French-German-Spanish Conference on Optimization
Gijón, June 18-21, 2024
(pp. 116-123)

Progress and future directions in machine learning through control theory

Enrique Zuazua[1,2,3]

1. enrique.zuazua@fau.de Chair for Dynamics, Control, Machine Learning, and Numerics, Alexander von Humboldt-Professorship, Department of Mathematics, Friedrich-Alexander-Universität Erlangen-Nürnberg, 91058 Erlangen, Germany
2. Departamento de Matemáticas, Universidad Autónoma de Madrid, 28049 Madrid, Spain
3. Chair of Computational Mathematics, Fundación Deusto. Av. de las Universidades, 24, 48007 Bilbao, Basque Country, Spain

Abstract

This paper presents our recent advancements at the intersection of machine learning and control theory. We focus specifically on utilizing control theoretical tools to elucidate the underlying mechanisms driving the success of machine learning algorithms. By enhancing the explainability of these algorithms, we aim to contribute to their ongoing improvement and more effective application. Our research explores several critical areas:

Firstly, we investigate the memorization, representation, classification, and approximation properties of residual neural networks (ResNets). By framing these tasks as simultaneous or ensemble control problems, we have developed nonlinear and constructive algorithms for training. Our work provides insights into the parameter complexity and computational requirements of ResNets.

Similarly, we delve into the properties of neural ODEs (NODEs). We demonstrate that autonomous NODEs of sufficient width can ensure approximate memorization properties. Furthermore, we prove that by allowing biases to be time-dependent, NODEs can track dynamic data. This showcases their potential for synthetic model generation and helps elucidate the success of methodologies such as Reservoir Computing.

Next, we analyze the optimal architectures of multilayer perceptrons (MLPs). Our findings offer guidelines for designing MLPs with minimal complexity, ensuring efficiency and effectiveness for supervised learning tasks.

The generalization and prediction capacity of trained networks plays a crucial role. To address these properties, we present two nonconvex optimization problems related to shallow neural networks, capturing the ”sparsity” of parameters and robustness of representation. We introduce a ”mean-field” model, proving, via representer theorems, the absence of a relaxation gap. This aids in designing an optimal tolerance strategy for robustness and, through convexification, efficient algorithms for training.

In the context of large language models (LLMs), we explore the integration of residual networks with self-attention layers for context capture. We treat ”attention” as a dynamical system acting on a collection of points and characterize their asymptotic dynamics, identifying convergence towards special points called leaders. These theoretical insights have led to the development of an interpretable model for sentiment analysis of movie reviews, among other possible applications.

Lastly, we address federated learning, which enables multiple clients to collaboratively train models without sharing private data, thus addressing data collection and privacy challenges. We examine training efficiency, incentive mechanisms, and privacy concerns within this framework, proposing solutions to enhance the effectiveness and security of federated learning methods.

Our work underscores the potential of applying control theory principles to improve machine learning models, resulting in more interpretable and efficient algorithms. This interdisciplinary approach opens up a fertile ground for future research, raising profound mathematical questions and application-oriented challenges and opportunities.

1. Introduction

The impact of machine learning (ML) and artificial intelligence (AI) in science is leading to rich and innovative lines of research in applied mathematics. There is a significant need for theoretical foundations that ensure the performance, reliability, and interpretability of ML methods. Specifically, mathematical models are required to understand and optimize rapidly emerging computational architectures. This challenge can be addressed through the lens of control theory, a combination that offers great potential.

In this paper, we discuss recent results from our group that explore the application of control tools to some of the main architectures and methods in ML, namely neural networks, self-attention mechanisms, and federated learning.

Control theory lies at the foundation of ML [15]. Aristotle anticipated control theory when he described the need for automated processes to free humans from their heaviest tasks [4]. In the 1940s, Norbert Wiener redefined the term ”cybernetics,” previously coined by André-Marie Ampère, as ”the science of communication and control in animals and machines,” which reflected the discipline’s definitive contribution to the industrial revolution.

Wiener’s definition involves two conceptual binomials. The first is control-communication: the need for quality information about the state of the system to make the right decisions, reach given objectives, and avoid risky regimes. The second binomial is animal-machine: as anticipated by Aristotle, humans aim to build machines to perform routine tasks. These concepts are integral to contemporary ML. The close link between control theory and ML, and more generally AI, is thus inherent in Wiener’s definition. Once more, we stand on the shoulders of giants.

2. Control-based supervised learning via neural networks

Supervised learning is one of the main paradigms of machine learning (ML), aiming to define a map that approximates an unknown function $f : \mathcal{X} \to \mathcal{Y}$ using a training dataset $\{(x_i, y_i)\}_{i=1}^N$. Neural networks form a widely used class of functions to approximate f, and among these, residual networks have proven to be particularly effective. In the continuous-time limit, these discrete systems, like for instance Residual Neural Networks (ResNets),

$$x^{k+1} = x^k + W_k \boldsymbol{\sigma}(A^k x_i^k + b^k), \quad k \in [L], \tag{2.1}$$

transform into the so-called Neural ODE (NODE):

$$\begin{cases} \dot{x}(t) = W(t)\boldsymbol{\sigma}(A(t)x(t) + b(t)), & t \in (0, T), \\ x(0) = x_i, \end{cases} \tag{2.2}$$

for all $i \in [N] := \{1, \dots, N\}$. Here, $x = x(t)$ is the state if the system, representing the data under consideration, evolving continuously on time in the ambient space, $(W(t), A(t), b(t)) \in \Theta_p := L^\infty\big((0,T); \mathbb{R}^{d\times p} \times \mathbb{R}^{p\times d} \times \mathbb{R}^p\big)$ are piecewise constant controls with L discontinuities (which play the role of the NN parameters to be trained), $L, p \geq 1$ represent the depth and the width of the model, respectively, and $\boldsymbol{\sigma} : \mathbb{R}^p \to \mathbb{R}^p$ is a Lipschitz-continuous non-linearity defined component-wise, a common example being the *rectified linear unit* (ReLU): $x \mapsto \max\{x, 0\}$.

One of the main advantages of NODEs is the possibility to reinterpret several machine learning paradigms using tools from differential equations and their control. For example, data classification can be formulated as a simultaneous control problem for (2.2), the goal being to build controls (W, A, b) driving all initial data $\{x_i\}_{i=1}^N$ to their corresponding targets $\{y_i\}_{i=1}^N$ (prescribed according to the labels) through the flow map generated by (2.2).

In [11], we prove the simultaneous controllability of (2.2) for the single-neuron width case ($p = 1$) via an inductive algorithm that constructs explicit, piecewise constant controls (W, A, b) to sequentially guide each point x_i to its target y_i. Moreover, using similar techniques, we obtain a result of universal approximation in $\|\cdot\|_{L^2}$ for NODEs. Below, we state the two main results from [11]:

Theorem 2.1 (Controllability) *Let $N \geq 1$, $d \geq 2$, and $T > 0$. Consider any dataset $\{x_i, y_i\}_{i=1}^N \subset \mathbb{R}^d$ with $x_i \neq x_j$ and $y_i \neq y_j$ for $i \neq j$. Then, there exists a piecewise constant control $(W, A, b) \in \Theta_1$ (with $p = 1$) such that the flow map Φ_T generated by* (2.2) *satisfies*

$$\Phi_T(x_i) = y_i, \qquad \textit{for all } i = 1, \dots, N.$$

Furthermore, the depth of the model is $L = 3N$.

Theorem 2.2 (Approximation) *Let $d \geq 2$, $T > 0$ and a bounded set $\Omega \subset \mathbb{R}^d$. Then, for any $f \in L^2(\Omega; \mathbb{R}^d)$ and $\varepsilon > 0$ there exists a piecewise constant control $(W, A, b) \in \Theta_1$ (with $p = 1$) such that the flow map Φ_T generated by* (2.2) *satisfies*

$$\|\Phi_T - f\|_{L^2(\Omega)} < \varepsilon.$$

The simultaneous control result in theorem 2.1 and its proof opens paths for new methodologies in data classification, albeit requiring very high complexity (it scales with N). In [3], we reduce the complexity of the controls for binary classification by proposing new algorithms based on predetermined point clusterings. Our strategy aims to probabilistically reduce the number of parameters needed by leveraging the spatial structure of the data distribution, assuming that the points are in general position, i.e., no $d + 1$ points can lie on the same hyperplane in $\mathbb{R}^d$, which is generically fulfilled by random datasets.

Theorem 2.3 *Let $d \geq 2$ and $N \geq 1$. For any dataset $\{(x_i, y_i)\}_{i=1}^N \subset \mathbb{R}^d \times \{1, 0\}$ in general position and any $j \in \{1, \ldots, d\}$, there exist $T > 0$ and a piecewise constant control $(W, A, b) \in \Theta_1$ (with $p = 1$) with $L = 4\lceil m/d \rceil - 1$ discontinuities, where $m = \min(\#\{i : y_i = 1\}, \#\{i : y_i = 0\})$, such that the flow map generated by* (2.2) *satisfies*

$$\Phi_T(x_i)^{(j)} < 1 \quad \text{if } y_i = 1 \quad \text{and} \quad \Phi_T(x_i)^{(j)} > 1 \quad \text{if } y_i = 0, \qquad \text{for all } i = 1, \ldots, N.$$

The described results are focused on the simplified version of (2.2) with $p = 1$ neurons per layer. In [2], we focus on the role played by the architecture through the interplay between the depth L and width p. Our findings reveal a balancing trade-off, as shown in the following result:

Theorem 2.4 *Let $N \geq 1$, $d \geq 2$, $T > 0$. Consider any dataset $\{x_i, y_i\}_{i=1}^N \subset \mathbb{R}^d$ with $x_i \neq x_j$ and $y_i \neq y_j$ if $i \neq j$. For any $p \geq 1$, there exists a piecewise constant control $(W, A, b) \in \Theta_p$ such that the flow map Φ_T generated by* (2.2) *satisfies*

$$\Phi_T(x_i) = y_i, \qquad \text{for all } i = 1, \ldots, N.$$

Furthermore, the depth of the model is $L = 2\lceil N/p \rceil$.

In the wide limit, where $L = 0$, the system (2.2) becomes autonomous and a separate study is required. We address the relaxed problem of ε-approximate controllability of N pairs of points and establish an explicit error decay by uniformly approximating a custom-built Lipschitz vector field that effectively interpolates the dataset:

Theorem 2.5 *Let $N \geq 1$, $d \geq 2$ and $T > 0$ be fixed. Consider any dataset $\{x_i, y_i\}_{i=1}^N \subset \mathbb{R}^d$ with $x_i \neq x_j$. For each $p \geq 1$, there exists a control $(W, A, b) \in \Theta_p$ such that the flow map Φ_T generated by* (2.2) *satisfies*

$$\sup_{i=1,\ldots,N} |y_i - \Phi_T(x_i)| \leq C \frac{\log_2(\kappa)}{\kappa^{1/d}},$$

where $\kappa = (d+2)dp$ is the number of neurons in the model, and $C > 0$ is a constant depending on d, T, but independent of κ.

The study of the autonomous system is closely related to the turnpike principle paradigm, coined by John von Neumann, which ensures that optimal control strategies remain almost steady over long time periods. In [5], we have analyzed the implications of this principle for designing simplified and more stable architectures for deep ResNets.

An extension of the developed theory reformulates the continuous model in terms of transport equations, through the classical link between (2.2), seen as the ODE of characteristics, and the hyperbolic transport PDE, leading to the following neural transport model:

$$\partial_t \rho + \operatorname{div}_x(W(t)\boldsymbol{\sigma}(A(t)x + b(t))\rho) = 0. \tag{2.3}$$

Transforming one given probability measure into another, up to an arbitrarily small Wasserstein-1 error [2, 11] or total variation error [12], can be reinterpreted as a control problem for (2.3). The first approach allows us to build a bridge with the theory of optimal transport, whereas the latter, whose theorem statement we formulate below, has applications in generative modeling via the technique known as normalizing flows.

Theorem 2.6 *Given two probability densities $\rho_0, \rho_T \in L^1(\mathbb{R}^d)$, for any $T > 0$ and for all $\varepsilon > 0$, there exist piecewise constant controls $(w, a, b) \in \Theta_1$ such that the solution of* (2.3) *satisfies*

$$\|\rho(T) - \rho_T\|_{L^1(\mathbb{R}^d)} < \varepsilon.$$

In addition to ResNets and NODEs, we have analyzed the so-called multilayer perceptron deep NN:

$$x^{k+1} = \boldsymbol{\sigma}_{k+1}(A^k x^k + b^k), \quad k \in [L], \tag{2.4}$$

where $x^k \in \mathbb{R}^{d_k}$ denotes the state at layer/step $k \geq 1$, $A^k \in \mathbb{R}^{d_{k+1} \times d_k}$, $b^k \in \mathbb{R}^{d_{k+1}}$, and $\{d_k\}_{k=1}^L$ is a sequence of positive integers determining the dimension of the state and the width of (2.4) at the layer k. Here, $\boldsymbol{\sigma}_{k+1} : \mathbb{R}^{d_{k+1}} \to \mathbb{R}^{d_{k+1}}$ denotes the (component-wise) ReLU function, and $\max_k\{d_k\}$ the total width of (2.4). In [6], for a dataset of N elements in $\mathbb{R}^d$, $d \geq 1$, and M classes, we prove that (2.4) is simultaneously controllable with width 2 and at most $2N + 4M - 1$ layers. This is proven using an inductive algorithm that provides explicit values for the parameters. This result is sharp in the sense that (2.4) with width 1 cannot achieve simultaneous controllability. Additionally, in [6], the universal approximation (UA) for $L^p(\Omega; \mathbb{R}_+)$ functions (for $p \in [1, \infty)$ and $\Omega \subset \mathbb{R}^d$ bounded) is proven, using (2.4) with width $d + 1$, together with explicit convergence rates for $W^{1,p}$ functions, which can be extended to changing-sign functions too.

3. Representer theorem for shallow neural networks: sparsity and generalization

Besides NODEs, ResNets and deep NNs, we have also analysed the representational and generalization capacity of shallow NN, as conducted in [9]. The shallow NN is expressed as:

$$f_{\text{shallow}}(x,\Theta) := \sum_{j=1}^{P} \omega_j \sigma(\langle a_j, x\rangle + b_j), \tag{3.1}$$

where $\Theta = \{(\omega_j, a_j, b_j) \in \mathbb{R}\times\Omega\}_{j=1}^{P}$, P denotes its width, and Ω is a compact subset of $\mathbb{R}^d$ containing a neighborhood of 0. We first investigate the representational capacity of (3.1).

Theorem 3.1 *Assume that σ is continuous and $\sigma(x) = 0$ for $x \le 0$ and $\sigma(x) > 0$ for $x > 0$. Fix any consistent dataset $\{(x_i, y_i) \in \mathbb{R}^{d+1}\}_{i=1}^{N}$. If $P \ge N$, then there exists $\Theta \in (\mathbb{R}\times\Omega)^P$ such that*

$$f_{\text{shallow}}(x_i,\Theta) = y_i, \quad \textit{for all } i = 1,\dots,N.$$

For a fixed dataset $\{(x_i, y_i) \in \mathbb{R}^{d+1}\}_{i=1}^{N}$, Theorem 3.1 shows the existence of parameters for its exact representation by (3.1), $P = N$ being sufficient. Next, we consider an optimization problem, where the objective is to minimize the ℓ_1 norm of the neuron weights:

$$\inf_{\{(\omega_j,a_j,b_j)\in\mathbb{R}\times\Omega\}_{j=1}^{N}} \sum_{j=1}^{N} |\omega_j|, \qquad \text{s.t.} \sum_{j=1}^{N} \omega_j\sigma(\langle a_j, x_i\rangle + b_j) = y_i, \quad \text{for all } i = 1,\dots,N. \tag{P$_0$}$$

When $\{y_i\}_{i=1}^{N}$ represent observed labels affected by some level of noise, it is more meaningful to consider the previous optimization problem under certain tolerance on the error of the prediction. This leads to the following optimization problem parameterized by $\epsilon \ge 0$:

$$\inf_{\{(\omega_j,a_j,b_j)\in\mathbb{R}\times\Omega\}_{j=1}^{N}} \sum_{j=1}^{N} |\omega_j|, \qquad \text{s.t.} \left|\sum_{j=1}^{N} \omega_j\sigma(\langle a_j, x_i\rangle + b_j) - y_i\right| \le \epsilon, \quad \text{for all } i = 1,\dots,N. \tag{P$_\epsilon$}$$

Problems (P$_0$) and (P$_\epsilon$) are non-convex due to the non-linearity of σ, which induces the lack of convexity in their feasible sets. To cure this lack of convexity we consider the following convex relaxation problems:

$$\inf_{\mu\in\mathcal{M}(\Omega)} \|\mu\|_{\text{TV}}, \quad \text{s.t.} \int_\Omega \sigma(\langle a, x_i\rangle + b)d\mu(a,b) = y_i, \quad \text{for all } i = 1,\dots,N; \tag{PR$_0$}$$

$$\inf_{\mu\in\mathcal{M}(\Omega)} \|\mu\|_{\text{TV}}, \quad \text{s.t.} \left|\int_\Omega \sigma(\langle a, x_i\rangle + b)d\mu(a,b) - y_i\right| \le \epsilon, \quad \text{for all } i = 1,\dots,N, \tag{PR$_\epsilon$}$$

where $\mathcal{M}(\Omega)$ represents the space of Radon measures on Ω, and $\|\cdot\|_{\text{TV}}$ denotes the total variation norm. We demonstrate that there is no gap between the primal problems and the relaxed ones, and that the extreme points of the relaxed solution sets have an atomic structure.

Theorem 3.2 *Under the setting of Theorem 3.1, the solution sets of* (PR$_0$) *and* (PR$_\epsilon$)*, denoted by S(PR$_0$) and S(PR$_\epsilon$), are non-empty, convex and compact in the weak-$*$ sense. Moreover,*

$$\text{val}(\text{PR}_0) = \text{val}(\text{P}_0), \qquad \text{Ext}(S(\text{PR}_0)) \subseteq \left\{ \sum_{j=1}^{N} \omega_j \delta_{(a_j,b_j)} \,\middle|\, (\omega_j, a_j, b_j)_{j=1}^{N} \in S(\text{P}_0) \right\}, \tag{3.2}$$

$$\text{val}(\text{PR}_\epsilon) = \text{val}(\text{P}_\epsilon), \qquad \text{Ext}(S(\text{PR}_\epsilon)) \subseteq \left\{ \sum_{j=1}^{N} \omega_j \delta_{(a_j,b_j)} \,\middle|\, (\omega_j, a_j, b_j)_{j=1}^{N} \in (\text{P}_\epsilon) \right\}, \tag{3.3}$$

where Ext(S) *represents the set of all extreme points of S.*

To study the generalization capacity of the shallow NN, we consider some testing dataset $\{(X', Y')\} = \{(x_i', y_i') \in \mathbb{R}^{d+1}\}_{i=1}^{N'}$ with $N' \in \mathbb{N}_+$, which differs from the training one. The generalization quality is determined by the performance of this shallow NN on the testing set (X', Y'), which is assessed by comparing the actual values $\{y_i'\}_{i=1}^{N'}$ with the predictions $\{f_{\text{shallow}}(x_i', \Theta)\}_{i=1}^{N'}$. Rather than evaluating differences individually, we analyze the discrepancies in their overall distributions to simplify the analysis. Let us denote by

$$m_x = \frac{1}{N}\sum_{i=1}^{N}\delta_{x_i}, \quad m_y = \frac{1}{N}\sum_{i=1}^{N}\delta_{y_i}, \quad \bar{m}_y = \frac{1}{N}\sum_{i=1}^{N}\delta_{f_{\text{shallow}}(x_i,\Theta)};$$

$$m_x' = \frac{1}{N'}\sum_{i=1}^{N'}\delta_{x_i'}, \quad m_y' = \frac{1}{N'}\sum_{i=1}^{N'}\delta_{y_i'}, \quad \bar{m}_y' = \frac{1}{N'}\sum_{i=1}^{N'}\delta_{f_{\text{shallow}}(x_i',\Theta)}.$$

Theorem 3.3 *Assume that σ is L-Lipschitz. Let Θ be a solution of* (P_ϵ) *for some $\epsilon \geq 0$. Then,*

$$d_{\mathrm{KR}}(m_y', \tilde{m}_y') \leq d_{\mathrm{KR}}(m_y, m_y') + \begin{cases} \epsilon + d_{\mathrm{KR}}(m_x, m_x')LD\,\mathrm{val}(\mathrm{P}_\epsilon), & \text{if } 0 \leq \epsilon \leq \|Y\|_{\ell^\infty}, \\ \|Y\|_{\ell^\infty}, & \text{otherwise,} \end{cases}$$

where $D = \sup_{(a,b)\in\Omega} \|a\|$ and d_{KR} represents the Kantorovich–Rubinstein distance.

In view of Theorem 3.3, the problem of minimizing the right-hand-side upper bound with respect to ϵ arises:

$$\inf_{0\leq\epsilon\leq\|Y\|_{\ell^\infty}} \mathcal{U}(\epsilon) := \epsilon + d_{\mathrm{KR}}(m_x, m_x')LD\,\mathrm{val}(\mathrm{P}_\epsilon). \tag{UB}$$

By employing the dual analysis of problems (P_ϵ) and (P_0), we obtain the first-order optimality condition of (UB) in the following theorem. Let us denote by c_ϵ (resp. C_ϵ) the minimum (resp. maximum) value of the ℓ^1 norm of the dual solutions of (PR_ϵ) for any $\epsilon \geq 0$.

Theorem 3.4 *Under the setting of Theorem 3.1, the solution set of problem* (UB)*, denoted by S*(UB)*, is non-empty. Moreover, the following holds:*

1. *If $d_{\mathrm{KR}}(m_x, m_x') < (LDc_0)^{-1}$, then $S(\mathrm{UB}) = \{0\}$.*
2. *If $d_{\mathrm{KR}}(m_x, m_x') \geq (LDc_0)^{-1}$, then $\epsilon \in S(\mathrm{UB})$ if and only if $1/d_{\mathrm{KR}}(m_x, m_x')LD \in [c_\epsilon, C_\epsilon]$.*

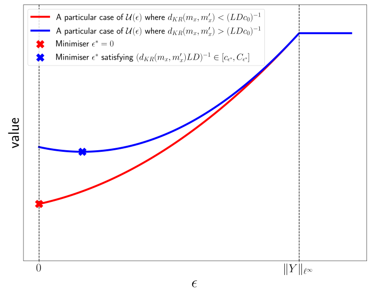


Fig. 3.1 The red and blue curves represent point 1 and 2 of Theorem 3.4, respectively. According to Theorem 3.4, when the distance between the training and testing sets is less than the threshold $(LDc_0)^{-1}$, it suffices to consider the exact representation problem (P_0). If $d_{\mathrm{KR}}(m_x, m_x')$ exceeds this threshold, the optimal ϵ can be determined by solving the dual problem of (PR_ϵ).

4. Dynamical System Approximation via Semi-Autonomous NODEs

Going back to the NODE context, and with the aim of reducing their complexity, measured in terms of the number of switchings of the parameters, while preserving the exact representation capacity, in the upcoming work [7], we study NODEs of the form

$$\begin{cases} \dot{x} = W\boldsymbol{\sigma}(Ax + b(t)), & t \in (0,T), \\ x(0) = x_i, \end{cases} \tag{4.1}$$

where now the only time-dependent parameter is the bias $b = b(t)$. For this reason, we dub the model *Semi-Autonomous* NODE (SA-NODE), which is still non-autonomous, but with a complexity which is greatly reduced, since W and A are now time-independent. Theorem 2.1 continues to hold for (4.1) with no change in the hypotheses. Furthermore, the semi-autonomous structure appears naturally in the proof, as the time-dependency of the biases $b(t)$ is quickly seen to be necessary for tracking dynamic data, as the following result assures, [7].

Theorem 4.1 *Let $K \in \mathbb{R}^d$ be a fixed compact set and consider the non-autonomous ODE*

$$\begin{cases} \dot{\boldsymbol{z}}(t) = f(\boldsymbol{z}, t), \ t \in (0,T), \\ \boldsymbol{z}(0) = \boldsymbol{z}_0 \in K, \end{cases} \tag{4.2}$$

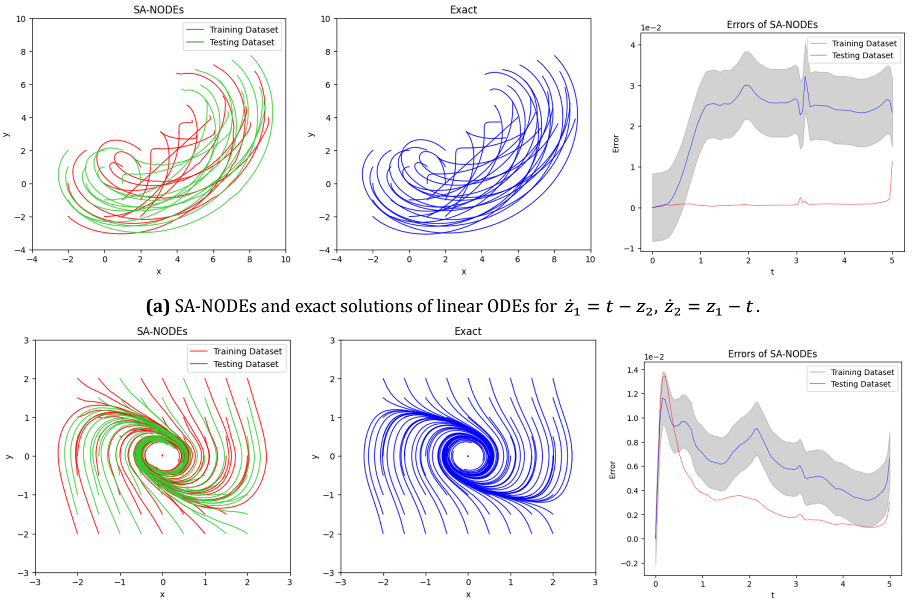
where $f : \mathbb{R}^d \times [0,T] \to \mathbb{R}^d$ is a continuous function and uniformly Lipschitz continuous in $\boldsymbol{z}$. For every $\varepsilon > 0$, there exist $p = p(\varepsilon)$, matrices $W \in \mathbb{R}^{d\times p}, A \in \mathbb{R}^{p\times d}$, and a function $b = b(t) \in L^\infty((0,T);\mathbb{R}^p)$ such that, for every $\boldsymbol{z}_0 \in K$, the solution $\boldsymbol{x} = \boldsymbol{x}(t)$ to the SA-NODE

$$\begin{cases} \dot{\boldsymbol{x}} = \displaystyle\sum_{i=1}^{p} w_i \sigma(a_i \cdot \boldsymbol{x} + b_i(t)), \\ \boldsymbol{x}(0) = \boldsymbol{z}_0, \end{cases} \tag{4.3}$$

satisfies

$$\|\boldsymbol{z}-\boldsymbol{x}\|_{L^\infty([0,T];\mathbb{R}^d)} \leq \varepsilon. \tag{4.4}$$

In other words, SA-NODEs learn the *global* flow of the ODE, and not just the local information around one single trajectory or just the final profiles at $t = T$.



(a) SA-NODEs and exact solutions of linear ODEs for $\dot{z}_1 = t - z_2$, $\dot{z}_2 = z_1 - t$.

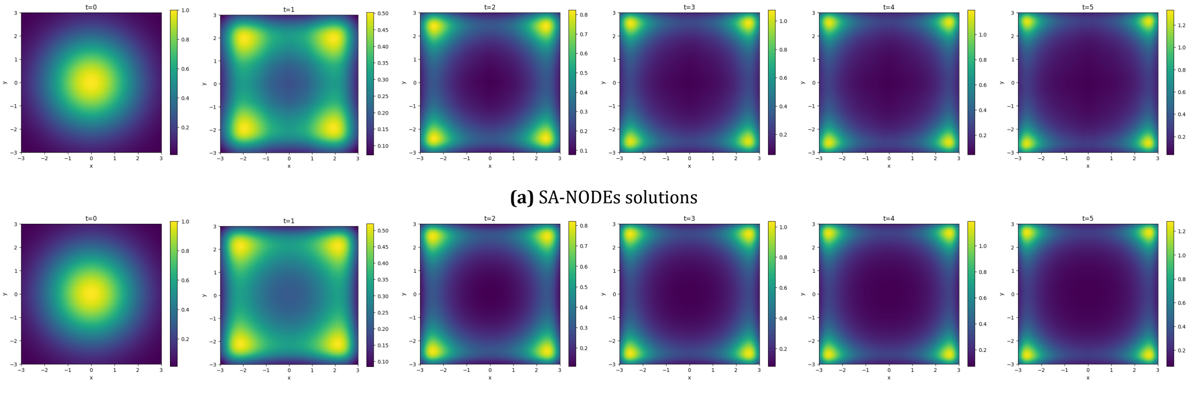
(b) SA-NODEs and exact solutions of nonlinear ODEs for $\dot{z}_1 = z_2$, $\dot{z}_2 = -z_1 - z_2^3$.

Fig. 4.1 SA-NODEs (left) and exact solutions (center) of linear and nonlinear ODEs. On the right, the mean and standard deviation bounds of the error $e(t)$, computed as the euclidean distance between the exact value of the trajectory and the predicted one.

Notably, the semi-autonomous structure emerges spontaneously, roughly because Cybenko's universal approximation theorem yields an approximation of $f(\boldsymbol{z}, t)$ of the form

$$f(\boldsymbol{z}, t) \sim \sum_{i=1}^{p} w_i \sigma(a_i \cdot (\boldsymbol{z}, t)^\top + b_i) = \sum_{i=1}^{p} w_i \sigma(a_i \cdot \boldsymbol{z} + a_i^{d+1} t + b_i).$$

The SA-NODE structure arises naturally when renaming the term $a_i^{d+1} t + b_i$ as $b_i(t)$.



(a) SA-NODEs solutions

(b) Exact solutions

Fig. 4.2 SA-NODEs and exact solution of 2D transport equations $\rho_t + \mathrm{div}_x(f(x, y, t)\rho) = 0$, where the velocity field is $f(x, y, t) = \left[\sin(x)/(1 + t^2), \sin(y)/(1 + t^2)\right]^\top$. The initial datum is the gaussian profile $e^{-x^2-y^2}$.

Numerical results confirm that SA-NODEs are a promising architecture. They not only perform well on benchmark examples, such as linear and nonlinear dynamical systems (see Figures 4.1a-4.1b), but also on transport equations (as shown in Figure 4.2). In Figures 4.1a-4.1b, the simulated trajectories used for training are plotted in red. In contrast, the trajectories predicted from previously unseen initial data are plotted in green, demonstrating the excellent generalization properties of SA-NODEs.

Furthermore, SA-NODEs significantly outperform vanilla NODEs in terms of the number of epochs and neurons required to achieve suitable approximations of dynamical systems. On benchmark examples, and for a fixed number of epochs and neurons, SA-NODEs consistently achieve significantly smaller errors than vanilla NODEs, often by a couple of orders of magnitude. Additionally, even though the network widths are the same, SA-NODEs require less time to train than vanilla NODEs. This is because the number of parameters is reduced, with constant W and A. Consequently, by decreasing the number of parameters, SA-NODEs mitigate the tendency of vanilla NODEs to overfit. This showcases the potential of SA-NODEs for synthetic model generation and helps elucidate the success of methodologies such as Reservoir Computing.

5. Self-attention as a clustering mechanism and its role in LLMs

For supervised learning tasks in large language models (LLMs), capturing ”context” or how words relate to one another in a sentence, is a key feature. The *transformer* is a state-of-the-art neural networks in LLMs, which builds on ResNets by alternating with *self-attention* layers exploiting the data structure. Heuristically, these layers capture the ”context” at the sample level by mixing its rows based on similarity between them.

For this reason, the data samples used to train such models contain collections of words (i.e. sentences or paragraphs). More precisely, the training dataset is of the form $\{(Z_s, y_s)\}_{s=1}^N$, for matrices $Z_s \in \mathbb{R}^{n\times d}$, whose n rows encode words as points in Euclidean space $\mathbb{R}^d$.

For a fixed data sample $Z \in \mathbb{R}^{n\times d}$ with rows $z_1, \dots, z_n \in \mathbb{R}^d$, the (hardmax) self-attention model is given by

$$z_i^{k+1} = z_i^k + \frac{\alpha}{1+\alpha} \frac{1}{|\mathcal{C}_i(Z^k|} \sum_{j\in\mathcal{C}_i(Z^k} \left(z_j^k - z_i^k\right), \quad k \geq 0, \tag{5.1a}$$

where $z_i^0 = z_i$, Z^k contains the rows $z_1^k, \dots, z_n^k$, $A \in \mathbb{R}^{d\times d}$ is a symmetric positive definite matrix, $\alpha > 0$, and

$$\mathcal{C}_i(Z^k = \left\{ j \in [n] \ : \ \left\langle Az_i^k, z_j^k \right\rangle = \max_{\ell\in[n]} \left\langle Az_i^k, z_\ell^k \right\rangle \right\}. \tag{5.1b}$$

In [1], we study the asymptotic behaviour of the self-attention dynamics (5.1) as $k \to \infty$. In particular, we prove that it exhibits clustering behaviour towards special points called *leaders*. As an application, we use our clustering results to design a simple and interpretable transformer-based model to solve the supervised learning task in LLMs of *sentiment analysis*. We use a benchmark dataset with movie reviews, labeled as positive or negative. The proposed model contains only three components with distinct roles: the encoder, mapping words to points in $\mathbb{R}^d$, whose role is to select meaningful words as leaders; our transformer (5.1), whose role is to capture ”context” by clustering the majority of words towards the few most meaningful ones; and the decoder, whose role is to project the final point values to a real prediction by dividing $\mathbb{R}^d$ in two half-spaces and identifying each half-space with each sentiment. After training the model, our interpretation is verified with examples (cf. Figure 5.1).

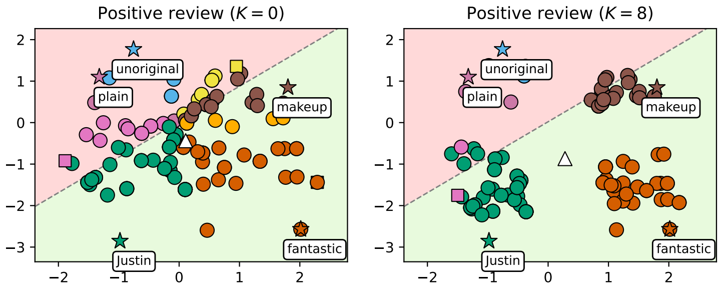


Fig. 5.1 Evolution of the encoded words of a positive review as they are processed by $K = 8$ transformer layers. Points are colored according to the point they follow, leaders are stars and tagged with the word they encode, squares are non-leaders who are followed by other points, circles are the remaining points, and the triangle is the mean word. The dashed line is the hyperplane separating the negative class (red) from the positive class (green).

6. Federated learning: training, incentive, and privacy

With the growing amount of distributed data, *federated learning* (FL) has emerged as a promising paradigm to address challenges like data collection and privacy protection in centralized learning approaches.

As in supervised learning, FL aims to learn a model to approximate $f : \mathcal{X} \to \mathcal{Y}$, but under the constraint that training data and labels are stored across distributed clients. Given m clients, the training of FL can be formulated as

$$\min_{\theta\in W} \sum_{k=1}^{m} p_k \ell_k(\theta), \tag{6.1}$$

where $\theta \in \mathcal{W}$ are trainable parameters, $\ell_k : \mathcal{W} \to \mathbb{R}$ is client k's local loss function, commonly set as the empirical risk over its local dataset, and $p_k \geq 0$ with $\sum_{k=1}^{m} p_k = 1$ specifies the relative impact of client k.

To solve (6.1) efficiently, we propose in [14] an inexact and self-adaptive algorithm termed FedADMM-InSa. We design an inexactness criterion to guide each client to independently adjust its local training accuracy, leading to personalized training and better adaptation to heterogeneous data. Additionally, we present a self-adaptive scheme that dynamically adjusts each client's penalty parameter to enhance the robustness of our algorithm.

As in [14], existing research on FL primarily focuses on designing efficient learning algorithms. Most existing works do not consider that clients may be reluctant to engage without appropriate compensation (rewards from the server) for their training efforts. We address this issue in [8] by formulating incentive mechanisms in FL within a potential game framework. We investigate the uniqueness of the Nash equilibrium in these games and offer the server an easily calculable threshold for the reward, under which it can achieve effective incentives concerning clients' training efforts.

Moreover, the privacy benefits of FL (exchanging model parameters instead of data) can be compromised by data reconstruction attacks. In [13], we propose an approximate and weighted attack method to recover clients' private data under the widely used multiple-step local update scenarios. Experimental results validate the superiority of our attack method, emphasizing the need for effective defense mechanisms in FL to enhance privacy.

Acknowledgements

The author thanks all team members who contributed to develop the research summarised here and, in particular, Albert Alcalde, Antonio Álvarez-López, Martin Hernández, Ziqian Li, Kang Liu, Lorenzo Liverani, Yongcun Song and Ziqi Wang who contributed to draft this synthesis article.

Our research has been funded by the Alexander von Humboldt-Professorship program, ModConFlex Marie Curie Action, HORIZON-MSCA-2021-DN-01, COST Action MAT-DYN-NET, Transregio 154 Project "Mathematical Modelling, Simulation and Optimization Using the Example of Gas Networks" of the DFG, AFOSR-24IOE027, grants PID2020-112617GB-C22 and TED2021-131390B-I00 of MICINN (Spain).

For brevity we have only included references from our team. The articles below contain a much richer bibliography on each of the topics discussed.

References

[1] Albert Alcalde, Giovanni Fantuzzi, and Enrique Zuazua. Clustering in pure-attention hardmax transformers and its role in sentiment analysis. In preparation (2024).

[2] Antonio Álvarez-López, Arselane Hadj Slimane, and Enrique Zuazua. Interplay between depth and width for interpolation in neural ODEs. *arXiv preprint arXiv:2401.09902*, 2024.

[3] Antonio Álvarez-López, Rafael Orive-Illera, and Enrique Zuazua. Optimized classification with neural ODEs via separability. *arXiv preprint arXiv:2312.13807*, 2023.

[4] Enrique Fernández-Cara, and Enrique Zuazua. Control theory: History, mathematical achievements and perspectives. *Bol. Soc. Esp. Mat. Apl.*, 26, 79-140, 2003.

[5] Borjan Geshkovski and Enrique Zuazua. Turnpike in optimal control of PDEs, ResNets, and beyond. *Acta Numerica*, 31:135–263, 2022. Cambridge University Press.

[6] Martin Hernández and Enrique Zuazua. Deep neural networks: Multi-classification and universal approximation. In preparation (2024).

[7] Ziqian Li, Kang Liu, Lorenzo Liverani and Enrique Zuazua. Universal Approximation of Dynamical Systems by Semi-Autonomous Neural ODEs and Applications In preparation (2024).

[8] Kang Liu, Ziqi Wang, and E. Zuazua. Game theory in federated learning: A potential game perspective. In preparation (2024).

[9] Kang Liu and Enrique Zuazua. On the sparse representation of Neural Networks In preparation (2024).

[10] Domènec Ruiz-Balet, Elisa Affili, and Enrique Zuazua. Interpolation and approximation via Momentum ResNets and Neural ODEs. *Systems & Control Letters*, 162:105182, 2022.

[11] Domènec Ruiz-Balet and Enrique Zuazua. Neural ODE Control for Classification, Approximation, and Transport. *SIAM Review*, 65(3):735–773, 2023. doi: 10.1137/21M1411433.

[12] Domènec Ruiz-Balet and Enrique Zuazua. Control of neural transport for normalising flows. *Journal de Mathématiques Pures et Appliquées*, 181:58-90, 2024.

[13] Yongcun Song, Ziqi Wang, and Enrique Zuazua. Approximate and weighted data reconstruction attack in federated learning. *arXiv preprint arXiv:2308.06822*, 2023.

[14] Yongcun Song, Ziqi Wang, and Enrique Zuazua. Fedadmm-insa: An inexact and self-adaptive admm for federated learning. *arXiv preprint arXiv:2402.13989*, 2024.

[15] Enrique Zuazua. Control and Machine Learning. *SIAM News*, 55(8), October 2022.