Beyond the Readout: Reservoir State Statistics for Model-Space Learning under Sparse Observations

Ao Chen[1,2,3] **Yijing Lu**[1,2,3] **Pengpeng Chen**[1,2,3*]

[1]School of Computer Science and Technology / School of Artificial Intelligence,
China University of Mining and Technology
[2]Mine Digitization Engineering Research Center of the Ministry of Education
[3]Jiangsu Provincial Industrial Technology Engineering Center for Intelligent Sensing
and Emergency IoT in Underground Space
Xuzhou 221116, China
{chenao57, luyijing, chenp}@cumt.edu.cn

Abstract

Multivariate time series often arrive with non-uniform timestamps and asynchronous per-channel observations, violating regular-grid assumptions of standard sequence models, while scarce labels and limited training resources make learning tasks more challenging. Model-space learning offers a lightweight alternative by fitting a dynamical model to each sequence and performing downstream analysis in the resulting model space. However, under irregular sampling, channels with very few observations weakly constrain the per-channel regression, leaving the fitted readout an unreliable summary. In this paper, we propose Reservoir State Statistics (RSS), a readout-complementary representation that augments the readout with three inversion-free trajectory summaries: centroid, spread, and terminal state. RSS is computed on the trajectory of the Global Decay Reservoir Network (GDRN), a continuous-time reservoir with per-channel time decay, cross-channel global pooling, and event-driven masked updates for asynchronous multivariate observations. Across multiple datasets, GDRN-RSS achieves competitive few-shot accuracy with CPU runtimes of seconds. Code is available at https://github.com/Alexs57/GDRN-RSS.

1 Introduction

Multivariate time series in industrial process monitoring, distributed seismic sensing, and clinical care often exhibit irregular sampling in real-world settings [Barros et al., 2026]. Heterogeneous sampling rates, event-driven acquisition, and sensor failures produce non-uniform timestamps and asynchronous per-channel observations, violating the regular-grid assumptions of standard sequence models. Scarce labeled data and compute-constrained deployment further motivate lightweight learning methods.

A common method is to resample or interpolate irregular observations onto a regular grid, but this preprocessing can distort the underlying dynamics. Recent work avoids imputation by handling irregular sampling directly inside the model, including decay-augmented recurrence [Che et al., 2018], continuous-time hidden states [Rubanova et al., 2019, Kidger et al., 2020], and time-aware attention [Shukla and Marlin, 2021, Luo et al., 2025]. However, these methods are typically trained end-to-end with gradients, making them less practical when labeled data and compute are scarce.

*Corresponding author.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

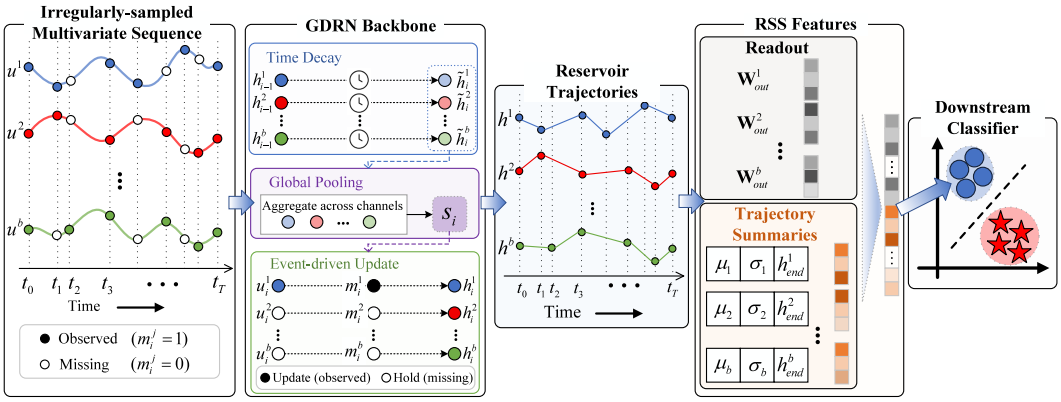


Figure 1: Pipeline of GDRN-RSS. An irregularly sampled multivariate sequence is processed by the GDRN backbone (time decay, global pooling, and event-driven update), producing per-channel reservoir trajectories. The trajectories feed two parallel branches that together form the RSS feature: a fitted readout and three bounded inversion-free trajectory summaries. Their concatenation gives the RSS features for a downstream classifier.

Model-space learning (MSL) [Chen et al., 2014, Bianchi et al., 2021] offers a lightweight alternative: rather than train an end-to-end sequence model, it fits a proper model to each sequence and uses the fitted models as the per-sequence representation. With Echo State Network (ESN) [Jaeger, 2001] as backbone, these readout models can be obtained efficiently by ridge regression. Recent work [Chen et al., 2025] extends this paradigm to irregular sequences via a continuous-time reservoir network.

However, fitting a per-channel readout reduces to a regression on the observed reservoir states. When a channel is observed only a few times, this regression is weakly constrained by data, so the fitted readout can be an unstable or incomplete summary of the observed trajectory (Lemma 2.1). This motivates complementing the readout with direct trajectory summaries.

In this paper, we propose *Reservoir State Statistics* (RSS), a readout-complementary four-component representation that keeps the fitted readout and adds three bounded trajectory summaries computed directly from the reservoir trajectory: the centroid μ, spread σ, and terminal state h_{end} of each channel's states. Producing those trajectories on asynchronous multivariate observations requires a backbone that handles non-uniform timestamps and partially observed events, which we instantiate as the *Global Decay Reservoir Network* (GDRN; Fig. 1). GDRN couples per-channel continuous-time decay, cross-channel global pooling, and event-driven masked updates inside a single fixed reservoir. The full pipeline runs on CPU in seconds and uses no gradient-based training.

Contributions.

- **Reservoir State Statistics (RSS).** We identify a structural failure of readout-only model-space representations: when a channel is observed fewer times than the reservoir dimension, the per-channel regression is weakly constrained and the fitted readout becomes an unreliable summary on its own (Lemma 2.1). Motivated by this failure mode, RSS augments the fitted readout with three bounded trajectory summaries (centroid, spread, terminal state) computed directly from the reservoir trajectory without additional fitting.

- **Global Decay Reservoir Network (GDRN).** We introduce a three-module continuous-time reservoir for asynchronous multivariate observations. Per-channel time decay enters inter-event intervals into the state update, cross-channel global pooling couples channels through a shared context, and event-driven masked updates restrict updates to observed channels. GDRN provides the trajectories from which RSS is computed.

- **Empirical evaluation.** Across nine benchmarks in five domains, including synthetic dynamics, industrial monitoring, wearable activity, seismic sensing, and clinical care, GDRN-RSS attains competitive few-shot accuracy on CPU in seconds.

2 Method

This section details the GDRN-RSS pipeline. §2.1 introduces the GDRN backbone, a continuous-time reservoir for asynchronous multivariate observations. §2.2 fits the per-channel readout and identifies its limitation under sparsity (Lemma 2.1). §2.3 defines the RSS representation to handle this limitation. §2.4 establishes a sufficient condition for GDRN's echo state property.

Background. An ESN [Jaeger, 2001, Lukoševičius and Jaeger, 2009] consists of a randomly initialized, fixed reservoir $W \in \mathbb{R}^{n\times n}$ with input map W_{in}; the hidden state evolves as $\mathbf{h}_t = f(W\mathbf{h}_{t-1} + W_{\text{in}}\mathbf{u}_t)$ and only the readout W_{out} is fitted by ridge regression, avoiding backpropagation through time. The echo state property (ESP) [Yildiz et al., 2012] ensures, under suitable spectral conditions, that the state depends only on input history, not on initialization. Model-space learning [Chen et al., 2014, Bianchi et al., 2021] represents each sequence by the fitted readout weights $W^{\star}_{\text{out}} = \arg\min_{W_{\text{out}}} \frac{1}{T}\sum_t \|y_t - W_{\text{out}}h_t\|^2 + \gamma\|W_{\text{out}}\|^2$, which a downstream classifier (e.g., Random Forest) operates on.

2.1 Global Decay Reservoir Network

GDRN runs a continuous-time reservoir directly on the observed events of an irregular multivariate sequence, without imputing missing inputs. Multivariate irregular sequences pose three challenges: irregular intervals call for continuous-time state updates; channel-wise asynchrony calls for masked updates that leave unobserved channels alone; and cross-channel structure calls for context shared across channels.

GDRN responds with three coupled operations within a single fixed reservoir. *Time decay* ages each channel's stored state by its own elapsed time; *global pooling* aggregates the decayed states across channels into a shared context; and an *event-driven update* modifies only the observed channels using this context. At each event t_i, these operations advance the stored state from H_{i-1} to H_i. The full procedure is summarized in Algorithm 1 (Appendix E.1).

Notation. Let $\{t_i\}_{i=1}^{T}$ be observation times with input $\mathbf{u}_i \in \mathbb{R}^b$ and diagonal mask $M_i \in \{0,1\}^{b\times b}$ indicating observed channels. The hidden state $H_i \in \mathbb{R}^{n\times b}$ stores n-dimensional reservoir states for b channels. The reservoir weights $W \in \mathbb{R}^{n\times n}$ and input weights $W_{\text{in}} \in \mathbb{R}^{n\times b}$ are randomly initialized and fixed. Let $\ell_i(j)$ record the last observation time of channel j. Channel-level quantities use subscripts, e.g., μ_j, σ_j, T_j, $\mathbf{y}_j$, $\mathcal{I}_j$, and $h_j \in \mathbb{R}^{T_j\times n}$. Superscripts indicate single-channel extractions from multi-indexed states or parameters, e.g., h_i^j, h_{end}^j, and W_{out}^j.

Time decay. Each channel runs on its own continuous-time clock. Between observations, its state decays under the linear ODE $\dot{h} = -\lambda h$ with closed-form solution $h(t+\Delta t) = e^{-\lambda\Delta t}h(t)$. At event t_i, GDRN scales each channel's states by the time elapsed since that channel's own last observation:

$$\widetilde{H}_i = H_{i-1}\,\text{diag}\big(e^{-\lambda(t_i-\ell_{i-1}(1))},\ldots,e^{-\lambda(t_i-\ell_{i-1}(b))}\big). \tag{1}$$

The decay rate λ sets the forgetting timescale and shares an operator-norm budget with W in the ESP condition (Remark C.4).

Global pooling. Per-channel decay alone evolves the reservoir as b independent subsystems, with no cross-channel coupling. GDRN averages the decayed states across channels, passes the result through the reservoir map W, and broadcasts it back as a shared context:

$$S_i = \Big(W \cdot \tfrac{1}{b}\widetilde{H}_i\mathbf{1}\Big)\mathbf{1}^{\top} \in \mathbb{R}^{n\times b}. \tag{2}$$

Each observed channel therefore enters its update with information from all channels' decayed states aggregated through this shared context.

Event-driven update. At event t_i, only observed channels update their state: their new value is the activation of the shared context plus channel-wise input injection, while unobserved channels keep their stored value. The diagonal mask M_i encodes the split:

$$H_i = H_{i-1}(I - M_i) + f\big(S_i + W_{\text{in}}\,\text{diag}(\mathbf{u}_i)\big)\,M_i, \tag{3}$$

where $f = \tanh$. Unobserved channels still contribute to S_i through their decayed copies. The last-observation vector ℓ_i refreshes for observed channels and carries over otherwise, so each channel's next decay covers the full elapsed interval since its last observation.

2.2 Model-space representation and its limitation

Given the GDRN trajectory, each sequence is represented by fitting a per-channel linear readout via ridge regression:

$$W_{\text{out}}^{j\star} = (h_j^\top h_j + \gamma T_j I)^{-1} h_j^\top \mathbf{y}_j, \qquad j = 1, \ldots, b, \tag{4}$$

where $h_j \in \mathbb{R}^{T_j \times n}$ stacks the hidden states at the T_j observation times of channel j, and $\mathbf{y}_j \in \mathbb{R}^{T_j}$ are the corresponding observed values. The base model-space representation is $\phi = \text{vec}(W_{\text{out}}^\star) \in \mathbb{R}^{bn}$, and through h_j it inherits the time-awareness, cross-channel coupling, and asynchronous-update structure of GDRN.

However, this readout breaks down when channel j has fewer observations than the reservoir dimension. With $T_j < n$, the Gram matrix $h_j^\top h_j$ is rank-deficient, and the ridge solution is identified only within a T_j-dimensional subspace of $\mathbb{R}^n$ (a standard linear-algebra fact):

Lemma 2.1 (Readout rank deficiency under sparsity)**.** *Let $h_j \in \mathbb{R}^{T_j \times n}$ be the stacked reservoir states for channel j with T_j observations and reservoir dimension n. When $T_j < n$, the empirical Gram matrix $h_j^\top h_j \in \mathbb{R}^{n \times n}$ has rank at most T_j and therefore at least $n - T_j$ zero eigenvalues. Consequently:*

(a) The ridge solution $W_{\text{out}}^{j\star}$ lies entirely in the row space of h_j, whose dimension is at most T_j.

(b) At least $n - T_j$ directions in the null space of $h_j^\top h_j$ are unsupported by the data: the corresponding components of $W_{\text{out}}^{j\star}$ vanish under the ridge regularizer rather than being identified from data.

(c) In contrast, the bounded reservoir state summaries $\mu_j = \frac{1}{T_j}\sum_i h_i^j$, $\sigma_j = \text{std}(\{h_i^j\})$, and $h_{\text{end}}^j = h_{\max(\mathcal{I}_j)}^j$ remain well-defined for any $T_j \geq 1$.

Proof sketch. Let $h_j = U\Lambda V^\top$ be the SVD. For $T_j < n$, at most T_j singular values are nonzero, and the remaining $n - T_j$ right-singular vectors contribute zero to the ridge solution $\sum_k \frac{\sigma_k}{\sigma_k^2 + \gamma T_j} v_k u_k^\top \mathbf{y}_j$. The bounded summaries $\mu_j, \sigma_j, h_{\text{end}}^j$ require no inversion and exist for any $T_j \geq 1$. Full proof is presented in Appendix A.

Lemma 2.1 suggests a design criterion under sparsity. When $T_j < n$, the data-supported term $h_j^\top h_j$ is rank-deficient: ridge regularization keeps the readout numerically well-defined, but identifies it only within the row space of h_j, and the components in the orthogonal directions vanish under the regularizer rather than being identified from data. We therefore complement the readout with trajectory-level functionals computed directly from h_j without solving an inverse problem and well-defined for any $T_j \geq 1$. Among single-pass summaries of the trajectory $\{h_i^j\}_{i \in \mathcal{I}_j}$, the natural choices are the centroid μ_j and spread σ_j of where the state settles under the input, plus the terminal state h_{end}^j at the last observation, which captures the recursively accumulated reservoir state up to the channel's most recent observation.

2.3 Reservoir State Statistics

RSS keeps the fitted readout and adds three trajectory summaries that satisfy this design rule. The readout $W_{\text{out}}^{j\star}$ encodes how the reservoir state should be combined to reconstruct observed values. Its solution lies in the row space of h_j and inherits the rank deficiency of Lemma 2.1. The trajectory summaries take the complementary view: μ_j is the centroid of the reservoir trajectory, σ_j its spread, and h_{end}^j its terminal value, defined independently of any fitting objective and for any $T_j \geq 1$:

$$\mu_j = \frac{1}{T_j}\sum_{i \in \mathcal{I}_j} h_i^j, \quad \sigma_j = \sqrt{\frac{1}{T_j}\sum_{i \in \mathcal{I}_j} (h_i^j - \mu_j)^2}, \quad h_{\text{end}}^j = h_{\max(\mathcal{I}_j)}^j \in \mathbb{R}^n, \tag{5}$$

where $\mathcal{I}_j = \{i : M_i(j,j) = 1\}$ and $T_j = |\mathcal{I}_j|$. The terminal state h_{end}^j is the reservoir state at channel j's most recent observation, shaped by preceding events through decay and pooled context. Under extreme sparsity (median $T_j \approx 3$ on P12), it can be more informative than averages over very

few observations. The full RSS representation concatenates all four components:

$$\phi_{\mathrm{RSS}} = \left[W_{\mathrm{out}}^{1\star};\, \mu_1;\, \sigma_1;\, h_{\mathrm{end}}^{1};\, \ldots;\, W_{\mathrm{out}}^{b\star};\, \mu_b;\, \sigma_b;\, h_{\mathrm{end}}^{b}\right] \in \mathbb{R}^{4bn}. \tag{6}$$

Computing μ_j, σ_j, and h_{end}^{j} adds running-sum accumulators and a terminal-index lookup, and the representation grows from bn to $4bn$ (2,880 dimensions for a 36-channel dataset at n=20). For channels never observed in a sequence (T_j=0), the four components default to $\mathbf{0}$ to keep the representation fixed-dimensional.

Proposition 2.2 (Reservoir variance separates lagged structure)**.** *Consider the scalar linear reservoir $h_t = \alpha h_{t-1} + \beta x_t$ with $|\alpha| < 1$ and $\beta \neq 0$, driven in class $c \in \{0, 1\}$ by a zero-mean weakly stationary process with autocovariance $C_c(k)$=$\mathrm{Cov}(x_t, x_{t-k} \mid c)$. The stationary class-conditional state variance is*

$$\nu_c \;:=\; \mathrm{Var}(h_t \mid c) \;=\; \frac{\beta^2}{1-\alpha^2}\Big(C_c(0) \;+\; 2\sum_{k\geq 1}\alpha^k C_c(k)\Big).$$

If the two classes share $\mathbb{E}[x_t]$ and $C_0(0)$=$C_1(0)$, they are indistinguishable in raw first and second moments, yet the reservoir variances differ by

$$\nu_1 - \nu_0 \;=\; \frac{2\beta^2}{1-\alpha^2}\,\Delta_\alpha, \qquad \Delta_\alpha \;:=\; \sum_{k\geq 1}\alpha^k\big(C_1(k) - C_0(k)\big),$$

which is nonzero whenever the α-weighted lag structure differs ($\Delta_\alpha \neq 0$).

Proposition 2.2 isolates the role of σ_j in the scalar linear setting: even when two classes share their raw first and second moments, the reservoir variance separates them whenever the α-weighted lag structure differs. The full nonlinear multivariate GDRN does not inherit this closed form, and the proposition only motivates σ_j as a statistic retaining lag-sensitive variation beyond raw moments. Proof and an implicit-kernel view of RSS as a structured random projection are in Appendix B and B.1.

2.4 Echo State Property of GDRN

The echo state property (ESP) makes the reservoir state forget its initial condition: trajectories from any two initial states converge under any bounded input. Trajectory boundedness ($\|h_i^j\|_\infty \leq 1$) follows unconditionally from $\tanh$.

Theorem 2.3 (ESP for GDRN, sufficient condition)**.** *Assume K-coverage (every channel is updated within any K consecutive events; formal statement in Appendix C), and let $\Delta t_{\min}^{\mathrm{all}} = \inf_i \min_j \Delta t_i(j) > 0$. Then GDRN satisfies the ESP if*

$$L_f\, e^{-\lambda \Delta t_{\min}^{\mathrm{all}}} \|W\|_2 < 1. \tag{7}$$

Theorem 2.3 is a sufficient condition for the ESP, not a guarantee on accuracy, and its K-coverage assumption fails on P12 and P19 (Remark C.3). When K-coverage fails in regimes of extreme per-channel sparsity, any rescue observed there should be read as the boundedness of the trajectory summaries $\mu_j, \sigma_j, h_{\mathrm{end}}^{j}$ in Lemma 2.1 rather than a consequence of this theorem (Appendix I, Remark C.3).

The bound uses $\|W\|_2$ rather than spectral radius because the latter is not a general ESP certificate for nonlinear, non-normal reservoirs [Yildiz et al., 2012]. The decay term $e^{-\lambda\Delta t}$ multiplies the standard ESN operator-norm budget by $e^{\lambda \Delta t_{\min}^{\mathrm{all}}}$, so λ and $\|W\|_2$ share a budget (Remark C.4). Full proof is in Appendix C; an empirical (ρ, λ) sweep is in Appendix K.

3 Experiments

We evaluate GDRN-RSS across nine datasets under a single fixed configuration. §3.1 and §3.2 describe the data and baselines, §3.3 reports the main few-shot results, §3.4 isolates the contributions of GDRN's architectural pieces and RSS's components, §3.5 examines the label budget, and §3.6 reports wall-clock and robustness analyses.

Table 1: k-shot classification (mean over 5 seeds; k=5 for Lorenz, k=20 for Person Activity, k=10 otherwise). Reported metric is Accuracy (%) except AUROC for the binary P12/P19 tasks. **Bold** per column = best, underline = second. Column codes are below the table. Full ±std version and per-method details in Appendix Table 3 and Appendix E.3.

	Lz			**Ku**		**CM**		**TE**		**PA**			**Vo**			**P12**	**P19**	**PAM**
Method	M_{20}	B_{71}	E_{23}	M_{20}	E_8	M_{20}	M_5	M_{50}	E_{11}	N_{37}	M_{19}	$M_{7.5}$	N_{44}	M_{22}	M_9			
Feature-based																		
RF-stats	98.0	99.5	**100.0**	56.4	72.8	96.7	**95.0**	85.8	79.0	85.9	79.9	73.6	63.2	61.8	61.1	68.9	75.2	77.8
StatsGap	96.2	98.7	**100.0**	56.2	72.8	96.9	94.9	86.6	79.2	85.9	78.1	71.9	62.4	59.6	59.0	65.6	**77.8**	78.0
catch22	43.7	69.8	74.5	79.1	72.7	89.5	73.7	51.5	51.3	33.3	36.3	23.2	58.0	52.3	51.8	58.5	73.8	72.0
MiniROCKET	79.2	79.1	81.5	75.0	76.6	95.8	82.6	72.1	66.1	73.3	58.9	46.1	63.1	62.4	61.4	67.3	75.7	**80.0**
MultiROCKET	80.4	80.8	85.5	80.0	82.0	97.0	87.7	71.4	68.7	69.3	60.6	48.4	58.9	59.6	59.8	67.2	74.7	77.2
Deep learning																		
GRU-D	34.3	32.3	34.4	33.7	33.8	32.6	33.7	24.9	5.2	41.4	16.9	20.4	37.2	35.4	31.4	63.9	70.3	37.7
L-ODE	33.5	35.4	33.8	35.9	44.6	39.2	34.2	55.9	47.7	45.7	28.5	22.1	53.2	34.2	34.1	59.7	61.2	52.7
Raindrop	33.3	33.3	33.3	33.3	38.1	33.3	33.4	5.5	9.7	19.8	18.3	18.5	32.1	34.9	34.8	58.7	70.4	19.8
mTAND	63.2	70.5	60.8	51.4	53.1	51.1	37.6	33.5	36.3	63.3	54.0	42.3	31.4	31.5	31.8	64.7	62.0	70.8
ATENet	53.4	56.3	57.7	37.6	38.1	45.9	39.4	40.5	39.2	60.2	39.3	26.5	34.0	34.5	34.4	62.7	60.6	66.8
Warpf.	39.6	40.0	39.9	43.3	55.0	40.0	41.7	44.4	23.8	52.8	18.4	20.8	34.7	35.1	37.0	64.5	74.2	31.5
Hi-Patch	—	—	—	59.5	47.9	43.5	40.8	49.1	48.7	72.4	38.1	29.8	45.2	49.7	43.8	54.9	61.0	23.5
PrimeNet	46.8	53.8	60.1	36.9	38.3	34.3	39.1	25.1	23.2	48.2	20.0	17.1	40.0	30.1	30.3	68.6	63.8	59.4
ViTST	90.0	86.8	97.6	86.2	**86.4**	98.1	**95.0**	53.9	49.7	37.5	37.3	25.2	64.5	63.5	60.3	63.5	68.5	54.8
Model-space learning (MSL)																		
ESN-MS	35.4	63.2	39.0	72.0	63.9	43.0	36.0	66.9	54.3	50.6	44.2	46.6	61.7	55.9	56.5	56.5	69.6	69.3
Ct-Echo	80.6	91.2	94.0	72.7	77.9	95.2	80.3	85.8	75.5	77.4	69.0	65.5	**65.0**	62.8	60.8	62.0	72.6	71.6
GDRN-RSS (ours)	**100.0**	**100.0**	**100.0**	**91.0**	76.1	**99.3**	93.7	**90.1**	**80.4**	**87.2**	**86.5**	**80.1**	64.0	**64.9**	**63.4**	**72.7**	72.2	77.2

Lz=Lorenz, **Ku**=Kuramoto, **CM**=CML, **TE**=TEP, **PA**=Person Activity, **Vo**=Volcano CAU. M=MCAR, B=Burst, E=Event-triggered, N=natural; subscript = retained-observation %. '—'=OOM.

3.1 Datasets

We evaluate on nine datasets across five domains: three synthetic dynamical systems (Lorenz-96, Kuramoto, CML) for controlled difficulty, and six real-world tasks spanning industrial process monitoring (TEP), wearable activity recognition (Person Activity, PAM), seismic event classification (Volcano CAU), and clinical ICU outcome prediction (P12, P19). For P12/P19/PAM we use the preprocessed data and the *same five train/val/test splits* as [Zhang et al., 2022]. Full statistics are in Appendix D (Table 2).

3.2 Baselines and setup

The comparison spans sixteen baselines in seven categories (details and references in Appendix E.3): simple statistics (RF-stats, StatsGap, catch22 [Lubba et al., 2019]), random convolutional kernels (MiniROCKET [Dempster et al., 2021], MultiROCKET [Tan et al., 2022]), RNN (GRU-D [Che et al., 2018]), ODE/graph (Latent ODE [Rubanova et al., 2019], Raindrop [Zhang et al., 2022]), time-aware attention (mTAND [Shukla and Marlin, 2021], ATENet [Lee et al., 2025], Warpformer [Zhang et al., 2023], Hi-Patch [Luo et al., 2025]), pre-trained encoders (PrimeNet [Chowdhury et al., 2023], ViTST [Li et al., 2023]), and two MSL methods (ESN-MS [Chen et al., 2014], Ct-Echo [Chen et al., 2025]). For all datasets, all methods are re-run under identical k-shot subsamples, seeds, test splits, and hardware.

Protocol. All baselines, including GDRN-RSS, share the same k-shot subsamples, seeds, test splits, and hardware (full settings in Appendix E.3). GDRN-RSS uses a single fixed configuration (n=20, ρ=0.9, λ=0.1, sparsity 0.5, $\gamma=10^{-4}$, Random Forest with 300 trees) on every dataset and every experiment, with the robustness of this default verified in Appendix N and the RSS feature remaining informative across alternative downstream classifiers (RF, SVM, LogReg, kNN, RidgeCV, with RF consistently strongest) in Appendix Q.

3.3 Main results

Table 1 reports k-shot accuracy across 18 conditions. The closed-form GDRN-RSS pipeline reaches the few-shot lead on Lorenz across all three masking variants, Kuramoto and CML at 20% MCAR,

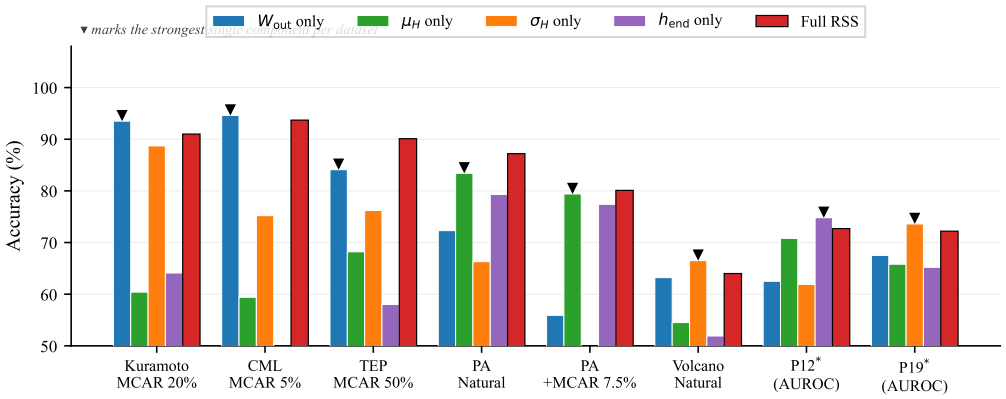


Figure 2: RSS component decomposition across eight conditions (accuracy %, k-shot, 5 seeds; AUROC for P12*/P19*). Each bar shows W_{out}, μ_H, σ_H, or h_{end} as the sole feature set and *Full RSS* as their concatenation, all with the same downstream RF. ▼ marks the strongest single component per condition. Full 22-condition ±std table in Appendix L.

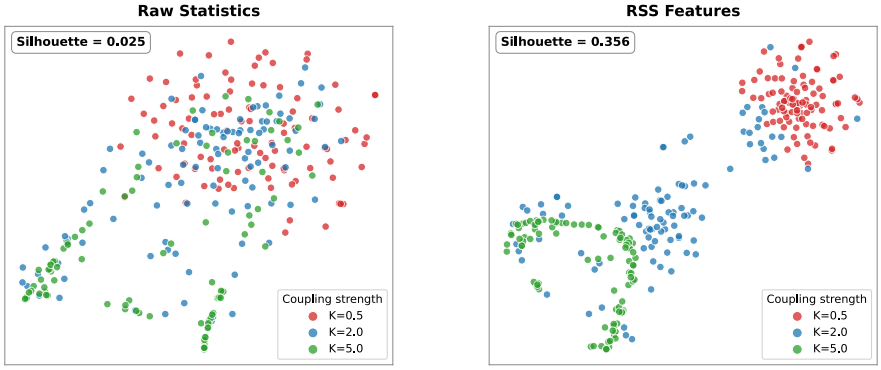


Figure 3: t-SNE on Kuramoto MCAR 20% (N=300, three coupling strengths $K \in \{0.5, 2, 5\}$). **Left**: raw per-channel statistics collapse into one overlapping blob (silhouette 0.025). **Right**: RSS features resolve three classes (silhouette 0.356, 14.2× improvement). The same condition where RSS reaches 91.0% versus 56.4% for raw stats in Table 1.

both TEP regimes, Person Activity at all three observation rates, the masked Volcano settings, and the clinical P12. Lorenz holds at 100% throughout, and Person Activity stays above 80% even at 7.5% observation. The remaining conditions fall within a few accuracy points of the leader, with the exception of Kuramoto Event-triggered (8% observation), where ViTST takes a larger margin through its image-rendered attention. Fig. 9 (Appendix T) extends the picture with accuracy-vs.-missingness curves on TEP, Person Activity, and Volcano CAU.

The result is consistent with the design. Closed-form ridge fitting sidesteps the gradient training that handicaps few-shot baselines, and the bounded RSS summaries do not break under sparsity that would render the readout alone rank-deficient. This makes GDRN-RSS a competitive choice for irregular multivariate time series under label scarcity. Component-level diagnostics, including the bounded summaries' lift on the rank-deficient clinical extreme, are in §3.4.

3.4 Ablation study

Fig. 2 previews the per-component decomposition of RSS, with the full 22-condition ablation table in Appendix L (Table 9). Fig. 3 adds a qualitative view on Kuramoto MCAR 20%, where RSS features resolve three coupling regimes that raw per-channel statistics blur into one cluster.

Architecture decomposition. GDRN-RSS extends two simpler MSL baselines, ESN-MS (no decay, no pooling) and Ct-Echo (decay only, no pooling). The ESN-MS → Ct-Echo step adds

continuous-time decay and recovers the missing-data gain (+18.9 on TEP MCAR 50%, +52.2 on CML MCAR 20%). The Ct-Echo → GDRN-RSS step adds cross-channel pooling and the four-component RSS feature jointly (+17.1 on Kuramoto Full, +18.3 on Kuramoto MCAR 20%, +4.3 on TEP MCAR 50%). Since this step changes both the backbone and the feature, we read it as evidence for the combined design rather than an isolated estimate of pooling. A backbone × feature factorial (Appendix M, Table 12) separates the backbone from the feature. RSS adds 4.7 points on Ct-Echo and 5.0 on GDRN. The GDRN backbone adds 2.3 with W and 2.6 with RSS, largest on Kuramoto Full and MCAR (+19 to +21). The two stages contribute distinct mechanisms. Continuous-time decay handles non-uniform gaps, while pooling and RSS together add cross-channel coupling and the bounded summaries that remain well-defined under rank deficiency.

Lemma 2.1's failure mode has an empirical signature. The lemma states that, under rank deficiency, $W_{\text{out}}^{j\star}$ is identified only within the row space of h_j, while $\mu_j, \sigma_j, h_{\text{end}}^j$ stay well-defined for any $T_j \geq 1$. On Person Activity, as observation rate drops 37%→19%→7.5%, W-only loses 16.4 points (72.3→64.4→55.9) while μ-only loses 4.0 (83.4→83.1→79.4). On TEP and the masked Volcano conditions the ordering does not hold: μ-only loses as much as or more than W-only (TEP Full→MCAR 50%: 5.5 vs. 2.9; Volcano Natural→MCAR 9%: 2.8 vs. 0.9). It reverses on CML, where the readout stays nearly flat under added MCAR while μ degrades sharply. The class signal there is concentrated in the per-step value distribution, which the readout (whose targets are per-step values) retains more directly than the trajectory averages. The clinical benchmarks P12 and P19 (median T_j=3 and 2, well below the reservoir dimension) are the strongest case for the lemma. The W-only readout there is rank-deficient by construction, and adding the bounded summaries lifts it on both, reaching the few-shot lead on P12 and narrowing the P19 gap (Appendix I; Table 5). The dominant single component varies by dataset (W on Kuramoto 93.8 and CML 94.6; σ on Volcano 66.5 and P19 73.6; μ on PA 83.4 and PAM 76.4; h_{end} on P12 74.8).

Concatenation vs. best subset. A single component sometimes matches or marginally beats Full RSS. On Kuramoto MCAR 20%, W-only reaches 93.5 against Full RSS at 91.0; on Volcano natural, σ-only reaches 66.5 against Full at 64.0; on P12, h_{end}-only reaches 74.8 against Full at 72.7. The largest gap is under 3 accuracy points, and selecting the best subset per dataset would require labeled validation that the few-shot regime does not afford, so we concatenate by default.

3.5 Effect of the label budget

At the budgets of Table 1 (k=5–20 per class), GDRN-RSS ranks first on 17 of the 22 conditions of Table 3 (14 outright, 3 tied). At k=100 (Table 19), the deep baselines recover on the synthetic systems (Warpformer 43.3→91.8 on Kuramoto MCAR, vs. RSS 97.0). ViTST passes RSS on Kuramoto Event (91.9 vs. 90.0) and Volcano Natural (81.2 vs. 79.5). No other deep baseline passes RSS in that table. At k=1–3, accuracy degrades gradually (Appendix J), but no baselines were run at these budgets. On P19 at $k \geq 10$, tabular features lead (Table 4).

3.6 Efficiency and robustness

GDRN-RSS extracts features in 2.9 s on P19 (N=38,803), with Random Forest fitting adding ∼54 s. On P12 full-data (Fig. 4), the pipeline runs end-to-end in 11.8 s, faster than every gradient-trained baseline. The heaviest baselines, on GPU, take orders of magnitude longer (GRU-D 608.7 s, Latent ODE 1804 s, ViTST 4002 s).

At k=100 (Appendix R), the supervised DL baselines that drop under few-shot scarcity recover, while the proposed GDRN-RSS retains competitive accuracy at orders-of-magnitude lower compute.

Sensitivity to (ρ, λ) (Appendix N), reservoir size n (Appendix P), and reservoir-seed initializations (Appendix O) show consistent accuracy across the grid.

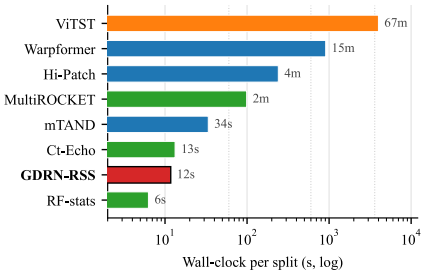


Figure 4: Wall-clock per split on P12 fulldata (N=11,988). Full 17-method bar in Appendix S.

4 Related work

4.1 Methods for irregular multivariate time series

Irregular multivariate time series violate the uniform-grid assumption underlying most sequence models. Timestamps are non-uniform, and channels are observed at different events. The naive workaround is to resample onto a uniform grid via forward-fill, resampling or imputation, which injects spurious values between observations and distorts the underlying dynamics, especially under heavy missingness or channel-wise asynchrony.

One line of work pre-trains encoders that handle irregular sampling natively. PrimeNet [Chowdhury et al., 2023] contrastively pre-trains a time-aware transformer on irregular time series and fine-tunes for classification. ViTST [Li et al., 2023] renders each channel as a 2D image (time on the x-axis, value on the y-axis) and applies an ImageNet-pretrained Vision Transformer.

A separate line consumes the timestamps explicitly at the model level. Decay-augmented RNNs [Che et al., 2018] accumulate evidence across observation gaps via learned input and hidden decay. Neural ODE and CDE encoders [Rubanova et al., 2019, Kidger et al., 2020, Chen et al., 2023] model the latent trajectory between observations as a continuous flow integrated by an ODE solver. Graph pooling methods [Zhang et al., 2022] treat sensors as nodes and learn inter-channel structure through message passing. Time-aware attention reshapes the transformer to admit irregular timestamps, with the recent wave including mTAN [Shukla and Marlin, 2021], Warpformer [Zhang et al., 2023], Hi-Patch [Luo et al., 2025], and ATENet [Lee et al., 2025]. These architectures train end-to-end on GPU hardware and rely on substantial label budgets.

4.2 Model-space learning

Model-space learning has been studied for efficient time series analysis. MSL represents each sequence by the readout of a fitted dynamical model and classifies in the resulting model space [Chen et al., 2014, 2015]. With a reservoir backbone [Jaeger, 2001, Maass et al., 2002, Jaeger et al., 2007], the dynamics are fixed and the readout is fitted in closed form via ridge regression. Some researchers [Bianchi et al., 2021] enumerate three single-element per-channel choices, namely the terminal state, the next-input readout, or the next-state readout. Recent work has added spectral-aware initialization [Liu et al., 2025a] and multi-scale aggregation [Liu et al., 2025b], and applied reservoir readout features to subsurface anomaly detection [Zhou et al., 2025]. Recently, Ct-Echo [Chen et al., 2025] has been proposed to address irregular sampling via continuous-time dynamics but keeps each channel's reservoir state independent, leaving asynchronous cross-channel observations unmodeled.

4.3 Our position

In this paper, GDRN integrates continuous-time decay, cross-channel global pooling, and event-driven masked updates into a single reservoir. The cross-channel pooling addresses the per-channel-isolated dynamics of prior reservoir MSL. RSS uses the fitted readout, the terminal state and two trajectory moments μ_j, σ_j. The resulting four-component per-channel feature bypasses the rank-deficient readout failure under sparse observations (Lemma 2.1), giving a lightweight framework for irregular multivariate time series classification under few-shot label budgets.

5 Limitations

(L1) *Fixed reservoir parameters.* (W, W_{in}) is drawn once and frozen, and a single $(\rho, \lambda, n, \text{sparsity})$ configuration serves every dataset. This keeps the pipeline closed-form and gradient-free but precludes per-dataset hyperparameter adaptation. Closed-form schemes for selecting reservoir hyperparameters from sequence statistics would relax this without reintroducing gradient training.

(L2) *Uniform cross-channel pooling.* The pooling step weights every channel equally regardless of how informative it is, so noisy or near-empty channels enter the shared context with the same weight as carrier channels. Channel-aware aggregation respecting per-channel signal-to-noise ratios is conceptually straightforward, but reconciling it with the closed-form readout is an open question. Learning the channel groups from data is a possible extension (Appendix M).

(L3) *Channel-failure encoding.* When a channel is never observed in a sequence ($T_j{=}0$), the current implementation sets all four RSS components to zero to keep the representation fixed-dimensional. This is a simple convention that conflates an unobserved channel with a near-zero trajectory. In applications where channel unavailability reflects sensor failure, device dropout, or other physical conditions correlated with the label, encoding the absence directly is a future direction.

(L4) *Baseline tuning at small k.* The deep baselines use the authors' configurations where available, with early stopping on validation AUROC or accuracy. We did not tune their model size, regularisation, or pretraining for the few-shot budgets.

6 Conclusion

RSS extends the reservoir model-space family with three bounded trajectory summaries (centroid, spread, terminal state) that remain well-defined under rank-deficient sparsity (Lemma 2.1). The GDRN backbone produces these summaries on asynchronous multivariate observations through continuous-time decay, cross-channel global pooling, and event-driven masked updates. The closed-form pipeline reaches competitive accuracy across nine benchmarks under few-shot label budgets, running in seconds on CPU. Open directions include adapting reservoir hyperparameters within the closed-form constraint, weighting channels by informativeness during pooling, and encoding channel availability when sensor or device failures correlate with the label.

Acknowledgments and Disclosure of Funding

This work was supported by the National Natural Science Foundation of China (62272462) and the Natural Science Foundation of Jiangsu Province of China for Distinguished Young Scholars (BK20230045). The authors declare no competing interests.

References

Marcel Barros, Lucas P Fontenele, Mariana S Silva, Thiago Rissi, Eduardo Cabrera, Eduardo A Tannuri, Edson S Gomi, Rodrigo A Barreira, and Anna H Reali Costa. Embracing data irregularities in multivariate time series. *Applied Soft Computing*, 186:114039, 2026.

Filippo Maria Bianchi, Simone Scardapane, Sigurd Løkse, and Robert Jenssen. Reservoir computing approaches for representation and classification of multivariate time series. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):2169–2179, 2021.

Zhengping Che, Sanjay Purushotham, Kyunghyun Cho, David Sontag, and Yan Liu. Recurrent neural networks for multivariate time series with missing values. *Scientific reports*, 8(1):6085, 2018.

Ao Chen, Xiren Zhou, and Huanhuan Chen. Efficient anomaly detection of irregular sequences in ct-echo model space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 15731–15739, 2025.

Huanhuan Chen, Peter Tiňo, Ali Rodan, and Xin Yao. Learning in the model space for cognitive fault diagnosis. *IEEE Transactions on Neural Networks and Learning Systems*, 25(1):124–136, 2014.

Huanhuan Chen, Fengzhen Tang, Peter Tiňo, Anthony G Cohn, and Xin Yao. Model metric co-learning for time series classification. In *Twenty-fourth international joint conference on artificial intelligence*, pages 3387–3394, 2015.

Yuqi Chen, Kan Ren, Yansen Wang, Yuchen Fang, Weiwei Sun, and Dongsheng Li. Contiformer: continuous-time transformer for irregular time series modeling. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA, 2023. Curran Associates Inc.

Ranak Roy Chowdhury, Jiacheng Li, Xiyuan Zhang, Dezhi Hong, Rajesh K. Gupta, and Jingbo Shang. PrimeNet: Pre-training for irregular multivariate time series. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 7184–7192, 2023.

Marco Cuturi. Fast global alignment kernels. In *Proceedings of the 28th International Conference on Machine Learning (ICML)*, pages 929–936, 2011.

Angus Dempster, Daniel F Schmidt, and Geoffrey I Webb. MINIROCKET: A very fast (almost) deterministic transform for time series classification. In *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 248–257, 2021.

James J. Downs and Ernest F. Vogel. A plant-wide industrial process control problem. *Computers & Chemical Engineering*, 17(3):245–255, 1993.

Camilo Espinosa-Curilem, Millaray Curilem, and Daniel Basualto. A framework for real-time volcano-seismic event recognition based on multi-station seismograms and semantic segmentation models. *IEEE Transactions on Geoscience and Remote Sensing*, 64:1–13, 2026.

Ary L Goldberger, Luis AN Amaral, Leon Glass, Jeffrey M Hausdorff, Plamen Ch Ivanov, Roger G Mark, Joseph E Mietus, George B Moody, Chung-Kang Peng, and H Eugene Stanley. PhysioBank, PhysioToolkit, and PhysioNet: components of a new research resource for complex physiologic signals. *Circulation*, 101(23):e215–e220, 2000.

Lukas Gonon and Juan-Pablo Ortega. Reservoir computing universality with stochastic inputs. *IEEE Transactions on Neural Networks and Learning Systems*, 31(1):100–112, 2020.

Herbert Jaeger. The "echo state" approach to analysing and training recurrent neural networks-with an erratum note. *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report*, 148(34):13, 2001.

Herbert Jaeger, Mantas Lukoševičius, Dan Popovici, and Udo Siewert. Optimization and applications of echo state networks with leaky-integrator neurons. *Neural Networks*, 20(3):335–352, 2007.

Boštjan Kaluža, Violeta Mirchevska, Erik Dovgan, Mitja Luštrek, and Matjaž Gams. An agent-based approach to care in independent living. In *Ambient Intelligence (AmI 2010)*, volume 6439 of *Lecture Notes in Computer Science*, pages 177–186. Springer, 2010.

Kunihiko Kaneko. Spatiotemporal chaos in one- and two-dimensional coupled map lattices. *Physica D: Nonlinear Phenomena*, 37(1–3):60–82, 1989.

Patrick Kidger, James Morrill, James Foster, and Terry Lyons. Neural controlled differential equations for irregular time series. *Advances in Neural Information Processing Systems*, 33:6696–6707, 2020.

Yoshiki Kuramoto. *Chemical Oscillations, Waves, and Turbulence*, volume 19 of *Springer Series in Synergetics*. Springer-Verlag, Berlin, 1984.

Sangho Lee, Kyeongseo Min, Youngdoo Son, and Hyungrok Do. Adaptive time encoding for irregular multivariate time-series classification. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

Zekun Li, Shiyang Li, and Xifeng Yan. Time series as images: Vision transformer for irregularly sampled time series. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Shikang Liu, Chuyang Wei, Xiren Zhou, and Huanhuan Chen. Spectral-aware reservoir computing for fast and accurate time series classification. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 39774–39788. PMLR, 13–19 Jul 2025a.

Shikang Liu, Xiren Zhou, and Huanhuan Chen. Multiscale temporal dynamic learning for time series classification. *IEEE Transactions on Knowledge and Data Engineering*, 37(6):3543–3555, 2025b.

Edward N Lorenz. Predictability: A problem partly solved. In *Proc. Seminar on Predictability*, volume 1, pages 1–18. ECMWF, 1996.

Carl H. Lubba, Sarab S. Sethi, Philip Knaute, Simon R. Schultz, Ben D. Fulcher, and Nick S. Jones. catch22: CAnonical Time-series CHaracteristics. *Data Mining and Knowledge Discovery*, 33(6): 1821–1852, 2019.

Mantas Lukoševičius and Herbert Jaeger. Reservoir computing approaches to recurrent neural network training. *Computer science review*, 3(3):127–149, 2009.

Yicheng Luo, Bowen Zhang, Zhen Liu, and Qianli Ma. Hi-patch: Hierarchical patch gnn for irregular multivariate time series. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 41494–41519. PMLR, 2025.

Wolfgang Maass, Thomas Natschläger, and Henry Markram. Real-time computing without stable states: A new framework for neural computation based on perturbations. *Neural Computation*, 14 (11):2531–2560, 2002.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 20, pages 1177–1184, 2007.

Attila Reiss and Didier Stricker. Introducing a new benchmarked dataset for activity monitoring. In *2012 16th International Symposium on Wearable Computers*, pages 108–109, 2012.

Matthew A Reyna, Christopher S Josef, Russell Jeter, Supreeth P Shashikumar, M Brandon Westover, Shamim Nemati, Gari D Clifford, and Ashish Sharma. Early prediction of sepsis from clinical data: the PhysioNet/Computing in Cardiology Challenge 2019. *Critical Care Medicine*, 48(2): 210–217, 2020.

Yulia Rubanova, Ricky TQ Chen, and David K Duvenaud. Latent ordinary differential equations for irregularly-sampled time series. *Advances in neural information processing systems*, 32, 2019.

Satya Narayan Shukla and Benjamin Marlin. Multi-time attention networks for irregularly sampled time series. In *International Conference on Learning Representations*, 2021.

Chang Wei Tan, Angus Dempster, Christoph Bergmeir, and Geoffrey I. Webb. MultiRocket: multiple pooling operators and transformations for fast and effective time series classification. *Data Mining and Knowledge Discovery*, 36(5):1623–1646, 2022.

Izzet B. Yildiz, Herbert Jaeger, and Stefan J. Kiebel. Re-visiting the echo state property. *Neural Networks*, 35:1–9, 2012. ISSN 0893-6080.

Jiawen Zhang, Shun Zheng, Wei Cao, Jiang Bian, and Jia Li. Warpformer: A multi-scale modeling approach for irregular clinical time series. In *Proceedings of the 29th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 3273–3285. Association for Computing Machinery, 2023.

Xiang Zhang, Marko Zeman, Theodoros Tsiligkaridis, and Marinka Zitnik. Graph-guided network for irregularly sampled multivariate time series. In *International Conference on Learning Representations*, 2022.

Xiren Zhou, Shikang Liu, Xinyu Yan, Yizhan Fan, Xiangyu Wang, Yu Kang, Jian Cheng, and Huanhuan Chen. Reservoir-enhanced segment anything model for subsurface diagnosis. *Nature Communications*, 16(1):11080, 2025.

Appendix Outline

The appendix follows the order of the main text. Parts A–B back the method (proofs, datasets, experimental setup); Parts C–D extend the empirical evaluation (full few-shot tables, clinical-extreme analysis, mechanism details); Parts E–F collect robustness, efficiency, and supplementary figures.

Part A. Supporting analysis. Proof of Lemma 2.1 (§A); proof of Proposition 2.2 and the kernel view of RSS (§B, §B.1); ESP definitions, scope remark, corollary, and proof of Theorem 2.3 (§C).

Part B. Reproducibility. Dataset details (§D); experimental details including algorithm, hyperparameters, baselines, masking, normalization, and splits (§E).

Part C. Extended empirical evaluation. Full 17-method few-shot table with ±std (§F); few-shot k-sweep on P12/P19/PAM including AUPRC at k=10 (§G); full-data comparison on P12/P19/PAM (§H); per-channel coverage on the clinical benchmarks (§I); behaviour at the boundary of data scarcity (§J).

Part D. Mechanism details. Empirical (ρ, λ) landscape on Kuramoto (§K); full RSS component ablation (§L); attribution controls (§M).

Part E. Robustness and efficiency. Hyperparameter robustness (§N); reservoir-seed sensitivity (§O); reservoir-size sensitivity (§P); classifier robustness of RSS features (§Q); effect of training-set size k=100 (§R); extraction-time comparison (§S).

Part F. Supplementary figures. Robustness curves and raw-vs-reservoir comparison referenced from the main body (§T).

Part A: Supporting Analysis

Proofs, Kernel View, Operator Norm

A Proof of Lemma 2.1

Proof. The channel-j ridge solution is

$$W_{\text{out}}^{j\star} = (h_j^\top h_j + \gamma T_j I)^{-1} h_j^\top \mathbf{y}_j.$$

Let $h_j = U\Lambda V^\top$ be the SVD of h_j, with singular values $\sigma_1 \geq \cdots \geq \sigma_r > 0$ and $r = \operatorname{rank}(h_j) \leq \min(T_j, n)$. Substituting the SVD gives

$$W_{\text{out}}^{j\star} \;=\; \sum_{k=1}^{r} \frac{\sigma_k}{\sigma_k^2 + \gamma T_j}\, v_k\, u_k^\top \mathbf{y}_j,$$

where u_k, v_k are the left/right singular vectors. The remaining $n-r$ right singular vectors $\{v_k\}_{k=r+1}^{n}$ correspond to zero singular values and contribute $0/(0+\gamma T_j) = 0$ to the readout. Hence:

(a) $W_{\text{out}}^{j\star} \in \operatorname{span}\{v_1, \ldots, v_r\}$, the row space of h_j, whose dimension is at most T_j.

(b) When $T_j < n$, $r \leq T_j < n$, so at least $n - T_j$ singular directions $\{v_k\}_{k=r+1}^{n}$ lie in the null space of $h_j^\top h_j$. The corresponding components of $W_{\text{out}}^{j\star}$ are unidentified by the readout fit, regardless of any input–state relationship along those directions.

(c) The state summaries μ_j, σ_j, and h_{end}^j are bounded sums, standard deviations, and indexed values of the bounded reservoir states $h_i^j \in [-1, 1]^n$ (since $f = \tanh$). They require no matrix inversion and are well-defined for any $T_j \geq 1$. □

B Proof of Proposition 2.2 and kernel view

Remark B.1 (Implicit kernel perspective). The RSS representation defines an implicit positive-definite kernel $k_{\mathrm{RSS}}(\mathbf{x}, \mathbf{x}') = \langle \phi_{\mathrm{RSS}}(\mathbf{x}), \phi_{\mathrm{RSS}}(\mathbf{x}') \rangle$ on the space of irregular multivariate sequences, whose feature map is a structured random projection (via the random fixed reservoir W, W_{in}) that respects temporal ordering and the event-driven masking of irregular sampling. Unlike explicit sequence kernels such as dynamic time warping or global-alignment kernels [Cuturi, 2011], k_{RSS} operates directly on raw observations without imputation or time warping and produces a fixed-dimensional feature map whose length does not grow with the sequence. The kernel form is given in Sec. B.1.

Proof of Proposition 2.2. Since $|\alpha| < 1$ and x_t has finite second moment, the stable stationary solution of $h_t = \alpha h_{t-1} + \beta x_t$ is the absolutely convergent series

$$h_t \;=\; \beta \sum_{r=0}^{\infty} \alpha^r x_{t-r}.$$

Using zero mean and class-conditional weak stationarity,

$$\mathrm{Var}(h_t \mid c) \;=\; \beta^2 \sum_{j,k \geq 0} \alpha^{j+k}\, C_c(j-k),$$

with $C_c(-\ell) = C_c(\ell)$. Decompose the double sum by the lag $\ell = j - k$. The diagonal $j = k$ contributes $\sum_{k \geq 0} \alpha^{2k}\, C_c(0) \;=\; C_c(0)/(1-\alpha^2)$. For each positive lag $\ell \geq 1$, the terms with $j - k = \ell$ contribute $\alpha^\ell C_c(\ell)/(1-\alpha^2)$, and the terms with $k - j = \ell$ contribute the same quantity since $C_c(-\ell) = C_c(\ell)$. Adding the diagonal and the positive/negative-lag contributions yields

$$\nu_c \;=\; \frac{\beta^2}{1-\alpha^2}\Big(C_c(0) + 2\sum_{\ell \geq 1} \alpha^\ell C_c(\ell)\Big).$$

Subtracting the two class variances under $C_0(0) = C_1(0)$ gives

$$\nu_1 - \nu_0 \;=\; \frac{2\beta^2}{1-\alpha^2} \sum_{\ell \geq 1} \alpha^\ell \big(C_1(\ell) - C_0(\ell)\big) \;=\; \frac{2\beta^2}{1-\alpha^2}\, \Delta_\alpha,$$

which is nonzero whenever $\Delta_\alpha \neq 0$, while $\mathbb{E}[x_t]$ and $C_0(0) = C_1(0)$ leave the raw first and second moments equal across classes. □

Remark B.2 (Scope of this analysis). The proposition assumes a scalar linear reservoir, weak stationarity, and an absolutely convergent autocovariance series. The full GDRN has $\tanh$ nonlinearity, multivariate state, irregular time decay, mask-based event-driven updates, and global pooling, none of which are captured by this derivation. The proposition only identifies one case where lagged temporal information is recoverable from state variance and not from raw moments; Fig. 10 measures the full nonlinear method directly across all 18 main-table conditions.

B.1 Kernel view of RSS

Given the GDRN feature map $\phi_{\mathrm{RSS}} : \mathcal{X} \to \mathbb{R}^{4bn}$ on the space $\mathcal{X}$ of irregularly sampled multivariate sequences, define

$$\begin{aligned} k_{\mathrm{RSS}}(\mathbf{x}, \mathbf{x}') \;&=\; \langle \phi_{\mathrm{RSS}}(\mathbf{x}), \phi_{\mathrm{RSS}}(\mathbf{x}') \rangle \\ &=\; \sum_{j=1}^{b} \Big(\langle W_{\mathrm{out}}^{j\star}(\mathbf{x}), W_{\mathrm{out}}^{j\star}(\mathbf{x}') \rangle + \langle \mu_j(\mathbf{x}), \mu_j(\mathbf{x}') \rangle \\ &\qquad + \langle \sigma_j(\mathbf{x}), \sigma_j(\mathbf{x}') \rangle + \langle h_{\mathrm{end}}^j(\mathbf{x}), h_{\mathrm{end}}^j(\mathbf{x}') \rangle \Big). \end{aligned} \tag{8}$$

As a finite sum of inner products in $\mathbb{R}^n$, k_{RSS} is positive semi-definite by construction. Three properties matter for irregular multivariate inputs:

(a) *Asynchrony-compatible.* $k_{\mathrm{RSS}}(\mathbf{x}, \mathbf{x}')$ is well-defined for any pair $(\mathbf{x}, \mathbf{x}')$; the per-channel observation times of $\mathbf{x}$ and $\mathbf{x}'$ need not coincide, because ϕ_{RSS} is computed separately on each sequence through GDRN's event-driven dynamics.

(b) *Channel-additive.* The kernel decomposes into per-channel contributions with no explicit pairwise cross-channel term. This mirrors the per-channel readout design (Sec. 2.2), which was adopted precisely to avoid the cross-channel imputation required by sequence kernels that align channels on a common time grid.

(c) *Fixed-dimensional feature map.* Unlike dynamic time warping kernels or signature-kernel Gram matrices whose computation scales with sequence length, $\phi_{\mathrm{RSS}}(\mathbf{x}) \in \mathbb{R}^{4bn}$ has a dimension that does not depend on the sequence length T.

Random-feature view. Over the random draw of (W, W_{in}), ϕ_{RSS} is a random feature map in the sense of Rahimi and Recht [2007], but for a kernel implicit to the GDRN architecture rather than approximating a pre-specified one, matching the reservoir-as-random-feature view of Gonon and Ortega [2020]. Compared to random Fourier features on fixed-dimensional vectors, ϕ_{RSS} encodes temporal ordering through the recurrence and event-driven validity through the mask M_i.

C Echo state property: definitions, scope, and proof

C.1 Definitions

Definition C.1 (ESP)**.** The network satisfies the ESP if, for any bounded input $\{(t_i, \mathbf{u}_i, M_i)\}_{i\geq 1}$ and any two initial states $H_0, \hat{H}_0$, the resulting trajectories converge: $\lim_{i\to\infty} \|H_i - \hat{H}_i\|_F = 0$.

Definition C.2 (K-coverage)**.** An observation sequence $\{(t_i, M_i)\}_{i\geq 1}$ satisfies *K-coverage* if there exists a finite K such that every channel is updated at least once within any K consecutive events.

C.2 Scope and corollary

Remark C.3 (Scope of Theorem 2.3). Theorem 2.3 is a sufficient condition. The operator-norm form is loose for non-normal sparse W where $\rho(W) \ll \|W\|_2$; the spectral-radius initialization default $\rho(W)$=0.9 is the standard ESN convention [Yildiz et al., 2012], and trajectory boundedness $\|h_i^j\|_\infty \leq 1$ from $\tanh$ holds unconditionally. The boundedness, not the operator-norm bound, is what every empirical result in this paper relies on. Coverage (Definition C.2) is the stricter requirement to flag for practitioners: it holds on the synthetic, industrial, and activity datasets, partially on Volcano CAU ($\sim 56\%$ of station-event pairs are silent, but active stations carry dense waveform trajectories), and fails strictly on P12 (median T_j=3) and P19 (median T_j=2). Both sit in Lemma 2.1's rank-deficient regime ($T_j \ll n$=20), and RSS lifts the W-only readout on both (P12: $62.5 \to 72.7$; P19: $67.5 \to 72.2$ AUROC, Table 9). The rescued representation reaches the few-shot lead on P12; on P19 it trails StatsGap.

Remark C.4 (Decay relaxes the ESP operator-norm budget). Under Theorem 2.3, the sufficient condition $L_f\|W\|_2 e^{-\lambda\Delta t_{\min}^{\mathrm{all}}} < 1$ is equivalent to $\|W\|_2 < \frac{1}{L_f} e^{\lambda\,\Delta t_{\min}^{\mathrm{all}}}$, which exceeds the standard ESN bound $\|W\|_2 < 1/L_f$ by a factor of $e^{\lambda\Delta t_{\min}^{\mathrm{all}}}$. Read λ and the weight norm as memory- and dynamics-oriented knobs sharing this budget; an empirical (ρ, λ) landscape on Kuramoto with a performance plateau extending beyond $\rho(W)$=1 appears in Appendix K.

C.3 Proof of Theorem 2.3

Proof. Let $H_i, \hat{H}_i \in \mathbb{R}^{n\times b}$ be two GDRN trajectories under the same input. Define per-channel errors $e_i^{(j)} = H_i(:, j) - \hat{H}_i(:, j) \in \mathbb{R}^n$ and the max-channel error $m_i = \max_j \|e_i^{(j)}\|_2$.

Step 1: Decay bound (per channel). The decayed state at event i is $\widetilde{H}_i = H_{i-1}D_i$ with $D_i = \mathrm{diag}(e^{-\lambda\Delta t_i(j)})$. The per-channel decayed error is

$$\|\tilde{e}_i^{(j)}\|_2 = e^{-\lambda\Delta t_i(j)}\,\|e_{i-1}^{(j)}\|_2 \;\leq\; e^{-\lambda\Delta t_{\min}^{\mathrm{all}}}\, m_{i-1}.$$

Step 2: Pooling bound. The pooled context $p_i = \frac{1}{b}\widetilde{H}_i\mathbf{1} \in \mathbb{R}^n$ satisfies

$$\|p_i - \hat{p}_i\|_2 \;=\; \Big\|\tfrac{1}{b}\sum_{j=1}^{b} \tilde{e}_i^{(j)}\Big\|_2 \;\leq\; \tfrac{1}{b}\sum_{j=1}^{b}\|\tilde{e}_i^{(j)}\|_2 \;\leq\; e^{-\lambda\Delta t_{\min}^{\mathrm{all}}}\, m_{i-1}.$$

The broadcast structure $S_i = (Wp_i)\mathbf{1}^\top$ implies that every column of S_i equals Wp_i, so for each j,

$$\|S_i^{(j)} - \hat{S}_i^{(j)}\|_2 \;=\; \|W(p_i - \hat{p}_i)\|_2 \;\le\; \|W\|_2\, e^{-\lambda \Delta t_{\min}^{\text{all}}}\, m_{i-1}.$$

Step 3: Per-channel contraction on updated channels. For channel j updated at event i ($M_i(j,j) = 1$), the input terms cancel in the difference and Lipschitz continuity of f gives

$$\|e_i^{(j)}\|_2 \;\le\; L_f\, \|S_i^{(j)} - \hat{S}_i^{(j)}\|_2 \;\le\; \underbrace{L_f \|W\|_2 e^{-\lambda \Delta t_{\min}^{\text{all}}}}_{\kappa < 1}\, m_{i-1}.$$

For channel j *not* updated at event i ($M_i(j,j) = 0$), the stored state is preserved: $\|e_i^{(j)}\|_2 = \|e_{i-1}^{(j)}\|_2 \le m_{i-1}$. Therefore $m_i \le m_{i-1}$, with strict contraction κ on every updated channel. No $\sqrt{b}$ factor enters: the pooling acts as an average (which contracts), and the broadcast preserves the per-column bound, so the per-channel L2 norm bound is tight.

Step 4: Convergence via coverage. By the coverage condition, within any K consecutive events every channel is updated at least once. After K events from any starting point t, each channel has been contracted by at least one factor of κ, so $m_{t+K} \le \kappa\, m_t$. Iterating: $m_{t+qK} \le \kappa^q\, m_t \to 0$. Since $\|H_i - \hat{H}_i\|_F \le \sqrt{b}\, m_i$, we conclude $\|H_i - \hat{H}_i\|_F \to 0$. □

Remark C.5 (Coverage assumption). The coverage condition requires each channel to be observed at least once within any K consecutive events. For a permanently unobserved channel, the stored state $H_i^{(j)}$ does not change in the update (3), but its *contribution to the pooled context* S_i decays exponentially via $\exp(-\lambda(t_i - \ell_{i-1}(j))) \to 0$, so the coupled dynamics become dominated by active channels. In practice, the reservoir remains bounded even when coverage is violated, though classification performance may degrade.

Part B: Reproducibility

Datasets and Experimental Details

D Dataset details

Table 2: Overview of the nine evaluation datasets. Obs% is the inherent fraction of non-missing entries before artificial masking, and "Irregular" marks naturally non-uniform timestamps (✓) or artificial masks (∼). Median T_j and $\%(T_j{<}n)$ are per-channel observation counts at the main-table condition (MCAR20 for synthetic, natural for irregular), with $n{=}20$. †Volcano CAU is bimodal: 56% of (event, station) pairs are silent ($T_j{=}0$), and the rest carry full waveforms ($T_j{=}512$).

Dataset	Domain	N	T	F	Classes	Obs%	Irreg.	Med. T_j	$\%(T_j{<}n)$
Lorenz-96	Synthetic (chaotic)	300	1024	36	3 (forcing)	100%	∼	205	0%
Kuramoto	Synthetic (oscillator)	4500	500	16	3 (coupling)	100%	∼	100	0%
CML	Synthetic (map lattice)	4500	500	20	3 (regimes)	100%	∼	100	0%
TEP	Industrial (chemical)	1972	60	52	19 (faults)	100%	∼	30	<1%
P12	ICU mortality	11,988	215	36	2 (mortality)	5.2%	✓	3	83%
P19	ICU sepsis	38,803	60	34	2 (sepsis)	11.6%	✓	2	86%
PAM	Wearable activity	5333	600	17	8 (activities)	23.1%	∼	137	25%
Person Act.	RFID activity	2244	40	12	5 (activities)	37.4%	✓	17	100%
Volcano CAU	Seismic events	7212	512	8	3 (event types)	43.7%	✓	0†	56%

D.1 Synthetic datasets

Lorenz-96 [Lorenz, 1996]. A 36-dimensional chaotic dynamical system with three forcing values ($F_{\text{force}} \in \{4, 6, 8\}$) as class labels. Trajectories are integrated over 100 time units ($\Delta t{=}0.01$) with

Gaussian noise (std 0.5). N=300 (100 per class), T=1024, F=36. Irregularity is introduced by artificial masking.

Kuramoto [Kuramoto, 1984]. A 16-dimensional system of coupled phase oscillators. Classes correspond to coupling strengths $K \in \{0.5, 2.0, 5.0\}$, which produce distinct synchronization regimes. N=4500 (1,500 per class), T=500, F=16.

CML (Coupled Map Lattice) [Kaneko, 1989]. A 20-dimensional coupled map lattice with three class-defining dynamical regimes. N=4500 (1,500 per class), T=500, F=20.

D.2 Real-world datasets

The six real-world datasets cover industrial process control, ICU clinical monitoring, wearable activity recognition, and seismic event classification. P12, P19, and PAM are taken in their preprocessed form from Zhang et al. [2022], with the same five train/val/test splits as that and subsequent work [Che et al., 2018, Shukla and Marlin, 2021, Luo et al., 2025, Lee et al., 2025]. TEP, Person Activity, and Volcano CAU are processed from their original public sources as described per dataset below.

TEP (Tennessee Eastman Process) [Downs and Vogel, 1993]. A chemical process simulation with 52 variables (41 measured plus 11 manipulated). Faults 3, 9, and 15 (widely reported as noise-dominated and unclassifiable) are excluded, giving a 19-class problem. N=1972, T=60, F=52.

P12 (PhysioNet 2012, ICU Mortality) [Goldberger et al., 2000]. ICU patient records with 36 clinical variables measured at irregular intervals over a 48-hour window. The task is binary prediction of in-hospital mortality (N=11,988, T=215, F=36, obs rate 5.2%, positive rate ≈14%). With n=20 reservoir dimensions and a median of about 3 observations per channel, this is the most underdetermined setting in our evaluation for readout-based methods.

P19 (PhysioNet 2019, Sepsis Prediction) [Reyna et al., 2020]. ICU encounters with 34 clinical variables sampled irregularly over up to 60 hours. The task is binary prediction of sepsis onset (N=38,803, T=60, F=34, obs rate 11.6%, positive rate ≈4.5%). Median per-channel observation count is $T_j \approx 2$, the most underdetermined setting in our evaluation alongside P12.

PAM (PAMAP2 Activity Monitoring) [Reiss and Stricker, 2012]. Wearable-sensor recordings from 9 subjects carrying 3 IMUs (wrist, chest, ankle), producing 17 sensor channels at 100 Hz. Following the Raindrop preprocessing, the data yield N=5,333 samples across 8 activity classes (T=600, F=17, obs rate 23.1%).

Person Activity [Kaluža et al., 2010]. RFID-based activity recognition from the UCI Machine Learning Repository, using 4 body tags with 3 coordinates each (12 channels) across 5 activity classes. RFID sensors report at non-uniform event-driven intervals, so the data are naturally irregular. N=2244, T=40, F=12, obs rate 37.4%.

Volcano CAU (Puyehue-Cordón Caulle) [Espinosa-Curilem et al., 2026]. Chilean volcano seismic events recorded by 8 broadband stations, with three event classes: volcano-tectonic (VT, 2298 events), long-period (LP, 2081), and tremor (TR, 2833). Each event is detected by only a subset of stations (VT: avg. 2.5; LP: 3.5; TR: 4.0 active out of 8), so the data are naturally irregular at the per-station level. N=7212, T=512, F=8, obs rate 43.7%.

Across the suite, the smallest dataset is Lorenz-96 (N=300) and the largest is P19 (N=38,803); sequence lengths span T=40 (Person Activity) to T=1024 (Lorenz-96); channel counts span F=8 (Volcano CAU) to F=52 (TEP); class counts span binary to 19-class fault classification. Observation rates span more than an order of magnitude, from 5.2% (P12) to 100% (fully observed synthetic and industrial trajectories prior to artificial masking). Four datasets are naturally irregular; five are synthetic or regularly sampled and receive artificial missingness via MCAR, burst, or event-triggered mechanisms (Appendix E.4).

E Experimental details

E.1 GDRN-RSS algorithm

Algorithm 1 summarizes the combined forward pass, readout fit, and RSS feature assembly described in Sections 2.1–2.3.

Algorithm 1 GDRN-RSS: Forward pass and feature extraction

Input: Sequence $\{(t_i, \mathbf{u}_i, M_i)\}_{i=1}^{T}$; reservoir W, W_{in}; decay λ; ridge γ.
Output: RSS representation $\phi_{\text{RSS}} \in \mathbb{R}^{4bn}$.

Init $H_0 \leftarrow \mathbf{0}$, $\ell_0 \leftarrow \mathbf{0}$, accumulators for μ, σ, terminal-index store.
for $i = 1$ **to** T **do**
  Decay: $\widetilde{H}_i \leftarrow H_{i-1}\, D_i$ {Eq. (1)}
  Pool: $S_i \leftarrow (W \cdot \frac{1}{b}\widetilde{H}_i \mathbf{1})\mathbf{1}^\top$ {Eq. (2)}
  Update: $H_i \leftarrow H_{i-1}(I - M_i) + f(S_i + W_{\text{in}}\text{diag}(\mathbf{u}_i))M_i$
  Update ℓ_i; accumulate h_i^j for observed channels; record latest observation index per channel.
end for
for $j = 1$ **to** b **do**
  if $T_j = 0$ **then**
    $W_{\text{out}}^{j\star} \leftarrow \mathbf{0}$, $\mu_j \leftarrow \mathbf{0}$, $\sigma_j \leftarrow \mathbf{0}$, $h_{\text{end}}^j \leftarrow \mathbf{0}$ {never-observed channel default}
  else
    Fit readout: $W_{\text{out}}^{j\star} \leftarrow (h_j^\top h_j + \gamma T_j I)^{-1} h_j^\top \mathbf{y}_j$.
    Compute: $\mu_j \leftarrow \frac{1}{T_j}\sum_{i \in \mathcal{I}_j} h_i^j$, $\sigma_j \leftarrow \text{std}(\{h_i^j\}_{i \in \mathcal{I}_j})$, $h_{\text{end}}^j \leftarrow h_{\max(\mathcal{I}_j)}^j$.
  end if
end for
return $\phi_{\text{RSS}} = [W_{\text{out}}^{1\star}; \mu_1; \sigma_1; h_{\text{end}}^1; \ldots; W_{\text{out}}^{b\star}; \mu_b; \sigma_b; h_{\text{end}}^b]$.

E.2 GDRN hyperparameters

All experiments use the following default configuration unless stated otherwise:

- Reservoir dimension: $n = 20$
- Spectral radius: $\rho = 0.9$
- Decay rate: $\lambda = 0.1$
- Sparsity: 0.5 (fraction of zero weights in W)
- Ridge parameter: $\gamma = 10^{-4}$
- Activation: $f = \tanh$
- Classifier: Random Forest with 300 trees

Hardware. All experiments were run on a single AutoDL instance: Intel Xeon Platinum 8470Q CPU (25 cores allocated), 90 GB RAM, 1× NVIDIA RTX 5090 (32 GB VRAM). Software: Ubuntu 22.04, CUDA 13.0, PyTorch 2.8.0, Python 3.12. GDRN-RSS runs on CPU only; deep-learning baselines (GRU-D, Latent ODE, Raindrop, mTAND, ATENet, Warpformer, Hi-Patch, PrimeNet, ViTST) use the RTX 5090.

E.3 Baseline descriptions and implementation details

Per-baseline configurations and preprocessing follow the seven categories of §3.2.

(1) RF-stats (Simple Statistics). Per-channel summary statistics (mean, standard deviation, minimum, maximum, and observation rate) are computed for each sequence, yielding a $5F$-dimensional feature vector. A Random Forest classifier (300 trees, default scikit-learn settings) is trained on these features.

(2) StatsGap (Simple Statistics with Inter-Event Gaps). Extends RF-stats with per-channel inter-event-gap statistics (mean inter-event gap, gap standard deviation), yielding $7F$ features. Random Forest classifier (300 trees, default scikit-learn settings).

(3) catch22 [Lubba et al., 2019] (Canonical Time-Series Features). Twenty-two canonical time-series features computed per channel, yielding $22F$ features. Computed on each channel's observed values. Implementation: `pycatch22`; Random Forest classifier (300 trees).

(4) MiniROCKET [Dempster et al., 2021] (Random Convolutional Kernels). A fast, (nearly) deterministic convolutional kernel transform with 10,000 random kernels. Missing values are forward-filled before transformation, as MiniROCKET requires complete data. The classifier is RidgeClassifierCV with $\alpha \in \{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$ selected by leave-one-out cross-validation. Implementation: `aeon` library.

(5) MultiROCKET [Tan et al., 2022] (Random Convolutional Kernels, Multi-Pooling). Extension of MiniROCKET with n_{kernels}=10,000 and four pooling functions (PPV, MPV, MIPV, LSPV), yielding $\sim$50,000 features per sequence. Missing values are forward-filled before transformation. Implementation: `aeon` library; classifier is RidgeClassifierCV with $\alpha \in \{10^{-3}, 10^{-2}, \ldots, 10^{3}\}$ selected by leave-one-out cross-validation.

(6) GRU-D [Che et al., 2018] (RNN with Decay). GRU augmented with trainable time-decay and input-decay mechanisms for handling missing values. Architecture: single-layer GRU with hidden size 64, followed by a linear classifier. Training: Adam optimizer (lr $= 10^{-3}$), batch size 128, 50 epochs with early stopping on validation AUROC or accuracy. Dropout 0.3 on the GRU output.

(7) Latent ODE [Rubanova et al., 2019] (Neural ODE). ODE-RNN encoder maps irregularly-sampled observations into a latent state, which is then integrated forward via a learned ODE and decoded for classification. PhysioNet recipe: `rec_dims`=40, `rec_layers`=3, `gen_layers`=3, `units`=50, `latents`=20, Adamax (lr $= 10^{-2}$, decay 0.999), Euler solver, stride-4 time subsampling on P12/P19 full-data.

(8) Raindrop [Zhang et al., 2022] (Graph-Guided Network). Graph-guided network that learns sensor-level dependencies through dynamic graphs. Architecture: 2 transformer layers, hidden size 64, 4 attention heads, learned inter-sensor graphs. Training: Adam (lr $= 10^{-3}$), batch size 128, 100 epochs with early stopping.

(9) mTAND [Shukla and Marlin, 2021] (Multi-Time Attention). Multi-time attention network with learned time embeddings and cross-time attention. Architecture: hidden size 64, 2 attention heads, 128 reference points. Training: Adam (lr $= 10^{-3}$), batch size 128, 100 epochs with early stopping.

(10) ATENet [Lee et al., 2025] (Adaptive Time Encoding). Adaptive time encoding network with per-channel time encodings. Architecture: hidden size 64, 2 transformer layers, adaptive time encoding module. Training: Adam (lr $= 10^{-3}$), batch size 128, 100 epochs with early stopping.

(11) Warpformer [Zhang et al., 2023] (Warping Transformer). Multi-scale transformer with warping-based attention that adapts to irregular sampling intervals. Architecture: 2 encoder layers, hidden size 64, 4 attention heads, 3 warping scales. Training: Adam (lr $= 10^{-3}$), batch size 128, 100 epochs with early stopping.

(12) Hi-Patch [Luo et al., 2025] (Hierarchical Patching). Hierarchical patch-based tokenization with variable-length patches. We use the authors' recipe: n_{layer}=3 transformer layers, hidden size 64, `patch_size`$\approx T/4$. Training: Adam (lr $= 10^{-3}$), batch size 128, 100 epochs with early stopping.

(13) PrimeNet [Chowdhury et al., 2023] (Self-Supervised Pre-training). Self-supervised contrastive pre-training for irregular multivariate time series. A time-aware Transformer encoder (Time-BERT) is followed by an MLP classification head. Implementation: official codebase.

(14) ViTST [Li et al., 2023] (Vision Transformer for Time Series). Converts each time series into a line-graph image and classifies it using `google/vit-base-patch16-224` (ImageNet-21k pre-trained). Hyperparameters: lr=2×10^{-5}, batch 16, 20 epochs with early stopping.

(15) Ct-Echo [Chen et al., 2025] (Continuous-Time Echo State). Per-channel continuous-time leaky echo state network for irregular sequences, with the per-channel ridge readout protocol used by GDRN-RSS. Configuration: n=20, ρ=0.9, λ=0.1, γ=10^{-4}, sparsity 0.5, Random Forest with 300 trees on the readout vector.

(16) ESN-MS [Chen et al., 2014] (Discrete-Time Echo State Model-Space). Standard echo state network model-space without irregular-sequence-specific design. Configuration matches the GDRN default (n=20, ρ=0.9, γ=10^{-4}, sparsity 0.5, Random Forest with 300 trees).

Implementation sources. RF-stats, StatsGap, catch22, MiniROCKET, and MultiROCKET use `scikit-learn` and the `aeon` library. GRU-D: `https://github.com/zhiyongc/GRU-D`. Latent ODE: `https://github.com/YuliaRubanova/latent_ode`. Raindrop: `https://github.com/mims-harvard/Raindrop`. mTAND: `https://github.com/reml-lab/mTAN`. ATENet: `https://github.com/shlee-labs/ATENet`. Warpformer: `https://github.com/imJiawen/Warpformer`. Hi-Patch: `https://github.com/qianlima-lab/Hi-Patch`. PrimeNet: `https://github.com/ranakroychowdhury/PrimeNet`. ViTST: `https://github.com/Leezekun/ViTST`.

E.4 Missing data mechanisms

Table 1 evaluates robustness under three artificial masking mechanisms, in addition to naturally irregular datasets:

MCAR (Missing Completely At Random). Each observation is independently dropped with probability p. This is the simplest mechanism and the most commonly studied in the literature; $p \in \{0.5, 0.8, 0.95\}$ depending on the dataset.

Burst Missingness. Contiguous blocks of observations are removed, simulating sensor outages. Each time step initiates a burst (of geometrically distributed length) with probability p_{burst}; $p_{\text{burst}} = 0.02$ on Lorenz-96 produces bursts that remove $\sim$29% of observations on average.

Event-Triggered Missingness. Observations are retained only when the absolute change exceeds a threshold δ, set as a fraction of the per-channel standard deviation. This simulates event-driven sensors that report only significant changes, producing highly non-uniform sampling; $\delta_{\text{ratio}} = 0.1$ retains 8–23% of observations depending on the dataset's temporal dynamics.

For naturally irregular datasets (Person Activity, Volcano CAU), additional MCAR masking is applied *on top of* the existing observation pattern, further reducing the observation rate.

E.5 Data normalization

All methods use the same normalization. For k-shot experiments, z-score normalization is fitted on the k-shot training subset only (per-channel mean and standard deviation), then applied to the test set. This prevents information leakage from the test set. For full-data experiments, normalization is fitted on the full training set.

E.6 Train/test splits

For P12, P19, and PAM we use the official 5 train/val/test splits released with Zhang et al. [2022], also used by Luo et al. [2025] and other deep baselines in the comparison. For all other datasets, k-shot subsamples are drawn uniformly at random per class with a fixed seed; the unit of sampling is the natural sample of each dataset (one independent simulated trajectory for Lorenz/Kuramoto/CML, one fault-run window for TEP, one 2-second activity window for Person Activity, one seismic event for Volcano CAU). The same split is applied to every baseline on a given dataset and condition.

Part C: Extended Empirical Evaluation

Full Few-Shot Tables, Clinical Extreme

F Full few-shot main table

Table 3 reports the same 17 methods as main Table 1 on 22 conditions, adding the four fully observed ones, with explicit $\pm$std.

Table 3: Classification under different missing-data patterns (k-shot, 5 seeds): full 17-method × 22-condition table with ±std. Bold = best per row, underline = second.

		Feature-based					RNN	ODE/Graph		Attention/Transformer				Pre-trained		Model-space learning		
Dataset	**Mask (obs%)**	**RF-stats**	**StatsGap**	**catch22**	**MiniR**	**MultiR**	**GRU-D**	**L-ODE**	**Raindrop**	**mTAND**	**ATENet**	**Warpf.**	**Hi-Patch**	**PrimeNet**	**ViTST**	**ESN-MS**	**Ct-Echo**	**RSS (ours)**
Lorenz	Full (100%)	**100.0**±0.0	**100.0**±0.0	80.3±10.7	81.6±9.1	86.0±7.3	31.3±2.8	35.2±3.5	33.3±0.0	89.8±7.1	52.7±15.8	40.0±13.3	—	71.1±20.4	97.8±1.9	73.5±3.6	98.4±0.8	**100.0**±0.0
	MCAR (20%)	98.0±1.1	96.2±3.7	43.7±4.6	79.2±9.4	80.4±9.6	34.3±1.8	33.5±2.1	33.3±0.0	63.2±20.1	53.4±14.2	39.6±12.5	—	46.8±16.8	90.0±4.7	35.4±1.9	80.6±4.0	**100.0**±0.0
	Burst (71%)	99.5±0.8	98.7±1.2	69.8±9.3	79.1±4.9	80.8±7.3	32.3±3.6	35.4±2.1	33.3±0.0	70.5±20.5	56.3±12.1	40.0±13.3	—	53.8±15.5	86.8±7.8	63.2±2.8	91.2±4.4	**100.0**±0.0
	Event (23%)	**100.0**±0.0	**100.0**±0.0	74.5±6.6	81.5±8.5	85.5±7.3	34.4±2.6	33.8±2.1	33.3±0.0	60.8±21.5	57.7±12.4	39.9±13.2	—	60.1±17.3	97.6±1.0	39.0±3.1	94.0±1.7	**100.0**±0.0
Kuramoto	Full (100%)	62.5±3.3	62.9±3.9	80.3±2.1	75.7±3.6	83.6±3.8	34.2±5.2	42.1±5.2	33.3±0.0	54.8±2.6	37.3±5.1	48.4±10.2	—	40.1±6.9	86.5±3.3	76.4±3.4	74.6±3.4	**91.7**±2.4
	MCAR (20%)	56.4±3.6	56.2±2.7	79.1±2.1	75.0±3.7	80.0±2.5	33.7±0.5	35.9±2.5	33.3±0.0	51.4±7.1	37.6±5.0	43.3±8.7	59.5±19.4	36.9±3.0	86.2±2.3	72.0±2.2	72.7±3.8	**91.0**±2.2
	Event (8%)	72.8±4.9	72.8±4.4	72.7±4.6	76.6±2.8	82.0±1.3	33.8±7.9	44.6±10.8	38.1±8.7	53.1±3.0	38.1±11.5	55.0±9.4	47.9±9.5	38.3±2.9	**86.4**±3.0	63.9±7.7	77.9±5.2	76.1±3.7
CML	Full (100%)	98.4±1.0	98.5±1.0	96.4±1.2	**99.3**±0.2	98.9±0.3	53.4±8.0	64.5±12.2	33.3±0.0	64.9±2.7	50.8±14.6	45.9±15.5	—	39.7±12.7	97.8±1.8	98.0±1.7	98.0±1.6	**99.3**±0.2
	MCAR (20%)	96.7±1.0	96.9±1.0	89.5±7.5	95.8±0.8	97.0±0.9	32.6±1.1	39.2±2.7	33.3±0.0	51.1±8.5	45.9±10.7	40.0±13.3	43.5±14.5	34.3±1.9	98.1±1.4	43.0±2.2	95.2±1.5	**99.3**±0.2
	MCAR (5%)	**95.0**±1.6	94.9±2.0	73.7±6.9	82.6±1.8	87.7±1.4	33.7±0.4	34.2±1.3	33.4±0.0	37.6±5.9	39.4±7.1	41.7±10.9	40.8±7.9	39.1±8.4	**95.0**±2.6	36.0±0.8	80.3±4.0	93.7±3.2
TEP	Full (100%)	87.7±2.0	87.6±1.5	64.4±1.9	79.4±2.2	77.2±1.8	29.8±1.5	66.6±4.2	5.5±0.8	36.5±6.4	37.1±8.3	38.5±18.4	77.7±3.2	17.6±6.2	60.3±3.3	83.1±2.3	88.8±2.8	**91.0**±1.5
	MCAR (50%)	85.8±2.7	86.6±2.3	51.5±1.3	72.1±2.5	71.4±2.2	24.9±6.3	55.9±2.0	5.5±0.8	33.5±8.2	40.5±2.1	44.4±5.9	49.1±4.0	25.1±2.6	53.9±1.5	66.9±3.8	85.8±3.6	**90.1**±1.1
	Event (11%)	79.0±1.3	79.2±1.0	51.3±1.7	66.1±2.7	68.7±2.3	5.2±1.3	47.7±0.9	9.7±0.3	36.3±7.1	39.2±1.9	23.8±9.7	48.7±2.4	23.2±4.4	49.7±1.3	54.3±0.4	75.5±1.5	**80.4**±1.4
Person Activity	Natural (37%)	85.9±1.1	85.9±1.7	33.3±4.1	73.3±2.3	69.3±2.7	41.4±6.9	45.7±5.2	19.8±13.0	63.3±14.7	60.2±18.1	52.8±28.1	72.4±7.6	48.2±11.6	37.5±4.2	50.6±4.9	77.4±1.2	**87.2**±1.1
	+MCAR (19%)	79.9±3.7	78.1±5.1	36.3±6.8	58.9±2.2	60.6±2.5	16.9±5.1	28.5±4.6	18.3±12.4	54.0±4.3	39.3±15.6	18.4±12.1	38.1±14.5	20.0±7.8	37.3±5.6	44.2±4.6	69.0±0.9	**86.5**±1.0
	+MCAR (7.5%)	73.6±5.2	71.9±5.2	23.2±4.1	46.1±1.4	48.4±1.4	20.4±8.3	22.1±2.3	18.5±12.4	42.3±2.4	26.5±8.0	20.8±11.7	29.8±7.0	17.1±5.8	25.2±6.1	46.6±1.1	65.5±1.7	**80.1**±0.7
Volcano CAU	Natural (44%)	63.2±4.0	62.4±4.0	58.0±3.7	63.1±4.6	58.9±3.5	37.2±4.2	53.2±1.1	32.1±3.8	31.4±3.6	34.0±4.4	34.7±4.6	45.2±5.5	40.0±5.1	64.5±6.2	61.7±1.7	**65.0**±4.0	64.0±3.5
	+MCAR (22%)	61.8±3.7	59.6±2.0	52.3±1.0	62.4±4.9	59.6±4.8	35.4±2.3	34.2±2.5	34.9±3.6	31.5±4.1	34.5±4.3	35.1±5.1	49.7±13.4	30.1±1.4	63.5±4.4	55.9±2.3	62.8±3.9	**64.9**±4.1
	+MCAR (9%)	61.1±3.1	59.0±1.2	51.8±0.8	61.4±4.3	59.8±3.9	31.4±3.8	34.1±2.4	34.8±3.6	31.8±3.9	34.4±4.1	37.0±4.1	43.8±11.7	30.3±1.4	60.3±3.5	56.5±1.4	60.8±2.6	**63.4**±3.7
Std. bench. (k=10)	P12 (5%, AUROC)	68.9±2.9	65.6±2.8	58.5±3.0	67.3±3.1	67.2±3.2	63.9±2.7	59.7±4.7	58.7±2.4	64.7±6.9	62.7±5.5	64.5±1.7	54.9±7.0	68.6±2.1	63.5±4.6	56.5±3.3	62.0±1.8	**72.7**±2.3
	P19 (12%, AUROC)	75.2±4.3	**77.8**±1.9	73.8±10.1	75.7±1.5	74.7±4.4	70.3±7.9	61.2±3.8	70.4±9.4	62.0±3.4	60.6±3.7	74.2±4.6	61.0±5.2	63.8±2.9	68.5±6.4	69.6±3.2	72.6±3.0	72.2±3.6
	PAM (23%, Acc)	77.8±2.5	78.0±2.1	72.0±3.5	**80.0**±1.4	77.2±1.8	37.7±3.2	52.7±3.3	19.8±14.8	70.8±1.8	66.8±2.2	31.5±5.4	23.5±4.5	59.4±6.2	54.8±3.8	69.3±1.0	71.6±2.0	77.2±2.6

G Few-shot classification on P12, P19, and PAM

P12 and P19 sit at the extreme low end of per-channel coverage (Appendix I), so their results should be read alongside the scope of Theorem 2.3 and the boundedness argument behind Lemma 2.1 rather than as a test of Proposition 2.2, which assumes enough reservoir steps per channel to stabilise.

The protocol samples k examples per class from the training set (split 1) and evaluates on the full test set across 5 random seeds; all 17 methods share the same subsamples and seeds.

Key findings. (i) *Tabular features lead P19 at every* $k \geq 10$. StatsGap AUROC climbs $71.1 \rightarrow 77.8 \rightarrow 82.2 \rightarrow 85.4 \rightarrow 87.3$ as k grows from 5 to 100; RF-stats trails by 0.8–6.5 points across the same range. Both lead the DL baselines and RSS at every $k \geq 10$, consistent with the coverage-condition scope (Remark C.3) at the rank-deficient extreme.

(ii) On P12, RSS leads at every tested k. The closest competitors are PrimeNet at $k=5$, RF-stats at $k=10$, and mTAND from $k=20$ on, which comes within 0.8 and 0.5 points at $k=50$ and $k=100$.

(iii) Most supervised DL baselines drop substantially under label scarcity; on PAM at $k=5$, Raindrop achieves only 8.5% (vs. 71.8% for RSS). Pre-trained methods (PrimeNet, ViTST) fare better than supervised DL but still trail RSS on PAM by 11–18 points at $k=5$.

(iv) Without any gradient-based training, RSS stays within 3.4 points of the PAM leader at every k and leads the comparison on P12 at every k. On P19 the tabular features remain ahead across the k-sweep, consistent with the per-channel coverage being low enough (median $T_j=2$) that trajectory-based methods are limited (Remark C.3).

(v) Our reading of the two datasets where a feature baseline stays ahead is as follows. P19 has a median of two observations per channel, and the gap-aware tabular statistics record which variables were measured and at what level. PAM is densely observed (median $T_j=137$), and its classes differ mainly in per-channel intensity profiles, which suits MiniROCKET. The gap is 2.8 points at $k=10$, and RSS leads at $k=20$ (83.6 vs. 83.5).

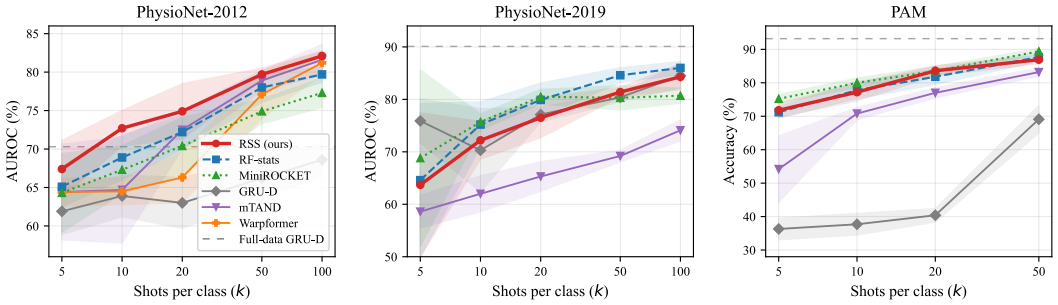


Figure 5: Few-shot learning curves on P12, P19, and PAM. RSS (red) approaches the full-data GRU-D reference at $k=50$–100 labeled samples per class.

G.1 P12 / P19 AUPRC at $k=10$

Because P19 has a 4.5% positive rate, AUROC alone is not a reliable summary of clinical predictive quality. Table 5 adds AUPRC on P12 and P19 at $k=10$ for six methods: RF-stats, MiniROCKET, MultiROCKET, catch22, StatsGap, and RSS. StatsGap extends RF-stats with two per-channel time-gap statistics (mean and standard deviation of inter-event intervals); catch22 [Lubba et al., 2019] uses the canonical 22 features per channel; MultiROCKET [Tan et al., 2022] uses ∼10K convolutional kernels on forward-filled input.

P12 vs. P19 split. On P12, the terminal state h_{end} recovers signal from the rank-deficient readout, lifting RSS to 72.7 AUROC / 29.8 AUPRC, ahead of every non-DL baseline. On P19 the tabular features take the lead, with StatsGap reaching 77.8 AUROC / 31.0 AUPRC and RF-stats close behind at 75.2 / 25.4. The split is consistent with per-channel coverage: P12's median $T_j=3$ leaves enough trajectory for h_{end} to integrate, while P19's median $T_j=2$ leaves the reservoir with too little to summarize, the rank-deficient extreme of Lemma 2.1's regime.

Table 4: Few-shot classification on P12, P19, and PAM. P12 and P19 report AUROC (%); PAM reports Accuracy (%). All 17 methods share the same protocol. Bold = best per row, underline = second.

		Feature-based					RNN	ODE/Graph		Attention/Transformer				Pre-trained		Model-space learning		
Data	k	**RF-stats**	**StatsGap**	**catch22**	**MiniR**	**MultiR**	**GRU-D**	**L-ODE**	**Raindrop**	**mTAND**	**ATENet**	**Warpformer**	**Hi-Patch**	**PrimeNet**	**ViTST**	**ESN-MS**	**Ct-Echo**	**RSS (ours)**
P12	5	65.1±2.6	60.6±5.2	55.9±2.3	64.3±5.1	58.6±6.2	61.9±3.0	51.6±4.3	56.9±4.0	64.4±6.2	59.4±6.4	64.4±1.8	54.6±8.7	66.3±3.9	58.6±7.7	56.4±1.3	59.5±3.1	**67.4**±3.7
	10	68.9±2.9	65.6±2.8	58.5±3.0	67.3±3.1	67.2±3.2	63.9±2.7	59.7±4.7	58.7±2.4	64.7±6.9	62.7±5.5	64.5±1.7	54.9±7.0	68.6±2.1	63.5±4.6	56.5±3.3	62.0±1.8	**72.7**±2.3
	20	72.2±2.2	67.2±4.0	59.0±4.5	70.4±3.7	71.1±1.7	63.0±3.3	59.0±4.2	58.7±2.0	72.5±2.5	72.2±3.2	66.3±3.3	61.5±5.2	71.6±2.3	65.2±2.9	57.7±2.3	64.7±3.6	**74.9**±3.6
	50	78.0±1.9	75.5±3.6	65.2±2.9	74.9±1.6	76.3±1.8	66.2±2.6	68.0±2.7	60.6±2.7	78.9±1.3	75.3±3.0	77.1±3.1	69.5±3.8	77.6±0.5	67.3±3.4	62.7±1.5	70.5±1.4	**79.7**±0.8
	100	79.7±1.0	78.7±1.2	70.2±2.6	77.3±1.9	76.5±2.6	68.6±2.1	71.9±1.5	67.4±4.1	81.6±2.0	78.1±3.6	81.2±1.5	73.8±2.5	81.1±1.6	72.0±0.5	64.0±1.3	72.1±0.4	**82.1**±0.6
P19	5	64.6±14.6	71.1±13.1	67.9±17.5	68.8±16.8	64.9±12.9	**75.9**±4.2	63.5±8.0	58.1±21.5	58.6±3.1	58.6±2.9	57.7±9.1	58.2±3.8	61.8±5.1	63.2±12.0	58.3±14.8	65.4±7.9	63.7±13.6
	10	75.2±4.3	**77.8**±1.9	73.8±10.1	75.7±1.5	74.7±4.4	70.3±7.9	61.2±3.8	70.4±9.4	62.0±3.4	60.6±3.7	74.2±4.6	61.0±5.2	63.8±2.9	68.5±6.4	69.6±3.2	72.6±3.0	72.2±3.6
	20	79.9±3.2	**82.2**±1.6	80.9±1.1	80.5±1.4	80.3±3.1	77.1±1.4	67.7±3.9	77.4±3.3	65.3±2.8	64.0±2.2	74.8±7.7	69.7±7.0	69.1±2.0	76.0±2.8	72.3±1.8	76.3±3.6	76.5±3.8
	50	84.6±1.4	**85.4**±2.0	82.2±0.9	80.3±2.3	83.0±1.6	80.4±0.8	66.8±5.6	78.2±3.2	69.2±1.1	68.3±1.1	80.7±3.3	80.0±1.6	71.4±3.1	80.9±0.7	73.7±1.1	80.6±1.6	81.4±0.9
	100	86.0±1.2	**87.3**±0.7	83.3±0.5	80.7±1.4	82.2±1.0	84.5±2.3	70.1±6.0	79.6±1.3	74.1±2.0	70.9±2.6	83.8±1.1	81.8±1.6	75.7±1.8	82.9±1.8	76.1±0.9	82.1±1.2	84.3±2.4
PAM	5	71.2±1.7	71.5±2.2	65.8±1.5	**75.2**±1.3	70.3±2.1	36.3±3.2	41.6±2.8	8.5±4.2	54.2±9.9	59.5±2.6	24.5±5.5	20.2±2.1	60.9±4.6	53.9±4.7	61.5±2.6	67.6±2.7	71.8±2.1
	10	77.8±2.5	78.0±2.1	72.0±3.5	**80.0**±1.4	77.2±1.8	37.7±3.2	52.7±3.3	19.8±14.8	70.8±1.8	66.8±2.2	31.5±5.4	23.5±4.5	59.4±6.2	54.8±3.8	69.3±1.0	71.6±2.0	77.2±2.6
	20	81.8±2.3	81.5±2.4	77.5±0.7	83.5±1.4	82.2±2.2	40.4±2.0	73.3±2.7	54.1±2.0	77.0±1.5	74.3±2.3	39.2±1.9	31.6±2.5	66.9±7.5	67.5±2.7	74.3±1.8	75.2±2.2	**83.6**±1.5
	50	87.7±1.3	87.5±1.3	81.7±1.1	**89.4**±0.8	87.1±0.6	69.1±4.1	83.1±2.1	66.7±2.8	83.2±1.2	82.6±2.7	49.2±2.6	34.0±0.9	73.5±11.9	80.9±1.8	79.8±1.0	81.2±1.4	86.9±1.2

Table 5: P12 / P19 few-shot (k=10, 5 seeds): AUROC / AUPRC. Bold = best per column; underline = second best.

	P12		P19	
Method	AUROC	AUPRC	AUROC	AUPRC
RF-stats	68.9±2.9	26.3±1.0	75.2±4.3	25.4±9.3
MiniROCKET	67.3±3.1	26.0±2.8	75.7±1.5	19.6±8.4
MultiROCKET	67.2±3.2	24.0±2.1	74.7±4.4	16.0±5.8
catch22	58.5±3.0	18.4±1.0	73.8±10.1	28.2±11.8
StatsGap	65.6±2.8	23.6±1.8	**77.8**±1.9	**31.0**±9.7
RSS (ours)	**72.7**±2.3	**29.8**±3.1	72.2±3.6	22.4±8.2

H Full-data comparison on P12, P19, and PAM

Table 6 compares RSS against all baselines on the standard 5-split full-data protocol. All methods share splits, hardware, and per-method configurations following the authors' recommendations (Appendix E.3).

Table 6: Full-data classification on P12, P19, and PAM (5-split mean±std). P12/P19 report AUROC/AUPRC (%); PAM reports Accuracy/F1-macro (%). Bold = best per column, underline = second.

	P12		P19		PAM	
Method	AUROC	AUPRC	AUROC	AUPRC	Acc	F1
Non-DL baselines						
RF-stats	84.1±1.0	51.7±2.7	86.9±1.7	50.4±3.7	96.2±0.4	96.7±0.5
StatsGap	84.6±0.8	51.5±2.8	87.6±1.2	54.2±3.4	96.4±0.9	96.8±0.8
catch22	74.9±0.2	36.2±1.5	82.1±3.2	40.3±5.1	87.5±0.8	89.7±0.7
MiniROCKET	78.3±2.0	40.3±1.2	80.5±1.4	34.6±2.9	**98.1**±0.5	**98.3**±0.4
MultiROCKET	75.6±0.9	40.6±4.0	80.2±1.7	45.7±3.9	93.9±0.8	94.5±0.8
RNN						
GRU-D	70.3±2.7	27.3±1.2	**90.1**±1.6	**61.1**±2.4	93.2±1.1	94.2±1.1
ODE / Graph						
Latent ODE	81.1±1.8	44.4±3.8	80.4±2.3	26.7±3.0	94.2±0.7	94.9±0.4
Raindrop	83.4±1.6	47.5±1.7	86.8±2.5	54.4±2.8	90.6±0.4	91.8±0.4
Attention / Transformer						
mTAND	**86.6**±0.5	**55.5**±1.1	82.8±1.5	33.9±4.0	95.0±0.9	95.7±0.9
ATENet	85.3±0.7	51.9±3.3	80.7±1.8	26.0±3.1	93.0±1.5	93.8±1.5
Warpformer	86.5±1.0	55.4±2.5	89.8±1.8	59.5±5.0	64.3±2.2	65.0±2.1
Hi-Patch	84.5±1.0	49.9±2.2	86.6±1.6	49.8±4.1	50.7±1.2	51.3±1.7
Pre-trained						
PrimeNet	86.1±0.7	54.8±1.3	83.6±0.9	35.9±1.8	93.7±1.0	94.5±0.9
ViTST	82.6±1.7	45.5±4.1	84.2±2.0	44.0±4.0	95.8±1.4	96.0±1.5
Model-space learning (MSL)						
ESN-MS (ablation)	70.8±2.1	31.6±2.5	75.4±2.5	30.6±3.5	89.0±1.3	90.6±1.0
Ct-Echo [Chen et al., 2025]	79.1±2.1	42.5±3.0	82.9±2.3	42.1±3.0	91.1±1.0	92.2±0.9
GDRN-RSS (ours)	84.4±0.6	52.6±1.7	85.7±1.8	42.4±3.3	95.7±0.7	96.3±0.7

RSS extraction: <5 s (CPU) vs. DL training: 10–60 min per split (GPU)

MultiROCKET uses the standard n_{kernels}=10,000 on all three datasets (P19 full-data: 32 GB AutoDL run, ∼60 s/split extraction).

I Per-channel coverage on the clinical benchmarks

P12 and P19 sit at the extreme low end of per-channel coverage in this suite (Table 2: median T_j=3 and T_j=2 respectively, with >80% of (sample, channel) pairs at T_j<n). Coverage in the

sense of Theorem 2.3 fails strictly (some lab variables have $T_j{=}0$ for individual patients), $W_{\text{out}}^{j\star}$ is rank-deficient (Lemma 2.1), and μ_j, σ_j are computed from very few states.

The trajectory functionals still lift the W-only readout on both. On P12 the terminal state h_{end}^j is the single best component (Table 9: 74.8 AUROC, ahead of μ at 70.8), and the full RSS reaches 72.7 (vs. W-only 62.5), leading the few-shot non-DL comparison. On P19 the full RSS reaches 72.2 (vs. W-only 67.5) but trails the tabular features (StatsGap 77.8, RF-stats 75.2). The split tracks per-channel coverage: P12's median $T_j{=}3$ leaves enough trajectory for h_{end}^j to integrate, while P19's median $T_j{=}2$ does not.

Person Activity, also at 100% of pairs with $T_j{<}n$ but with median $T_j{=}17$, shows the same Lemma 2.1 mechanism in the opposite direction: the bounded summaries remain informative when T_j is moderate even if below n, and RSS leads. As PA observation rate halves, μ-only loses 0.3 points while W-only loses 7.9, consistent with the readout needing more observations per channel than μ. PAM (23.1% observation rate, 17 channels, $T_j{=}137$) sits comfortably inside the regime: at $k{=}10$ RSS reaches 77.2%, ahead of every supervised deep baseline, with three non-DL methods edging it (MiniROCKET 80.0, StatsGap 78.0, RF-stats 77.8).

J Behaviour at the boundary of data scarcity

Table 7 extends the label budget down to $k{=}1$ on Person Activity, P12, and P19 under the main protocol (5 seeds; the entries at $k{\geq}5$ on P12/P19 and at $k{=}20$ on Person Activity are the runs of Tables 1 and 4). Accuracy degrades gradually: Person Activity reaches 45.2% at $k{=}1$ (chance 20%), while on P12 and P19 the margin over chance at $k{=}1$–2 is comparable to the seed spread. P19 does not clear 70 AUROC until $k{=}10$. With 10 seeds, the entries at $k{=}1$–3 move by under 4 points.

Table 7: Label axis of scarcity: GDRN-RSS, W only, and μ only at $k{=}1$ to 20 labelled sequences per class (mean$\pm$std over the 5 seeds of the main protocol; chance is 20% on Person Activity, which has 5 classes, and 50 AUROC on P12/P19). Entries at $k{\geq}5$ on P12/P19 and at $k{=}20$ on Person Activity are the same runs as Tables 1 and 4. The lower block repeats the three smallest budgets for RSS with 10 seeds as a variance check.

		$k{=}1$	$k{=}2$	$k{=}3$	$k{=}5$	$k{=}10$	$k{=}20$
Person Activity (Acc)	**RSS**	45.2±5.9	58.8±14.3	65.4±8.3	81.7±1.9	86.0±0.4	87.2±1.1
	W only	35.9±11.3	44.9±10.4	49.5±10.8	57.8±8.1	66.7±4.1	72.3±1.5
	μ only	40.2±3.8	56.6±11.7	63.3±6.1	75.9±3.6	79.2±2.6	83.4±1.8
P12 (AUROC)	**RSS**	56.3±5.3	59.1±7.0	66.2±7.8	67.4±3.7	72.7±2.3	74.9±3.6
	W only	51.4±3.7	51.6±6.4	58.3±6.3	59.5±3.3	62.5±1.2	64.1±3.0
	μ only	54.4±7.2	58.3±9.3	63.8±7.0	68.0±4.8	70.8±2.7	74.5±0.8
P19 (AUROC)	**RSS**	59.1±9.2	58.5±13.5	64.1±11.5	63.7±13.6	72.2±3.6	76.5±3.8
	W only	54.6±6.9	56.9±8.4	59.8±7.0	61.7±9.8	67.5±3.0	71.7±3.0
	μ only	56.5±11.9	58.4±8.2	62.4±4.5	61.4±5.5	65.8±5.4	67.9±4.4
10-*seed check at the three smallest budgets*							
Person Activity (Acc)	10 seeds	42.2±8.0	57.4±12.0	67.5±9.0			
P12 (AUROC)	10 seeds	57.7±6.2	59.2±6.2	63.3±7.3			
P19 (AUROC)	10 seeds	56.2±10.8	54.9±11.6	60.4±10.6			

Table 8 thins Person Activity at $k{=}20$ from 7.5% to 1% observed. W and σ, which need about n and at least two observations per channel, approach chance, while μ and h_{end}, defined from a single observation, stay more than 20 points above it. At 1%, Full RSS (42.8) falls slightly below μ alone (45.1).

Part D: Mechanism Details

(ρ, λ) Landscape, Component Ablation

Two pieces of theory generate testable predictions about the GDRN-RSS mechanism. Remark C.4 predicts that the decay term $e^{-\lambda\Delta t}$ relaxes the standard ESN operator-norm budget, so the reservoir

Table 8: Observation axis of scarcity on Person Activity at k=20: the observed fraction is thinned from the 7.5% of Table 9 down to 1% (accuracy %, mean±std over 5 seeds; chance 20%). The 7.5% row is the PA +MCAR$_{7.5}$ row of Table 9. **Bold** = best per row.

Observed	W	σ	μ	h_{end}	**RSS**
7.5%	55.9±5.7	41.4±3.3	79.4±0.9	77.4±1.7	**80.1**±0.7
5%	54.7±3.3	32.7±1.7	**74.8**±2.1	73.2±2.5	74.6±2.0
3%	49.5±0.6	27.4±1.3	67.7±1.9	**68.5**±2.5	67.9±1.9
2%	42.9±1.8	23.6±1.0	58.8±5.0	**59.1**±4.8	**59.1**±4.6
1%	32.9±3.3	21.5±5.3	**45.1**±2.3	44.8±2.0	42.8±3.5

admits an operating regime past the conventional $\rho(W)$=1 boundary; Appendix K maps that regime empirically on Kuramoto. Lemma 2.1 shows that the readout $W_{\text{out}}^{j\star}$ is only partially identified as T_j shrinks while the bounded summaries $\mu_j, \sigma_j, h_{\text{end}}^j$ remain defined; Appendix L exposes that asymmetry per condition.

K Empirical (ρ, λ) landscape on Kuramoto

An empirical ρ–λ sweep on Kuramoto, the simplest dynamics-dependent dataset in the suite, maps the operating regime opened by Remark C.4. The plateau extending beyond $\rho(W)$=1 (reported below) is consistent with Yildiz et al. [2012], who show that for nonzero bounded inputs $\rho(W)$<1 is neither necessary nor sufficient for ESP.

Setup. Kuramoto, k=5 few-shot, MCAR 5%, 5 seeds. The sweep uses *GDRN-base* (readout W_{out} only, *without* the RSS summaries $\mu, \sigma, h_{\text{end}}$), which exposes raw sensitivity to (ρ, λ); the RSS summaries would otherwise flatten the grid by adding rank-deficient-robust statistics (cf. Lemma 2.1). Grid: $\rho \in \{0.3, 0.5, 0.7, 0.9, 1.0, 1.1, 1.3, 1.5, 1.7\}$, $\lambda \in \{0.01, 0.05, 0.10, 0.20, 0.50\}$, n=20.

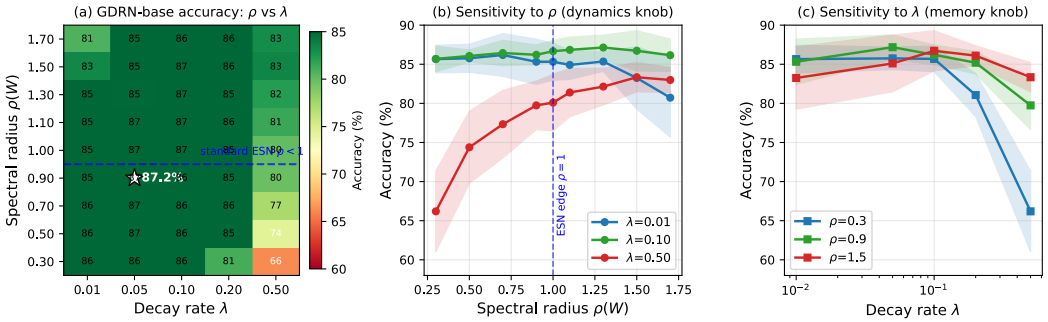


Figure 6: GDRN-base accuracy as a function of (ρ, λ) on Kuramoto (k=5, MCAR 5%, 5 seeds). (a) Accuracy heatmap; white star marks the empirical best (ρ=0.9, λ=0.05, 87.2%); the plateau extends across $\rho \in [0.9, 1.5]$, $\lambda \in [0.05, 0.10]$ and crosses the conventional ESN boundary ρ=1 (dashed blue), with accuracy at ρ=1.7 within 1 point of the peak. (b) Slices at fixed λ: accuracy is flat-to-mildly-increasing in ρ. (c) Slices at fixed ρ: accuracy is bell-shaped in λ with peak near 0.05–0.10.

Observations (Fig. 6). (i) *A broad accuracy plateau crosses the conventional ρ=1 boundary.* The empirical peak is ρ=0.9, λ=0.05 at 87.2% ± 1.6; the plateau extends through ρ=1.3 (87.1%), ρ=1.5 (86.7%), and ρ=1.7 (86.2%) at λ=0.1, with no performance cliff. Reservoir states remain bounded by construction via the tanh activation ($|h_i^j| \leq 1$ coordinate-wise), and no numerical divergence appeared across the grid. This is an empirical observation under the chosen initialization convention and input regime, not a theoretical claim about stability at $\rho(W) \geq 1$.

(ii) *The two axes produce different response shapes on this dataset.* Along the ρ axis (panel b), accuracy is flat-to-gently-increasing for moderate λ and more strongly monotone at large λ. Along the λ axis (panel c), accuracy is bell-shaped with a clear peak near $\lambda \in [0.05, 0.10]$ and degradation for large λ when ρ is small. The contrast is consistent with the design intuition in which λ and the

weight norm play distinct roles, but the claim here goes no further than the visual difference between panels (b) and (c).

(iii) *Empirical tradeoff between the two knobs.* At small ρ (0.3), λ sensitivity spans 19.6 points; at large ρ (1.5), it shrinks to 3.5 points. Symmetrically, at small λ (0.01) the ρ spread is 5.5 points, and at large λ (0.5) it is 17.1 points. Either parameter can partially compensate for suboptimal setting of the other.

Relation to the main-paper defaults. The main body uses fixed ρ=0.9, λ=0.1 for GDRN-RSS. The GDRN-base empirical peak above sits at ρ=0.9, λ=0.05, and the plateau extends to ρ=1.5, so the main-paper default lies on the low-ρ edge of the plateau, an empirical justification for the default being stable rather than tuned. The RSS features (μ_j, σ_j) flatten the (ρ, λ) response further (cf. Lemma 2.1 and the tight grid on P12 in Appendix N), so we do not recommend ρ>1 as a default: the observation is that GDRN *admits* this regime, not that one should use it.

L Full RSS component ablation

Table 9 reports the per-component decomposition behind main-text Fig. 2: each of the four single components (W, μ, σ, h_{end}) and the full RSS as a downstream feature for the same Random Forest, with ±std across all 22 conditions of Table 3. The mechanism analysis built on this table is in §3.4; the table here is the numerical reference.

Table 9: RSS feature ablation across all 22 conditions of Table 3 (accuracy %, mean±std over 5 seeds; AUROC for P12*/P19*). Same k-shot protocol as Table 1. **Bold** = best per row.

Condition (obs)	W	μ	σ	h_{end}	**RSS**
Lorenz Full (100%)	99.3±0.6	99.9±0.1	**100.0**±0.0	37.4±6.1	**100.0**±0.0
Lorenz MCAR (20%)	94.5±2.6	97.0±3.4	99.2±0.4	41.2±5.6	**100.0**±0.0
Lorenz Burst (71%)	98.5±0.9	98.5±1.4	**100.0**±0.0	41.3±4.2	**100.0**±0.0
Lorenz Event (23%)	96.7±2.6	99.6±0.7	99.9±0.1	39.7±4.1	**100.0**±0.0
Kuramoto Full (100%)	**93.8**±1.6	61.8±3.4	88.2±2.3	65.2±3.9	91.7±2.4
Kuramoto MCAR (20%)	**93.5**±1.8	60.4±4.7	88.7±2.6	64.1±3.6	91.0±2.2
Kuramoto Event (8%)	75.8±5.2	63.5±4.3	75.0±4.6	45.6±1.5	**76.1**±3.7
CML Full (100%)	97.9±1.7	92.2±5.5	98.3±1.0	71.4±2.2	**99.3**±0.2
CML MCAR (20%)	97.6±2.0	76.7±9.7	87.9±4.8	48.5±2.1	**99.3**±0.2
CML MCAR_5 (5%)	**94.6**±2.9	59.4±2.3	75.2±1.5	47.6±2.0	93.7±3.2
TEP Full (100%)	87.0±1.7	73.7±2.7	79.5±0.7	59.0±1.2	**91.0**±1.5
TEP MCAR (50%)	84.1±1.3	68.2±1.8	76.2±1.0	58.0±0.9	**90.1**±1.1
TEP Event (11%)	75.2±1.6	62.1±1.7	61.6±1.3	58.3±0.9	**80.4**±1.4
PA Natural (37%)	72.3±1.5	83.4±1.8	66.3±2.3	79.3±1.3	**87.2**±1.1
PA +MCAR (19%)	64.4±3.4	83.1±1.4	61.5±1.4	80.4±1.2	**86.5**±1.0
PA +$\text{MCAR}_{7.5}$ (7.5%)	55.9±5.7	79.4±0.9	41.4±3.3	77.4±1.7	**80.1**±0.7
Volcano Nat. (44%)	63.2±4.4	54.5±2.9	**66.5**±4.2	51.9±1.2	64.0±3.5
Volcano +MCAR (22%)	63.0±5.2	51.3±3.7	**65.8**±4.0	53.0±1.5	64.9±4.1
Volcano +MCAR_9 (9%)	62.3±4.1	51.7±1.4	**65.6**±3.6	53.0±1.8	63.4±3.7
PAM (23%)	70.2±1.5	76.4±2.6	67.3±1.9	63.4±3.2	**77.2**±2.6
P12* (5%)	62.5±1.2	70.8±2.7	61.9±6.0	**74.8**±2.9	72.7±2.3
P19* (12%)	67.5±3.0	65.8±5.4	**73.6**±3.6	65.2±5.1	72.2±3.6

M Attribution controls

All runs below use the main protocol: same seeds, splits, reservoir parameters, normalisation and classifier, 5 seeds, all 22 conditions.

Matched dimension and alternative statistics (Table 10). Relative to W alone, padding it to the RSS dimension $4bn$ with three Gaussian blocks or three random trajectory functionals lowers accuracy by 2.4 (SE 0.4, $22/22$ conditions) and 1.9 (SE 0.6, $17/22$), whereas RSS raises it by 5.1 (SE 1.5, $19/22$). The median or the maximum used alone is worse than RSS (-10.4 and -2.5). Swapping μ for the median or appending the maximum inside RSS changes the mean by less than 0.2.

Progressive ablation (Table 11). Adding μ, σ, and h_{end} to W in turn gives mean marginals of $+4.19$, $+0.78$, and $+0.08$. Averaged over all six insertion orders, they are $+2.2$, $+1.3$, and $+1.5$. h_{end} has its largest gain on P12 ($+5.6$ AUROC).

Backbone $\times$ feature factorial (Table 12). Adding RSS raises accuracy by 4.7 (SE 1.4) on Ct-Echo and 5.0 (SE 1.5) on GDRN, on $19/22$ conditions for each backbone. The GDRN backbone adds 2.3 with W and 2.6 with RSS, largest on Kuramoto Full and MCAR ($+19$ to $+21$) and negative on $11/22$ conditions with W, most on Person Activity and P19 (-4.6 to -9.6).

Group pooling (Table 13). G contiguous channel groups interpolate between global pooling (G=1) and uncoupled per-channel reservoirs (G=F). Per-channel pooling loses 9.3 to 22.8 points on four of the six Kuramoto and CML conditions. On the 16 conditions outside Kuramoto and CML, the variants stay within 1.5 points of each other.

Configuration regret (Table 14). Full RSS is within 1.0 point of the best of the 15 component subsets on $14/22$ conditions and within 2.5 on $20/22$. The two larger gaps are Volcano conditions where $W+\sigma$ is better by 2.6 and 4.2. We found no label-free rule that selects the best subset. Excluding Volcano, whose coverage is bimodal, Full RSS is never below W alone on the conditions where the median number of observations per channel $\bar{m}$ (Table 14) is below n, and every condition where W alone is better has $\bar{m} \geq n$.

Part E: Robustness and Efficiency

Hyperparameter, Seed, Classifier, k, Wall-Clock

N Hyperparameter robustness

To probe sensitivity to the default (ρ, λ) settings, a grid search is performed over $\rho \in \{0.5, 0.7, 0.9, 1.0, 1.1, 1.3\}$ and $\lambda \in \{0.01, 0.05, 0.1, 0.2, 0.5\}$ on P12 and P19 (the two most challenging clinical datasets in the suite, 5.2% and 11.6% observation rate). Configuration selection uses the validation set (split 1); test AUROC is reported independently. The reservoir size n is held at its default 20 and swept separately in Appendix P.

Table 15 shows the test AUROC for all (ρ, λ) combinations on P12 and P19 side-by-side.

Key findings:

- **Robustness across configurations.** Across all 30 configurations on P12, test AUROC ranges from 84.4 to 85.6 (spread 1.2 points). On P19, the range is 86.5–87.8 (spread 1.3 points). No tested configuration degrades performance substantially.
- **Default is near-optimal.** On P12, the default (ρ=0.9, λ=0.10, n=20) achieves 85.0%, within 0.6 points of the best (85.6% at ρ=0.5, λ=0.01). On P19, the default (87.5%) is within 0.3 points of the best (87.8% at ρ=0.5, λ=0.10). Both defaults sit on flat plateaus rather than narrow optima, so val-selected configs offer no test-time gain over the fixed default.
- **Sparse observations stabilise the operating regime.** High ρ with low λ can drive a densely-observed reservoir towards divergence; the clinical datasets show no such instability because their large inter-observation gaps generate strong decay ($e^{-\lambda \Delta t} \approx 0$), which damps the carried-over context regardless of ρ.
- **Reservoir size sensitivity is reported separately.** The $n \in \{10, 20, 30, 50, 100\}$ sweep on six representative conditions is in Appendix P: per-condition spread sits within the per-seed

Table 10: Matched-dimension controls and alternative statistics on the same GDRN trajectories (accuracy %, mean±std over 5 seeds; AUROC for P12*/P19*). W+noise pads the readout with three i.i.d. Gaussian blocks and W+r.traj with three random time-weighted averages of the reservoir states, both at the RSS dimension $4bn$. max and med are the per-channel coordinate-wise maximum and median of the trajectory used alone; $\mu\to$med replaces μ by the median inside RSS. All nine columns come from a single extraction pass over the same trajectories (the W and RSS columns agree with Table 9 to within 0.7 on the Lorenz rows and 0.1 elsewhere). **Bold** = best per row.

Condition (obs)	W (bn)	W+noise ($4bn$)	W+r.traj ($4bn$)	**RSS** ($4bn$)	max (bn)	med (bn)	$\mu\to$med ($4bn$)	$W+\max+\text{med}+h_{\text{end}}$	RSS+ max ($5bn$)
Lorenz Full (100%)	99.4±0.5	98.3±0.6	97.3±1.6	**100.0**±0.0	93.5±3.3	99.5±0.6	**100.0**±0.0	99.8±0.4	**100.0**±0.0
Lorenz MCAR (20%)	93.8±3.0	89.3±3.4	85.1±3.6	**99.6**±0.4	92.4±3.8	95.9±4.1	**99.6**±0.3	98.0±2.0	99.4±0.5
Lorenz Burst (71%)	98.0±0.8	95.6±2.5	92.6±5.5	**100.0**±0.0	95.6±2.4	98.5±1.1	99.9±0.1	99.6±0.4	**100.0**±0.0
Lorenz Event (23%)	96.9±2.0	95.0±2.7	89.5±6.0	**99.9**±0.1	95.9±2.3	99.3±0.9	**99.9**±0.1	99.3±0.9	**99.9**±0.2
Kuramoto Full (100%)	**93.8**±1.6	93.3±1.4	93.3±1.3	91.7±2.4	85.4±3.0	65.2±5.9	91.7±2.3	92.9±1.3	91.7±2.1
Kuramoto MCAR (20%)	**93.5**±1.8	92.5±1.8	92.5±1.7	91.0±2.2	88.0±1.2	63.9±4.4	91.1±2.1	92.4±1.4	91.4±2.0
Kuramoto Event (8%)	75.8±5.2	74.9±5.1	72.9±5.3	76.1±3.7	72.9±3.8	63.4±4.2	76.0±3.7	76.5±4.9	**76.7**±2.9
CML Full (100%)	97.9±1.7	97.9±1.7	99.0±1.1	99.3±0.2	98.6±1.5	96.1±1.8	99.4±0.2	99.5±0.2	**99.6**±0.1
CML MCAR (20%)	97.6±2.0	97.4±1.9	97.3±1.9	**99.3**±0.2	97.1±1.6	79.6±4.3	**99.3**±0.3	97.4±1.7	98.8±1.3
CML MCAR_5 (5%)	94.6±2.9	94.2±3.7	93.8±3.0	93.7±3.2	94.6±1.0	67.4±3.3	94.4±2.9	96.1±2.0	**96.3**±1.9
TEP Full (100%)	87.0±1.7	85.0±1.4	86.8±1.9	91.0±1.5	91.2±1.2	73.4±3.2	90.9±1.1	91.4±1.6	**91.6**±1.5
TEP MCAR (50%)	84.1±1.3	81.8±1.2	83.8±2.4	90.1±1.1	89.7±1.7	70.5±2.4	90.0±0.8	**90.6**±1.3	90.1±1.5
TEP Event (11%)	75.2±1.6	74.2±1.6	75.9±1.1	80.4±1.4	81.2±0.8	60.0±1.6	80.4±1.1	81.0±1.2	**81.3**±0.9
PA Natural (37%)	72.3±1.5	70.2±1.7	72.8±3.0	87.2±1.1	85.6±1.9	84.3±1.4	87.5±1.3	**87.9**±1.2	87.3±1.0
PA +MCAR (19%)	64.4±3.4	61.5±3.7	65.1±2.4	86.5±1.0	85.2±1.5	83.0±1.1	**86.6**±1.0	86.3±1.0	86.3±0.8
PA +$\text{MCAR}_{7.5}$ (7.5%)	55.9±5.7	51.6±6.0	55.1±1.7	**80.1**±0.7	79.8±1.3	78.8±0.7	**80.1**±0.2	79.4±1.1	79.9±1.5
Volcano Nat. (44%)	63.2±4.4	59.5±2.5	61.0±2.0	64.0±3.5	63.3±4.5	54.7±1.1	**65.2**±3.9	61.5±2.8	64.4±3.8
Volcano +MCAR (22%)	63.0±5.2	59.6±2.7	60.8±2.5	64.9±4.1	62.5±4.4	53.9±3.1	**65.3**±3.7	61.2±2.6	65.0±4.7
Volcano +MCAR_9 (9%)	62.3±4.1	58.3±1.5	59.3±1.3	63.4±3.7	61.8±3.9	51.8±1.5	63.1±4.1	60.7±3.4	**63.5**±3.5
PAM (23%)	70.2±1.5	67.5±1.1	70.8±1.8	**77.2**±2.6	73.5±3.6	75.7±2.3	76.7±2.6	77.1±3.5	**77.2**±2.2
P12* (5%)	62.5±1.2	59.4±3.0	60.5±2.8	72.7±2.3	69.6±3.7	70.1±2.3	72.7±1.8	**73.8**±2.0	71.6±2.5
P19* (12%)	67.5±3.0	58.9±4.3	61.7±3.7	**72.2**±3.6	68.9±4.1	66.2±5.0	71.8±3.4	69.1±4.4	71.0±3.8
Mean (22)	80.4	78.0	78.5	85.5	83.0	75.1	85.5	85.1	85.6

Table 11: Incremental RSS ablation: components added successively onto the readout W (accuracy %, mean±std over 5 seeds; AUROC for P12*/P19*). Δ columns give the sequential marginal gain of each added component; the last two rows give the mean over the 22 conditions and the number of conditions on which the marginal is positive. Same runs and protocol as Table 9. **Bold** = best per row.

Condition (obs)	W	Δ_μ	$W{+}\mu$	Δ_σ	$W{+}\mu{+}\sigma$	$\Delta_{h_{\text{end}}}$	**RSS**
Lorenz Full (100%)	99.3±0.6	+0.7	**100.0**±0.0	+0.0	**100.0**±0.0	+0.0	**100.0**±0.0
Lorenz MCAR (20%)	94.5±2.6	+4.1	98.6±1.8	+1.3	99.9±0.2	+0.1	**100.0**±0.0
Lorenz Burst (71%)	98.5±0.9	+1.3	99.9±0.2	+0.1	99.9±0.1	+0.1	**100.0**±0.0
Lorenz Event (23%)	96.7±2.6	+3.1	99.8±0.4	+0.2	**100.0**±0.0	+0.0	**100.0**±0.0
Kuramoto Full (100%)	**93.8**±1.6	+0.0	**93.8**±1.3	−1.9	92.0±2.6	−0.3	91.7±2.4
Kuramoto MCAR (20%)	**93.5**±1.8	−0.2	93.3±1.4	−1.8	91.5±2.6	−0.5	91.0±2.2
Kuramoto Event (8%)	75.8±5.2	−0.5	75.4±6.0	+1.2	**76.6**±3.7	−0.5	76.1±3.7
CML Full (100%)	97.9±1.7	+1.5	**99.4**±0.2	−0.1	**99.4**±0.2	−0.0	99.3±0.2
CML MCAR (20%)	97.6±2.0	+1.4	99.0±0.8	+0.4	**99.4**±0.2	−0.0	99.3±0.2
CML MCAR_5 (5%)	94.6±2.9	+0.4	**95.0**±2.5	−0.7	94.2±2.4	−0.5	93.7±3.2
TEP Full (100%)	87.0±1.7	+3.5	90.5±1.4	+0.7	**91.2**±1.1	−0.2	91.0±1.5
TEP MCAR (50%)	84.1±1.3	+4.9	89.0±1.6	+1.0	90.0±1.0	+0.1	**90.1**±1.1
TEP Event (11%)	75.2±1.6	+4.9	80.1±1.2	+0.7	**80.8**±1.2	−0.4	80.4±1.4
PA Natural (37%)	72.3±1.5	+14.4	86.7±0.9	+0.4	87.1±0.2	+0.1	**87.2**±1.1
PA +MCAR (19%)	64.4±3.4	+20.7	85.2±1.0	+0.8	86.0±0.9	+0.6	**86.5**±1.0
PA +$\text{MCAR}_{7.5}$ (7.5%)	55.9±5.7	+23.3	79.3±1.5	+1.5	**80.7**±1.3	−0.6	80.1±0.7
Volcano Nat. (44%)	63.2±4.4	−2.8	60.4±2.0	+4.5	**64.9**±3.8	−0.9	64.0±3.5
Volcano +MCAR (22%)	63.0±5.2	−1.8	61.2±4.3	+4.0	**65.3**±4.9	−0.3	64.9±4.1
Volcano +MCAR_9 (9%)	62.3±4.1	−2.1	60.2±3.0	+3.9	**64.1**±3.8	−0.7	63.4±3.7
PAM (23%)	70.2±1.5	+7.2	77.4±1.3	+0.1	**77.5**±1.8	−0.3	77.2±2.6
P12* (5%)	62.5±1.2	+6.4	68.9±2.5	−1.8	67.1±2.4	+5.6	**72.7**±2.3
P19* (12%)	67.5±3.0	+1.5	68.9±4.1	+2.9	71.9±4.3	+0.3	**72.2**±3.6
Mean (22)	80.5	+4.19	84.6	+0.78	85.4	+0.08	85.5
Conditions improved		17/22		16/22		7/22	

std on five of the six rows, while feature dimension and extraction time grow linearly and quadratically in n, motivating the $n{=}20$ default.

These results support a fixed default ($\rho{=}0.9$, $\lambda{=}0.10$, $n{=}20$) across datasets.

O Reservoir-seed vs. data-split sensitivity

Because (W, W_{in}) are drawn once and frozen, RSS performance could in principle depend on the reservoir draw as much as on the method itself. A two-factor design separates the two sources of variance: 5 data-split seeds × 5 reservoir seeds = 25 runs per dataset, with all other hyperparameters fixed at the paper default. Reservoir-seed std is consistently smaller than data-split std (ratio 1.4–43×), so reservoir randomization is not the dominant source of variation in any condition we report.

On Volcano CAU and P19 (AUPRC) the reservoir-seed std is more than an order of magnitude smaller than data-split std. This agrees with the kernel perspective of Appendix B.1: the implicit kernel k_{RSS} is determined by the distribution of (W, W_{in}), and different draws from that distribution produce similar downstream decisions; reservoir ensembling is therefore not expected to help, and does not.

P Reservoir size sensitivity

Reservoir size n is swept across $\{10, 20, 30, 50, 100\}$ on six representative conditions covering all four dominance patterns plus both clinical benchmarks (Table 17, Fig. 7), with five seeds per cell, RF classifier, and all other hyperparameters fixed at the paper default. Per-condition spread across the five n values sits within the per-seed std on five of the six rows, so reservoir size does not materially

Table 12: Backbone × feature factorial (accuracy %, mean±std over 5 seeds; AUROC for P12*/P19*), with Std ESN+RSS as a reference. Ct-Echo+W is the Ct-Echo column of Table 3, GDRN+W and GDRN+RSS are the W and RSS columns of Table 9; Ct-Echo+RSS and Std ESN+RSS are run under identical data, seeds, normalisation and classifier. **Bold** = best of the four factorial cells per row.

Condition (obs)	Ct-Echo+W	Ct-Echo+RSS	GDRN+W	**GDRN+RSS**	Std ESN+RSS
Lorenz Full (100%)	98.4±0.8	**100.0**±0.0	99.3±0.6	**100.0**±0.0	100.0±0.0
Lorenz MCAR (20%)	80.6±4.0	98.9±1.1	94.5±2.6	**100.0**±0.0	98.9±0.8
Lorenz Burst (71%)	91.2±4.4	**100.0**±0.0	98.5±0.9	**100.0**±0.0	99.9±0.1
Lorenz Event (23%)	94.0±1.7	99.7±0.3	96.7±2.6	**100.0**±0.0	99.6±0.4
Kuramoto Full (100%)	74.6±3.4	71.0±4.8	**93.8**±1.6	91.7±2.4	72.9±4.8
Kuramoto MCAR (20%)	72.7±3.8	69.7±4.8	**93.5**±1.8	91.0±2.2	71.3±4.6
Kuramoto Event (8%)	**77.9**±5.2	73.1±6.1	75.8±5.2	76.1±3.7	64.9±6.4
CML Full (100%)	98.0±1.6	**99.5**±0.1	97.9±1.7	99.3±0.2	99.4±0.1
CML MCAR (20%)	95.2±1.5	95.8±2.5	97.6±2.0	**99.3**±0.2	85.9±2.9
CML MCAR_5 (5%)	80.3±4.0	83.0±3.6	**94.6**±2.9	93.7±3.2	75.3±4.7
TEP Full (100%)	88.8±2.8	**91.0**±1.1	87.0±1.7	**91.0**±1.5	90.9±1.4
TEP MCAR (50%)	85.8±3.6	**90.2**±1.0	84.1±1.3	90.1±1.1	89.6±1.0
TEP Event (11%)	75.5±1.5	80.3±1.2	75.2±1.6	**80.4**±1.4	79.9±1.7
PA Natural (37%)	77.4±1.2	**87.5**±0.6	72.3±1.5	87.2±1.1	87.6±0.9
PA +MCAR (19%)	69.0±0.9	**86.8**±0.6	64.4±3.4	86.5±1.0	86.9±0.5
PA +$\text{MCAR}_{7.5}$ (7.5%)	65.5±1.7	79.5±1.4	55.9±5.7	**80.1**±0.7	79.2±1.6
Volcano Nat. (44%)	65.0±4.0	**65.7**±3.4	63.2±4.4	64.0±3.5	64.0±2.9
Volcano +MCAR (22%)	62.8±3.9	64.0±3.6	63.0±5.2	**64.9**±4.1	62.9±3.1
Volcano +MCAR_9 (9%)	60.8±2.6	**63.7**±3.5	62.3±4.1	63.4±3.7	63.4±3.2
PAM (23%)	71.6±2.0	**78.8**±1.9	70.2±1.5	77.2±2.6	78.4±1.6
P12* (5%)	62.0±1.8	72.5±1.8	62.5±1.2	**72.7**±2.3	73.1±2.6
P19* (12%)	72.6±3.0	**73.0**±4.0	67.5±3.0	72.2±3.6	71.1±3.7
Mean (22)	78.2	82.9	80.5	85.5	81.6

Table 13: Group pooling: the shared context is the decayed mean within each of G contiguous channel groups (accuracy %, mean±std over 5 seeds; AUROC for P12*/P19*), with full four-component RSS on top. G=1 is the default global pooling (the RSS column of Table 9); G=F gives fully per-channel, uncoupled reservoirs. **Bold** = best per row.

Condition (obs)	G=1 (global)	G=2	G=4	G=F (per-channel)
Lorenz Full (100%)	**100.0**±0.0	**100.0**±0.0	**100.0**±0.0	**100.0**±0.0
Lorenz MCAR (20%)	**100.0**±0.0	99.6±0.3	99.7±0.4	99.4±0.5
Lorenz Burst (71%)	**100.0**±0.0	**100.0**±0.0	**100.0**±0.0	**100.0**±0.0
Lorenz Event (23%)	**100.0**±0.0	99.8±0.3	99.9±0.1	99.6±0.4
Kuramoto Full (100%)	**91.7**±2.4	91.0±1.9	89.7±2.0	71.7±5.7
Kuramoto MCAR (20%)	**91.0**±2.2	90.5±2.4	88.4±1.7	68.2±4.8
Kuramoto Event (8%)	**76.1**±3.7	75.2±4.0	74.3±3.9	73.8±5.3
CML Full (100%)	**99.3**±0.2	**99.3**±0.2	**99.3**±0.1	99.2±0.5
CML MCAR (20%)	**99.3**±0.2	99.2±0.2	98.9±0.6	90.0±1.9
CML MCAR_5 (5%)	**93.7**±3.2	92.4±2.7	91.2±2.7	83.0±3.7
TEP Full (100%)	91.0±1.5	90.6±1.3	**91.1**±1.1	91.0±0.8
TEP MCAR (50%)	90.1±1.1	89.2±1.1	89.8±1.3	**90.5**±1.0
TEP Event (11%)	80.4±1.4	80.3±1.1	**80.6**±1.1	80.3±1.2
PA Natural (37%)	87.2±1.1	87.2±0.8	87.3±0.6	**87.7**±1.1
PA +MCAR (19%)	**86.5**±1.0	86.2±0.8	86.4±0.9	**86.5**±0.5
PA +$\text{MCAR}_{7.5}$ (7.5%)	80.1±0.7	**80.5**±1.5	80.1±1.2	79.1±1.7
Volcano Nat. (44%)	64.0±3.5	64.3±4.3	64.5±4.2	**65.2**±4.0
Volcano +MCAR (22%)	64.9±4.1	64.8±3.8	**65.0**±4.4	64.1±3.5
Volcano +MCAR_9 (9%)	63.4±3.7	63.4±3.3	**64.1**±3.7	63.8±3.7
PAM (23%)	77.2±2.6	**78.7**±2.6	78.4±1.8	78.5±2.4
P12* (5%)	**72.7**±2.3	72.4±2.1	72.3±2.3	72.4±2.1
P19* (12%)	72.2±3.6	72.5±3.8	**72.8**±3.6	72.3±3.5
Mean (22)	85.5	85.3	85.2	82.6

Table 14: Configuration regret of the fixed four-component RSS against the best of all 15 non-empty subsets of $\{W, \mu, \sigma, h_{\text{end}}\}$ (same runs as Table 9; accuracy %, AUROC for P12*/P19*), next to two label-free coverage statistics: $\bar{m}$, the median number of observations per channel over all (sequence, channel) pairs, and the share of pairs with $T_j{<}n$ ($n{=}20$). †Volcano is bimodal (56% of station-event pairs silent, the rest dense), so its raw median is not informative.

Condition (obs)	$\bar{m}$	$\%(T_j{<}n)$	W only	**RSS**	best subset	value	regret
Lorenz Full (100%)	1024	0	99.3	100.0	$\sigma{+}h_{\text{end}}$	100.0	0.0
Lorenz MCAR (20%)	205	0	94.5	100.0	$W{+}\mu{+}\sigma{+}h_{\text{end}}$	100.0	0.0
Lorenz Burst (71%)	734	0	98.5	100.0	σ	100.0	0.0
Lorenz Event (23%)	215	0	96.7	100.0	$\mu{+}\sigma{+}h_{\text{end}}$	100.0	0.0
Kuramoto Full (100%)	500	0	93.8	91.7	$W{+}h_{\text{end}}$	93.9	2.2
Kuramoto MCAR (20%)	100	0	93.5	91.0	$W{+}h_{\text{end}}$	93.5	2.5
Kuramoto Event (8%)	17	55	75.8	76.1	$W{+}\sigma$	77.1	1.0
CML Full (100%)	500	0	97.9	99.3	$W{+}\mu{+}h_{\text{end}}$	99.5	0.1
CML MCAR (20%)	100	0	97.6	99.3	$W{+}\mu{+}\sigma$	99.4	0.0
CML MCAR_5 (5%)	25	13	94.6	93.7	$W{+}\mu$	95.0	1.2
TEP Full (100%)	60	0	87.0	91.0	$W{+}\mu{+}\sigma$	91.2	0.2
TEP MCAR (50%)	30	0	84.1	90.1	$W{+}\mu{+}\sigma{+}h_{\text{end}}$	90.1	0.0
TEP Event (11%)	1	86	75.2	80.4	$W{+}\mu{+}\sigma$	80.8	0.4
PA Natural (37%)	17	100	72.3	87.2	$W{+}\mu{+}h_{\text{end}}$	87.5	0.3
PA +MCAR (19%)	8	100	64.4	86.5	$W{+}\mu{+}\sigma{+}h_{\text{end}}$	86.5	0.0
PA $+\text{MCAR}_{7.5}$ (7.5%)	3	100	55.9	80.1	$W{+}\mu{+}\sigma$	80.7	0.6
Volcano Nat. (44%)	0†	56	63.2	64.0	$W{+}\sigma$	68.2	4.2
Volcano +MCAR (22%)	0†	56	63.0	64.9	$W{+}\sigma$	66.9	2.0
Volcano $+\text{MCAR}_9$ (9%)	0†	56	62.3	63.4	$W{+}\sigma$	66.0	2.6
PAM (23%)	137	25	70.2	77.2	$W{+}\mu{+}h_{\text{end}}$	77.8	0.6
P12* (5%)	3	83	62.5	72.7	h_{end}	74.8	2.1
P19* (12%)	2	86	67.5	72.2	σ	73.6	1.4

Table 15: Test AUROC (%) across (ρ, λ) at $n{=}20$ on the two clinical datasets, full-data protocol with the four-component RSS feature. All 60 configurations fall within narrow ranges: P12 84.4–85.6 (1.2 pts), P19 86.5–87.8 (1.3 pts). Default config ($\rho{=}0.9, \lambda{=}0.10$) underlined; best per dataset bold.

	P12					**P19**				
$\rho\backslash\lambda$	0.01	0.05	0.10	0.20	0.50	0.01	0.05	0.10	0.20	0.50
0.5	**85.6**	85.1	85.3	84.9	84.8	86.6	87.1	**87.8**	87.0	87.2
0.7	85.3	85.2	85.2	84.9	85.0	86.9	87.5	87.0	87.0	86.7
0.9	84.9	85.3	<u>85.0</u>	85.0	84.6	86.8	87.1	<u>87.5</u>	87.0	87.1
1.0	84.7	85.2	85.1	84.6	84.4	86.9	87.6	87.3	86.5	86.6
1.1	84.7	84.8	85.3	85.2	84.4	87.1	86.5	87.4	86.9	87.1
1.3	84.4	84.8	84.9	84.9	85.1	87.1	87.6	86.7	87.0	87.5

affect accuracy in this regime; meanwhile feature dimension grows linearly ($4Fn$) and extraction time roughly quadratically (per-step reservoir matmul $O(n^2)$), which motivates the $n{=}20$ default.

Two patterns are worth noting. First, the largest spread across n is 3.4 points (Kuramoto MCAR 20%); on Volcano and Person Activity the spread is at most 1.0. Reservoir size is therefore not a hyperparameter that needs per-dataset tuning. Second, extraction time scales much more sharply: on Volcano CAU $n{=}100$ is $11.5\times$ slower than $n{=}10$ (7.48 s vs 0.65 s per dataset pass), and feature dimension grows from 320 ($n{=}10, F{=}8$) to 3,200 ($n{=}100$). $n{=}20$ provides a stable sweet spot where the trajectory is dimensional enough for the four RSS components to encode useful structure but extraction cost stays sub-second on most datasets.

Q Classifier robustness of RSS features

RSS is meant as a *representation-level* contribution, so it should remain informative under classifiers other than the Random Forest used in the main experiments. The same RSS feature vectors are passed to five downstream classifiers: Random Forest (RF, 300 trees), SVM with RBF kernel, Logistic

Table 16: Reservoir-seed vs. data-split-seed sensitivity, 5×5 data-seed by reservoir-seed grid (reservoir seeds 1000–1004, the canonical range used in Table 1). *Data-split std* is computed from the 5 data-seed row-means averaged over reservoir seeds; *Res std* is the symmetric quantity over reservoir seeds; *Pooled* is the std over all 25 runs. Reservoir-seed variance is consistently smaller than data-split variance (ratio 1.4–43×).

	Mean	Data-split std	Res std	Pooled
Kuramoto (MCAR 20% obs)	91.03	1.07	0.79	1.53
TEP (MCAR 50% obs)	90.21	1.14	0.17	1.18
Person Activity (+MCAR, 19% obs)	86.46	1.07	0.31	1.14
Volcano CAU (natural 44%)	64.16	3.42	0.08	3.47
P19 (natural, AUROC)	71.69	4.07	0.43	4.15
P19 (natural, AUPRC)	21.09	7.95	0.58	8.05

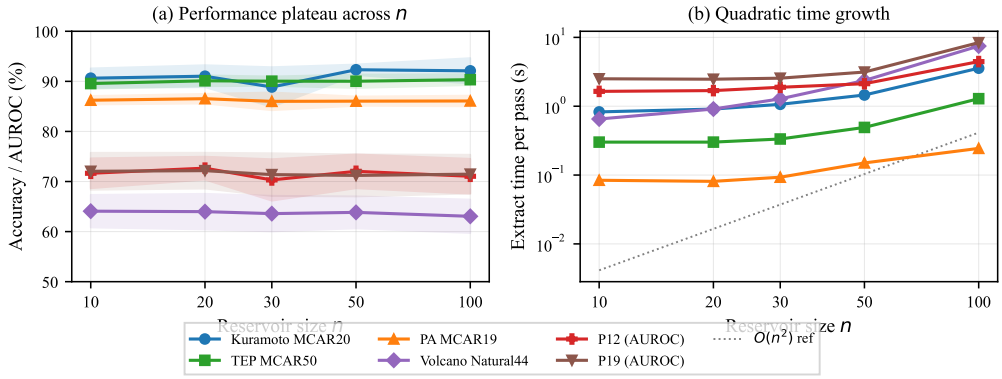


Figure 7: Reservoir-size sensitivity on six representative conditions (5 seeds). (a) Performance is flat across $n \in \{10, 20, 30, 50, 100\}$, with per-condition spread within the per-seed std on five of six conditions. (b) Extract time grows consistently with the per-step matmul $O(n^2)$ reference (dotted); n=100 is 3–11× slower than n=10 at no measurable accuracy gain, motivating the n=20 default.

Regression (L2, C=1), k-Nearest Neighbours (k=5), and the cross-validated Ridge classifier used by MiniROCKET / MultiROCKET. Table 18 reports 5-seed mean±std across nine conditions spanning six datasets (Kuramoto, TEP, and Person Activity at two observation rates each; Volcano at its natural rate; P12 and P19 at k=10 reporting AUROC for the binary clinical task).

RF wins every row but RSS produces usable accuracy under every classifier, so the representation is not co-adapted with the tree ensemble. Linear separability varies with task character: on Kuramoto, LogReg (88.9) is within 3 points of RF (91.7); on TEP (19 fault classes) the gap to the strongest linear classifier widens to 9–12 points, consistent with multi-class boundary complexity. kNN trails by 7–30 points on every row, reflecting the mismatch between Euclidean distance and a $4bn$-dimensional reservoir feature space (4,160 for TEP, 640 for Volcano). On the binary clinical tasks, RF dominates by a +7.7 AUROC gap on both P12 and P19; this is the regime where the bounded trajectory summaries do most of the work (h_{end} on P12, σ on P19; Table 9) and tree-based interactions across components pay off most.

R Effect of training set size (k=100)

Table 19 reports results when training set size is increased from k=5–20 (Table 1) to k=100 per class. Supervised DL recovers above 95% on Lorenz and CML Full at this budget, while RSS keeps the lead on Person Activity and the masked Volcano CAU conditions.

Key observations. (i) *DL baselines recover at k=100.* The same models that drop to near-chance at k=10 achieve strong performance at k=100 with no changes to code or hyperparameters: Warpformer rises from 48.4% at k=10 on Kuramoto Full to 91.8% at k=100 on Kuramoto MCAR (Kuramoto Full OOMs at k=100); MiniROCKET reaches 100% on CML and TEP; mTAND achieves 99.9% on

Table 17: Reservoir size sensitivity, 5 seeds per cell, RF classifier (Acc %; AUROC for P12*/P19*). Best per row in **bold**. Average extraction wall-clock per dataset pass is reported in seconds for n=10 and n=100 to show the cost–benefit tradeoff: n=100 is 3–11× slower than n=10 with no measurable accuracy gain.

Condition	n=10	n=20	n=30	n=50	n=100	$t_{n=10}$(s)	$t_{n=100}$(s)
Kuramoto MCAR (20%)	90.6±1.9	91.0±2.2	88.9±4.0	**92.3**±1.0	92.1±2.5	0.83	3.57
TEP MCAR (50%)	89.6±1.0	90.1±1.1	90.0±0.9	90.0±1.3	**90.3**±1.1	0.30	1.29
PA +MCAR (19%)	86.2±0.8	**86.5**±1.0	86.0±1.8	86.0±0.9	86.1±1.1	0.08	0.24
Volcano Natural (44%)	**64.1**±3.2	64.0±3.5	63.6±3.5	63.8±3.2	63.1±3.3	0.65	7.48
P12* AUROC (k=10)	71.6±3.0	**72.7**±2.3	70.3±4.1	72.0±3.4	71.0±3.5	1.64	4.45
P19* AUROC (k=10)	72.0±3.7	**72.2**±3.6	71.4±4.3	71.2±4.2	71.5±3.9	2.50	8.38

Table 18: Five-classifier robustness on RSS features (%, 5-seed mean±std). Multiclass conditions report Accuracy; P12* and P19* are binary clinical tasks at k=10 reporting AUROC. All classifiers use default hyperparameters with StandardScaler preprocessing (no scaling for RF).

Dataset	**Obs.**	**RF**	**SVM**	**LogReg**	**kNN**	**RidgeCV**
Kuramoto	100%	**91.7**±2.4	86.1±3.5	88.9±2.8	75.9±3.7	88.8±3.6
Kuramoto	20%	**91.0**±2.2	84.3±2.7	86.3±3.1	75.1±3.6	85.0±5.6
TEP	100%	**91.0**±1.5	74.8±0.9	77.5±0.7	61.2±1.4	82.3±2.0
TEP	50%	**90.1**±1.1	72.6±0.8	74.8±0.4	59.6±1.6	78.4±0.8
PA	37%	**87.2**±1.1	83.3±2.1	82.4±1.4	77.3±2.1	80.5±4.4
PA	19%	**86.5**±1.0	81.8±2.2	81.1±1.8	75.2±3.6	76.7±5.5
Volcano	44%	**64.0**±3.5	62.4±3.4	61.8±3.2	56.3±6.0	62.5±3.4
P12* AUROC	5%	**72.7**±2.3	62.9±13.9	65.0±7.3	61.8±3.8	65.0±7.2
P19* AUROC	12%	**72.2**±3.6	54.6±18.0	64.5±3.9	60.9±1.2	64.2±4.0

Lorenz (up from 89.8% at k=5); Latent ODE jumps from 64.5% to 95.9% on CML Full and from 66.6% to 88.4% on TEP Full.

(ii) *RSS advantage narrows but persists on sparse data.* On Kuramoto at full observation, the RSS lead over the next-best method drops from +5.2 (vs. ViTST at k=10) to +3.7 (vs. ViTST at k=100, since Warpformer OOMs on this column). On Person Activity under severe missingness (7.5% obs), RSS still leads all methods by +2.8 points over RF-stats. On Volcano CAU at k=100, RSS leads the two MCAR conditions by +0.7 (over ViTST) and +2.3 (over MultiROCKET), but ViTST overtakes RSS on Volcano Natural (81.2 vs. 79.5), where the dense observations let the ImageNet-21k pre-training carry signal that the reservoir does not.

(iii) *ViTST and MiniROCKET are the broadest beneficiaries of more data.* MiniROCKET reaches 100% on TEP Full at k=100 (100%/86.7%/98.1%) where it trailed RSS by 11.6 points at k=10. Warpformer narrows but does not surpass RSS on Kuramoto Event (85.9% vs. 90.0%), and stays well below RSS on Kuramoto MCAR (91.8% vs. 97.0%); even with abundant labels this Warpformer configuration remains less effective than RSS on these coupled-oscillator tasks. ViTST recovers across the grid at k=100: 100% on every Lorenz condition, 99.3/99.2/98.3 on CML, 96.3 on TEP Full, and 77.4 on Person Activity Natural. The k=10 row in Table 3 has TEP Full and PA Natural at 60.3 and 37.5. ViTST's earlier weakness in those domains therefore reflects label scarcity rather than a domain mismatch; at k=100 it leads on Volcano Natural (81.2 vs. RSS 79.5) while still trailing RSS on Person Activity under additional MCAR and on Volcano under MCAR, where the irregular and sparse observations leave less for the image renderer to exploit. PrimeNet remains consistently weak across all datasets at k=100.

(iv) *Practical implication*: RSS is most useful when labeled data are scarce and expensive, the setting that motivates this work. When labels are abundant ($k \geq 100$), supervised methods close the gap on most datasets, while RSS retains its lead on Person Activity and the masked Volcano CAU conditions at orders-of-magnitude lower training cost (seconds vs. minutes on GPU).

Table 19: Classification accuracy (%) at k=100 per class (5 seeds). Lorenz uses $N_{\text{per-class}}$=200 (600 total). Hi-Patch OOMs on dense-observation datasets (—). Bold = best per row, underline = second.

		Feature-based					RNN	ODE/Graph		Attention/Transformer				Pre-trained		Model-space learning		
Dataset	**Mask (obs%)**	**RF-stats**	**StatsGap**	**catch22**	**MiniR**	**MultiR**	**GRU-D**	**L-ODE**	**Raindrop**	**mTAND**	**ATENet**	**Warpf.**	**Hi-Patch**	**PrimeNet**	**ViTST**	**ESN-MS**	**Ct-Echo**	**RSS (ours)**
Lorenz	Full (100%)	**100.0**±0.0	**100.0**±0.0	93.3±13.3	93.3±13.3	**100.0**±0.0	71.1±18.0	**100.0**±0.0	91.9±3.3	99.9±0.1	71.7±13.9	**100.0**±0.0	—	86.7±16.3	**100.0**±0.0	**100.0**±0.0	**100.0**±0.0	**100.0**±0.0
	MCAR (20%)	**100.0**±0.0	**100.0**±0.0	80.0±16.3	86.7±16.3	**100.0**±0.0	50.8±11.6	50.0±30.7	65.1±5.2	99.1±0.7	80.2±13.7	99.9±0.1	—	60.0±24.9	**100.0**±0.0	44.4±41.6	88.9±15.7	**100.0**±0.0
	Burst (71%)	**100.0**±0.0	**100.0**±0.0	80.0±16.3	86.7±16.3	**100.0**±0.0	60.5±4.7	**100.0**±0.0	81.9±4.5	99.3±0.8	79.6±12.1	**100.0**±0.0	—	53.3±16.3	**100.0**±0.0	88.9±15.7	**100.0**±0.0	**100.0**±0.0
	Event (23%)	**100.0**±0.0	**100.0**±0.0	**100.0**±0.0	93.3±13.3	**100.0**±0.0	63.9±24.1	55.6±15.7	96.4±4.1	98.9±0.9	73.8±21.9	99.9±0.1	—	60.0±13.3	**100.0**±0.0	44.4±41.6	**100.0**±0.0	**100.0**±0.0
Kuramoto	Full (100%)	79.3±1.4	79.3±0.9	90.0±1.4	90.7±1.0	92.0±0.4	61.4±10.7	75.2±7.4	68.4±8.8	65.2±5.2	63.5±2.1	—	—	61.2±4.5	93.5±0.5	87.2±0.6	86.8±0.9	**97.2**±0.5
	MCAR (20%)	69.3±1.2	66.6±1.4	87.4±0.8	90.9±1.3	92.9±0.7	33.1±0.9	56.8±6.2	36.0±3.3	66.3±0.6	61.6±2.8	91.8±2.4	88.9±5.6	59.2±5.0	91.4±0.5	82.5±0.6	80.7±2.7	**97.0**±0.7
	Event (8%)	83.0±1.0	83.9±0.6	82.5±1.3	93.4±0.6	**95.6**±0.8	66.2±2.6	72.6±7.7	86.1±1.7	67.2±0.6	61.2±2.3	85.9±3.6	82.3±2.7	53.6±5.9	91.9±0.9	78.6±0.9	88.0±1.2	90.0±1.1
CML	Full (100%)	99.6±0.2	99.7±0.1	99.1±0.7	**100.0**±0.0	99.8±0.1	96.7±1.3	95.9±0.3	89.5±2.3	90.5±2.1	43.0±13.2	—	—	72.4±8.7	99.3±0.2	99.6±0.0	99.5±0.1	99.5±0.1
	MCAR (20%)	99.4±0.6	**99.6**±0.3	96.9±1.1	98.6±0.2	99.0±0.2	62.9±15.1	81.2±11.1	69.8±7.6	77.6±1.6	56.9±12.2	93.5±1.8	92.9±2.1	65.8±3.2	99.2±0.4	60.3±1.3	99.5±0.1	99.5±0.1
	MCAR (5%)	**99.7**±0.1	**99.7**±0.1	89.4±1.3	92.3±0.6	94.3±0.4	35.5±0.9	66.4±0.6	54.3±17.2	62.9±1.7	50.2±10.6	72.3±4.3	65.5±3.1	52.2±10.2	98.3±1.5	44.8±4.4	94.9±0.7	99.2±0.2
TEP	Full (100%)	99.7±0.6	**100.0**±0.0	90.8±3.5	**100.0**±0.0	99.2±1.1	65.0±11.7	88.4±5.6	89.4±5.9	50.6±7.5	60.0±13.2	81.7±15.4	60.3±8.0	55.0±8.0	96.3±0.7	97.2±1.1	98.1±0.7	99.5±0.7
	MCAR (50%)	**97.5**±2.0	96.1±3.7	58.1±2.0	86.7±4.5	83.1±4.3	55.0±8.7	63.9±5.2	76.1±3.0	54.7±10.6	65.0±12.7	82.5±11.6	62.2±10.9	45.3±11.3	90.7±2.4	68.1±3.4	80.6±3.0	96.3±3.3
	Event (11%)	93.1±3.4	92.5±3.8	78.3±3.9	**98.1**±1.4	96.1±2.7	50.0±12.8	51.9±4.3	48.1±16.5	45.6±6.7	56.7±4.9	65.3±17.2	61.4±15.7	46.1±5.5	66.7±1.1	74.1±0.7	81.5±2.6	86.6±2.4
Person Activity	Natural (37%)	89.3±0.9	88.9±0.8	39.6±2.0	88.4±1.2	87.0±1.2	82.0±3.3	86.1±1.9	33.1±26.6	90.2±2.7	86.9±2.9	88.7±1.1	84.9±2.4	75.5±14.3	77.4±2.6	61.8±4.8	81.9±1.0	**92.0**±0.4
	+MCAR (19%)	87.5±0.9	87.5±1.3	48.4±2.2	74.7±1.3	77.1±0.5	64.0±2.0	70.6±1.2	26.7±19.8	66.8±0.9	58.3±7.8	82.3±2.9	73.3±5.4	64.0±4.0	78.4±1.5	61.4±1.8	75.7±1.0	**90.7**±1.2
	+MCAR (7.5%)	83.0±1.4	82.6±2.0	28.3±2.2	63.8±2.2	66.9±1.5	31.6±17.2	57.3±3.6	18.8±14.8	54.0±1.9	36.3±12.8	71.2±6.0	61.4±4.0	50.5±8.4	62.3±1.3	57.9±0.2	73.1±1.4	**85.8**±1.3
Volcano CAU	Natural (44%)	78.1±0.9	77.3±1.0	72.0±0.8	78.5±1.7	79.2±1.0	34.3±4.5	62.5±2.3	62.0±17.9	38.6±7.0	35.4±4.5	65.7±5.5	38.6±3.4	51.2±2.0	**81.2**±0.7	74.1±0.6	76.6±0.5	79.5±0.4
	+MCAR (22%)	76.1±1.0	73.4±1.3	62.3±1.7	77.0±1.2	77.0±1.1	33.3±4.8	49.9±9.2	55.8±2.8	36.0±6.0	31.6±4.2	70.6±2.2	35.2±5.3	34.3±4.8	78.0±1.9	64.5±1.0	76.0±0.8	**78.7**±0.2
	+MCAR (9%)	74.9±0.6	72.7±0.9	57.0±1.0	75.0±1.2	75.0±1.1	33.5±3.4	47.8±8.5	46.1±8.1	31.0±2.0	34.2±4.6	69.0±3.2	36.3±4.3	34.1±4.5	74.0±2.6	64.8±0.8	74.1±0.3	**77.3**±0.6

Strong non-DL columns in Table 19. (a) At k=100, StatsGap and MultiROCKET reach 100% on every Lorenz condition; catch22 saturates only on Event (100.0), with 93.3 on Full and 80.0 on MCAR and Burst. Lorenz is therefore saturated by marginal-value statistics at this budget and not informative for distinguishing temporal feature extractors.

(b) *MultiROCKET is strong on coupled-oscillator datasets at* k=100: it reaches 92.0–95.6 on Kuramoto, ahead of Warpformer (85.9–91.8 where it runs; OOM on Kuramoto Full) and below RSS on Full and MCAR (97.0–97.2) but above it on Event (95.6 vs. 90.0). At k=100 MultiROCKET is the closest non-DL competitor on Kuramoto.

(c) *Tabular and convolutional features lead TEP at* k=100. Across the three TEP conditions (Full, MCAR 50%, Event 11%), RF-stats reaches 99.7/**97.5**/93.1, StatsGap reaches **100.0**/96.1/92.5, MultiROCKET reaches 99.2/83.1/96.1, and MiniROCKET reaches **100.0**/86.7/**98.1**. RSS is competitive but not the leader at k=100 on the masked TEP conditions. The TEP signal is well captured by per-channel features once labels are abundant.

(d) On Person Activity, RSS retains a 3−58-point lead over StatsGap, catch22, and MultiROCKET across all three conditions, consistent with PA being a clear case where reservoir dynamics add value over hand-crafted statistics under sparse observations.

(e) *On Volcano at* k=100, StatsGap (77.3/73.4/72.7), catch22 (72.0/62.3/57.0), and MultiROCKET (79.2/77.0/75.0) all trail RSS (79.5/78.7/77.3) by 0.3–20.3 points, replicating the k=10 pattern in Table 3.

S Extraction time comparison

All six non-DL methods are measured on the same 25-core Xeon 8470Q (AutoDL, 3 seeds, all cores allocated). Table 20 reports the per-split feature-extraction wall-clock; Table 21 adds the downstream classifier fit time.

Table 20: Feature-extraction wall-clock per condition (seconds, mean over 3 seeds; extraction only, classifier fit excluded). GDRN-RSS uses numba `prange`; RF-stats and StatsGap are numpy-vectorized reductions over the masked tensor; catch22 uses joblib over per-sequence `pycatch22` calls; MiniROCKET and MultiROCKET use aeon's convolution-based transforms. "P12 full" and "P19 full" use the entire training set; other rows use the few-shot subsample from Table 3.

Dataset	N	GDRN-RSS	RF-stats	StatsGap	catch22	MiniR.	MultiR.
Lorenz-96 (MCAR 20%)	300	1.5	0.3	0.4	27.4	2.2	3.3
Kuramoto (MCAR 20%)	1,500	1.2	1.5	2.2	72.4	5.0	14.5
CML (MCAR 20%)	1,500	1.5	1.8	2.6	101.3	6.0	15.9
TEP (MCAR 50%)	1,972	0.4	1.7	1.6	20.6	1.1	2.5
Person Activity	2,244	0.1	0.5	0.4	6.7	0.3	1.8
PAM	5,333	1.8	1.5	2.4	156.7	6.2	15.1
P12 (full data)	11,988	2.3	5.1	5.8	71.4	9.5	22.1
P19 (full data)	38,803	2.9	13.0	11.8	165.4	10.0	37.1

All six CPU methods finish extraction in at most a few minutes at the largest scale (N=38,803, P19 full training set); GDRN-RSS and the two simple-statistics methods (RF-stats, StatsGap) stay under 15 seconds across every dataset. catch22 is the outlier within the CPU group because its 22 canonical features per channel are individual C calls with little vectorization headroom. The heaviest deep baselines (Warpformer, Latent ODE, ViTST) take tens of GPU-minutes per split on the same AutoDL instance (Fig. 8), two to three orders of magnitude longer than GDRN-RSS; lighter deep baselines (mTAND, ATENet) finish in tens of seconds on GPU, comparable to the CPU range.

End-to-end wall-clock (including classifier fit). Extraction alone understates the kernel methods' cost because their features feed a cross-validated ridge classifier whose efficient-LOO formulation scales with the square and cube of the feature dimension. On P19 full-data, MultiROCKET's ∼50K features dominate: the 854-second RidgeCV fit is over 20× its extraction time. Methods whose features feed a Random Forest (GDRN-RSS, RF-stats, StatsGap, catch22) keep the classifier stage under a minute on every condition in Table 21.

Across all eight conditions, every CPU method completes a full train+test split in at most 15 minutes wall-clock, and GDRN-RSS stays under one minute on every condition including P19 full-data. The heaviest deep baselines on the same AutoDL instance take tens of GPU-minutes per split (Fig. 8), one to three orders of magnitude above any CPU row here; lighter deep baselines are roughly on par in wall-clock but still require a GPU. The whole GDRN-RSS pipeline therefore fits inside a single CPU-minute on the largest irregular-TS classification dataset in our suite, with no gradient-based training and no GPU required.

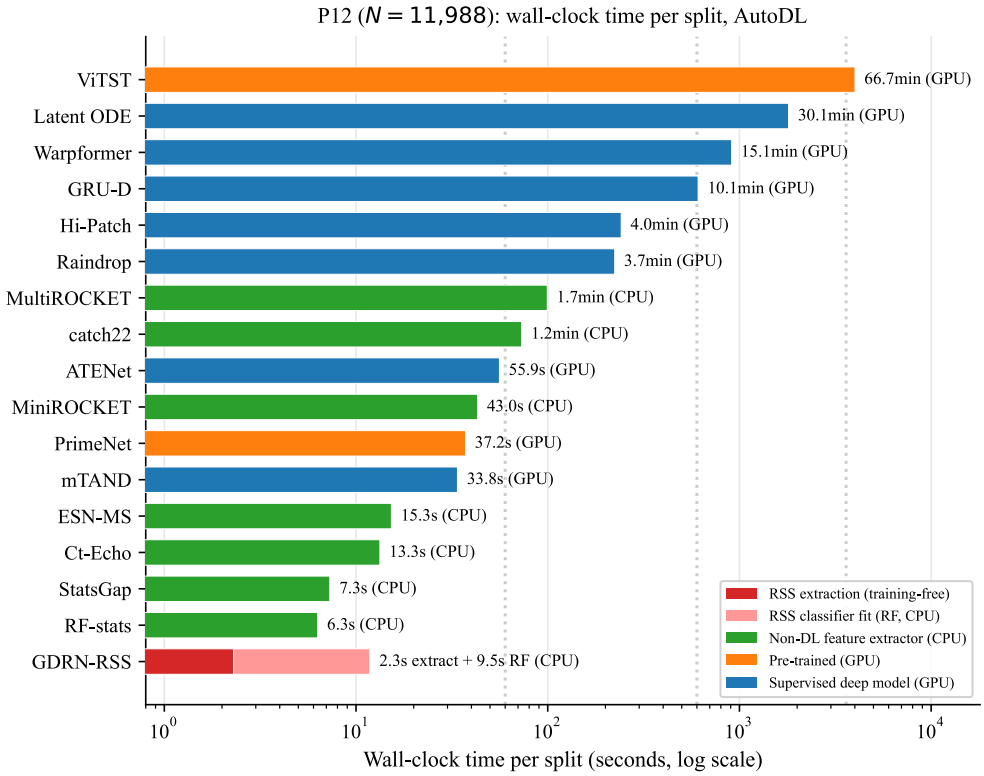


Figure 8: Wall-clock per split on P12 fulldata (N=11,988) on the AutoDL instance. RSS is split into feature extraction (dark red) and classifier fit (light red); other CPU baselines (green) report total extraction + classifier time. Deep baselines run on GPU (blue / orange).

Table 21: End-to-end wall-clock breakdown: extract / classifier / total, in seconds (mean over 3 seeds). Classifier is Random Forest (300 trees) for GDRN-RSS, RF-stats, StatsGap, and catch22; RidgeCV (7-α, built-in efficient LOO) for MiniROCKET and MultiROCKET. Same hardware and thread discipline as Table 20.

Condition	Method	Extract	Classifier	Total
Lorenz MCAR20 (N=300)	GDRN-RSS	1.5	0.7	2.2
	RF-stats	0.3	0.8	1.0
	StatsGap	0.4	0.6	1.0
	catch22	27.4	0.7	28.1
	MiniROCKET	2.2	0.0	2.3
	MultiROCKET	3.3	0.2	3.5
Kuramoto MCAR20 (N=1,500)	GDRN-RSS	1.2	0.7	2.0
	RF-stats	1.5	0.7	2.2
	StatsGap	2.2	0.6	2.8
	catch22	72.4	0.8	73.2
	MiniROCKET	5.0	0.4	5.4
	MultiROCKET	14.5	4.3	18.9
CML MCAR20 (N=1,500)	GDRN-RSS	1.5	0.7	2.2
	RF-stats	1.8	0.6	2.4
	StatsGap	2.6	0.6	3.3
	catch22	101.3	0.7	101.9
	MiniROCKET	6.0	0.3	6.3
	MultiROCKET	15.9	4.2	20.1
TEP MCAR50 (N=1,972)	GDRN-RSS	0.4	0.5	0.9
	RF-stats	1.7	0.8	2.5
	StatsGap	1.6	0.8	2.4
	catch22	20.6	0.9	21.5
	MiniROCKET	1.1	0.2	1.2
	MultiROCKET	2.5	2.0	4.5
PA Natural37 (N=2,244)	GDRN-RSS	0.1	0.8	0.9
	RF-stats	0.5	0.6	1.0
	StatsGap	0.4	0.7	1.1
	catch22	6.7	0.8	7.4
	MiniROCKET	0.3	0.2	0.4
	MultiROCKET	1.8	1.7	3.5
PAM full (N=5,333)	GDRN-RSS	1.8	1.4	3.2
	RF-stats	1.5	0.6	2.1
	StatsGap	2.4	0.7	3.1
	catch22	156.7	0.9	157.6
	MiniROCKET	6.2	6.7	12.9
	MultiROCKET	15.1	18.3	33.4
P12 full (N=11,988)	GDRN-RSS	2.3	9.5	11.8
	RF-stats	5.1	1.2	6.3
	StatsGap	5.8	1.5	7.3
	catch22	71.4	1.7	73.1
	MiniROCKET	9.5	33.5	43.0
	MultiROCKET	22.1	77.1	99.3
P19 full (N=38,803)	GDRN-RSS	2.9	53.5	56.3
	RF-stats	13.0	4.2	17.2
	StatsGap	11.8	4.7	16.5
	catch22	165.4	6.3	171.7
	MiniROCKET	10.0	168.3	178.2
	MultiROCKET	37.1	853.9	890.9

Part F: Supplementary Figures

T Supplementary figures from the main body

Two supplementary figures referenced from the main body collect here: robustness curves under increasing missingness (Fig. 9), and a per-condition comparison of raw input summaries against reservoir-state summaries under the same downstream classifier (Fig. 10). The wall-clock comparison referenced from §3.6 sits with the timing tables in Appendix S.

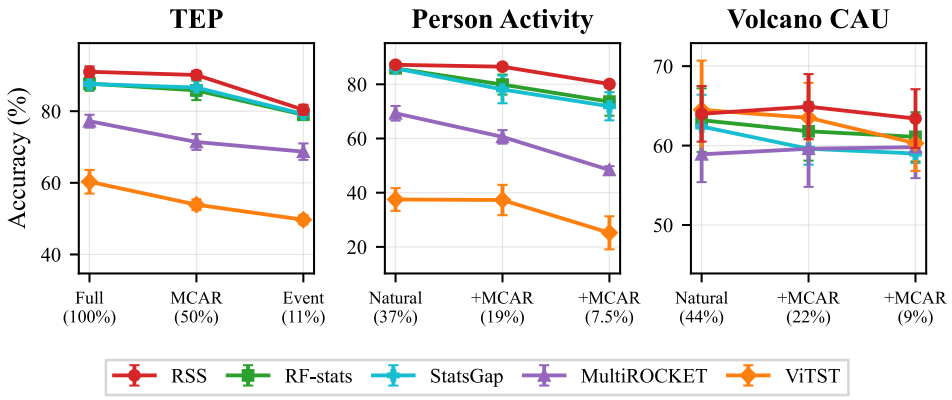


Figure 9: Accuracy vs. observation rate at the few-shot protocol of Table 1 (5 seeds, error bars = std). Compared methods: RSS (ours), RF-stats, StatsGap, MultiROCKET, and ViTST.

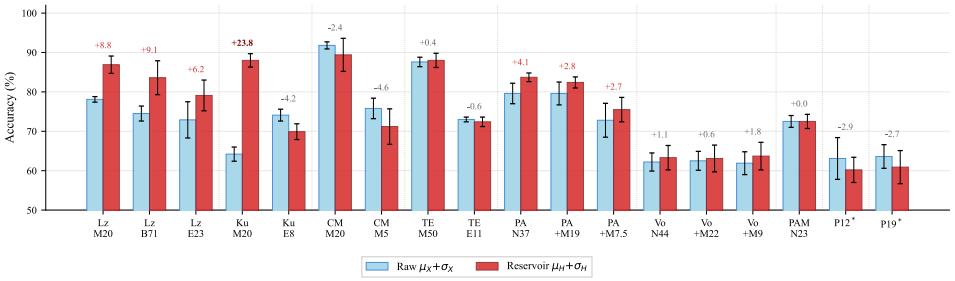


Figure 10: Per-bar accuracy of raw input $\mu_X+\sigma_X$ vs. reservoir-state $\mu_H+\sigma_H$ under the same RF classifier and k-shot protocol, across all 18 main-table conditions (Acc; AUROC for P12*/P19*).

Where the reservoir transform helps. Reservoir-state $\mu_H+\sigma_H$ improves over raw input $\mu_X+\sigma_X$ on Kuramoto MCAR ($+23.8$), Lorenz ($+6$ to $+9$ across mask patterns), and Person Activity ($+2.7$ to $+4.1$). It ties on TEP, Volcano, and PAM, and underperforms on Kuramoto Event 8%, CML, and the clinical benchmarks. The pattern lines up with Lemma 2.1: the transform helps when T_j supports a meaningful trajectory summary, and stops helping when per-channel coverage is too low (Kuramoto Event, P12, P19) or when the class signal already lives in the per-step value distribution that the raw summaries capture (CML).

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction state the contribution as a closed-form, gradient-free pipeline (RSS + GDRN) for asynchronous multivariate irregular time series, with the ESP guarantee (Theorem 2.3) explicitly bounded to settings where the per-channel coverage condition holds (Remark C.3). The experimental sections report the coverage-failure regimes separately, showing that RSS still leads P12 via the terminal-state component while falling behind tabular time-gap features on P19, the strictest coverage-failure regime.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: A dedicated Limitations section (Sec. 5) discusses the fixed, non-adaptive reservoir parameters and the uniform cross-channel pooling step. Clinical-benchmark coverage failures and the rescue patterns of RSS are characterized in Appendix I, and §3.4 additionally discloses the cases where a single RSS component matches or marginally beats Full RSS.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best

judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

   Justification: Lemma 2.1 (rank-deficient readout), Proposition 2.2 (reservoir variance separates lagged structure), and Theorem 2.3 (ESP for GDRN) state assumptions explicitly. Full proofs appear in Appendix A, Appendix B, and Appendix C, respectively.

   Guidelines:

   - The answer [N/A] means that the paper does not include theoretical results.
   - All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
   - All assumptions should be clearly stated or referenced in the statement of any theorems.
   - The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
   - Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
   - Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: A public code repository is linked in the abstract, dataset preprocessing follows the public Raindrop release for P12/P19/PAM, and Appendix E lists the GDRN-RSS algorithm, hyperparameters, baseline configurations, masking mechanisms, normalisation, and train/test splits.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.

(c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).

(d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: Code, run scripts, and preprocessing recipes are available at the public URL given in the abstract. All datasets are public (cited per dataset in Appendix D).

   Guidelines:

   - The answer [N/A] means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Sec. 3.2 states the fixed GDRN-RSS configuration. Appendix E.3 gives per-baseline architecture, optimiser, and stopping criteria. Appendix E.4 gives masking parameters, and Appendix N (with reservoir-size and seed sweeps in Appendix P and O) reports hyperparameter robustness.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [Yes]

Justification: Unless stated otherwise, results are mean±std over 5 random seeds (per-method splits for P12/P19/PAM following Zhang et al. [2022]). The main Table 1 reports means with ±std in Appendix Table 3.

Guidelines:

- The answer [N/A] means that the paper does not include experiments.
- The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: All wall-clock measurements were taken on a shared AutoDL instance (Intel Xeon 8470Q + NVIDIA RTX 5090). Per-method extraction times are reported in Appendix S and visualised in Fig. 4. GDRN-RSS feature extraction is CPU-only and takes a few seconds per split.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The paper uses public benchmark datasets (PhysioNet, UCI, PAMAP2, TEP, etc.) without any private or scraped data. No human-subjects experiments were conducted by the authors, and the paper does not introduce models with high misuse risk.

   Guidelines:

- The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: Lower compute and label requirements lower the barrier to on-device wearable, edge industrial, and low-resource deployments. On the risk side, any clinical / hazard / fault-handling classifier (including ours) should be used as a calibrated decision aid with human-in-the-loop review, and the explicit scope statement (Remark C.3, Sec. 2.4) helps communicate the failure regime on extremely sparse per-channel benchmarks (P12, P19).

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper releases a closed-form feature extractor and classifier. The method does not generate text, images, or other content with elevated misuse risk, and no scraped or sensitive data are released.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.

- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All datasets and code packages used are cited at first mention (Appendix D and Appendix E.3). Preprocessed P12/P19/PAM follow the redistribution from Zhang et al. [2022] (Raindrop, MIT), and other datasets are obtained from their original public sources under their stated licences.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: The public code repository linked in the abstract ships the GDRN-RSS implementation, run scripts, and a README documenting environment, dependencies, and reproduction commands. No new datasets are released.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or new human-subjects research. PhysioNet, PAMAP2, and Person Activity datasets are pre-existing public benchmarks released by their original collectors.

Guidelines:

- The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: No new human-subjects research was conducted. Pre-existing public datasets were obtained under their original IRB approvals and licenses.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [Yes]

    Justification: LLMs assisted with prose editing, proof drafting, and verification-code engineering. The scientific framing and final results rest with the authors.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.