Statistics and Data Analysis
Unit 01 – Lecture 01 Notes
Data Types, Sources, and Cleaning Basics

Tofik Ali

February 17, 2026

Why This Lecture Exists

Before we compute statistics or build models, we must ensure the data is:

- correctly **typed** (numbers are numbers, dates are dates),
- correctly **formatted** (consistent schema and representation),
- and reasonably **clean** (no obvious errors, duplicates, or impossible values).

Otherwise, we can get very convincing but completely wrong conclusions.

1. Dataset Basics

1.1 Observation vs variable

- An **observation** is one record/row (e.g., one student).
- A **variable** (or feature/attribute) is one column (e.g., attendance%).
- A **dataset** is a table of observations and variables.

1.2 Why type matters

If a numeric column is stored as text, then:

- sorting can become wrong ("100" comes before "20" in string order),
- mean/median cannot be computed correctly,
- plots may fail or mislead.

So the first step in almost every analysis is: **inspect data types**.

2. Data Types and Formats

2.1 Common data types (practical)

- **Numeric**: integers (count) and real values (measurement).
- **Categorical**:
  - **Nominal**: no natural order (branch = CSE/ECE).
  - **Ordinal**: ordered categories (rating = low/medium/high).
- **Binary**: yes/no, 0/1, pass/fail.
- **Datetime**: dates and timestamps.
- **Text**: comments, feedback (often unstructured).

2.2 Data formats

- **Structured**: fixed schema, tabular (CSV, SQL tables).
- **Semi-structured**: key-value or tagged (JSON, XML).
- **Unstructured**: free form (text documents, images, audio).

Why formats matter. Structured data is easiest to analyze directly. Semi-structured data needs parsing and may have missing keys. Unstructured data typically needs **feature extraction** (e.g., word counts from text, embeddings, image features).

Exercise 1 (solution)

Classify:

- Age: numeric (integer)
- Program/Branch: categorical (nominal)
- Attendance (%): numeric (real)
- Join date: datetime
- Feedback comment: text

3. Data Sources and Acquisition

3.1 Common sources

- **Surveys/forms**: can have missing fields and user entry errors.
- **Databases**: usually structured but can include stale/inconsistent codes.
- **Logs**: large volume, semi/unstructured, need parsing.
- **Sensors**: frequent readings, can have noise and missing intervals.
- **APIs**: provide JSON/XML, rate limits, schema changes.

3.2 Acquisition methods

- file import (CSV/Excel)
- database query (SQL)
- API requests (JSON)
- manual entry (small datasets only; double-check)

Exercise 2 (solution)

- Daily attendance: database export (or CSV export)
- Platform clicks: logs
- Feedback comments: survey + text field (unstructured text)
- Weather readings: sensors or API

4. Data Cleaning Basics

4.1 What is "dirty" data?

Dirty data commonly includes:

- Missing values (blank, NaN, NULL)
- Duplicate records
- Inconsistent categories (`cse`, `CSE`, `CSE`)
- Out-of-range values (attendance 105%, CGPA 12)
- Wrong type ("nine" in a numeric column)

4.2 Missing values

Missing values occur for many reasons: non-response in surveys, sensor failure, system bugs, etc.

Basic options.

1. **Drop rows/columns**: only if missingness is small and not biased.
2. **Impute**: fill missing values using a rule.
3. **Flag**: create a new column indicating missingness.

Mean vs median imputation (why median is common). The mean is sensitive to outliers. The median is more robust. So for a numeric column like income or CGPA, median is often a safer default imputation.

Exercise 3 (solution)

If 2 values are missing out of 20:

$$\text{missing } \% = \frac{2}{20} \times 100\% = 10\%$$

A reasonable action: **median imputation** for CGPA and optionally add a flag column `cgpa_was_missing`.

4.3 Outliers

An outlier is a value that looks unusually far from the rest. Important: outliers can be **errors** or **true extremes**. So the goal is not to automatically delete outliers; the goal is to **detect and investigate**.

IQR rule (fences). Compute:

$$\text{IQR} = Q_3 - Q_1$$

Then:

$$\text{Lower fence} = Q_1 - 1.5 \times \text{IQR}, \quad \text{Upper fence} = Q_3 + 1.5 \times \text{IQR}$$

Values outside fences are flagged as possible outliers.

Exercise 4 (solution)

Attendance (%): 70, 75, 80, 85, 90, 95, 150. Median is 85.

- $Q_1 = 75$ (median of 70,75,80)
- $Q_3 = 95$ (median of 90,95,150)
- IQR $= 95 - 75 = 20$
- Fences: $75 - 30 = 45$ and $95 + 30 = 125$
- Since $150 > 125$, 150 is an outlier (by IQR rule).

4.4 Duplicates and inconsistent categories

Duplicates can happen due to repeated exports, multiple submissions, or system errors. Always check duplicates using a sensible key (e.g., `student_id`).

Inconsistent categories occur due to case and whitespace differences. Common fixes:

- strip whitespace
- convert to a standard case (e.g., uppercase)
- map synonyms (e.g., “Male” and “M” to “M”)

5. Mini Demo (Python)

Run this from the lecture folder:

```
python demo/cleaning_demo.py
```

The demo performs these steps:

- prints shape, head, and dtypes of `data/messy_students.csv`
- reports missingness and duplicates
- trims and standardizes categorical values (program, gender, city)
- converts numeric columns, parses dates
- flags out-of-range values and imputes numeric missing values using median
- removes duplicate `student_id` rows
- saves `data/students_clean.csv`
- saves plots in `images/` (missingness and outlier visualization)

References

- Wickham, H. *Tidy Data.* Journal of Statistical Software, 2014.
- McKinney, W. *Python for Data Analysis.* O'Reilly, 2022.
- Montgomery, D. C., & Runger, G. C. *Applied Statistics and Probability for Engineers.* Wiley, 7th ed., 2020.

Appendix: Slide Deck Content (Reference)

The material below is a reference copy of the slide deck content. Exercise solutions are explained in the main notes where applicable.

Title Slide

https://github.com/tali7c/Statistics-and-Data-Analysis

Quick Links

Types & Formats Sources Cleaning Demo Summary

Agenda

- Overview
- Data Types and Formats
- Data Sources and Acquisition
- Data Cleaning
- Demo
- Summary

Learning Outcomes

- Identify common data types and formats used in analytics
- List common data sources and acquisition methods
- Detect typical data quality issues (missing values, duplicates, outliers)
- Apply basic cleaning steps in Python and save a cleaned dataset

Dataset, Observation, Variable

- **Dataset:** a collection of observations (rows) and variables (columns)
- **Observation:** one record (e.g., one student)
- **Variable/Feature:** one attribute (e.g., attendance, CGPA)

Goal: convert raw data into a form suitable for analysis and modeling.

Common Data Types (Practical View)

- **Numeric:** integers (count), real values (measurements)
- **Categorical:** nominal (branch), ordinal (rating: low/med/high)
- **Binary:** yes/no, pass/fail
- **Date/Time:** join date, timestamp
- **Text:** feedback, comments

Exercise 1: Classify Variable Types

For each variable, write the type (numeric/categorical/binary/datetime/text):

1. Age
2. Program/Branch (CSE, ECE, ...)
3. Attendance (%)
4. Join date
5. Feedback comment

Solution 1

- Age: numeric (integer)
- Program/Branch: categorical (nominal)
- Attendance (%): numeric (real)
- Join date: datetime
- Feedback: text (unstructured)

Data Formats

- **Structured:** fixed schema (tables)
  Examples: CSV, SQL tables
- **Semi-structured:** flexible schema with tags/keys
  Examples: JSON, XML
- **Unstructured:** free-form content
  Examples: text documents, images, audio

Structured Example (Table)

student_id	program	attendance_pct	cgpa
1001	CSE	92	8.2
1002	CSE	85	7.5
1003	ECE	105	8.9

Note: 105% attendance is an example of an out-of-range value.

Semi-structured Example (JSON)

```
{
  "student_id": 1001,
  "program": "CSE",
  "attendance_pct": 92,
  "courses": ["Math", "DSA", "Stats"]
}
```

Keys may vary from record to record (flexible schema).

Unstructured Example (Text/Log)

```
2026-02-08 10:02:11 INFO login user=1007 device=android city=Delhi
```

Useful information exists, but it requires parsing and feature extraction.

Common Data Sources

- Surveys and forms (Google Forms, LMS exports)
- Databases (student records, attendance systems)
- Web and app logs (clickstream)
- Sensors/IoT (temperature, GPS)
- Public datasets (government portals, research repositories)

Acquisition Methods

- Files: CSV/Excel export → `read_csv`, `read_excel`
- Database query: SQL → extract tables
- API calls: JSON responses → parse and store
- Manual entry: small datasets (careful with errors)

Exercise 2: Choose a Source

For each case, suggest a likely source (survey/database/log/API):

1. Daily attendance of students
2. Online learning platform clicks
3. Student feedback comments
4. Weather readings every minute

Solution 2

- Attendance: database export (or CSV from attendance system)
- Clicks: logs (web/app logs)
- Feedback: survey + text field (unstructured text)
- Weather readings: sensors/IoT or API

Why Cleaning Matters

- Models and statistics assume data is meaningful and consistent
- "Garbage in, garbage out" → wrong conclusions
- Cleaning improves: accuracy, fairness, and reproducibility

Common Data Quality Issues

- Missing values (blank, NaN, NULL)
- Duplicates (same record repeated)
- Inconsistent categories (`cse`, `CSE`, `CSE`)
- Out-of-range values (attendance 105%, CGPA 12)
- Wrong data type ("nine" instead of 9.0)

Handling Missing Values (Basic Options)

- **Drop:** remove rows/columns (only if few missing and safe)
- **Impute:** fill with mean/median/mode (simple baseline)
- **Domain rule:** fill with a meaningful default (carefully)
- **Flag:** create an indicator feature "was_missing"

Exercise 3: Missingness Decision

In a dataset of 20 students, the column `cgpa` has 2 missing values.

- What is the missingness percentage?
- Suggest one reasonable action for this column.

Solution 3

- Missingness $= 2/20 \times 100\% = 10\%$
- Action: impute using **median** CGPA (robust) and optionally add a flag

Outliers (Basic Idea)

An outlier is a value that is unusually far from typical values.

- Outliers can be **errors** (wrong entry) or **real extremes**
- They can strongly affect mean, variance, and some models
- Use rules like IQR fences as a **screening** step

IQR Rule (Fences)

$$\text{Lower fence} = Q_1 - 1.5 \times \text{IQR}, \quad \text{Upper fence} = Q_3 + 1.5 \times \text{IQR}$$

$$\text{IQR} = Q_3 - Q_1$$

Values outside fences are *possible* outliers.

Exercise 4: IQR Outlier Check

Attendance (%): 70, 75, 80, 85, 90, 95, 150

Task: Compute Q_1, Q_3, IQR, fences, and decide if 150 is an outlier.

Solution 4

Sorted data: 70, 75, 80, 85, 90, 95, 150 (n=7). Median = 85.
Lower half: 70, 75, 80 $\Rightarrow Q_1 = 75$
Upper half: 90, 95, 150 $\Rightarrow Q_3 = 95$
IQR $= 95 - 75 = 20$
Fences: $75 - 30 = 45$ and $95 + 30 = 125$
Conclusion: $150 > 125 \Rightarrow$ outlier (by IQR rule).

Cleaning Checklist (Fast)

- Check shape, column names, and data types
- Check missingness and duplicates
- Standardize categories (trim whitespace, normalize case)
- Check ranges and impossible values
- Save a cleaned version (do not overwrite raw file)

Mini Demo (Python)

Run from the lecture folder:

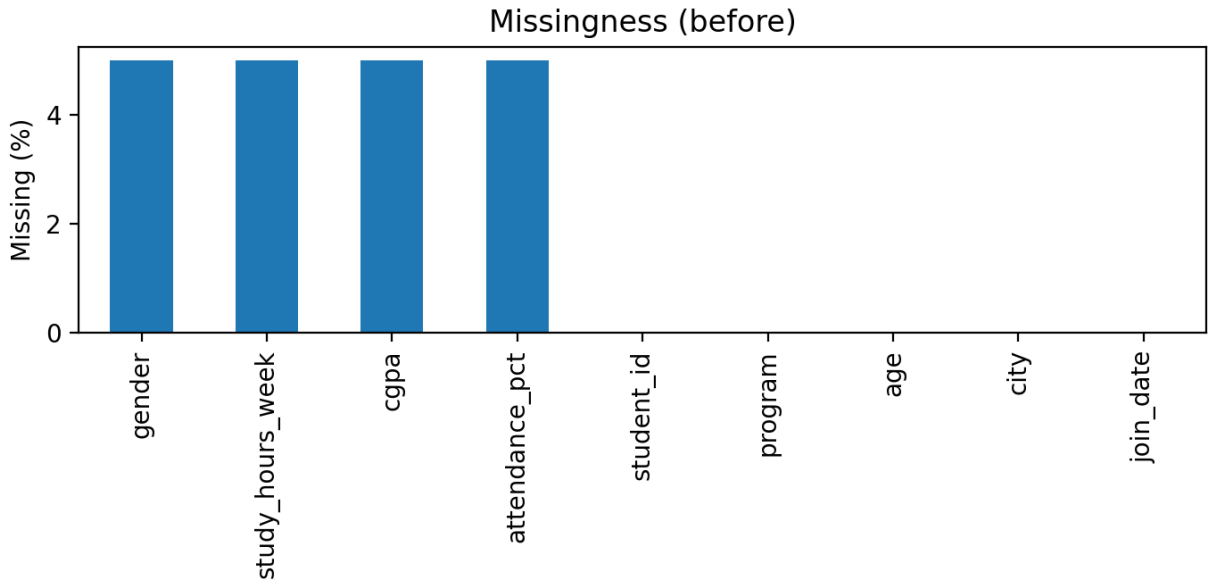
```
python demo/cleaning_demo.py
```

Outputs:

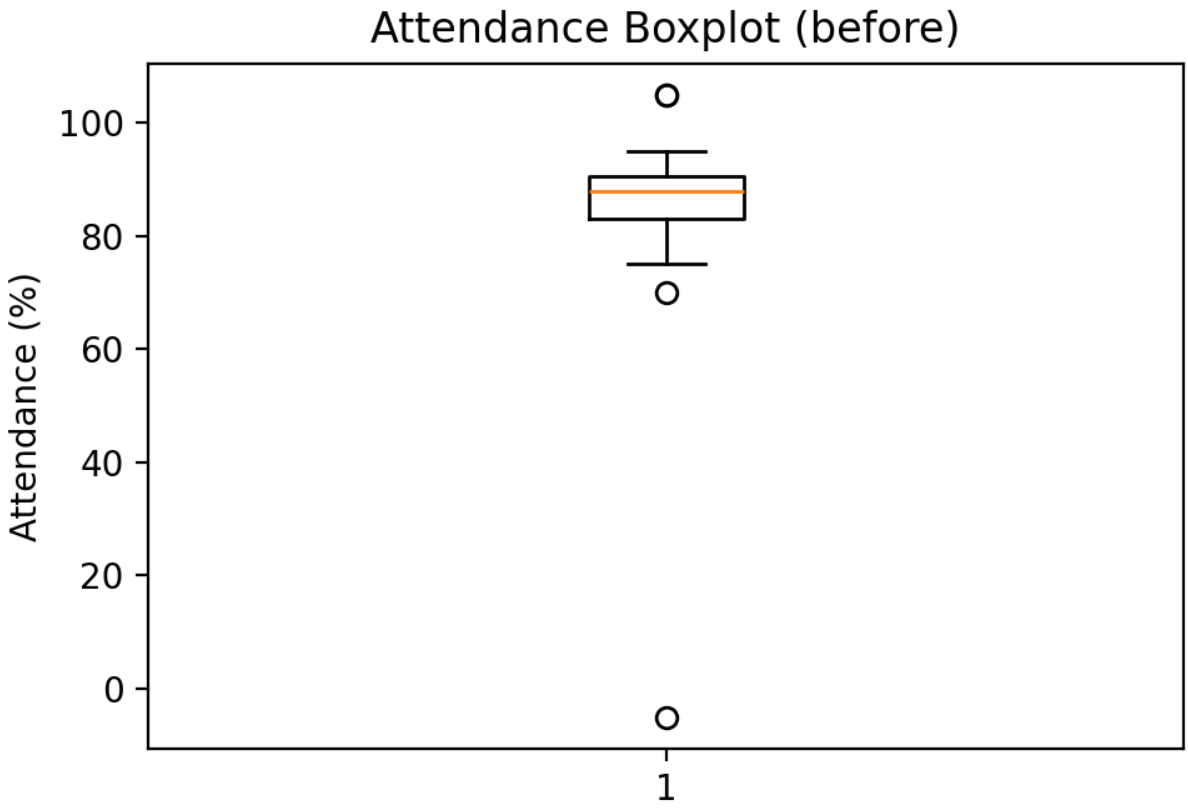
- `data/students_clean.csv`
- plots in `images/` (missingness and outlier visual)

Demo Output (Example)

Missingness



Attendance Outliers



Summary

- Data types and formats determine how we store and process data
- Different sources require different acquisition and validation steps
- Cleaning deals with missing values, duplicates, inconsistencies, and outliers
- Always save a cleaned dataset and document the rules you applied

Exit question: In one sentence, why can "attendance 105%" be dangerous in analysis?