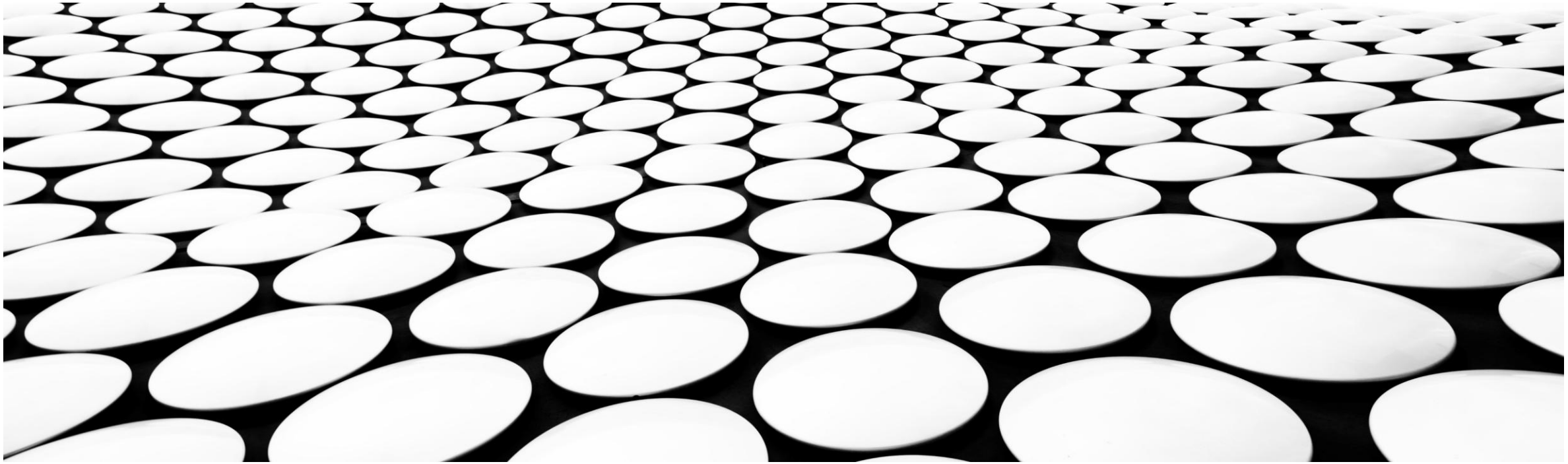
DATA CLEANING

Toward better quality data



DATA CLEANING

- Data wrangling
- Data quality
- Data cleaning
  - Duplicates
  - Invalid data
  - Outliers

Part 1

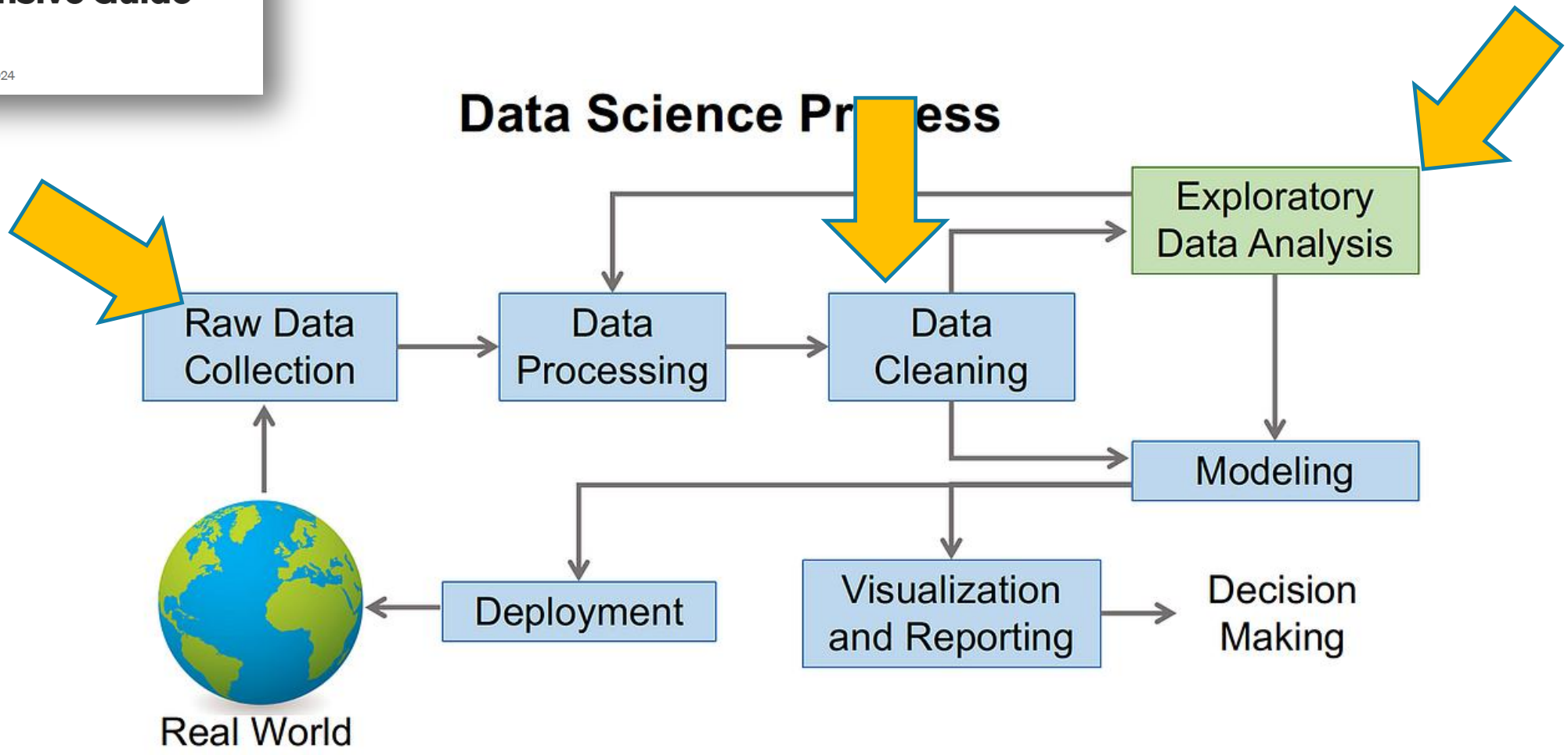
Part 2

I'll present imputation methods for missing data later (week 6)

DATA WRANGLING

Data Science Process: A Comprehensive Guide

Abhijit · Follow
6 min read · Jan 15, 2024

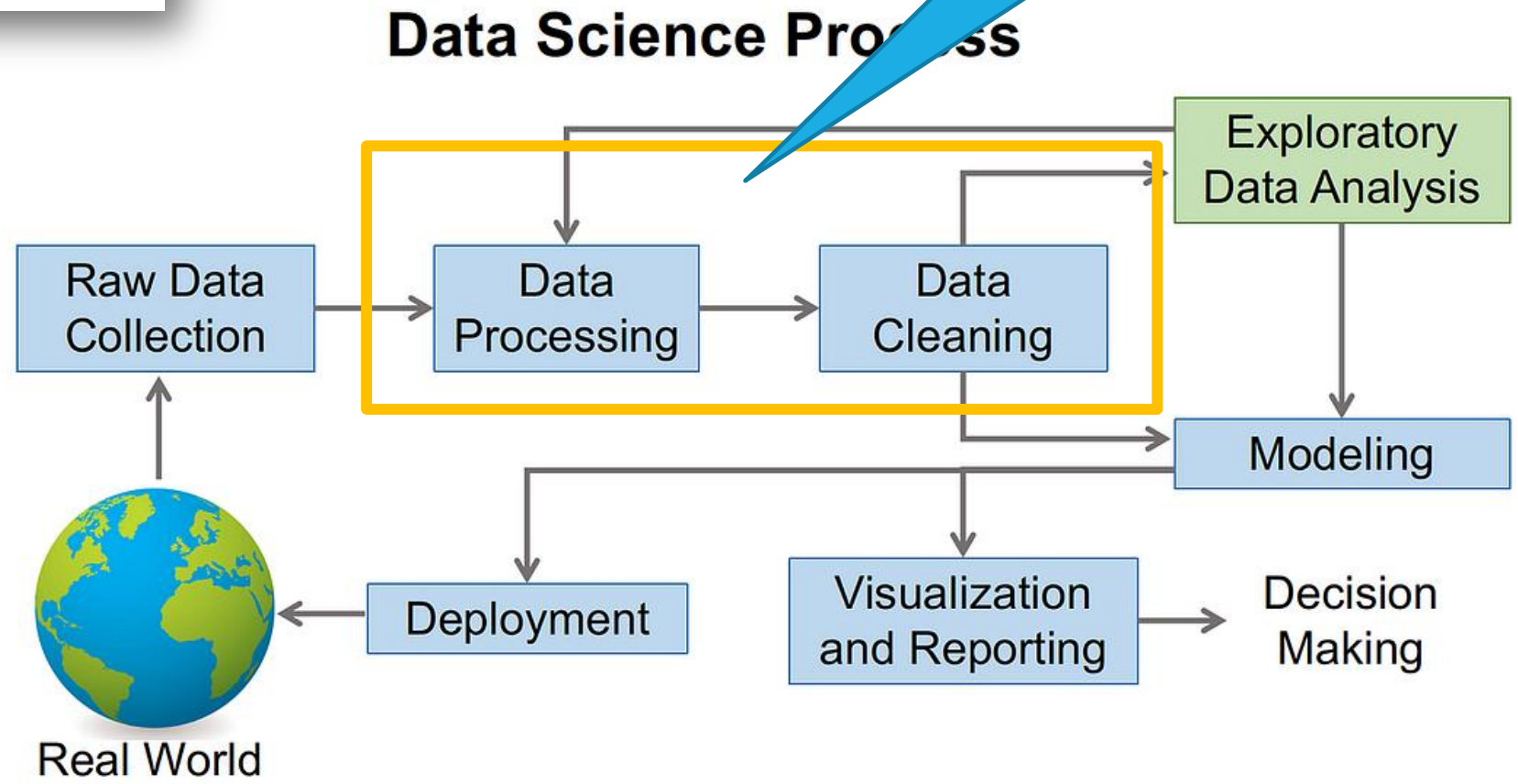


Source

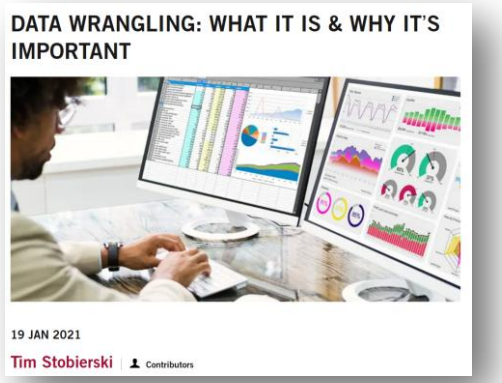
Data Science Process: A Comprehensive Guide

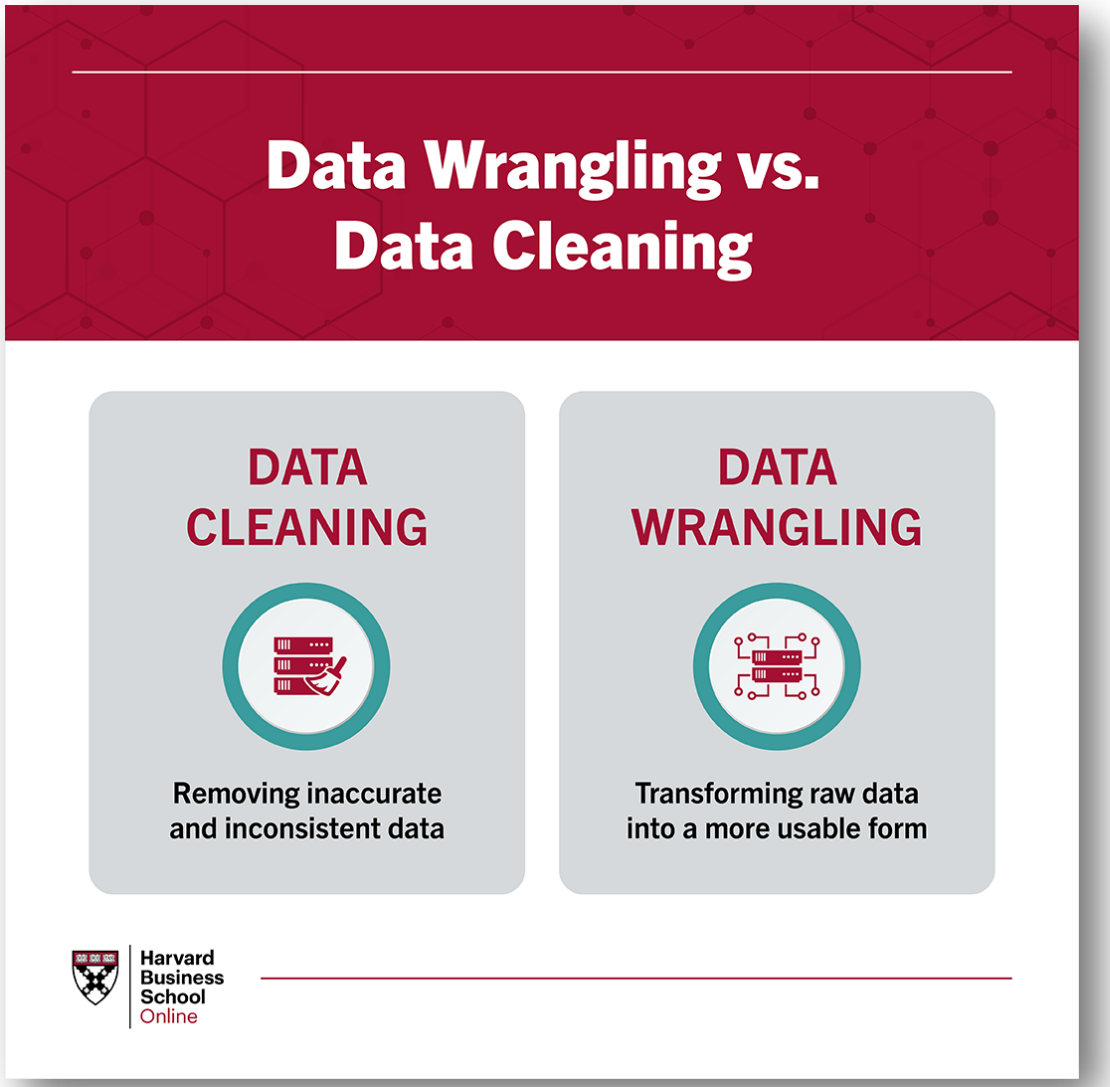
Abhijit · Follow
6 min read · Jan 15, 2024

You might see the term DATA WRANGLING



Source

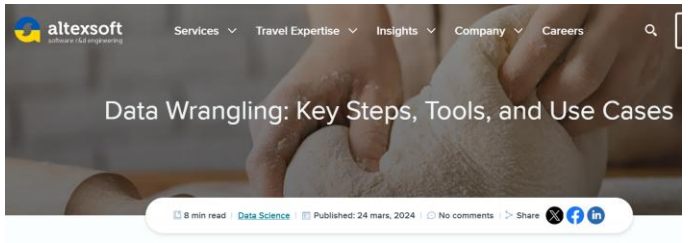




Data Wrangling vs. Data Cleaning

Despite the terms being used interchangeably, data wrangling and data cleaning are two different processes.

It's important to make the distinction that data cleaning is a critical step in the data wrangling process to remove inaccurate and inconsistent data. Meanwhile, data-wrangling is the overall process of transforming raw data into a more usable form.


altexsoft
Services
Travel Expertise
Insights
Company
Careers
Data Wrangling: Key Steps, Tools, and Use Cases
8 min read
Data Science
Published: 24 mars, 2024
No comments
Share


altexsoft
DATA WRANGLING STEPS
1
Discovering
2
Structuring
3
Cleaning
4
Enriching
5
Verifying
6
Publishing

Data collection

- Sensor Data
- Data obtained through API
- Data extracted from Social Media

Transform into tabular data

Convert unstructured data to structured data

API call

```
https://api.openweathermap.org/data/3.0/onecall?lat=
{lat}&lon={lon}&exclude={part}&appid={API key}
```

OpenWeather

https://openweathermap.org/api/one-call-3

Example of API response


```
{
   "lat":33.44,
   "lon":-94.04,
   "timezone":"America/Chicago",
   "timezone_offset":-18000,
   "current":{
      "dt":1684929490,
      "sunrise":1684926645,
      "sunset":1684977332,
      "temp":292.55,
      "feels_like":292.87,
      "pressure":1014,
      "humidity":89,
      "dew_point":290.69,
      "uvi":0.16,
      "clouds":53,
      "visibility":10000,
      "wind_speed":3.13,
      "wind_deg":93,
      "wind_gust":6.71,
      "weather":[
         {
            "id":803,
            "main":"Clouds",
            "description":"broken clouds",
            "icon":"04d"
         }
      ]
   },
```

This is complex enough that we will hopefully have time to come back to this topic,,,

Dataset 1

Dataset 2

Gather and merge

Case Study
Energy Consumption

Sensors

APIs

Going back to example from last week

Sensor Types Included

1. Smart electricity meter
2. Smart thermostat
3. Indoor temperature sensor
4. Occupancy / motion sensor

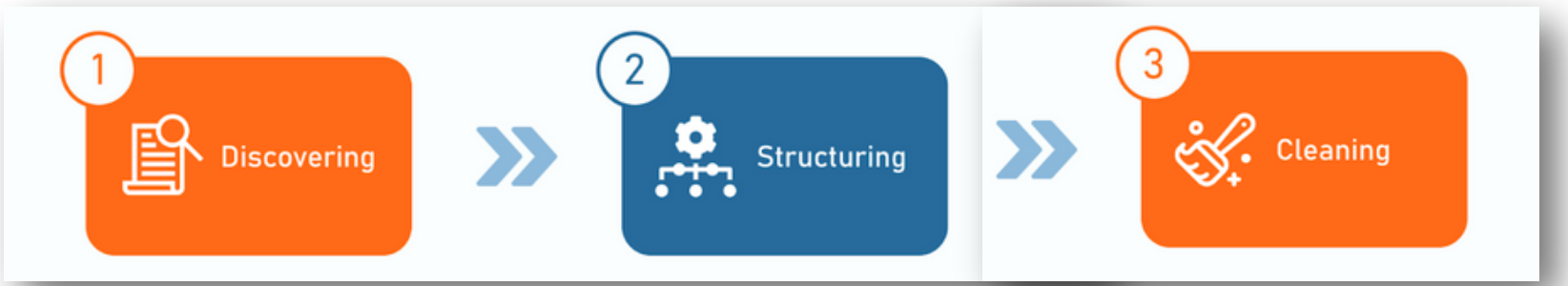
Merging on TimeStamp, but with non-corresponding times. Time series analysis (end of semester)

Electricity Smart Meter (Whole-Home)

timestamp	meter_id	power_kW
2025-02-12 06:00:00	MTR_88421	0.8
2025-02-12 06:15:00	MTR_88421	1.1
2025-02-12 06:30:00	MTR_88421	1.6
2025-02-12 06:45:00	MTR_88421	2.3
2025-02-12 07:00:00	MTR_88421	3.0

Smart Thermost

timestamp	thermostat_id	setpoint_C	indoor_temp_C	hvac_mode
2025-02-12 05:50:12	THM_33210	21.0	19.4	heating
2025-02-12 06:10:08	THM_33210	21.0	19.9	heating
2025-02-12 06:28:44	THM_33210	22.0	20.3	heating
2025-02-12 06:47:01	THM_33210	22.0	20.9	heating
2025-02-12 07:15:33	THM_33210	20.0	21.2	off



Data cleaning includes data profiling, which is the process of reviewing the content and quality of a dataset to detect potential issues or anomalies.

But… quality is more than that.

DATA QUALITY

Collibra

The 6 data quality dimensions with examples

Data quality goes beyond what cleaning can do...

1. Completeness	2. Accuracy	3. Consistency
How complete is your dataset?	Is the data an accurate representation of the element it describes?	Is your data synchronized across your organization?

4. Validity	5. Uniqueness	6. Integrity
Does the data fit the defined range and definition?	Are there any duplicates within your data?	Can your data be traced and connected across the organization?

Source

1. Completeness

How complete is your dataset?

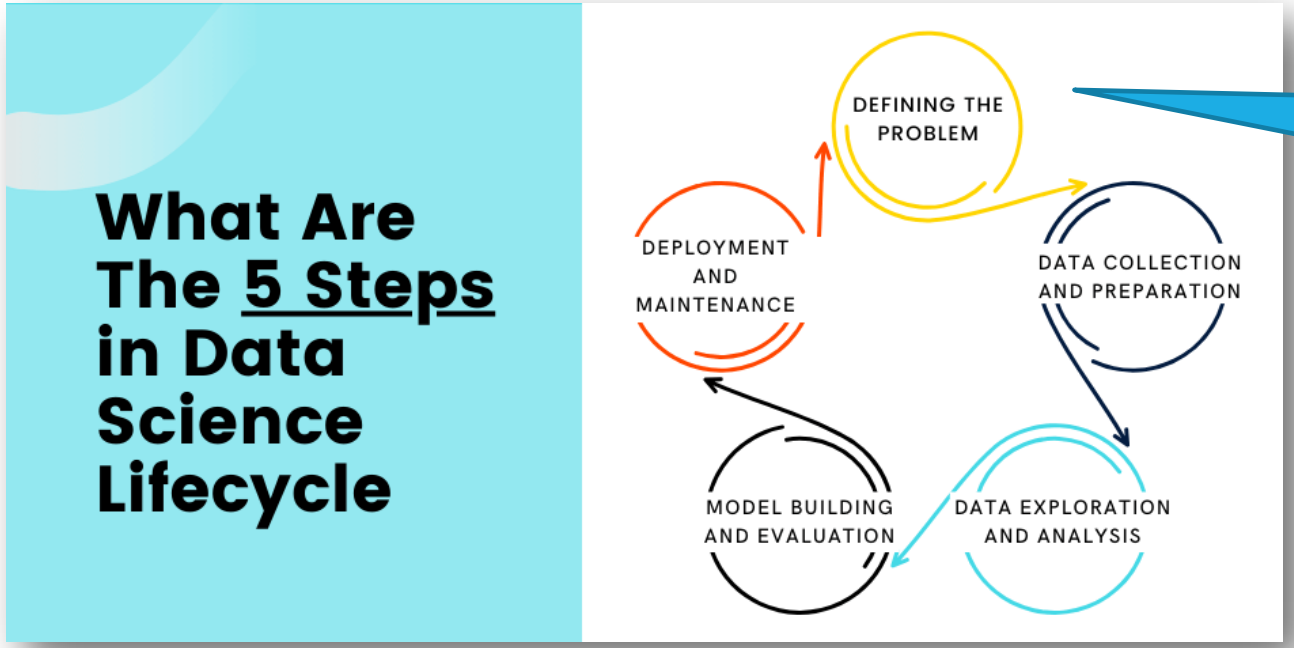
SURVEY

Population targetted is the proper one ?
- University students versus Computer Science students

Attributes gathered sufficient ?
- Hours of sleep per week was not asked but we would like to find correlations with academic achievement.

SECONDARY DATA

Data does not contain required attributes.



Completeness is related to a good problem definition and good planning of the data collection

2. Accuracy

Is the data an accurate representation of the element it describes?

Does the information reflect reality?

- Price of food (where is it taken?)
- Age of employees (how is it entered)

This is related to trust of sources/users.

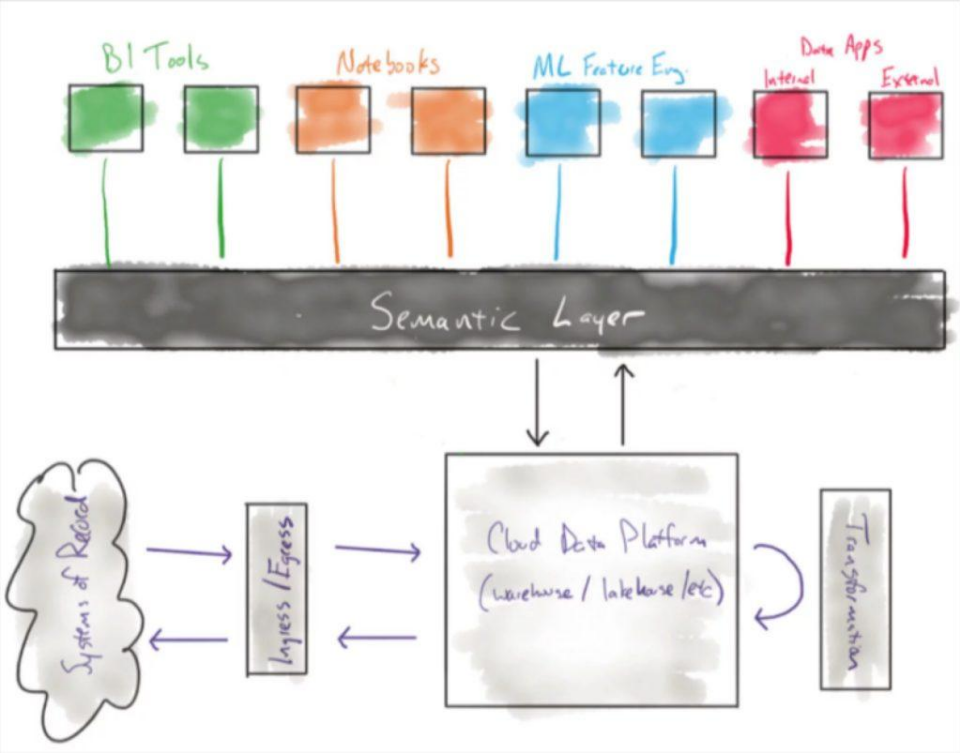
Cleaning can detect anomalies… but does not replace a trustable source!

20$/kg

3. Consistency

Is your data synchronized across your organization?

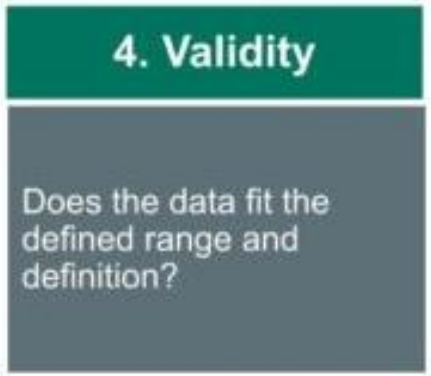
Unique dataset with shared views within the organization?
Data fields different from dataset to another?



Column names (surface form)
How-Much, Cost, Price, Amount

How much money is paid (meaning)

Source



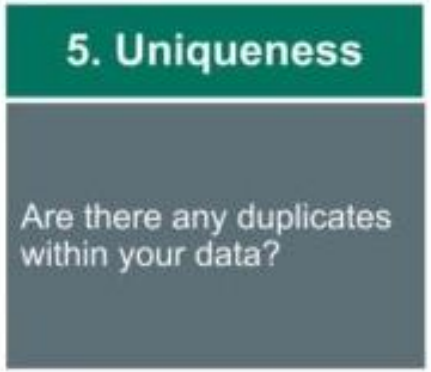
A LOT of cleaning is about these types of errors

Rules on attributes

- Salary cannot be negative

Rules on pairs of attributes

- Job title cannot be « Manager » and salary smaller than X



Redundant data can impact results in descriptive and predictive analysis.

Removing duplicates is always mentioned in cleaning

But what about near-duplicates

I let you explore this term!!

Sometimes defined as part of Data Governance…
Sometimes defined as encompassing Data Quality….

Whenever something changes and such change cannot be traced back, there is a problem of integrity.

Collibra

The 6 data quality dimensions with examples

Data quality goes beyond what cleaning can do...

1. Completeness	2. Accuracy	3. Consistency
How complete is your dataset?	Is the data an accurate representation of the element it describes?	Is your data synchronized across your organization?

4. Validity	5. Uniqueness	6. Integrity
Does the data fit the defined range and definition?	Are there any duplicates within your data?	Can your data be traced and connected across the organization?

Let's see what we can clean!

Source

Duplicates

Duplicate

The repetition of identical data points.

Complete case: Rows are identical over all columns

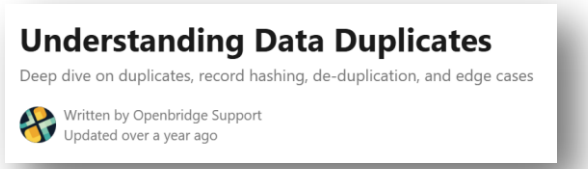
```
# 23. Using the duplicated() method to create a series of boolean values
duplicates = df.duplicated()

# 24. Displaying the duplicated rows
df[duplicates]
```

Find the complete duplicates.
Remove rows so only one datapoint is left.

Source

Near-duplicates are more challenging!

ID	Name	Gender	Salary ($)	Loan Amount ($)	Loan Outcome	Loan Type
1	Ms. Phi	F	120,000	48,000	No default	Car
2	Mr. Psi	M	100,000	Unknown	No default	Student loan
3	Ms. Tau	F	(40,000)	16,000	No default	Mortgage
4	Mr. Epsilon	F	90,000	36,000	Defaulted	Car
5	Mr. Rho	M	83,000	33,200	No default	
6	Mr. Rho	M	83,000	33,200	No default	Mortgage
7	Ms. Chi	F	95,000	38,000	No default	Mortgage



Order ID	Customer ID	Product ID		Quantity	Order Date
1	1234	5678		2	2021-01-01
1	1234	5678		2	2021-01-01
1	1234	5678		2	2021-01-01
1	1234	5678		2	2021-01-01
1	1234	5678		2	2021-01-01

Careful with transactional data!

False Positives

Source

In the same order, the customer bought 5 items.

Order Item ID	Order ID	Customer ID	Product ID		Quantity	Order Date
1	1	1234	5678		2	2021-01-01
2	1	1234	5678		2	2021-01-01
3	1	1234	5678		2	2021-01-01
4	1	1234	5678		2	2021-01-01
5	1	1234	5678		2	2021-01-01

Order Item ID	Order ID	Customer ID	Product ID		Quantity	Order Date
1	1	1234	5678		2	2021-01-01
2	1	1234	5678		2	2021-01-01
3	1	1234	5678		2	2021-01-01
4	1	1234	5678		2	2021-01-01
5	1	1234	5678		2 10	2021-01-01

If too fine grain, perhaps transactional data need to be grouped for data analysis

On Jan 1, 2023, there was a product called `Red Widget Max` in your listings:

Product ID	Product Title	Price	Timestamp
1	Red Widget Max	$10	2023-01-01

However, on March 1, 2023, a product manager decided to change the Product title to `Red Widget Ultimate`

Product ID	Product Title	Price	Timestamp
1	Red Widget Ultimate	$10	2023-03-01

Slowly Changing Data

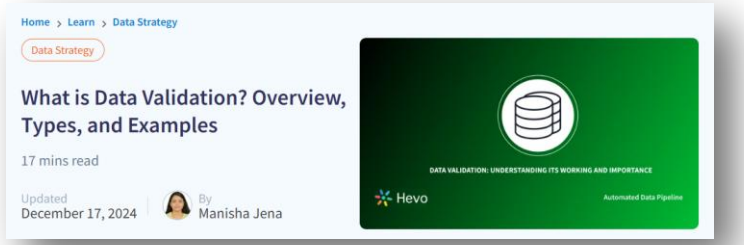
This will result in two rows for the same product ID:

Product ID	Product Title	Price
1	Red Widget Max	$10
1	Red Widget Ultimate	$10

Some ID fields should be unique... so how to deal with this?

Duplicates	
What can cause them?	Human entry error Dataset merging Slow changing data
What harm can they do?	Can bias an analysis by inflating counts.
Easy cases	Identical rows (complete cases)
Difficult cases	Near-duplicates (often require domain expertise for plausibility) Duplicates arising from slow changing data
Cleaning errors	False positives (removing non-duplicates) False negatives (not-removing duplicates)

Invalid data



The following are the common Types:

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Validation at entry time versus post-processing (cleaning)

Link to UI and validation (front end/Surveys) – if NOT done there… lots of additional work in the back

Source

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Data Type

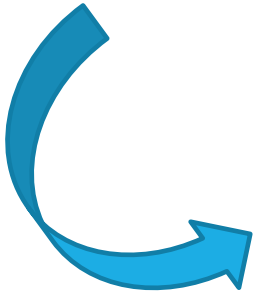
Input validation

A **Data Type check** ensures that data entered into a field is of the correct data type. A field, for example, may only accept numeric data. The system should then reject any data containing other characters, such as letters or special symbols, and an error message should be displayed.

Dataset cleaning

Name	Age	Salary	Join_Date	Is_Manager
Alice	25	50000.0	2020-05-10	True
Bob	Unknown	45000.0	2021-07-15	False
Charlie	30	None	Invalid Date	Yes
Diana	28.5	"60000"	2022-03-20	No
Edward	27	55000.0	2023-01-01	TRUE

Should fit these types



```
Name            object
Age            float64
Salary         float64
Join_Date    datetime64[ns]
Is_Manager       bool
```

we need these types

Codes

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Input validation

A **Code Check** ensures that a field is chosen from a valid list of values or that certain formatting rules are followed. For example, it is easier to verify the validity of a postal code by comparing it to a list of valid codes. Other items, such as country codes and NAICS industry codes, can be approached in the same way.

Dataset cleaning

Harmonized System

From Wikipedia, the free encyclopedia

The **Harmonized Commodity Description and Coding System** (**HS**) of tariff nomenclature is an internationally standardized system of names and numbers for classifying traded products developed and maintained by the World Customs Organization (WCO) (formerly the Customs Co-operation Council), an independent intergovernmental organization with over 170 member countries based in Brussels, Belgium.

Example for import/export

01-05	Animal & Animal Products
06-15	Vegetable Products
16-24	Foodstuffs
25-27	Mineral Products
28-38	Chemicals & Allied Industries
39-40	Plastics / Rubbers
41-43	Raw Hides, Skins, Leather, & Furs
44-49	Wood & Wood Products
50-63	Textiles
64-67	Footwear / Headgear
68-71	Stone / Glass
72-83	Metals
84-85	Machinery / Electrical
86-89	Transportation
90-97	Miscellaneous
98-99	Service

07.10 **Vegetables (uncooked or cooked by steaming or boiling in water), frozen.**

0710.10 - Potatoes

- Leguminous vegetables, shelled or unshelled:
0710.21 - - Peas (*Pisum sativum*)
0710.22 - - Beans (*Vigna spp., Phaseolus spp.*)
0710.29 - - Other

0710.30 - Spinach, New Zealand spinach and or ache spinach (garden spinach)

0710.40 - Sweet corn

0710.80 - Other vegetables

0710.90 - Mixtures of vegetables

Example for import/export

22.02 **Waters, including mineral waters and aerated waters, containing added sugar or other sweetening matter or flavoured, and other non-alcoholic beverages, not including fruit or vegetable juices of heading 20.09.**

2202.10 - Waters, including mineral waters and aerated waters, containing added sugar or other sweetening matter or flavoured

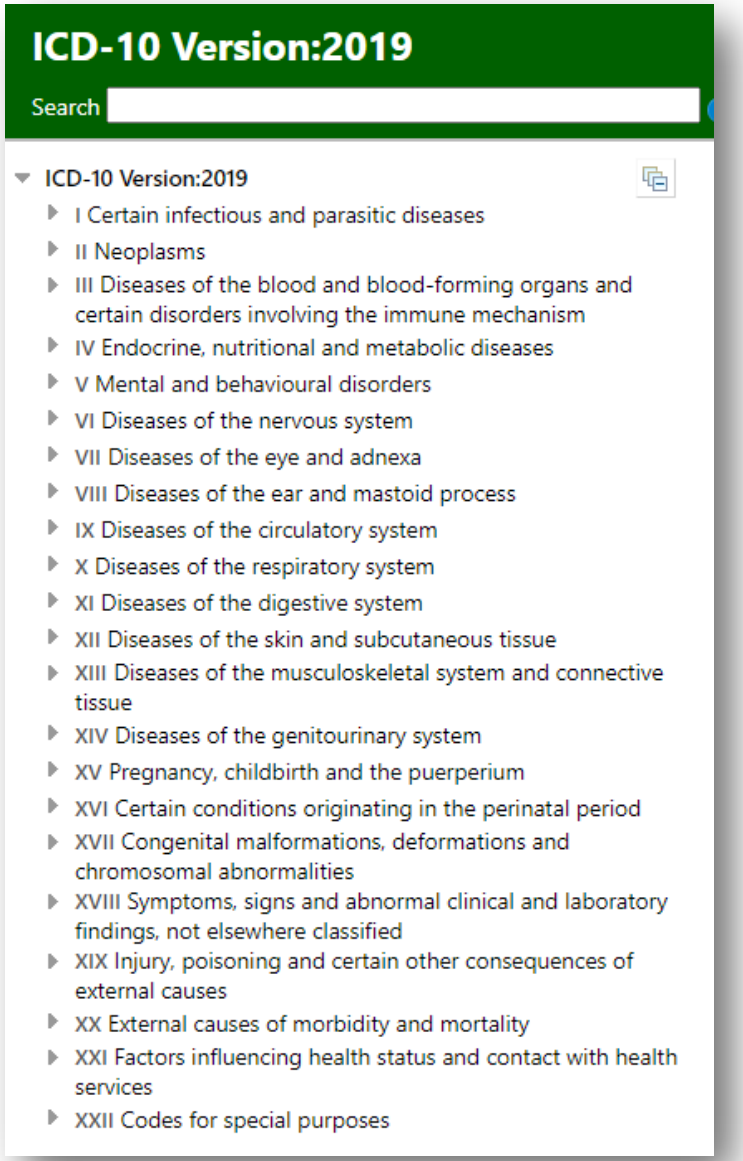
2202.90 - Other

22.03 **Beer made from malt.**

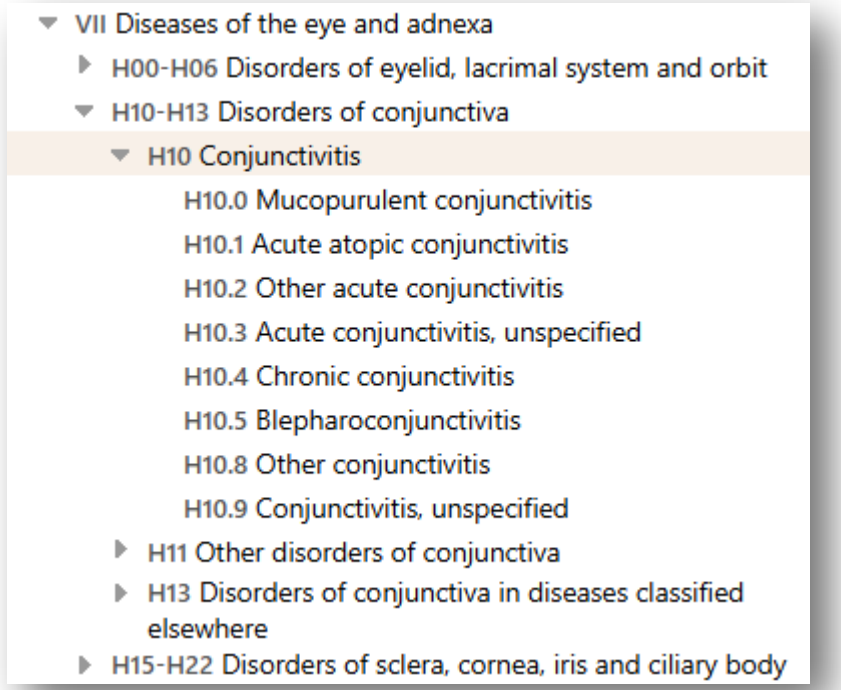
2203.00 no subheadings

22.04 **Wine of fresh grapes, including fortified wines; grape must other than that of heading 20.09.**

2204.10 - Sparkling wine



Example for patient discharge coding (insurance)



Source

Range

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Input validation

A **Range Check** will determine whether the input data falls within a given range. Latitude and longitude, for example, are frequently used in geographic data. Latitude should be between -90 and 90, and longitude should be between -180 and 180. Any values outside of this range are considered invalid.

need to know range

Negative value

1	ID	Name	Gender	Salary ($)	Loan ount ($)	Loan Outcome	Loan Type
2	1	Ms. Phi	F	120,000	48 0	No default	Car
3	2	Mr. Psi	M	100,000	nknown	No default	Student loan
4	3	Ms. Tau	F	(40,000)	16,000	No default	Mortgage
5	4	Mr. Epsilon	F	90,000	36,000	Defaulted	Car
6	5	Mr. Rho	M	83,000	33,200	No default	
7	6	Mr. Rho	M	83,000	33,200	No default	Mortgage
8	7	Ms. Chi	F	95,000	38,000	No default	Mortgage

cannot be negative

```python
# Example in Python: Removing invalid data based on conditions
df = df[df['age'] > 0]
```

Impossible age

Employee_ID	Age	Salary	Hours_Worked_Per_Week
E001	25	50000	40
E002	16	45000	50
E003	120	60000	38
E004	35	250000	20
E005	30	-1000	70

What are the ranges?

Out-of-range or outlier?

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Format

Input validation

Many data types have a predefined format. A Format Check will ensure that the data is in the correct format. Date fields, for example, are stored in a fixed format such as “YYYY-MM-DD” or “DD-MM-YYYY.” If the date is entered in any other format, it will be rejected.

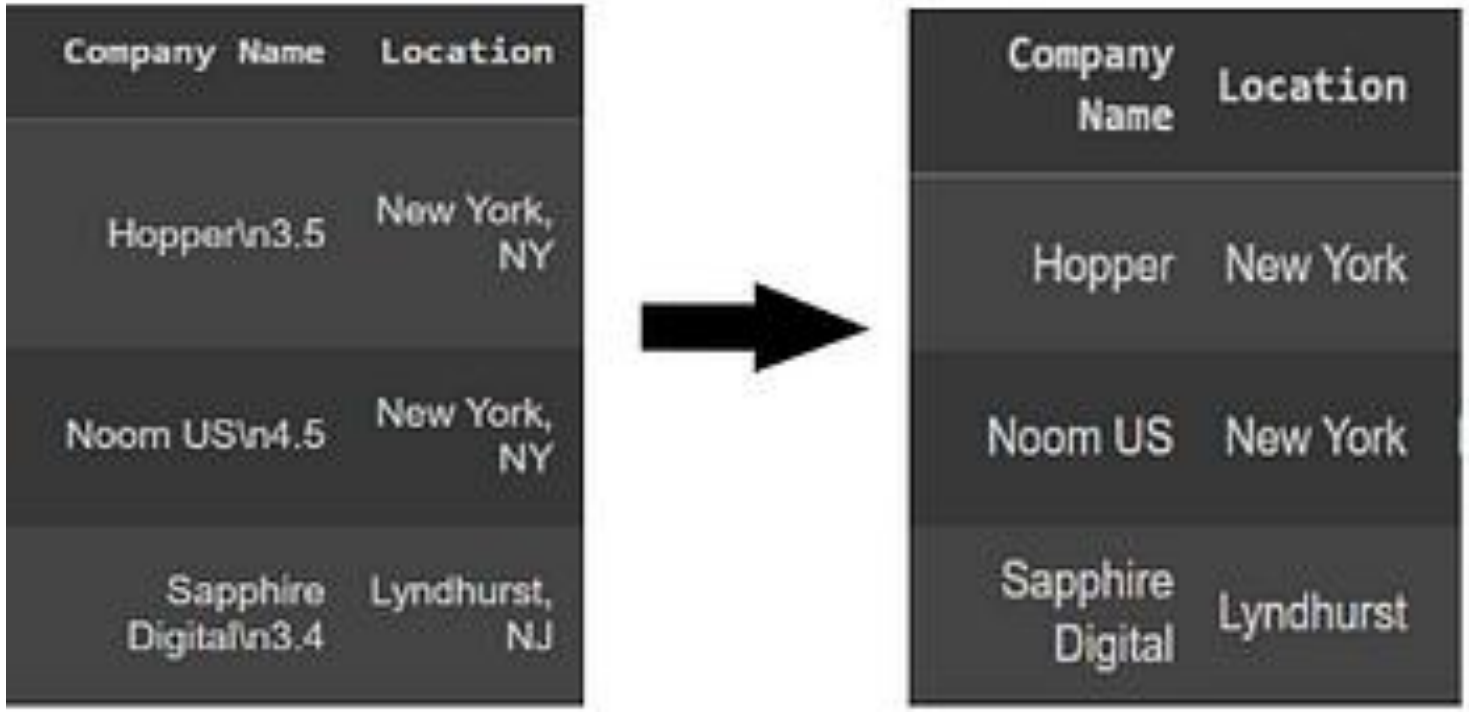
Dataset cleaning

Use of standards

For instance, if a dataset has a “Date” column with both “YYYY-MM-DD” and “MM/DD/YYYY” formats, you can standardize them into a single format for uniformity.

```
# Example in Python: Converting dates to a single format
df['date'] = pd.to_datetime(df['date'], format='%Y-%m-%d')
```

Non-interpretable text format



Example of a string expressing a numerial range.

Non-interpretable text format

Salary Estimate
$111K-$181K (Glassdoor est.)
$111K-$181K (Glassdoor est.)
$111K-$181K (Glassdoor est.)

Salary Estimate
111000-181000
111000-181000
111000-181000

Estimate Minimum Salary	Estimate Maximum Salary
111000	181000
111000	181000
111000	181000

Consistency

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Input validation

A **Consistency Check** is a type of logical check that ensures data is entered in a logically consistent manner. Checking if the delivery date for a parcel is after the shipping date is one example.

Business rules

Dataset cleaning

Loan outcome depends on loan amount

Shouldn't be available

1	ID	Name	Gender	Salary ($)	Loan Amount (	Loan Outcome	Loan Type
2	1	Ms. Phi	F	120,000	48,000	No default	Car
3	2	Mr. Psi	M	100,000	Unknown	No default	Student loan
4	3	Ms. Tau	F	(40,000)	16,000	No default	Mortgage
5	4	Mr. Epsilon	F	90,000	36,000	Defaulted	Car
6	5	Mr. Rho	M	83,000	33,200	No default	
7	6	Mr. Rho	M	83,000	33,200	No default	Mortgage
8	7	Ms. Chi	F	95,000	38,000	No default	Mortgage

Salary is dependent of company's rules

Employee ID	Job Title	Salary ($)
201	Intern	25,000
202	Senior Engineer	120,000
203	CEO	250,000
204	Junior Analyst	180,000

Uniqueness

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Input validation

Some data, such as IDs or e-mail addresses, are inherently unique. These fields in a database should most likely have unique entries. A **Uniqueness Check** ensures that an item is not entered into a database more than once.

Uniqueness of value for a column

Dataset cleaning

Employee ID	Name	Email	Job Title	Salary ($)
401	Emily Rivera	info@travelco.com	Tour Guide	45,000
402	James Carter	james@travelco.com	Travel Agent	55,000
403	Michael Lee	info@travelco.com	Hotel Manager	80,000
404	Sophia Adams	sophia@travelco.com	Event Coordinator	50,000

Are emails really unique? Credit card number? Phone number?

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Presence

Input validation

A **Presence Check** ensures that all mandatory fields are not left blank. If someone tries to leave the field blank, an error message will be displayed, and they will be unable to proceed to the next step or save any other data that they have entered. A key field, for example, cannot be left blank in most databases.

Dataset cleaning

Employee ID	Name	Email	Job Title	Salary ($)
901	Alice Green	alice@company.com	Software Engineer	95,000
902	Bob White		Data Scientist	85,000
903	Charlie Black	charlie@compan om	HR Manager	75,000
904	Dana Blue	dana@company.com	ting Sp	65,000

Missing values!! We'll look at this later (very important!)

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Length

Input validation

A **Length Check** ensures that the appropriate number of characters are entered into the field. It verifies that the entered character string is neither too short nor too long. Consider a password that must be at least 8 characters long. The Length Check ensures that the field is filled with exactly 8 characters.

Dataset cleaning

This is length and also format

Employee ID	Name	Phone Number	Em	Job Title	Salary ($)
701	John A. Smith	123-45-6789	n.smith@email.com	Software Engineer	85,000
702	Olivia	123-4	olivia@email.com	Data Analyst	75,000
703	Samuel Johnson	987-654-3210	samuel.johnson@email.com	HR Specialist	60,000
704	Catherine Elizabeth Alexandra Montgomery-Smith	555-123-4567	catherine@email.com	Project Manager	95,000

Is this really « impossible »?

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Look-up

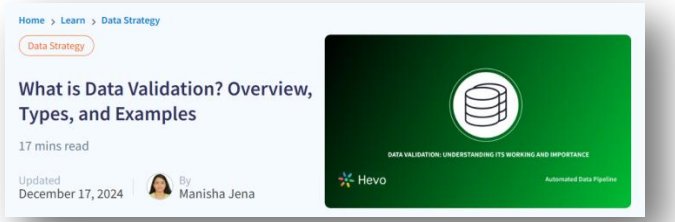
Input validation

Look Up assists in reducing errors in a field with a limited set of values. It consults a table to find acceptable values. The fact that there are only 7 possible days in a week, for example, ensures that the list of possible values is limited.

Dataset cleaning

Product_ID	Category	Price_Code	Region_Code	Stock_Status
P001	Electronics	A	North_1	In_Stock
P002	Electroncs	B	N_1	InStok
P003	Clothing	C	South_1	Out_of_Stock
P004	Clothing	4	S_1	OutofStock
P005	Electronics	B+	East_1	In_Stock

Don’t let people type, provide a selection in a menu/list



The following are the common Types:

1) Data Type Check
2) Code Check
3) Range Check
4) Format Check
5) Consistency Check
6) Uniqueness Check
7) Presence Check
8) Length Check
9) Look Up

Validation problems	
What can cause them?	
What harm can they do?	
Easy cases	
Difficult cases	
Cleaning errors	

Source

DATA CLEANING

- Data wrangling
- Data quality
- Data cleaning
  - Duplicates
  - Invalid data
  - Outliers

Part 1

Part 2