SA-NODEs and Universal Approximation of Dynamical Systems – Numeric Aspect

Ziqian Li

Department of Mathematics, Jilin University

Benasque, August 22, 2024

joint work with Kang Liu, Lorenzo Liverani and Enrique Zuazua

Contents

Contents

Neural ODEs

The neural ODEs (NODEs) (Chen et al. 2018) read

$$\begin{cases} \dot{\boldsymbol{x}} = \sum_{i=1}^{P} W_i(t) \circ \boldsymbol{\sigma}(A_i(t)\boldsymbol{x} + B_i(t)), \quad t \in [0, T], \\ \boldsymbol{x}(0) = x_0 \in \mathbb{R}^d. \end{cases}$$

Here $\circ$ is the Hadamard product $(a, b) \circ (c, d) = (ac, bd)$, and P is the width of the neural network. The parameters $A_i(t)$, $W_i(t)$ and $B_i(t)$ depend on time.

- The number of parameters $((2d + 1)MP)$ scales as the number of time steps $M \Rightarrow$ High complexity
- Impossible to calculate the solution after T

SA-NODEs

The semi-autonomous neural ODEs (SA-NODEs) read

$$\begin{cases} \dot{\boldsymbol{x}} = \sum_{i=1}^{P} W_i \circ \boldsymbol{\sigma}(A_i^1 \boldsymbol{x} + A_i^2 t + B_i), \quad \boldsymbol{x} \in \mathbb{R}^d, t \in [0, T], \\ \boldsymbol{x}(0) = x_0. \end{cases}$$

Here the parameters W_i, A_i^1, A_i^2 and B_i are constant matrixes.

- The number of parameters $(Pd(d+3))$ is independent of time steps. $\Rightarrow$ Low complexity
- Able to calculate the solution after T

Universal Approximation Theory

To approximate an ODE system

$$\begin{cases} \dot{\boldsymbol{z}}(t) = f(\boldsymbol{z}(t), t), \quad t \in [0, T], \\ \boldsymbol{z}(0) = \boldsymbol{z}_0, \end{cases}$$

by SA-NODEs, we obtain the universal approximation theory

Theorem (Li, Liu, L., Zuazua, 2024)

Let f be uniformly Lipschitz in $\boldsymbol{z}$ with respect to t. For any compact set $K \subseteq \mathbb{R}^d$ and any $\varepsilon > 0$, there exists a constant $P_{\varepsilon,T,K,f}$ such that for any $P \geq P_{\varepsilon,T,K,f}$, there exist parameters $(W_i, A_i^1, A_i^2, B_i) \in \mathbb{R}^d \times \mathbb{R}^{d\times d} \times \mathbb{R}^d \times \mathbb{R}^d$, for $i = 1, \ldots, P$, such that

$$\|\boldsymbol{z}_{z_0}(\cdot) - \boldsymbol{x}_{z_0}(\cdot)\|_{\mathbb{L}^\infty([0,T];\mathbb{R}^d)} \leq \varepsilon, \quad \forall z_0 \in K.$$

Approximation Rate

Further, we obtain the approximation rate with respect to the number of neurons P

Theorem (Li, Liu, L., Zuazua, 2024)

Let $\boldsymbol{f} \in \mathcal{H}^k_{\text{loc}}(\mathbb{R}^d \times [0,T]; \mathbb{R}^d)$, for $k > (d+1)/2 + 2$. Fix any compact set $K \subseteq \mathbb{R}^d$. Then, for any $P \in \mathbb{N}_+$, there exist parameters $(W_i, A_i^1, A_i^2, B_i) \in \mathbb{R}^d \times \mathbb{R}^{d\times d} \times \mathbb{R}^d \times \mathbb{R}^d$, for $i = 1, \dots, P$, such that

$$\sup_{t\in[0,T]} \int_K \|\boldsymbol{z}_{z_0}(t) - \boldsymbol{x}_{z_0}(t)\|^2 dz_0 \leq \frac{C_{T,K,f}}{P},$$

where $C_{T,K,f}$ is a constant independent of P.

Transport Equations

The transport equation of divergence form:

$$\begin{cases} \partial_t \rho(x,t) + \mathrm{div}_x(f(x,t)\rho(x,t)) = 0, & (x,t) \in \mathbb{R}^d \times [0,T], \\ \rho(x,0) = \rho_0(x), & x \in \mathbb{R}^d. \end{cases}$$

The characteristic system of the transport equation is

$$\begin{cases} \dfrac{d}{dt}\begin{pmatrix} X \\ \rho \end{pmatrix} = \begin{pmatrix} f(X,t) \\ -\mathrm{div}_x(f(X,t))\rho \end{pmatrix}, & t \in [0,T], \\ \begin{pmatrix} X(0) \\ \rho(0) \end{pmatrix} = \begin{pmatrix} x_0 \\ \rho_0(x_0) \end{pmatrix}. \end{cases}$$

Transport Equations

The approximated characteristic system:

$$\begin{cases} \dfrac{d}{dt}\begin{pmatrix} X_\Theta \\ \rho_\Theta \end{pmatrix} = \begin{pmatrix} f_\Theta(X_\Theta, t) \\ -\mathrm{div}_x(f_\Theta(X_\Theta, t))\rho_\Theta \end{pmatrix}, \quad t \in [0, T], \\ \begin{pmatrix} X_\Theta(0) \\ \rho_\Theta(0) \end{pmatrix} = \begin{pmatrix} x_0 \\ \rho_0(x_0) \end{pmatrix}. \end{cases}$$

The corresponding neural transport equation:

$$\begin{cases} \partial_t \rho_\Theta(x,t) + \mathrm{div}_x\left(f_\Theta(x,t)\rho_\Theta(x,t)\right) = 0, \quad (x,t) \in \mathbb{R}^d \times [0,T], \\ \rho_\Theta(x,0) = \rho_0(x), \quad x \in \mathbb{R}^d. \end{cases}$$

Transport Equations

We obtain the approximation theory of transport equations

Theorem (Li, Liu, L., Zuazua, 2024)

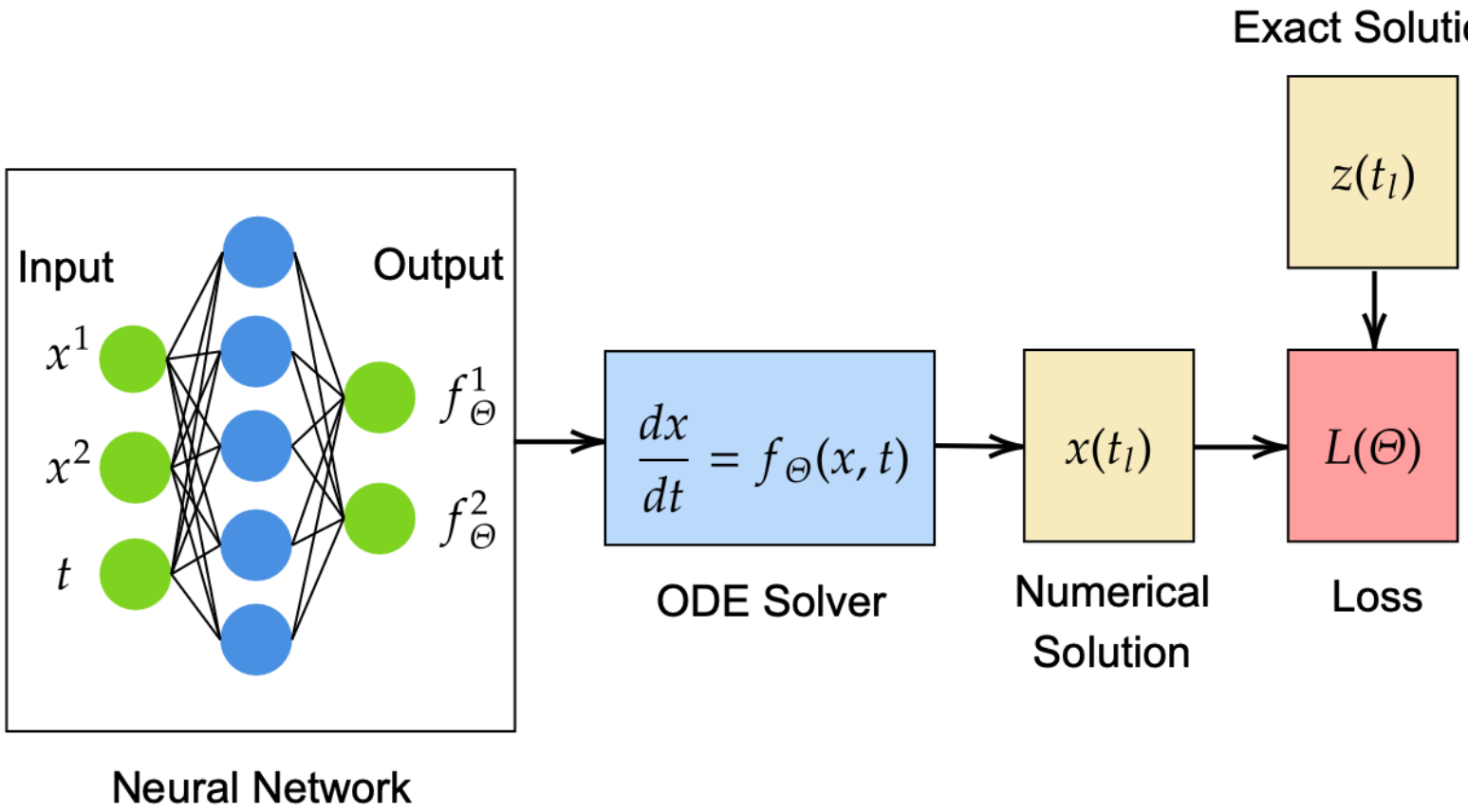
Let ρ_0 be a probability measure supported in a compact set K such that $\rho_0 \in \mathbb{L}^2(K)$. Then, for any $P \in \mathbb{N}_+$, there exist parameters $\Theta = \{(W_i, A_i^1, A_i^2, B_i)\}_{i=1}^P$ such that

$$\sup_{t\in[0,T]} \mathbb{W}_1(\rho(\cdot,t), \rho_\Theta(\cdot,t)) \leq \frac{C_{T,f,\rho_0}}{\sqrt{P}},$$

where C_{T,f,ρ_0} is a constant independent of P, $\mathbb{W}_1(\cdot,\cdot)$ is the Wasserstein-1 distance, and $\rho(\cdot,t)$ (resp. $\rho_\Theta(\cdot,t)$) is the solution of the transport equation (resp. the Neural transport equation) at the time $t \in [0,T]$.

Contents

Approximating ODEs: Workflow



Approximating ODEs: Training Dataset

For an ODE system

$$\begin{cases} \dot{\boldsymbol{z}}(t) = f(\boldsymbol{z}(t), t), \quad t \in [0, T], \\ \boldsymbol{z}(0) = \boldsymbol{z}_0. \end{cases}$$

Data: N trajectories $\mathcal{D} = \{\boldsymbol{z}_k(\cdot)\}_{k=1}^N \subset C([0, T]; \mathbb{R}^d)$.

In practice: $\mathcal{D} = \{\boldsymbol{z}_k(t_l)\}_{k,l} \subset \mathbb{R}^d$, for $k = 1, \ldots, N$, $l = 1, \ldots, M$.

Approximating ODEs: Lipschitz Constant

For an SA-NODE system

$$\begin{cases} \dot{\boldsymbol{x}} = f_\Theta(\boldsymbol{x}(t), t), \quad t \in [0, T] \\ \boldsymbol{x}(0) = x_0, \end{cases}$$

where $\Theta = (W_i, A_i^1, A_i^2, B_i)_{i=1}^P$ and the approximated vector field

$$f_\Theta = \sum_{i=1}^{P} W_i \circ \boldsymbol{\sigma}(A_i^1 \boldsymbol{x} + A_i^2 t + B_i),$$

f_Θ is uniformly Lipschitz continuous in x with the estimate:

$$\|f_\Theta(x,t) - f_\Theta(y,t)\| \leq \left\| \sum_{i=1}^{P} |W_i| \circ \|A_i^1\|_{\ell^2} \right\| \|x - y\|.$$

Approximating ODEs: Loss Function

Data: $\mathcal{D} = \{\boldsymbol{z}_k(t_l)\}_{k,l} \subset \mathbb{R}^d$, for $k = 1, \ldots, N$, $l = 1, \ldots, M$.

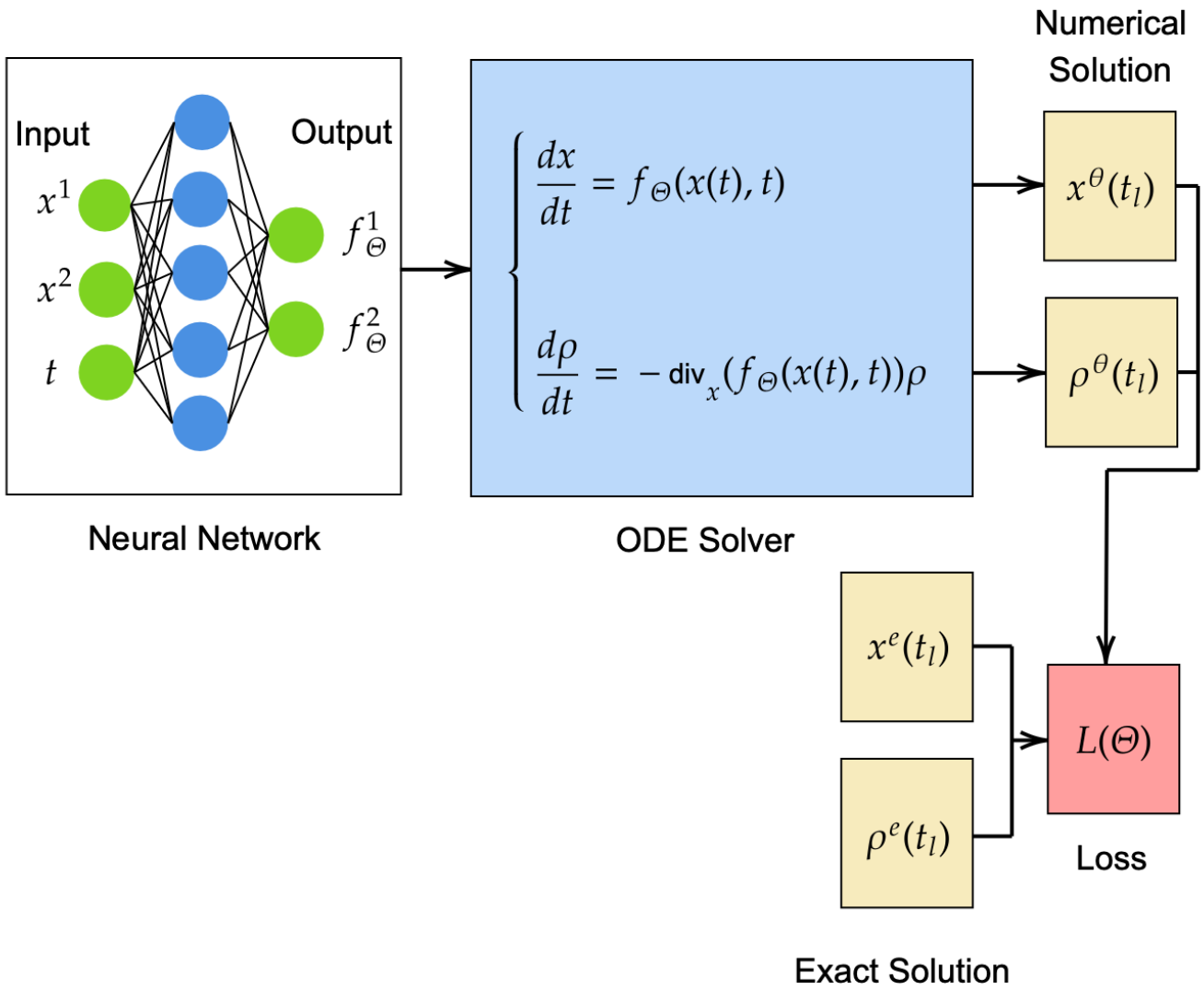
Lipschitz Constant: $\left\| \sum_{i=1}^{P} |W_i| \circ \|A_i^1\|_{\ell^2} \right\|$

Loss Function:

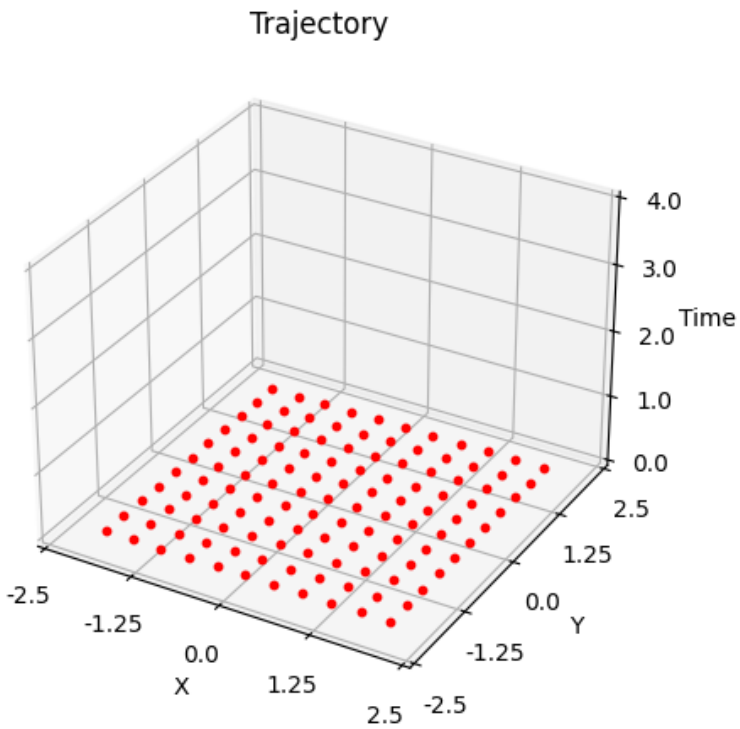
$$L(\Theta) = \frac{1}{NM} \sum_{k=1}^{N} \sum_{l=1}^{M} \left(\boldsymbol{z}_k(t_l) - \boldsymbol{x}_k(t_l, \Theta)\right)^2 + \lambda \left\| \sum_{i=1}^{P} |W_i| \circ \|A_i^1\|_{\ell^2} \right\|$$

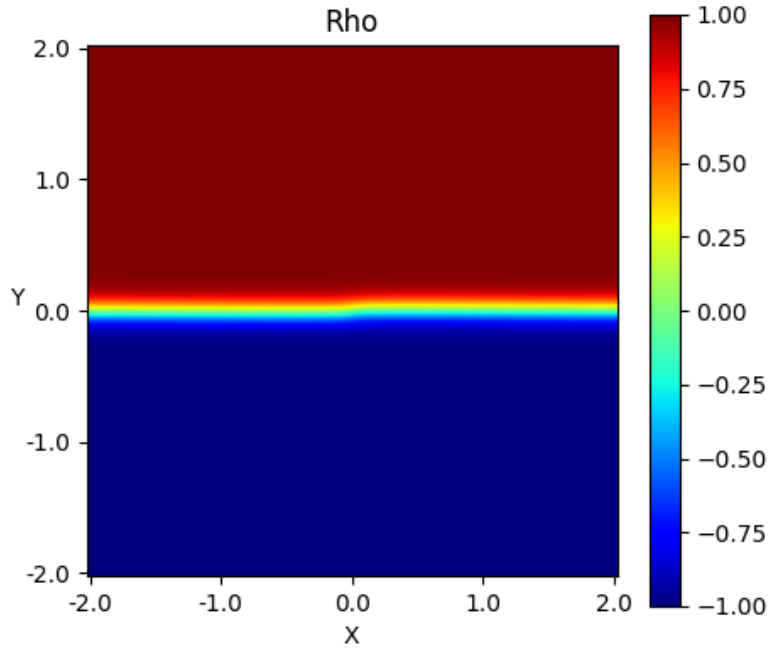
$\rightsquigarrow$ Stochastic gradient descent

Approximating Transport: Workflow



Approximating Transport: Workflow





t = 0.01

Approximating Transport: Data & Loss Function

Data: $\mathcal{D} = \{x_k(t_l), \rho_k(t_l)\}, k = 1, 2, \cdots, N,\ l = 1, 2, \cdots, M.$

Loss Function:

$$L(\Theta) = \frac{1}{NM} \sum_{k=1}^{N} \sum_{l=1}^{M} \Big((x_k(t_l) - x_k(t_l, \Theta))^2 + (\rho_k(t_l) - \rho_k(t_l, \Theta))^2 \Big) + \lambda \left\| \sum_{i=1}^{P} |W_i| \circ \|A_i^1\|_{\ell^2} \right\|,$$

Contents

ODEs: Autunomous & Nonlinear Case

Approximate autonomous and nonlinear ODE system:

$$\begin{cases} \dot{z}_1 = z_2, \\ \dot{z}_2 = -\sin(z_1). \end{cases}$$

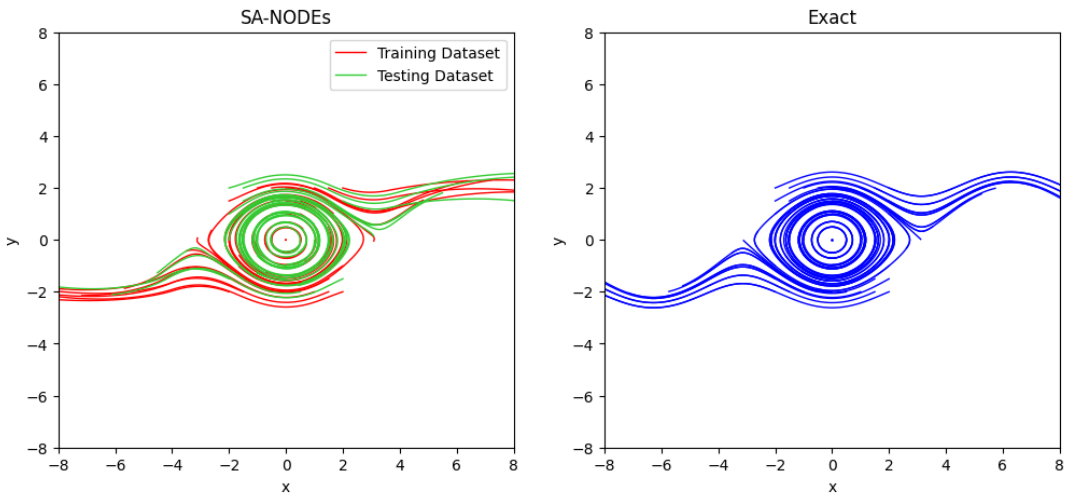


Figure: SA-NODEs and exact solution

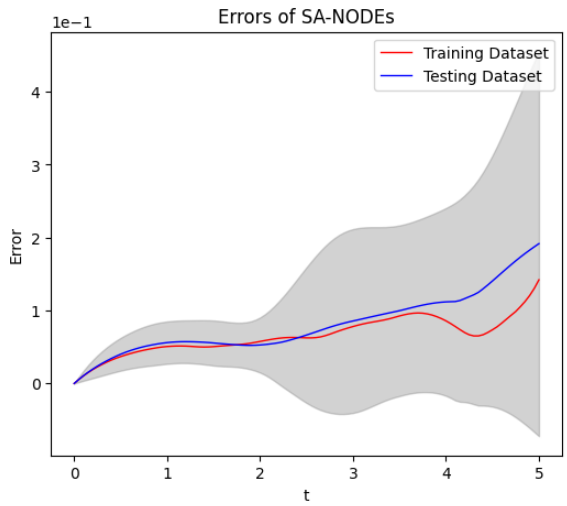
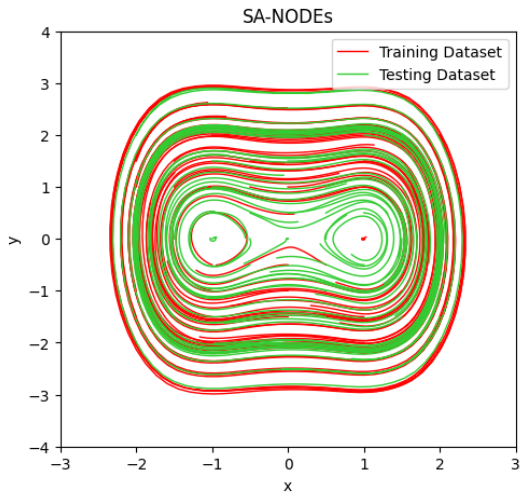


Figure: Errors

ODEs: Non-Autunomous & Nonlinear Case

Approximate non-autonomous and nonlinear ODE system:

$$\begin{cases} \dot{z}_1 = z_2, \\ \dot{z}_2 = z_1 - z_1^3 + \delta\cos(\omega t). \end{cases}$$



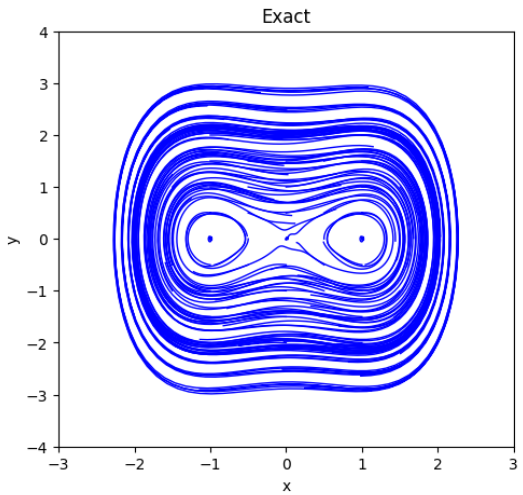


Figure: SA-NODEs and exact solution

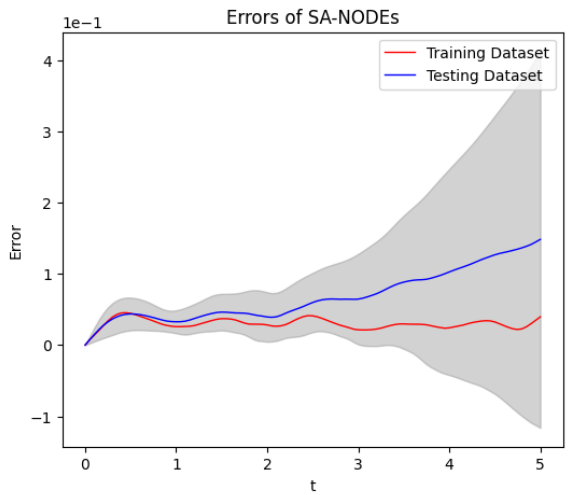


Figure: Errors

Comparison with Vanilla NODEs

Approximate autonomous and linear ODE system:

$$\begin{cases} \dot{z}_1 = z_2, \\ \dot{z}_2 = -2z_1 - 3z_2. \end{cases}$$

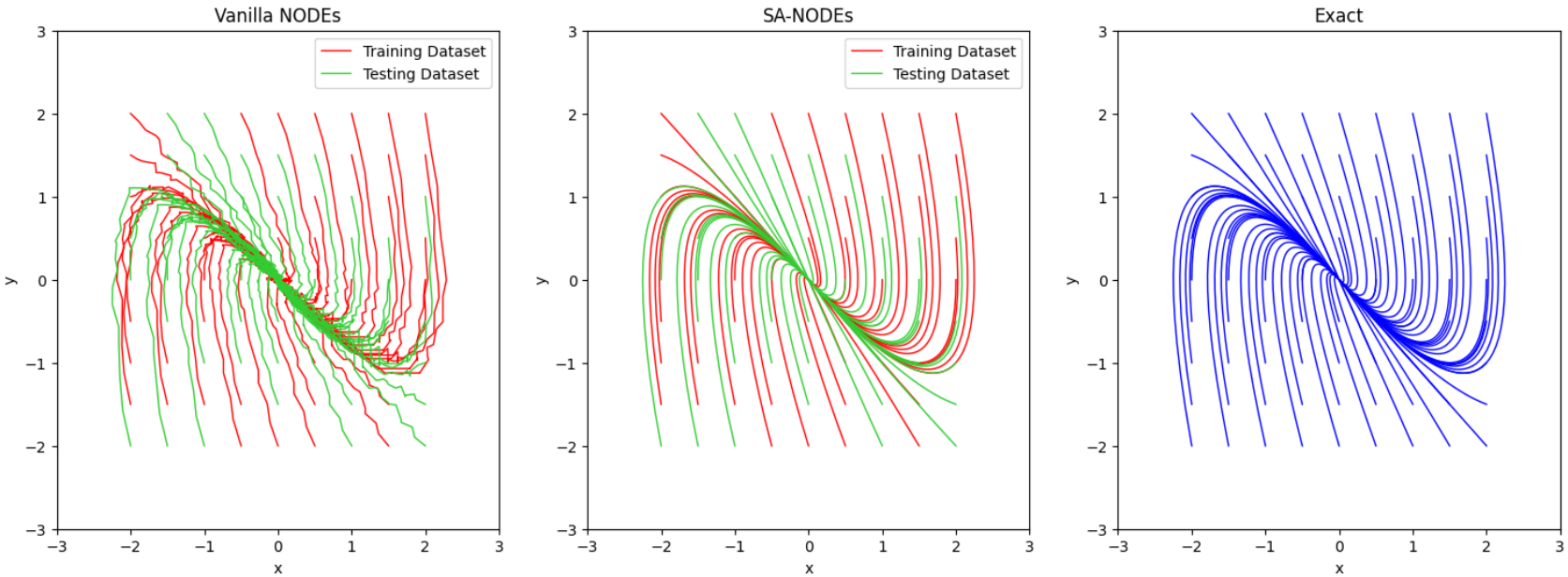


Figure: Vanilla NODEs, SA-NODEs and exact solution

Comparison with Vanilla NODEs

Approximate autonomous and linear ODE system:

$$\begin{cases} \dot{z}_1 = z_2, \\ \dot{z}_2 = -2z_1 - 3z_2. \end{cases}$$

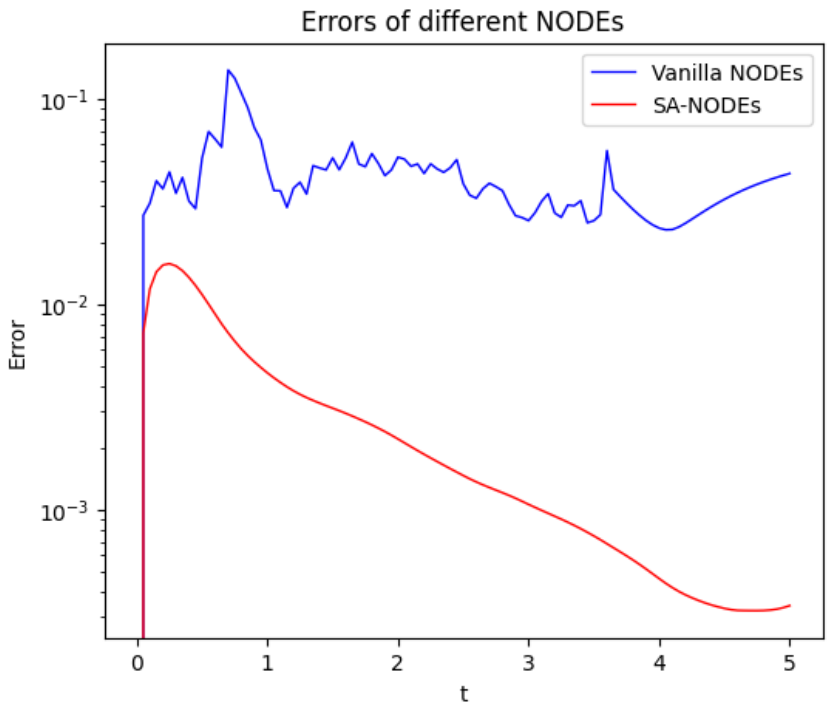


Figure: Testing Errors

Comparison with Vanilla NODEs

Approximate autonomous and linear ODE system:

$$\begin{cases} \dot{z}_1 = z_2, \\ \dot{z}_2 = -2z_1 - 3z_2. \end{cases}$$

P	Neural ODEs	$e_{\max}$	e_T	DoF
100	Vanilla NODEs	2.60e-01	1.79e-01	50000
	SA-NODEs	4.65e-02	3.29e-03	1200
500	Vanilla NODEs	1.91e-01	9.21e-02	250000
	SA-NODEs	2.16e-02	3.83e-04	6000
1000	Vanilla NODEs	1.38e-01	4.34e-02	500000
	SA-NODEs	1.58e-02	3.42e-04	12000

Table: Comparison of errors and degrees of freedom (DoF) between vanilla NODEs and SA-NODEs on autonomous ODEs.

Comparison with Vanilla NODEs

Approximate non-autonomous and linear ODE system:

$$\begin{cases} \dot{z}_1 = t - z_2, \\ \dot{z}_2 = z_1 - t. \end{cases}$$

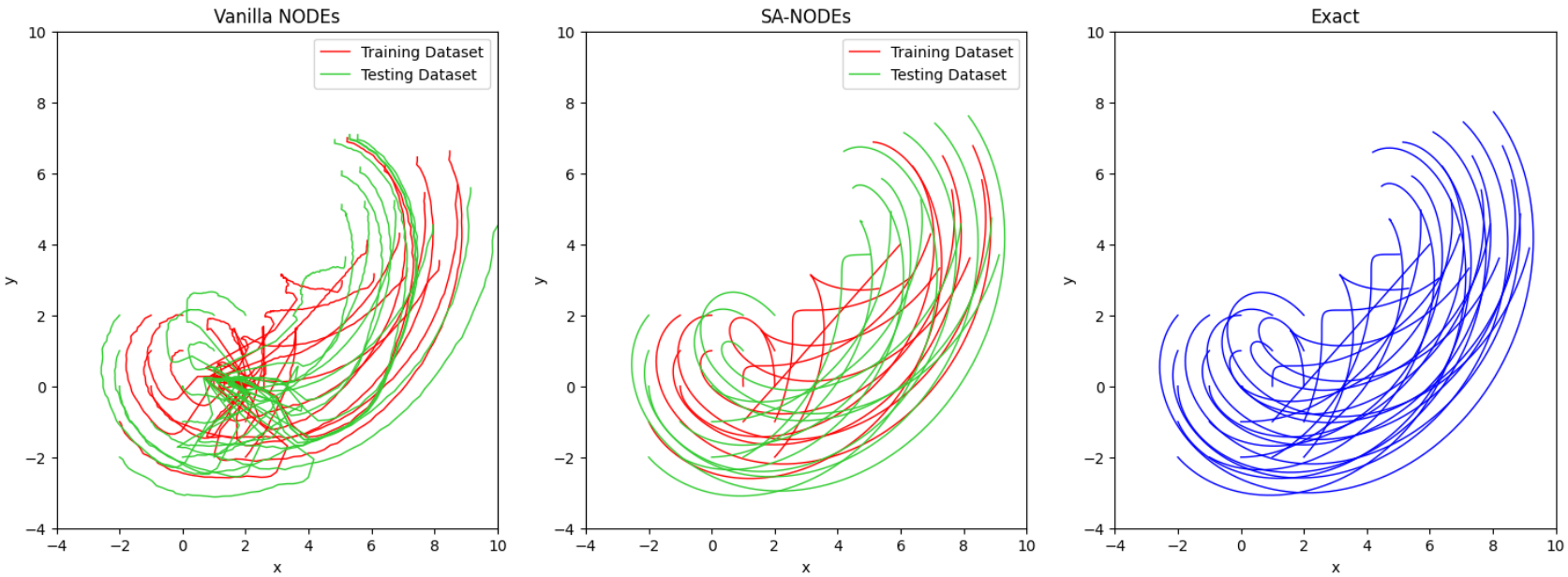


Figure: Vanilla NODEs, SA-NODEs and exact solution

Comparison with Vanilla NODEs

Approximate non-autonomous and linear ODE system:

$$\begin{cases} \dot{z}_1 = t - z_2, \\ \dot{z}_2 = z_1 - t. \end{cases}$$

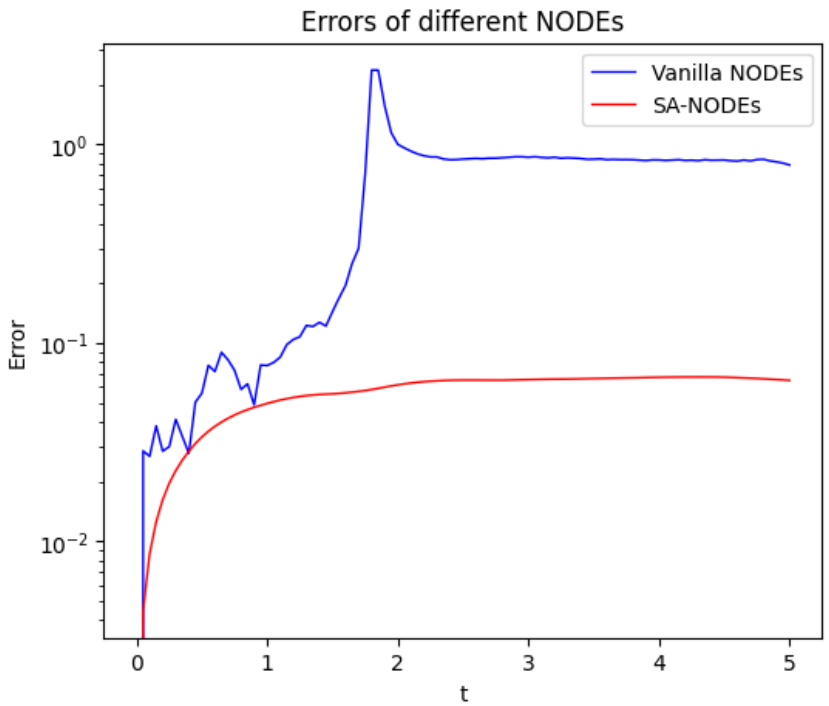


Figure: Testing Errors

Comparison with Vanilla NODEs

Approximate non-autonomous and linear ODE system:

$$\begin{cases} \dot{z}_1 = t - z_2, \\ \dot{z}_2 = z_1 - t. \end{cases}$$

P	Neural ODEs	$e_{\max}$	e_T	DoF
100	Vanilla NODEs	3.66e+00	3.16e+00	50000
	SA-NODEs	7.78e-02	7.13e-02	1200
500	Vanilla NODEs	2.54e+00	2.08e+00	250000
	SA-NODEs	7.35e-02	6.94e-02	6000
1000	Vanilla NODEs	2.37e+00	7.87e-01	500000
	SA-NODEs	6.73e-02	6.47e-02	12000

Table: Comparison of errors and degrees of freedom (DoF) between vanilla NODEs and SA-NODEs on non-autonomous ODEs.

Transport: Non-Autonomous Case

For the non-autonomous transport equation:

$$\begin{cases} \partial_t \rho(x,y,t) + \operatorname{div}\left(\left(\frac{\sin(x)}{1+t^2}, \frac{\sin(y)}{1+t^2}\right)\rho(x,y,t)\right) = 0, \\ \rho(\cdot,0) = \rho_0. \end{cases}$$

Initial measure for training:

$$\rho_0^{\text{train}}(x,y) = e^{-(x^2+y^2)}.$$

Initial measure for testing:

$$\rho_0^{\text{test}}(x,y) = e^{-\frac{x^2+y^2}{4}}.$$

Error for testing:

$$e_{\text{test}}(t) = \frac{\|\rho_\Theta(\cdot,t) - \rho(\cdot,t)\|_{\mathbb{L}^1(\mathbb{R}^2)}}{\|\rho(\cdot,0)\|_{\mathbb{L}^1(\mathbb{R}^2)}}.$$

Transport: Non-Autonomous Case

For the non-autonomous transport equation:

$$\begin{cases} \partial_t \rho(x,y,t) + \operatorname{div}\left(\left(\dfrac{\sin(x)}{1+t^2}, \dfrac{\sin(y)}{1+t^2}\right)\rho(x,y,t)\right) = 0, \\ \rho(\cdot,0) = \rho_0. \end{cases}$$

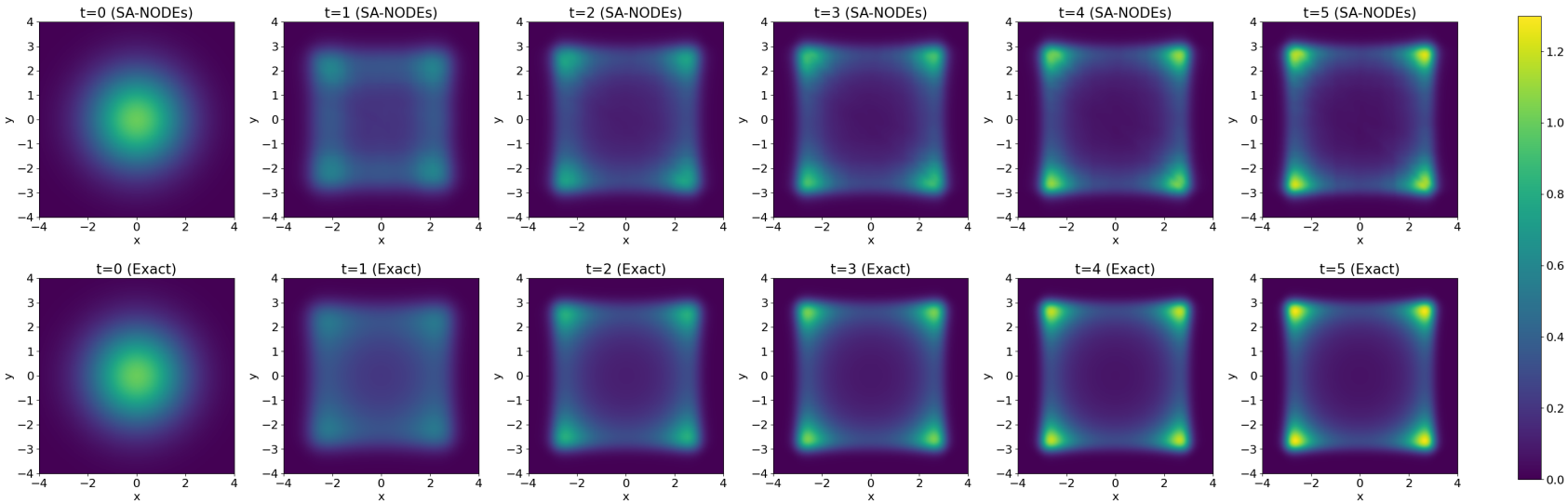


Figure: SA-NODEs and exact solutions

Transport: Non-Autonomous Case

For the non-autonomous transport equation:

$$\begin{cases} \partial_t \rho(x,y,t) + \operatorname{div}\left(\left(\dfrac{\sin(x)}{1+t^2}, \dfrac{\sin(y)}{1+t^2}\right)\rho(x,y,t)\right) = 0, \\ \rho(\cdot,0) = \rho_0. \end{cases}$$

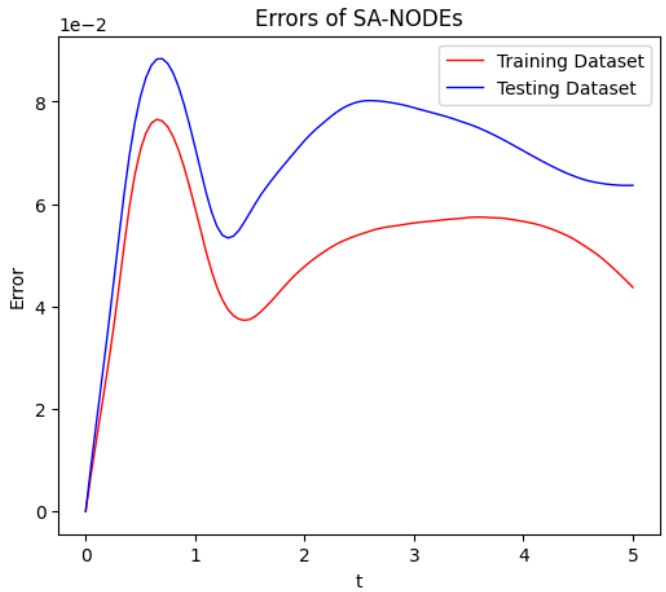


Figure: Training and testing errors

Transport: Doswell Frontogenesis

For the Doswell frontogenesis:

$$\begin{cases} \partial_t \rho(x,y,t) + \operatorname{div}\left((-yg(r(x,y)), xg(r(x,y)))\,\rho(x,y,t)\right) = 0, \\ \rho(\cdot,0) = \rho_0, \end{cases}$$

where

$$g(r(x,y)) = \frac{1}{r(x,y)}\ \overline{v}\ \operatorname{sech}^2(r(x,y))\tanh\left(r(x,y)\right),$$

with $r(x,y) = \sqrt{x^2+y^2}$ and $\overline{v} = 2.59807$. The initial measures for the training and testing are set as:

$$\rho_0^{\text{train}}(x,y) = \tanh\left(y\right), \quad \rho_0^{\text{test}}(x,y) = \tanh\left(10\,y\right).$$

Transport: Doswell Frontogenesis

For the Doswell frontogenesis:

$$\begin{cases} \partial_t \rho(x,y,t) + \operatorname{div}\left((-yg(r(x,y)), xg(r(x,y)))\,\rho(x,y,t)\right) = 0, \\ \rho(\cdot,0) = \rho_0, \end{cases}$$

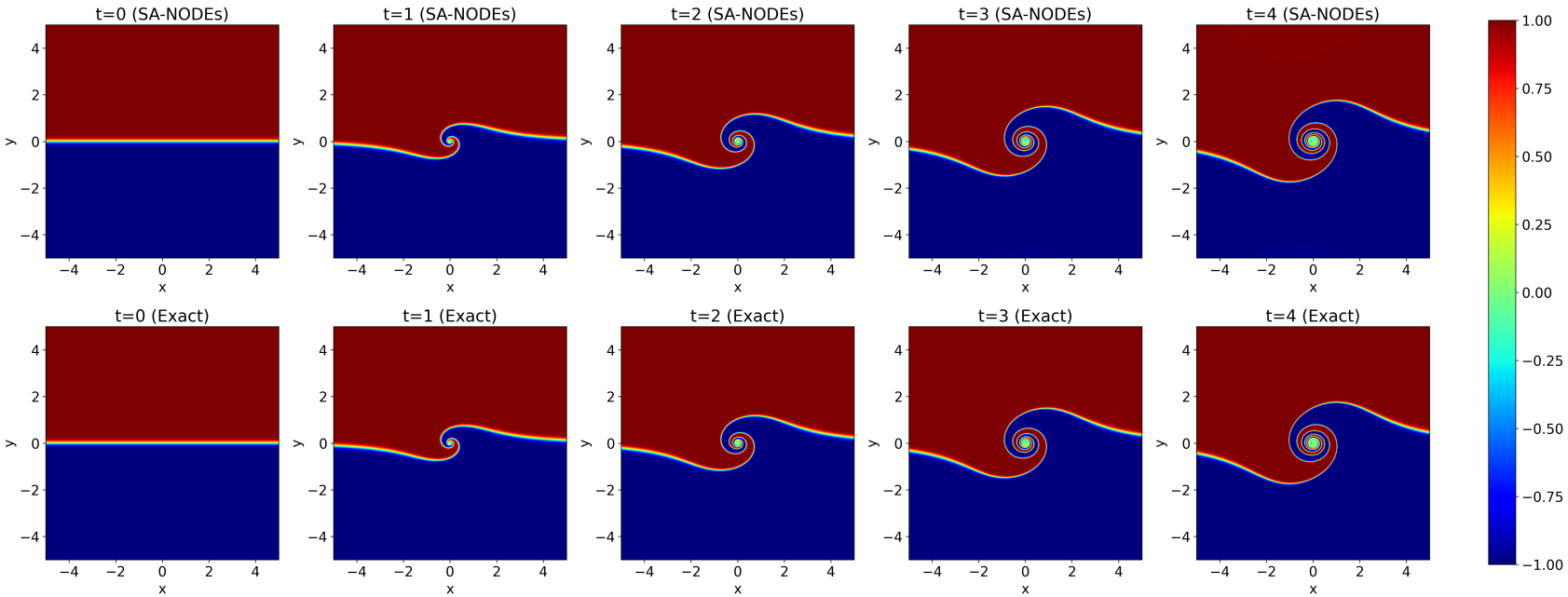


Figure: SA-NODEs and exact solutions

Transport: Doswell Frontogenesis

For the Doswell frontogenesis:

$$\begin{cases} \partial_t \rho(x,y,t) + \operatorname{div}\left((-yg(r(x,y)), xg(r(x,y)))\,\rho(x,y,t)\right) = 0, \\ \rho(\cdot,0) = \rho_0, \end{cases}$$

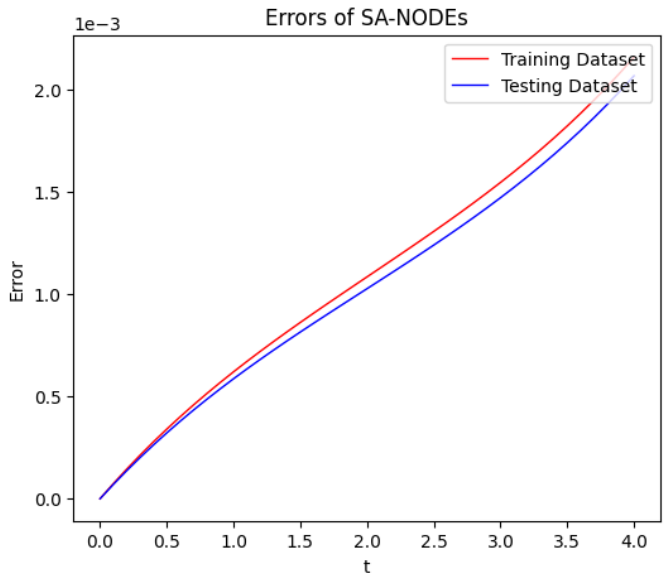


Figure: Training and testing errors

Thank you!