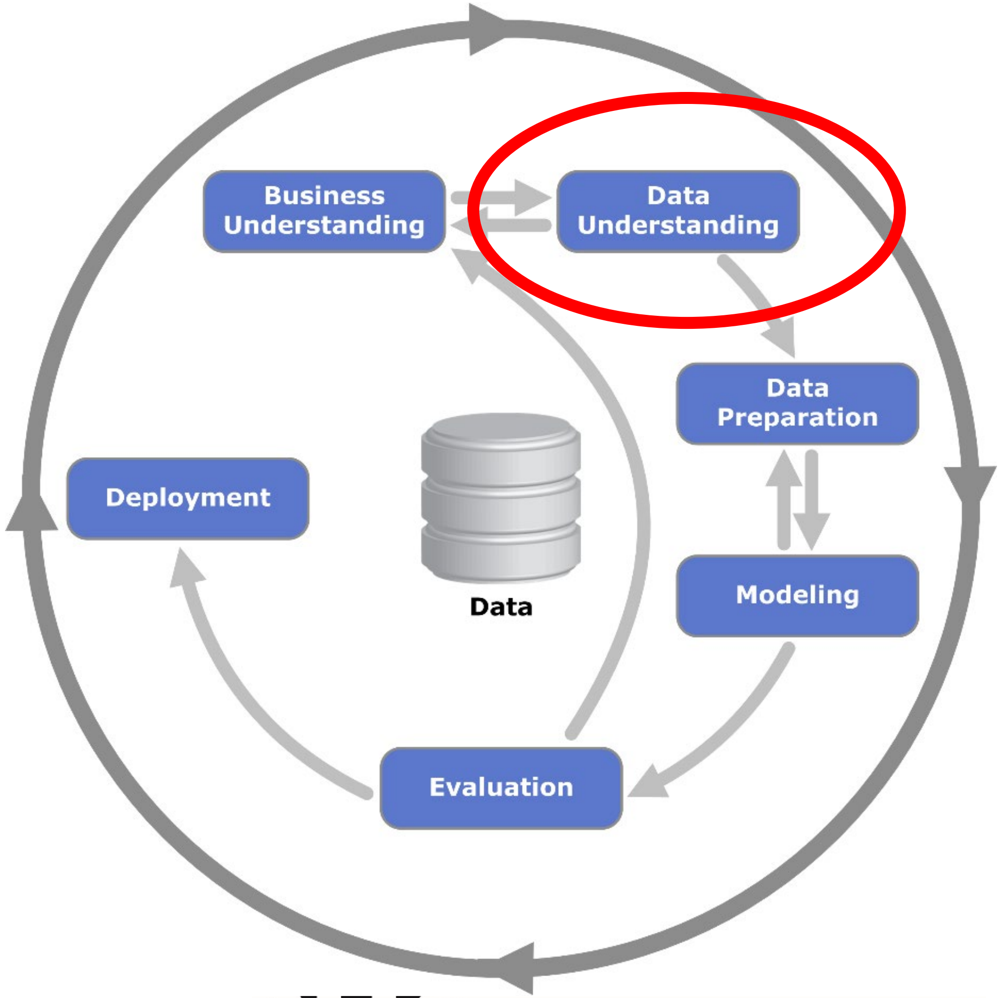
- **The Data Science process**
  - CRISP-DM: Cross-industry standard process for data mining (https://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining)
  - Data Transformation
    - Other words for data transformation (or similar steps): *Data preparation, data wrangling, data cleaning, ETL (extract, transform, load)*
    - It is all about making the data ready for further analysis/modeling (and sometimes to enable the exploratory data analysis)
    - It involves: Correcting formatting issues, restructuring the data, creating new features, dealing with missing values, …



Roskilde Universitet

Data transformation and exploratory data analysis

- **The Data Science process**
  - CRISP-DM: Cross-industry standard process for data mining (https://en.wikipedia.org/wiki/Cross-industry_standard_process_for_data_mining)
  - Exploratory Data Analysis (EDA)
    - EDA help us understand the data we have
    - This is essential for:
      - Figuring out whether our data can solve the (business) problem we are aiming to solve
      - Getting an idea about what such a solution will look like
      - Figuring out what data transformation is needed for further analysis
    - EDA usually involves calculating a lot of descriptive statistics and making a lot of visualizations



Roskilde Universitet

Outline of this lecture

- Data types
- Data transformation and data cleaning
- Data visualization
- Exploratory Data Analysis (EDA)
- Exercises

Data types

- Data can come from multiple sources
  - books and paper record, surveys, sensors, the web, business IT-systems and databases (ERP and CRM systems, etc.), Social media, cameras, our brains and genes, etc…
- Data can be in many different formats
  - tables/spreadsheets, relational databases/SQL, No-SQL databases, plain text, XML, JSON, graphs, documents, images, videos, … etc.
- Tabular data (spreadsheet format) is structured data that is “easy” to work with contrary unstructured data such as images and text
  - A particular form of tabular data where **rows represent cases/observations/objects** and **columns represent attributes/variables/features** is especially useful for data analysis and machine learning (sometimes referred to as “tidy data” – a term coined by Hadley Wickham)

Data types

- Tidy tabular data (spreadsheet format)
  - Rows represent cases/observations/objects and columns represent attributes/variables/features
  - Examples of cases: A persons, a censor measurement, a transaction
  - Examples of attributes: Eye color of a person, temperature of a sensor at particular time, the costumer of the transaction
  - Every cell represent one piece of information
  - Every column have same number of entries
  - Each observation contains all values measured on that same unit/individual across attributes
  - It is not always obvious what are observations and what are variables
    - However, a general rule of thumb: Easier to describe functional relationships between variables. Easier to make comparison between groups of observations.
  - Tidy tabular data is naturally stored in pandas DataFrames in Python

Data types

- Types of data – scale of measurements (from a semantic perspective)

categorical

  - ***Nominal***
    - Examples: ID numbers, eye color, zip codes
  - ***Ordinal***
    - Examples: rankings (e.g., taste of potato chips on a scale from 1 to 10), grades, height in {tall, medium, short}

numerical

  - ***Interval***
    - Examples: calendar dates, temperatures in Celsius or Fahrenheit.
  - ***Ratio***
    - Examples: temperature in Kelvin, length, time, counts

Data types

Provides:	Nominal	Ordinal	Interval	Ratio
The "order" of values is known		✔	✔	✔
"Counts," aka "Frequency of Distribution"	✔	✔	✔	✔
Mode	✔	✔	✔	✔
Median		✔	✔	✔
Mean			✔	✔
Can quantify the difference between each value			✔	✔
Can add or subtract values			✔	✔
Can multiple and divide values				✔
Has "true zero"				✔

Data types

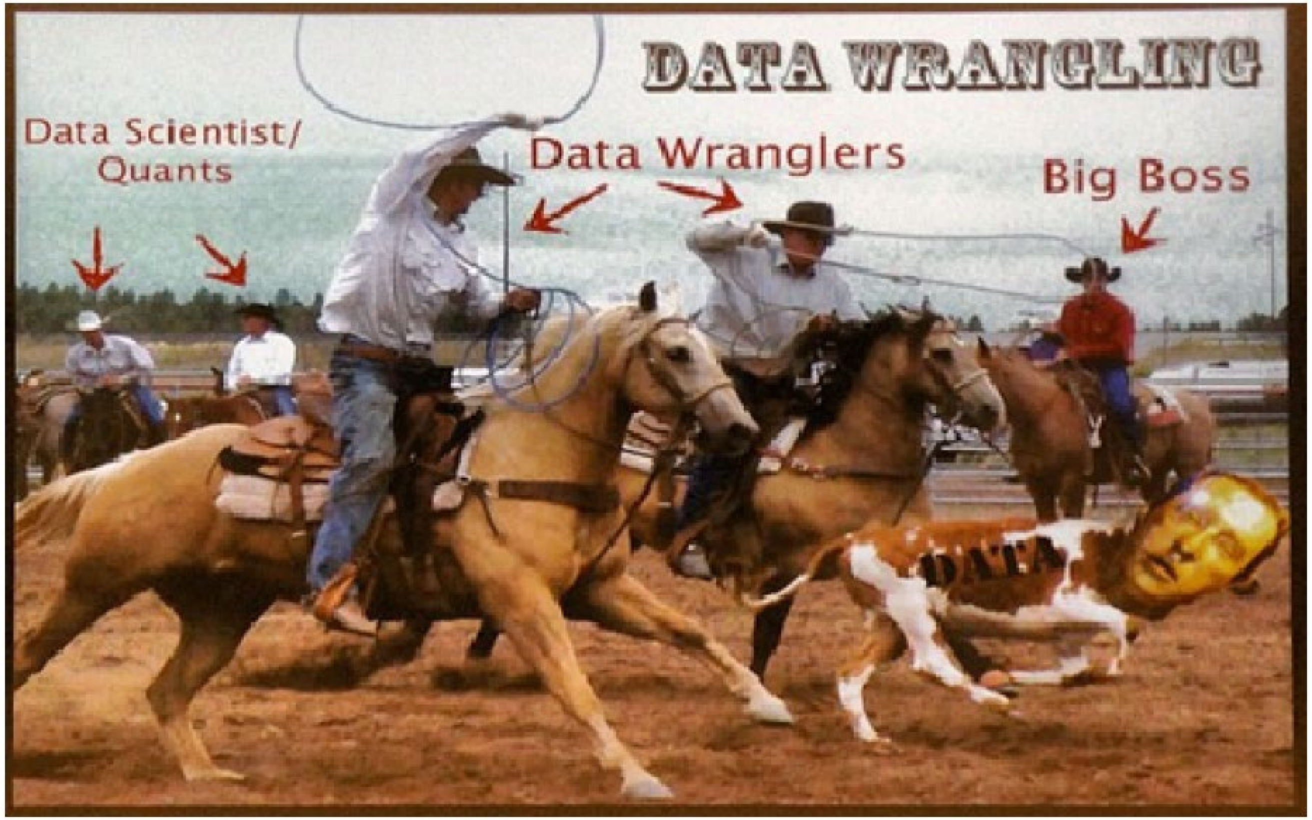
- Data types in Python (from a syntactic perspective)
  - Each column of a DataFrame can contain data as:
    - Integers, floats, dates, date-times, strings, … etc.
  - Categorical data can both be represented as strings or as integers
    - If it is strings, it appears as “object” when one does “.info()” on a dataframe
  - Numerical data is often represented as floats (but sometimes integers) or as date-times (if it is data-times)

Outline of this lecture

- Data types
- Data transformation and data cleaning
- Data visualization
- Exploratory Data Analysis (EDA)
- Exercises

Data transformation and data cleaning

- A time-consuming task, more art than science
- We will look at
  1. Transforming data
  2. Dealing with missing values and outliers



Roskilde Universitet

Data transformation and data cleaning

1. **Transforming data**
   - Grouping and aggregation
     - Generating aggregated or summarized measure for individual groups (often based on a categorical variable)
   - Joins
     - Combine multiple data frames on particular keys
     - Inner, Outer, left, and right
   - Pivoting
     - Making sure that rows represent cases and columns features
     - From long to wide, and from wide to long

Data transformation and data cleaning

1. **Transforming data**
   - Removing Duplicates
     - Sometimes useful, but I have rarely seen it for machine learning models (One has to know that a data entry has been duplicated – that two rows are identical might just mean that the cases have the same features – two student might have taken the same classes and gotten the same grade, for instance)
     - See the book for how to do this, if necessary
   - Creating new columns from mapping
     - Use to group categorical variables with many groups to fewer high-level group
   - Replacing specific values with other values
     - We will talk about this when we talk about replacing missing values or outliers
   - Discretization and Binning
     - Useful, when one wants to turn a numeric age variable into a categorical age group variable, for instance – often not that used in machine learning, but more in descriptive statistics
     - See the book for how to do this, if necessary

Data transformation and data cleaning

1. **Transforming data**
   - Transforming a categorical variable into dummy variables
     - Some machine learning models (most!) cannot deal directly with categorical variables. Instead, one needs to encode them as “dummy variables”: A categorical variable X with values “catA”, “catB”, “catC” are turned into three boolean variables catA, catB and catC, such that catA is 1 if X is “catA” and 0 otherwise, catB is 1 if X is “catB” and 0 otherwise, and catC is 1 fi X is “catC” and 0 otherwise.
   - String manipulation
     - Having string values in a variable can potentially create a lot of work if it is not a simple categorial variable
     - If it is a categorical variable, one just wants to make sure that countries are spelled the same – we do not both have “Denmark”, “denmark”, and “Danmark”.
     - If the variable contains more elaborate text data like tweets or free text answers to a questionnaire question, it is not to be considered as a categorical variable. If one wants to analyze this type of data, one has to do NLP (Natural Language Processing), which is beyond the scope of this course.
   - Categorical Data
     - Pandas also contains a specific data type to deal with categorical data. It is inspired by the factor data type in R, which play a large and important role in R.
     - It has not been widely adopted in the Python community as far as I know, so we will skip talking about it here.

Data transformation and data cleaning

1. **Transforming data**
    - Let us look at the notebook “Data transformation.ipynb”

Data transformation and data cleaning

2. **Dealing with missing values and outliers**
   - **Garbage-in-garbage-out (GIGO):** If your data is of poor quality, then any analysis you base upon it, will be poor, as well
     - Thus, you need to make the data as good as it can get before you start your analysis
     - During an analysis, you might need to go back and improve the quality of your data
   - **The right/best way to fix/clean your data might depend on the problem of your data analysis**
   - **Missing values** can affect your analysis greatly
   - So can **outliers**

Data transformation and data cleaning

- **Missing values** can be:
  - **Explicitly missing** (in Python represented by nan, null, etc.)
  - **Implicitly missing** (there is no record of it in the data)

	year	quarter	return
1	2015	1	1.88
2	2015	2	0.59
3	2015	3	0.35
4	2015	4	NA
5	2016	2	0.92
6	2016	3	0.17
7	2016	4	2.66

Data transformation and data cleaning

- **What are missing values?**
  - An explicit nan may represent that the data is simply not available
  - An implicit missing value might represent an error in the data
  - In either case, it usually means that the data is not available.
  - That the data is not available can mean:
    - The data was not collected
    - The data is lost
    - The data does not exist
    - The data cannot exist

Data transformation and data cleaning

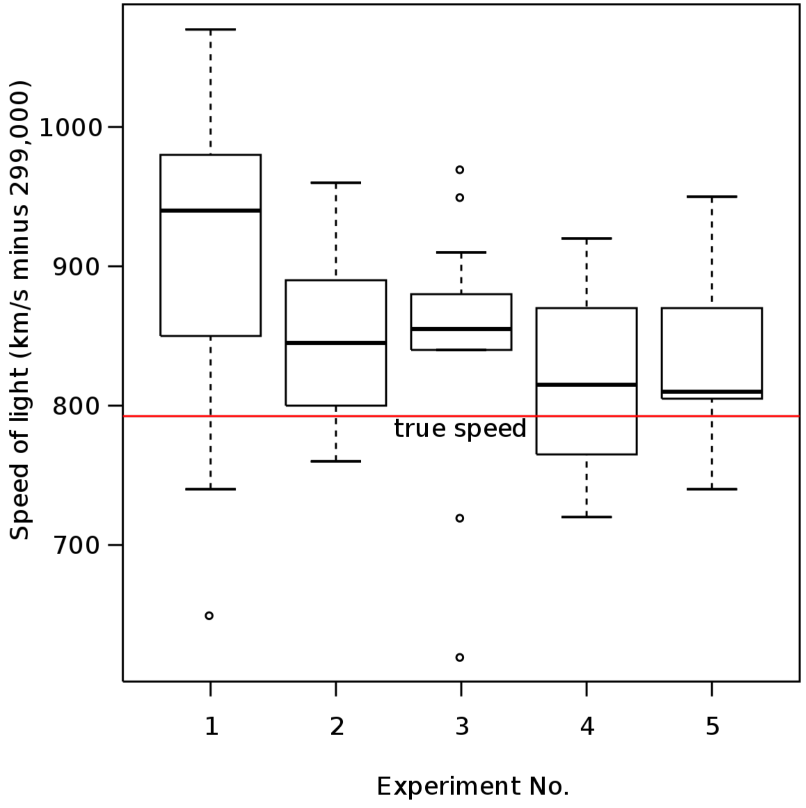
- **How to treat missing values?**
  - The reason for the missing data and what type of data it is, have implications for how to treat the missing values
    - For instance, in weather data it might make sense to impute data by the mean
    - In sales data, it might make sense to replace a missing value with 0
    - The context decide!
  - Note, however, several functions or methods in Python can ignore missing values (such as Pandas DataFrame.describe()), and depending on the type and frequency of the missing values, the result might be fine by just ignoring the missing values!
  - There is a whole subfield dealing methods for **imputing** missing values. One could for instance use predictive machine learning models to come up with suggestions for imputation values.

Data transformation and data cleaning

- **Outliers**
  - Outliers are values that fall well outside the central tendency of your variables (as can be seen in boxplots)
  - Outliers can represent different things:
    - Special extreme events. In betting data, the Football world cup final will likely generate an outlier; or a closing day in transaction data
    - Errors in data. A sales entry of a million times the usual sale is probably an error
  - Sometimes special tricks can be used to deal with an outlier
    - In the first case, we want the information present
    - In the second case, we might want to replace the outlier by a missing value and deal with it as a such

Data transformation and data cleaning

- **Detecting outliers**
  - There is not one definitive definition of what is an outlier – it depends on the context!
  - This is now a very operational message, however several criteria has been devised
    - Ssee https://en.wikipedia.org/wiki/Outlier and https://en.wikipedia.org/wiki/Box_plot, for instance.
  - One is to define outliers to be every points below or above the whiskers of a box plot:
    - The whiskers of a boxplot are defined by:
      - The lower: max(25thQuantile(data) - 1.5*IQR(data), min(data))
      - The upper: min(75thQuantile(data) + 1.5*IQR(data), max(data)),
    - where the inter quartile range (IQR) is defined by:
      - IRQ = 75thQuantile(data) - 25thQuantile(data)



Roskilde Universitet

Data transformation and data cleaning

2. **Dealing with missing values and outliers**
    - Let us look at the notebook “Missing values and outliers.ipynb”

Outline of this lecture

- Data types
- Data transformation and data cleaning
- Data visualization
- Exploratory Data Analysis (EDA)
- Exercises

Data visualization

- Let us look at the notebook “Visualizing data.ipynb”

Outline of this lecture

- Data types
- Data transformation and data cleaning
- Data visualization
- Exploratory Data Analysis (EDA)
- Exercises

Exploratory Data Analysis (EDA)

- EDA – what is it?
  - Wikipedia: ***"In statistics, exploratory data analysis (EDA) is an approach of analyzing data sets to summarize their main characteristics, often using statistical graphics and other data visualization methods."***, https://en.wikipedia.org/wiki/Exploratory_data_analysis, retrieved 2024-02-04
  - R for Data Science: ***"EDA is not a formal process with a strict set of rules. More than anything, EDA is a state of mind."***, https://r4ds.hadley.nz/eda, retrieved 2024-02-04
- EDA is about
  - Getting to know/understand your data
  - Getting to know and improve the quality of your data
- EDA often involves
  - A lot of plotting and visual exploration
  - summary statistics (descriptive statistics) of the data
- **EDA is the initial necessary stage of getting to know your data**

Exploratory Data Analysis (EDA)

- Exploratory Data Analysis – step 1:
  - What variables/features are in your data set?
  - What does the different variables represent?
  - What observations does the data set contain?
  - What is the (intended) type/scale of measurement of each of the variables/features in your data set
- Step 2: Standard questions about your data
  - What type of variation occurs within my variables?
  - What type of variation occurs between my variables?
  - What is the quality of my data? Are there outliers, missing data etc.?
- Step 3 – only your imagination set the limitation for what you can ask about your data…

Exploratory Data Analysis (EDA)

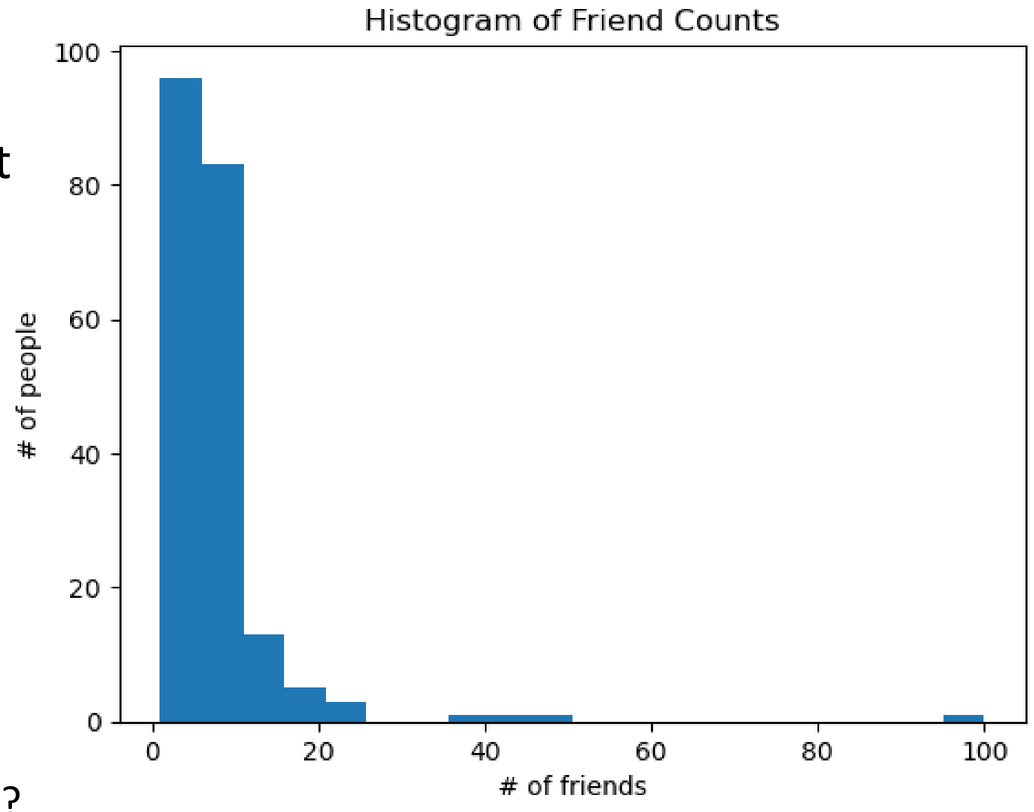
- Exploratory Data Analysis – step 1:
  - ~~What variables/features are in your data set?~~
  - ~~What does the different variables represent?~~
  - ~~What observations does the data set contain?~~
  - ~~What is the (intended) type/scale of measurement of each of the variables/features in your data set~~
- Step 2: Standard questions about your data
  - What type of variation occurs within my variables?
  - What type of variation occurs between my variables?
  - ~~What is the quality of my data? Are there outliers, missing data etc.?~~
- Step 3 – only your imagination set the limitation for what you can ask about your data…

Exploratory Data Analysis (EDA)

- **Variation within a variable**
  - The variation within a variable is the tendency of the variable to change from observation to observation
  - A variation within a variable can be due to several things
    - The variable simply varies within the population: If we measure the height of people in this room, we will get different heights for different people
    - There can be measurement errors or noise: If we measure the same person several times with very high accuracy, we are bound to get a little bit of different heights every time
  - The variation of a variable can be understood through its distribution, which can be visualized as well as quantified . . .

Exploratory Data Analysis (EDA)

- **Visualizing variation within a variable**
  - For categorical variables use a bar plot
  - For numerical variables: Use a histogram or a boxplot
- **Quantifying variation within a variable**
  - With descriptive statistics we can measure the tendencies we see in the distribution plots
  - For categorical variables: a table of the numbers of each category (or maybe the most frequent value (the mode))
  - For numerical variables:
    - ***Centrality tendencies:*** Where do most values fall? What values am I most likely to get if I draw a random value from the distribution?
    - ***Variation/spread tendencies:*** How spread out is my data? What are the most extreme values I can get by a random draw? How far from the "centrality" of the distribution am I like to end up by a random draw?



Exploratory Data Analysis (EDA)

- **Centrality tendencies**
  - The ***mean*** of a distribution:
    - (Also known as the arithmetic mean or the average, in Danish "gennemsnit" or "middelværdi")
    - The sum of all values divided by the number of values:
      - $sum(x_1 + x_2 + \ldots + x_n)/n$
    - In numpy, there is a function *np.mean()*
    - In pandas, Series and DataFrames has methods called *.mean()*
    - Both of these ignore missing values
    - The mean is sensitive to outliers or extreme values

1
2
5
6
6

Mean =
(1+2+5+6+6)/5 = 4

Exploratory Data Analysis (EDA)

- **Centrality tendencies**
  - The ***median*** of a distribution:
    - The value such that half of the of values are below that value and the other half are above that value
    - If you sort the list of values, the median is the middle value
    - In numpy, there is a function *np.median()*
    - In pandas, Series and DataFrames has methods called *.median()*
    - Note that np.median() do not ignore missing values, while *.median()* does. Numpy has nanmedian() that ignores missing values.
    - The median is not sensitive to outliers or extreme values

1
2
5
6
6

Mean =
(1+2+5+6+6)/5 = 4
Median = 5

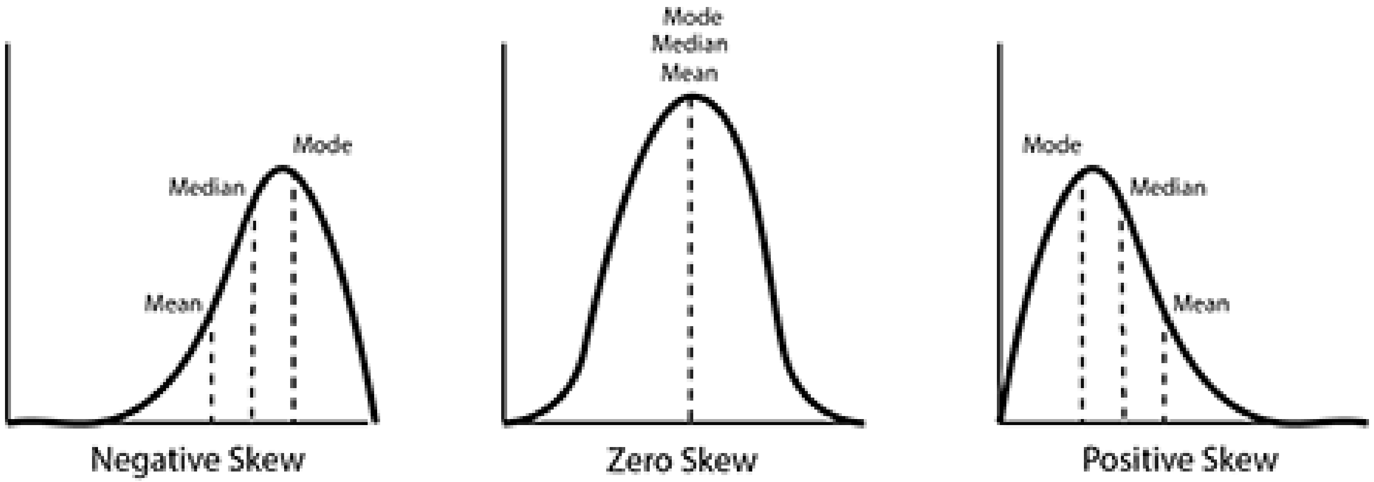
Roskilde Universitet

Exploratory Data Analysis (EDA)

- **Centrality tendencies**
  - ***Quartiles*** of a distribution:
    - Like the median, but with “different location than half way”.
    - The ***first quartile*** is such that 25% of the values are below this value (and 75% above)
    - The ***second quartile*** is the median
    - The ***third quartile*** is such that 75% of the values are below this value (and 25% above)
  - ***Quantiles*** of a distribution:
    - We can also talk about ***quantiles*** in general, such as the 35% quantile, i.e. the value such that 35% of the values are below this value
    - The ***first quartile*** is the same as the 25% ***quantile***, etc.
    - Quantiles can be calculated using the numpy functions *quantile* and *nanquantile*.
    - There is also a *.quantile* method on pandas Series and DataFrames. (That ignores missing values)

Exploratory Data Analysis (EDA)

- **Mean vs. Median**
  - These values can be very different, especially when the distribution is ***skewed***
    - ***Left skewed (negative skewed):*** The tail to the left is longer than the tail to the right – the mean is less than the median
    - ***Right skewed (positive skewed):*** The tail to right is longer than the tail to the left – the mean is greater than the median



Roskilde Universitet

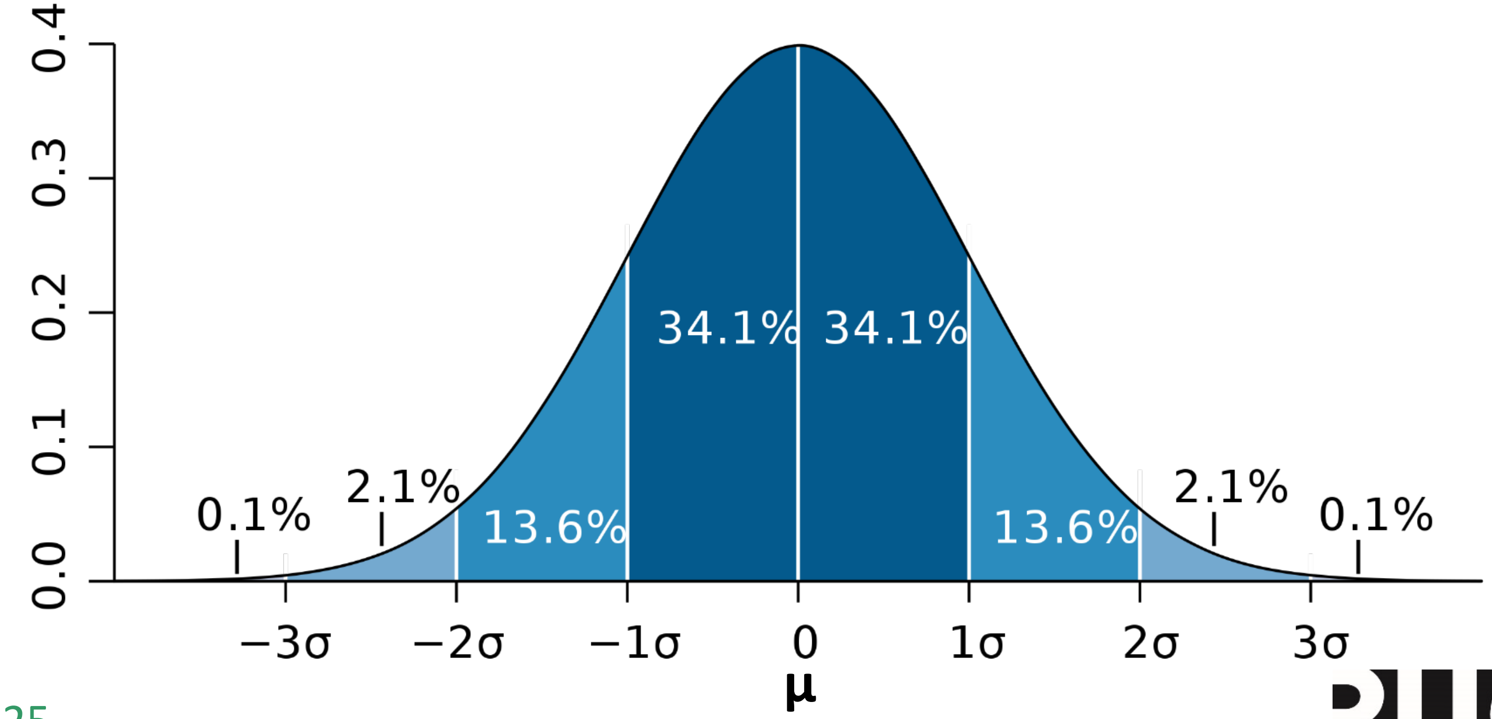
Exploratory Data Analysis (EDA)

- **Variation/spread tendencies**
  - The ***range*** of a distribution:
    - The minimum and maximum values. (Sometimes the difference between the max and the min can also be of interest.)
  - The ***variance*** of a distribution:
    - A measure for how far from the mean values of variable are:
    - $sum((x_1 - mean)^2 + (x_2 - mean)^2 + \ldots +(x_n - mean)^2) / (n - 1)$
    - Note the “(n - 1)” it is not an error, it has to do with the *degrees of freedom* (we will not get into this any further)
    - Note, the square makes the signs not important and make extreme values more important
    - In numpy there is a *var* function and in pandas there is a *.var* method
  - The ***standard deviation*** of a distribution:
    - Note that the unit of the variance is the square of the unit of the variable. Sometimes it is nice to have measure in the same unit as the variable
    - The standard deviation is the square root of the variance
    - In numpy there is a *std* function and in pandas there is a *.std* method

Roskilde Universitet

Exploratory Data Analysis (EDA)

- **The normal distribution**
  - **μ** (my) denotes the mean
  - **σ** (sigma) denotes the standard deviation

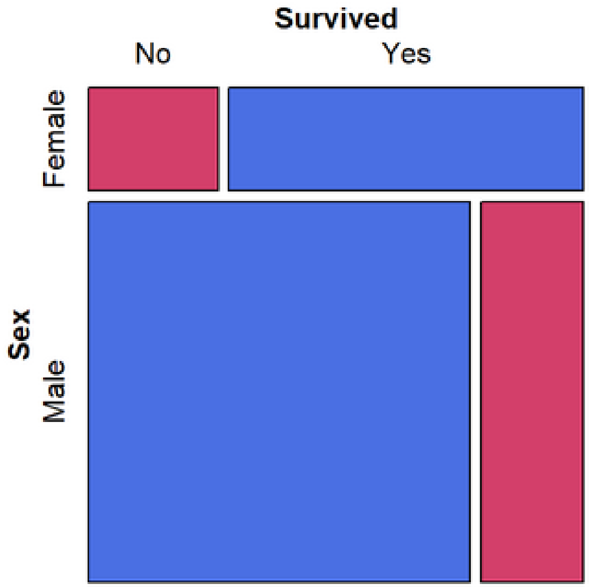


Exploratory Data Analysis (EDA)

- **Variation between two variables**
  - Covariance/correlation/causation
  - How to plot and what descriptive statistics to look at, depends on what are the types of the involved variables. There are the following three cases:
    - Two categorical variables
    - Two numerical variables
    - One categorical variable and one numerical variable

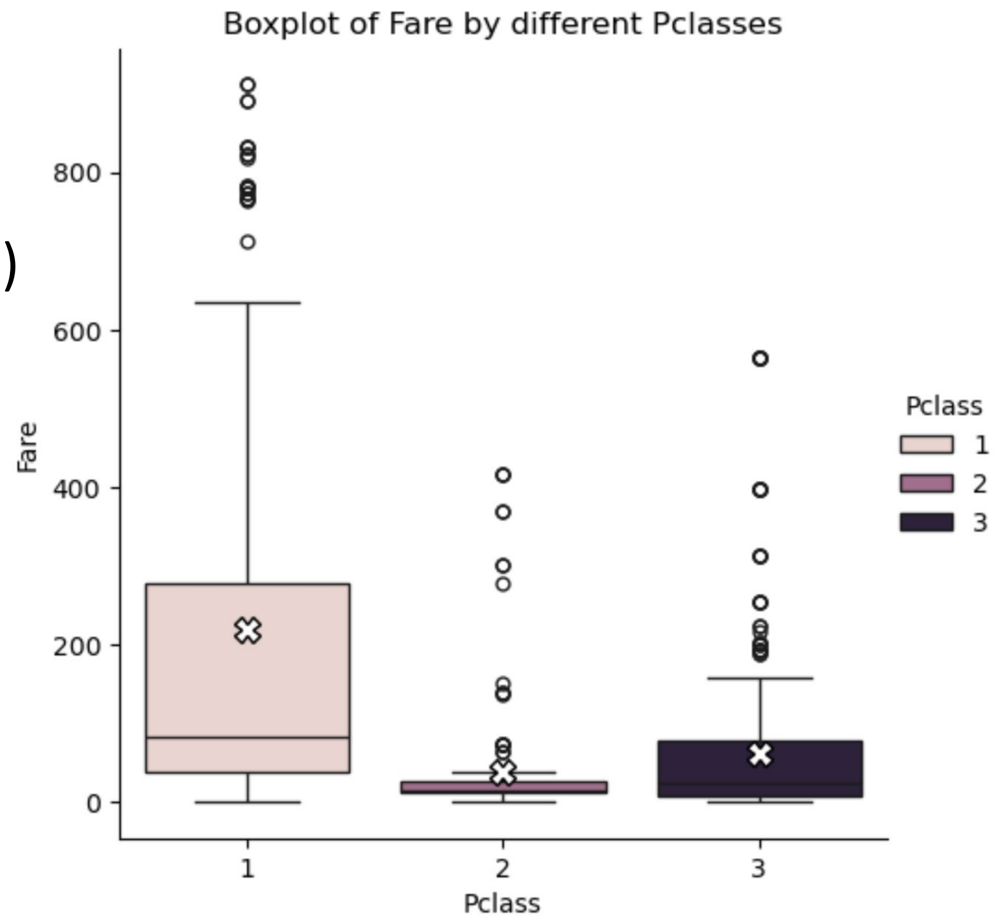
Exploratory Data Analysis (EDA)

- **Variation between two variables**
  - ***Two categorical variables***
    - Plotting: Mosaic plot – not always that usefull a plot
    - Descriptive statistics: A table – showing the number of cases in each combination of values of the categorical variables
  - ***A categorical variable and a numerical variable***
    - Plotting: boxplot - see next slide
    - Descriptive statistics: numeric descriptive statistics (mean, median, var, sd, …, etc.) for each group of the categorical values



Exploratory Data Analysis (EDA)

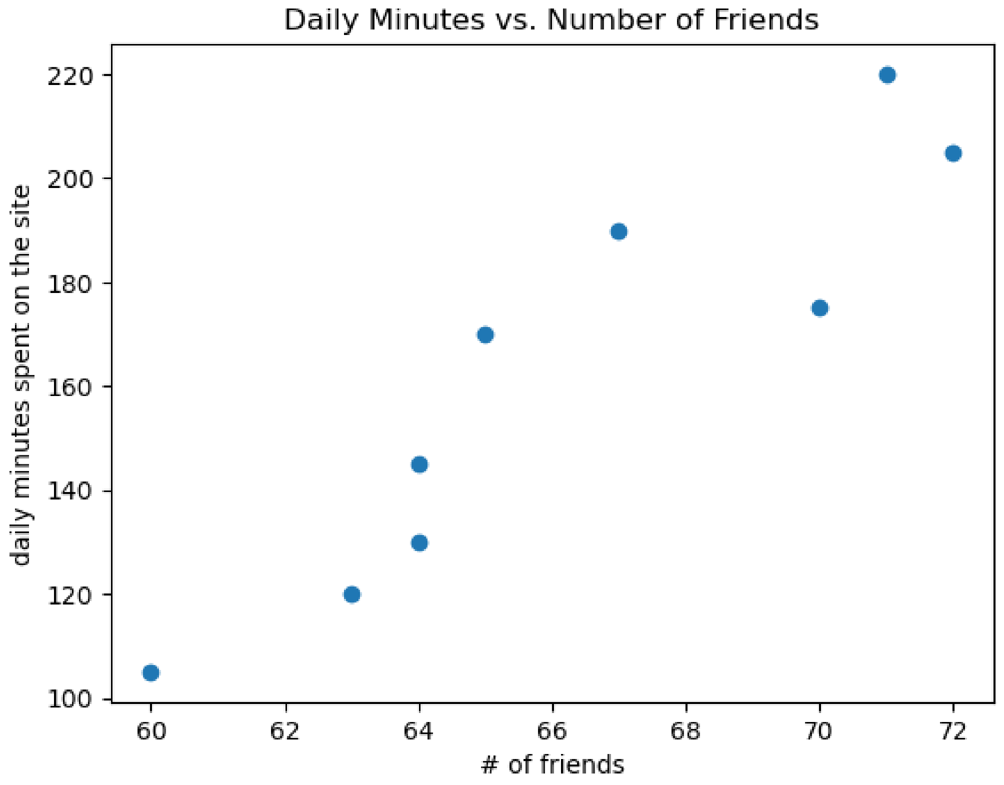
- **Boxplots**
  - The center line is the median
  - The lower edge of the box is the first quantile (Q1)
  - The upper edge of the box is the third quantile (Q3)
  - The Interquartile Range: IQR = Q3-Q1
  - Bottom whisker:
    - max(min(data values), Q1-1.5*IQR)
  - Top whisker:
    - min(max(data values), Q3+1.5*IQR)
  - Points below the bottom whisker or above the top whisker are referred to as outliers




```
sns.catplot(x="Pclass", y = "Fare", hue = "Pclass", data = titanic_data, kind="box",
            showmeans=True,
            meanprops={"marker":"X", "markerfacecolor":"white", "markeredgecolor":"black", "markersize": "10"})
plt.title("Boxplot of Fare by different Pclasses")
plt.show()
```

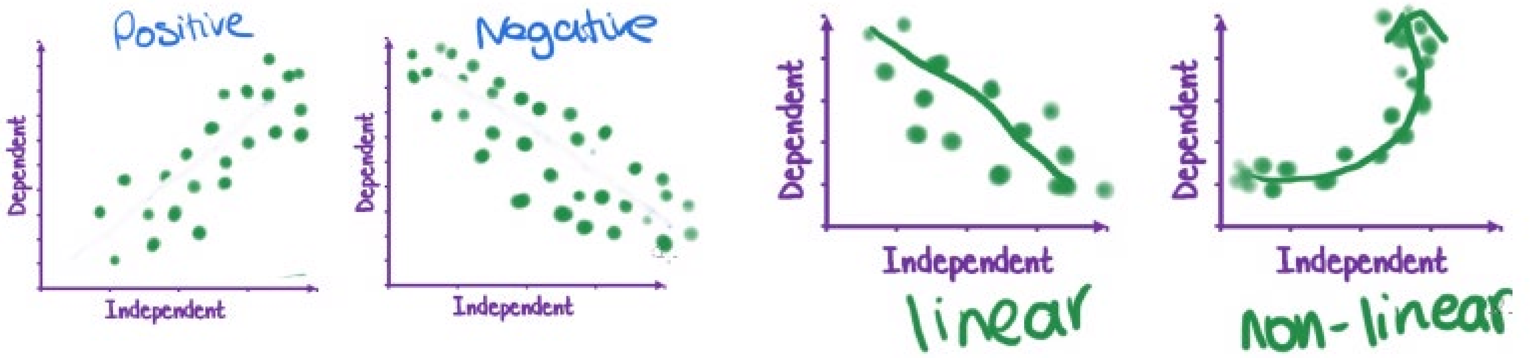
Exploratory Data Analysis (EDA)

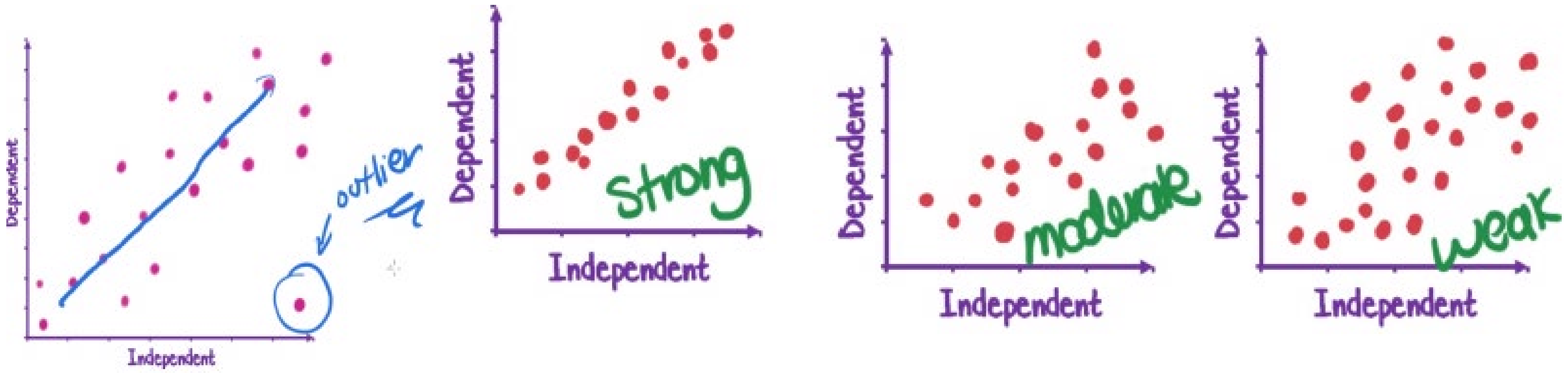
- **Variation between two variables**
  - ***Two numerical variables***
    - Plotting: Scatter plot – as we have already seen!
    - Descriptive statistics: ***Pearson's correlation coefficient***
      - The standard correlation coefficient to measure *linear* correlation
      - Returns a value between -1 and 1.
        - -1 is perfect negative correlation
        - 1 is perfect positive correlation
        - 0 is no correlation
      - In pandas we can use the *.corr* method on a Series and in SciPy there is the function *pearsonr*.



Exploratory Data Analysis (EDA)

- **Types of Correlation**
  - ***Direction***
    - Positive
    - negative
  - ***Shape***
    - Linear
    - non-linear
  - ***Strength***
    - Weak
    - Moderate
    - strong
  - ***Outliers***





- See: https://www.youtube.com/watch?v=PE_BpXTyKCE

Roskilde Universitet

Exploratory Data Analysis (EDA)

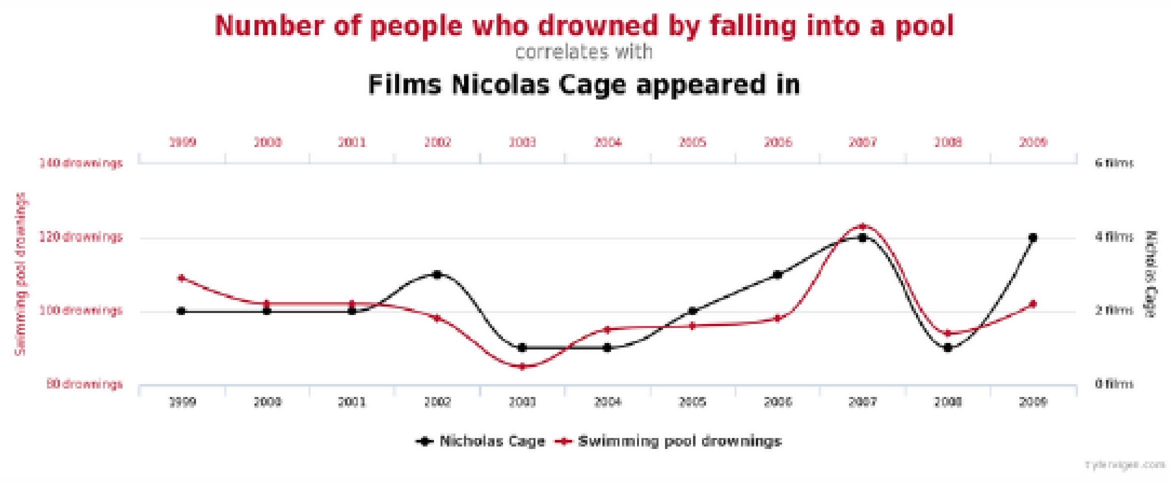
- **Correlation caveats**
  - ***Simpsons paradox*** *– correlation can change direction for sub populations*

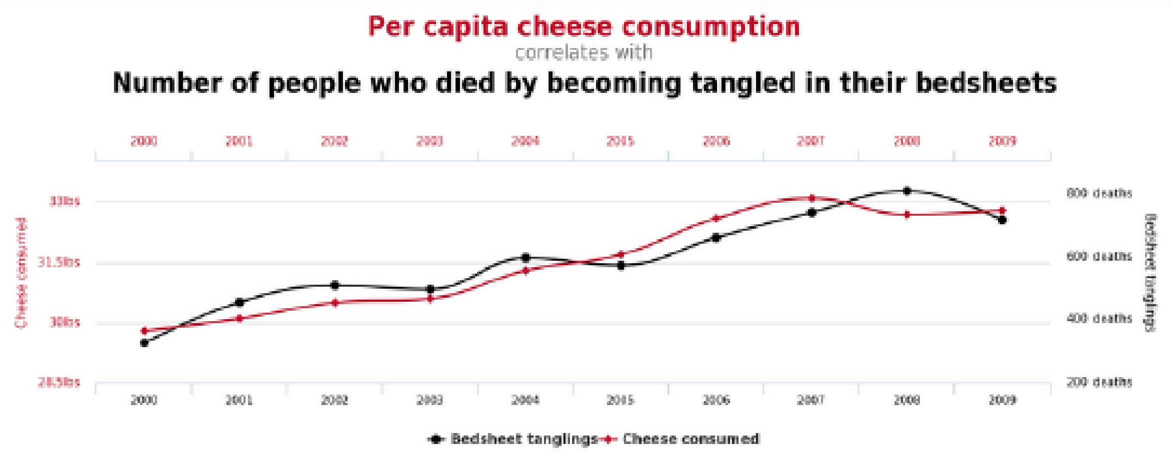
Coast	# of members	Avg. # of friends
West Coast	101	8.2
East Coast	103	6.5

Coast	Degree	# of members	Avg. # of friends
West Coast	PhD	35	3.1
East Coast	PhD	70	3.2
West Coast	No PhD	66	10.9
East Coast	No PhD	33	13.4

Exploratory Data Analysis (EDA)

- **Correlation caveats**
  - ***Correlation vs causation***
    - *Just because two variables correlates, it does not mean that there is a causal relationship between them*
    - *Spurious Correlations*
      - *(http://www.tylervigen.com/spurious-correlations)*
    - *There can be multiple explanations...*

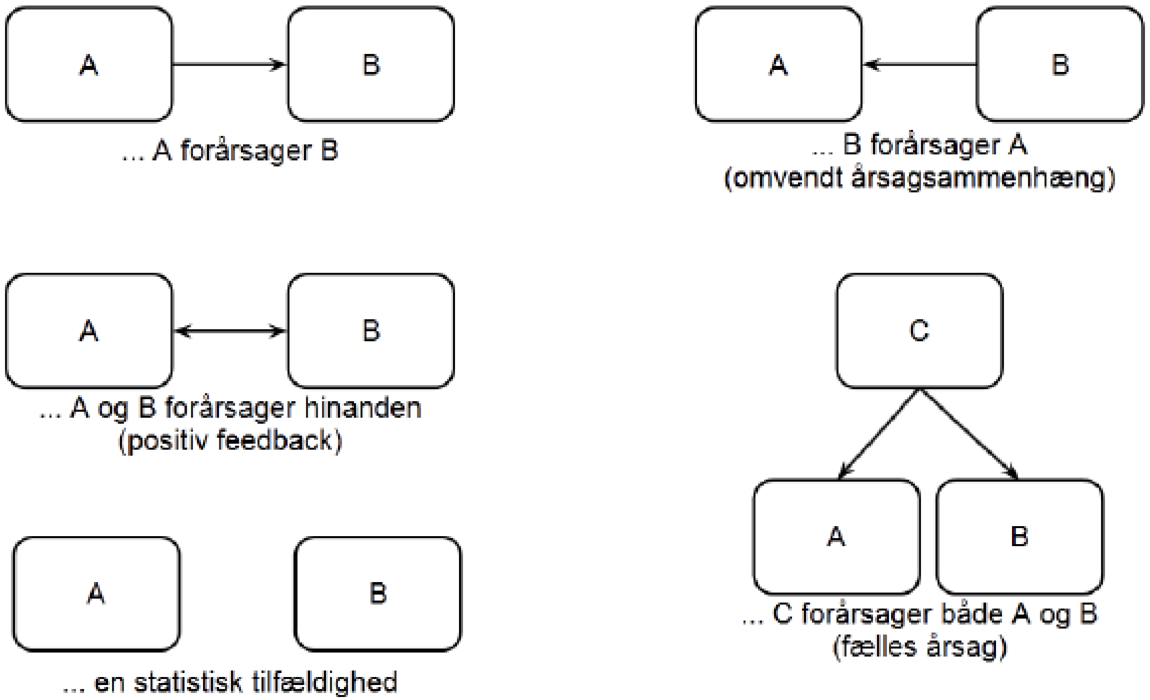




Exploratory Data Analysis (EDA)

- **Correlation caveats**
  - ***Correlation vs causation***
    - *There can be multiple explanations…*
      - *A cause B*
      - *B cause A*
      - *A and B cause each other*
      - *a statistical coincidence*
      - *C cause both A and B (a common cause)*

Hvis A korrelerer med B, så kan det være fordi…



Exploratory Data Analysis (EDA)

- Let us look at the notebook “Exploratory data analysis.ipynb”

Outline of this lecture

- Data types
- Data transformation and data cleaning
- Data visualization
- Exploratory Data Analysis (EDA)
- Exercises

Exercises

- Do the exercises in the notebook “Exercises in DT and EDA.ipynb”