# Tabular Models for Chaotic Time Series Prediction

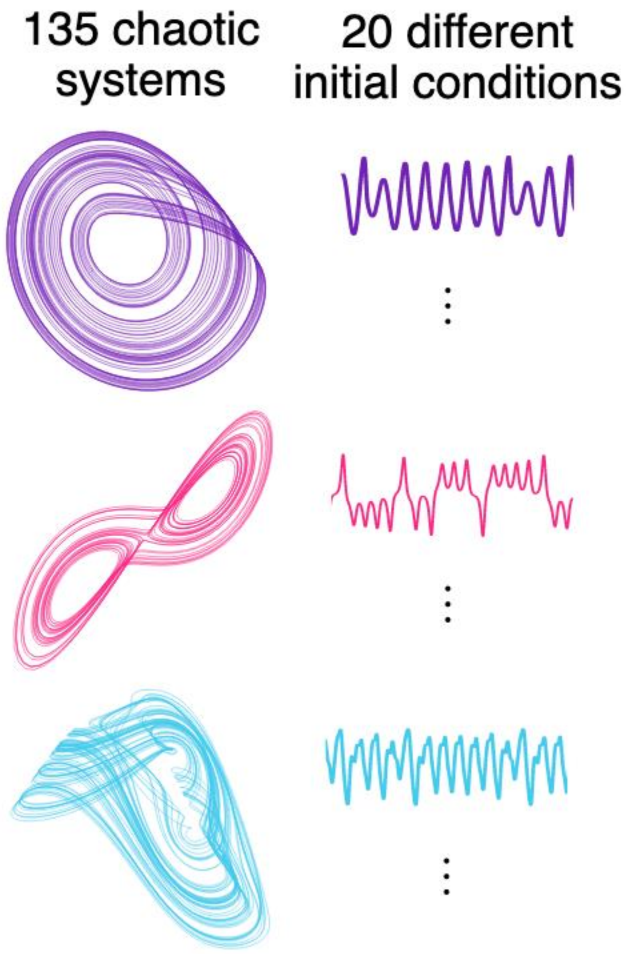
Chiara Roverato

McGill – Electrical and Computer Engineering

Supervisors: Mark Coates & Pablo Piantanida

# Chaotic Time Series

- from a **deterministic** system,
- appears **random**,
- **Never repeats**,
- **highly sensitive** to initial conditions,
- but still has **underlying structure**.




Dysts dataset [1]

[1] William Gilpin. Chaos as an interpretable benchmark for forecasting and data-driven modelling. NeurIPS, 34, 2021

# Problem Statement

**Objective:** Evaluate the zero-shot capabilities of *time series* foundational models

**Motivation:** No need to retrain for specialized tasks

**Dataset:** Chaotic Time Series

→Independent test set

→Invariant statistical and geometric properties (fractal dimensions, Lyapunov exponents, power spectra)

→ Out-of-distribution generalization ability

# Which model to choose ?

- Current state of the art methods include Chronos [2], Panda [3], ChaosNexus [4] : Time series transformers
- According to Gilpin [5]: Large, domain-agnostic time series models forecast diverse chaotic systems better

→ We choose TabPFN for its strong zero-shot and generalization capabilities

[2] Fatir Ansari et al, . Chronos: Learning the Language of Time Series, 2024.
[3] Jeffrey Lai, Anthony Bao, and William Gilpin. Panda: A pretrained forecast model for universal representation of chaotic dynamics, 2025.
[4] Chang Liu, Bohao Zhao, Jingtao Ding, and Yong Li. ChaosNexus: A Foundation Model for Universal Chaotic System Forecasting with Multi-scale Representations, 2025.
[5] William Gilpin. Model scale versus domain knowledge in statistical forecasting of chaotic systems, 2023.

# TabPFN for Chaotic Time Series

| Time | Target Dim 0 | Target Dim 1 | Target Dim 2 |
|---|---|---|---|
| 0 | $t_{0,0}$ | $t_{0,1}$ | $t_{0,2}$ |
| 1 | $t_{1,0}$ | $t_{1,1}$ | $t_{1,2}$ |
| … | … | … | … |
| T | $t_{T,0}$ | $t_{T,1}$ | $t_{T,2}$ |

**Dataset**

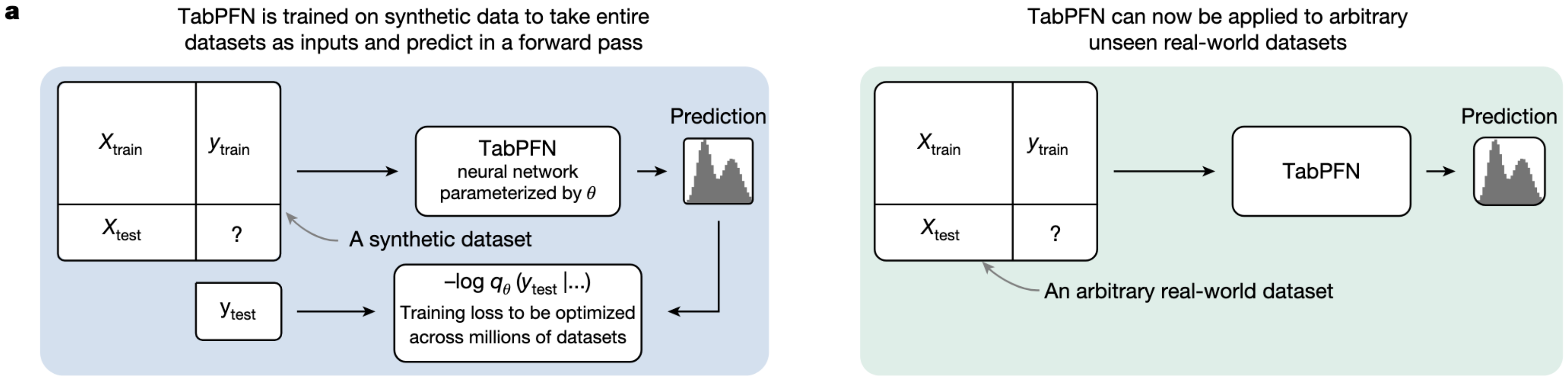
| Time | Dim |
|---|---|
| 0 | Dim 0 |
| 0 | Dim 1 |
| 0 | Dim 2 |
| 1 | Dim 0 |
| 1 | Dim 1 |
| 1 | Dim 2 |
| … | … |
| T | Dim 0 |
| T | Dim 1 |
| T | Dim 2 |

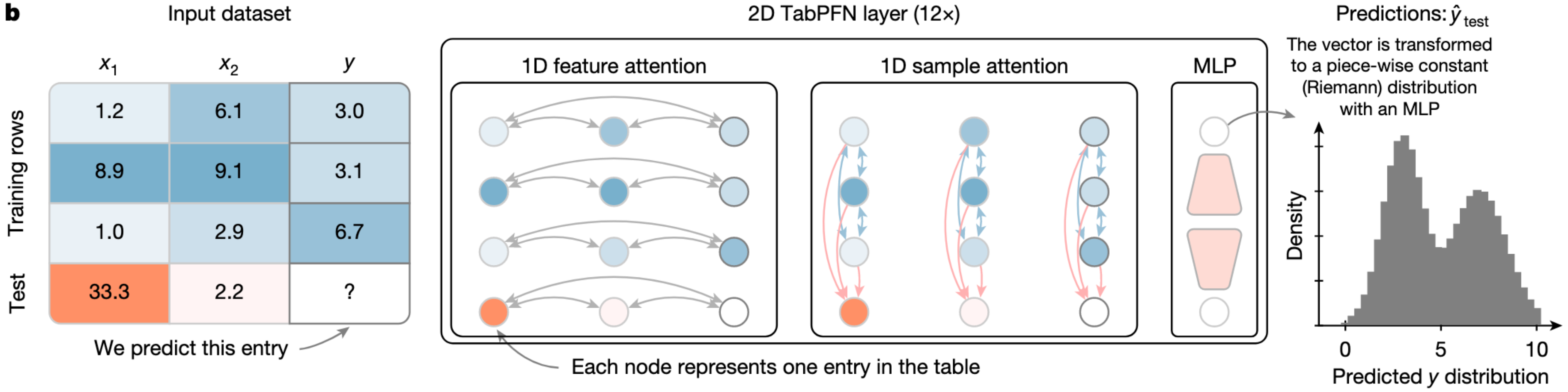
**$X_{train}$**

| Target |
|---|
| $t_{0,0}$ |
| $t_{0,1}$ |
| $t_{0,2}$ |
| $t_{1,0}$ |
| $t_{1,1}$ |
| $t_{1,2}$ |
| … |
| $t_{T,0}$ |
| $t_{T,1}$ |
| $t_{T,2}$ |

**$y_{train}$**

# TabPFN [6,7]





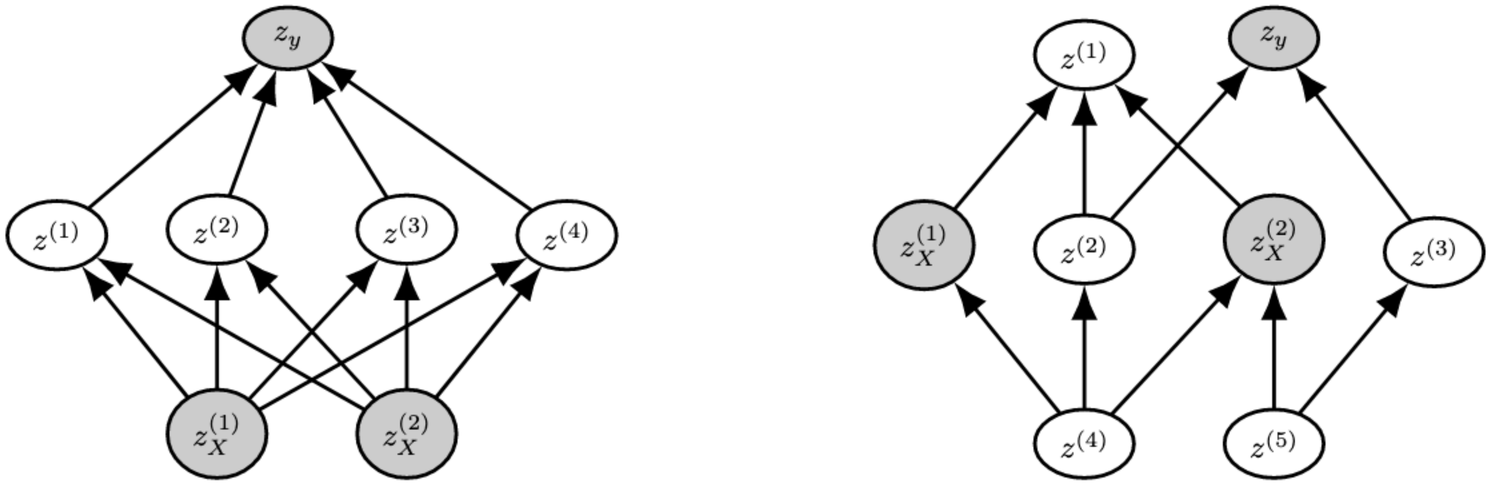
[6] N. Hollmann, S. Müller, K. Eggensperger, and F. Hutter. Tabpfn: A transformer that solves small tabular classification problems in a second. In The Eleventh International Conference on Learning Representations, 2023.

[7] N. Hollmann, S. Müller, L. Purucker, A. Krishnakumar, M. Körfer, S. B. Hoo, R. T. Schirrmeister, and F. Hutter. Accurate predictions on small data with a tabular foundation model. Nature, 637 (8045):319–326, 2025.

# PFN: Prior-Data Fitted Network [8]

With $\phi \in \Phi$ a hypothesis on a relationship between input x and its label y,
The predictive distribution of label y for input x is:

$$\boldsymbol{p(y|x, D) \propto \int_{\Phi} p(y|x, \phi)p(D|\phi)p(\phi)d\phi}.$$



(a) A BNN (b) A SCM

[8] S. Müller, N. Hollmann, S. Arango, J. Grabocka, and F. Hutter. Transformers can do bayesian inference. In Proceedings of the International Conference on Learning Representations (ICLR 22), 2022.

# PFN: Prior-Data Fitted Network [8]

How to train it ?

1. Sample hypothesis $\phi \sim p(\phi)$,
2. Sample datasets $D \sim p(\, D \mid \phi\,)$ and $p(D) = \mathbb{E}_{\phi \sim p(\phi)} p(D|\phi)$
3. Minimize cross-entropy loss

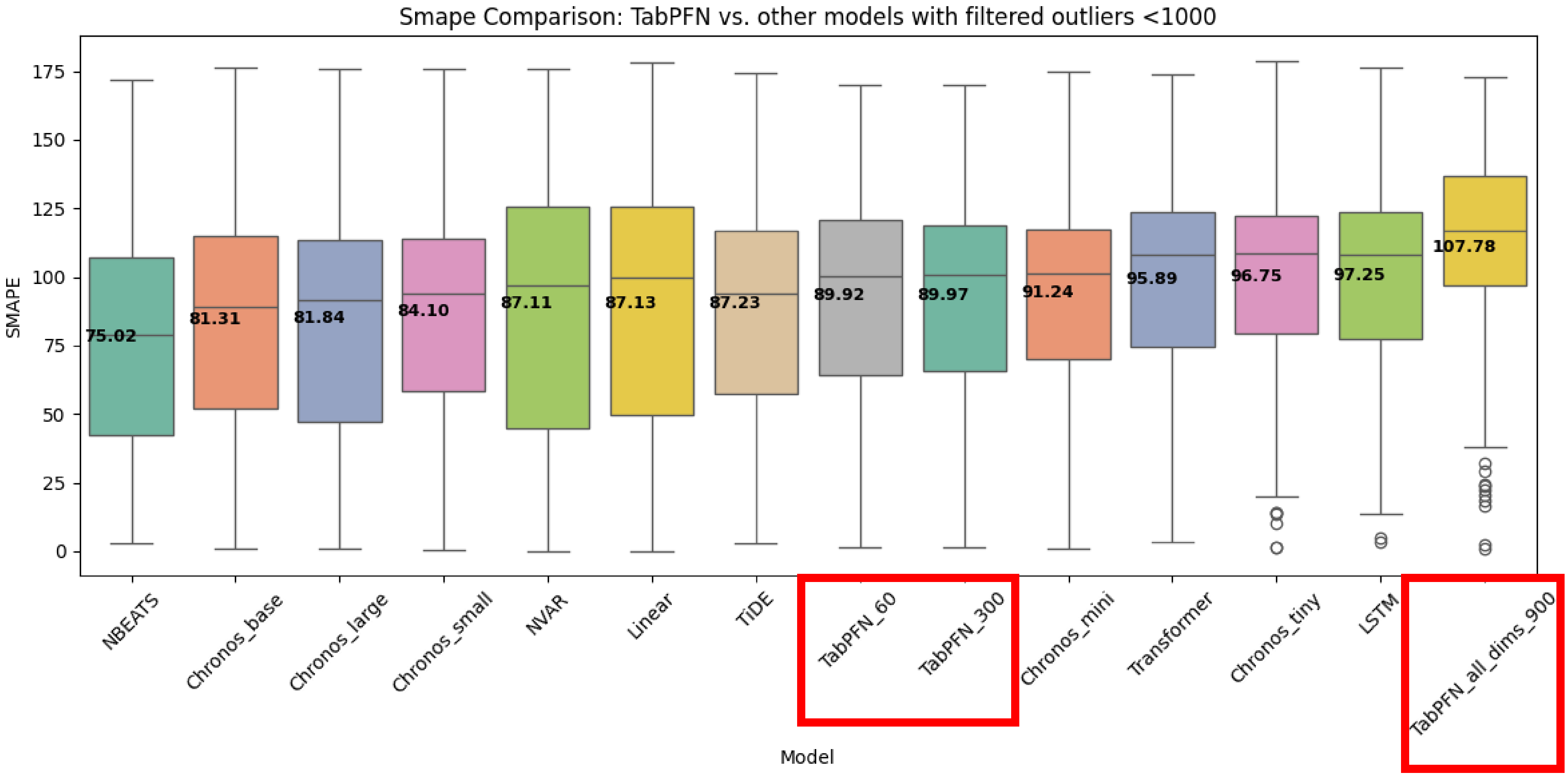
$$L_{PFN} = \mathbb{E}\ [-log\ q_\theta(y_{test}|x_{test}, D_{train})].$$

What about inference ?

One single forward pass $q_\theta(y_{test}|x_{test}, D_{real})$

[8] S. Müller, N. Hollmann, S. Arango, J. Grabocka, and F. Hutter. Transformers can do bayesian inference. In Proceedings of the International Conference on Learning Representations (ICLR 22), 2022.

# Results - TabPFN



# Disappointing but …

We can add some features !

# Disappointing but …

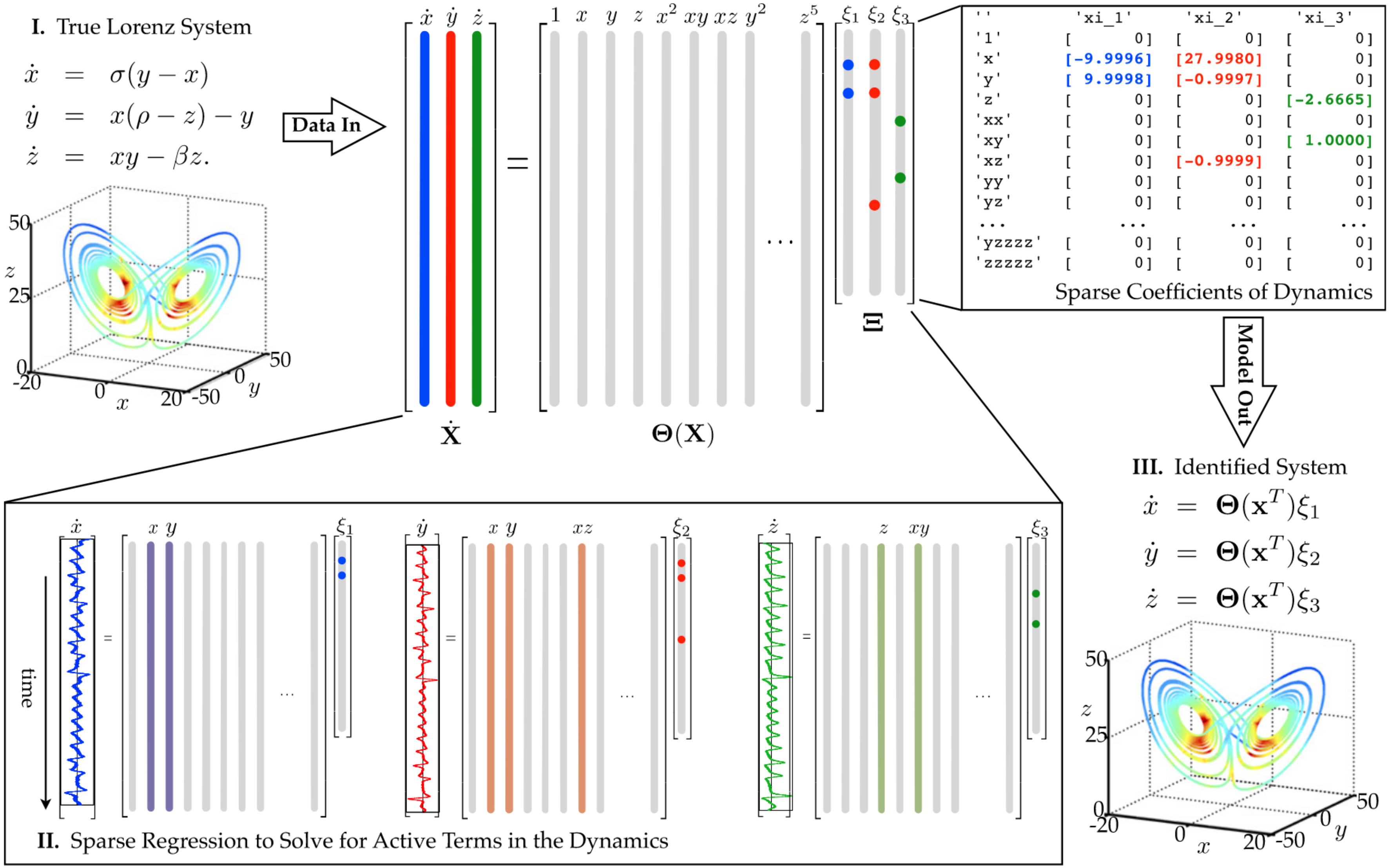
We can add some features !

Or like Florence said:

~~Cross entropy~~ + ML Researcher = ♡

Nice
features

# Sparse Identifiation of Nonlinear Dynamics [9]




| '' | 'xi_1' | 'xi_2' | 'xi_3' |
|---|---|---|---|
| '1' | [ 0] | [ 0] | [ 0] |
| 'x' | [-9.9996] | [27.9980] | [ 0] |
| 'y' | [ 9.9998] | [-0.9997] | [ 0] |
| 'z' | [ 0] | [ 0] | [-2.6665] |
| 'xx' | [ 0] | [ 0] | [ 0] |
| 'xy' | [ 0] | [ 0] | [ 1.0000] |
| 'xz' | [ 0] | [-0.9999] | [ 0] |
| 'yy' | [ 0] | [ 0] | [ 0] |
| 'yz' | [ 0] | [ 0] | [ 0] |
| ... | ... | ... | ... |
| 'yzzzz' | [ 0] | [ 0] | [ 0] |
| 'zzzzz' | [ 0] | [ 0] | [ 0] |

[9] Steven L Brunton, Joshua L Proctor, and J Nathan Kutz. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. Proc. Natl. Acad. Sci. U.S.A., 113(15): 3932–3937, 2016.

# Great features but …

TabPFN  requires the same features for $X_{train}$ and $X_{test}$

| Time | Dim | $\dot{X}$ | X² | Cos(X) | Target |
|---|---|---|---|---|---|
| 0 | Dim 0 | $\dot{x}_{0,0}$ | $x_{0,0}^2$ | $\cos x_{0,0}$ | $t_{0,0}$ |
| 0 | Dim 1 | $\dot{x}_{0,1}$ | $x_{0,1}^2$ | $\cos x_{0,1}$ | $t_{0,1}$ |
| 0 | Dim 2 | $\dot{x}_{0,2}$ | $x_{0,2}^2$ | $\cos x_{0,2}$ | $t_{0,2}$ |
| 1 | Dim 0 | $\dot{x}_{1,0}$ | $x_{1,0}^2$ | $\cos x_{1,0}$ | $t_{1,0}$ |
| 1 | Dim 1 | $\dot{x}_{1,1}$ | $x_{1,1}^2$ | $\cos x_{1,1}$ | $t_{1,1}$ |
| 1 | Dim 2 | $\dot{x}_{1,2}$ | $x_{1,2}^2$ | $\cos x_{1,2}$ | $t_{1,2}$ |
| … | … | … | … | … | … |
| T | Dim 0 | $\dot{x}_{T,0}$ | $x_{T,0}^2$ | $\cos x_{T,0}$ | $t_{T,0}$ |
| T | Dim 1 | $\dot{x}_{T,1}$ | $x_{T,1}^2$ | $\cos x_{T,1}$ | $t_{T,1}$ |
| T | Dim 2 | $\dot{x}_{T,2}$ | $x_{T,2}^2$ | $\cos x_{T,2}$ | $t_{T,2}$ |

# TabPFN for Chaotic Time Series

| Time | Dim | $\dot{X}$ | $X^2$ | Cos(X) | Target |
|---|---|---|---|---|---|
| 0 | Dim 0 | $\dot{x}_{0,0}$ | $x^2_{0,0}$ | $\cos x_{0,0}$ | $t_{0,0}$ |
| 0 | Dim 1 | $\dot{x}_{0,1}$ | $x^2_{0,1}$ | $\cos x_{0,1}$ | $t_{0,1}$ |
| 0 | Dim 2 | $\dot{x}_{0,2}$ | $x^2_{0,2}$ | $\cos x_{0,2}$ | $t_{0,2}$ |
| 1 | Dim 0 | $\dot{x}_{1,0}$ | $x^2_{1,0}$ | $\cos x_{1,0}$ | $t_{1,0}$ |
| 1 | Dim 1 | $\dot{x}_{1,1}$ | $x^2_{1,1}$ | $\cos x_{1,1}$ | $t_{1,1}$ |
| 1 | Dim 2 | $\dot{x}_{1,2}$ | $x^2_{1,2}$ | $\cos x_{1,2}$ | $t_{1,2}$ |
| … | … | … | … | … | … |
| T | Dim 0 | $\dot{x}_{T,0}$ | $x^2_{T,0}$ | $\cos x_{T,0}$ | $t_{T,0}$ |
| T | Dim 1 | $\dot{x}_{T,1}$ | $x^2_{T,1}$ | $\cos x_{T,1}$ | $t_{T,1}$ |
| T | Dim 2 | $\dot{x}_{T,2}$ | $x^2_{T,2}$ | $\cos x_{T,2}$ | $t_{T,2}$ |

Shift Up

| Time | Dim | $\dot{X}$ | $X^2$ | Cos(X) | Target |
|---|---|---|---|---|---|
| 1 | Dim 0 | $\dot{x}_{0,0}$ | $x^2_{0,0}$ | $\cos x_{0,0}$ | $t_{1,0}$ |
| 1 | Dim 1 | $\dot{x}_{0,1}$ | $x^2_{0,1}$ | $\cos x_{0,1}$ | $t_{1,1}$ |
| 1 | Dim 2 | $\dot{x}_{0,2}$ | $x^2_{0,2}$ | $\cos x_{0,2}$ | $t_{1,2}$ |
| 2 | Dim 0 | $\dot{x}_{1,0}$ | $x^2_{1,0}$ | $\cos x_{1,0}$ | $t_{2,0}$ |
| 2 | Dim 1 | $\dot{x}_{1,1}$ | $x^2_{1,1}$ | $\cos x_{1,1}$ | $t_{2,1}$ |
| 2 | Dim 2 | $\dot{x}_{1,2}$ | $x^2_{1,2}$ | $\cos x_{1,2}$ | $t_{2,2}$ |
| … | … | … | … | … | … |
| T | Dim 0 | $\dot{x}_{T-1,0}$ | $x^2_{T-1,0}$ | $\cos x_{T-1,0}$ | $t_{T,0}$ |
| T | Dim 1 | $\dot{x}_{T-1,1}$ | $x^2_{T-1,1}$ | $\cos x_{T-1,1}$ | $t_{T,1}$ |
| T | Dim 2 | $\dot{x}_{T-1,2}$ | $x^2_{T-1,2}$ | $\cos x_{T-1,2}$ | $t_{T,2}$ |

$X_{train}$ $y_{train}$

# TabPFN for Chaotic Time Series

**1.** $t_{N,:}$ → $\dot{x}_{N,:}$ | $x^2_{N,:}$ | $\cos x_{N,:}$ **$F_N$**

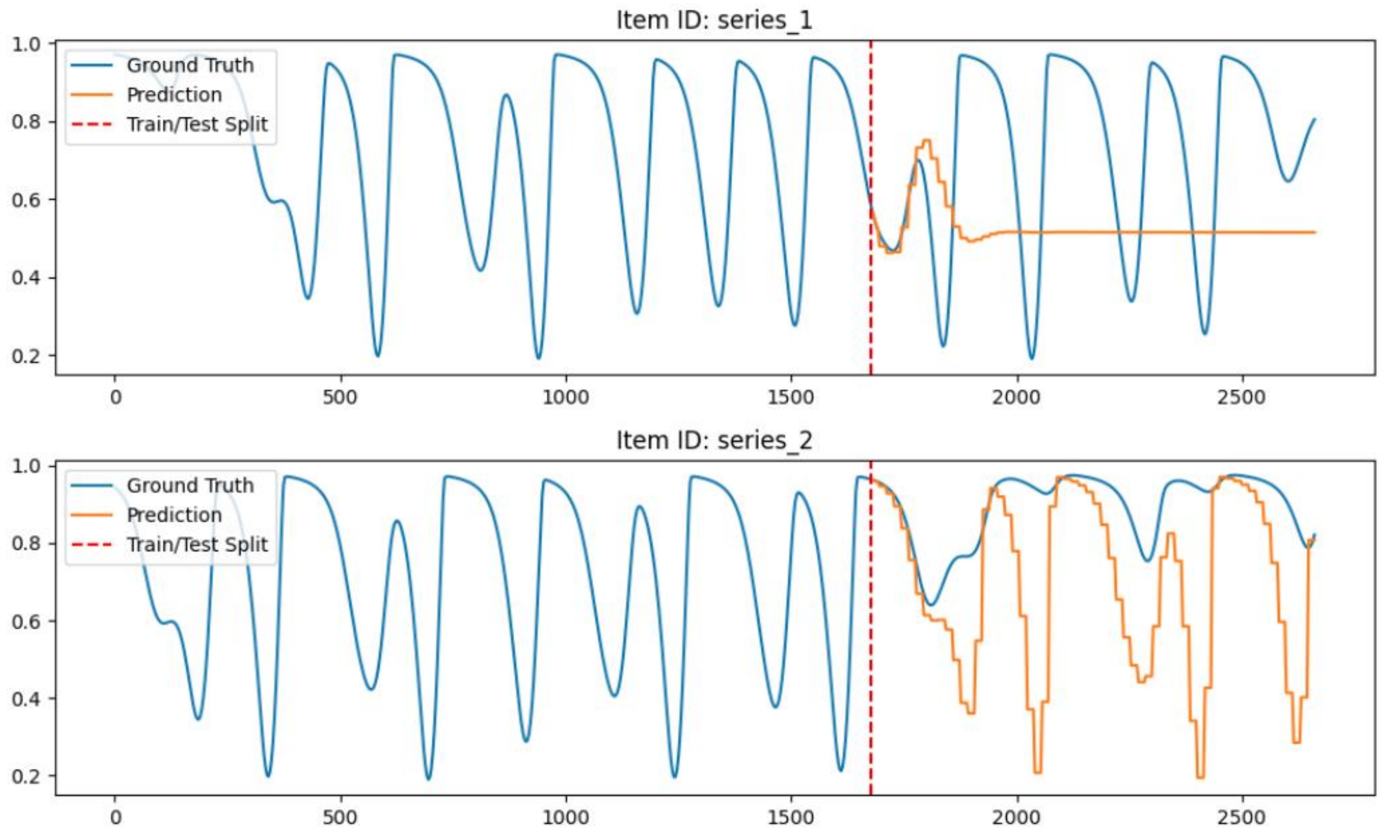
**2.** **$L_{N+1}$** N+1 | Dim 0-2 | $\dot{x}_{N,:}$ | $x^2_{N,:}$ | $\cos x_{N,:}$ → TabPFN → $t_{N+1,:}$

**3.** **$X_{train} = [X_{train}, L_{N+1}]$** **$y_{train} = [y_{train}, t_{N+1,:}]$**

$F_N$: Features at time N
$L_N$: Line in $X_{train}$ at time N
$t_N$ : target at time N

# Results

- fails to predict: we observe **context parroting**, flat lines
- fails to model the **coupling** between dimensions: **line independance**



# Discussion

- **Takens' theorem**: states that time-delayed copies of a low-dimensional measurement of a dynamical system result in a multivariate time series that preserves key topological features of the true attractor.
- **Koopman theory**, stating that nonlinear dynamics can be linearized by lifting them to a suitable high-dimensional space of observables.
- Introduce **channel attention**? (Panda or ChaosNaxus paper)
- Introduce multi-dimensional regression tabular model?