OPIT Open Institute of Technology

Toward a Dynamical Theory of Deep Learning

Coupled State–Parameter Dynamics and Time-Scale Interaction

Lorenzo Livi

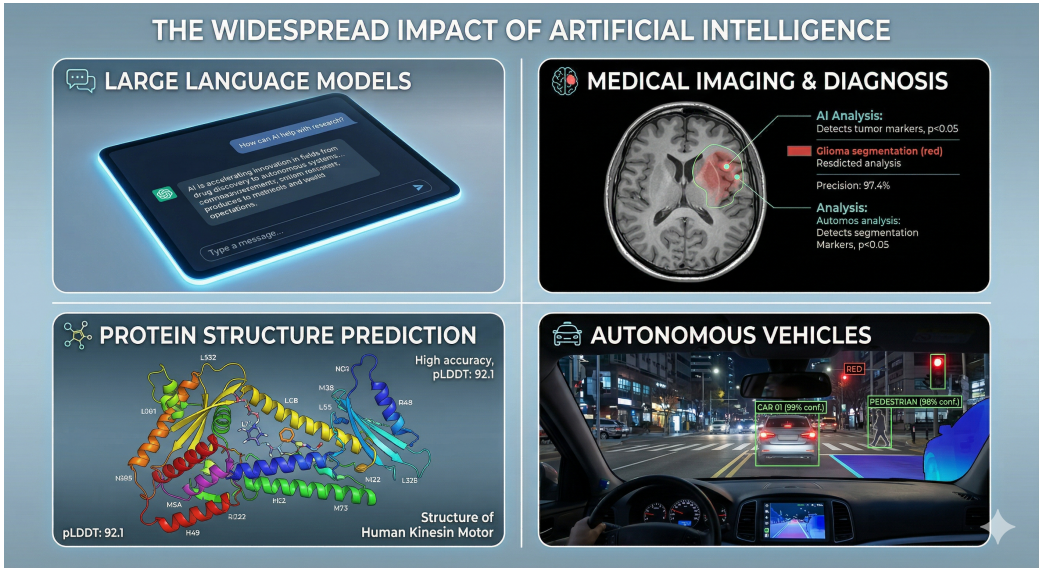
OPIT – Open Institute of Technology

lorenz.livi@gmail.com

Outline

Motivation

Why a theory of deep learning?



- Deep learning – differentiable architectures optimized with SGD-like dynamics – has achieved remarkable results across domains
- Yet, we lack a unifying theory that explains **why** it works and guides practitioners on **when** it will fail
- This is both a **risk** for large-scale deployment and an **opportunity** to build **foundational understanding** and **robust behaviour**

A dynamical perspective on deep learning

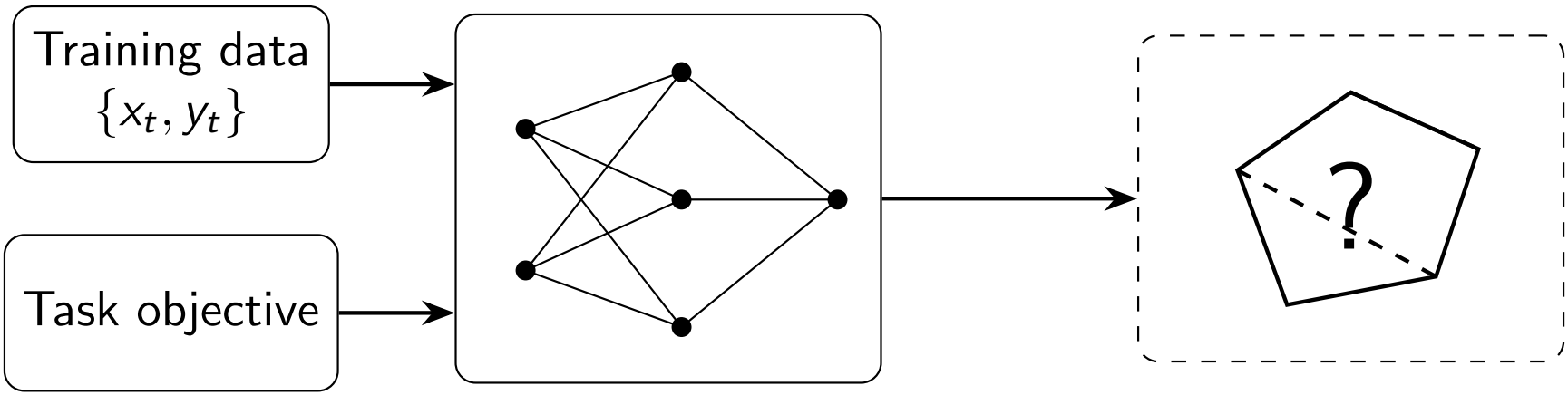
Deep learning systems are **inherently dynamical**:

- States evolve nonlinearly under input forcing
- Parameters evolve under SGD-like dynamics
- These two processes run simultaneously and interact

Yet, we lack a **unifying dynamical theory** of learning under SGD-like training across deep learning architectures:

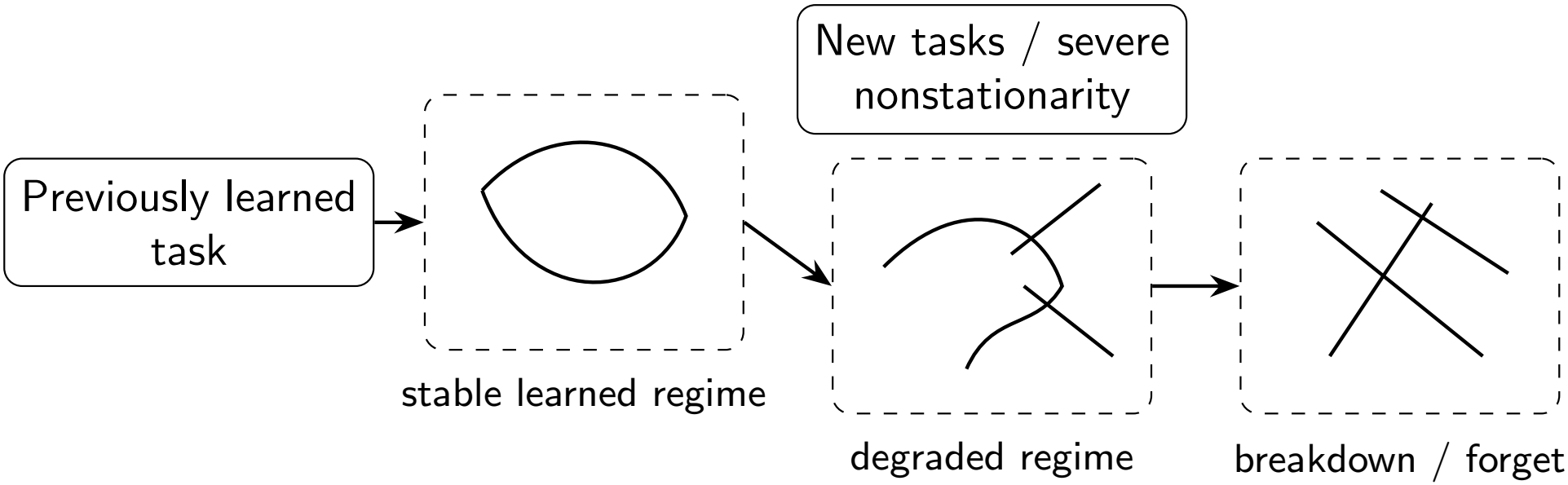
- How do state and parameter dynamics jointly organize into task-specific regimes?
- How do these regimes adapt or degrade under changing conditions?
- What governs the time scales of this **coupled process**?

Task formation: Is it just optimization?



- Training organizes the network into a **task-specific dynamical regime**
- **Question:** What is the internal formal representation of the task learned by the network? How does it emerge during training?

Forgetting: Is it just an optimization bottleneck?

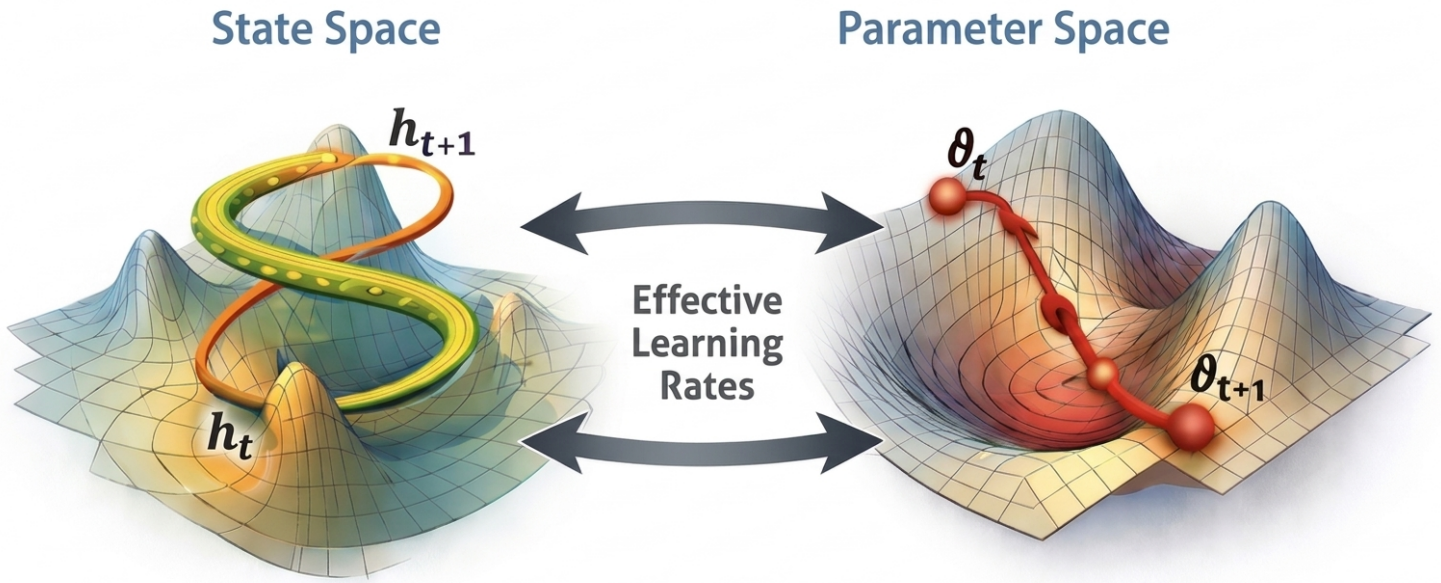


Question:
Why and how do previously learned dynamical regimes degrade under continual learning and severe nonstationarity? Can we predict when breakdown occurs at a mechanistic level?

Key research question

Can task formation, adaptation, and forgetting be explained as manifestations of one coupled dynamical process?

Key conceptual shift



- **Traditional view:** parameters determine model's behaviour through optimization
- **Proposed view:** learning forms a **coupled dynamical system** involving the joint action of state dynamics and parameter dynamics, governed by **time-scale interactions** – a structure common to all deep learning architectures trained with SGD-like dynamics

Research trajectory

Three works progressively establish the foundation for understanding the coupling between state and parameter dynamics

- **Time-scale coupling**:
  - Effective learning rates reveal the coupling
  - *Time-Scale Coupling Between States and Parameters in RNNs* [1]
- **Learnability theory**:
  - The decay of this coupling determines the limits of temporal learning
  - *Learnability Window in Gated Recurrent Neural Networks* [2]
- **Anti-collapse dynamics**:
  - Training self-organizes the time-scale spectrum to sustain learnability
  - *Anti-Collapse Dynamics and the Emergence of Multi-Time-Scale Learning* [3]

Background

(Quick) Related work

- Task representation in reservoir computing [4], [5]; memory and computational capability [6], [7], [8]
- Neural ODE [9], [10] and neural operators [11]
- State-space time scales in RNNs [12], [13]
- Control-theoretic ML [14]
- Dynamics of SGD [15] and heavy-tail gradient statistics [16], [17], [18]
- Continual learning [19]
- Explainable AI [20]

Why gated RNNs as a starting point?

Gated RNNs are already dynamical systems, the natural testbed for a dynamical theory of learning

State dynamics:

$$\begin{aligned} h_t &= (1 - s_t) \odot h_{t-1} + s_t \odot \tilde{h}_t, \\ \tilde{h}_t &= \tanh(W_h x_t + U_h h_{t-1} + b_h) \end{aligned} \tag{1}$$

- $s_t \in (0,1)^H$: time-varying gate controlling the state update
- Parameters shared across time, trained with SGD-like optimizers
- More complex architectures exist [13], [21]

SGD-like optimization

- Given trainable parameters θ and learning rate μ, the SGD update is

$$\theta_{r+1} = \theta_r - \mu \nabla_\theta \mathcal{L}(\theta_r), \qquad \mathcal{L} = \sum_{t=1}^{T} \mathcal{E}_t. \tag{2}$$

- Under an adaptive optimizer (e.g., Adam, AdamW, RMSprop),

$$\theta_{r+1} = \theta_r - \Lambda_r \nabla_\theta \mathcal{L}(\theta_r), \tag{3}$$

where $\Lambda_r = \mathrm{diag}(\lambda_{1,r}, \ldots, \lambda_{P,r}) \in \mathbb{R}^{P \times P}$ are adaptive learning rates [22], [23].

Jacobian products

- The total (per sequence) gradient of the loss with respect to the parameters can be written as

$$\nabla_\theta \mathcal{L} = \sum_{t=1}^{T} \frac{\partial \mathcal{E}_t}{\partial \theta} = \sum_{t=1}^{T} \frac{\partial \mathcal{E}_t}{\partial h_t} \sum_{\ell=1}^{t} \frac{\partial h_t}{\partial h_\ell} \frac{\partial h_\ell}{\partial \theta}. \tag{4}$$

- Define $J_j = \frac{\partial h_j}{\partial h_{j-1}}$ and $B_\ell(\theta) = \frac{\partial h_\ell}{\partial \theta}$. Substituting into Eq. (4),

$$\frac{\partial \mathcal{E}_t}{\partial \theta} = \delta_t^\top \sum_{\ell=1}^{t} \mathcal{M}_{t,\ell}\, B_\ell(\theta). \tag{5}$$

$$\mathcal{M}_{t,\ell} := \prod_{j=\ell+1}^{t} J_j, \tag{6}$$

$\mathcal{M}_{t,\ell}$ transports gradient signals across the temporal displacement $t - \ell$.

Effective learning rates

Time-scale coupling[1]

Main question:

How to demonstrate the state–parameter interaction mathematically?

Key insight:

- We already know that optimization affects state dynamics..

→ Show that states modulate **effective learning rates**, affecting parameter dynamics

- This creates a coupling between

$$\text{state dynamics} \leftrightarrow \text{parameter dynamics}$$

[1] (L. Livi, "Time-scale coupling between states and parameters in recurrent neural networks," *arXiv preprint arXiv:2508.12121*, 2025. DOI: 10.48550/arXiv.2508.12121. **url**: https://arxiv.org/abs/2508.12121)

First-order expansion of Jacobian products

- Decompose each J_j in $\prod_{j=\ell+1}^{t} J_j$ as follows:

$$J_j = A_j + \epsilon B_j$$

  - A_j: diagonal gate component
  - B_j: recurrent mixing and higher-order sensitivity
  - $\epsilon > 0$ is a perturbation coefficient

- First-order expansion (general formulation):

$$\prod_{j=1}^{n}(A_j+\varepsilon B_j) = \prod_{j=1}^{n} A_j + \varepsilon \underbrace{\sum_{m=1}^{n} \left(\prod_{j=1}^{m-1} A_j\right) B_m \left(\prod_{j=m+1}^{n} A_j\right)}_{\text{perturbative corrections}} + O(\varepsilon^2)$$

  - Based on the Fréchet derivative of a matrix product
  - Full discussion in [1]

Per-neuron, per-lag effective learning rates

- For instance, consider an RNN with the following gate:

$$s_t = \sigma(a_t^s) \in (0,1)^H, \qquad a_t^s = W_s x_t + U_s h_{t-1} + b_s. \tag{7}$$

- Write each one-step Jacobian in $\mathcal{M}_{t,\ell}$ as follows:

$$J_t = \underbrace{(I - S_t)}_{\text{diagonal leak}} + \underbrace{\widehat{E}}_{\text{gate sensitivity and mixed terms}}. \tag{8}$$

- Under first-order expansion of $\mathcal{M}_{t,\ell}$, the RNN model generates per-lag, per-neuron effective learning rates:

$$\mu_{t,\ell}^{(q)} = \mu\big(\gamma_{t,\ell}^{(0,q)} + \gamma_{t,\ell}^{(1,q)}\big), \tag{9}$$

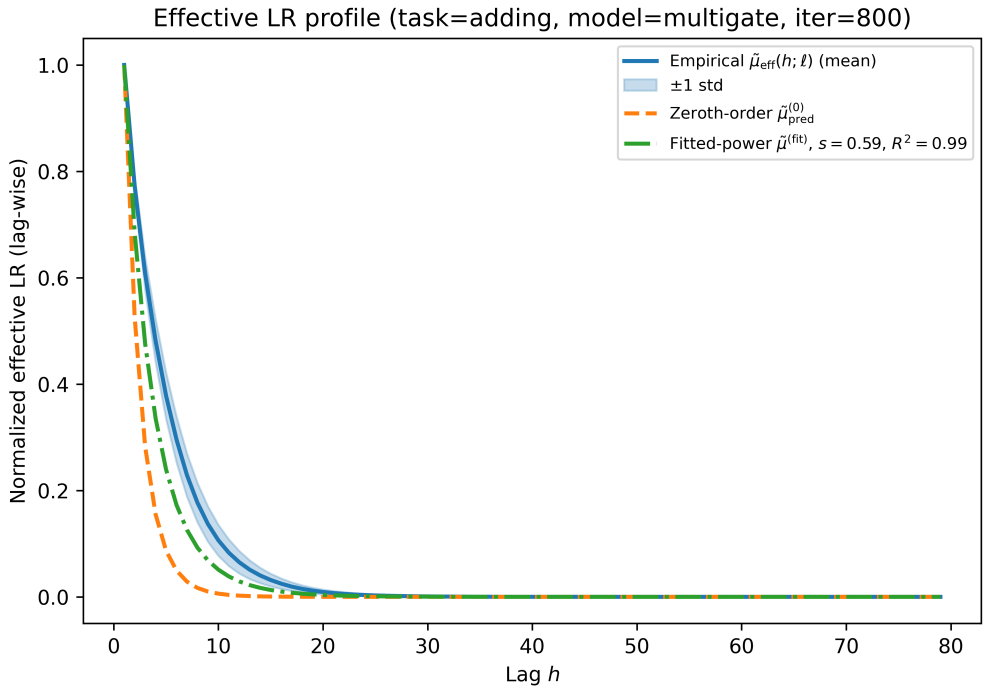
where $\gamma_{t,\ell}^{(0,q)} = \prod_{j=\ell+1}^{t}(1 - s_{j,q})$ and $\gamma_{t,\ell}^{(1,q)}$ are first-order *correction* terms.

Effective learning rates: Empirical evidence

Evidence that state dynamics influence optimization dynamics via

$$\mu_{t,\ell}^{(q)}$$

- Neuron-specific dynamics
- Lag-dependent learning rates
- Anisotropic update geometry



Learnability theory

From coupling to learnability[2]

The effective learning rates $\mu_{t,\ell}^{(q)}$ describe the coupling between state and parameter dynamics.

Their aggregate, i.e. the *envelope*

$$f(\ell) = \sum_{q=1}^{H} |\mu_{t,\ell}^{(q)}|$$

measures the total gradient signal available at temporal displacement ℓ.

Central question: given N training sequences and noisy gradients, up to which lag does $f(\ell)$ remain **detectable**?

[2] (L. Livi, "Learnability window in gated recurrent neural networks," *arXiv preprint arXiv:2512.05790*, 2025. DOI: 10.48550/arXiv.2512.05790. **url**: https://arxiv.org/abs/2512.05790)

Learnability at finite data

Temporal credit assignment is limited by a **signal-to-noise trade-off**:

- Signal: the envelope $f(\ell)$ decays with lag
- Noise: statistical fluctuations decay with N

Only lags where $f(\ell)$ exceeds noise are learnable

This induces a data-dependent horizon:

$$\mathcal{H}_N = \max\{\ell : f(\ell) \text{ detectable at sample size } N\}$$

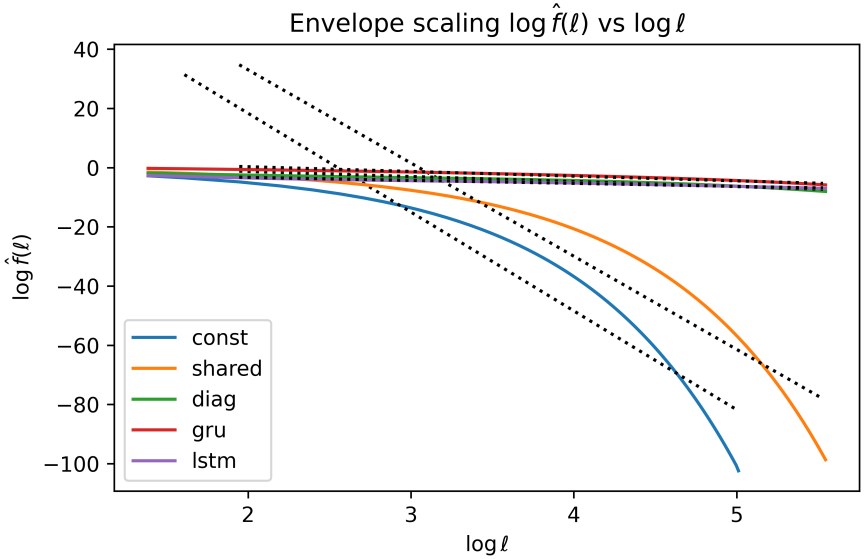
Scaling laws for the learnability window

The learnability theory predicts three canonical classes:

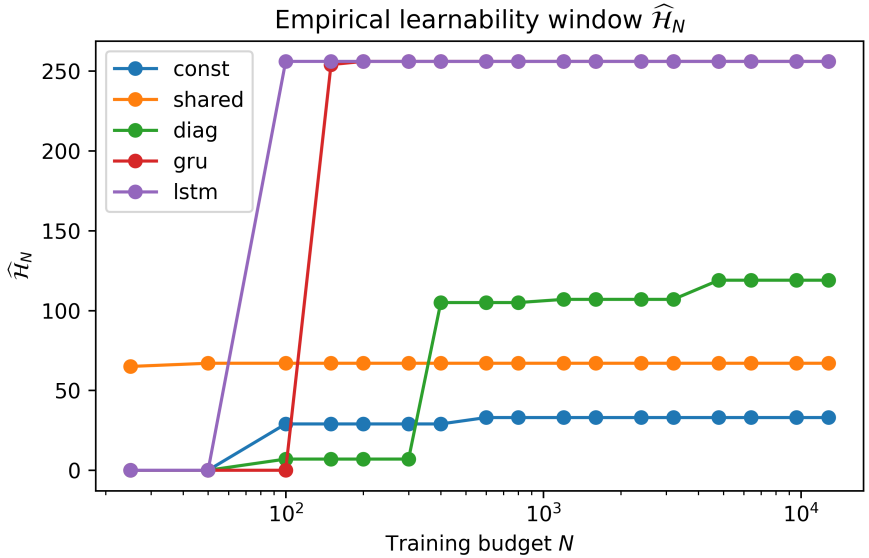
Envelope decay $f(\ell)$	Sample complexity $N(\ell)$	Learnability window $\mathcal{H}_N$
$f(\ell) \sim \lambda^{\ell}$	$N(\ell) \sim \lambda^{-\kappa_\alpha \ell}$	$\mathcal{H}_N \sim \dfrac{\log N}{\kappa_\alpha \log(1/\lambda)}$
$f(\ell) \sim \ell^{-\beta}$	$N(\ell) \sim \ell^{\kappa_\alpha \beta}$	$\mathcal{H}_N \sim N^{1/(\kappa_\alpha \beta)}$
$f(\ell) \sim c/\log(1+\ell)$	$N(\ell) \sim [\log(1+\ell)]^{\kappa_\alpha}$	$\mathcal{H}_N \sim \exp\left(\kappa\, N^{1/\kappa_\alpha}\right) - 1$

In all cases, smaller α (heavier tails) increases κ_α and compresses $\mathcal{H}_N$.

Empirical envelope decay and learnability windows



Envelope decay $f(\ell)$ across architectures.



Learnability window $\mathcal{H}_N$.

Slower envelope decay produces systematically larger learnability windows.

Limits of temporal learning and self-organization

- **Envelope decay is not fixed by architecture**
  - It emerges from state–parameter coupling during training
  - Varies across architectures, setups, and optimizers
- **Exponential forgetting is statistically unsustainable**
  - $f(\ell) \sim \lambda^{\ell} \Rightarrow N(\ell) \sim \lambda^{-\kappa_\alpha \ell}$ (exponential sample complexity)
  - $f(\ell) \sim \ell^{-\beta} \Rightarrow N(\ell) \sim \ell^{\kappa_\alpha \beta}$ (polynomial sample complexity)
  - Long-range learning becomes infeasible $\Rightarrow$ slower decay must emerge
- **Learning with SGD-like dynamics induces constrained self-organization**
  - The architecture–optimizer pair *tries* to develop slower envelopes to remain learnable
  - The achievable regime depends on the precise instance of the architecture–optimizer pair

Anticollapse of time scales

What determines the envelope?[3]

The learnability theory imposes a constraint: viable learning over large lags requires slow envelope decay

What determines the decay of $f(\ell)$?

Answer: the distribution of neuron time scales

- Each neuron has an effective time scale τ_q
- The **decay** of $f(\ell)$ is controlled by the **tail** of the distribution of time scales

[3] (L. Livi, "Anti-collapse dynamics and the emergence of multi-time-scale learning in recurrent neural networks," *Manuscript in preparation*, 2026)

Time-scale spectrum determines envelope decay

Let $p_\infty(\tau)$ be the distribution of neuron-wise time scales.

In the large-width network limit, the envelope takes a **Laplace-type form**:

$$f(\ell) \approx \int_0^\infty e^{-\ell/\tau}\, p_\infty(\tau)\, d\tau$$

The tail of $p_\infty(\tau)$ determines the decay of $f(\ell)$

- Concentrated spectrum $\Rightarrow$ fast (exponential) decay
- Heavy-tailed spectrum $\Rightarrow$ slow (power-law) decay

Tail geometry determines the envelope class

Via Tauberian theory [24], [25], each tail model of $p_\infty(\tau)$ implies a specific envelope class, and hence a learnability regime:

Tail of $p_\infty(\tau)$	**Decay of $f(\ell)$**	**Learnability $\mathcal{H}_N$**
log-normal	λ^{ℓ}	$\dfrac{\log N}{\kappa_\alpha \log(1/\lambda)}$
$\tau^{-1-\beta}$	$\ell^{-\beta}$	$N^{1/(\kappa_\alpha\,\beta)}$
$\dfrac{1}{\tau(\log\tau)^{1+\gamma}}$	$(\log\ell)^{-\gamma}$	$\exp\!\left(\kappa\, N^{1/\kappa_\alpha}\right)$

The correspondence is rigorous: **heavier tails produce slower envelopes and larger learnability windows**.

What shapes the time-scale spectrum?

Empirical picture:

- Broad time-scale spectra tend to co-occur with heavy-tailed fluctuations
- Flexible gated architectures with adaptive optimizers tend to develop broad spectra with heavy-tailed fluctuations
- Constrained architectures with simple optimizers tend to produce synchronized time scales and near-Gaussian fluctuations

Interpretation:

- The time-scale spectrum evolves under the joint action of state and parameter dynamics (**coupled dynamics**)
- The architecture–optimizer pair and specific experimental setup determines the accessible dynamical regime realized during training

Ongoing work: Modeling directions

Key insight: the anti-collapse mechanism arises from competition between deterministic drift and stochastic noise

Time scales evolve as a stochastic process driven by three forces:

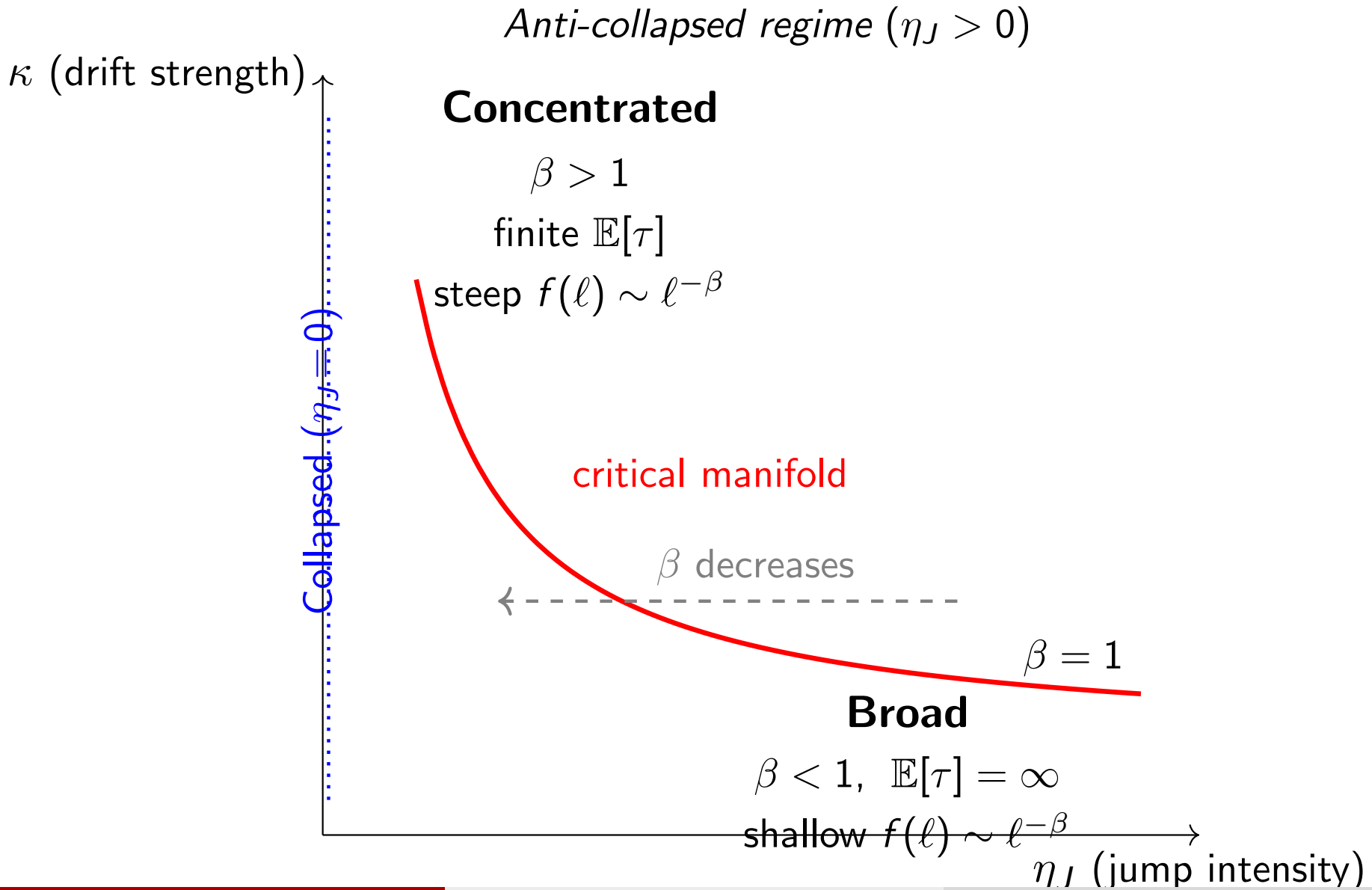
- **Deterministic drift** – time-scale collapse tendency
- **Continuous diffusion** – baseline Gaussian noise component
- **Jump component** – enriches the noise with heavy-tailed (α-stable) fluctuations observed in deep learning

The presence or absence of jumps selects the dynamical regime:

No jumps $\Rightarrow$ log-normal spectrum $\rightarrow$ exponential envelope

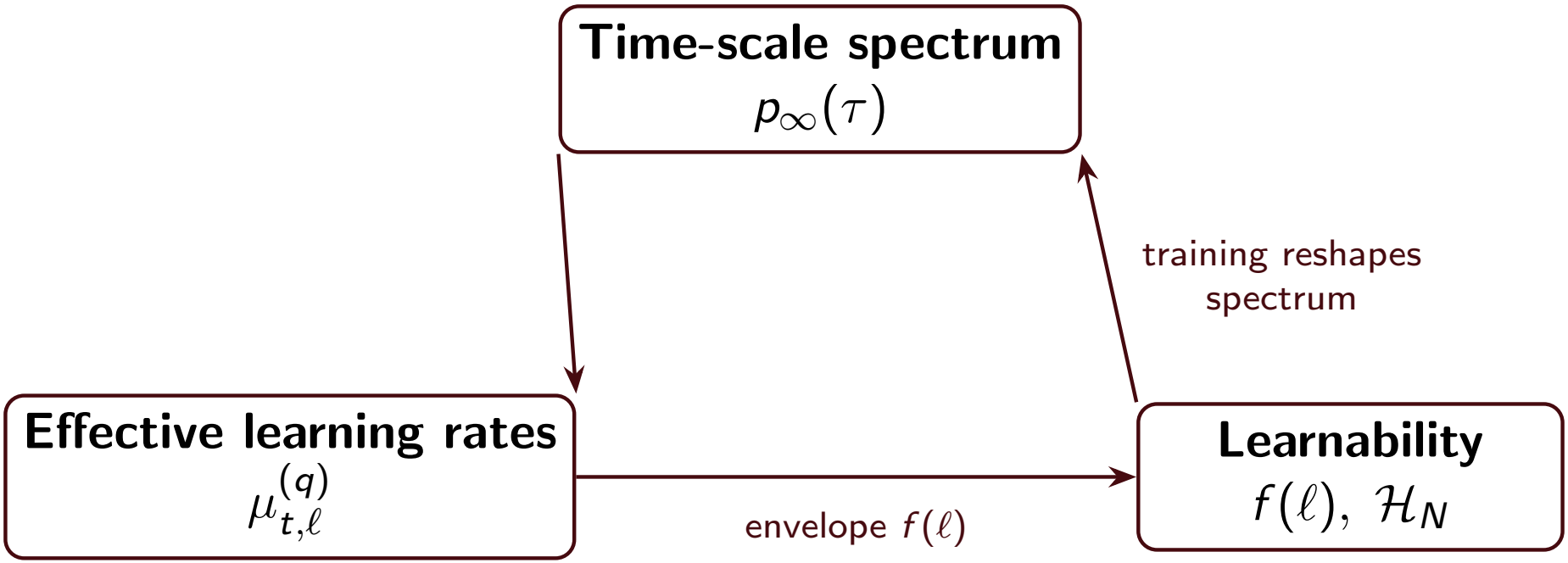
Jumps present $\Rightarrow$ power-law tail $\rightarrow$ algebraic envelope

Preliminary phase diagram



Implications and future directions

Unified picture



$$p_\infty(\tau) \xrightarrow{\text{Laplace}} f(\ell) \xrightarrow{\text{learnability}} \mathcal{H}_N$$

Take-home message

Learning and temporal organization of time scales are facets of **one coupled dynamical process**

- Gates do not just control state dynamics: they shape **how the network learns**, coupling state and parameter dynamics **across time scales**
- This coupling imposes **sharp limits** on what can be learned from finite data over long time horizons
- The tail of the **time-scale spectrum** determines learnability

$$p_\infty(\tau) \xrightarrow{\text{Laplace}} f(\ell) \xrightarrow{\text{detection}} \mathcal{H}_N$$

Future research directions

- **Training by design**
  Constrain training to remain within a computable region $\mathcal{N}^*$ of control space where long-horizon computation is provably reliable and controllable – separating dynamical *feasibility* from task *optimality*
- **Depth as time**
  Extend the theory to deep feed-forward, convolutional networks and transformers, trading time for layer-depth
- **Fast–slow mechanistic models of learning**

$$\underbrace{h_t}_{\text{fast state dynamics}} \qquad \underbrace{\theta_k}_{\text{slow parameter dynamics}}$$

A foundation for mechanistic models of what the network learns and how it forgets

Speculation

Does deep learning self-organize similarly to SOC?

- Concentrated spectra carry exponential sample complexity: a **pressure to escape collapse**
- When the architecture–optimizer pair has sufficient capacity, training dynamics appear to **self-organize** into the anti-collapsed regime without explicit tuning
- This is reminiscent of **self-organized criticality**: finite-sample learnability constraints act as a **slow driving force** pushing the system toward broad, heavy-tailed spectra

Not a claim that SGD training **is** *SOC,*
but the structural correspondence may not be accidental

Thank you!

I look forward to your questions

ORCID — Google Scholar — lorenz.livi@gmail.com

Fast–slow dynamical formulation (conceptual)

Learning dynamics can be written as a coupled fast–slow system

$$h_{t+1} = \mathcal{F}(h_t, x_t, \theta_k)$$

$$\theta_{k+1} = \theta_k - \eta \nabla_\theta \mathcal{L}(h_t, \theta_k)$$

Fast variables:

$$h_t \quad \text{(state dynamics)}$$

Slow variables:

$$\theta_k \quad \text{(parameter dynamics)}$$

Analyzing this system may reveal

- dynamical regimes associated with learned tasks
- stability and transitions during continual learning
- mechanisms underlying catastrophic forgetting

Envelope as a classical Laplace transform (backup)

Introduce the reciprocal variable $u = \tau^{-1}$ with density

$$p_U(u) = u^{-2}\, p_\infty(1/u).$$

Then the envelope becomes a standard Laplace transform:

$$f(\ell) = \int_0^\infty e^{-\ell\, u}\, p_U(u)\, du.$$

Classical Tauberian theory [24], [25] then relates

$$p_U(u) \text{ near } u = 0 \quad \longleftrightarrow \quad f(\ell) \text{ as } \ell \to \infty.$$

Since $u \to 0$ corresponds to $\tau \to \infty$, the large-ℓ decay of the envelope is governed by the **heavy tail** of the time-scale distribution $p_\infty(\tau)$.

References I

[1] L. Livi, "Time-scale coupling between states and parameters in recurrent neural networks," *arXiv preprint arXiv:2508.12121*, 2025. DOI: 10.48550/arXiv.2508.12121. **url**: https://arxiv.org/abs/2508.12121.

[2] L. Livi, "Learnability window in gated recurrent neural networks," *arXiv preprint arXiv:2512.05790*, 2025. DOI: 10.48550/arXiv.2512.05790. **url**: https://arxiv.org/abs/2512.05790.

[3] L. Livi, "Anti-collapse dynamics and the emergence of multi-time-scale learning in recurrent neural networks," *Manuscript in preparation*, 2026.

References II

[4] D. Sussillo **and** O. Barak, "Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks," *Neural Computation*, **jourvol** 25, **number** 3, **pages** 626–649, 2013. DOI: 10.1162/NECO_a_00409.

[5] A. Ceni, P. Ashwin, L. Livi **and** C. Postlethwaite, "The echo index and multistability in input-driven recurrent neural networks," *Physica D*, **jourvol** 412, 2020. DOI: 10.1016/j.physd.2020.132609.

[6] J. Dambre, D. Verstraeten, B. Schrauwen **and** S. Massar, "Information processing capacity of dynamical systems," *Scientific Reports*, **jourvol** 2, 2012. DOI: 10.1038/srep00514.

[7] A. S. Charles, H. L. Yap **and** C. J. Rozell, "Short-term memory capacity in networks via the restricted isometry property," *Neural Computation*, **jourvol** 26, **number** 6, **pages** 1198–1235, 2014. DOI: 10.1162/NECO_a_00590.

References III

[8] L. Grigoryeva **and** J. Ortega, "Universal discrete-time reservoir computers with stochastic inputs and linear readouts using non-homogeneous state-affine systems," *Journal of Machine Learning Research*, **jourvol** 19, **number** 24, 2018.

[9] P. Kidger, J. Morrill, J. Foster **and** T. Lyons, "Neural controlled differential equations for irregular time series," *Advances in Neural Information Processing Systems*, **jourvol** 33, **pages** 6696–6707, 2020.

[10] R. T. Q. Chen, Y. Rubanova, J. Bettencourt **and** D. K. Duvenaud, "Neural ordinary differential equations," *Advances in neural information processing systems*, **jourvol** 31, 2018.

[11] N. Kovachki **andothers**, "Neural operator: Learning maps between function spaces with applications to PDEs," *Journal of Machine Learning Research*, **jourvol** 24, **number** 89, **pages** 1–97, 2023.

References IV

[12] C. Tallec **and** Y. Ollivier, "Can recurrent neural networks warp time?" **in***International Conference on Learning Representations* 2018.

[13] T. Can, K. Krishnamurthy **and** D. J. Schwab, "Gating creates slow modes and controls phase-space complexity in GRUs and LSTMs," **in***Proceedings of The First Mathematical and Scientific Machine Learning Conference* **jourser** Proceedings of Machine Learning Research, **volume** 107, 2020, **pages** 476–511. **url**: https://proceedings.mlr.press/v107/can20a.html.

[14] C. Amo Alonso, J. Sieber **and** M. N. Zeilinger, "State space models as foundation models: A control theoretic overview," **in***2025 American Control Conference (ACC)* 2025, **pages** 146–153. DOI: 10.23919/ACC63710.2025.11107969.

References V

[15] T. Bonnaire, D. Ghio, K. Krishnamurthy, F. Mignacco, A. Yamamura **and** G. Biroli, "High-dimensional non-convex landscapes and gradient descent dynamics," *Journal of Statistical Mechanics: Theory and Experiment*, **jourvol** 2024, **number** 10, **page** 104 004, 2024. DOI: 10.1088/1742-5468/ad2929. **url**: https://doi.org/10.1088/1742-5468/ad2929.

[16] T. H. Nguyen, U. Şimşekli, M. Gürbüzbalaban **and** G. Richard, "First exit time analysis of stochastic gradient descent under heavy-tailed gradient noise," **in***Advances in Neural Information Processing Systems* **volume** 32, 2019.

References VI

[17] M. Barsbey, M. Sefidgaran, M. A. Erdogdu, G. Richard **and** U. Şimşekli, "Heavy tails in SGD and compressibility of overparametrized neural networks," **in***Advances in Neural Information Processing Systems* **volume** 34, 2021, **pages** 29 364–29 378.

[18] U. Simsekli, L. Sagun **and** M. Gurbuzbalaban, "A tail-index analysis of stochastic gradient noise in deep neural networks," **in***International Conference on Machine Learning* 2019, **pages** 5827–5837.

[19] G. M. Van de Ven, N. Soures **and** D. Kudithipudi, "Continual learning and catastrophic forgetting," *arXiv preprint arXiv:2403.05175*, 2024.

References VII

[20] A. Barredo Arrieta **andothers**, "Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI," *Information Fusion*, **jourvol** 58, **pages** 82–115, 2020, ISSN: 1566-2535. DOI: https://doi.org/10.1016/j.inffus.2019.12.012. **url**: https://www.sciencedirect.com/science/article/pii/S1566253519308103.

[21] A. Ceni, "Random orthogonal additive filters: A solution to the vanishing/exploding gradient of deep neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, **jourvol** 36, **number** 6, **pages** 10 794–10 807, 2025. DOI: 10.1109/TNNLS.2025.3538924.

[22] D. Kingma **and** J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

References VIII

[23] S. Ruder, "An overview of gradient descent optimization algorithms," *arXiv preprint arXiv:1609.04747*, 2016.

[24] W. Feller, *An Introduction to Probability Theory and Its Applications. Vol. II*, 2nd. New York: John Wiley & Sons, 1971.

[25] N. H. Bingham, C. M. Goldie **and** J. L. Teugels, *Regular Variation* (Encyclopedia of Mathematics and Its Applications). Cambridge: Cambridge University Press, 1989, **volume** 27.