pyampute: a Python library for data amputation

Rianne Schouten, **Davina Zamanzadeh**, Prabhant Singh

Hello! I’m...

Davina

I’m a PhD Candidate at UCLA in the Computer Science Department.

Table of Contents

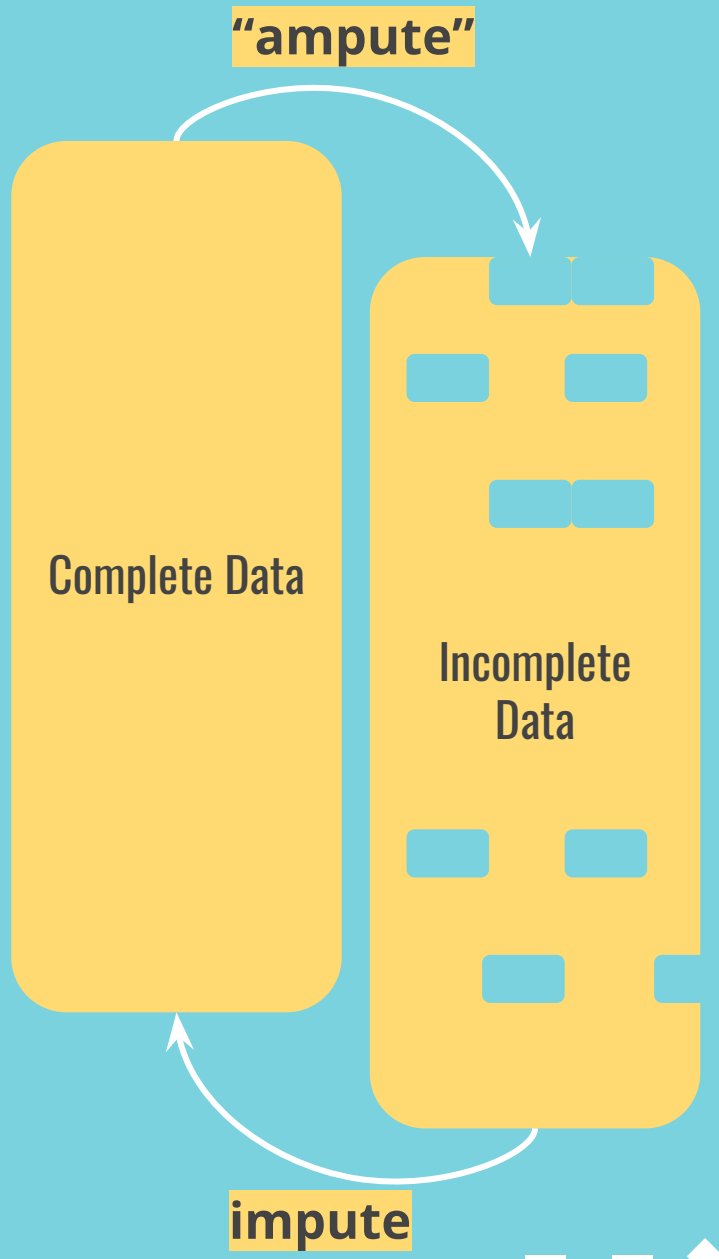
01

Motivation

Why make pyampute?

What is pyampute?

pyampute executes **multivariate amputation** (masking or removing data, therefore introducing missingness) in an already *complete* dataset.

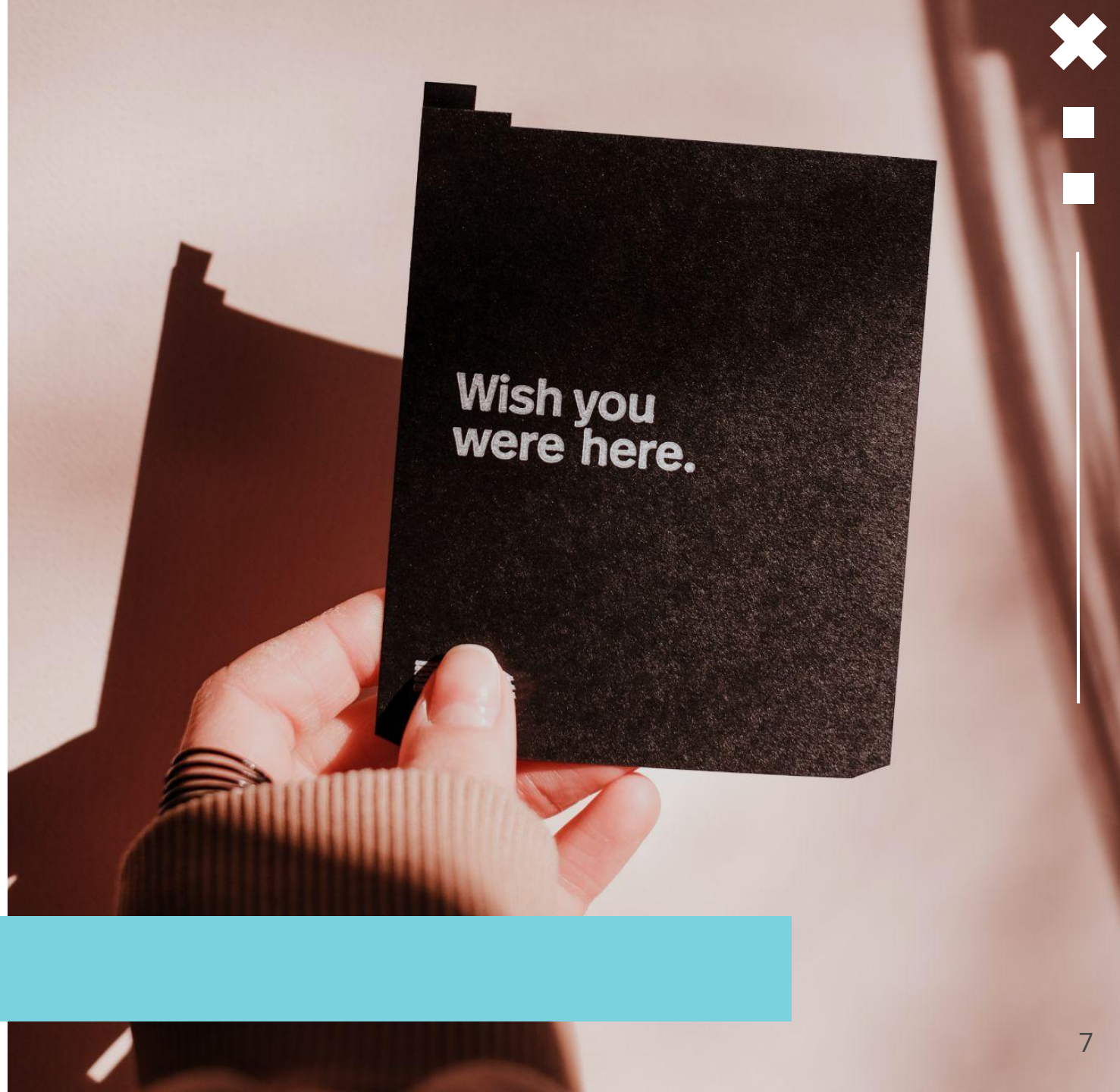


You may be asking...

What's wrong with you? Why would you get rid of perfectly good, usable, precious complete data?

Missing Data

So what's the big deal?



Missing data introduces uncertainty that affects downstream tasks.

Uncertainty

Logic and Probability says...

Uncertainty is caused by things unknown

- ➔ Not enough data / confidence (epistemic)
- ➔ Not knowable / stochastic in nature (aleatoric)

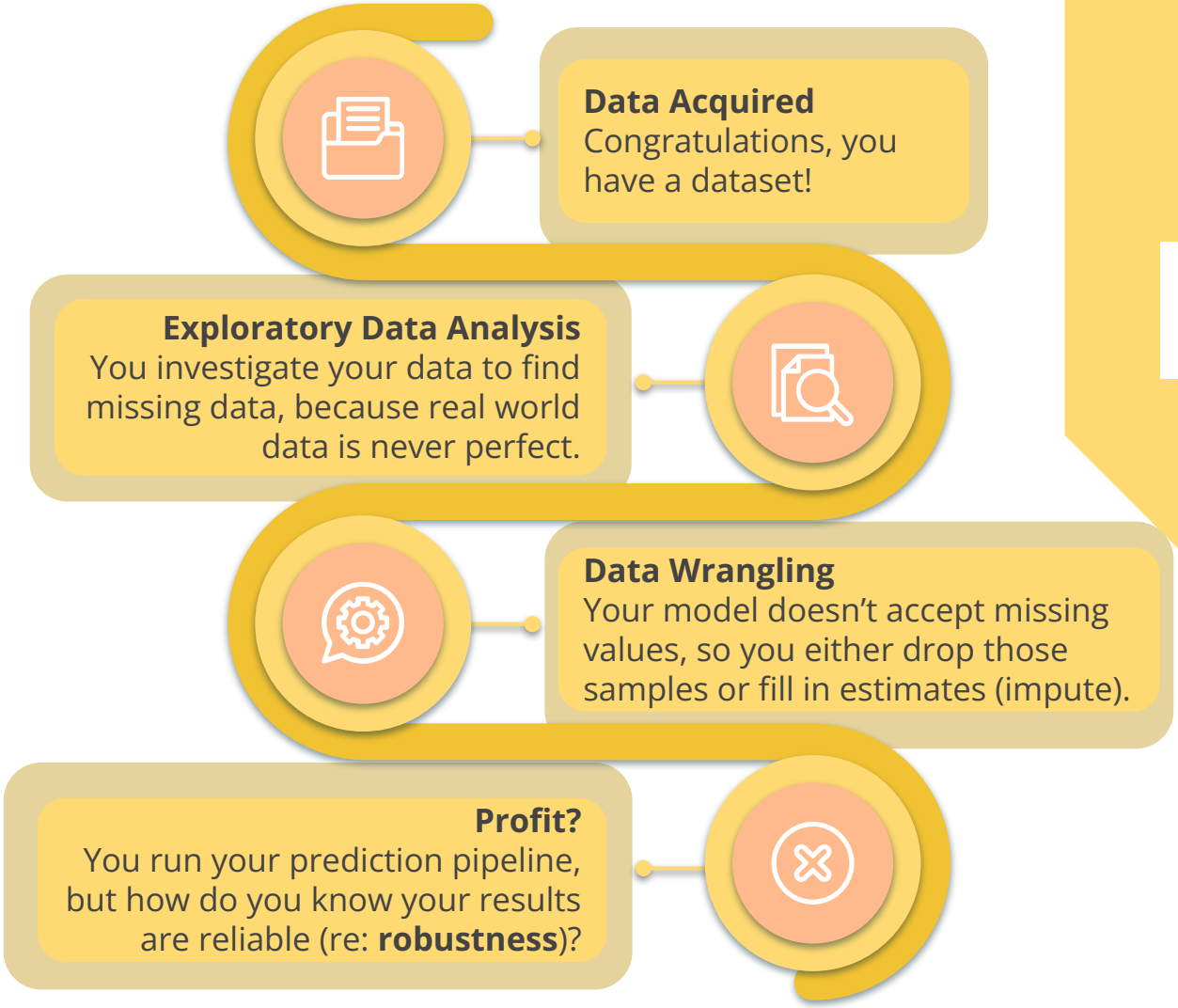
Statistics says...

Uncertainty is related to error.

- ➔ Statistical uncertainty: variation (less precise).
- ➔ Systematic uncertainty: bias (systematically inaccurate)

Missing data is everywhere...

How does it affect our analyses?

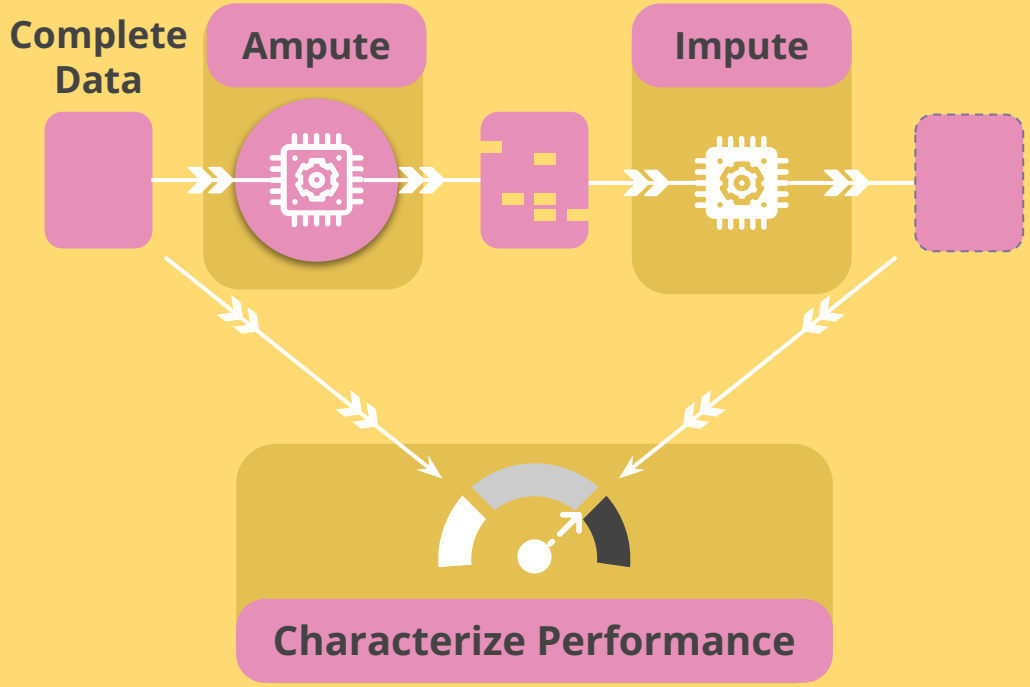


Data Wrangling

The process of cleaning/transforming data to be usable for a downstream task (e.g., imputation, error handling).

How can we understand how missingness affects our analyses?

Controlled Experiments



Imputation

The process of providing estimates for missing values.

Amputation

The process of masking or removing data (introducing missingness).

02

Background

What causes missing data?

Missingness Mechanisms

Missing Completely At Random
Missingness for a variable is unrelated to any variables (observed or not).

[e.g.] Equipment malfunctions for a day.

Missing At Random
Missingness for a variable explained by observed variables.

[e.g.] Patients under 21 in the US* are less likely to fill out alcohol usage.

Missing Not At Random
Missingness for a variable explained by the value itself, or another unobserved value/variable.

[e.g.] Equipment doesn't register values over 100, or patient refuses testing for religious reasons but religion is not recorded.

The only known tests for mechanisms can only test for MCAR.

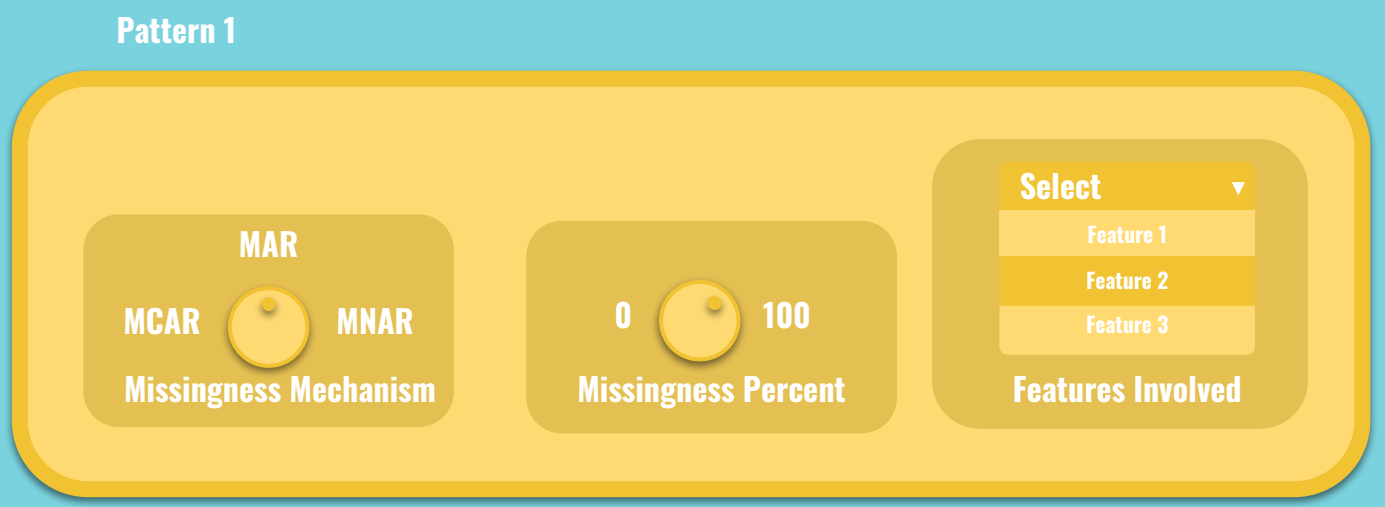
*Note: Legal age to drink in the US is 21.

03

Approach

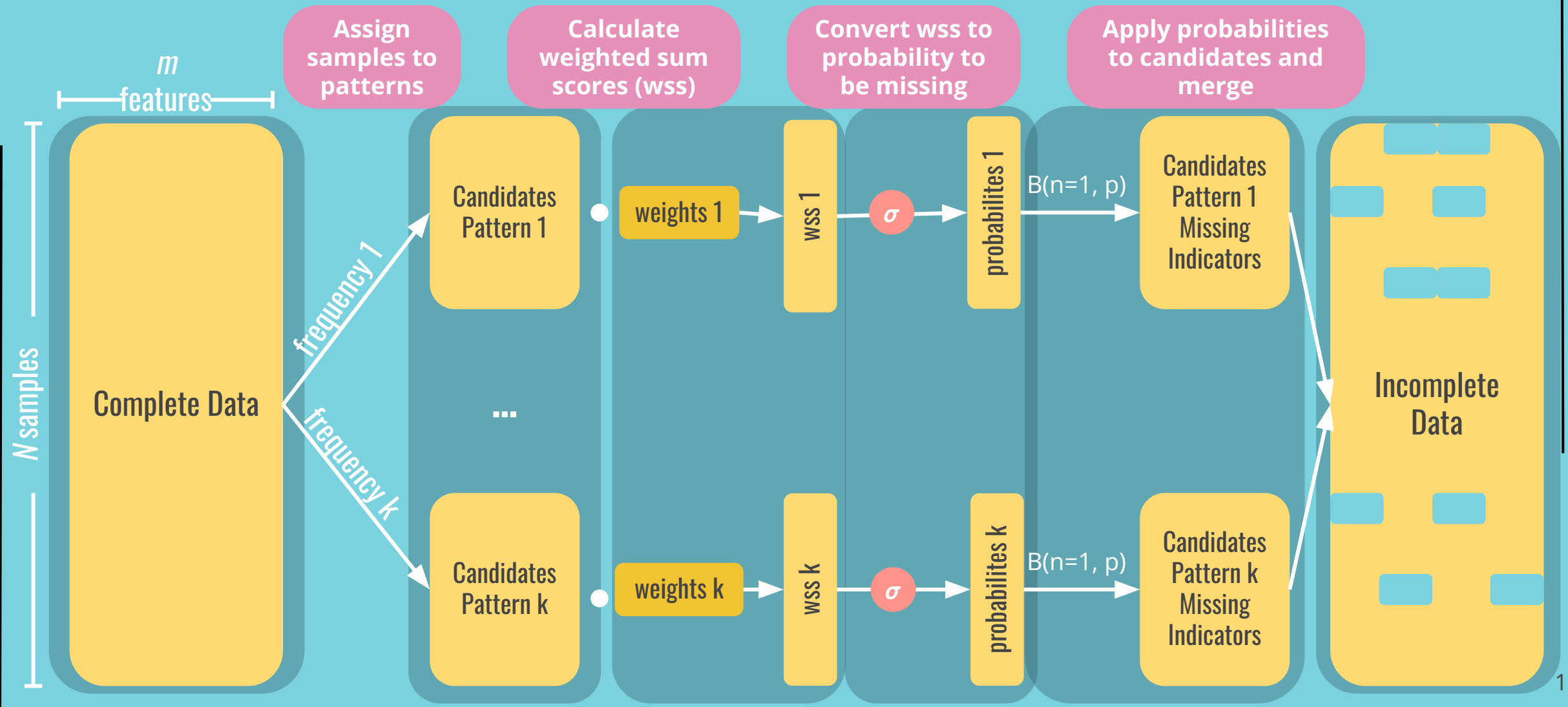
How does pyampute introduce missingness?

Characteristics of missing data



Multiple patterns of missingness within a single dataset.

Multivariate Amputation



What does this buy us?

Use Cases

Model/Pipeline Robustness

How robust is your model/data pipeline to different missingness scenarios?

Bias Analysis

How does different missingness scenarios and imputation methods affect the bias of the resulting dataset?

Downstream Performance

How does different missingness scenarios affect the performance of a downstream predictive task?

Imputation Performance

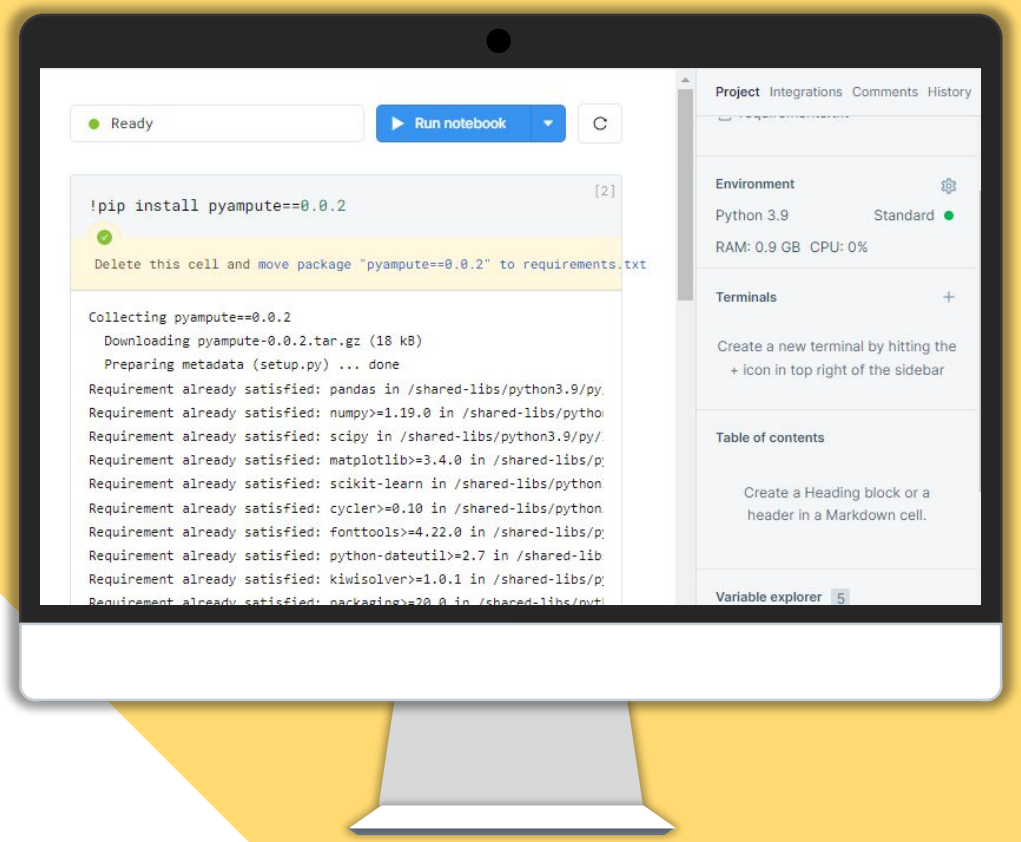
How accurate are different imputation methods on a given dataset under different missingness scenarios?

Stress Testing

How robust is your model when certain subpopulations are missing data?

Demo

Demonstrating how to use pyampute



04

Discussion and Future Work

How do we plan on expanding upon pyampute?

pyampute vs...

Previous methodologies ampute only in a univariate way.

Amputation as part of a larger multi-step pipeline.

Grid search over missingness scenarios.

Multivariate

Sklearn integration

Systematic evaluation

Future Work

When to Split?

- ➔ After amputation
  - ◆ Mimic real-world process of receiving a missing dataset in a simulated setting.
- ➔ Before amputation
  - ◆ Prevent leakage as the weighted sum scores are calculated per record.

Longitudinal Amputation

- ➔ Naive: ignore time dependency
  - ◆ Ampute each time point independently
- ➔ Introduce mechanisms (MCAR, MAR, MNAR) into time dimension
- ➔ Replace weighted sum scores with time-series model score

Contact

davina@cs.ucla.edu

davinaz.me

Thank you!

Do you have any questions?

We would love to hear any feedback you have!

Find us on github:
https://github.com/RianneSchouten/pyampute

pip package:
https://pypi.org/project/pyampute/

Documentation
https://rianneschouten.github.io/pyampute/build/html/index.html

Credits.

- ★ Schouten et. al. originally developed multivariate amputation and implemented it in the *mice* package in R as the *ampute()* function with the support of Dr. Gerko Vink and Prof. Stef van Buuren.
- ★ Multivariate amputation was initially ported over to Python by Rianne Schouten with the support of Dr. Wouter Duivesteijn and Prof. Mykola Pechenizkiy.

- Rianne M Schouten, Peter Lugtig, and Gerko Vink. Generating missing values for simulation purposes: a multivariate amputation procedure. Journal of Statistical Computation and Simulation, 88(15):2909–2930, 2018.
- Rianne M Schouten and Gerko Vink. The dance of the mechanisms: How observed information influences the validity of missingness assumptions. Sociological Methods & Research, 50:1243–1258, 2021.

- ★ Davina implemented most of pyampute's features, including all tests, and assisted with documentation. Davina is funded by the NIH grants TL1 DK132768 and U2C DK129496.

- ★ Prabhant contributed by testing the functionality and assisting with continuous integration tests, documentation, package licensing, and other package logistics.