



Data Wrangling

Cleaning, Transforming and Organizing Your Data





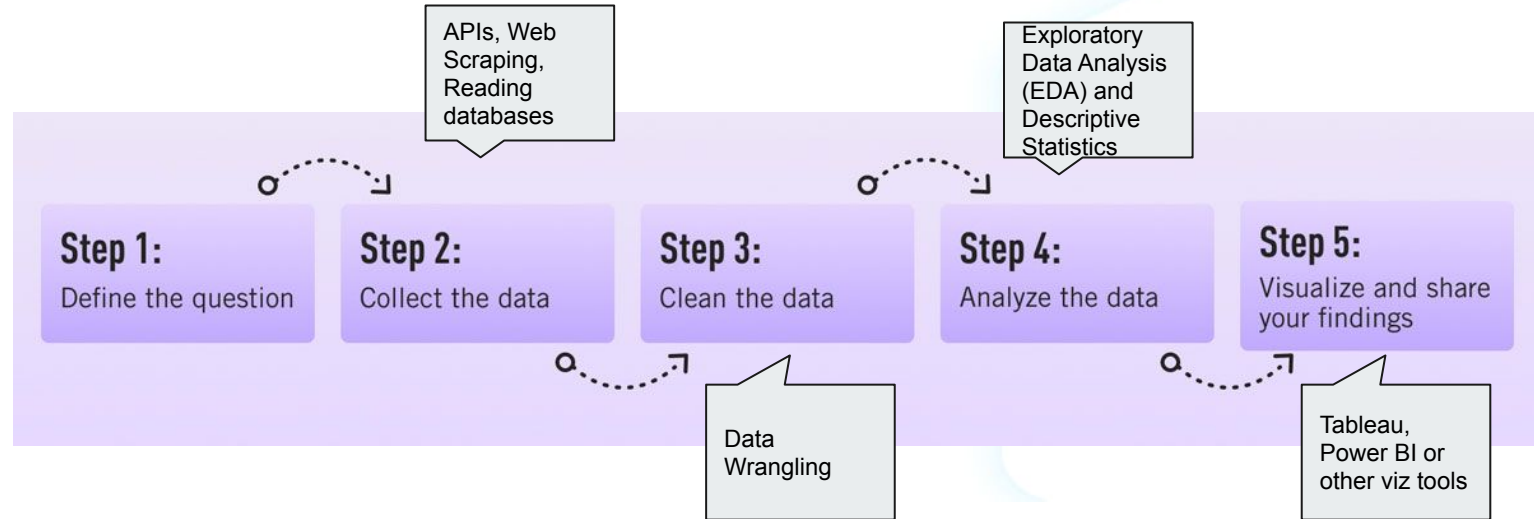
Topics Covered

- Introduction
- Data Cleaning
 - Missing data (Null values)
 - Outliers
 - Duplicated
 - Formatting
- Data Transformation:
 - Structuring: Reshaping Data
 - Combining: Merging and Joining
- Data Organization:
 - Grouping and Aggregating

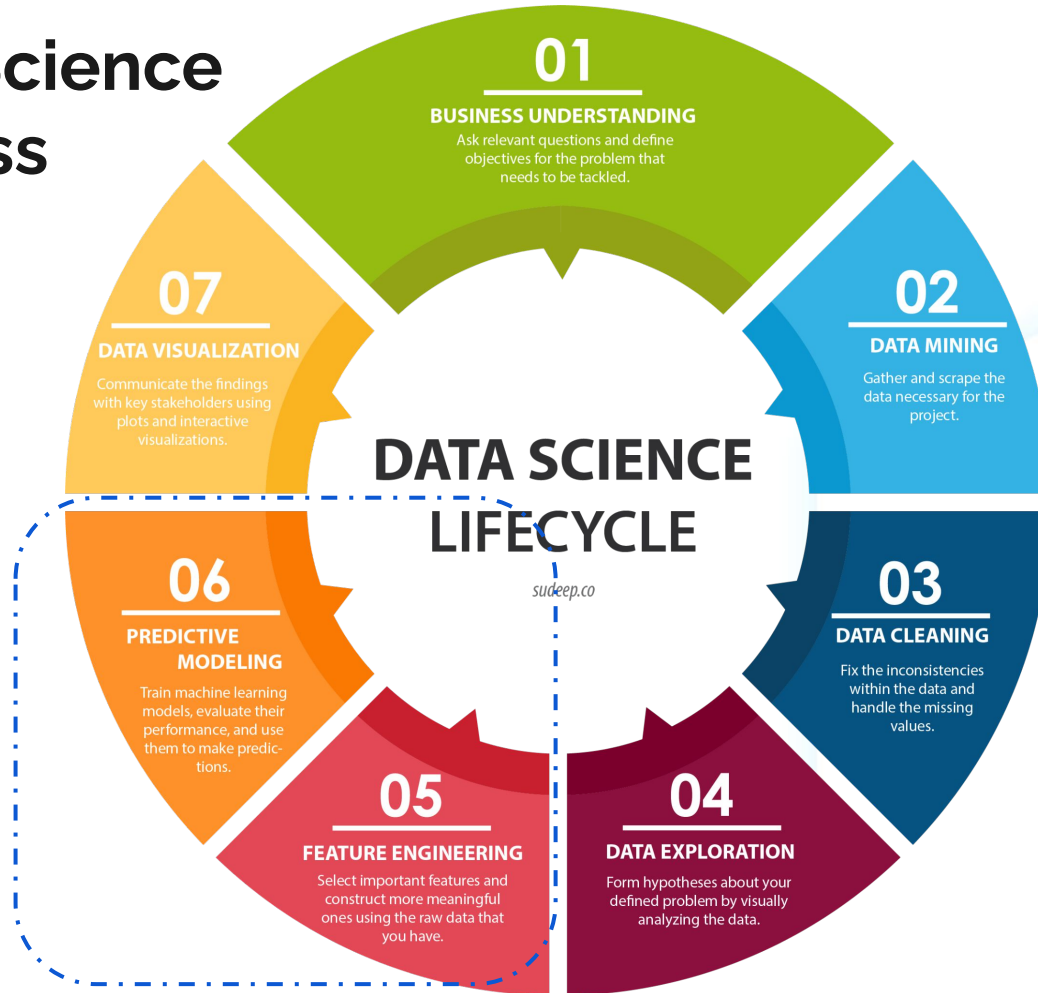


Introduction

Data Analytics Process



Data Science Process



Data Science Steps



What is Data Wrangling?

Data wrangling a **critical step** in the **data analytics process**. It involves **cleaning**, **transforming**, and **preparing** raw data to make it reliable data suitable for analysis.

60 to 80 percent of the total time is spent on cleaning the data before you can make any meaningful sense of it



Data Wrangling Steps

- **Data Cleaning**
 - Missing data (Null values)
 - Outliers
 - Duplicated
 - Formatting
- **Data Transformation**
 - Structuring: Reshaping Data
 - Combining: Merging and Joining
 - More: indexes, renaming columns, adding records/columns, editing information etc.
- **Data Organization**
 - Filtering data
 - Grouping and Aggregating
 - Ordering

Data Cleaning: missing values

Missing Data (null values)

Missing Values are incomplete or unavailable data in the dataset.

- They have different **origins**:
 - Data collection errors
 - Human errors
 - System failures
 - Undisclosed data etc.
- **Impact on Analysis**: Biased results, inaccurate insights, incomplete conclusions.

Row no	State	Salary	Yrs of Experience
1	NY	57400	Mid
2	TX		Entry
3	NJ	90000	High
4	VT	36900	Entry
5	TX		Mid
6	CA	76600	High
7	NY	85000	High
8	CA		Entry
9	CT	45000	Entry

Missing values

Missing Data (null values)

Row no	State	Salary	Yrs of Experience
1	NY	57400	Mid
2	TX		Entry
3	NJ	90000	High
4	VT	36900	Entry
5	TX		Mid
6	CA	76600	High
7	NY	85000	High
8	CA		Entry
9	CT	45000	Entry

Missing values

Handling Approaches:

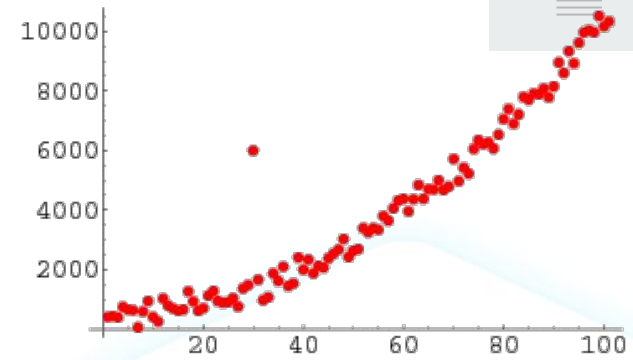
- Replace it with values by making use of information from **other columns**.
- **Remove** Rows with Missing Values: May lead to data loss and skewed analysis.
- **Impute** with Mean/Median/Interpolation: Fill missing values with a constant, central tendencies or interpolated values.
- **Machine Learning** to predict the value.



Data Cleaning: outliers



Outliers: Data Anomalies



Unusually extreme or rare data points. It is an abnormal observation that lies far away from other values.

- They have different **origins**:
 - Data measurement errors, natural variations, or data entry mistakes.
- **Impact on Analysis**
 - Skewed statistical measures, biased models, inaccurate predictions.
- **Example: Obvious errors**
 - Values too extreme, so that they are not plausible (1000 years)
 - Values that do not make sense (-10 years)

Do we discard or include?



Outliers: Data Anomalies

Handling Approaches:

- **Identify Outliers:**
 - Statistical methods (Z-score, IQR) or visualization (box plots, scatter plots).
- **Handle Outliers:**
 - Filtering: Removing outliers based on predefined thresholds.
 - Capping: Capping extreme values to a specified range.
 - Transformation: Applying mathematical transformations to mitigate impact.

Data Cleaning: duplicates

Duplicated values

Duplicates are records that are “identical” (with minor variations) but that represent the same entity..

- **Common Occurrence**
 - Data entry errors, system glitches, merging data from different sources.
- **Impact on Analysis**
 - Inflated statistical results, inaccurate insights, skewed findings.

ID	CITY
1	New York
2	New Yorkk



Duplicated values

Duplicates are “identical” records appearing multiple times in the dataset.

Handling Approaches:

- **Detect Duplicates:**
 - Check for identical rows or key columns.
 - Fuzzy matching, string similarity algorithms.
- **Remove Duplicates:**
 - Eliminate duplicate entries to maintain data accuracy.
 - Group similar records and retain clean, representative data.

Dealing with duplicates: Fuzzy Logic

Percentage of identical fields ($> 70\%$) or n identical values (> 10)

ID	MODEL
1	2021 Ford Focus
2	2019 Ford Focus

A field contains another one

ID	NAME
1	Peppa Pig
2	Mrs. Peppa Pig

Typing errors: number of different characters (< 3)

ID	CITY
1	New York
2	New Yorkk

Dealing with duplicates

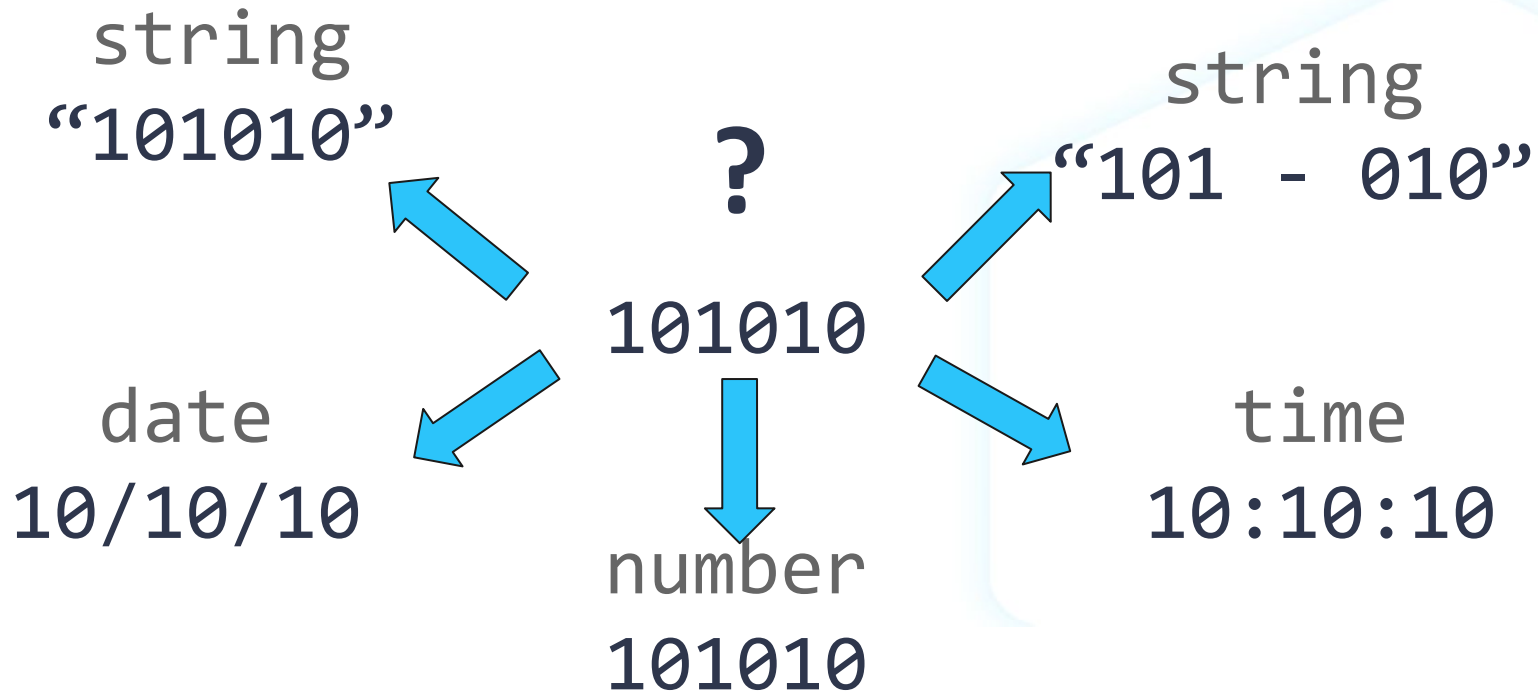
ID	NAME	ADDRESS	CITY	ZIP CODE
1	Peppa Pig	145 Brooklyn Ave	NEW YORK	11213
2	Mrs. Peppa Pig	Brooklyn Ave	New York	11213
3	PEPPA PIG	145 BROOKLYN AVE	New York	11213



Consolidate entries into one

ID	NAME	ADDRESS	CITY	ZIP CODE
1	Peppa Pig	145 Brooklyn Ave	New York	11213

Data Cleaning: formatting





Interpreting the data correctly

A variable can be interpreted in different ways, depending on:

- Meaning
- Data type

Correctly interpreting the data type is crucial. The format must fit all "known" cases.

Data Transformation: Structuring

Structuring Data

Reshaping data involves **converting** data from a wide format to a long format or vice versa.

Wide Format

Team	Points	Assists	Rebounds
A	88	12	22
B	91	17	28
C	99	24	30
D	94	28	31

Long Format

Team	Variable	Value
A	Points	88
A	Assists	12
A	Rebounds	22
B	Points	91
B	Assists	17
B	Rebounds	28
C	Points	99
C	Assists	24
C	Rebounds	30
D	Points	94
D	Assists	28
D	Rebounds	31

Pivot

df

	foo	bar	baz	zoo
0	one	A	1	x
1	one	B	2	y
2	one	C	3	z
3	two	A	4	q
4	two	B	5	w
5	two	C	6	t



```
df.pivot(index='foo',  
          columns='bar',  
          values='baz')
```

bar	A	B	C
foo			
one	1	2	3
two	4	5	6

The pivot operation involves reshaping data by converting rows into columns.

Melt

df3

	first	last	height	weight
0	John	Doe	5.5	130
1	Mary	Bo	6.0	150

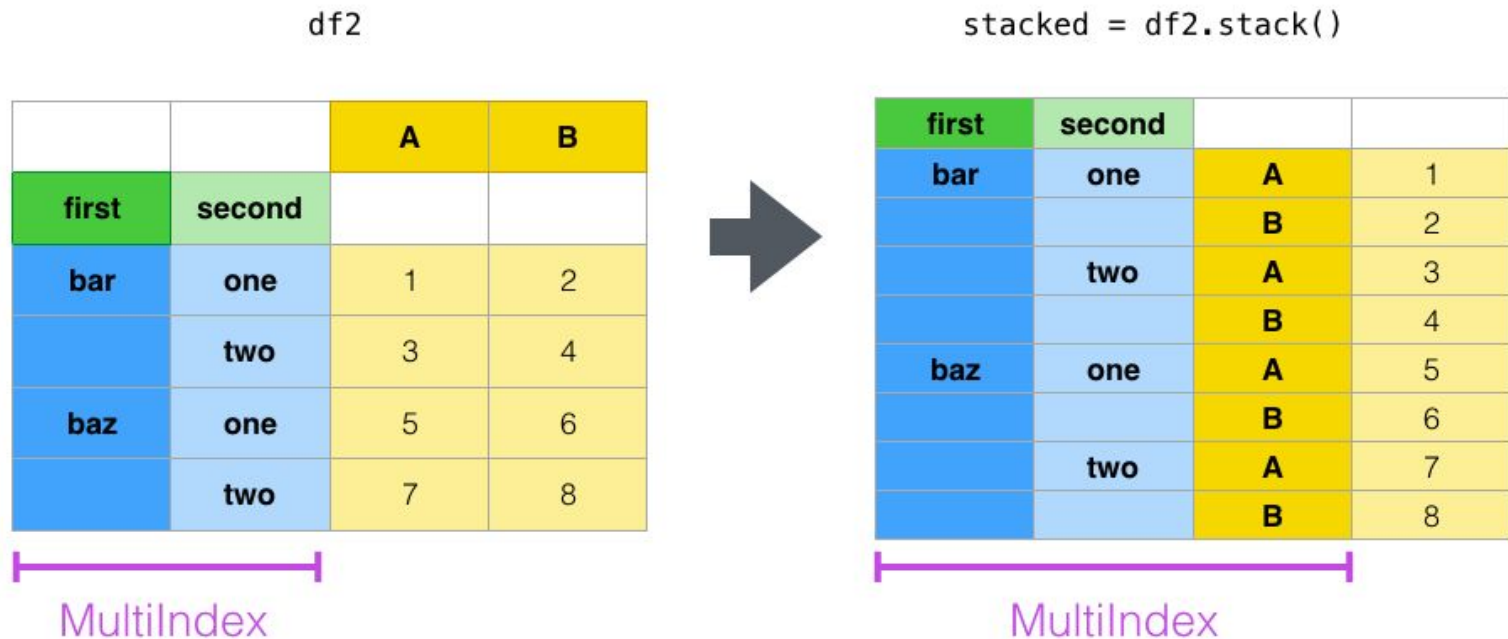


df3.melt(id_vars=['first', 'last'])

	first	last	variable	value
0	John	Doe	height	5.5
1	Mary	Bo	height	6.0
2	John	Doe	weight	130
3	Mary	Bo	weight	150

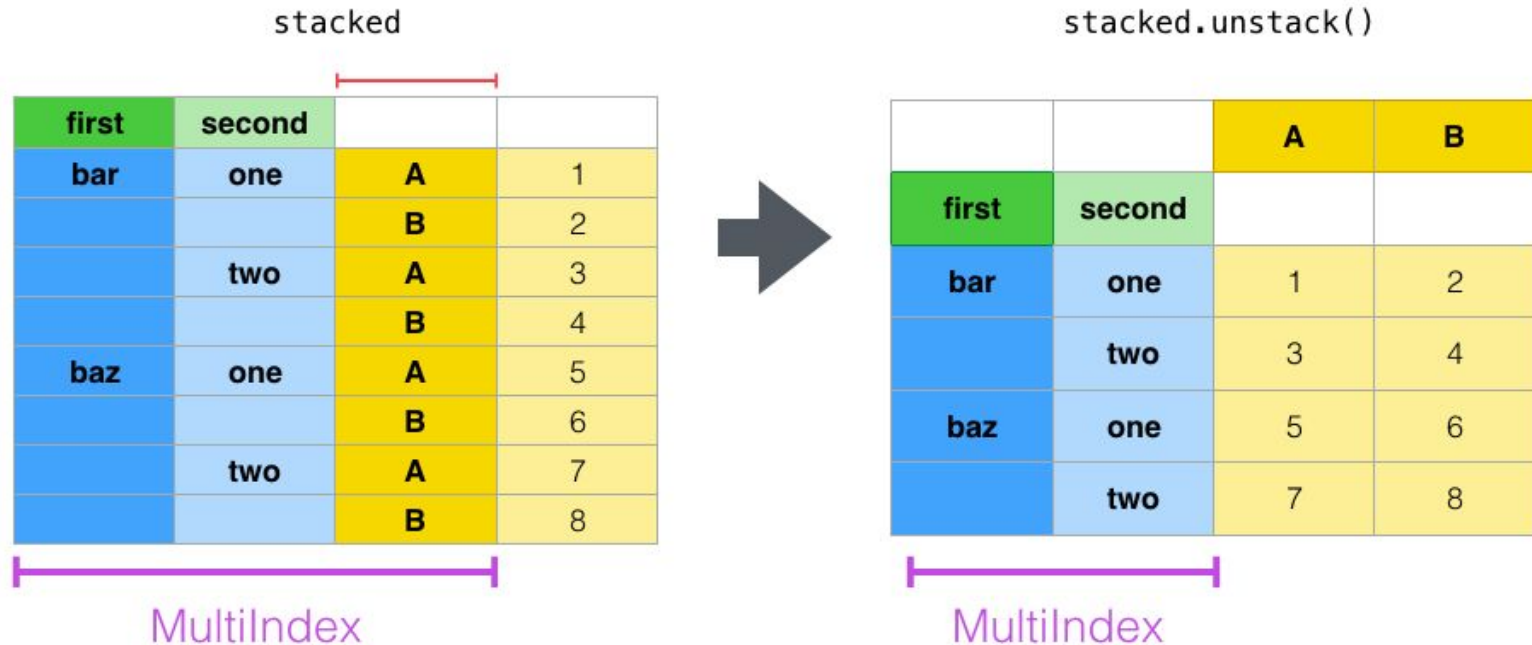
Melt is used to convert data from a wide format to a long format, typically for better analysis and visualization.

Stack



Stack is used to convert data from a wide format to a long format, similar to melt.

Unstack

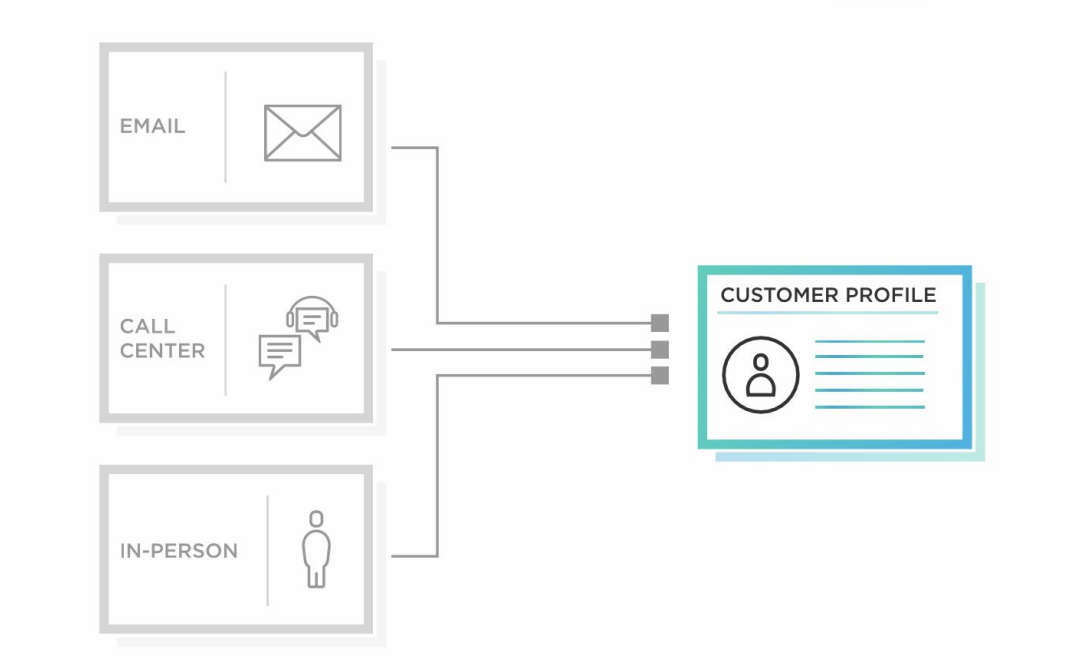


Unstack, on the other hand, performs the reverse operation, converting data from long to wide format.

Data Transformation: Combining

Combining Data

In data analytics, **combining data from different sources** is a common and essential task.



Vertical Join

Id	Product	Price
1	Ford Focus	5000€
2	Mercedes	6000€



id	Product	Price	Discount
3	Ford Focus 2	7000€	500€
4	BMW	8000€	300€

id	Product	Price	Discount
1	Ford Focus	5000€	
2	Mercedes	6000€	
3	Ford Focus 2	7000€	500€
4	BMW	8000€	300€



Vertical Join

- In this type of join, the datasets are stacked vertically to create a larger dataset.
- They **usually have the same columns (not necessary)**. If they do, they need to have the same data type.
- Concatenation is typically done when datasets represent the **same entities or observations over different periods or categories**.

In Pandas: *“concat”*, *“append”*, ...

Horizontal Join (Merge)

Id	Product	Price
1	Ford Focus	5000€
2	Mercedes Class A	6000€



id	Product	Country
1	Ford Focus	USA
2	Mercedes Class A	Germany

id	Product	Price	Country
1	Ford Focus	5000€	USA
2	Mercedes Class A	6000€	Germany

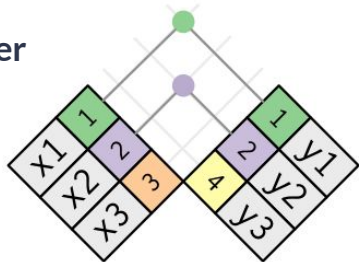


Horizontal Join (Merge)

- Performed when datasets have the **same rows but different columns**.
- Merging or joining is commonly done when datasets contain **complementary information about the same entities** or observations.

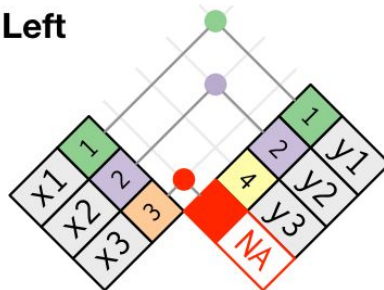
Types of Joins

Inner



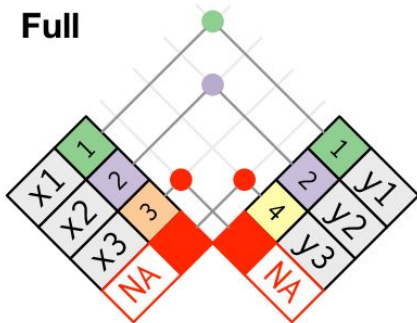
key	val_x	val_y
1	x1	y1
2	x2	y2

Left



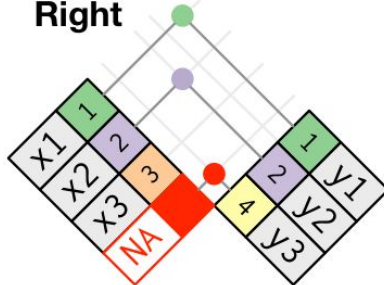
key	val_x	val_y
1	x1	y1
2	x2	y2
3	x3	NA

Full



key	val_x	val_y
1	x1	y1
2	x2	y2
3	x3	NA
4	NA	y3

Right



key	val_x	val_y
1	x1	y1
2	x2	y2
4	NA	y3

Data Organization: Grouping and Aggregating



Grouping and Aggregation

- Grouping data is the first step in data aggregation, where we **create groups** based on specific criteria. For example, we can group data by categories, time periods, or any other relevant attributes.
- Once the data is grouped, we can apply **aggregation functions to compute summary statistics or perform operations within each group**. Common aggregation functions include sum, mean, max, min, count, median, most frequent value etc.

Example


title	genre	price
book 1	adventure	11.90
book 2	fantasy	8.49
book 3	romance	9.99
book 4	adventure	9.99
book 5	fantasy	7.99
book 6	romance	5.88

genre	avg_price
adventure	$(11.90 + 9.99)/2$ 10.945
fantasy	$(8.49 + 7.99)/2$ 8.24
romance	$(9.99 + 5.88)/2$ 7.935

- Grouping by genre
- Aggregation function: calculating the average

Example

Team	Goals
F	3
F	4
F	5
A	6
A	2
A	8
A	10



F	3
A	2

Min Value in each Group

- Grouping by Team
- Aggregation function: minimum value

Example

Team	Score
A	15
A	18
B	11
B	17
B	10
C	13



Team	Row Count
A	2
B	3
C	1

- Grouping by Team
- Aggregation function: counting cases (rows)