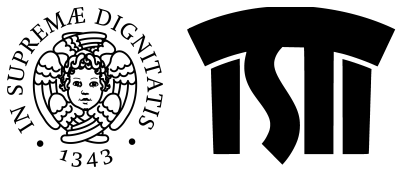
PYRREGULAR: A Unified Framework for Irregular Time Series, with Classification Benchmarks

*Francesco Spinnato**, Cristiano Landi

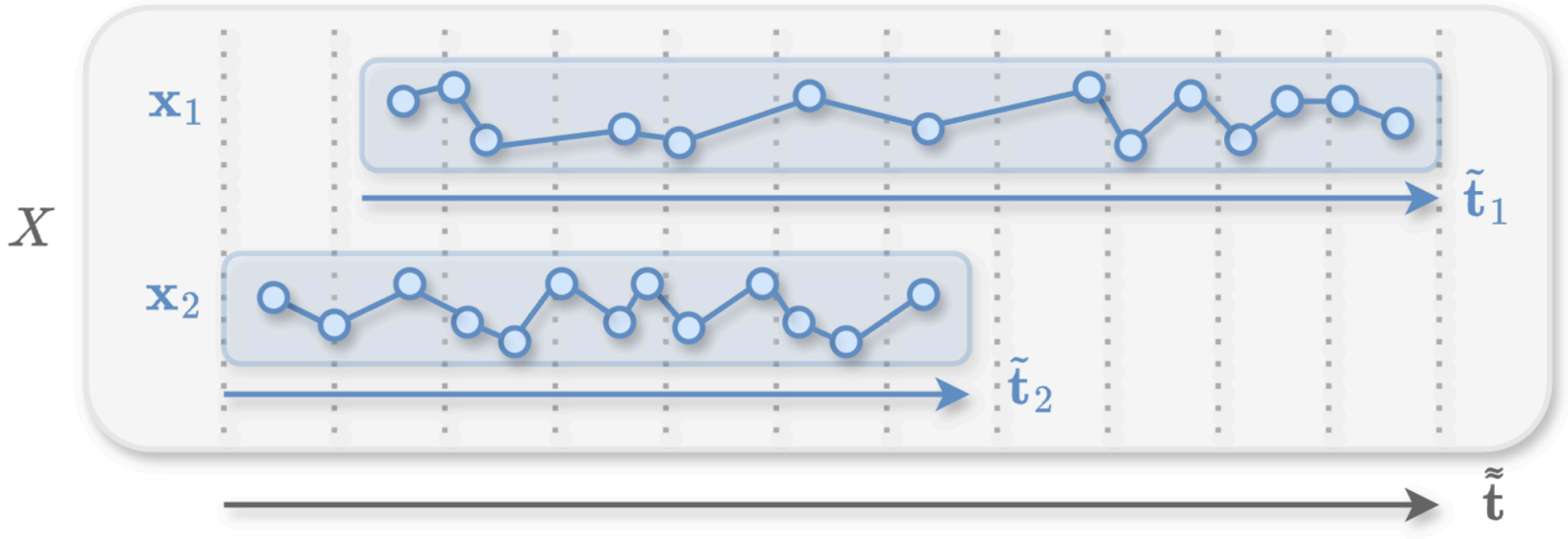
ICLR-2026

* francesco.spinnato@unipi.it
University of Pisa



Introduction

An **irregular time series** is a collection of signals (or channels), i.e., timestamped sequences of observations.



There can be problems both in the **observations** and the **timestamps**.

Introduction

Irregularity in recordings can arise because of many reasons:

- intermittent or **uneven sampling**
- gaps and **missing values**
- raggedness, jaggedness, sparsity:
  - **Unequal lengths** across signals
  - **Channel shift**
  - **Variable** recording **frequencies**

Kreindler, David M., et al. "The effects of the irregular sample and missing data in time series analysis." Nonlinear Dynamical Systems Analysis for the Behavioral Sciences Using Real Data. 2016.

Related Works

For **regular** time series classification we have:

📈 Repositories

- UEA
- UCR

💻 Libraries

- aeon
- sktime
- tslearn
- and many others...

📊 Benchmarks

- several bake-offs

- Dau, Hoang Anh, et al. "The UCR time series archive." IEEE/CAA Journal of Automatica Sinica 6.6 (2019).
- Godahewa, Rakshitha, et al. "Monash time series forecasting archive." arXiv preprint (2021).
- Middlehurst, Matthew, et al. "Bake off redux: a review and experimental evaluation of recent time series classification algorithms." DAMI (2024).

Related Works

For **irregular** time series we have:

📈 Repositories

- no standardized repository
- simulated missingness

💻 Libraries

- pypots
- diffrax
- treasure hunts on github

📊 Benchmarks

- no generalized benchmarks
- single datasets:
  - Physionet
  - PAMAP
  - MIMIC

Weerakody, Philip B., et al. "A review of irregular time series data handling with gated recurrent neural networks." Neurocomputing (2021)

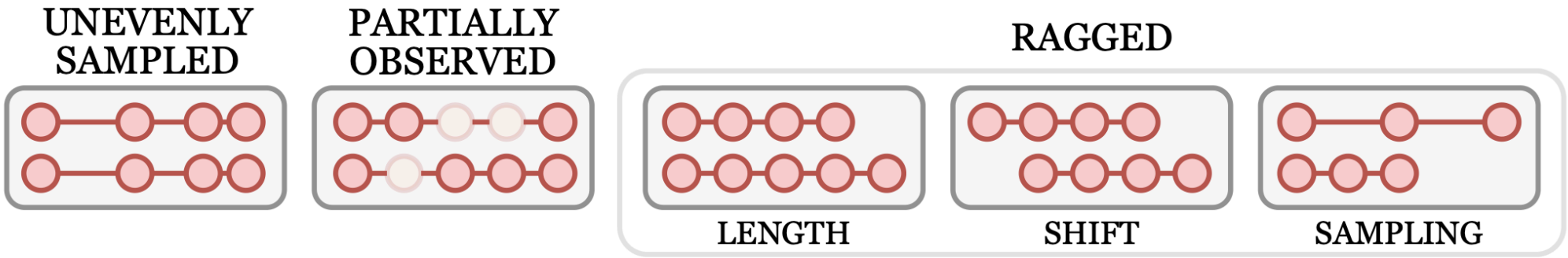
Contributions

We tackle these limitations by:

1. proposing an **array format** that can handle all possible types of irregularities;
2. collecting the first standardized irregular time series dataset **repository**;
3. providing the first bake-off style **classification benchmark** for irregular time series models.

Organizing Irregularity

We identify five **independent** irregularity types.



A Sparse Xarray

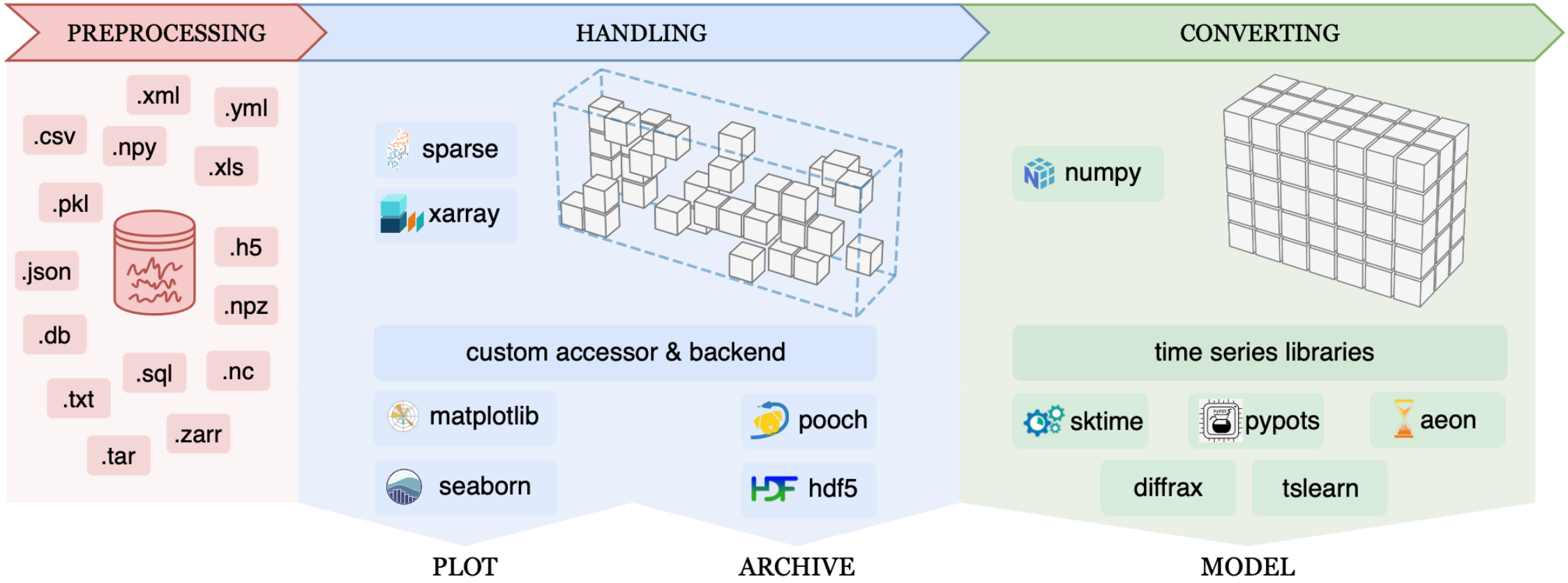
We extend `xarray` to allow underlying `sparse` COO arrays.

- `sparse` Coordinate Format arrays allow to efficiently store only the observations that were recorded.
- `xarray` wraps sparse data allowing *Pandas-like* indexing, plotting...

---

https://docs.xarray.dev/en/stable/
https://sparse.pydata.org/en/stable/

Pyrregular


PREPROCESSING
HANDLING
CONVERTING
.xml
.yml
.csv
.npy
.xls
.pkl
.h5
.json
.npz
.db
.sql
.nc
.txt
.zarr
.tar
sparse
xarray
custom accessor & backend
matplotlib
pooch
seaborn
hdf5
numpy
time series libraries
sktime
pypots
aeon
diffrax
tslearn
PLOT
ARCHIVE
MODEL

Datasets

Table 1: Datasets used for our benchmarks, divided by irregularity type: unevenly sampled (US), partially observed (PO), unequal length (UL), shift (SH), ragged sampling (RS).

	health			*human activity recognition*												*mobility*									*sensor*			*other*					*synth*
	MI3	P12	P19	CT	GM1	GM2	GM3	GP1	GP2	GX	GY	GZ	LPA	PAM	PGZ	SGZ	AN	AOC	APT	ARC	GS	MP	SE	TA	VE	DD	DG	DW	IW	JV	PGE	PL	SAD	ABF
US	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗	✓	✗	✗	✗	✓	✗	✓	✓	✓	✗	✗	✗	✗	✗	✓	✗	✗	✓
PO	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✓	✗	✗	✗	✗	✗	✗
UL	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✗	✗	✗	✓	✓	✓	✓	✓	✗
SH	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✗	✗	✓	✗	✓	✓	✗	✗	✗	✗	✗	✗	✓	✗	✗	✗
RS	✓	✓	✓	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗	✓	✗	✗	✓	✓	✗	✓	✓	✓	✗	✗	✗	✗	✗	✓	✗	✗	✗

We collected 34 irregular time series datasets from various domains.

Models

Table 2: Summary of evaluated classifiers.

Library		Model	Type	Domain
`aeon`	[73]	BORF	dictionary-based transform + LGBM classifier	regular, ragged
		RIFC	interval-based transform + LGBM classifier	partially observed
`diffrax`	[43]	NCDE	neural controlled differential equations	unevenly sampled
`pypots`	[9]	BRITS	bidirectional recurrent imputation network	partially observed
	[11]	GRU-D	gated recurrent unit with decay	partially observed
	[84]	RAINDROP	graph neural network	partially observed
	[22]	SAITS	self-attention-based imputation transformer	partially observed
	[82]	TIMESNET	temporal 2d-variation transformer.	partially observed
`sktime`	[41]	LGBM	gradient boosted tree	tabular
	[20]	ROCKET	kernel-based transform + LGBM classifier	regular
	[4]	SVM	support vector machine with distance kernel	regular, ragged
`tslearn`	[68]	KNN	distance-based with dynamic time warping	regular, ragged

And compare 12 diverse classifiers from different libraries.

Classification Benchmarks

We perform three kinds of analyses:

1. a **bake-off** style benchmark, evaluating the overall aggregated classification rankings on all datasets, with default hyperparameters;
2. an aggregated performance benchmark for **subsets** of datasets, to assess performance in datasets with specific characteristic;
3. a **fine-tuning** benchmark on the most commonly used irregular healtcare datasets (Physionet 2012 and 2019).

Classification Bake-Off

Overall, regular time series classifiers seem to perform better...

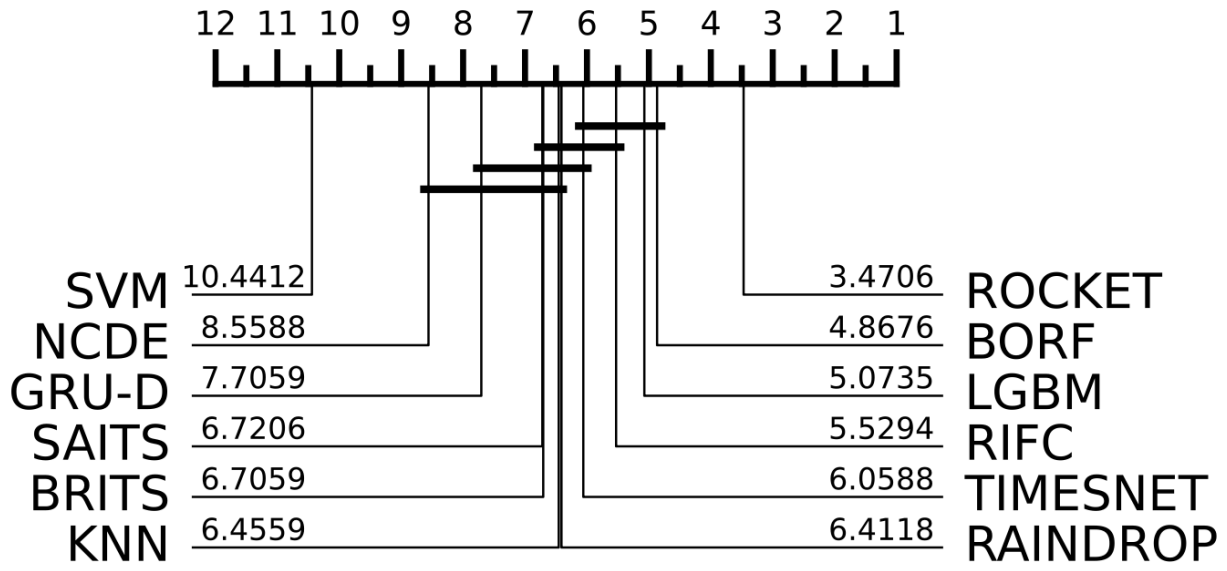
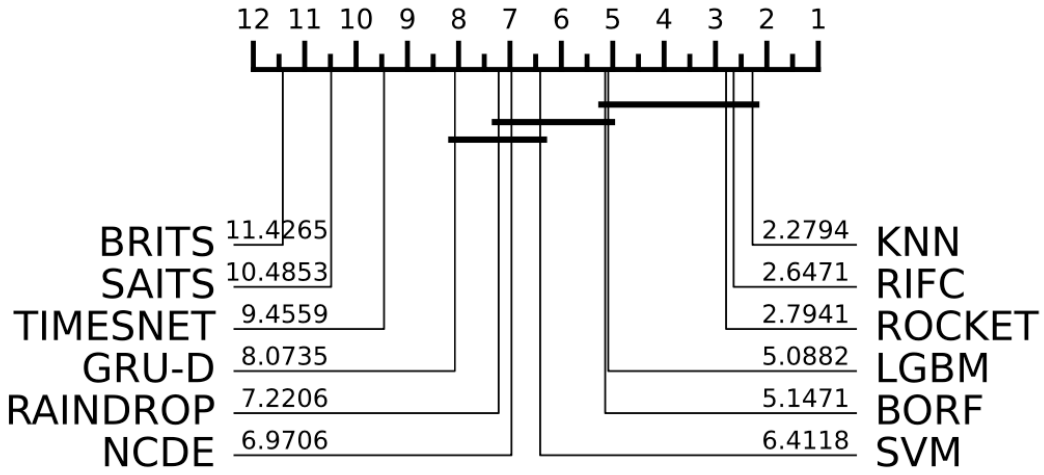


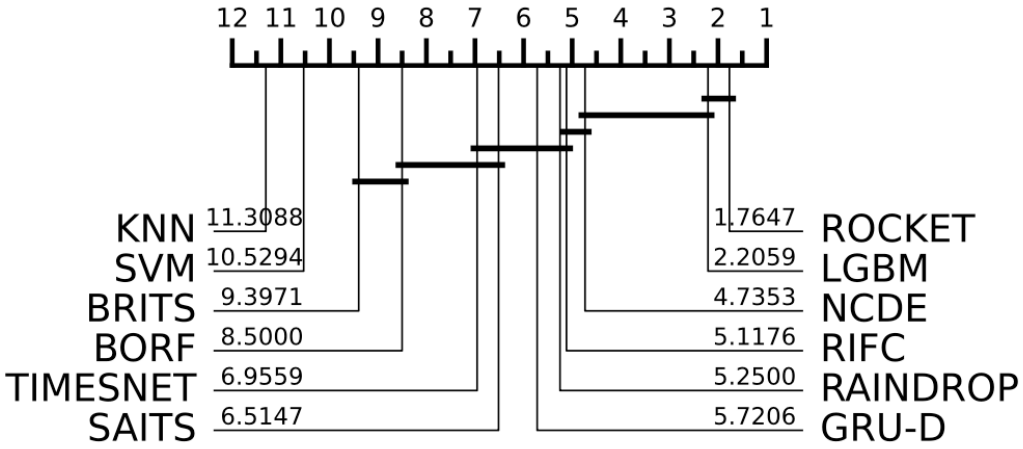
Figure 5: CD plot for the benchmarked models in terms of F1. Best models to the right. Connected models are statistically tied.

Classification Bake-Off

... with a very fast training time.



(g) Train Runtime.



(h) Inference Runtime.

Performance vs Irregularity

Regular classifiers do not work well with **partially observed** data.

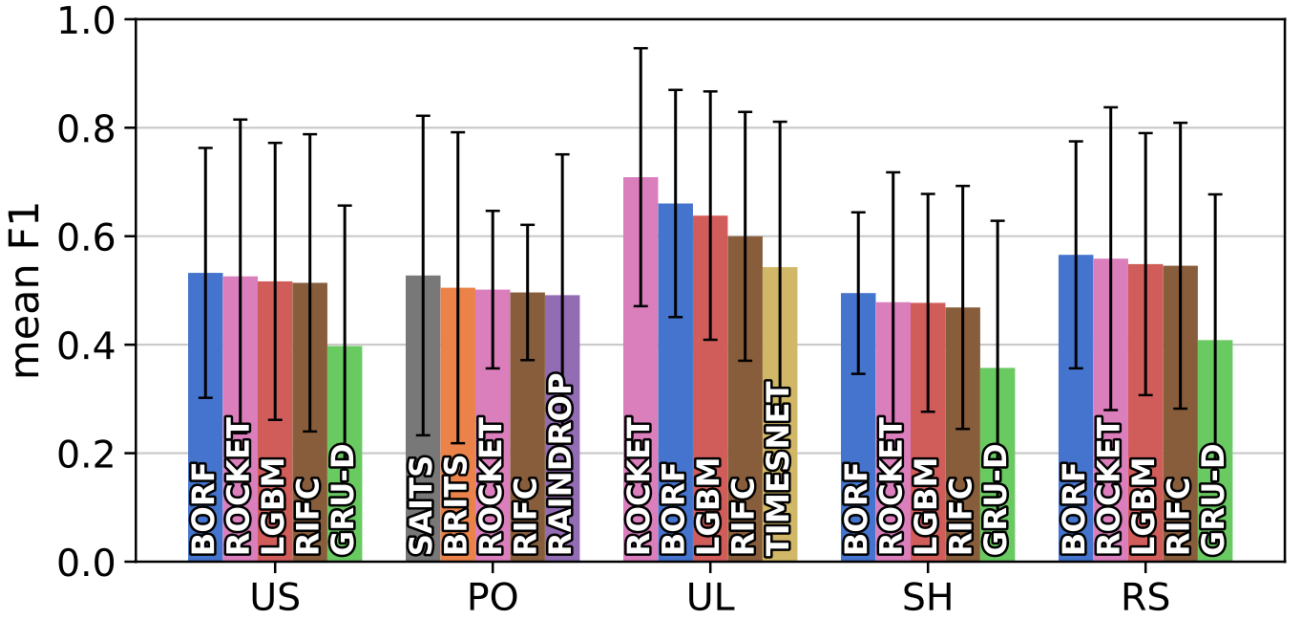


Figure 8: Mean F1 (higher is better) of the 5 best-performing models for each type of irregularity.

Performance after Fine-tuning

We **finetune** the top-3 models, and compare them with reference results on Physionet 2012 and Physionet 2019.

Table 3: Comparison of best-performing models from the bake-off, against baseline reference results (higher is better). Best values in bold, second best underlined.

		BORF	CONTI FORMER	GRU-D	LGBM	MTS FORMER	MUSIC NET	RAIN DROP	ROCKET
P12	*auc*	74.9±0.0	81.2±0.8	81.9±2.1	78.4±0.0	84.9±1.4	**86.1**±0.4	82.8±1.7	53.4±0.0
	aupr	33.4±0.0	43.9±3.0	46.1±4.7	38.1±0.0	51.1±3.7	**54.1**±2.2	44.0±3.0	15.8±0.0
P19	*auc*	80.1±0.0	79.2±2.3	83.9±1.7	85.2±0.0	**88.8**±1.5	86.8±1.4	87.0±2.3	77.3±0.0
	aupr	38.1±0.0	35.8±2.3	46.9±2.1	44.1±0.0	**57.7**±4.4	45.4±2.7	51.8±5.5	35.2±0.0

Discussion

There seem to be two kinds of models:

- **generalist**: ROCKET, BORF, LGBM $\rightarrow$ robust across a wide range of tasks, low margin for fine-tuning
- **specialist**: SAITS, RAINDROP, TIMESNET $\rightarrow$ not very robust with default hyperparameters, high potential when fine-tuned

A good **generalist** model tailored to irregular time series is missing!

Limitations and Future Works

More models: add deeper architectures, improve ease of use

More datasets: expand diversity and scale

More tasks: time series regression, forecasting, anomaly detection

THANK YOU FOR THE ATTENTION!

francesco.spinnato@di.unipi.it

https://github.com/fspinna/pyrregular

SOBIGDATA RESEARCH INFRASTRUCTURE

Future Artificial Intelligence Research

MI MO SA