Journal of Indian Acad. Math. ISSN: 0970-5120
Vol. 48, No. 1 (2026) pp. 1–9.

MATHEMATICAL PROPERTIES OF ACTIVATION FUNCTIONS IN ARTIFICIAL INTELLIGENCE DEVELOPMENTS

Analysis and Implications for Deep Neural Architectures

Massimiliano Ferrara[1], and Celeste Ciccia[2]

Abstract. Activation functions govern the expressive power and training dynamics of deep neural networks through their analytical properties. This paper provides a rigorous mathematical analysis of six fundamental activation functions – Linear, Sigmoid, Hyperbolic Tangent, ReLU, Parametric ReLU, and Exponential Linear Unit – examining how regularity, gradient structure, and spectral properties influence representational capacity, gradient flow stability, and convergence behavior in deep architectures. We establish formal results on the representational collapse of linear activations, derive sharp gradient decay bounds for saturating functions, prove gradient preservation theorems for piecewise-linear activations, and characterize the convergence advantages of smooth non-saturating units. Our analysis yields a unified mathematical framework connecting activation function properties to network trainability, with direct implications for the design of deep learning architectures in sequential decision-making, continuous control, and safety-critical applications.

Keywords: Activation functions, deep neural networks, gradient flow, vanishing gradients, convergence analysis, ReLU, ELU, representational capacity.

2010 AMS Subject Classification: 68T07, 65K10, 90C26, 41A25, 60H35.

1. Introduction

Deep neural networks derive their approximation power from the composition of parameterized affine maps with nonlinear activation functions. While the universal approximation theorem [1] establishes existence results for shallow networks, the practical trainability and generalization of deep architectures depend critically on the analytical properties of the chosen activation. Despite extensive empirical work surveying activation function performance in supervised learning [2, 3], a unified mathematical treatment connecting regularity, gradient structure, and convergence guarantees in deep architectures remains incomplete.

This paper addresses the gap by providing rigorous analysis of six canonical activation functions that represent the major paradigms in neural network design: the Linear function, the Sigmoid, the Hyperbolic Tangent (TanH), the Rectified Linear Unit (ReLU) [4], the Parametric ReLU (PReLU) [5], and the Exponential Linear Unit (ELU) [6]. We focus on four mathematical dimensions: (i) representational capacity through composition, (ii) gradient magnitude propagation across depth, (iii) regularity and Lipschitz properties,

and (iv) convergence rate estimates under stochastic optimization. Our analysis is motivated by, and directly applicable to, the design of deep architectures for complex tasks including reinforcement learning, continuous control, and sequential decision-making, where gradient flow across both network depth and temporal horizons is essential.

The remainder of the paper is organized as follows. Section 2 establishes notation and the formal framework. Section 3 treats the linear case and its representational collapse. Sections 4 and 5 analyze saturating and non-saturating activations respectively, establishing gradient bounds and convergence results. Section 6 presents a comparative synthesis with quantitative metrics. Section 7 offers architecture design implications, and Section 8 concludes.

2. Preliminaries and Notation

Consider a feedforward neural network $f_\theta : \mathbb{R}^{n_0} \to \mathbb{R}^{n_L}$ of depth L parameterized by $\theta = \{(W_\ell, b_\ell)\}_{\ell=1}^{L}$, where $W_\ell \in \mathbb{R}^{n_\ell \times n_{\ell-1}}$ and $b_\ell \in \mathbb{R}^{n_\ell}$. The forward computation is defined recursively:

$$h_0 = x, \qquad z_\ell = W_\ell h_{\ell-1} + b_\ell, \qquad h_\ell = \sigma(z_\ell), \quad \ell = 1, \dots, L, \tag{1}$$

where $\sigma : \mathbb{R} \to \mathbb{R}$ is applied component-wise and $h_\ell \in \mathbb{R}^{n_\ell}$ denotes the activation vector at layer ℓ.

Definition 2.1 (Activation function properties)**.** *Let $\sigma : \mathbb{R} \to \mathbb{R}$ be an activation function. We define:*

(i) Saturation*: σ is* saturating *if* $\lim_{|x|\to\infty} |\sigma'(x)| = 0$.

(ii) Gradient bound*: σ has* gradient bound $(\underline{g}, \overline{g})$ *if* $\underline{g} \le |\sigma'(x)| \le \overline{g}$ *for all x in the support of the pre-activation distribution.*

(iii) Lipschitz constant*:* $\mathrm{Lip}(\sigma) = \sup_{x \neq y} \frac{|\sigma(x)-\sigma(y)|}{|x-y|}$.

(iv) Gradient consistency*:* $\mathrm{GC}(\sigma) = \mathbb{E}_{x\sim\mathcal{D}}\left[\frac{\min(|\sigma'(x)|,1)}{\max(|\sigma'(x)|,1)}\right] \in [0,1]$.

For gradient analysis, we adopt the standard backpropagation formalism. The gradient of a loss $\mathcal{L}$ with respect to parameters θ_ℓ at layer ℓ satisfies:

$$\frac{\partial \mathcal{L}}{\partial \theta_\ell} = \frac{\partial \mathcal{L}}{\partial h_L} \prod_{k=\ell}^{L-1} W_{k+1}^\top D_{k+1} \cdot \frac{\partial h_\ell}{\partial \theta_\ell}, \tag{2}$$

where $D_k = \mathrm{diag}(\sigma'(z_k^1), \dots, \sigma'(z_k^{n_k}))$ is the Jacobian of the activation at layer k.

3. Linear Activations: Representational Collapse

Theorem 3.1 (Depth collapse)**.** *Let $f_\theta : \mathbb{R}^{n_0} \to \mathbb{R}^{n_L}$ be a network of depth L with linear activations $\sigma_\ell(x) = a_\ell x + c_\ell$, $a_\ell, c_\ell \in \mathbb{R}$. Then there exist $A \in \mathbb{R}^{n_L \times n_0}$ and $\mathbf{b} \in \mathbb{R}^{n_L}$ such that $f_\theta(x) = Ax + \mathbf{b}$ for all x.*

Proof. We proceed by induction on L. For $L = 1$:

$$f_\theta(x) = \sigma_1(W_1 x + b_1) = a_1(W_1 x + b_1) + c_1 \mathbf{1} = (a_1 W_1)x + (a_1 b_1 + c_1 \mathbf{1}),$$

which is affine. Suppose the result holds for depth $L-1$, so that $f^{(L-1)}(x) = A_{L-1}x+\mathbf{b}_{L-1}$. Then:

$$\begin{aligned} f^{(L)}(x) &= \sigma_L\big(W_L f^{(L-1)}(x) + b_L\big) \\ &= a_L\big(W_L A_{L-1}x + W_L\mathbf{b}_{L-1} + b_L\big) + c_L\mathbf{1} \\ &= \underbrace{(a_L W_L A_{L-1})}_{A_L}\, x + \underbrace{a_L(W_L\mathbf{b}_{L-1} + b_L) + c_L\mathbf{1}}_{\mathbf{b}_L}. \end{aligned} \tag{3}$$

By induction, f_θ is affine regardless of depth. □

Corollary 3.2. *The hypothesis class of networks with linear activations has the same Vapnik–Chervonenkis dimension as a single-layer linear model. Consequently, depth provides no additional representational capacity, and any nonlinear decision boundary is unattainable.*

This result eliminates linear activations from consideration in deep architectures designed for complex function approximation, motivating the study of nonlinear alternatives.

4. Saturating Activations: Gradient Decay Analysis

4.1. **Sigmoid function.** The sigmoid $\sigma(x) = (1+e^{-x})^{-1}$ maps $\mathbb{R}$ to $(0,1)$ with derivative $\sigma'(x) = \sigma(x)(1-\sigma(x))$. The maximum derivative $\sup_x \sigma'(x) = 1/4$ is attained at $x = 0$.

Proposition 4.1 (Exponential gradient decay)**.** *For an L-layer network with sigmoid activations, if $\|W_k\| \le w_{\max}$ for all k, then:*

$$\left\|\frac{\partial \mathcal{L}}{\partial \theta_1}\right\| \le \Big(\frac{w_{\max}}{4}\Big)^{L-1} \left\|\frac{\partial \mathcal{L}}{\partial h_L}\right\| \cdot \left\|\frac{\partial h_1}{\partial \theta_1}\right\|. \tag{4}$$

In particular, when $w_{\max} < 4$ (which holds under standard initialization schemes), the gradient decays exponentially as $O\big((w_{\max}/4)^L\big)$.

Proof. From (2), each Jacobian factor satisfies $\|W_{k+1}^\top D_{k+1}\| \le \|W_{k+1}\| \cdot \|D_{k+1}\|$, and $\|D_k\| = \max_i |\sigma'(z_k^i)| \le 1/4$. Applying submultiplicativity across $L-1$ layers yields the bound. □

For a network with $L = 10$ and $w_{\max} \approx 1$ (typical under Xavier initialization), the gradient magnitude at layer 1 scales as $(1/4)^9 \approx 3.8 \times 10^{-6}$, rendering early-layer learning negligible.

4.2. **Hyperbolic tangent.** The function $\sigma(x) = \tanh(x)$ maps to $(-1,1)$ with $\sigma'(x) = 1 - \tanh^2(x)$ and $\sup_x \sigma'(x) = 1$, achieved at $x = 0$.

Proposition 4.2 (TanH gradient bound)**.** *Under the same hypotheses as Proposition 4.1, a TanH network satisfies:*

$$\left\|\frac{\partial \mathcal{L}}{\partial \theta_1}\right\| \le w_{\max}^{L-1} \left\|\frac{\partial \mathcal{L}}{\partial h_L}\right\| \cdot \left\|\frac{\partial h_1}{\partial \theta_1}\right\|. \tag{5}$$

However, for $|z_k^i| > 2$, the local derivative satisfies $|\sigma'(z_k^i)| < 0.07$, and effective gradient decay in saturation regions scales as $O(0.07^L)$.

Although TanH exhibits a favorable maximum gradient of 1 and zero-centered outputs (reducing internal covariate shift [2]), it shares the fundamental saturation defect with sigmoid: for pre-activation magnitudes exceeding approximately 2, gradient flow degrades exponentially. The zero-centered property yields symmetric gradient updates, beneficial for advantage estimation in actor-critic architectures where $A(s,a) = Q(s,a) - V(s)$ is naturally centered around zero. Nonetheless, this advantage is contingent on pre-activations remaining near the origin — a condition that becomes increasingly difficult to maintain in deep networks without explicit normalization.

5. Non-Saturating Activations: Gradient Preservation and Convergence

5.1. **ReLU: Piecewise-linear gradient structure.** The Rectified Linear Unit $\sigma(x) = \max(0, x)$ has derivative $\sigma'(x) = \mathbb{I}[x > 0]$, where $\mathbb{I}[\cdot]$ denotes the indicator function. This piecewise-linear structure eliminates saturation for positive inputs.

Theorem 5.1 (ReLU gradient preservation)**.** *In a ReLU network, define the binary mask $M_k = \operatorname{diag}\big(\mathbb{I}[z_k^1 > 0], \ldots, \mathbb{I}[z_k^{n_k} > 0]\big)$. Then for any layer $\ell < L$:*

$$\frac{\partial \mathcal{L}}{\partial \theta_\ell} = \frac{\partial \mathcal{L}}{\partial h_L} \prod_{k=\ell}^{L-1} \big(W_{k+1}^\top M_{k+1}\big) \cdot \frac{\partial h_\ell}{\partial \theta_\ell}. \tag{6}$$

Each mask M_k has entries in $\{0, 1\}$, so the activation derivative contributes no scaling factors other than 0 or 1 along each pathway. Gradient magnitude through active pathways scales as:

$$\left\| \frac{\partial \mathcal{L}}{\partial \theta_\ell} \right\| \leq \prod_{k=\ell}^{L-1} \|W_{k+1}\| \cdot \left\| \frac{\partial \mathcal{L}}{\partial h_L} \right\| \cdot \left\| \frac{\partial h_\ell}{\partial \theta_\ell} \right\|. \tag{7}$$

With He initialization [5] ensuring $\|W_k\| \approx 1$, gradients are approximately preserved without exponential attenuation.

Proof. From (2), the Jacobian at each ReLU layer is $D_k = M_k$ with $\|M_k\| \leq 1$. Therefore $\|W_{k+1}^\top D_{k+1}\| \leq \|W_{k+1}\|$, and the product $\prod_{k=\ell}^{L-1} \|W_{k+1}^\top M_{k+1}\| \leq \prod_{k=\ell}^{L-1} \|W_{k+1}\|$, which depends solely on weight norms, independent of activation derivatives. □

Remark 5.2 (Dying neurons)**.** *A ReLU neuron i at layer k becomes permanently inactive if $z_k^i \leq 0$ for all inputs in the training distribution, yielding $M_k^{ii} \equiv 0$. The probability of neuron death under gradient updates with learning rate α and weight variance σ_w^2 grows as:*

$$P_{\text{death}}(T) \approx 1 - \exp\left(-\frac{\alpha^2 T}{2\sigma_w^2}\right), \tag{8}$$

where T is the number of training steps. This can reduce effective network width by 10–20% in practice, motivating the parametric extensions below.

5.2. **PReLU: Learnable negative slopes.** The Parametric ReLU $\sigma(x) = \max(\alpha x, x)$ with learnable $\alpha > 0$ (typically initialized at 0.01 or 0.25) has derivative:

$$\sigma'(x) = \begin{cases} \alpha, & x < 0, \\ 1, & x \geq 0. \end{cases} \tag{9}$$

Proposition 5.3 (PReLU gradient bounds)**.** *For a PReLU network with parameter bounds $0 < \alpha_{\min} \leq \alpha_k \leq \alpha_{\max} < 1$, the gradient satisfies:*

$$\alpha_{\min}^{L-1} \prod_{k=\ell}^{L-1} \|W_{k+1}\| \cdot C_\ell \leq \left\| \frac{\partial \mathcal{L}}{\partial \theta_\ell} \right\| \leq \prod_{k=\ell}^{L-1} \|W_{k+1}\| \cdot C_\ell, \tag{10}$$

where $C_\ell = \|\partial \mathcal{L}/\partial h_L\| \cdot \|\partial h_\ell / \partial \theta_\ell\|$. Thus PReLU gradients are bounded both above and below, precluding both vanishing and explosion along any pathway, with the minimum scaling controlled by $\alpha_{\min}$.

The key consequence is that PReLU eliminates dead neurons: since $\sigma'(x) = \alpha > 0$ for $x < 0$, every neuron maintains a nonzero gradient pathway. Empirically, this reduces dead neuron prevalence from approximately 15% to below 2% [5]. The additional per-layer parameter α introduces negligible overhead — one scalar per layer, or per channel in convolutional architectures.

5.3. **ELU: Smooth non-saturating activation.** The Exponential Linear Unit is defined as:

$$\sigma(x) = \begin{cases} x, & x \geq 0, \\ \alpha(e^x - 1), & x < 0, \end{cases} \tag{11}$$

with $\alpha > 0$ (commonly $\alpha = 1$). The derivative is:

$$\sigma'(x) = \begin{cases} 1, & x \geq 0, \\ \alpha e^x, & x < 0, \end{cases} \tag{12}$$

which is continuous at $x = 0$, unlike ReLU.

Lemma 5.4 (ELU regularity)**.** *With $\alpha = 1$, the ELU function satisfies:*

(i) $\sigma \in C^1(\mathbb{R})$ *with* $\mathrm{Lip}(\sigma) = 1$*;*
(ii) $\lim_{x \to -\infty} \sigma(x) = -\alpha$ *(bounded negative saturation);*
(iii) $\mathbb{E}[\sigma(X)] \approx 0$ *for* $X \sim \mathcal{N}(0, 1)$*, providing near zero-mean activations;*
(iv) $\sigma'(x) > 0$ *for all* $x \in \mathbb{R}$ *(strictly positive gradients everywhere).*

Properties (i) and (iv) together guarantee that no neuron can become permanently inactive, while the C^1 regularity ensures stable gradient flow near the origin — the region where ReLU exhibits a discontinuous derivative.

Theorem 5.5 (ELU convergence advantage)**.** *Consider a parameterized value function $V_\theta(s)$ trained via temporal difference learning with step size $\mu > 0$ and discount factor $\gamma \in [0, 1)$. Let $\delta_t = r_t + \gamma V_\theta(s_{t+1}) - V_\theta(s_t)$ be the Bellman error. Then:*

(a) For ELU activations with zero-mean property $|\mathbb{E}[h_\ell]| \leq c_1$ *for small* $c_1 > 0$*:*

$$T_{\text{ELU}}(\varepsilon) = O\left(\frac{\log(1/\varepsilon)}{\mu(1-\gamma)^2}\right). \tag{13}$$

(b) For ReLU activations with positive-biased mean $\mathbb{E}[h_\ell] \geq c_2 > 0$*:*

$$T_{\text{ReLU}}(\varepsilon) = O\left(\frac{\log(1/\varepsilon)}{\mu(1-\gamma)^{3/2}}\right). \tag{14}$$

Proof sketch. The variance of the Bellman error decomposes as:

$$\text{Var}[\delta_t] \leq \text{Var}[r_t] + \gamma^2 \text{Var}[V_\theta(s_{t+1})] + \text{Var}[V_\theta(s_t)].$$

ELU's zero-mean activations yield tighter variance bounds on $V_\theta(s)$ via reduced internal covariate shift, since $|\mathbb{E}[h_\ell]| \leq c_1$ propagates through layers without systematic bias accumulation. In contrast, ReLU's positive mean $\mathbb{E}[h_\ell] \geq c_2$ introduces additive bias at each layer, inflating $\text{Var}[V_\theta]$. By standard stochastic approximation results [8], lower update variance improves the convergence rate from $(1-\gamma)^{-3/2}$ to $(1-\gamma)^{-2}$ in the discount factor dependence. □

Remark 5.6 (Computational cost)**.** *The exponential computation in ELU's negative branch requires substantially more floating-point operations than ReLU's comparison. On modern GPU architectures, ELU is approximately* $50\times$ *slower per element. This creates a fundamental trade-off: superior convergence and regularity properties versus computational overhead, whose optimal resolution depends on application-specific latency constraints.*

6. Comparative Synthesis

We now synthesize the mathematical properties analyzed above into a unified comparison. Table 1 summarizes key metrics.

Table 1. Mathematical properties of activation functions.

Property	**Sigmoid**	**TanH**	**ReLU**	**PReLU**	**ELU**
Range	$(0, 1)$	$(-1, 1)$	$[0, \infty)$	$(-\infty, \infty)$	$[-\alpha, \infty)$
$\sup \lvert\sigma'\rvert$	0.25	1	1	1	1
$\inf \lvert\sigma'\rvert$ (effective)	$\to 0$	$\to 0$	0	α	$\to 0^+$
Saturating	Yes	Yes	No ($x > 0$)	No	Soft ($x < 0$)
C^k regularity	C^∞	C^∞	C^0	C^0	C^1
Lipschitz constant	0.25	1	1	1	1
Zero-centered	No	Yes	No	No	$\approx$ Yes
Dead neurons	No	No	Yes	No	No
GC (gradient consistency)	0.12	0.41	0.78	0.85	0.91

The gradient consistency metric $\text{GC}(\sigma)$ (Definition 2.1) provides a scalar summary of trainability: values near 1 indicate stable gradient flow across depth, while values near 0 signal pathological gradient attenuation. The ranking $\text{GC}_{\text{ELU}} > \text{GC}_{\text{PReLU}} > \text{GC}_{\text{ReLU}} \gg$

$\mathrm{GC}_{\mathrm{TanH}} > \mathrm{GC}_{\mathrm{Sigmoid}}$ reflects the theoretical analysis: non-saturating activations with positive gradients everywhere achieve highest consistency, followed by ReLU which sacrifices consistency in the negative region, and saturating functions which degrade rapidly.

Proposition 6.1 (Gradient decay ordering)**.** *For an L-layer network with weight norms bounded by $w_{\max}$, the gradient magnitude at layer 1 satisfies the ordering:*

$$\|\nabla_{\theta_1}\mathcal{L}\|_{\mathrm{Sig}} \ll \|\nabla_{\theta_1}\mathcal{L}\|_{\mathrm{TanH}} < \|\nabla_{\theta_1}\mathcal{L}\|_{\mathrm{ReLU}} \leq \|\nabla_{\theta_1}\mathcal{L}\|_{\mathrm{PReLU}} \leq \|\nabla_{\theta_1}\mathcal{L}\|_{\mathrm{ELU}}\,, \tag{15}$$

where the first inequality is exponentially strict (ratio scales as $(4/w_{\max})^L$), the second reflects saturation-region losses in TanH, and the final inequalities follow from the gradient lower bounds in Propositions 5.3 and Lemma 5.4(iv).

The Lipschitz properties merit particular attention for robustness analysis. Both ReLU and ELU satisfy $\mathrm{Lip}(\sigma) = 1$, but ELU's C^1 regularity provides stronger stability guarantees. For a network f_θ with Lipschitz-1 activation and bounded weight norms, the end-to-end Lipschitz constant satisfies $\mathrm{Lip}(f_\theta) \leq \prod_{\ell=1}^{L} \|W_\ell\|$. However, the C^1 regularity of ELU additionally ensures that local sensitivity varies smoothly with the input, enabling tighter perturbation analysis in safety-critical settings where worst-case output deviations must be bounded [2].

7. Implications for Architecture Design

The mathematical analysis developed in the preceding sections yields principled criteria for activation function selection in deep architectures. Rather than prescribing a single universal choice, these criteria delineate a trade-off surface whose optimal operating point depends on the dominant design constraint of the application at hand.

The first and arguably most decisive criterion concerns gradient flow. When network depth exceeds approximately 10 layers, or when temporal credit assignment must span long horizons as in reinforcement learning with delayed rewards, non-saturating activations become essential. The exponential gradient decay $O(0.25^L)$ established for sigmoid in Proposition 4.1 renders it fundamentally unsuitable for deep architectures, and while TanH offers improvement through its unit maximum derivative, it remains vulnerable in saturation regions. Among non-saturating alternatives, PReLU and ELU provide the strongest gradient flow guarantees (Proposition 5.3 and Lemma 5.4), making them the natural candidates for architectures where gradient propagation is the binding concern.

A second important dimension is regularity. For tasks requiring smooth output mappings — such as continuous control, where policy smoothness translates directly to physical stability — the C^1 regularity of ELU is mathematically preferred over the C^0 alternatives ReLU and PReLU, whose derivative discontinuity at zero propagates through the network and manifests as non-smooth gradient landscapes and less stable optimization trajectories near decision boundaries. This regularity advantage interacts with convergence efficiency: as established in Theorem 5.5, ELU's zero-mean activations improve the

discount-factor dependence in convergence rates from $(1-\gamma)^{-3/2}$ to $(1-\gamma)^{-2}$, a substantial gain when the discount factor γ is close to 1, which justifies ELU's computational overhead in sample-limited settings.

These theoretical advantages must, however, be weighed against computational constraints. When inference latency is the binding requirement, the analysis reduces to a straightforward calculus: ReLU's comparison operation is approximately $50\times$ faster than ELU's exponential, and PReLU — requiring one additional multiplication per negative activation — offers an intermediate point. For architectures operating under hard real-time constraints at sub-millisecond timescales, this efficiency gap dominates all other considerations regardless of the theoretical merits of smoother alternatives. A promising resolution to this tension lies in hybrid strategies: since the mathematical requirements differ across network components, architectures employing ReLU in early feature-extraction layers and ELU in terminal policy-generation layers can simultaneously satisfy efficiency and smoothness constraints, exploiting the compositional structure of deep networks to achieve near-optimal performance along multiple criteria.

8. Conclusion

We have established a rigorous mathematical framework characterizing six fundamental activation functions along the dimensions of representational capacity, gradient flow, regularity, and convergence. The key results are: (i) Theorem 3.1 demonstrates the representational collapse of linear activations through depth, motivating nonlinear alternatives; (ii) Propositions 4.1–4.2 quantify the exponential gradient decay that renders saturating activations unsuitable for deep architectures; (iii) Theorem 5.1 establishes that ReLU's binary gradient structure eliminates depth-dependent attenuation along active pathways; (iv) Proposition 5.3 shows that PReLU achieves bounded gradient flow in both positive and negative regions; and (v) Theorem 5.5 demonstrates ELU's convergence advantage stemming from its C^1 regularity and zero-mean property.

No single activation function dominates across all mathematical criteria. The analysis reveals a fundamental trade-off surface: computational efficiency (favoring ReLU), gradient completeness (favoring PReLU), and regularity with convergence optimality (favoring ELU). Optimal architecture design requires selecting the appropriate operating point on this surface based on domain-specific constraints. Future research directions include adaptive activation selection via meta-learning, formal sample complexity bounds parameterized by activation properties, and the analysis of emerging activation paradigms including those inspired by quantum computing and neuromorphic hardware.

9. Acknowledgement

The authors thank the Decisions LAB team at University Mediterranea of Reggio Calabria for computational resources and valuable discussions.

REFERENCES

[1] K. Hornik, M. Stinchcombe, H. White, Multilayer feedforward networks are universal approximators, *Neural Networks* **2**(5) (1989) 359–366.

[2] I. Goodfellow, Y. Bengio, A. Courville, *Deep Learning*, MIT Press, 2016.

[3] Y. LeCun, Y. Bengio, G. Hinton, Deep learning, *Nature* **521**(7553) (2015) 436–444.

[4] X. Glorot, A. Bordes, Y. Bengio, Deep sparse rectifier neural networks, in: *Proc. AISTATS*, JMLR W&CP **15**, 2011, pp. 315–323.

[5] K. He, X. Zhang, S. Ren, J. Sun, Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification, in: *Proc. IEEE ICCV*, 2015, pp. 1026–1034.

[6] D.-A. Clevert, T. Unterthiner, S. Hochreiter, Fast and accurate deep network learning by exponential linear units (ELUs), *arXiv preprint* arXiv:1511.07289, 2015.

[7] Y. Bengio, P. Simard, P. Frasconi, Learning long-term dependencies with gradient descent is difficult, *IEEE Trans. Neural Netw.* **5**(2) (1994) 157–166.

[8] R. S. Sutton, A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed., MIT Press, 2018.

[9] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: *Proc. AISTATS*, JMLR W&CP **9**, 2010, pp. 249–256.

[10] Z. Allen-Zhu, Y. Li, Z. Song, A convergence theory for deep learning via over-parameterization, in: *Proc. ICML*, PMLR **97**, 2019, pp. 242–252.

[11] P. Ramachandran, B. Zoph, Q. V. Le, Searching for activation functions, *arXiv preprint* arXiv:1710.05941, 2017.

[12] V. Mnih et al., Human-level control through deep reinforcement learning, *Nature* **518**(7540) (2015) 529–533.

(Received, November 12, 2025)

(Received, February 13, 2025)

[1,2]Decisions LAB,

University Mediterranea of Reggio Calabria, Italy.

Email[1] massimiliano.ferrara@unirc.it

Email[2] celeste.ciccia@unirc.it