Tag 4: Datenaufbereitung & Feature Engineering

Samuel Schlenker

06.11.2025, WWI 2025F

Students should ...

- Differentiate between objects, data, databases, and information using the knowledge hierarchy
- Understand key properties of quality datasets (volume, history, consistency, purity, level of detail, clarity, transparent origin)
- Distinguish between structured and unstructured data formats
- Apply dimensional data structures (dimensions vs. facts) for analysis
- Perform data operations: slicing, dicing, roll-up, and drill-down
- Understand the importance of data preparation and cleansing (representing 45-80% of data scientists' time)
- Identify and handle missing data using different strategies (MCAR, MAR, MNAR)
- Apply imputation techniques for missing values
- Detect and handle outliers using methods like IQR, Z-score, and Isolation Forest
- Apply normalization techniques (Min-Max, Z-Score standardization)
- Perform categorical encoding using one-hot encoding
- Conduct feature selection and extraction to improve model performance
- Understand the data preparation pipeline from raw data to analysis-ready datasets

Differentiation of data

- **Differentiation between Object**, **Data**, **Database** and **Information**
- Differentation by Prof. Klaus North
- Differentation by various properties

Differentiation of data

- **Object** (also data set): Object of consideration, has various interesting features (which are generic to its class).
- **Data**: Value of a single (characteristic) property of an object
- **Database**: is the group of objects in which we are interested.
- **Information**: Created by interpreting the data (e.g., assignment of meanings)

Object

Data

Database

Information

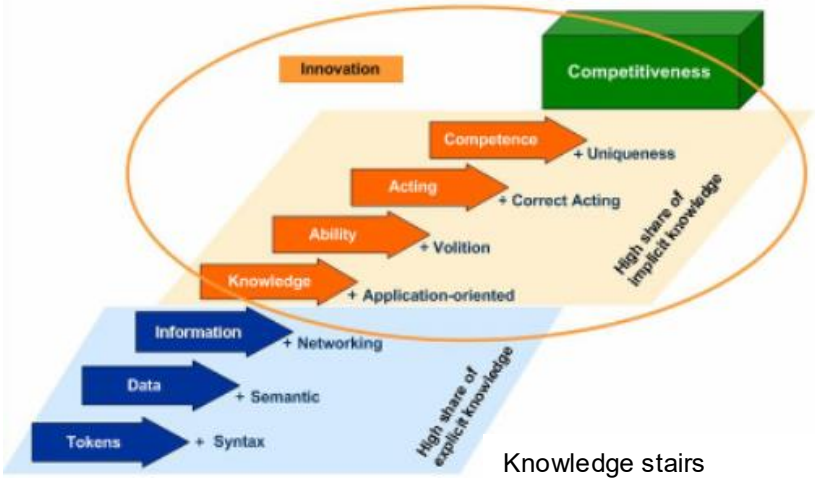
Differentiation of data

Data: Syntactically correct, symbolic representations consisting of individual data elements and values.
T = 16, P = 928, R = CEU
(Data type: Syntactic structure of data elements)

Information: Places data in a semantic context.
T = 16 °C, P = 928 mbar, R = CEU (Central Europe)

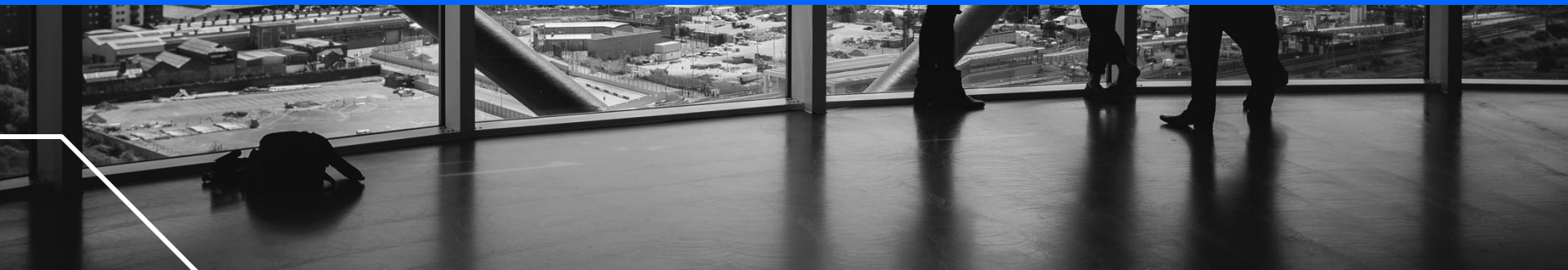
Knowledge: Systematic linking of information
T = 16 C°, P = 928 mbar, R = CEU (Central Europe), probable rain

Skills and actions: (Automated) application of knowledge
Mountain guide, farmer
Weather report, weather app => Models and algorithms



Knowledge stairs
according to Prof. Klaus North

Properties of Data and Datasets



Properties of Data and Datasets

Volume
Amount of data
A large amount of relevant, available data means there's a better chance you have what you need to answer your questions.

Note: There is no need to collect data simply for its own sake. Relevance is important!

Properties of Data and Datasets

History / Temporal Development

Data obtained at different points in time allows us to see how the current situation developed (based on patterns).

Example: the sales trends of the last 10 years make it possible to detect increases or decreases.

Properties of Data and Datasets

Consistency
If the facts change, the data should

- reflect this (the data must be adjusted)
- do not create inconsistencies
- continue to be compliant with the data model

Example: Inflation and price adjusted salary and price data.

Properties of Data and Datasets

Purity (cleanliness)
For data to be meaningful, it should not be inaccurate or incomplete and should not contain errors.

Note:

- Data that is inaccurate or misleading (high degree of impurity) is also said to be corrupt.
- Often related to consistency.

Properties of Data and Datasets

Level of **Detail**
The more detailed the data, the better we can examine it at different levels.
Operations:

- **Aggregation** (Roll-Up)
- **Disaggregation** (Drill-Down)

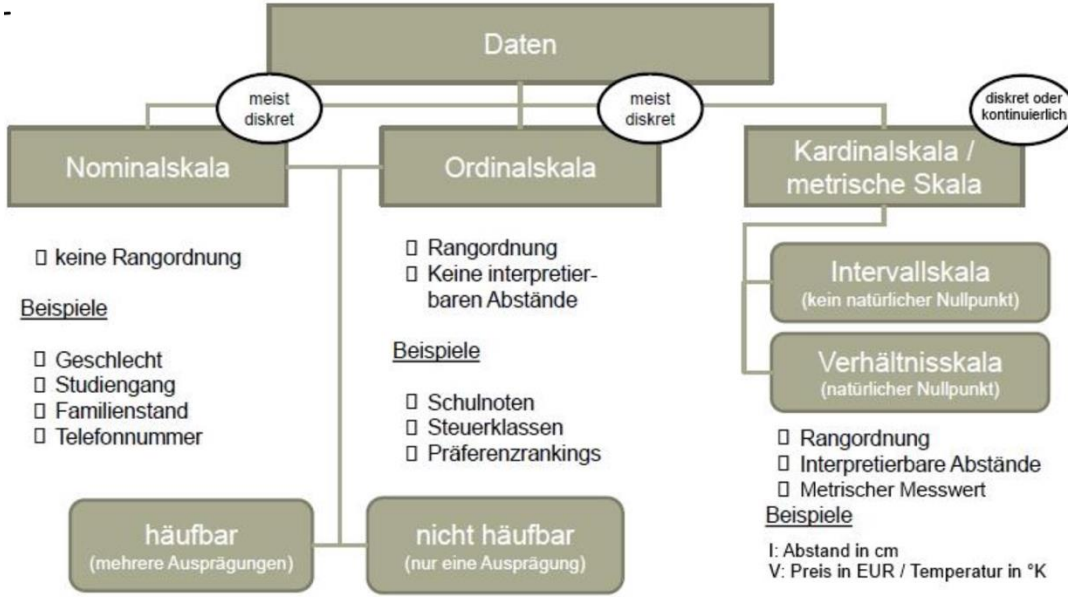
Example: We want to understand cycling trends in Baden-Württemberg. Then it would be helpful t

Properties of Data and Datasets

Variables (also: field, column)
Data usually includes **quantitative** (numerically measurable) as well as **qualitative** (characteristic, non-numerically measurable) variables.

Note: More variables usually let us discover more!

Properties of Data and Datasets



Properties of Data and Datasets

Segments

Grouping variables based on similar characteristics can be integrated into the data for easier analysis.

Example: Data about movies can be grouped by genre (e.g., action, science fiction, romance, comedy).

Properties of Data and Datasets

Clarity (comprehensibility)
Data should be described in terms that are easy to understand (not in code/encodings)

Example: Description of apartment types such as "single-family house", "two-family conversion" and "terraced house" are much easier to understand than "1Fam", "2fmCon" and "TwnhsE".

Properties of Data and Datasets

Transparent Origin

To trust the data, we need to know if it comes from a reliable source and if it has been treated in a trustworthy way.

Properties of Data and Datasets

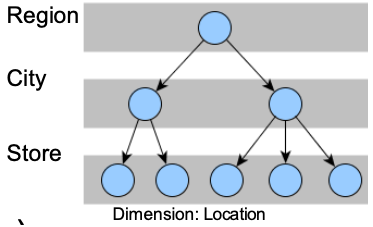
Dimensional Structure

Structuring of data into two types:

- dimensions (usually qualitative values) and
- facts or measures (quantitative values)

Note: data can be examined in terms of dimensions. In many cases, individual (operational) data sets/values are not of interest, but aggregated values/quantities provide statements about data in its entirety.

Dimensions



Are **descriptive** in nature

- are scaled ordinal or nominal
- can be hierarchically structured and
- index facts

A dimension forms a view of the facts (side of a data cube)
The hierarchy set H represents the set of hierarchical attributes of a dimension

- granularity and levels of compaction
- irreflexive half-order $(H, \prec)$

Examples: Product, Time or Country, ...

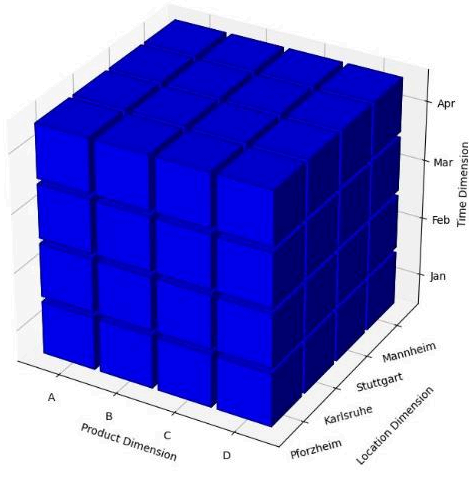
Facts

Represent key performance indicators

- also called numerical or summary attributes
- can be calculated by aggregate functions and formulas
- for example: min, max, count, sum, avg

Facts always have a **quantitative** (numeric) data type
Examples of facts:

- sum(turnover)
- count(employee-number)
- min(contribution-margin)

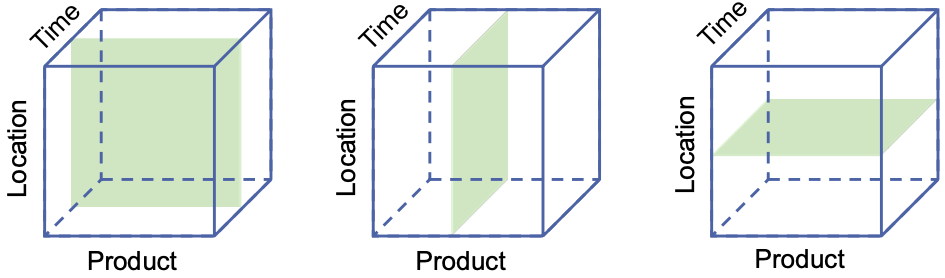


Slicing

Slicing or selection $\sigma T\,(cond)$

- selects tuples from the cube
- according to selection conditions ($cond$)

Conditioning (Statistics)

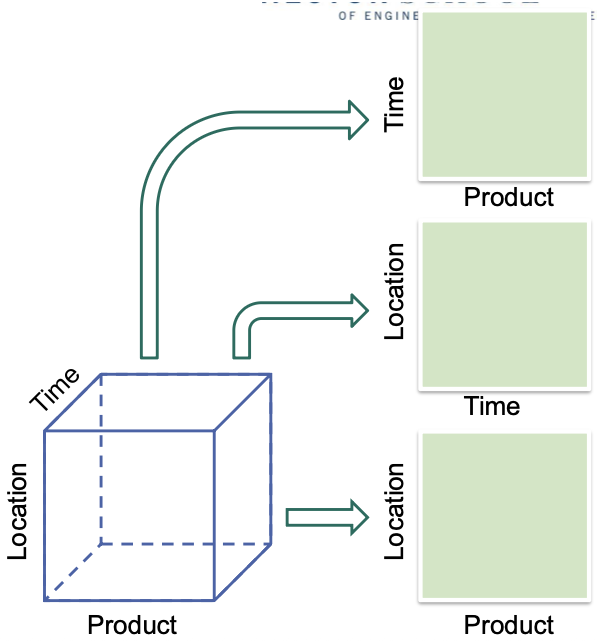


Dicing

Dicing or projection $\pi T\,(D'\ )$

- sets dimensions for projection
- creates a sub cube

Marginalizing (multivariate statistics)



Database (relational model)

- Data is structured in rows and columns
- Each column represents a different variable (characteristic, field).
  - a variable is the measurement of a property that can vary or change. each variable is (in) a column with a column header.
  - any other observation of a variable (value) is in a different row.
- Each row represents a record.

elevation	latitude	longitude	population	country	province	name
247	9	49	585.890	D	Baden-Württemberg	Stuttgart
97	8	49	290.117	D	Baden-Württemberg	Mannheim
115	8	49	289.173	D	Baden-Württemberg	Karlsruhe
278	8	48	209.628	D	Baden-Württemberg	Freiburg
114	9	49	146.751	D	Baden-Württemberg	Heidelberg
261	9	49	114.411	D	Baden-Württemberg	Pforzheim
478	10	48	116.761	D	Baden-Württemberg	Ulm

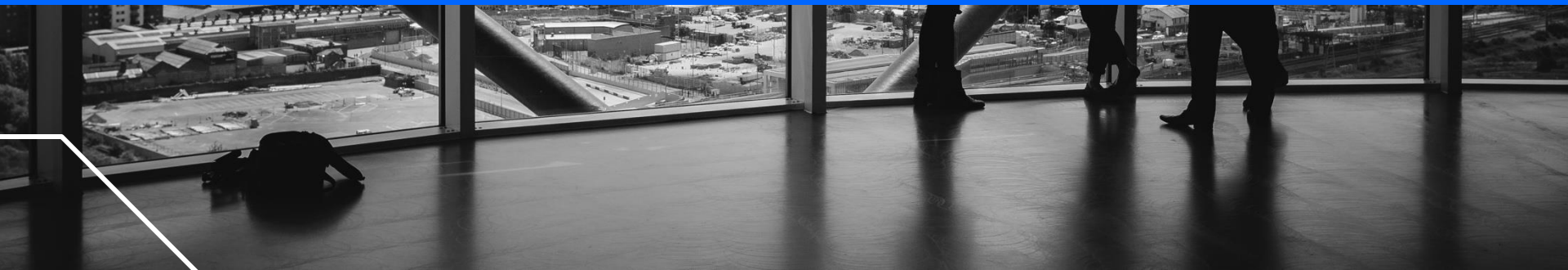
Database

- Data can also be poorly structured.
- Variables (fields) are not in a column with a column heading at a time. Not every observation is on a different line.
- Labels (headings) are inserted several times as rows above the column headings or as
- additional columns.

- What should you do if your data is not well structured?
- change the underlying database
- use of a programming language (e.g., Python) for modification
- use ETL (Extract, Transform, Load) tools

	Structured Data	Unstructured Data
Data Definition	• Has clearly defined data types	• Undefined and stored in its native format
	• Stored in rows and columns, can be mapped to fields	• No predefined model
Data Analysis	• Easy to search and process by humans and algorithms	• Difficult to search and process
Data Nature	• Quantitative in nature	• Qualitative in nature
	• Processing methods include clustering, regression, relationships, and classification	• Not processed and analyzed using conventional tools
		• Processing methods used include data mining and stacking
Data Storage	• Stored in data warehouses in a relational database	• Stored in data lakes in non-relational (NoSQL) databases
	• Require little storage space	• Requires more storage space
Data Format	• Format: numbers and text	• Wide variety of data sizes and shapes, from imagery to email, audio, video, etc.
	• The data format is defined beforehand	• It has no data model and requires no transformation

Data preparation & cleansing

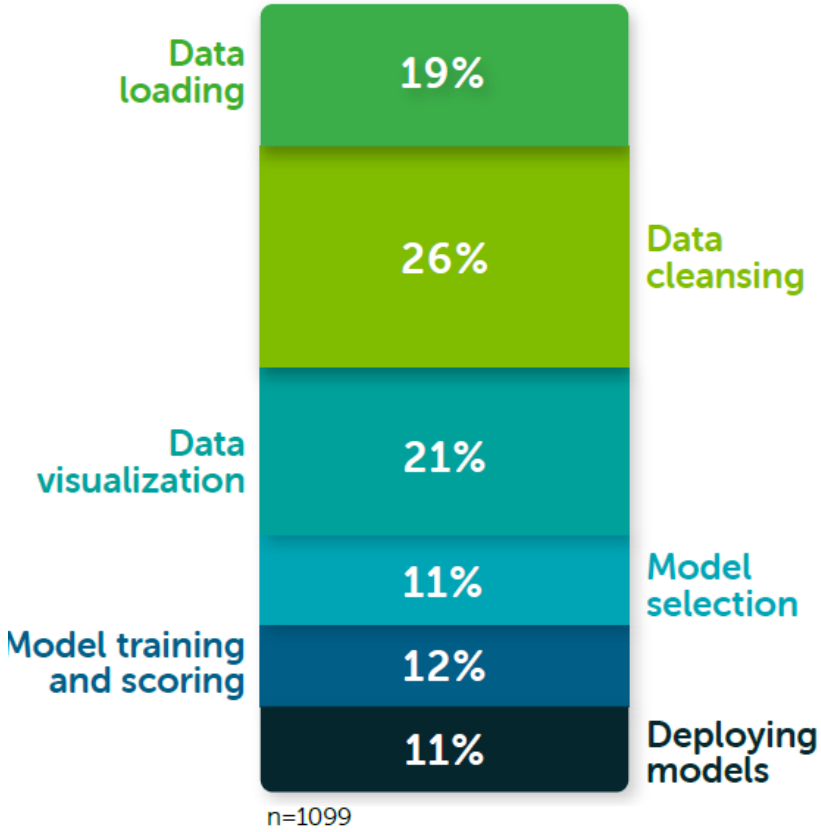


Introduction to Data Preparation and Cleansing

- **Data Preparation** and **Cleansing** is a fundamental step in the machine learning workflow that involves getting the data ready for analysis and model building.

- This step is crucial because the quality and format of your data can determine the performance of your machine learning algorithms.

45% of the time for preparation



Anaconda, 2020, State of Data Science: Moving From Hype Toward Maturity


The data may not contain the answer. The combination of some data and an aching desire for an answer does not ensure that a reasonable answer can be extracted from a given body of data. – John Turkey (1986)

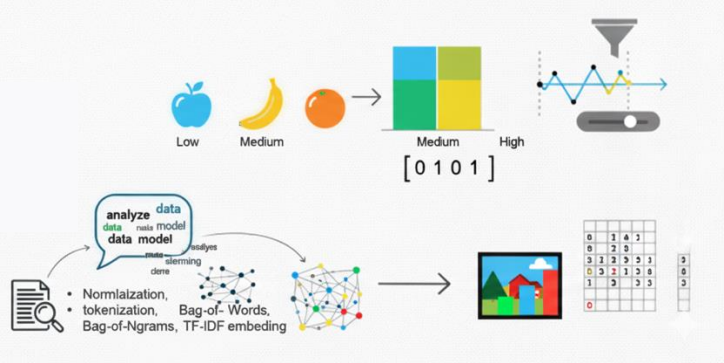
Importance of Data Preparation and Cleansing

- **Accuracy**: Clean data is representative of the problem at hand and leads to more accurate analysis and predictions.
- **Efficiency**: Algorithms work more efficiently with data in a consistent format.
- **Reliability**: Decision-making based on analysis of clean data is more reliable.

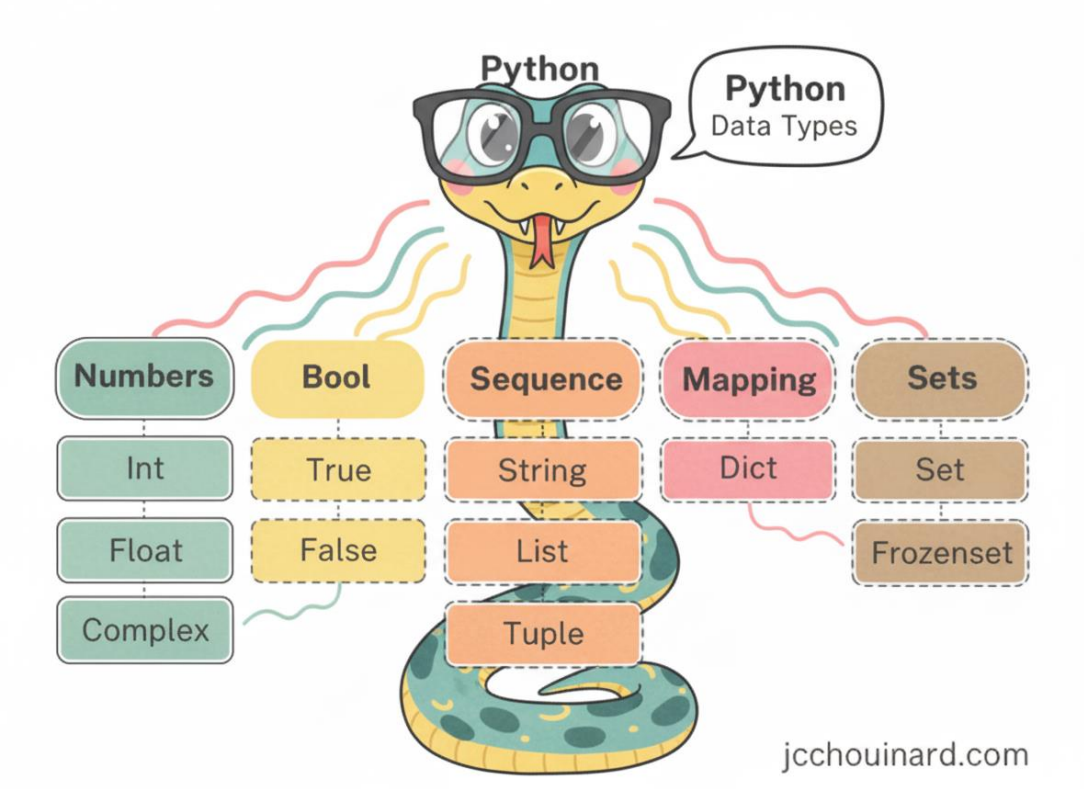
Data Preprocessing

Transform extracted attributes first. For example:

- Number to normalized number, e.g., min-max normalization
- Number to category, e.g., binning (quantile, equidistant) Category to numeric vector, e.g., one-hot encoding Text to numeric vector, e.g.:
  - Normalization, tokenization, stemming Bag-of-Words, Bag-of-Ngrams, TF-IDF Latent word embedding
  - Image to numeric value, e.g., color histogram



Data Type Conversion



Data Preparation

The process of converting raw data into a clean dataset.

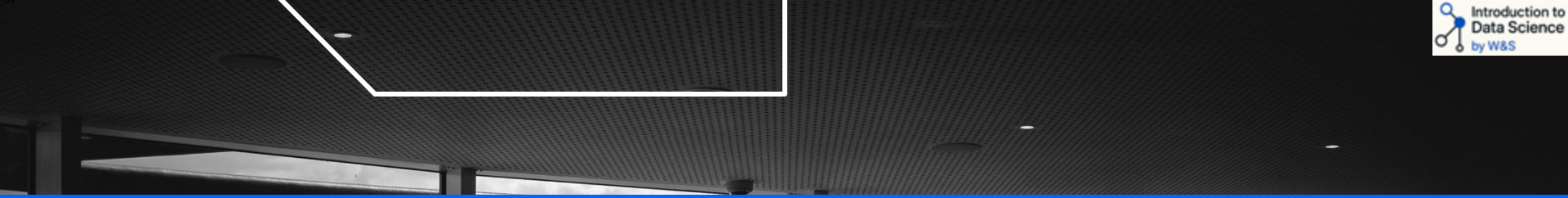
Data preparation might include:

1. **Data Collection**: Gathering data from multiple sources, which could be databases, files, sensors, or online repositories.
2. **Data Integration**: Combining data from different sources, which may require resolving data conflicts and inconsistencies.
3. **Data Transformation**: Converting data into a suitable format or structure for analysis. This might involve normalizing data (scaling data within a range, typically 0 to 1, or -1 to 1), encoding categorical variables (changing text-based categories into numerical values), or creating new variables through feature engineering.

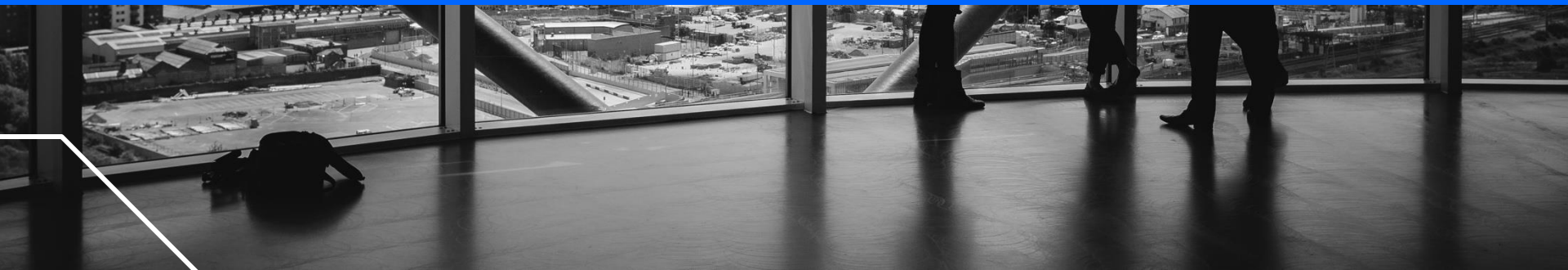
Data Cleansing

Once the data is in a preliminary structured format, data cleansing takes place, which involves:

1. **Handling Missing Data**: Data can have missing values due to various reasons, and dealing with them is essential. Techniques include imputing the missing values using statistical methods (like mean, median, or mode) or predictive modeling, or discarding the rows or columns with missing data altogether.
2. **Identifying and Correcting Errors**: This includes spotting and rectifying mistakes or inconsistencies in the data, such as typos, incorrect entries, or mislabeled classes.
3. **Removing Duplicates**: Duplicate data can bias the analysis, so it's important to identify and remove any duplicates from the dataset.
4. **Detecting and Filtering Outliers**: Outliers are data points that deviate significantly from the rest of the dataset. They can be due to variability in the data or experimental errors, and they can affect the results of the analysis.



Missing values

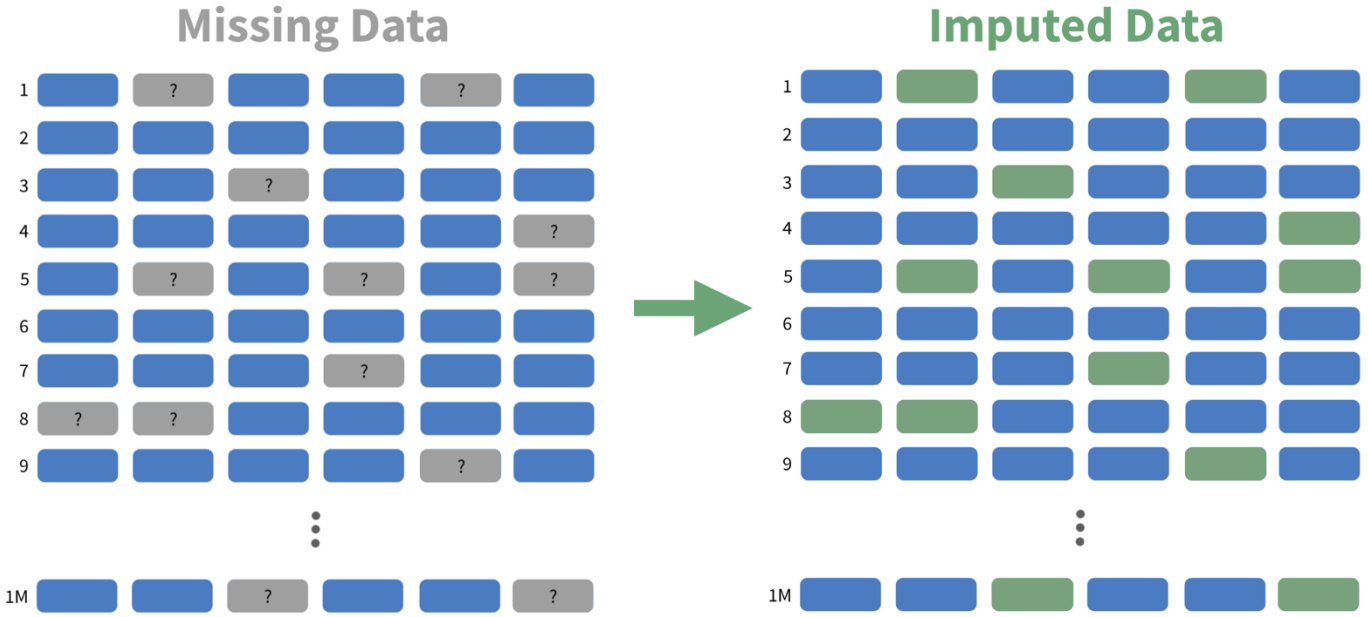


Understanding Missing Values

What are Missing Values? Missing values (data) occur(s) when no data value is stored for a variable in an observation.

Implications of Missing Values: The absence of data can lead to biased estimates, loss of efficiency, and complications in analyzing and interpreting results.

Understanding Missing Values



Types of Missing Data

- **Missing Completely at Random (MCAR)**
  - Definition: The probability of a data point being missing is the same for all cases.
  - Implication: Analyses can be unbiased, but power is reduced due to a smaller sample size.

- **Missing at Random (MAR)**
  - Definition: The likelihood of a data point being missing is related to the observed data but not the missing data.
  - Implication: Statistical techniques can be employed to handle MAR data, provided the model is correctly specified.

- **Missing Not at Random (MNAR)**
  - Definition: The probability of a data point being missing is related to the unobserved data, i.e., the missingness is related to the missing value itself.
  - Implication: MNAR can lead to biased analyses if not properly addressed.

Identifying Missing Data Patterns

Exploratory Analysis

- Analyzing patterns of missingness.
- Investigating the relationship between missing and observed values.

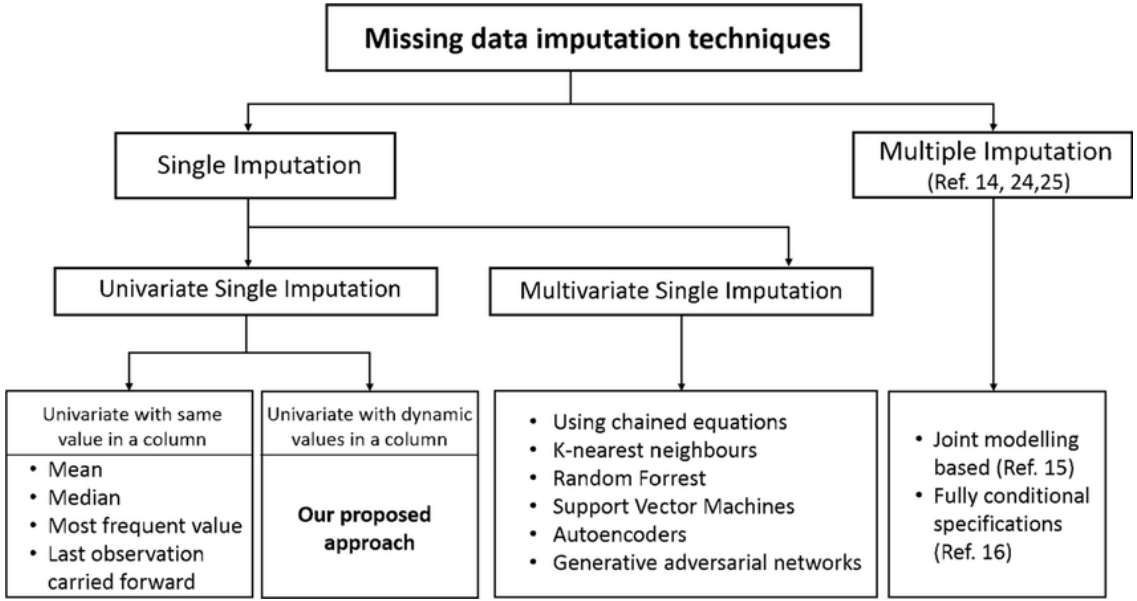
Sensitivity Analysis

- Assessing how different assumptions about the missing data affect the results.

Challenges in determining missing data

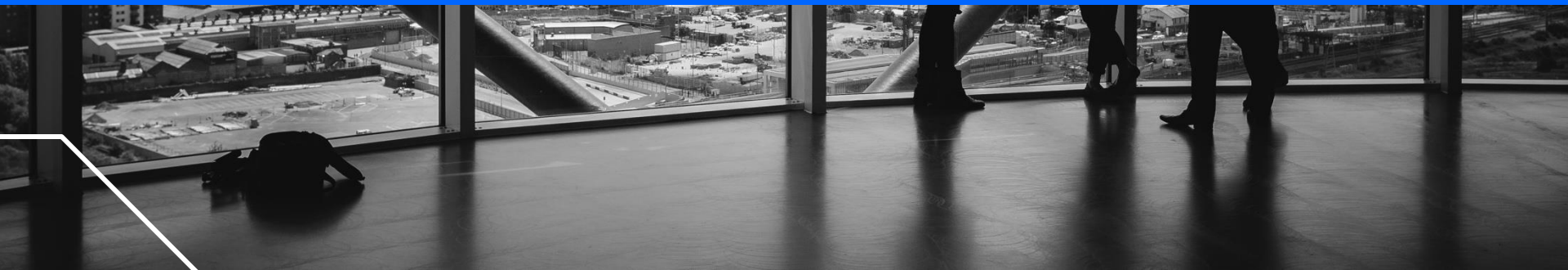
- Often, the true mechanism of missingness is unknown and must be inferred.
- The analysis is contingent on assumptions which should be tested for robustness.

Imputing Missing Data

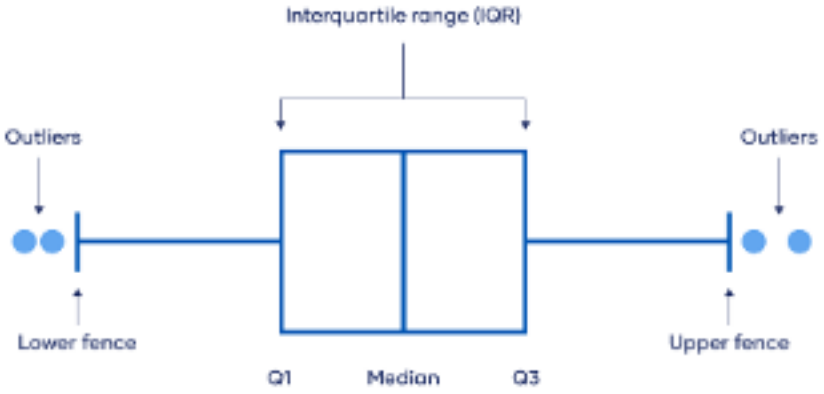


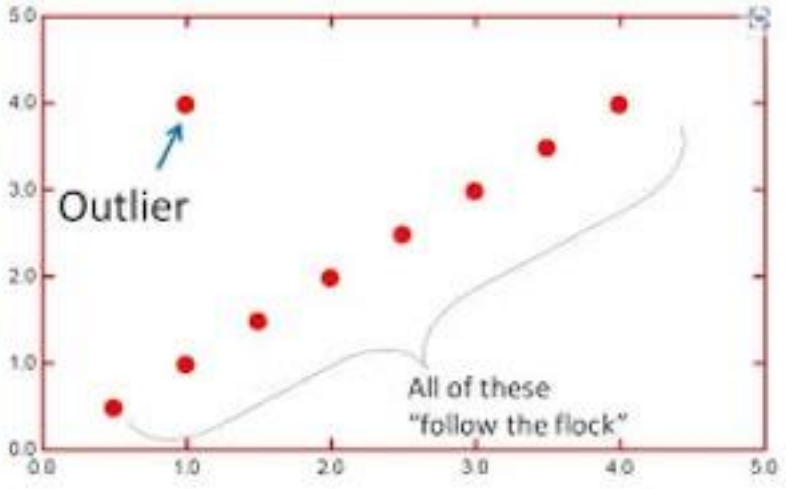
https://www.researchgate.net/figure/A-broad-categorization-of-missing-data-imputation-techniques_fig1_358529185

Data Cleansing



Understanding outliers

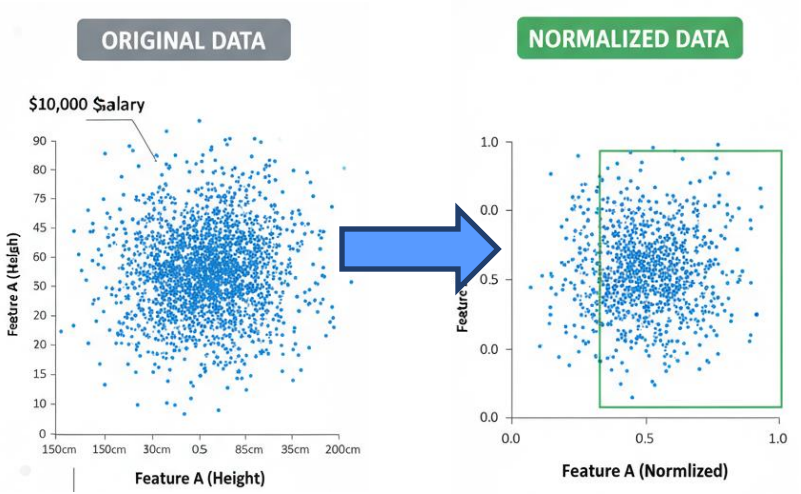


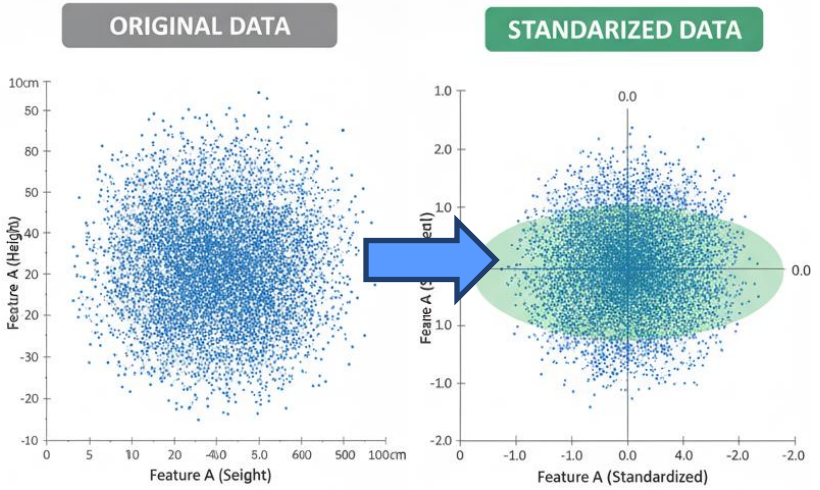


Outlier Detection Methods:
IQR, Z-score, Isolation Forest with formulas/examples

Outlier Handling:
Removal, capping, transformation strategies

Further pre-processing





Normalization scales numerical data to a fixed range, typically between 0 and 1. This is useful when features have different ranges and you want to prevent features with larger values from dominating the learning process.

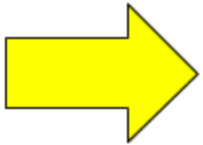
Standardization, on the other hand, transforms data to have a mean of 0 and a standard deviation of 1. This is particularly useful when your data has a Gaussian distribution or when algorithms assume normally distributed data.

Normalization Formulas

Purpose: Scale data to a standard range for better comparison and model performance		
	Min-Max Normalization	**Z-Score (Standardization)**
Formula	(x - min) / (max - min)	(x - μ) / σ μ = mean, σ = standard deviation
Result	Scales data to range [0, 1]	Centers data around 0 with unit variance
Example	Dataset: [10, 20, 30, 40, 50] Normalize x = 30: (30 - 10) / (50 - 10) = 20/40 = **0.5** Results: [0, 0.25, 0.5, 0.75, 1.0]	Dataset: [10, 20, 30, 40, 50] Mean (μ) = 30, Std Dev (σ) = 14.14 Normalize x = 30: (30 - 30) / 14.14 = **0** Results: [-1.41, -0.71, 0, 0.71, 1.41]
When to use	When you need bounded range; sensitive to outliers	When data is normally distributed; handles outliers better

One hot encoding

Color
Red
Red
Yellow
Green
Yellow



Red	Yellow	Green
1	0	0
1	0	0
0	1	0
0	0	1

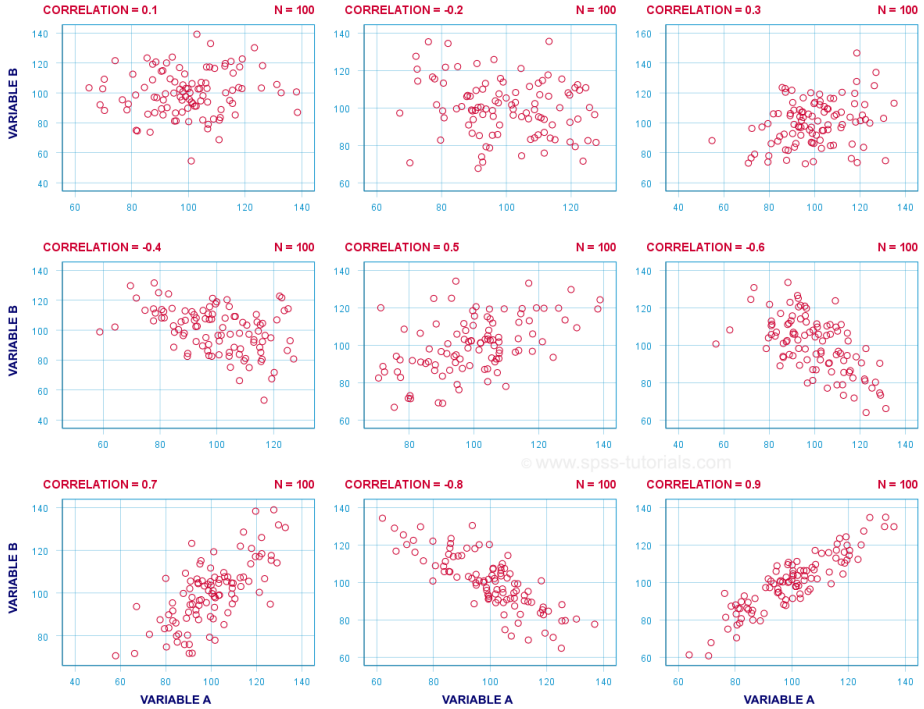
https://www.kaggle.com/code/dansbecker/using-categorical-data-with-one-hot-encoding

Featuring

Features = dimensions that describe each data point.

- Large number of features may have negative impact on performance of Machine Learning approaches.
- In the featuring step, the **number of features can be reduced** by deleting non-relevant data.
  - Challenge: How to identify non-relevant data in each use case?
- **Feature extraction** can be performed in this step which consists on deriving new dimensions to describe the data (=> getting more/ new features).

Featuring



SPSS Correlation Analysis - The Ultimate Guide