Data, Data Storage, Data Collection

Lecture 6: Data Wrangling

Romain Pascual

MICS, CentraleSupélec, Université Paris-Saclay

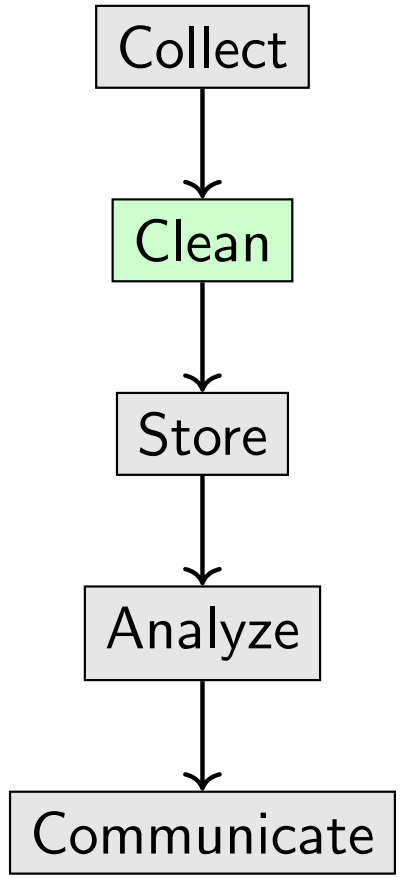
Recap

Recap: Practical Data Collection

1. Different methods (experiments, surveys, sensors, APIs) serve different purposes
2. Every method comes with **advantages** and **limitations**
3. Bias is unavoidable, but can be identified, reduced, and documented
4. Good provenance and FAIR principles make data **trustworthy** and **reusable**
5. Poor documentation = wasted time, incorrect conclusions, or ethical risks.
6. Always ask: does my dataset reflect what I think it reflects?

Introduction

Within the lifecycle



Session Objectives

At the end of this session, you should be able to:

- Explain what **data quality** means and why it matters
- Identify **issues in raw data**: missingness, duplicates, outliers, inconsistent formats.
- Apply strategies for **handling missing data** (deletion, imputation, flagging).
- Perform basic **data transformation** and wrangling tasks in Python (Pandas).
- Discuss the **ethical considerations** of data cleaning.

Why Data Quality Matters?

- Models and algorithms are only as good as the data they use.
- Biased or incorrect data $\rightarrow$ biased or incorrect conclusions.
- Automation amplifies errors

"Garbage In, Garbage Out (GIGO)"

NASA (1999)

Mars Climate Orbiter crashed due to metric vs. imperial unit mismatch.

Excel Gene Names

Auto-conversion corrupted research (e.g., "SEPT2" $\rightarrow$ "September 2").

⚠ So we need to clean the data.

Not just a technical step but a **critical thinking** exercise.

Data Cleaning

Definition (Data Cleaning)

Process of identifying and correcting (or removing) corrupt, inaccurate, or irrelevant data.

Goal: Transform data from a **raw state** into a **reliable, consistent, and usable** state.

Cleaning depends on **how** and **what** data was collected:

- The collection method introduces specific data quality issues
- Qualitative and quantitative data might have different issues

Poor collection leads to harder cleaning.
But good documentation helps guiding the cleaning process.

Initial Thoughts on Data Quality

Depending on the source, we often have different expectations for data quality:

- Some datasets require **extensive** wrangling to be made analyzable.
- Other arrive clean and ready for analysis with **minimal** wrangling.

Scientific Experiment or Study

How much cleaning do you expect for this data?

Scientific Experiment or Study

How much cleaning do you expect for this data?

- Usually **clean**, **well-documented**, and with **a simple structure**.
- Designed for **reproducibility and sharing**.

$\rightarrow$ Typically ready for analysis after little to no wrangling.

Administrative Data from Organizations

How much cleaning do you expect for this data?

Administrative Data from Organizations

How much cleaning do you expect for this data?

- May be clean but requires **inside knowledge** of the source.
- Often repurposed for **secondary analysis**.
- May need **transformation or combining tables**.

$\rightarrow$ Often requires moderate cleaning, especially if used for a new purpose.

Informally Collected Data

How much cleaning do you expect for this data?

Informally Collected Data

How much cleaning do you expect for this data?

- Collected in an **ad-hoc or non-systematic** way.
- Often **messy with little documentation**.

$\rightarrow$ Typically requires extensive formatting and cleaning.

Data Wrangling: The Roadmap Ahead

Definition (Data wrangling, or data munging)

Process of preparing data for downstream tasks like modeling, visualization, or reporting.

Data wrangling can be splitted into the following stages:

1. Assessing data quality
2. Handling missing values
3. Transforming the features
4. Reshaping the data (by modifying its structure and granularity)

Goal: prepare data for analysis **without distorting its meaning**, balancing usability and integrity.

Data Quality

Data Quality

Definition (Data Quality)

How **fit for purpose** the data is for its intended use.

1. **Validity**: Does data conform to rules?
   - E.g., correct data types, ranges, or mandatory fields.
   - Example: Age = "-5" is invalid.
2. **Accuracy**: Does data reflect the true value?
   - Hard to verify! E.g., self-reported survey data.
3. **Completeness**: Are all required values present?
   - Missingness often stems from **poor collection**.
4. **Consistency**: Is data equivalent across systems?
   - E.g., customer IDs matching in CRM and sales databases.
5. **Uniformity**: Are units standardized?
   - E.g., kg vs. lbs, "USA" vs. "United States."

Data Quality is further specified by the **ISO 25012** standard.

How to Assess Data Quality?

5 Key Questions

1. **Scope**: Does the data cover the target population?
2. **Measurement**: Are values precise and recorded correctly?
3. **Features**: Are relationships between variables logical?
4. **Analysis-Readiness**: Can the data answer your research question?
5. **Actionability**: Should you fix issues, or document limitations?

In practice: quality metrics and profiling tools to automate assessments.

Data Quality Metrics

Common Metrics

- % of missing values (e.g., 15% of "Age" entries are null).
- Number of duplicate rows (e.g., 10 duplicate customer IDs).
- % of valid entries (e.g., 95% of "Email" fields are valid).
- Distribution statistics (e.g., outliers in "Income").
- Number of unique values (e.g., categories in "Gender").

Definition (Data Profiling)

Systematic analysis to understand **structure, content, and quality** of data **before** cleaning.

Some tooling can help the automation, e.g., `ydata-profiling`.

Handling Missing Data

Why Would Data Be Missing?

Data can be missing for many reasons:


Sensor Failure


Connection Failure


Missing Answers


Human Error


Privacy Concerns


Data Corruption

Missing data is often not random: it reflects underlying issues in collection or context (or storage).

Sentinel Values

Definition (Sentinel Value)

A special symbol value indicating missing data.

Example: the value returned by an algorihtm might indicate that it did not find a solution.

Problem: Find the index of a value in an array, return `-1` if the value does not exist.

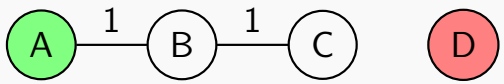
Algorithm: Linear search

```
[3, 7, 1, 5, 9]
```

- For `5` $\rightarrow$ `3`
- For `2` $\rightarrow$ `-1`

Problem: Find the value of the shortest path from s to t in a graph.

Algorithm: Dijkstra's



- From `A` to `C` $\rightarrow$ `2`
- From `A` to `D` $\rightarrow \infty$

The exact values and their meanings **depend on the context**.

Missing Data Representation

In practice, standard libraries offer dedicated sentinel values:

- `nan` (**N**ot **a** **N**umber) in NumPy: special floating point number
- `None` in base Python: poor integratation with libraries (NumPy, Pandas)
- `NaT` (**N**ot **a** **T**ime) in Pandas: specific object of type `Timestamp`
- `NA` (**N**ot **A**vailable) in Pandas

Sentinel Values Behave Differently

Equality testing

```
>>> None == None
```

Sentinel Values Behave Differently

Equality testing

```
>>> None == None
True
>>> np.nan == np.nan
```

Sentinel Values Behave Differently

Equality testing

```
>>> None == None
True
>>> np.nan == np.nan
False
>>> pd.NaT == pd.NaT
```

Sentinel Values Behave Differently

Equality testing

```
>>> None == None
True
>>> np.nan == np.nan
False
>>> pd.NaT == pd.NaT
False
>>> pd.NA == pd.NA
```

Sentinel Values Behave Differently

Equality testing

```
>>> None == None
True
>>> np.nan == np.nan
False
>>> pd.NaT == pd.NaT
False
>>> pd.NA == pd.NA
<NA>
```

Sentinel Values Behave Differently

Equality testing

```
>>> None == None
True
>>> np.nan == np.nan
False
>>> pd.NaT == pd.NaT
False
>>> pd.NA == pd.NA
<NA>
```

Formally: NA uses a three-values logic (Kleene's strong logic of indeterminacy)

Pandas documentation (`NA` semantics)

"Experimental: the behaviour of `NA` can still change without warning".

Strategies for Handling Missing Data

Deletion

- Simple but may introduce bias
- Only remove features if they are mostly empty or irrelevant (wrt. to the question aim)

Imputation

- Statistical methods (mean, median)
- Forward or backward fill
- Prediction (K-NN, ML)

Flagging

- Add a binary column (e.g., `is_missing`) to indicate missing values.

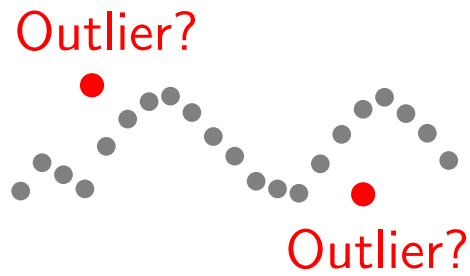
You need to understand why data are missing, not just how many values are missing.

Outliers

Definition (Outliers)

Outliers are data points that differ significantly from others.

⚠ Not necessarily irrelevant: possible interesting phenomena.



Detection methods

- Vary and are often case-specific or based on domain knowledge.
- Statistical tests and visualization are a good starting point.

Handling outliers

- Strategies are the same as for missing data: deletion, imputation, or flagging.

Data Transformation

Why Transform Data?

After handling missing values and outliers, we need to ensure the data is usable and relevant for analysis.

- Ensure data is in a format compatible with analysis tools.
- Improve data quality and usability.
- Optimize data for better performance in analysis or modeling.

Underlying questions:

- Are all observations relevant?
- Are all features necessary for the analysis?

⚠ Ensure that transformations are **reproducible** and **readable** (document them!)

Simple Transformations

Transforming Features

Transformations are typically operations on columns in a dataframe.

- **Renaming columns:** Ensure consistency and readability.
  - `First Name` $\rightarrow$ `first_name`
- **Removing irrelevant or redundant columns:** Focus on the question at hand.
  - Is the column relevant to the analysis? The answer depends on the research aim (the question).
- **Applying functions to values:**
  - **Numeric:** Adjust, normalize, or compute new values.
  - **Textual:** Trim whitespace, fix capitalization, correct typos.
  - **Dates:** Convert to standard formats.

Filtering Data

Definition (Filtering)

Selecting a subset of observations (rows) based on specific criteria or conditions (logical formulae).

Example

Name	Age	Income
Alice	25	50000
Bob	30	60000
Carol	28	55000
Dave	32	65000

$\xrightarrow{Age > 28}$

Name	Age	Income
Bob	30	60000
Dave	32	65000

Use Cases

- Extracting data for specific analyses or visualizations.
- Improve computational efficiency by reducing dataset size.

Technically similar to missing data, but different motivation.

Normalization

Why Normalize?

- Ensures comparability of features.
- Improves performance of distance-based algorithms (e.g., k-NN, k-means).

Min-Max Normalization

- Scales features to [0, 1].
- Formula: $x' = \frac{x - \min(X)}{\max(X) - \min(X)}$.
- Use Case: Suitable when you know the bounds of your data.

Z-Score Standardization

- Scales features to have mean 0 and standard deviation 1.
- Formula: $x' = \frac{x - \mu}{\sigma}$.
- Use Case: Suitable when your data follows a Gaussian distribution.

Encoding Categorical Variables

Definition (Encoding)

Converting categorical data into a numeric format suitable for analysis or modeling.

Label Encoding

Assign integer codes to categories.
Cloth Size: "Small"=0, "Medium"=1, "Large"=2

One-Hot Encoding

One binary column per category (or $n-1$ to avoid redundancy).
Color: "Red", "Blue", "Green" $\rightarrow$ [1,0,0], [0,1,0], [0,0,1]

Considerations

Always check if encoding preserves the meaning of the data.

Discretization

Definition (Discretization)

Converting continuous numerical data into discrete categories.

Why Discretize? Simplification and interpretability.

Methods

- Equal Width Binning: intervals of equal width.
- Equal Frequency Binning: bins with the same number of observations.
- Clustering: group similar data points

Example

- Temperature: Cold, Warm, Hot.

Aggregation

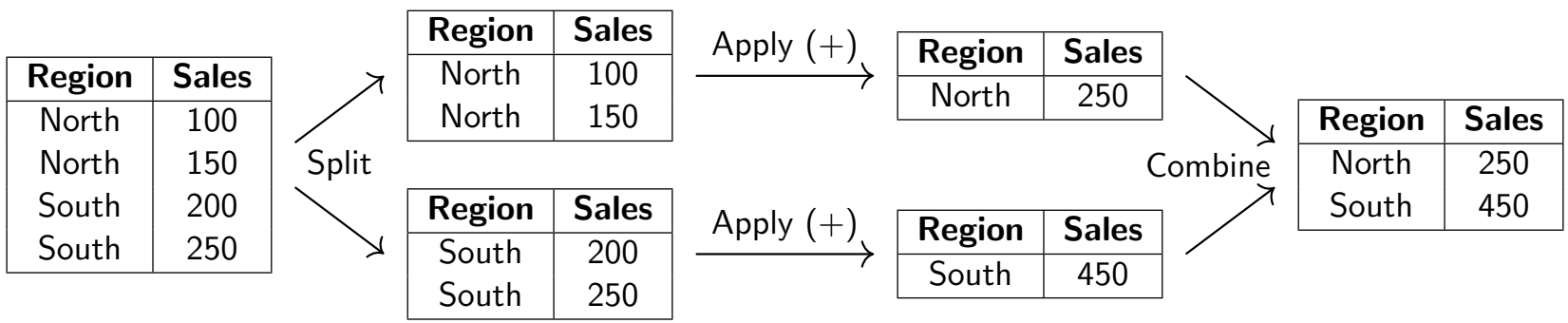
Definition (Aggregation)

Grouping data and computing statistics over groups.

Process ($\sim$ Map-Reduce)

1. **Split:** Divide the data into groups based on a key.
2. **Apply:** Compute statistics within each group.
3. **Combine:** Merge the results back.

Example: Sum of Sales by Region



Feature Engineering

Definition (Feature Engineering)

Deriving new columns from existing ones to improve model performance.

Examples

- Creating a "Body Mass Index" feature from height and weight data.
- Extracting day, month, and year from a date column.

Best Practices

- Ensure new features are meaningful and relevant.
- Document the process for reproducibility.

Chaining Operations with `.pipe()`

Chaining operations can create a pipeline of transformations.

```
# Sample DataFrame
df = pd.DataFrame({
    'First Name': ['John', 'Jane', 'Jim'],
    'Age': [25, 30, 35],
    'Income': [50000, 60000, 70000]
})

def rename_columns(df):
    return df.rename(columns={'First Name': 'first_name'})

def normalize_income(df):
    df['income_normalized'] = (df['Income'] - df['Income'].min()) / (df['Income'
     ].max() - df['Income'].min())
    return df

def discretize_age(df):
    bins = [0, 30, 40, 100]
    labels = ['Young', 'Middle-aged', 'Senior']
    df['age_group'] = pd.cut(df['Age'], bins=bins, labels=labels)
    return df

df_transformed = (df
    .pipe(rename_columns)
    .pipe(normalize_income)
    .pipe(discretize_age)
)
```

Reshaping Data

Why Reshape Data?

- Facilitate data analysis and visualization.
- Combine datasets for comprehensive analysis.
- Improve data readability and usability.

Pivoting

Definition (Pivoting)

Turning unique values from one column into new columns, converting data from long to wide format.

Computing the pivot for a dataset D with columns K (key), A (attribute), and V (values):

- Let $K = \{k_1, k_2, \ldots, k_m\}$ be the set of unique values in the key column.
- Let $A = \{a_1, a_2, \ldots, a_n\}$ be the set of unique values in the attribute column.
- The pivot operation constructs a new dataset D' where:
  - Rows are indexed by K.
  - Columns are indexed by A.
  - The value at $D'_{i,j}$ is the value v from D where $k = k_i$ and $a = a_j$.

Pivoting: Application

Example: Pivoting wrt. Date, Cat., and Sales

Date	Cat.	Sales
2025-10-10	A	100
2025-10-10	B	150
2025-10-10	C	100
2025-10-11	A	200
2025-10-11	B	250
2025-10-11	C	250

$\rightarrow$

Date	A	B	C
2025-10-10	100	150	100
2025-10-11	200	250	250

Use Cases

- Creating summary tables.
- Preparing data for visualization.

Melting

Definition (Melting)

Converting data from wide to long format.

Example

Date	A	B	C
2025-10-10	100	150	100
2025-10-11	200	250	250

$\rightarrow$

Date	Cat.	Sales
2025-10-10	A	100
2025-10-10	B	150
2025-10-10	C	100
2025-10-11	A	200
2025-10-11	B	250
2025-10-11	C	250

Use Cases

- Preparing data for analysis.
- Converting summary tables back to detailed records.

Transposing

Definition (Transpose)

Flipping a table over its diagonal, swapping rows and columns.

Example

Name	Age	Income
Alice	25	50000
Bob	30	60000

$\rightarrow$

	Alice	Bob
Age	25	30
Income	50000	60000

Use Cases

- Switching the orientation of data for better readability.
- Preparing data for specific types of analysis.

Concatenation

Definition (Concatenation)

Combining datasets along an axis (rows or columns).

Example

Name	Age
Alice	25
Bob	30

$+$

Name	Age
Carol	28
Dave	32

$\rightarrow$

Name	Age
Alice	25
Bob	30
Carol	28
Dave	32

Use Cases

- Combining datasets with similar structures.
- Stacking datasets vertically or horizontally.

Merging

Definition (Merging)

Combines datasets based on one or more keys (common columns).

Example

ID	Name
1	Alice
2	Bob

$\bowtie$

ID	Age
1	25
2	30

$\rightarrow$

ID	Name	Age
1	Alice	25
2	Bob	30

Use Cases

- Combining datasets with related information.
- Joining datasets based on common columns.

Conclusion

Ethical Dimension of Data Cleaning

⚠ Data cleaning is not a neutral act: removing, imputing, or transforming data can introduce or hide variability and bias.

Always question how cleaning decisions might affect the interpretation of the data.

Be particularly cautious with social or behavioral datasets, where biases can have significant real-world implications.

Takeaways: Data Wrangling

1. Start with initial assessment of the data quality
2. Handle missing data and outliers
3. Transform the data to prepare it for analysis
4. Reshape the data for better readability or comparability with specific analysis techniques.
5. Keep the original data and document the cleaning steps


SPENDING A MONTH CLEANING DATA
TOPPING THE RESULTS
OF A YEAR OF FINE TUNING
imgflip.com