Random Neural Network Expressivity for Non-Linear Partial Differential Equations

Muhammed Ali Mehmood
Department of Mathematics
Imperial College London
UK
muhammed.mehmood21@imperial.ac.uk

Lukas Gonon
School of Computer Science
University of St. Gallen
Switzerland
lukas.gonon@unisg.ch

Abstract

Neural networks with randomly generated hidden weights (RaNNs) have been extensively studied, both as a standalone learning method and as an initialization for fully trainable deep learning methods. In this work, we study RaNN expressivity for learning solutions to non-linear partial differential equations (PDEs). Despite their widespread use in practical applications, a rigorous theoretical understanding of the approximation properties of RaNNs in this context remains limited. Here, we derive error bounds for RaNN approximations to time-dependent Sobolev functions and obtain a dimension-free approximation rate $\frac{1}{2}$ for sufficiently regular functions. We apply our results to two important classes of non-linear PDEs: Porous Medium Equations and Compressible Navier-Stokes Equations, whose solutions have the required Sobolev regularity under sufficiently smooth initial data, and obtain bounds on the PINN residual of RaNN approximations to these solutions.

1 Introduction

Partial Differential Equations (PDEs) are foundational to our understanding of the natural world, with applications across all areas of science and engineering. Many complex phenomena are modelled by non-linear PDEs (e.g. Navier-Stokes, Schrödinger, Porous medium equations), which exhibit disorderly behaviour that renders them intractable to classical analytic/numerical approaches. Therefore, it is crucial to develop numerical methods for solving non-linear PDEs efficiently. In the past years, a variety of deep learning methods for solving PDEs have been introduced and analysed. Neural networks with randomly generated hidden weights (RaNNs) play an important role in many of these methods; either as standalone learning method or as initialization for fully trainable deep neural networks. In both cases, when employing these methods, a precise understanding of the approximation error is crucial for controlling the overall error.

In this paper, we are concerned with the expressivity of RaNNs for learning solutions of non-linear PDEs. To tackle this problem, we derive approximation error bounds for time-dependent Sobolev functions, which encompass the solution spaces for many important non-linear PDEs. Our obtained bounds show that sufficiently regular Sobolev functions can be approximated by RaNNs at a dimension-independent rate of $\frac{1}{2}$. We then apply our results to two important classes of non-linear PDEs: Porous Medium Equations (PME) and Compressible Navier-Stokes Equations, whose solutions have the required Sobolev regularity under sufficiently smooth initial data. We prove that with high probability, there exist RaNNs whose PDE residual for Physics-Informed Neural Networks (PINNs) [53] is bounded by a constant multiple of N^{-1} for these PDEs, where the constant is given explicitly in terms of the Sobolev norms of the solution, the sup-norms of the network and its derivatives and the problem parameters (Corollaries 4.1 and 4.2). In particular, our results provide quantitative approximation guarantees for RaNN-based PINNs for learning non-linear PDEs, as have

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

been extensively studied in computational experiments (cf. the references below). We complement our theoretical analysis by numerical experiments for the two benchmark PDEs. These experiments fit the read-out weights by regression onto a reference solution, which isolates the approximation error bounded by Theorem 3.4 from optimisation and collocation errors. Experiments with Gaussian hidden weights indicate that the rate may also hold beyond the weight distribution assumed in the main result of Theorem 3.4. Note that our results concern expressivity, and we make no claim of practical superiority of RaNNs over fully trained networks.

1.1 Related Works

In recent years, a variety of deep learning-based methods for solving PDEs have been introduced, addressing the challenges of classical mesh-based methods such as finite difference methods. Seminal works include [58], [27] [53], [26]. We refer, e.g., to the survey articles [6, 29, 17, 32] for an extensive overview and further references on deep learning methods for PDEs and their theoretical foundations.

PINNs [53] constitute a flexible and widely applicable deep learning-based approach for solving PDEs. PINNs reframe the problem as training a neural network to solve the PDE, by minimising a loss function that encodes the PDE residual along with the boundary/initial conditions. While this approach has been demonstrated to be highly effective in many settings [9, 33], for non-linear PDEs the loss landscape may become exceptionally complex. This has motivated the use of RaNN-based PINN methods, for which several recent studies have carried out extensive empirical experiments, see, e.g., [25, 56, 55, 61, 69, 71, 39, 20, 14, 45].

RaNNs [34, 51, 52] are neural networks with randomly generated hidden weights. RaNNs have been used both as standalone learning methods and as means for studying the effects of random initialization for neural networks trained using gradient-based optimization [8, 13]. Generalization properties of random feature models have been studied in [54, 41, 38, 15]. In addition to RaNN-based PINNs, many other RaNNs-based methods have been developed for solving PDEs [45, 30, 37, 47]. More broadly, RaNNs and related random feature models have demonstrated state-of-the-art performance and speed across various tasks [7, 23, 28, 49, 72].

The reduced number of trainable parameters of RaNNs in comparison to fully trainable models results in a simpler training phase with a reduced computational cost, potentially at the expense of lower expressivity. Therefore, a precise theoretical understanding of RaNN approximation capabilities is crucial. Quantitative approximation properties of RaNNs for functions in the associated reproducing kernel Hilbert space have been studied in [52, 2, 62]. For smoothness-based function classes, RaNN approximation error bounds were derived in [31, 30] using Barron-type representations and further extended in [46, 21]. In the context of PDEs, [30] obtains a full RaNN learning error analysis free from the curse of dimensionality for a class of linear PDEs. In all these results, the random weight distribution is fixed (e.g. a uniform, normal or Student-t distribution). In contrast, RaNNs also appear as a means of proof for deriving deterministic approximation bounds [3, 4, 5, 57], with RaNN weight distributions depending on the function to be approximated.

While the approximation results in [31, 30] and [46, 21] allow to control the RaNN approximation errors in uniform, mean-squared or Sobolev-norms, respectively, in the context of non-linear PDEs these results would either require strict information on the solution (e.g. finiteness in Barron-ridgelet norms and decay on the Fourier transform of u) which is typically not known, or the bounds would be applicable only for approximating PDE solutions at a fixed point in time. However, for time-dependent non-linear PDEs, solutions often have significantly different behaviour in time versus space (e.g. solutions to Navier-Stokes or semi-linear heat equations). In contrast, our results allow to handle time-dependent functions in mixed Sobolev spaces, as arise in the context of non-linear PDEs. This is especially important because neural networks with randomised architectures have proven to be very effective, even for complex tasks such as solving non-linear PDEs [20].

1.2 Contributions

In this paper, we provide RaNN approximation error bounds tailored to the context of time-dependent, non-linear PDEs. We will denote the width of a RaNN by N. Our paper makes the following contributions:

1. **RaNN approximation bound with dimension-independent rate $N^{-1/2}$:** We derive RaNN approximation error bounds for sufficiently regular time-dependent Sobolev functions (Theorem 3.4). These functions encompass the solution spaces for many important non-linear PDEs. Our unbiased RaNN estimator approximates functions in mixed Sobolev norms $H_t^p H_x^q$ at the rate $N^{-1/2}$ independently of dimension, while only requiring minimal extra regularity in time and space.
2. **Implications for non-linear PDEs:** We showcase the implications of our bounds on two important classes of non-linear PDEs: Porous Medium Equations (PME) and Compressible Navier-Stokes Equations. In particular, our bound supplies the approximation error term in a PINN error decomposition, and yields explicit high-probability bounds on the PINN residual of RaNNs for PME and compressible Navier-Stokes. In numerical experiments that isolate the approximation error, the measured error decays at a rate consistent with $N^{-1/2}$, in agreement with the bounds.

To prove these results, we obtain a specific ridgelet-based representation for L^2 functions (Proposition 3.2) and a higher-order Plancherel-type estimate (Lemma 3.3) that connects Sobolev regularity of u with its ridgelet transform. Our obtained bounds may serve as building block for generalization error analyses of RaNNs for PINN-based learning as obtained for deterministic networks in [22, 42, 43, 1].

2 Preliminaries

2.1 Random Neural Networks

A random neural network (RaNN) is a fully connected neural network with one hidden layer in which the weights are randomly sampled, leaving only the output weights trainable. In this paper, we are interested in studying solutions to PDEs, which typically treat time as a separate dimension. Therefore we will consider time-dependent random neural networks of the following form.

Definition 2.1. A time-dependent random neural network of width N is a function $u_W^{\tau,\mathbf{a},\mathbf{b}} : \mathbb{R}\times\mathbb{R}^d \to \mathbb{R}$ with

$$u_W^{\tau,\mathbf{a},\mathbf{b}}(t,x) = \sum_{i=1}^{N} W_i \sigma(\tau_i t + \mathbf{a}_i \cdot x - b_i), \tag{1}$$

where $\sigma : \mathbb{R} \to \mathbb{R}$ is an activation function, $\mathbf{a}_1, ..., \mathbf{a}_N$ are $\mathbb{R}^d$-valued i.i.d. random variables, $b_1, ..., b_N$ are i.i.d. random variables in $\mathbb{R}$, $\tau_1, ..., \tau_N$ are i.i.d. random variables in $\mathbb{R}$ and $W_1, ..., W_N$ are trainable weights.[1]

A RaNN is used as learning system by optimizing the weights $W_1, ..., W_N$ of $u_W^{\tau,\mathbf{a},\mathbf{b}}$ with respect to a given loss function.

2.2 Physics-informed machine learning

Physics-informed neural networks (PINNs) have been introduced in [53] as an unsupervised learning method for solving partial differential equations. PINNs approximate the solution u to a PDE $\mathcal{L}[u] = 0$ by a neural network u_θ, with θ representing the trainable parameters. The solution is approximated by minimising a loss function $\mathcal{J}[u_\theta]$ encoding the structure of the PDE — including initial/boundary conditions — at collocation points $\{t_p^i, x_p^i\}_{i=1}^M$ in the interior domain $(0,T)\times D$, as well as $\{x_{ic}^i\}_{i=1}^M$, $\{t_{bc}^i\}_{i=1}^M$ on the slices $\{t=0\}\times D$ and $(0,T)\times\partial D$ on which the initial data / boundary conditions are defined, respectively. For example, if we have a PDE on the 1D domain $(0,T)\times(a,b)$ given by $\mathcal{L}[u]=0$ with initial condition $u(0,\cdot)=u_0$ and boundary conditions $u(t,a)=u(t,b)$, the PINN loss function is given by

$$\mathcal{J}[u_\theta] := \frac{1}{M}\sum_{i=1}^{M} |\mathcal{L}[u_\theta]|^2(t_p^i, x_p^i) + \frac{1}{M}\sum_{i=1}^{M} |u_\theta(0, x_{ic}^i) - u_0(x_{ic}^i)|^2 + \frac{1}{M}\sum_{i=1}^{M} |u_\theta(t_{bc}^i, b) - u_\theta(t_{bc}^i, a)|^2.$$

[1] Formally, $W_i = g_i(\tau_1, \ldots, \tau_N, \mathbf{a}_1, ..., \mathbf{a}_N, b_1, \ldots, b_N)$ for measurable functions g_i.

The parameters θ are then iteratively updated using a stochastic optimization algorithm.

2.3 Ridgelet transform

The ridgelet transform $\mathcal{R}_\psi u$ of $u : \mathbb{R} \times \mathbb{R}^d \to \mathbb{R}$ with respect to $\psi : \mathbb{R} \to \mathbb{R}$ is given by

$$\mathcal{R}_\psi u(\tau, \mathbf{a}, b) := \int_{\mathbb{R}^{d+1}} u(t, \mathbf{x}) \psi(\tau t + \mathbf{a} \cdot \mathbf{x} - b) \|(\tau, \mathbf{a})\|^s \, dt d\mathbf{x}, \tag{2}$$

where $\tau \in \mathbb{R}, \mathbf{x} \in \mathbb{R}^d, b \in \mathbb{R}$. The factor $\|(\tau, \mathbf{a})\|^s$ appears for convenience in the literature. We will take $s = 0$ in this paper. Then the dual ridgelet transform $R_\eta^\dagger T$ of $T : \mathbb{R}^{d+2} \to \mathbb{R}$ with respect to $\eta : \mathbb{R} \to \mathbb{R}$ is defined as

$$\mathcal{R}_\eta^\dagger T(t, \mathbf{x}) := \int_{\mathbb{R}^{d+2}} T(\tau, \mathbf{a}, b) \eta(\tau t + \mathbf{a} \cdot \mathbf{x} - b) \|(\tau, \mathbf{a})\|^{-s} d\tau d\mathbf{a} db. \tag{3}$$

The ridgelet transform originates from harmonic analysis in the work of Candès and Donoho [10, 12]. Ridgelets can be described as wavelets in the Radon domain. More precisely, the ridgelet transform is obtained by applying a one-dimensional wavelet transform to the Radon transform of a function. This construction is particularly suited to representing functions with singularities concentrated along hyperplanes, in contrast to classical wavelets which are better suited to point singularities. Ridgelets are also closely related to ridge functions of the form $x \mapsto \varphi(a \cdot x - b)$ [24], which can be interpreted as a single layer neural network. This connection underlies a second area of work linking ridgelet transforms to integral representations of neural networks and their approximation properties. We refer to [44, 11, 36] for early works that developed this research direction. Recent works [64, 48, 59] have further exploited the connection between ridgelet formulations and neural networks to study approximation rates / reconstruction properties for a broad class of activation functions. We refer to [59] for a more comprehensive overview of the ridgelet transform and its properties. In this paper, we extend the ridgelet-based approximation framework to time-dependent Sobolev spaces, which are the natural function spaces for solutions to evolutionary PDEs. We also show how approximation bounds obtained via ridgelet techniques can be propagated to bounds on the PDE residual, which is the quantity minimised in PINN-type methods.

2.4 Notation and structure of the paper

We adopt the shorthand notation $X_t Y_x$ for the Bochner space $X(\mathbb{R}; Y(\mathbb{R}^d))$ (or $X(0, T; Y(D))$, depending on the context). For example, we may write $H_t^p H_x^q$ for the space $H^p(\mathbb{R}; H^q(\mathbb{R}^d))$. We write $\widehat{f}$ to denote the Fourier transform of f, with the convention $\widehat{f}(\omega) = \int e^{-i\omega x} f(x) \, dx$. We also denote by $\|\cdot\|$ the ℓ^2 norm. In Section 3, we prove a representation formula for general time-dependent Sobolev functions, before proving a key inequality that connects the regularity of the ridgelet transform to the original function. We use this to derive our main result Theorem 3.4. In Section 4 we then apply Theorem 3.4 to RaNN-PINN approximators of two benchmark PDEs. Finally, some numerical experiments are carried out which validate the $N^{-1/2}$ convergence rate.

3 RaNN approximations of time-dependent Sobolev functions

Our first step is to obtain an integral representation for u based on the ridgelet transform. This representation will be used later to derive RaNN approximation error bounds for u.

3.1 Obtaining an integral representation

We first introduce the Lizorkin distribution space $\mathcal{S}_0'(\mathbb{R})$, which is the dual space of $\mathcal{S}_0(\mathbb{R})$; the space of Lizorkin functions. $\mathcal{S}_0(\mathbb{R})$ is a closed subspace of the space of Schwartz functions $\mathcal{S}(\mathbb{R})$ and contains the functions $f \in \mathcal{S}(\mathbb{R})$ such that all moments vanish, i.e. $\mathcal{S}_0(\mathbb{R}^d) = \{f \in \mathcal{S} : \int_{\mathbb{R}^d} \mathbf{x}^\alpha f(\mathbf{x}) \, d\mathbf{x} = 0 \text{ for any } \alpha \in \mathbb{N}_0^d\}$. The Lizorkin distribution space $\mathcal{S}_0'(\mathbb{R})$ itself includes many common activations such as $\tanh$, $\mathrm{sigmoid}$ and ReLU. We refer to [59] for a more detailed description of the $\mathcal{S}_0'(\mathbb{R})$ space. In this paper, we will consider activation functions $\sigma : \mathbb{R} \to \mathbb{R}$ that belong to the following subspace of $\mathcal{S}_0'(\mathbb{R})$ for some $k \geq 0$.

Definition 3.1. Let $k \in \mathbb{N}_0$. We say $\sigma \in \mathcal{S}_0'(\mathbb{R})$ belongs to $\mathcal{T}_k$ if (i) there exists $C_1 > 0$ such that

$$\sum_{j=0}^{k} |\sigma^{(j)}(x)| \leq C_1 \quad \forall x \in \mathbb{R}, \tag{4}$$

and (ii) there exists $\delta > 0$ and $\beta \in \mathbb{N}_0$ such that $\zeta^\beta \widehat{\sigma}(\zeta) \in C(-\delta, \delta)$ and for any $\alpha \in \mathbb{N}_0$

$$J_\sigma := \int_{\mathbb{R}} \zeta^{2\alpha+\beta} \widehat{\sigma}(\zeta) e^{-\zeta^2/2} \, d\zeta \neq 0. \tag{5}$$

Examples of admissible activation functions for which our results hold are $\tanh, \cos$ and $\mathrm{sigmoid}$ (each in $\mathcal{T}_k$ for all $k \geq 0$). This is shown in Remark A.1 in the appendix. For $\psi \in \mathcal{S}(\mathbb{R})$ and $m \in \mathbb{N}$ we also define the following (possibly infinite) admissibility constant

$$A_{\psi,m} := \int_{-1}^{1} |\widehat{\psi}(\omega)|^2 |\omega|^{-m} \, d\omega. \tag{6}$$

We now mention the following integral representation formula, which directly follows from results obtained by [59] using the theory of ridgelet transforms.

Proposition 3.2. *Let $m \geq 0$ and $\sigma \in \mathcal{T}_k$ for some $k \geq 0$. A function $u \in L^1(\mathbb{R}; L^1(\mathbb{R}^d))$ with $\widehat{u} \in L^1(\mathbb{R}^{d+1})$ can be expressed as*

$$u(t, \mathbf{x}) = \int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} (\mathcal{R}_\psi u)(\tau, \mathbf{a}, b) \sigma(\tau t + \mathbf{a} \cdot \mathbf{x} - b) \, d\tau d\mathbf{a} db, \tag{7}$$

where $\mathcal{R}_\psi u$ is the ridgelet transform of u with respect to a Schwartz function $\psi \in \mathcal{S}(\mathbb{R})$ such that

$$|\widehat{\psi}(\omega)| \leq C|\omega|^m \qquad \forall\, |\omega| < 1, \tag{8}$$

for some $C > 0$ independent of u, and therefore $A_{\psi,m} < +\infty$. We also have that (σ, ψ) is an admissible pair in the sense of [59], meaning that the following constant is finite and non-zero:

$$K_{\psi,\sigma} := (2\pi)^d \int_{\mathbb{R}} \frac{\overline{\hat{\psi}(\zeta)} \hat{\sigma}(\zeta)}{|\zeta|^{d+1}} \, d\zeta. \tag{9}$$

The construction of an appropriate ψ is given in the proof of Proposition 3.2 in Appendix A.1. From now on, for any activation $\sigma \in \mathcal{T}_k$ and function $u \in L^1(\mathbb{R}; L^1(\mathbb{R}^d))$ we will denote by $\mathcal{R}_\psi u$ the ridgelet transform of u with respect to $\psi \in \mathcal{S}(\mathbb{R})$ constructed according to Proposition 3.2.

3.2 Parseval relation for the ridgelet transform

In order to derive RaNN approximation error bounds, we will need a Parseval-type result which connects the regularity of $\mathcal{R}_\psi u$ in parameter space with the regularity of u in cartesian space. For convenience, we give an outline of the proof in Appendix A.2 and the full proof in Appendix A.3.

Lemma 3.3. *Suppose $u \in H^{p+1}(\mathbb{R}; H^{q+1}(\mathbb{R}^d))$ for some $p, q \geq 0$, and that u is compactly supported in time and space on $[-2T, 2T] \times [-2R, 2R]^d$ for some $T, R \geq 0$. Then there exists $\psi \in \mathcal{S}(\mathbb{R})$ with*

$$\int_{\mathbb{R}^{d+2}} |\mathcal{R}_\psi u(\tau, \mathbf{a}, b)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \cdot (1 + b^2) \, d\tau d\mathbf{a} db \leq \mathcal{L}_\psi \|u\|^2_{H^{p+1}(\mathbb{R}; H^{q+1}(\mathbb{R}^d))}, \tag{10}$$

where, for $M = (2p + 2q + d + 3)/2$ and $K_{T,R} := (2T + \lceil p \rceil + 1)^2 + (2R\sqrt{d} + \lceil q \rceil + 1)^2$,

$$\mathcal{L}_\psi := (4\pi + \|\psi\|^2_{L^2(\mathbb{R})})(1 + 4K_{T,R}) + 2(M+1)^2 \left(4\pi + \|\psi'\|^2_{L^2(\mathbb{R})}\right). \tag{11}$$

3.3 Random neural network approximation error bounds

In this subsection we provide our main result for approximating a function $u \in H^p(0, T; H^q(D))$ using RaNNs. The full proof is deferred to Appendix A.4.

Theorem 3.4. *Fix a bounded subset $(0,T)\times D\subset\mathbb{R}_+\times\mathbb{R}^d$ where D is Lipschitz and let $u\in H^{p+s_1}(0,T;H^{q+s_2}(D))$ for $p,q\geq 0, s_1>3/2$ and $s_2>(d+2)/2$. Furthermore, let $\sigma\in\mathcal{T}_{p+q}$. There exist weights $\{W_i\}_{i=1}^N$ such that the following random neural network is an unbiased estimator of u:*

$$u_N(t,\mathbf{x})=\sum_{i=1}^N W_i\sigma(\tau_i t+\mathbf{a}_i\cdot\mathbf{x}-b_i)$$

for $(\tau_i,\mathbf{a}_i,b_i)\sim\pi$, where

$$\pi(\tau,\mathbf{a},b)=\frac{1}{C_\pi}(1+\tau^2)^{-\lambda_\tau}(1+\|\mathbf{a}\|^2)^{-\lambda_a}(1+b^2)^{-1}, \tag{12}$$

where $\lambda_\tau=s_1-1,\lambda_a=s_2-1$ and C_π is the normalisation constant

$$C_\pi:=\int_{\mathbb{R}}\frac{1}{(1+\tau^2)^{\lambda_\tau}}\,d\tau\int_{\mathbb{R}^d}\frac{1}{(1+\|\mathbf{a}\|^2)^{\lambda_a}}d\mathbf{a}\int_{\mathbb{R}}\frac{1}{(1+b^2)}\,db. \tag{13}$$

The neural network u_N satisfies

$$\begin{aligned}&\mathbb{E}_\Theta\left(\|u-u_N\|^2_{H^p(0,T;H^q(D))}\right)\\&\quad\leq\frac{C_\Omega C_\pi\|\sigma^{(p+q)}\|^2_\infty T|D|(p+1)\binom{q+d}{d}\mathcal{L}_\psi}{N}\|u\|^2_{H^{p+s_1}(0,T;H^{q+s_2}(D))},\end{aligned} \tag{14}$$

where C_Ω is a constant dependent on p, q, d and the domain, $\|\sigma^{(p+q)}\|_\infty:=\max_{0\leq j\leq p+q}\|\sigma^{(j)}\|_\infty$ and $\mathcal{L}_\psi$ is given by (11)*.*

Remark 3.5. Let us note that the space of admissible functions does depend on d through the requirement $s_2>(d+2)/2$. Nonetheless, for a given function which satisfies the conditions of the theorem, the rate of decay $N^{-1/2}$ with respect to the network width N is independent of d. This is an improvement upon the work of [21] (see Theorem 3.9) which required $u\in H^s(\mathbb{R}^d)$ for $s\geq(d+9)/2$ for estimates in $H^1(\mathbb{R}^d)$ and $H^2(\mathbb{R}^d)$, with slower rates. Moreover, we obtain estimates in a time-dependent norm, whereas previous results only obtained error rates for solutions at a fixed time (see also Proposition 4.24 of [46]).

Remark 3.6 (Growth of the coefficient with respect to dimension). Obtaining a more explicit bound (in terms of d) on the coefficient

$$C_\Omega C_\pi\|\sigma^{(p+q)}\|^2_\infty T|D|(p+1)\binom{q+d}{d}\mathcal{L}_\psi \tag{15}$$

appearing in (14) is difficult for general u and arbitrary p,q. Nonetheless, we show in Remark A.2 that (14) holds with a constant that grows at most polynomially in d in the case where $\sigma\in\mathcal{T}_k$ and $u\in H^p(\mathbb{R};H^q(\mathbb{R}^d))$ (i.e. u is already defined on the full space, support in $[-2T,2T]\times[-2R,2R]^d$, $|D|\leq 1$).

4 Applications to non-linear PDEs

We look at two representative non-linear PDEs (the Porous Medium Equation and compressible Navier-Stokes) in order to understand how one can obtain specific asymptotic bounds on the residual loss and approximation error using the structure of the PDE.

4.1 Porous Medium Equation

The porous medium equation (PME) is an important example of a non-linear parabolic PDE that models the flow of gases through porous media. In dimension $d\in\mathbb{N}$ and for $m>0$, a function $u:\mathbb{R}_+\times\mathbb{R}^d\to\mathbb{R}$ is said to solve the porous medium equation if

$$\begin{cases}\partial_t u-\Delta(u^m)=0,\ \text{on}\ \mathbb{R}_+\times\mathbb{R}^d.\\ u(0,\cdot)=u_0,\end{cases} \tag{16}$$

In the case where u_0 is bounded below by a positive constant and $u_0-\bar{c}\in H^k(\mathbb{R}^d)$ for some constant $\bar{c}>0$ and $k\in\mathbb{N}$, the following result is classically known.

- [66]: Consider initial data u_0 satisfying

$$u_0 - \bar{c} \in H^k(\mathbb{R}^d) \text{ for some constant } \bar{c} > 0, \;\; k \in \mathbb{N}, \;\; 0 < c \leq u_0 \leq C, \tag{17}$$

Then there exists a classical solution u to (16) with $c \leq u(t,x) \leq C$ and

$$u - \bar{c} \in C^\infty((0,\infty) \times \mathbb{R}^d) \cap C([0,\infty); H^k(\mathbb{R}^d)).$$

In practice, we simulate solutions on a bounded domain $(0,T) \times D$. In this case, the smoothness of the solution implies that $u \in H^k((t_0,T) \times D)$ for any $k \geq 0$ and any $0 < t_0 < T$. We can deduce an approximation result on solutions to PME using Theorem 3.4. Before we state the result, let us note that the classical loss function one would use when training a physics-informed neural network to approximate solution u is a discretisation of the following metric:

$$\mathcal{J}_{PDE}(u_N) = \int_{(t_0,T)\times D} |\partial_t u_N - \Delta(u_N^m)|^2 \, dtdx. \tag{18}$$

Corollary 4.1. *Suppose u solves* (16) *with initial data u_0 satisfying* (17)*. Then there exists a random neural network $u_N(t,x)$ such that on the domain $(t_0,T) \times D$ for any $0 < t_0 < T$:*

1. *For any $p, q \geq 0$,*

$$\begin{aligned} \mathbb{E}_\Theta(\|u - u_N\|^2_{H_t^p H_x^q}) &\leq \frac{C_\Omega C_\pi \|\sigma^{(p+q)}\|^2_\infty T|D|(p+1)\binom{q+d}{d}\mathcal{L}_\psi}{N} \|u\|^2_{H_t^{p+s_1} H_x^{q+s_2}} \\ &=: \frac{\mathcal{M}_\psi}{N} \|u\|^2_{H_t^{p+s_1} H_x^{q+s_2}}, \end{aligned} \tag{19}$$

   for any $s_1 > 3/2, s_2 > (d+2)/2$.

2. *For any $\delta \in (0,1)$, with probability $1-\delta$ over the network parameters, the PDE residual* (18) *can be bounded as:*

$$\mathcal{J}_{PDE}(u_N) \leq \frac{C_m(1 + L^2 + C_{emb}^2 \|u\|^2_{L_t^\infty H_x^{2+k}})\mathcal{M}_\psi}{N\delta} \|u\|^2_{H_t^3 H_x^{3+k}}, \tag{20}$$

   for $k > d/2$, where $L := \|u_N\|_{L^\infty_{t,x}} + \|\nabla u_N\|_{L^\infty_{t,x}} + \|\Delta u_N\|_{L^\infty_{t,x}}$ is finite since u_N is smooth. Here, C_{emb} is the constant arising from the Sobolev embedding $H_x^k \hookrightarrow L_x^\infty$ and C_m is a polynomial in the PME parameter m.

In other words, for every $\delta \in (0,1)$ and every width N, with probability at least $1-\delta$ over the drawn hidden weights there exist read-out weights for which $\mathcal{J}_{PDE}(u_N) \leq C_L/(N\delta)$. Here the constant C_L depends on m (through C_m), on d and D (through C_{emb} and $\mathcal{M}_\psi$), on T, on the Sobolev norms of u appearing in (20)*, and on the network through L. In particular, the PDE residual decays at rate $1/N$ along any sequence of networks u_N for which L remains bounded.*

This is an existence statement; it does not assert that a particular training algorithm finds these networks.

4.2 Compressible Navier-Stokes Equations

We now look at a more delicate example, which is a system of equations that does not enjoy the instantaneous regularisation property of the PME. The compressible Navier-Stokes equations in dimension d are given by

$$\begin{cases} \partial_t \rho + \text{div}(\rho \mathbf{u}) = 0, \\ \partial_t(\rho \mathbf{u}) + \text{div}(\rho \mathbf{u} \otimes \mathbf{u}) - \nabla(\mu(\rho)\text{div } \mathbf{u}) + \nabla p(\rho) = 0, \end{cases} \tag{21}$$

on $(0,T) \times D$, where

$$p(\rho) = \rho^\gamma, \;\; \gamma > 0 \text{ and } \mu(\rho) = \rho^\alpha, \; \alpha > 0. \tag{22}$$

The solution is a pair $(\rho, \mathbf{u})$ where the density $\rho : \mathbb{R}_+ \times \mathbb{R}^d \to \mathbb{R}$ is the scalar density and $\mathbf{u} : \mathbb{R}_+ \times \mathbb{R}^d \to \mathbb{R}^d$ is the vector-valued velocity. We look at the one-dimensional setting $D = (0,1)$ with periodic boundary conditions, where global-in-time classical solutions exist under mild assumptions. In this case, the following global well-posedness result applies.

- Theorem 1.5, [16]. Given initial data (ρ_0, u_0) satisfying

$$(\rho_0, u_0) \in H^k(D),\ k \geq 3,\ \ 0 < \delta \leq \rho_0 \leq C, \tag{23}$$

then there exists a unique solution (ρ, u) on $(0, T)$ to (21) with initial data (ρ_0, u_0) such that

$$\rho \in C(0, T; H^k(D)),\ \ u \in C(0, T; H^k(D)) \cap L^2(0, T; H^{k+1}(D)). \tag{24}$$

We will take the pressureless case $p = 0$ and constant viscosity $\mu(\rho) = \mu$ for simplicity, although our computations can be easily extended to handle a more general setting where p, μ are smooth and convex (e.g. $p(\rho) = \rho^\gamma, \mu(\rho) = \rho^\alpha$, for $\alpha, \gamma > 0$). In this case, the PINN residual loss is the discretisation of the following loss functions:

$$\begin{aligned} \mathcal{J}^1_{PDE}(\mathbf{v}_N) &:= \int_{(0,T)\times D} |\partial_t(\rho_N) + \partial_x(\rho_N u_N)|^2 \, dxdt, \\ \mathcal{J}^2_{PDE}(\mathbf{v}_N) &:= \int_{(0,T)\times D} |\partial_t(\rho_N u_N) + \partial_x(\rho_N u_N^2) - \mu \partial_x^2 u_N|^2 \, dxdt. \end{aligned} \tag{25}$$

To apply Theorem 3.4, we first define the product norm $\|(f, g)\|^2_{H^p_t H^q_x} := \|f\|^2_{H^p_t H^q_x} + \|g\|^2_{H^p_t H^q_x}$. We then have the following result whose proof is given in Appendix A.6.

Corollary 4.2. *Suppose (ρ, u) is a solution to* (21) *in dimension $d = 1$ generated by initial data satisfying $(\rho_0, u_0) \in H^k(D),\ k \geq 9,\ 0 < \delta \leq \rho_0 \leq C$. Then there exists a random neural network $\mathbf{v}_N = (\rho_N, u_N)$ such that*

- *For any $q \geq 0$,*

$$\mathbb{E}_\Theta(\|(\rho, u) - \mathbf{v}_N\|^2_{L^2_t H^q_x}) \leq \frac{\mathcal{M}_\psi}{N}(\|\rho\|^2_{H^{s_1}_t H^{q+s_2}_x} + \|u\|^2_{H^{s_1}_t H^{q+s_2}_x}), \tag{26}$$

for $s_1, s_2 > 3/2$, where $\mathcal{M}_\psi$ is the coefficient from (14)*, later defined as $\mathcal{M}_\psi$ in* (93)*.*

- *For any $\delta \in (0, 1)$ with probability $1 - \delta$ over the network parameters, the PDE residual loss $\mathcal{J}_{PDE} := \mathcal{J}^1_{PDE} + \mathcal{J}^2_{PDE}$ can be bounded as:*

$$\mathcal{J}_{PDE}(\mathbf{v}_N) \leq \left(\frac{C_L \mathcal{M}_\psi}{N\delta}\right)(\|\rho\|^2_{H^3_t H^3_x} + (2\mu + 1)\|u\|^2_{H^3_t H^4_x}), \tag{27}$$

where $L := \|(\rho_N, u_N)\|^2_{W^{1,\infty}_{t,x}} + \|(\rho, u)\|^2_{W^{1,\infty}_{t,x}}$ is finite since ρ_N, u_N, ρ, u are sufficiently smooth, and

$$C_L := \left[16L^2 + 128L + 2\mu + 2 + 64(L + 1)\|u\|^2_{W^{1,\infty}_{t,x}}\right].$$

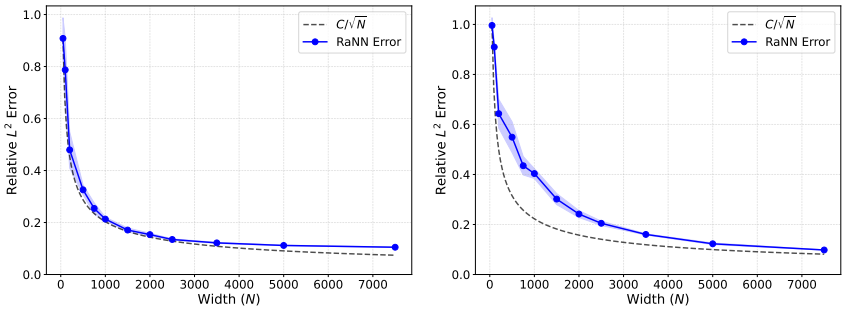
Remark 4.3. The choice $k \geq 9$ ensures that the norms on the right-hand side of (27) are finite. With this regularity, one can also show that the boundary/initial condition residuals can be controlled similarly to $\mathcal{J}_{PDE}$, using trace inequalities.

The use of these bounds inside solvers for the non-linear equations, via direct minimisation of the PDE residual or via Picard linearisation, is discussed in Appendix D.

5 Numerical illustrations

In this section, we provide numerical experiments to validate the obtained bounds by studying the effect of network width N on the error in practice.[2] A secondary goal of these experiments is to study whether the convergence rates extend beyond the heavy-tailed distribution considered in Theorem 3.4 to normally-distributed weights. We consider the Porous Medium Equation (PME) in dimensions $d = 1 - 5$ and the compressible Navier-Stokes equations in $d = 1$.

[2]Code: `https://github.com/M6LI/RaNN-Expressivity-NonLinearPDEs`.



(a) $d = 4$ (b) $d = 5$

Figure 1: Approximation error of RaNNs solving PME in dimensions $d = 4, 5$ with varying widths. The shaded band indicates the region within one standard deviation of the mean relative L^2 error.

5.1 Experiments for PME

We consider the PME with $m = 2$. An exact, self-similar solution to the PME is the Barenblatt-Kompaneets-Zeldovich solution

$$u(t,x) = \frac{1}{t^\alpha}\left(b - \frac{m-1}{2m}\beta\frac{\|x\|^2}{t^{2\beta}}\right)_+^{\frac{1}{m-1}}, \tag{28}$$

where $\|\cdot\|$ is the ℓ^2 norm, $(\cdot)_+$ is the positive part and $\alpha = \frac{d}{d(m-1)+2}$ and $\beta = \frac{1}{d(m-1)+2}$. This solution is compactly supported but not differentiable at the edges of the support, which causes difficulty for numerical schemes. We now investigate the effect of the network width N on the relative error between the network and the solution u. This allows us to validate the convergence rate obtained in our theoretical results. Theorem 3.4 shows that for given $u : (0,T) \times \mathbb{R}^d \to \mathbb{R}$ in some Sobolev space, a RaNN $\hat{u}_N$ of width N is able to achieve relative error $\mathcal{R}(u,\hat{u}) = \|\hat{u} - u\|_{L^2_{t,x}} / \|u\|_{L^2_{t,x}}$ bounded by $CN^{-1/2}$, where C depends on d, Ω and u.

To validate this convergence rate in practice, we train a RaNN to approximate the PME solution (28) in dimensions $d = 1, ..., 5$. In each dimension, we take a set of widths $N \in \{N_1, ..., N_k\}$, train the network for each width and plot the relative error of the final network against the true solution. For each dimension d and width N, we sample $M = 10N$ points ($M \gg N$ to ensure the problem remains well-posed) with a mixed strategy; 50% of the points are sampled uniformly on $(0,1)^d$ and 50% are sampled uniformly on $[0.2, 0.8]^d$, which is a box focused on the initial support of the solution. Then we find weights $\mathbf{W} = \{W_i\}_{i=1}^N$ minimising the Ridge regression loss and evaluate the relative error between RaNN approximation and true solution. The cases $d = 4, 5$ are included in Figure 1, while the cases $d = 1, 2, 3$ can be found in Appendix B, Figure 3. The error curves are plotted on log-log scales in Figure 4, Appendix B.

Remark 5.1 (Choice of weight distribution). The experiments above use normally-distributed hidden weights, in order to test whether the convergence rate extends beyond the heavy-tailed distribution π of Theorem 3.4 to a distribution that is more common in practice. We have also repeated these experiments with weights drawn from π itself, and with a (heavy-tailed) Student-t distribution, and obtained similar rates in both cases; these results are reported in Appendix B.1, together with a discussion of why the proof of Theorem 3.4 relies on the heavy-tailed form of π (Remark B.2).

Remark 5.2 (Training on the PDE residual). In Appendix B.2 we also train RaNNs only by minimising the PINN loss, without using the solution in the interior, for the PME in $d = 1, 2$ on the interior of the support, where the hypotheses of Corollary 4.1 hold. We compare with fully trainable PINNs with the same number N of trainable parameters, trained with the same optimiser. The RMS PDE residual of the RaNN decays at least as fast as $N^{-1/2}$, with fitted slopes -0.76 ± 0.07 ($d = 1$) and -1.47 ± 0.29 ($d = 2$), which is consistent with Corollary 4.1. The fully trainable network attains smaller residuals, by a factor of 5.7 ($d = 1$) and 1.7 ($d = 2$) at $N = 1280$, but its measured rates vary considerably more across random initialisations.

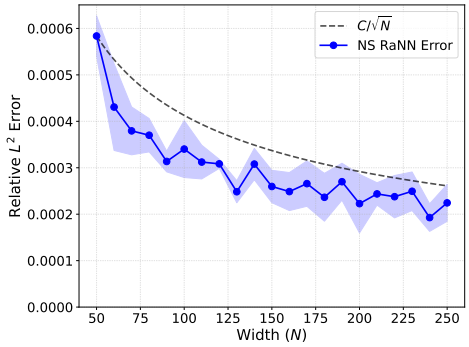


Figure 2: Approximation error of RaNNs solving the compressible NS system with varying widths. The shaded band indicates the region within one standard deviation of the mean relative L^2 error.

5.2 Experiments for compressible Navier-Stokes

We now turn to the one-dimensional compressible Navier-Stokes system, given by (21). As baseline, we consider the travelling shock-wave solutions considered by [18], where the pressure $p_\epsilon(v) = \epsilon/(v-1)^\gamma$ for $\gamma > 0$ is taken, where $v = 1/\rho$. The travelling wave solutions can be obtained by taking the ansatz $(v,u)(t,x) = (\mathfrak{v},\mathfrak{u})(x-st)$, where s is the shock speed. This reduces the PDE to an ODE for $\mathfrak{v}$. The velocity $\mathfrak{u}$ can then be obtained from the relationship $\mathfrak{u} = s(v_+ - \mathfrak{v})$, which follows from the conservation of mass.

For our experiment, we consider the domain $(0,T)\times(-5,5)$ with $T = 1.0$, $\mu = 10$, $\epsilon = 10^{-3}$, $\gamma = 2$. We then compute RaNN approximations $(\mathfrak{v}_N, \mathfrak{u}_N)$ for different widths N and measure the relative error to the baseline solution $(\mathfrak{v},\mathfrak{u})$. The results can be seen in Figure 2, which shows that the errors are close to the $C/\sqrt{N}$ curve, in support of the upper bound of Theorem 3.4. Further experimental details and a visual depiction of the travelling-wave solution can be found in Appendix B.3. To explain the choice of dimensions, we note that high-dimensional studies of randomised neural network approximations have been carried out for linear PDEs (see e.g. [30]), but for non-linear equations high quality analytical reference solutions are rare: travelling waves for compressible Navier-Stokes are essentially one-dimensional, and for the Barenblatt profile the support spreads more slowly as d grows, which amplifies sampling errors.

6 Conclusion

In this work, we have shown that neural networks with randomly generated hidden weights (RaNNs) approximate sufficiently regular functions in time-dependent Sobolev spaces at the dimension-free rate $N^{-1/2}$. This requires $u \in H_t^{p+s_1} H_x^{q+s_2}$ with $s_1 > 3/2$ and $s_2 > (d+2)/2$. An important future direction would be to explore whether these assumptions can be relaxed; in Remark A.3 we further discuss when these assumptions hold, which regimes they exclude, and provide a preliminary discussion on how they might be relaxed.

Future work may additionally investigate whether our results can be extended to solutions which are less smooth than those considered here. Moreover, our results assume a heavy-tailed weight distribution, while numerical experiments indicate that the same rate also holds for Gaussian weights. It would also be interesting to see whether the constant of proportionality (appearing in (14)) can be improved, either using the theory of ridgelet transforms or an alternative representation. It will also be important to explore whether RaNNs are prone to the same spectral bias issues that PINNs face, especially for complex PDEs such as compressible Navier-Stokes in the turbulent regime.

Acknowledgments and Disclosure of Funding

The work of M.A. Mehmood was supported by the Roth Scholarship funded by the Department of Mathematics, Imperial College London.

References

[1] Miguel A. Alejo, Lucrezia Cossetti, Luca Fanelli, Claudio Muñoz, and Nicolás Valenzuela. Error bounds for Physics Informed Neural Networks in Nonlinear Schrödinger equations placed on unbounded domains. *arXiv:2409.17938*, 2024.

[2] Francis Bach. Breaking the curse of dimensionality with convex neural networks. *Journal of Machine Learning Research*, 18(19):1–53, 2017.

[3] Andrew R Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information theory*, 39(3):930–945, 1993.

[4] Andrew R. Barron. Approximation and estimation bounds for artificial neural networks. *Machine Learning*, 14:115–133, 1994.

[5] Andrew R. Barron and Jason M. Klusowski. Approximation and estimation for high-dimensional deep learning networks. *Preprint, arXiv 1809.03090*, 2018.

[6] Christian Beck, Martin Hutzenthaler, Arnulf Jentzen, and Benno Kuckuck. An overview on deep learning-based approximation methods for partial differential equations. *arXiv preprint arXiv:2012.12348*, 2020.

[7] Erik L Bolager, Iryna Burak, Chinmay Datar, Qing Sun, and Felix Dietrich. Sampling weights of deep neural networks. In *Advances in Neural Information Processing Systems*, volume 36, pages 63075–63116. Curran Associates, Inc., 2023.

[8] Alina Braun, Michael Kohler, Sophie Langer, and Harro Walk. Convergence rates for shallow neural networks learned by gradient descent. *Bernoulli*, 30(1):475–502, 2024.

[9] Shengze Cai, Zhiping Mao, Zhicheng Wang, Minglang Yin, and George Em Karniadakis. Physics-informed neural networks (PINNs) for fluid mechanics: a review. *Acta Mech. Sin.*, 37(12):1727–1738, 2021.

[10] Emmanuel J. Candès. *Ridgelets: Theory and Applications*. PhD thesis, Stanford University, 1998.

[11] Emmanuel J Candès. Harmonic analysis of neural networks. *Applied and Computational Harmonic Analysis*, 6(2):197–218, 1999.

[12] Emmanuel J Candès and David L Donoho. Ridgelets: A key to higher-dimensional intermittency? *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, 357(1760):2495–2509, 1999.

[13] Luigi Carratino, Alessandro Rudi, and Lorenzo Rosasco. Learning with SGD and random features. In *Advances in Neural Information Processing Systems*, volume 31, 2018.

[14] Jingrun Chen, Xurong Chi, Weinan E, and Zhouwang Yang. Bridging traditional and machine learning-based algorithms for solving pdes: The random feature method. *arXiv preprint arXiv:2207.13380*, 2022.

[15] Tin Sum Cheng, Aurelien Lucchi, Anastasis Kratsios, Ivan Dokmanić, and David Belius. A theoretical analysis of the test error of finite-rank kernel ridge regression. In *Advances in Neural Information Processing Systems*, volume 36, pages 4767–4798. Curran Associates, Inc., 2023.

[16] Peter Constantin, Theodore D Drivas, Huy Q Nguyen, and Federico Pasqualotto. Compressible fluids and active potentials. In *Annales de l'Institut Henri Poincaré C, Analyse non linéaire*, volume 37, pages 145–180. Elsevier, 2020.

[17] Salvatore Cuomo, Vincenzo Schiano Di Cola, Fabio Giampaolo, Gianluigi Rozza, Maziar Raissi, and Francesco Piccialli. Scientific machine learning through physics-informed neural networks: where we are and what's next. *J. Sci. Comput.*, 92(3):Paper No. 88, 62, 2022.

[18] Anne-Laure Dalibard and Charlotte Perrin. Existence and stability of partially congested propagation fronts in a one-dimensional Navier-Stokes model. *Communications in Mathematical Sciences*, 2020.

[19] Anne-Laure Dalibard and Charlotte Perrin. Local and global well-posedness of one-dimensional free-congested equations. *Annales Henri Lebesgue*, 2024.

[20] Chinmay Datar, Taniya Kapoor, Abhishek Chandra, Qing Sun, Erik Lien Bolager, Iryna Burak, Anna Veselovska, Massimo Fornasier, and Felix Dietrich. Fast training of accurate physics-informed neural networks without gradient descent. In *International Conference on Learning Representations (ICLR)*, 2026.

[21] T De Ryck, S Mishra, Y Shang, and F Wang. Approximation theory and applications of randomized neural networks for solving high-dimensional pdes. *arXiv preprint arXiv:2501.12145*, 2025.

[22] Tim De Ryck and Siddhartha Mishra. Numerical analysis of physics-informed neural networks and related models in physics-informed machine learning. *Acta Numer.*, 33:633–713, 2024.

[23] Angus Dempster, Daniel F. Schmidt, and Geoffrey I. Webb. Hydra: competing convolutional kernels for fast and accurate time series classification. *Data Mining and Knowledge Discovery*, 37(5):1779–1805, Sep 2023.

[24] David L Donoho. Ridge functions and orthonormal ridgelets. *Journal of Approximation Theory*, 111(2):143–179, 2001.

[25] Vikas Dwivedi and Balaji Srinivasan. Physics informed extreme learning machine (pielm)–a rapid method for the numerical solution of partial differential equations. *Neurocomputing*, 391:96–118, 2020.

[26] W. E and B. Yu. The Deep Ritz method: A deep learning-based numerical algorithm for solving variational problems. *Commun. Math. Stat.*, 6(1):1–12, 2018.

[27] Weinan E, Jiequn Han, and Arnulf Jentzen. Deep learning-based numerical methods for high-dimensional parabolic partial differential equations and backward stochastic differential equations. *Commun. Math. Stat.*, 5(4):349–380, 2017.

[28] Guglielmo Gattiglio, Lyudmila Grigoryeva, and Massimiliano Tamborrino. Randnet-parareal: a time-parallel pde solver using random neural networks. *Advances in Neural Information Processing Systems*, 37:94993–95025, 2024.

[29] Maximilien Germain, Huyên Pham, and Xavier Warin. Neural networks-based algorithms for stochastic control and PDEs in finance. *arXiv:2101.08068*, pages 1–27, 2021.

[30] Lukas Gonon. Random feature neural networks learn black-scholes type pdes without curse of dimensionality. *Journal of Machine Learning Research*, 24(189):1–51, 2023.

[31] Lukas Gonon, Lyudmila Grigoryeva, and Juan-Pablo Ortega. Approximation bounds for random neural networks and reservoir systems. *The Annals of Applied Probability*, 33(1):28–69, 2023.

[32] Lukas Gonon, Arnulf Jentzen, Benno Kuckuck, Siyu Liang, Adrian Riekert, and Philippe von Wurstemberger. An overview on machine learning methods for partial differential equations: from physics informed neural networks to deep operator learning. *arXiv:2408.13222*, pages 1–59, 2024.

[33] Zheyuan Hu, Khemraj Shukla, George Em Karniadakis, and Kenji Kawaguchi. Tackling the curse of dimensionality with physics-informed neural networks. *Neural Networks*, 176:106369, 2024.

[34] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: theory and applications. *Neurocomputing*, 70(1-3):489–501, 2006.

[35] Jeffrey Humpherys, Olivier Lafitte, and Kevin Zumbrun. Stability of isentropic navier–stokes shocks in the high-mach number limit. *Communications in Mathematical Physics*, 293(1):1–36, 2010.

[36] Yoshifusa Ito. Representation of functions by superpositions of a step or sigmoid function and their applications to neural network theory. *Neural Networks*, 4(3):385–394, 1991.

[37] Antoine Jacquier and Zan Zuric. Random neural networks for rough volatility. *arXiv preprint arXiv:2305.01035*, 2023.

[38] Samuel Lanthaler and Nicholas H. Nelsen. Error bounds for learning with vector-valued random features. In *Advances in Neural Information Processing Systems*, volume 36, pages 71834–71861. Curran Associates, Inc., 2023.

[39] Jiale Linghu, Weifeng Gao, Hao Dong, Fei Wang, and Yufeng Nie. Higher-order multi-scale physics-informed randomized neural network method for efficient and high-accuracy simulation of dynamic thermo-mechanical coupling problems. *Mathematics and Mechanics of Solids*, 0(0):10812865251362165, 2025.

[40] Corrado Mascia and Kevin Zumbrun. Stability of large-amplitude viscous shock profiles of hyperbolic-parabolic systems. *Archive for rational mechanics and analysis*, 172(1):93–131, 2004.

[41] Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and double descent curve. *Comm. Pure Appl. Math.*, 75(4):667–766, 2022.

[42] Siddhartha Mishra and Roberto Molinaro. Estimates on the generalization error of physics-informed neural networks for approximating a class of inverse problems for PDEs. *IMA J. Numer. Anal.*, 42(2):981–1022, 2022.

[43] Siddhartha Mishra and Roberto Molinaro. Estimates on the generalization error of physics-informed neural networks for approximating PDEs. *IMA J. Numer. Anal.*, 43(1):1–43, 2023.

[44] Noboru Murata. An integral representation of functions using three-layered networks and their approximation bounds. *Neural Networks*, 9(6):947–956, 1996.

[45] Nicholas H. Nelsen and Andrew M. Stuart. The random feature model for input-output maps between banach spaces. *SIAM Journal on Scientific Computing*, 43(5):A3212–A3243, 2021.

[46] Ariel Neufeld and Philipp Schmocker. Universal approximation property of banach space-valued random feature models including random neural networks. *arXiv preprint arXiv:2312.08410*, 2023.

[47] Ariel Neufeld, Philipp Schmocker, and Sizhou Wu. Full error analysis of the random deep splitting method for nonlinear parabolic pdes and pides. *Communications in Nonlinear Science and Numerical Simulation*, 143:108556, 2025.

[48] Greg Ongie, Rebecca Willett, Daniel Soudry, and Nathan Srebro. A function space view of bounded norm infinite width relu nets: The multivariate case. *arXiv preprint arXiv:1910.01635*, 2019.

[49] Ameya Prabhu, Shiven Sinha, Ponnurangam Kumaraguru, Philip Torr, Ozan Sener, and Puneet K. Dokania. Random representations outperform online continually learned representations. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024.

[50] Nasim Rahaman, Aristide Baratin, Devansh Arpit, Felix Draxler, Min Lin, Fred Hamprecht, Yoshua Bengio, and Aaron Courville. On the spectral bias of neural networks. In *International conference on machine learning*, pages 5301–5310. PMLR, 2019.

[51] Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. *Advances in neural information processing systems*, 20, 2007.

[52] Ali Rahimi and Benjamin Recht. Uniform approximation of functions with random bases. In *2008 46th annual allerton conference on communication, control, and computing*, pages 555–561. IEEE, 2008.

[53] Maziar Raissi, Paris Perdikaris, and George E Karniadakis. Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations. *Journal of Computational physics*, 378:686–707, 2019.

[54] Alessandro Rudi and Lorenzo Rosasco. Generalization properties of learning with random features. In *Advances in Neural Information Processing Systems*, pages 3215–3225, 2017.

[55] Yong Shang and Fei Wang. Randomized neural networks with petrov–galerkin methods for solving linear elasticity and navier–stokes equations. *Journal of Engineering Mechanics*, 150(4):04024010, 2024.

[56] Yong Shang, Fei Wang, and Jingbo Sun. Randomized neural network with petrov–galerkin methods for solving linear and nonlinear partial differential equations. *Communications in Nonlinear Science and Numerical Simulation*, 127:107518, 2023.

[57] Jonathan W. Siegel and Jinchao Xu. Approximation rates for neural networks with general activation functions. *Neural Networks*, 128:313–321, 2020.

[58] J. Sirignano and K. Spiliopoulos. DGM: A deep learning algorithm for solving partial differential equations. *J. Comput. Phys.*, 375:1339–1364, 2018.

[59] Sho Sonoda and Noboru Murata. Neural network with unbounded activation functions is universal approximator. *Applied and Computational Harmonic Analysis*, 43(2):233–268, 2017.

[60] Elias M Stein. *Singular integrals and differentiability properties of functions*, volume 30 of *Princeton Mathematical Series*. Princeton university press, 1970.

[61] Jingbo Sun, Suchuan Dong, and Fei Wang. Local randomized neural networks with discontinuous galerkin methods for partial differential equations. *Journal of Computational and Applied Mathematics*, 445:115830, 2024.

[62] Yitong Sun, Anna Gilbert, and Ambuj Tewari. On the approximation properties of random relu features. *arXiv preprint arXiv:1810.04374*, 2018.

[63] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in neural information processing systems*, 33:7537–7547, 2020.

[64] Michael Unser. Ridges, neural networks, and the radon transform. *Journal of Machine Learning Research*, 24(37):1–33, 2023.

[65] Alexis F Vasseur and Lei Yao. Nonlinear stability of viscous shock wave to one-dimensional compressible isentropic Navier–Stokes equations with density dependent viscous coefficient. *Communications in Mathematical Sciences*, 14(8):2215–2228, 2016.

[66] Juan Luis Vázquez. *The porous medium equation: mathematical theory*. Oxford university press, 2007.

[67] Sifan Wang, Shyam Sankaran, and Paris Perdikaris. Respecting causality is all you need for training physics-informed neural networks. *arXiv preprint arXiv:2203.07404*, 2022.

[68] Sifan Wang, Shyam Sankaran, Hanwen Wang, and Paris Perdikaris. An expert's guide to training physics-informed neural networks. *arXiv preprint arXiv:2308.08468*, 2023.

[69] Yiran Wang and Suchuan Dong. An extreme learning machine-based method for computational pdes in higher dimensions. *Computer Methods in Applied Mechanics and Engineering*, 418:116578, 2024.

[70] Zhi-Qin John Xu, Yaoyu Zhang, Tao Luo, Yanyang Xiao, and Zheng Ma. Frequency principle: Fourier analysis sheds light on deep neural networks. *arXiv preprint arXiv:1901.06523*, 2019.

[71] Jinyong Ying, Jingying Hu, Zuoshunhua Shi, and Jiao Li. An accurate and efficient continuity-preserved method based on randomized neural networks for elliptic interface problems. *SIAM Journal on Scientific Computing*, 46(5):C633–C657, 2024.

[72] Nikita Zozoulenko, Thomas Cass, and Lukas Gonon. Random feature representation boosting. In *Forty-second International Conference on Machine Learning*, 2025.

A Proofs

A.1 Proof of Proposition 3.2

Proof. Fix $m \geq 0$ and an arbitrary $\psi \in \mathcal{S}(\mathbb{R})$. Using the definition of the ridgelet transform from [59], we have for any $s \geq 0$,

$$\mathcal{R}_\psi u(\tau, \mathbf{a}, b) := \int_{\mathbb{R}^{d+1}} u(t, \mathbf{x}) \psi(\tau t + \mathbf{a} \cdot \mathbf{x} - b) \|(\tau, \mathbf{a})\|^s \, dt d\mathbf{x}. \tag{29}$$

Note that the dual ridgelet transform $R_\eta^\dagger T$ of $T : \mathbb{R}^{d+2} \to \mathbb{R}$ with respect to $\eta : \mathbb{R} \to \mathbb{R}$ is defined as

$$\mathcal{R}_\eta^\dagger T(t, \mathbf{x}) := \int_{\mathbb{R}^{d+2}} T(\tau, \mathbf{a}, b) \eta(\tau t + \mathbf{a} \cdot \mathbf{x} - b) \|(\tau, \mathbf{a})\|^{-s} d\tau d\mathbf{a} db \tag{30}$$

We will take $s = 0$ in the above definitions. Recall that $\sigma \in \mathcal{S}_0'(\mathbb{R})$ is a fixed activation. Theorem 5.6 of [59] says that if we can find a function ψ so that $K_{\psi,\sigma} \in (0, \infty)$ then the reconstruction formula $u(t, \mathbf{x}) = \frac{1}{K_{\psi,\sigma}} R_\sigma^\dagger R_\psi u(t, \mathbf{x})$ holds. Therefore, to finish the proof we need to find ψ so that $K_{\psi,\sigma} = 1$ and $|\widehat{\psi}(\omega)| \leq C|\omega|^m$ for $|\omega| < 1$ and some $C > 0$ (condition (8)). We will make use of Corollary 5.5 of [59], which says that if $\zeta^\beta \widehat{\sigma}(\zeta) \in C(-\delta, \delta)$ for some $\delta > 0, \beta \in \mathbb{N}$ and $\psi_0 \in \mathcal{S}(\mathbb{R})$ is such that

$$J_\sigma := \int_{\mathbb{R}} \zeta^\beta \overline{\widehat{\psi_0}(\zeta)} \widehat{\sigma}(\zeta) \, d\zeta \neq 0, \tag{31}$$

then

$$\psi = \Lambda^{d+1} \psi_0^{(\beta)}$$

is admissible with σ, where for $m \in \mathbb{N}$, Λ^m is the backprojection filter that satisfies

$$\widehat{\Lambda^m F}(\mathbf{u}, \omega) = i^m |\omega|^m \widehat{F}(\mathbf{u}, \omega). \tag{32}$$

We let $G(z) := \exp(-z^2/2)$ be the standardised gaussian and take

$$\psi_0(z) := \frac{d^{2n}}{dz^{2n}} G(z), \tag{33}$$

where n is an arbitrary positive integer. Then since we assume $\sigma \in \mathcal{T}_k$ (see Definition 3.1), we have $\zeta^\beta \widehat{\sigma}(\zeta) \in C(-\delta, \delta)$ for some $\delta > 0$ and

$$J_\sigma = \int_{\mathbb{R}} \zeta^\beta \widehat{G^{(2n)}}(\zeta) \, \widehat{\sigma}(\zeta) \, d\zeta = \sqrt{2\pi}(-1)^n \int_{\mathbb{R}} \zeta^{2n+\beta} G(\zeta) \, \widehat{\sigma}(\zeta) \, d\zeta \neq 0, \tag{34}$$

where we have used $\widehat{G}(\zeta) = \sqrt{2\pi} G(\zeta)$. It is important to observe that we can take $n \in \mathbb{Z}^+$ to be as large as we like in this construction, since this is required by the second point in Definition 3.1 (of $\mathcal{T}_k$).

Thus, using Corollary 5.5 of [59] we can say that $\psi = \Lambda^{d+1} \psi_0^{(\beta)}$ and σ are jointly admissible. We will normalise ψ and therefore we can assume $K_{\psi,\sigma} = 1$. Applying the reconstruction formula (Theorem 5.6 of [59]) gives

$$\begin{aligned} u(t, \mathbf{x}) &= R_\sigma^\dagger R_\psi u(t, \mathbf{x}) \\ &= \int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} R_\psi u(\tau, \mathbf{a}, b) \sigma(\tau t + \mathbf{a} \cdot \mathbf{x} - b) db d\mathbf{a} d\tau. \end{aligned} \tag{35}$$

Lastly, we need to verify that $A_{\psi,m} < +\infty$. Using the property of the backfilter from (32), we have

$$\begin{aligned} \widehat{\psi}(\omega) &= i^{d+1} |\omega|^{d+1} \widehat{\psi_0'} = i^{d+2} \omega |\omega|^{d+1} \widehat{\psi_0} \\ &= \sqrt{2\pi} \, i^{d+2n+2} \omega |\omega|^{d+2n+1} G(\omega). \end{aligned} \tag{36}$$

Taking the absolute value gives $|\widehat{\psi}(\omega)| = \sqrt{2\pi} |\omega|^{d+2n+2} G(\omega)$. Then choosing n so that $d+2n+2 > m$ (e.g. $n > m - d - 2$), we have $|\widehat{\psi}(\omega)| \leq \sqrt{2\pi} |\omega|^m$ for $|\omega| < 1$. □

Remark A.1 (Proving admissibility of $\tanh, \cos$ and sigmoid). We show here that each of these activations belong to $\mathcal{T}_k$ for $k \geq 0$. Firstly, they each belong to $\mathcal{S}_0'(\mathbb{R})$; this is explicitly mentioned in Section 6.1 of [59]. In the case of $\sigma = \tanh$, we have $\widehat{\tanh}(\zeta) = -i\pi/\sinh(\pi\zeta/2)$ so $\zeta\widehat{\sigma}(\zeta)$ is continuous around 0, and

$$J_\sigma = \sqrt{2\pi}(-1)^{n+1} i\pi \int_{\mathbb{R}} \frac{\zeta^{2n+1} G(\zeta)}{\sinh(\frac{\pi\zeta}{2})}\, d\zeta \neq 0. \tag{37}$$

Furthermore, $\tanh$ and each of its derivatives are bounded uniformly. Therefore $\tanh \in \mathcal{T}_k$ for any $k \geq 0$. For $\sigma = \cos$, $\widehat{\cos}(\zeta) = \pi\left(\delta(\zeta - 1) + \delta(\zeta + 1)\right)$, which is continuous (in fact, zero) in a neighbourhood of 0, so we may take $\beta = 0$, and

$$J_\sigma = \sqrt{2\pi}(-1)^n \pi \left[G(1) + G(-1)\right] \neq 0. \tag{38}$$

Therefore it also belongs to $\mathcal{T}_k$ for all $k \geq 0$.

For $\sigma = \text{sigmoid}$, the argument is similar; $\widehat{\sigma} = -i\pi\text{csch}(\pi\zeta) + \pi\delta(\zeta)$, so $\zeta\widehat{\sigma}(\zeta)$ is continuous around the origin and $J_\sigma \neq 0$. Each of its derivatives are also uniformly bounded.

A.2 Sketching the proof of Lemma 3.3

For convenience, we start with a sketch of Lemma 3.3. The full proof is provided in A.3 below.

Proof (Outline). **Step 1: Plancherel in b.** We apply the Plancherel formula in b to get

$$\begin{aligned} I = &\int_{\mathbb{R}^{d+2}} |\widehat{\mathcal{R}_\psi u}(\tau, \mathbf{a}, \omega)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \, d\tau d\mathbf{a} d\omega \\ &+ \int_{\mathbb{R}^{d+2}} |\partial_\omega \widehat{\mathcal{R}_\psi u}(\tau, \mathbf{a}, \omega)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \, d\tau d\mathbf{a} d\omega \end{aligned} \tag{39}$$

Then we show using the definition of the Fourier Transform that $\widehat{R_\psi u}(\omega) = \widehat{u}(\tau\omega, \mathbf{a}\omega)\widehat{\psi}(-\omega)$, so I can be split up as

$$\begin{aligned} I \leq &\int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} |\widehat{u}(\tau\omega, \mathbf{a}\omega)\widehat{\psi}(-\omega)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \, d\omega d\mathbf{a} d\tau \\ &+ \int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} |\partial_\omega \left(\widehat{u}(\tau\omega, \mathbf{a}\omega)\widehat{\psi}(-\omega)\right)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \, d\omega d\mathbf{a} d\tau =: I_1 + I_2 \end{aligned} \tag{40}$$

The second term I_2 is the main obstacle in the proof.

Step 2: Change of variables. Performing a change of variables $s = \tau\omega, \xi = \mathbf{a}\omega$ allows us to estimate I_2 as

$$\begin{aligned} I_2 \leq\ & 2 \int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} |\widehat{u}(s, \xi) \partial_\omega \widehat{\psi}(-\omega)|^2 (1 + (\frac{s}{\omega})^2)^p (1 + (\frac{\|\xi\|}{\omega})^2)^q |\omega|^{-(d+1)} \, d\omega d\xi ds \\ &+ 2 \int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} |\frac{1}{\omega}(s\partial_s + \xi\nabla_\xi)\widehat{u}(s, \xi)\widehat{\psi}(-\omega)|^2 (1 + (\frac{s}{\omega})^2)^p (1 + (\frac{\|\xi\|}{\omega})^2)^q |\omega|^{-(d+1)} \, d\omega d\xi ds \\ &=: I_{2A} + I_{2B} \end{aligned} \tag{41}$$

Step 3: Estimating I_{2A} and I_{2B}. Here we show that one can find M large enough (but still finite) so that with the corresponding ψ generated from Proposition 3.2, the integrals I_{2A}, I_{2B} are both bounded by constants depending on the L^2 norms of ψ. Most of the difficulty lies in I_{2B} due to the derivatives that appear. Nonetheless, using properties of the Fourier transform and Fubini to exchange the order of integration, I_{2B} can be estimated as

$$\begin{aligned} I_{2B} \leq\ & C_{\psi_3} \int_{\mathbb{R}^{d+1}} |\widehat{(-it)u}|^2 (s, \xi)(1 + |s|^2)^{p+1} (1 + \|\xi\|^2)^q \, d\xi ds \\ &+ C_{\psi_3} \int_{\mathbb{R}^{d+1}} |\widehat{(-i\mathbf{x})u}|^2 (s, \xi)(1 + |s|^2)^p (1 + \|\xi\|^2)^{q+1} \, d\xi ds, \end{aligned} \tag{42}$$

where

$$C_{\psi_3} = 4 \int_{\mathbb{R}} |\widehat{\psi}(-\omega)|^2 \max(1, |\omega|^{-(2p+2q)}) |\omega|^{-(d+3)} \, d\omega. \tag{43}$$

Then by definition of the space $H^p(\mathbb{R}; H^q(\mathbb{R}^d))$ and the compact support of u, we get

$$I_{2B} \leq C_{\psi_3} K_{T,R} (\|u\|^2_{H_t^{p+1} H_x^q} + \|u\|^2_{H_t^p H_x^{q+1}}). \tag{44}$$

Similar arguments bound I_1 and I_{2A} by constants C_{ψ_1} and C_{ψ_2} of the same form as C_{ψ_3} (see Appendix A.3). Our final job is to show that we can construct ψ so that each of $C_{\psi_1}, C_{\psi_2}, C_{\psi_3}$ can be suitably bounded. This is done for each constant by separating $|\omega| < 1$ and $|\omega| \geq 1$. On $|\omega| < 1$ we take advantage of Proposition 3.2 which allows us to choose ψ with enough vanishing moments at 0 to ensure finiteness. On $|\omega| \geq 1$ we use $\psi \in \mathcal{S}(\mathbb{R})$. This is the main idea. The full details of the proof are deferred to Appendix A.3. □

A.3 Proof of Lemma 3.3

Proof. **Step 1: Plancherel in b.** Firstly, let's note that using the Plancherel formula in b we have

$$\begin{aligned} \int_{\mathbb{R}} |\mathcal{R}_\psi u(\tau, \mathbf{a}, b)|^2 (1 + b^2) \, db &= \int_{\mathbb{R}} |\mathcal{R}_\psi u(\tau, \mathbf{a}, b)|^2 \, db + \int_{\mathbb{R}} |b \mathcal{R}_\psi u(\tau, \mathbf{a}, b)|^2 \, db \\ &= \int_{\mathbb{R}} |\widehat{\mathcal{R}_\psi u}(\tau, \mathbf{a}, \omega)|^2 \, d\omega + \int_{\mathbb{R}} |\partial_\omega \widehat{\mathcal{R}_\psi u}(\tau, \mathbf{a}, \omega)|^2 \, d\omega. \end{aligned} \tag{45}$$

Then

$$\begin{aligned} I = &\int_{\mathbb{R}^{d+2}} |\widehat{\mathcal{R}_\psi u}(\tau, \mathbf{a}, \omega)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \, d\tau d\mathbf{a} d\omega \\ &+ \int_{\mathbb{R}^{d+2}} |\partial_\omega \widehat{\mathcal{R}_\psi u}(\tau, \mathbf{a}, \omega)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \, d\tau d\mathbf{a} d\omega \end{aligned} \tag{46}$$

We now find an expression for $\widehat{\mathcal{R}_\psi u}$. Using the definition of the Fourier transform and the ridgelet transform,

$$\begin{aligned} \widehat{R_\psi u}(\omega) &= \int_{\mathbb{R}} e^{-i\omega b} \, \mathcal{R}_\psi u \, db \\ &= \int_{\mathbb{R}} e^{-i\omega b} \left(\int_{\mathbb{R}^d} \int_{\mathbb{R}} u(t, \mathbf{x}) \psi(\tau t + \mathbf{a} \cdot \mathbf{x} - b) \, dt dx \right) db \\ &= \int_{\mathbb{R}^d} \int_{\mathbb{R}} u(t, \mathbf{x}) \left(\int_{\mathbb{R}} e^{-i\omega b} \psi(\tau t + \mathbf{a} \cdot \mathbf{x} - b) \, db \right) \, dt dx, \end{aligned} \tag{47}$$

where we have also used Fubini to exchange the order of the integrals. This is valid since we assume u is compactly supported, and therefore $u\psi$ is integrable on $\mathbb{R}^{d+1}$. By a change of variables $p = \tau t + \mathbf{a} \cdot \mathbf{x} - b$ the inner integral is equivalent to $e^{-i\omega(\tau t + \mathbf{a} \cdot \mathbf{x})} \int_{\mathbb{R}} e^{i\omega p} \psi(p) \, dp = e^{-i\omega(\tau t + \mathbf{a} \cdot \mathbf{x})} \widehat{\psi}(-\omega)$. Therefore

$$\begin{aligned} \widehat{R_\psi u}(\omega) &= \widehat{\psi}(-\omega) \int_{\mathbb{R}^d} u(t, \mathbf{x}) e^{-i\omega(\tau t + \mathbf{a} \cdot \mathbf{x})} \, dt d\mathbf{x} \\ &= \widehat{u}(\tau\omega, \mathbf{a}\omega) \widehat{\psi}(-\omega). \end{aligned} \tag{48}$$

Inserting this into (46), we have

$$\begin{aligned} I = &\int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} |\widehat{u}(\tau\omega, \mathbf{a}\omega) \widehat{\psi}(-\omega)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \, d\omega d\tau d\mathbf{a} \\ &+ \int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} |\partial_\omega \left(\widehat{u}(\tau\omega, \mathbf{a}\omega) \widehat{\psi}(-\omega) \right)|^2 (1 + |\tau|^2)^p (1 + \|\mathbf{a}\|^2)^q \, d\omega d\tau d\mathbf{a} \\ =: &\, I_1 + I_2. \end{aligned} \tag{49}$$

Step 2: Change of variables. We now introduce the change of variables $s = \tau\omega, \xi = \mathbf{a}\omega$. The corresponding jacobian is $d\mathbf{a} d\tau = |\omega|^{-(d+1)} d\xi ds$ and the operator changes to

$$\partial_\omega|_{(\tau, \mathbf{a})} = \partial_\omega|_{(s,\xi)} + \frac{1}{\omega}(s\partial_s + \xi \nabla_\xi)|_{(s,\xi)} =: \partial_\omega|_{(s,\xi)} + \mathcal{D}. \tag{50}$$

Therefore, separating the ω terms using Fubini,

$$\begin{aligned} I_1 &= \int_{\mathbb{R}^{d+2}} |\widehat{u}(s,\xi)|^2 |\widehat{\psi}(-\omega)|^2 (1+|\frac{s}{\omega}|^2)^p (1+(\frac{\|\xi\|}{\omega})^2)^q |\omega|^{-(d+1)}\, d\omega ds d\xi \\ &\leq C_{\psi_1} \int_{\mathbb{R}^{d+1}} |\widehat{u}(s,\xi)|^2 (1+|s|^2)^p (1+\|\xi\|^2)^q\, ds d\xi, \end{aligned} \tag{51}$$

where

$$C_{\psi_1} = \int_{\mathbb{R}} |\widehat{\psi}(-\omega)|^2 \max(1,|\omega|^{-2p}) \max(1,|\omega|^{-2q}) |\omega|^{-(d+1)}\, d\omega. \tag{52}$$

We will show the boundedness of this constant at the end of the proof. For I_2,

$$\begin{aligned} I_2 &\leq 2 \int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} |\widehat{u}(s,\xi) \partial_\omega \widehat{\psi}(-\omega)|^2 (1+(\frac{s}{\omega})^2)^p (1+(\frac{\|\xi\|}{\omega})^2)^q |\omega|^{-(d+1)}\, d\omega ds d\xi \\ &+ 2 \int_{\mathbb{R}} \int_{\mathbb{R}^d} \int_{\mathbb{R}} |\frac{1}{\omega}(s\partial_s + \xi \nabla_\xi) \widehat{u}(s,\xi) \widehat{\psi}(-\omega)|^2 (1+(\frac{s}{\omega})^2)^p (1+(\frac{\|\xi\|}{\omega})^2)^q |\omega|^{-(d+1)}\, d\omega ds d\xi \\ &=: I_{2A} + I_{2B} \end{aligned} \tag{53}$$

Step 3: Estimating I_{2A} and I_{2B}. First note that for any $\mathbf{x} \in \mathbb{R}^d$ and $k > 0$,

$$(1+\frac{\|\mathbf{x}\|^2}{|\omega|^2})^k \leq \max(1,|\omega|^{-2k})(1+\|\mathbf{x}\|^2)^k. \tag{54}$$

Therefore, for I_{2A} we can separate the ω dependence and write

$$I_{2A} \leq C_{\psi_2} \int_{\mathbb{R}^{d+1}} |\widehat{u}(s,\xi)|^2 (1+s^2)^p (1+\|\xi\|^2)^q ds d\xi, \tag{55}$$

where

$$C_{\psi_2} = 2 \int_{\mathbb{R}} |\partial_\omega \widehat{\psi}(-\omega)|^2 \max(1,|\omega|^{-(2p+2q)}) |\omega|^{-(d+1)}\, d\omega. \tag{56}$$

We will consider the boundedness of this constant at the end. For I_{2B}, we can estimate the (s,ξ) part as

$$\begin{aligned} \int_{\mathbb{R}^{d+1}} |\frac{1}{\omega}(s\partial_s + \xi\nabla_\xi)\widehat{u}(s,\xi)|^2\, ds d\xi &\leq 2|\omega|^{-2} \left(\int_{\mathbb{R}^{d+1}} |s\partial_s \widehat{u}(s,\xi)|^2 + \int_{\mathbb{R}^{d+1}} |\xi \nabla_\xi \widehat{u}(s,\xi)|^2 \right) \\ &= 2|\omega|^{-2} \left(\int_{\mathbb{R}^{d+1}} |\widehat{s(-it)u}(s,\xi)|^2 + \int_{\mathbb{R}^{d+1}} |\widehat{\xi(-i\mathbf{x})u}(s,\xi)|^2 \right) \\ &\leq 2|\omega|^{-2} \left(\int_{\mathbb{R}^{d+1}} |\widehat{(-it)u}(s,\xi)|^2 (1+s^2) + \int_{\mathbb{R}^{d+1}} |\widehat{(-i\mathbf{x})u}(s,\xi)|^2 (1+\|\xi\|^2) \right), \end{aligned} \tag{57}$$

where we used $s^2 \leq 1+s^2$ to obtain the final line. Thus we have

$$\begin{aligned} I_{2B} &\leq C_{\psi_3} \int_{\mathbb{R}^{d+1}} |\widehat{(-it)u}|^2 (s,\xi)(1+|s|^2)^{p+1} (1+\|\xi\|^2)^q\, d\xi ds \\ &+ C_{\psi_3} \int_{\mathbb{R}^{d+1}} |\widehat{(-i\mathbf{x})u}|^2 (s,\xi)(1+|s|^2)^p (1+\|\xi\|^2)^{q+1}\, d\xi ds, \end{aligned} \tag{58}$$

where

$$C_{\psi_3} = 4 \int_{\mathbb{R}} |\widehat{\psi}(-\omega)|^2 \max(1,|\omega|^{-(2p+2q)}) |\omega|^{-(d+3)}\, d\omega. \tag{59}$$

We will consider this constant at the end of the proof. Next, we normalise the norm of $H^p(\mathbb{R}; H^q(\mathbb{R}^d))$ on the Fourier side as

$$\int_{\mathbb{R}^{d+1}} |\widehat{u}(s,\xi)|^2 (1+|s|^2)^p (1+\|\xi\|^2)^q\, d\xi ds =: \|u\|^2_{H^p(\mathbb{R};H^q(\mathbb{R}^d))}, \tag{60}$$

and note that the factor $(2\pi)^{-1}$ from Plancherel in b compensates the factor 2π in $\|\widehat{\psi}\|_{L^2}^2 = 2\pi\|\psi\|_{L^2}^2$ and $\|\omega\widehat{\psi}\|_{L^2}^2 = 2\pi\|\psi'\|_{L^2}^2$.

This gives us (using the compact support of u)

$$\begin{aligned} I &\leq (C_{\psi_1} + C_{\psi_2})\|u\|_{H_t^p H_x^q}^2 + C_{\psi_3}(\||t|u\|_{H_t^{p+1} H_x^q}^2 + \||\mathbf{x}|u\|_{H_t^p H_x^{q+1}}^2) \\ &\leq (C_{\psi_1} + C_{\psi_2})\|u\|_{H_t^p H_x^q}^2 + C_{\psi_3} K_{T,R}\|u\|_{H_t^{p+1} H_x^{q+1}}^2 \\ &\leq (C_{\psi_1} + C_{\psi_2} + C_{\psi_3} K_{T,R})\|u\|_{H_t^{p+1} H_x^{q+1}}^2. \end{aligned} \tag{61}$$

Here $K_{T,R} := (2T + \lceil p \rceil + 1)^2 + (2R\sqrt{d} + \lceil q \rceil + 1)^2$, since $|t| \leq 2T$ and $\|\mathbf{x}\| \leq 2R\sqrt{d}$ on the support of u, and each derivative falling on t or $\mathbf{x}$ produces a lower-order term. For non-integer p or q, the bound follows by interpolation between the neighbouring integer orders. It remains to bound the constants C_{ψ_1}, C_{ψ_2} and C_{ψ_3}. Starting with the most recent C_{ψ_3}, we consider $|\omega| < 1$ and $|\omega| \geq 1$ separately. On $|\omega| < 1$, we have

$$C_{\psi_3} = 4\int_{-1}^{1} |\widehat{\psi}(-\omega)|^2 \max(1, |\omega|^{-(2p+2q)})|\omega|^{-(d+3)}\, d\omega \leq 4\int_{-1}^{1} |\widehat{\psi}(-\omega)|^2 |\omega|^{-(2p+2q+d+3)}\, d\omega,$$

so letting $M = (2p + 2q + d + 3)/2$ and choosing ψ as per the construction of Proposition 3.2 (see (36)), we have

$$|\widehat{\psi}(\omega)| \leq \sqrt{2\pi}|\omega|^M \text{ for } |\omega| < 1. \tag{62}$$

This implies that the integral is bounded by $\int_{-1}^{1} 2\pi d\omega = 4\pi$. On $|\omega| \geq 1$ we can bound it by $\|\psi\|_{L^2(\mathbb{R})}^2$ (by Plancherel), and so $C_{\psi_3} \leq 16\pi + 4\|\psi\|_{L^2(\mathbb{R})}^2$. We can apply the same estimate to C_{ψ_1} to get $C_{\psi_1} \leq 4\pi + \|\psi\|_{L^2(\mathbb{R})}^2$. For C_{ψ_2}, recall from (36) that $\widehat{\psi}(\omega) = \sqrt{2\pi}\, i^m \omega|\omega|^{m-1} G(\omega)$, for m which we can choose arbitrarily large. A simple computation gives us on $|\omega| < 1$ that $|\partial_\omega \widehat{\psi}| \leq \sqrt{2\pi}(m+1)\omega^{m-1} G(\omega)$. Similarly, on $|\omega| \geq 1$ we get $|\partial_\omega \widehat{\psi}| \leq \sqrt{2\pi}(m+1)|\omega|^{m+1} G(\omega)$. Therefore on $|\omega| < 1$, by choosing the same $m = M$,

$$\begin{aligned} \int_{-1}^{1} |\partial_\omega \widehat{\psi}(-\omega)|^2 \max(1, |\omega|^{-(2p+2q)})|\omega|^{-(d+1)}\, d\omega &\leq 2\pi(M+1)^2 \int_{-1}^{1} |\omega|^2\, d\omega \\ &\leq 4\pi(M+1)^2. \end{aligned} \tag{63}$$

For $|\omega| > 1$, notice that $|\partial_\omega \widehat{\psi}| \leq (M+1)|\omega||\widehat{\psi}(\omega)|$ and so

$$\int_{\mathbb{R}\setminus B_1(0)} |\partial_\omega \widehat{\psi}(-\omega)|^2 \max(1, |\omega|^{-(2p+2q)})|\omega|^{-(d+1)}\, d\omega \leq (M+1)^2 \int_{\mathbb{R}\setminus B_1(0)} |\omega\widehat{\psi}(\omega)|^2\, d\omega. \tag{64}$$

and so $C_{\psi_2} \leq 2(M+1)^2 \left(4\pi + \|\psi'\|_{L^2(\mathbb{R})}^2\right)$. This leads to

$$I \leq \left[(4\pi + \|\psi\|_{L^2(\mathbb{R})}^2)(1 + 4K_{T,R}) + 2(M+1)^2 \left(4\pi + \|\psi'\|_{L^2(\mathbb{R})}^2\right)\right] \|u\|_{H_t^{p+1} H_x^{q+1}}^2, \tag{65}$$

where $M = (2p + 2q + d + 3)/2$. Here ψ is the unnormalised function of (36); for the normalised $\psi/K_{\psi,\sigma}$ used in Theorem 3.4, $\mathcal{L}_\psi$ is divided by $|K_{\psi,\sigma}|^2$.

□

A.4 Proof of Theorem 3.4

Proof. **Step 1: extension of u to $\mathbb{R} \times \mathbb{R}^d$.** In order to make use of the previous results, we extend u to the full space $\mathbb{R} \times \mathbb{R}^d$. Since D is Lipschitz, there exists a bounded extension operator $E_x : H^{q+s_2}(D) \to H^{q+s_2}(\mathbb{R}^d)$ (see Theorem 5 in Chapter VI of [60]). Applying E_x pointwise in time and combining with a standard one-dimensional extension in time, we obtain $\bar{u} \in H^{p+s_1}(\mathbb{R}; H^{q+s_2}(\mathbb{R}^d))$ such that $\bar{u} = u$ on $(0,T) \times D$ and $\|\bar{u}\|_{H^{p+s_1}(\mathbb{R};H^{q+s_2}(\mathbb{R}^d))} \leq C_1\|u\|_{H^{p+s_1}(0,T;H^{q+s_2}(D))}$. Let $\eta \in C_c^\infty(\mathbb{R})$ and $\chi \in C_c^\infty(\mathbb{R}^d)$ be cut-off functions equal to 1 on $[0,T]$ and D respectively. Defining $\tilde{u}(t,x) := \eta(t)\chi(x)\bar{u}(t,x)$ we obtain $\tilde{u} \in H_t^{p+s_1} H_x^{q+s_2}(\mathbb{R} \times \mathbb{R}^d)$ with compact support and

$\|\tilde{u}\|^2_{H_t^{p+s_1}H_x^{q+s_2}} \le C_\Omega \|u\|^2_{H_t^{p+s_1}H_x^{q+s_2}}$. Note that the constant C_Ω collects two contributions. The first is the extension constant C_1, which depends on p, q, d and the Lipschitz character of ∂D; it is not explicit in [60] and grows with the Lipschitz constant of ∂D and with p, q. The second is the cost of multiplication by $\eta\chi$, which by the Leibniz rule is controlled by $\|\eta\|_{C^p}\|\chi\|_{C^q}$ times a combinatorial factor depending on q and d. A narrower collar between D and $\operatorname{supp}\chi$ increases $\|\chi\|_{C^q}$, while a wider one increases R and hence $\mathcal{L}_\psi$ in (11), so the cut-off cannot be chosen to improve both. In all cases C_Ω is independent of N, so it affects the constant in (14) but not the rate; cases in which its dependence on d is explicit are discussed in Remark A.2. Proposition 3.2 applies to $\tilde{u}$: it is compactly supported, and $\widehat{\tilde{u}} \in L^1$ by Cauchy–Schwarz, since $\tilde{u} \in H_t^{s_1}H_x^{s_2}$ with $s_1 > 1/2$, $s_2 > d/2$. Moreover, by Cauchy–Schwarz and Lemma 3.3 the integral below converges absolutely, so the limit in the dual ridgelet transform of [59] can be dropped. Then from (7), $\tilde{u}$ can be represented as

$$\begin{aligned}\tilde{u}(t,x) &= \int_{\mathbb{R}}\int_{\mathbb{R}^d}\int_{\mathbb{R}} (\mathcal{R}_\psi \tilde{u})(\tau,\mathbf{a},b)\sigma(\tau t + \mathbf{a}\cdot\mathbf{x} - b)\, d\tau d\mathbf{a} db \\ &= \int_{\mathbb{R}}\int_{\mathbb{R}^d}\int_{\mathbb{R}} \frac{(\mathcal{R}_\psi \tilde{u})(\tau,\mathbf{a},b)}{\pi(\tau,\mathbf{a},b)}\sigma(\tau t + \mathbf{a}\cdot\mathbf{x} - b)\pi(\tau,\mathbf{a},b) d\tau d\mathbf{a} db,\end{aligned} \tag{66}$$

where $\pi : \mathbb{R}^{d+2} \to \mathbb{R}_+$ is the probability density function from (12).

Step 2: construction of the unbiased estimator u_N. Our neural network approximation of u will be denoted u_N, and we define it as an unbiased estimator of $\tilde{u}$, i.e.

$$u_N(t,\mathbf{x}) := \frac{1}{N}\sum_{i=1}^N \frac{R_\psi \tilde{u}(\tau_i,\mathbf{a}_i,b_i)}{\pi(\tau_i,\mathbf{a}_i,b_i)}\sigma(\tau_i t + \mathbf{a}_i\cdot\mathbf{x} - b_i) \equiv \frac{1}{N}\sum_{i=1}^N X_i(t,\mathbf{x}), \tag{67}$$

where $(\tau_i,\mathbf{a}_i,b_i) \sim \pi$. By construction, it is clear that $\mathbb{E}_\Theta(u_N) = \tilde{u}$. More generally, for any $0 \le \ell \le p$ and multi-index β with $|\beta| \le q$, we have

$$\mathbb{E}_\Theta(\partial_t^\ell D_\mathbf{x}^\beta(u_N)) = \partial_t^\ell D_\mathbf{x}^\beta(\tilde{u}). \tag{68}$$

In order to estimate $u - u_N$, we have

$$\begin{aligned}\mathbb{E}_\Theta\left(\|\partial_t^\ell D_\mathbf{x}^\beta(\tilde{u} - u_N)\|^2_{L^2((0,T)\times D)}\right) &= \mathbb{E}_\Theta \int_{(0,T)\times D} |\partial_t^\ell D_\mathbf{x}^\beta(\tilde{u} - u_N)|^2\, dxdt \\ &= \int_{(0,T)\times D} \mathrm{Var}_\Theta\left(\partial_t^\ell D_\mathbf{x}^\beta(\tilde{u} - u_N)\right)\, dxdt.\end{aligned} \tag{69}$$

Let $Y_i(t,\mathbf{x}) := X_i(t,\mathbf{x}) - \mathbb{E}_\Theta(X_1(t,\mathbf{x}))$. Then $\mathbb{E}(Y_i) = 0$ and $\mathrm{Var}(Y_i) = \mathrm{Var}(X_i)$. Moreover, we have $u - u_N = \frac{1}{N}\sum_{i=1}^N Y_i(t,\mathbf{x})$ and therefore $\mathrm{Var}(u - u_N) = \frac{1}{N}\mathrm{Var}(Y_1(t,\mathbf{x})) = \frac{1}{N}\mathrm{Var}(X_1(t,\mathbf{x}))$. The same argument works if we replace $u - u_N$ with $\partial_t^\ell D_\mathbf{x}^\beta(u - u_N)$. As a result, we have (since $u = \tilde{u}$ on $(0,T)\times D$))

$$\begin{aligned}\mathbb{E}_\Theta\left(\|\partial_t^\ell D_\mathbf{x}^\beta(u - u_N)\|^2_{L^2((0,T)\times D)}\right) &= \mathbb{E}_\Theta\left(\|\partial_t^\ell D_\mathbf{x}^\beta(\tilde{u} - u_N)\|^2_{L^2((0,T)\times D)}\right) \\ &\le \frac{1}{N}\int_{(0,T)\times D} \mathbb{E}_\Theta|\partial_t^\ell D_\mathbf{x}^\beta X_i|^2\, dxdt.\end{aligned} \tag{70}$$

Step 3: bounding $\mathbb{E}_\Theta(|\partial_t^\ell D_\mathbf{x}^\beta X_i|^2)$. Let $\ell = 0, \beta = 0$ to begin with. A direct computation gives

$$\begin{aligned}&\mathbb{E}_\Theta(|X_i(t,\mathbf{x})|^2) \\ &= C_\pi \int_{\mathbb{R}^{d+2}} |R_\psi \tilde{u}(\tau,\mathbf{a},b)|^2 |\sigma(\tau t + \mathbf{a}\mathbf{x} - b)|^2 (1+\tau^2)^{\lambda_\tau}(1+\|\mathbf{a}\|^2)^{\lambda_a}(1+b^2)\, d\tau d\mathbf{a} db \\ &\le C_\pi \|\sigma\|^2_\infty \mathcal{L}_\psi \|\tilde{u}\|^2_{H^{s_1}(\mathbb{R};H^{s_2}(\mathbb{R}^d))},\end{aligned} \tag{71}$$

where, since $\lambda_\tau > 1/2$ and $\lambda_a > d/2$, we have $s_1 > 3/2, s_2 > (d+2)/2$ using Lemma 3.3. Therefore, we have

$$\mathbb{E}_\Theta\left(\|u - u_N\|^2_{L^2((0,T)\times D)}\right) \le \frac{T|D|\|\sigma\|^2_\infty \mathcal{L}_\psi}{N}\|\tilde{u}\|^2_{H^{s_1}(\mathbb{R};H^{s_2}(\mathbb{R}^d))}. \tag{72}$$

The same argument works if we take a general $0 \le \ell \le p$ and a non-zero multi-index $\beta = (\beta_1, ..., \beta_d)$ with $|\beta| \le q$. In this case we have

$$\begin{aligned}
&\mathbb{E}_\Theta(|\partial_t^\ell D_{\mathbf{x}}^\beta X_i(t,\mathbf{x})|^2) \\
&\quad \le C_\pi \int_{\mathbb{R}^{d+2}} |R_\psi \tilde{u}(\tau,\mathbf{a},b)|^2 |\sigma^{(p+q)}|^2 |\tau|^{2p} \|\mathbf{a}\|^{2q} (1+\tau^2)^{\lambda_\tau} (1+\|\mathbf{a}\|^2)^{\lambda_a} (1+b^2)\, d\tau d\mathbf{a} db \\
&\quad \le C_\pi \int_{\mathbb{R}^{d+2}} |R_\psi \tilde{u}(\tau,\mathbf{a},b)|^2 |\sigma^{(p+q)}|^2 (1+\tau^2)^{p+\lambda_\tau} (1+\|\mathbf{a}\|^2)^{q+\lambda_a} (1+b^2)\, d\tau d\mathbf{a} db.
\end{aligned} \tag{73}$$

Then using (10), we get

$$\mathbb{E}_\Theta(|\partial_t^\ell D_{\mathbf{x}}^\beta X_i(t,\mathbf{x})|^2) \le C_\pi \|\sigma^{(p+q)}\|_\infty^2 \mathcal{L}_\psi \|\tilde{u}\|^2_{H^{p+s_1}(\mathbb{R}; H^{q+s_2}(\mathbb{R}^d))}. \tag{74}$$

By (70) this implies (using $\|\tilde{u}\|^2_{H_t^{p+s_1} H_x^{q+s_2}} \le C_\Omega \|u\|^2_{H_t^{p+s_1} H_x^{q+s_2}}$)

$$\mathbb{E}_\Theta \left(\|\partial_t^\ell D_{\mathbf{x}}^\beta (u - u_N)\|^2_{L^2((0,T)\times D)} \right) \le \frac{C_\Omega T |D| C_\pi \|\sigma^{(p+q)}\|_\infty^2 \mathcal{L}_\psi}{N} \|u\|^2_{H^{p+s_1}(0,T; H^{q+s_2}(D))}, \tag{75}$$

where again $s_1 > 3/2$ and $s_2 > (d+2)/2$. Summing up this estimate over the $(p+1)\binom{q+d}{d}$ derivatives $\partial_t^\ell D_{\mathbf{x}}^\beta$ with $\ell \le p$, $|\beta| \le q$ leads to the claimed result. □

Remark A.2 (Growth factor for the leading coefficient). We can give a more precise description of the constant in the right-hand side of (14) for the case where $\sigma \in \mathcal{T}_k$ (e.g. $\tanh$; see Definition 3.1), $|D| \le 1$, and u is already defined on $\mathbb{R} \times \mathbb{R}^d$ with support in $[-2T, 2T] \times [-2R, 2R]^d$. In this case no extension is needed, so $C_\Omega = 1$. In such a case, we can show that if we let $\lambda_a = \frac{d}{2} + \alpha$ for any fixed $\alpha > 0$, then the coefficient in the right-hand side of (14) will grow at most polynomially in d. We will denote by C a positive constant independent of dimension and by $\Gamma(\cdot)$ the Gamma function. Then note that

$$C_\pi \le C \int_{\mathbb{R}^d} (1 + \|\mathbf{a}\|^2)^{-\lambda_a}\, d\mathbf{a}. \tag{76}$$

Going to spherical coordinates and letting $\lambda_a = \frac{d}{2} + \alpha$, we find

$$\int_{\mathbb{R}^d} (1+\|\mathbf{a}\|^2)^{-\lambda_a}\, d\mathbf{a} = \frac{2\pi^{d/2}}{\Gamma(d/2)} \int_0^\infty r^{d-1} (1+r^2)^{-\lambda_a}\, dr = \pi^{d/2} \frac{\Gamma(\alpha)}{\Gamma(d/2+\alpha)}. \tag{77}$$

Now, to bound $\mathcal{L}_\psi$, we go back to the proof of Lemma 3.3, which shows that (10) holds with $\mathcal{L}_\psi$ replaced by $\mathcal{L}'_\psi := C_{\psi_1} + C_{\psi_2} + K_{T,R}\, C_{\psi_3}$. The expression (11) is obtained from $\mathcal{L}'_\psi$ by bounding $|\omega|^{-(d+1)} \le 1$ on $|\omega| \ge 1$, which is too crude to track d, so we bound $\mathcal{L}'_\psi$ directly. In the proof of Theorem 3.4, Lemma 3.3 is applied with the exponents $p + \lambda_\tau$ and $q + \lambda_a$ in place of p and q, so the weight in $C_{\psi_1}, C_{\psi_2}, C_{\psi_3}$ is $\max(1, |\omega|^{-2Q})$ with

$$Q := p + q + \lambda_\tau + \lambda_a = \frac{d}{2} + c_0, \qquad c_0 := p + q + \lambda_\tau + \alpha.$$

We will use the construction of ψ from the proof of Proposition 3.2. Before normalisation, this gives $\bar{\psi}$ with $|\widehat{\bar{\psi}}(\omega)| = \sqrt{2\pi}\, |\omega|^\nu G(\omega)$ and $\nu := d + 1 + 2n + \beta$ (with $\beta = 1$ for $\tanh$ and $\beta = 0$ for $\cos$). Since $|\widehat{\bar{\psi}}(\zeta)||\zeta|^{-(d+1)} = \sqrt{2\pi}\, |\zeta|^{2n+\beta} G(\zeta)$, we have $K_{\bar{\psi},\sigma} = (2\pi)^d c_\sigma$ with $c_\sigma \ne 0$ independent of d. The normalised function $\psi = \bar{\psi}/K_{\bar{\psi},\sigma}$ therefore satisfies

$$\begin{gathered}
|\widehat{\psi}(\omega)| = \kappa |\omega|^\nu G(\omega), \qquad \kappa := \frac{\sqrt{2\pi}}{|K_{\bar{\psi},\sigma}|} \le C(2\pi)^{-d}, \\
|\partial_\omega \widehat{\psi}(\omega)| \le \kappa \left(\nu |\omega|^{\nu-1} + |\omega|^{\nu+1} \right) G(\omega).
\end{gathered} \tag{78}$$

On $|\omega| < 1$ the weight is $|\omega|^{-2Q}$, and by (78) the integrands of $C_{\psi_1}, C_{\psi_2}, C_{\psi_3}$ are at most $C\kappa^2(\nu+1)^2 |\omega|^e$, where

$$e = 2\nu - 2Q - (d+3) = 4n + 2\beta - 2c_0 - 1.$$

The dimension cancels, so we can fix n independently of d such that $e \geq 0$, and the contribution of $|\omega| < 1$ is at most $C\kappa^2(\nu+1)^2$. On $|\omega| \geq 1$ the weight is 1, and we keep the factor $|\omega|^{-(d+1)}$:

$$\begin{aligned}\int_{|\omega|\geq 1} |\widehat{\psi}(\omega)|^2 |\omega|^{-(d+1)}\, d\omega &\leq \kappa^2 \int_{|\omega|>1} |\omega|^{2\nu-d-1} e^{-\omega^2}\, d\omega \\ &= \kappa^2 \int_1^\infty u^{\nu-d/2-1} e^{-u}\, du \\ &\leq \kappa^2\, \Gamma(\nu - d/2),\end{aligned} \tag{79}$$

where we have used the substitution $u = \omega^2$. For $\partial_\omega \widehat{\psi}$, we can repeat the same process with (78) to find

$$\int_{|\omega|\geq 1} |\partial_\omega \widehat{\psi}(\omega)|^2 |\omega|^{-(d+1)}\, d\omega \leq 2\kappa^2(\nu^2+1)\, \Gamma(\nu - d/2 + 1), \tag{80}$$

and the weight $|\omega|^{-(d+3)}$ in C_{ψ_3} is smaller than $|\omega|^{-(d+1)}$. Since $\nu - d/2 + 1 = \frac{d}{2} + a$ with $a := 2 + 2n + \beta$, and $K_{T,R} \leq C(d+1)^2$ (here $\lceil q \rceil$ in $K_{T,R}$ becomes $\lceil q + \lambda_a \rceil \leq q + \lambda_a + 1$), we obtain

$$\mathcal{L}'_\psi \leq C(d+1)^2(\nu^2+1)\, \kappa^2\, \Gamma\left(\tfrac{d}{2} + a\right) \leq C(d+1)^4 (2\pi)^{-2d}\, \Gamma\left(\tfrac{d}{2} + a\right).$$

Now we consider the product $C_\pi \mathcal{L}'_\psi$. Notice that the leading term is given by

$$F(d) := \frac{\pi^{d/2}}{(2\pi)^{2d}} \frac{\Gamma(\frac{d}{2} + a)}{\Gamma(\frac{d}{2} + \alpha)}. \tag{81}$$

We now prove that $F(d)$ grows at most polynomially in d. The first factor equals $\left(\sqrt{\pi}/(4\pi^2)\right)^d < 1$. For the second, we use the inequality

$$\frac{\Gamma(x+a)}{\Gamma(x+b)} \leq (x+a)^{a-b}, \qquad a \geq b, \quad x + b > 0, \tag{82}$$

which follows from $\Gamma(y+1) = y\Gamma(y)$ applied to the integer part of $a - b$ and the log-convexity of Γ for the fractional part. Since $e \geq 0$ implies $a \geq \alpha$, (82) with $x = d/2$ and $b = \alpha$ gives $F(d) \leq (\frac{d}{2} + a)^{a-\alpha}$, and hence $C_\pi \mathcal{L}'_\psi \leq C(d+1)^{4+a-\alpha}$. Therefore, for any fixed $\alpha > 0$, the bound (14) holds with a constant that grows at most polynomially in d, since $\|\sigma^{(p+q)}\|_\infty$ does not depend on d and $T|D|(p+1)\binom{q+d}{d} \leq T(p+1)(d+1)^q$.

Remark A.3 (Scope of the regularity assumptions). Theorem 3.4 requires $u \in H_t^{p+s_1} H_x^{q+s_2}$ with $s_1 > 3/2$ and $s_2 > (d+2)/2$, so the class of admissible targets tightens as the spatial dimension grows. For the equations treated in Section 4 this regularity is supplied by the equations themselves rather than assumed: the porous medium equation regularises instantaneously away from the free boundary under (17), and one-dimensional compressible Navier–Stokes with H^k data, $k \geq 9$, yields (24). The assumptions are therefore natural for parabolic problems with smooth data, and our requirement is weaker than that of the closest comparable result [21, Thm 3.9], which needs $s \geq (d+9)/2$ and provides only fixed-time estimates. They nonetheless exclude several regimes of practical interest: solutions near a free boundary, where the profile loses differentiability (this is the case for the full-domain Barenblatt experiment of Appendix B.1, as noted in Remark B.1); shocks and low-regularity weak solutions; and turbulent regimes for Navier–Stokes. Within our proof strategy the requirement cannot be relaxed without either shrinking the admissible class or changing the integral representation, since the correspondence between the polynomially weighted integral of Lemma 3.3 and the Sobolev norm is what controls the variance of the estimator. Two directions seem worth pursuing. The first is to trade Sobolev regularity against decay conditions on $\hat{u}$, as in [46], although such information is rarely available for solutions of non-linear PDEs. The second, which we regard as more promising, is to note that $s_2 > (d+2)/2$ enters through the normalisability of $(1 + \|\mathbf{a}\|^2)^{-\lambda_a}$ in (12), and hence through the sampling measure rather than the ridgelet representation itself; replacing the isotropic weight by one adapted to the directions in which u is rough may permit the hypothesis to be stated on an anisotropic or Besov scale, rather than uniformly across all d spatial directions.

A.5 Proof of Corollary 4.1

Proof. The bound of (19) follows from an application of Theorem 3.4, so we now focus on bounding the PDE residual. To simplify notation, in this proof we write $(0,T)$ for the interval (t_0,T), on which u is smooth. We take $m=2$ for simplicity, but the computation is easily generalised to any $m \geq 1$. Using $\partial_t u - \Delta(u^2) = 0$, we have

$$\begin{aligned}
\mathcal{J}_{\text{PDE}}(u_N) &= \int_{(0,T)\times D} (\partial_t u_N - \Delta(u_N^2))^2 \, dtdx \\
&= \int_{(0,T)\times D} (\partial_t (u_N - u) - \Delta(u_N^2 - u^2))^2 \, dtdx \\
&\leq 2\|u_N - u\|^2_{H_t^1 L_x^2} + 2\int_{(0,T)\times D} |\Delta(u_N^2 - u^2)|^2 \, dtdx.
\end{aligned} \tag{83}$$

Now, Markov's inequality states that $\mathbb{P}(X > \eta) \leq \mathbb{E}[X]\frac{1}{\eta}$ for X non-negative, $\eta > 0$. In particular, for any $\delta \in (0,1)$, from the bound in Theorem 3.4 we obtain with the choices $X = \|u - u_N\|^2_{H^p(0,T;H^q(D))}$ and $\eta = \frac{1}{\delta}\frac{C_\Omega C_\pi \|\sigma^{(p+q)}\|_\infty^2 T|D|(p+1)\binom{q+d}{d}\mathcal{L}_\psi}{N} \cdot \|u\|^2_{H^{p+s_1}(0,T;H^{q+s_2}(D))}$ that

$$\mathbb{P}(X > \eta) \leq \mathbb{E}_\Theta\left(\|u - u_N\|^2_{H^p(0,T;H^q(D))}\right)\frac{1}{\eta} \leq \delta. \tag{84}$$

This implies that $\mathbb{P}(X \leq \eta) \geq 1-\delta$. That is, with probability at least $1-\delta$ it holds that $X \leq \eta = \frac{1}{\delta}\frac{C_\Omega C_\pi \|\sigma^{(p+q)}\|_\infty^2 T|D|(p+1)\binom{q+d}{d}\mathcal{L}_\psi}{N} \cdot \|u\|^2_{H^{p+s_1}(0,T;H^{q+s_2}(D))}$. We apply this once, with $X = \|u_N - u\|^2_{H_t^1 H_x^2}$ (i.e. $p=1$, $q=2$), which dominates both $\|u_N - u\|^2_{H_t^1 L_x^2}$ and $\|u_N - u\|^2_{L_t^2 H_x^2}$. Then we have with probability $1-\delta$ that

$$\|u_N - u\|^2_{H_t^1 L_x^2} \leq \frac{\mathcal{M}_\psi}{N\delta}\|u\|^2_{H_t^{1+s_1} H_x^{2+s_2}}, \tag{85}$$

where $s_1 > 3/2, s_2 > (d+2)/2$. Now we note that $\Delta(u_N^2 - u^2) = (u_N + u)\Delta e_N + 2\nabla(u_N + u)\nabla e_N + e_N\Delta(u_N + u)$, where $e_N := u_N - u$. Combining this with $(a+b)^2 \leq 2(a^2+b^2)$ and Holder's inequality,

$$\begin{aligned}
\int_{(0,T)\times D} |\Delta(u_N^2 - u^2)|^2 \, dtdx \leq{}& 8\|u_N + u\|^2_{L^\infty_{t,x}}\|\Delta(u_N - u)\|^2_{L^2_{t,x}} \\
&+ 8\|\nabla(u_N + u)\|^2_{L^\infty_{t,x}}\|\nabla(u_N - u)\|^2_{L^2_{t,x}} \\
&+ 8\|\Delta(u_N + u)\|^2_{L^\infty_{t,x}}\|u_N - u\|^2_{L^2_{t,x}}.
\end{aligned} \tag{86}$$

We also have

$$\|u\|_{L^\infty_{t,x}} + \|\nabla u\|_{L^\infty_{t,x}} + \|\Delta u\|_{L^\infty_{t,x}} \leq C_{emb}\|u\|_{L_t^\infty H_x^{2+k}}, \tag{87}$$

for $k > d/2$, where C_{emb} is the constant arising from the Sobolev embedding $H_x^k \hookrightarrow L_x^\infty$. Then on the same event, letting $L := \|u_N\|_{L^\infty_{t,x}} + \|\nabla u_N\|_{L^\infty_{t,x}} + \|\Delta u_N\|_{L^\infty_{t,x}}$, we have with probability $1-\delta$ that

$$\begin{aligned}
\int_{(0,T)\times D} |\Delta(u_N^2 - u^2)|^2 \, dtdx &\leq 16(L^2 + C_{emb}^2\|u\|^2_{L_t^\infty H_x^{2+k}})\|u_N - u\|^2_{L_t^2 H_x^2} \\
&\leq \frac{16(L^2 + C_{emb}^2\|u\|^2_{L_t^\infty H_x^{2+k}})\mathcal{M}_\psi}{N\delta}\|u\|^2_{H_t^{1+s_1} H_x^{2+s_2}}.
\end{aligned} \tag{88}$$

All in all, we get

$$\mathcal{J}_{\text{PDE}}(u_N) \leq \frac{(32(L^2 + C_{emb}^2\|u\|^2_{L_t^\infty H_x^{2+k}}) + 2)\mathcal{M}_\psi}{N\delta}\|u\|^2_{H_t^{1+s_1} H_x^{2+s_2}}, \tag{89}$$

with probability $1-\delta$. Taking $s_1 = 2$ and $s_2 = k+1$ with $k > d/2$, so that $s_2 > (d+2)/2$, we can estimate the leading norm by $\|u\|^2_{H_t^3 H_x^{3+k}}$. For the general case $m \geq 2$, the argument can be adapted

using a mean-value formula to re-write $u_N^m - u^m$ in terms of e_N. In this case, we get an extra constant in the coefficient of (89), C_m, which will be polynomial in m. This leads to the claimed result (20). In summary, taking initial data u_0 satisfying (17), we can find a RaNN which approximates the corresponding solution to the PDE such that the expected loss of the PDE residual is inversely proportional to the number of random features, and the rate is independent of dimension. □

A.6 Proof of Corollary 4.2

Proof. Since we are dealing with a system of equations, we need to illustrate how the result of Theorem 3.4 applies. Firstly, given a couple (ρ, u) belonging to class (24), the representation result of Proposition 3.2 tells us that we can find $\psi_\rho, \psi_u \in \mathcal{S}(\mathbb{R})$ such that we have the representations (assuming ρ, u have been smoothly extended to the full space):

$$\begin{aligned}\rho(t,x) &= \int_{\mathbb{R}}\int_{\mathbb{R}^d}\int_{\mathbb{R}}(A_\rho)(\tau,\mathbf{a},b)\sigma(\tau t+\mathbf{a}\cdot\mathbf{x}-b)\,d\tau d\mathbf{a}db,\\ u(t,x) &= \int_{\mathbb{R}}\int_{\mathbb{R}^d}\int_{\mathbb{R}}(A_u)(\tau,\mathbf{a},b)\sigma(\tau t+\mathbf{a}\cdot\mathbf{x}-b)\,d\tau d\mathbf{a}db,\end{aligned} \tag{90}$$

where $A_\rho := \mathcal{R}_{\psi_\rho}\rho$ and $A_u := \mathcal{R}_{\psi_u}u$. Then as in Theorem 3.4 we can introduce two unbiased estimators

$$\begin{aligned}\rho_N(t,x) &= \frac{1}{N}\sum_{i=1}^N \frac{A_\rho(\tau_i,\mathbf{a}_i,b_i)}{\pi(\tau_i,\mathbf{a}_i,b_i)}\sigma(\tau_i t+\mathbf{a}_i\cdot\mathbf{x}-b_i),\\ u_N(t,x) &= \frac{1}{N}\sum_{i=1}^N \frac{A_u(\tau_i,\mathbf{a}_i,b_i)}{\pi(\tau_i,\mathbf{a}_i,b_i)}\sigma(\tau_i t+\mathbf{a}_i\cdot\mathbf{x}-b_i).\end{aligned} \tag{91}$$

The result of Theorem 3.4 gives us the following bounds immediately:

$$\begin{aligned}\|\rho_N-\rho\|^2_{L^2_tH^q_x} &\le \frac{C_\Omega C_\pi\|\sigma^{(q)}\|^2_\infty T|D|(q+1)\mathcal{L}_\psi}{N}\cdot\|\rho\|^2_{H^{s_1}_tH^{q+s_2}_x},\\ \|u_N-u\|^2_{L^2_tH^q_x} &\le \frac{C_\Omega C_\pi\|\sigma^{(q)}\|^2_\infty T|D|(q+1)\mathcal{L}_\psi}{N}\cdot\|u\|^2_{H^{s_1}_tH^{q+s_2}_x},\end{aligned} \tag{92}$$

for $s_1, s_2 > 3/2$ since we are in dimension $d = 1$. From now on we label

$$\mathcal{M}_\psi := C_\Omega C_\pi\|\sigma^{(q+1)}\|^2_\infty\, 2T|D|(q+1)\,\mathcal{L}_\psi \tag{93}$$

The pair $v_N(t,x) := (\rho_N(t,x), u_N(t,x))$ can be interpreted as the outputs of a single random neural network of width $2N$, from which the first claim follows. We now bound the loss functions $\mathcal{J}^1_{PDE}$ and $\mathcal{J}^2_{PDE}$. For $\mathcal{J}^1_{PDE}$ we note the expression $\rho_N u_N - \rho u = \rho_N(u_N-u)+u(\rho_N-\rho)$ to get

$$\partial_x(\rho_N u_N-\rho u) = \partial_x\rho_N(u_N-u)+\rho_N\partial_x(u_N-u)+\partial_x u(\rho_N-\rho)+u\partial_x(\rho_N-\rho). \tag{94}$$

Therefore,

$$\begin{aligned}\mathcal{J}^1_{PDE}(\mathbf{v}_N) &= \int_{(0,T)\times D}|\partial_t(\rho_N-\rho)+\partial_x(\rho_N u_N-\rho u)|^2\\ &\le 2\|\partial_t(\rho_N-\rho)\|^2_{L^2_{t,x}}+8\|\partial_x\rho_N\|^2_{L^\infty_{t,x}}\|u_N-u\|^2_{L^2_{t,x}}+8\|\rho_N\|^2_{L^\infty_{t,x}}\|\partial_x(u_N-u)\|^2_{L^2_{t,x}}\\ &+8\|\partial_x u\|^2_{L^\infty_{t,x}}\|\rho_N-\rho\|^2_{L^2_{t,x}}+8\|u\|^2_{L^\infty_{t,x}}\|\partial_x(\rho_N-\rho)\|^2_{L^2_{t,x}}\\ &\le \Big[8(\|\partial_x\rho_N\|^2_{L^\infty_{t,x}}+\|\rho_N\|^2_{L^\infty_{t,x}}+\|\partial_x u\|^2_{L^\infty_{t,x}}+\|u\|^2_{L^\infty_{t,x}})+2\Big]\\ &\qquad\times(\|\rho_N-\rho\|^2_{H^1_{t,x}}+\|u_N-u\|^2_{H^1_{t,x}})\\ &=: \mathcal{B}_\mathcal{N}(\|\rho_N-\rho\|^2_{H^1_{t,x}}+\|u_N-u\|^2_{H^1_{t,x}}).\end{aligned} \tag{95}$$

Note that $\|\partial_x u\|_{L^\infty_{t,x}}$ is finite since $u \in C(0,T;H^9(D))$. An upper bound for the coefficient $\mathcal{B}_N$ is $8(\|\rho_N\|^2_{W^{1,\infty}_{t,x}}+\|u\|^2_{W^{1,\infty}_{t,x}})+2$. Applying Markov's inequality once, as in the proof of

Corollary 4.1, to $X := \|\rho_N - \rho\|^2_{H^1_t H^1_x} + \|u_N - u\|^2_{H^1_t H^2_x}$, we have with probability $1-\delta$ that $X \leq \frac{\mathcal{M}_\psi}{N\delta}\big(\|\rho\|^2_{H^{1+s_1}_t H^{1+s_2}_x} + \|u\|^2_{H^{1+s_1}_t H^{2+s_2}_x}\big)$. All estimates below hold on this event. In particular,

$$\mathcal{J}^1_{PDE}(\mathbf{v}_N) \leq \frac{(8(L+\|u\|^2_{W^{1,\infty}_{t,x}})+2)\mathcal{M}_\psi}{N\delta}(\|\rho\|^2_{H^{1+s_1}_t H^{1+s_2}_x} + \|u\|^2_{H^{1+s_1}_t H^{1+s_2}_x}), \tag{96}$$

where $s_1, s_2 > 3/2$,

$$L := \|\rho_N\|^2_{W^{1,\infty}_{t,x}} + \|u_N\|^2_{W^{1,\infty}_{t,x}} + \|\rho\|^2_{W^{1,\infty}_{t,x}} + \|u\|^2_{W^{1,\infty}_{t,x}}, \tag{97}$$

which is guaranteed to be finite since the neural network (ρ_N, u_N) is smooth. For the momentum equation, we use $\rho_N u_N^2 - \rho u^2 = \rho_N u_N(u_N - u) + u(\rho_N u_N - \rho u)$, and so

$$\begin{aligned}\partial_x(\rho_N u_N^2) - \partial_x(\rho u^2) = \partial_x(u_N\rho_N)(u_N - u) + u_N\rho_N\partial_x(u_N - u) + (\rho_N u_N - \rho u)\partial_x u \\ + u\partial_x(\rho_N u_N - \rho u).\end{aligned} \tag{98}$$

Therefore, in a similar way to the continuity equation, with probability $1-\delta$,

$$\begin{aligned}\mathcal{J}^2_{PDE}(\mathbf{v}_N) &= \int_{(0,T)\times D} |\partial_t(\rho_N u_N - \rho u) + \partial_x(\rho_N u_N^2 - \rho u^2) - \mu\partial_x^2(u_N - u)|^2 dxdt \\ &\leq \mathcal{C}_L(\|\rho_N - \rho\|^2_{H^1_{t,x}} + (2\mu+1)\|u_N - u\|^2_{H^1_t H^2_x}),\end{aligned} \tag{99}$$

where

$$\begin{aligned}\mathcal{C}_L = 32\|u_N\|^2_{W^{1,\infty}_{t,x}} + 64\|u\|^2_{W^{1,\infty}_{t,x}} + 80\|\rho\|^2_{W^{1,\infty}_{t,x}} + 16\|\rho_N\|^2_{W^{1,\infty}_{t,x}}\|u_N\|^2_{W^{1,\infty}_{t,x}} \\ + 64\|u\|^2_{W^{1,\infty}_{t,x}}\|u_N\|^2_{W^{1,\infty}_{t,x}}\end{aligned} \tag{100}$$

The assumption (97) ensures that $\mathcal{C}_L$ can be bounded as

$$C_L \leq (16L^2 + 112L + 2\mu) + 64(L+1)\|u\|^2_{W^{1,\infty}_{t,x}}. \tag{101}$$

Note that $\|\partial_t u\|_{L^\infty_{t,x}}$ is finite; this can be seen by rewriting the momentum equation as $\partial_t u = -u\partial_x u + \mu\rho^{-1}\partial_x^2 u$, and using the regularity $u \in C(0,T;H^k(D))$ for $k \geq 9$, along with the Sobolev embedding $H^1 \hookrightarrow L^\infty$. Then we have with probability $1-\delta$ that

$$\begin{aligned}&\mathcal{J}^2_{PDE}(\mathbf{v}_N) \leq \\ &\frac{\left[16L^2 + 112L + 2\mu + 64(L+1)\|u\|^2_{W^{1,\infty}_{t,x}}\right]\mathcal{M}_\psi}{N\delta}(\|\rho\|^2_{H^{1+s_1}_t H^{1+s_2}_x} + (2\mu+1)\|u\|^2_{H^{1+s_1}_t H^{2+s_2}_x}).\end{aligned}$$

Noting that $s_1, s_2 > 3/2$, we get

$$\begin{aligned}&(\mathcal{J}^1_{PDE} + \mathcal{J}^2_{PDE})(\mathbf{v}_N) \leq \\ &\frac{\left[16L^2 + 128L + 2\mu + 2 + 64(L+1)\|u\|^2_{W^{1,\infty}_{t,x}}\right]\mathcal{M}_\psi}{N\delta}(\|\rho\|^2_{H^3_t H^3_x} + (2\mu+1)\|u\|^2_{H^3_t H^4_x}),\end{aligned} \tag{102}$$

which is guaranteed to be finite if we take $(\rho_0, u_0) \in (H^k(D))^2$ for $k \geq 9$. This can be seen by noting that the continuity equation can be re-written as $\partial_t\rho = -u\partial_x\rho - \rho\partial_x u$, and the momentum equation as $\partial_t u = -u\partial_x u + \mu\rho^{-1}\partial_x^2 u$. Then using the regularity (24), one can bound the higher order Sobolev norms on the right-hand side of (102): each time derivative of u costs two spatial derivatives and each time derivative of ρ costs one, so for instance $\partial_t^3 u \in L^2(0,T;H^{k-5}(D))$, which requires $k \geq 9$ for $\|u\|_{H^3_t H^4_x}$ to be finite. □

B Experiments

In this section we give further details on the numerical experiments carried out in Section 5. All experiments were performed using a NVIDIA RTX 3080 GPU with 10GB VRAM. The code is available at `https://github.com/M6LI/RaNN-Expressivity-NonLinearPDEs`.

B.1 Porous Medium Equation

Recall that we consider the Barenblatt-Kompaneets-Zeldovich solution

$$u(t,x) = \frac{1}{t^\alpha}\left(b - \frac{m-1}{2m}\beta\frac{\|x\|^2}{t^{2\beta}}\right)_+^{\frac{1}{m-1}}, \tag{103}$$

where $\|\cdot\|$ is the ℓ^2 norm, $(\cdot)_+$ is the positive part and $\alpha = \frac{d}{d(m-1)+2}$ and $\beta = \frac{1}{d(m-1)+2}$. We set $m = 2$.

This solution is compactly supported but not differentiable at the edges of the support, which causes difficulty for numerical schemes. Since this solution is generated by a Dirac delta-valued initial data, we fix a small time $t_0 > 0$ and take the initial data to be $u_0(x) := u(t_0, x)$ and solve on the shifted time domain $(t_0, T + t_0)$.

We aim to investigate whether a convergence rate of $N^{-1/2}$ can be observed in practice. To this end, we train a RaNN to approximate the Barenblatt profile (28) in dimensions $d = 1, ..., 5$. In each dimension, we take a set of widths $N \in \{N_1, ..., N_k\}$, train the network for each width and plot the relative error of the final network against the true solution. The RaNN includes a Fourier feature layer, where frequencies ω_j are sampled from $\mathcal{N}(0, 10^2)$. For each dimension d and width N, we sample $M = 10N$ points (to ensure the problem remains well-posed) with a mixed strategy: 50% of the points are sampled uniformly and 50% are sampled uniformly on $[0.2, 0.8]^d$, which is a box focused on the support of the solution. Then we find weights $\mathbf{W} = \{W_i\}_{i=1}^N$ which minimise the Ridge regression loss

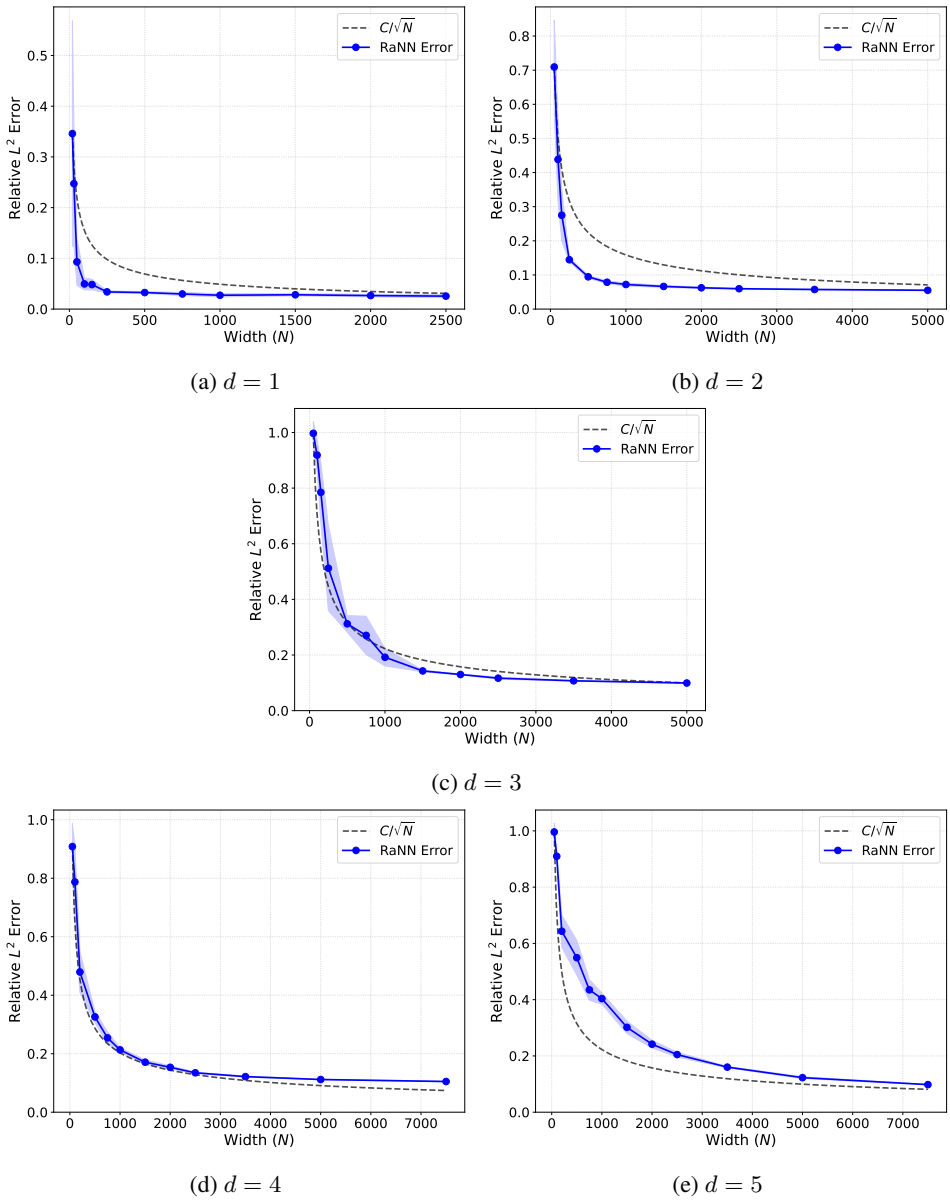
$$\mathcal{L}(\mathbf{W}) = \frac{1}{M}\sum_{i=1}^{M} \|\hat{\mathbf{u}}(t_i, \mathbf{x}_i) - \mathbf{y}_i\|_2^2 + \lambda\|\mathbf{W}\|_2^2, \qquad \lambda = 10^{-5}. \tag{104}$$

We compute the closed-form solution $\hat{\mathbf{W}}$ directly (using a Cholesky decomposition), and then evaluate the relative error. We train the RaNN five times for each width. The mean relative L^2 errors are plotted against the widths in Figure 3. The errors are also plotted on log-log scales in Figure 4. The key observation here is that the RaNN error points lie close to the $C/\sqrt{N}$ curve, which supports the upper bound of Theorem 3.4. Note that in $d = 1, 2, 3$ we take $t_0 = 0.1$ (i.e. the initial data is set to $u(t_0, x)$, where u is the Barenblatt profile from (28)) and in $d = 4, 5$ this is reduced to $t_0 = 0.01$ to keep the problem computationally manageable.

Remark B.1 (Smoothness of Barenblatt profile). Note that the lack of smoothness of the Barenblatt profile at the edges of the support places this experiment outside of the assumptions in Corollary 4.1. Nonetheless, if we restrict the solution to a compact subset strictly contained within the support, the result does apply. The full-domain experiment can therefore be viewed as a setting which slightly exceeds the strict theoretical assumptions.

Remark B.2 (Sampling distributions used in the experiments). In the experiments of Section 5 and Figures 3 and 4, the weights of the randomised neural networks are sampled from a Gaussian distribution rather than the distribution π (12) used in Theorem 3.4. This choice was made in order to test the robustness of the approximation rate beyond the specific (and somewhat unusual) π used in Theorem 3.4. Results for π and a similar Student-t distribution are reported in Table 1 and Figures 5 and 6. Empirically, we observe consistent $N^{-1/2}$ scaling behaviour which supports the relevance of our theory when sampling from a more practical Gaussian distribution. We note that the proof of Theorem 3.4 does rely on the heavy-tailed form of π: Lemma 3.3 controls a polynomially weighted integral of the ridgelet transform by a Sobolev norm of u, and for a Gaussian π this weight would become Gaussian, which can only be controlled if the Fourier transform of u decays faster than a Gaussian. Extending the result to Gaussian weights would therefore require a different integral representation of u, whereas an extension to other heavy-tailed distributions appears possible.

Heavy-tailed weight distributions. We have repeated the experiments above with the Gaussian law of the hidden weights replaced by two heavy-tailed laws. The first is the distribution π in (12) itself, with $\lambda_\tau = 3$ and $\lambda_a = (d + 5)/2$: the frequencies τ_i and $\mathbf{a}_i$ are drawn independently from a Student-t and a d-dimensional multivariate Student-t distribution with 5 degrees of freedom, and the biases b_i from a standard Cauchy distribution. Since $\sigma = \cos$ is 2π-periodic, only b_i modulo 2π



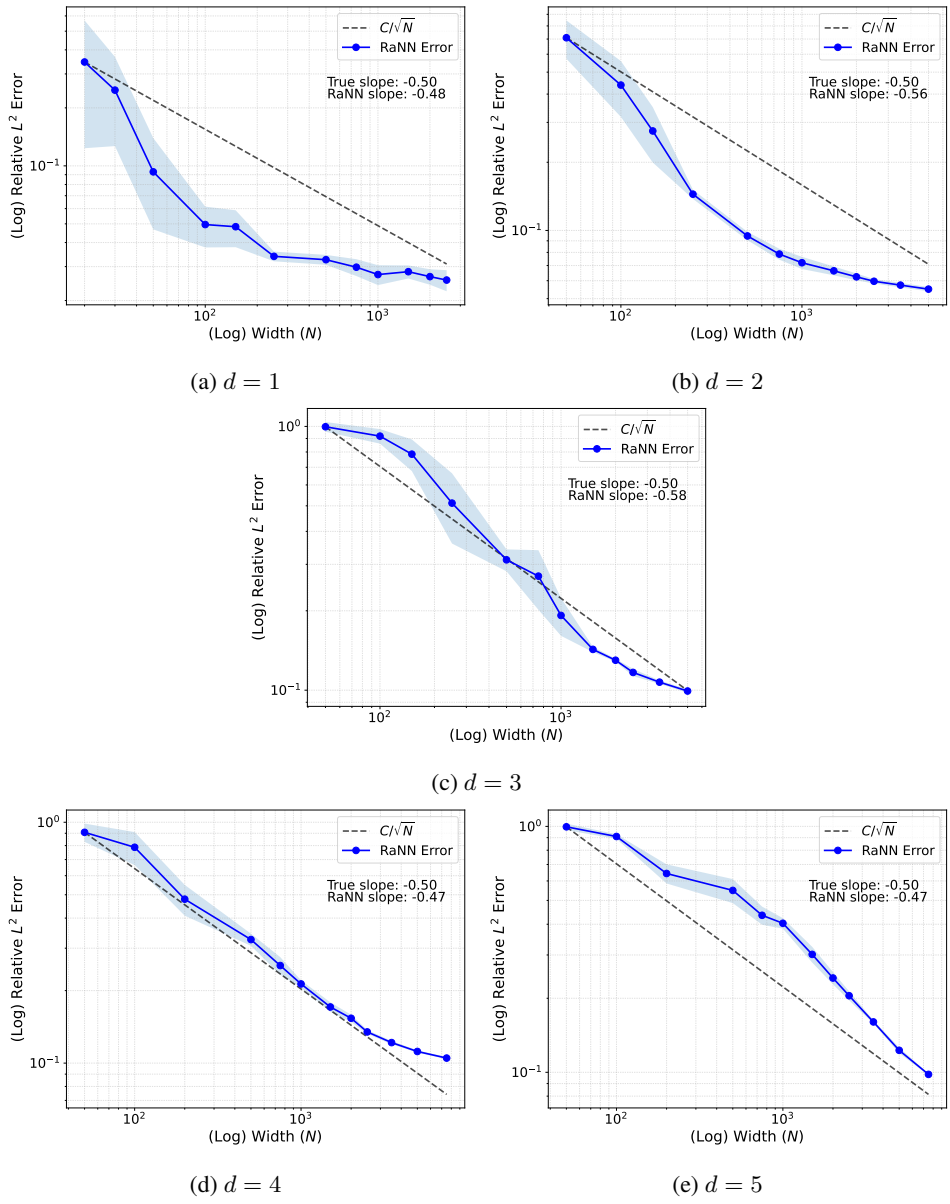
(a) $d = 1$ (b) $d = 2$

(c) $d = 3$

(d) $d = 4$ (e) $d = 5$

Figure 3: Approximation error of RaNNs of varying width for solving PMEs in dimensions $d = 1, \ldots, 5$. The shaded band indicates the region within one standard deviation of the mean relative L^2 error.

enters the network. The frequencies are applied to rescaled inputs $(s_t t, s\mathbf{x})$ for constants $s_t, s > 0$, which amounts to applying π on a rescaled domain and only affects the constant in (14). For the second law, $(\tau_i, \mathbf{a}_i)$ are drawn jointly from a $(d+1)$-dimensional multivariate Student-t distribution with 5 degrees of freedom, whose density is proportional to $(1 + \tau^2 + \|\mathbf{a}\|^2)^{-(d+6)/2}$ (with the same rescaling of the inputs), and the biases are drawn uniformly from $[0, 2\pi]$. Under this variant, τ_i and $\mathbf{a}_i$ have the same marginal distributions as under π but are not independent. The remaining set-up is as above, except that we use the ridge parameter $\lambda = 10^{-8}$, three independent draws per width, and the time shifts $t_0 = 6.8 \times 10^{-3}, 1.9 \times 10^{-2}, 2.8 \times 10^{-2}$ for $d = 1, 2, 3$, chosen such that the amplitude of the solution decays by the same factor 2.5 over the time window in each dimension, and $t_0 = 10^{-2}$ for $d = 4, 5$. Both laws use the same configuration and random seeds. The fitted slopes are reported in Table 1 and the error curves in Figures 5 and 6. The two laws give very similar results:



(a) $d = 1$ (b) $d = 2$

(c) $d = 3$

(d) $d = 4$ (e) $d = 5$

Figure 4: Log-log plot of the relative L^2 error versus the width N. The reference scaling $C/\sqrt{N}$ and the measured RaNN slope are shown. The shaded band indicates the region within one standard deviation of the mean relative L^2 error.

the slopes differ by at most 0.04 and the errors at the largest width by at most 7%. For both laws, four of the five 95% confidence intervals contain $-1/2$; at $d = 4$ the interval lies below $-1/2$, i.e. the error decays slightly faster than $N^{-1/2}$, which is consistent with Theorem 3.4 since it is an upper bound.

B.2 Training on the PDE residual loss

The experiments of Section 5 fit the read-out weights by regression onto the exact solution, which isolates the approximation error. Here we instead train RaNNs only by minimising the PINN loss, and compare them with fully trainable PINNs. We consider the PME with $m = 2$ in $d = 1, 2$. Since

Table 1: Fitted log-log slopes of the relative L^2 error for the PME with hidden weights drawn from π and from the multivariate Student-t variant. Slopes are fitted to the mean errors; the 95% confidence intervals are computed from slopes fitted to each of the 3 draws.

d	widths N	π	Student-t variant
1	$10, 14, 20, 50, 100, 150, 250, 500, 750, 1000, 1500, 2000, 2500$	-0.49 ± 0.16	-0.48 ± 0.15
2	$20, 50, 100, 150, 250, 500, 750, 1000, 1500, 2000, 2500$	-0.53 ± 0.09	-0.53 ± 0.14
3	$20, 50, 100, 150, 250, 500, 750, 1000, 1500, 2000, 2500$	-0.58 ± 0.11	-0.57 ± 0.12
4	$250, 500, 750, 1000, 1500, 2000, 2500, 5000$	-0.60 ± 0.04	-0.57 ± 0.04
5	$250, 500, 750, 1000, 1500, 2000, 2500, 5000$	-0.52 ± 0.08	-0.50 ± 0.05

the residual involves second derivatives, both models are trained and evaluated on the smooth interior of the support, $\{(t, x) : t \in [0, T],\ \|x - x_0\| \le 0.8\, R(t)\}$, with $T = 0.1$, where $R(t)$ is the radius of the support of (28), and the exact solution is imposed as data on the parabolic boundary of this region (the initial slice and the lateral surface). There the Barenblatt solution is bounded away from zero, the equation is uniformly parabolic and the solution is smooth, so the hypotheses of Corollary 4.1 hold; no values of the solution in the interior are used for training. Both models minimise

$$\mathcal{J} = \mathcal{J}_{\mathrm{res}} + 10\,(\mathcal{J}_{\mathrm{ic}} + \mathcal{J}_{\mathrm{bc}}),$$

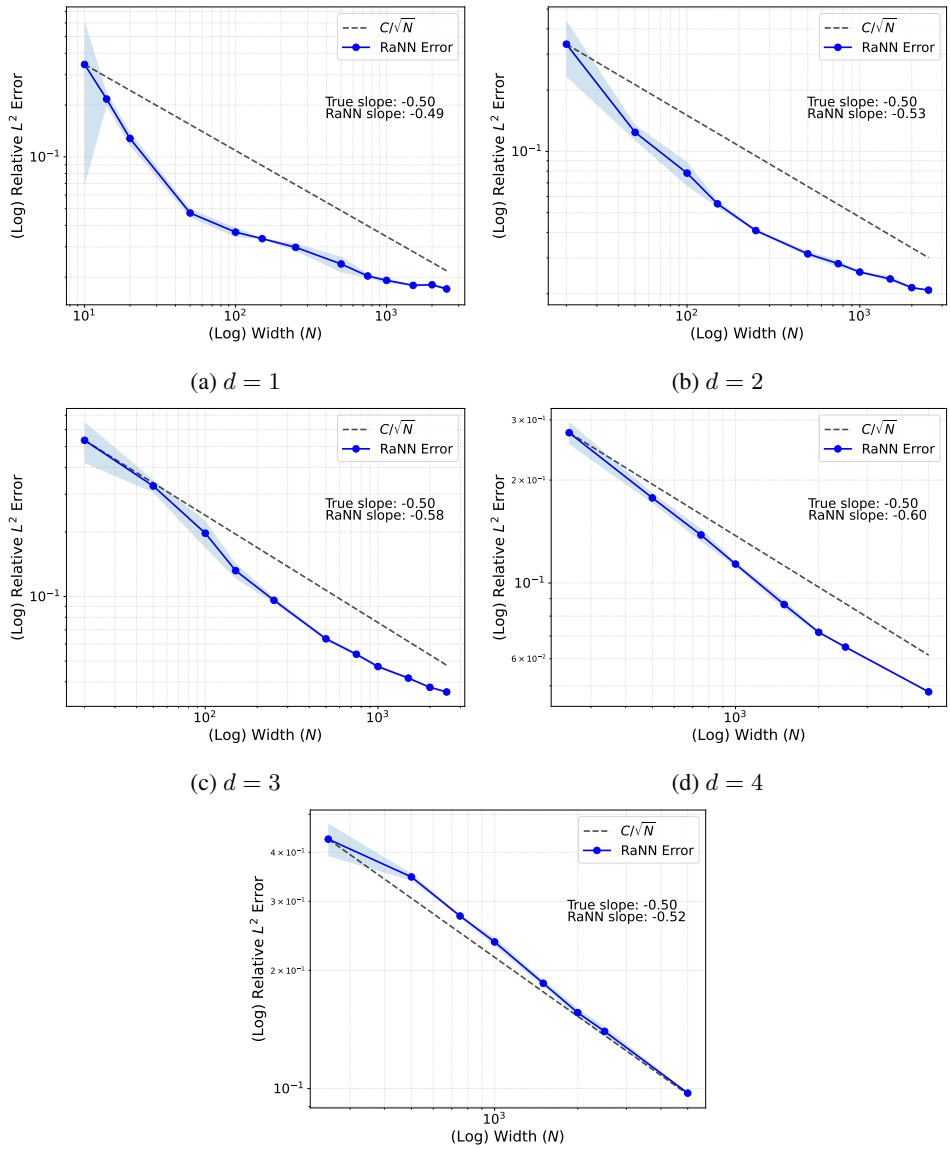
where $\mathcal{J}_{\mathrm{res}}$ is the discretisation of (18) on 20000 interior collocation points and $\mathcal{J}_{\mathrm{ic}}, \mathcal{J}_{\mathrm{bc}}$ are the squared errors on 4000 points of the initial slice and of the lateral boundary respectively, each term normalised by the corresponding norm of the exact solution.

The two models are compared at matched numbers N of trainable parameters. The RaNN has N read-out weights over $N/2$ random features whose hidden weights are drawn either from π (12) or from its multivariate Student-t variant (Appendix B.1), and kept fixed. The fully trainable network is a $\tanh$ MLP with two hidden layers whose total number of parameters is the largest one not exceeding N. Both are trained with the same procedure: Adam (1000 steps, learning rate 10^{-3}) followed by L-BFGS (up to 3000 iterations). We report the RMS PDE residual and the relative L^2 error on 20000 held-out interior points, averaged over 3 independent draws of the random features (respectively initialisations of the MLP), in Table 2.

Table 2: RaNN versus fully trainable PINN, both trained on the PINN loss for the PME on the interior of the support. Slopes are fitted to the mean values; the 95% confidence intervals are computed from slopes fitted to each of the 3 draws. RaNN (π) and RaNN (t) denote features drawn from π and from its multivariate Student-t variant.

		$N = 80$	160	320	640	1280	slope
RMS PDE residual							
$d = 1$	RaNN (π)	5.97×10^{-1}	4.32×10^{-1}	2.07×10^{-1}	1.34×10^{-1}	7.69×10^{-2}	-0.76 ± 0.07
	RaNN (t)	6.57×10^{-1}	3.78×10^{-1}	2.18×10^{-1}	1.26×10^{-1}	8.05×10^{-2}	-0.76 ± 0.08
	PINN	6.82×10^{-2}	2.22×10^{-2}	2.36×10^{-2}	2.76×10^{-2}	1.36×10^{-2}	-0.43 ± 0.78
$d = 2$	RaNN (π)	9.83×10^{-1}	2.98×10^{-1}	6.65×10^{-2}	3.26×10^{-2}	1.82×10^{-2}	-1.47 ± 0.29
	RaNN (t)	9.04×10^{-1}	2.59×10^{-1}	7.44×10^{-2}	3.22×10^{-2}	1.64×10^{-2}	-1.46 ± 0.12
	PINN	4.21×10^{-2}	3.60×10^{-2}	1.61×10^{-2}	1.36×10^{-2}	1.07×10^{-2}	-0.53 ± 0.69
Relative L^2 error							
$d = 1$	RaNN (π)	6.89×10^{-3}	4.25×10^{-3}	1.75×10^{-3}	1.36×10^{-3}	7.07×10^{-4}	-0.82 ± 0.14
	RaNN (t)	7.78×10^{-3}	3.25×10^{-3}	1.93×10^{-3}	1.00×10^{-3}	6.67×10^{-4}	-0.88 ± 0.05
	PINN	1.45×10^{-3}	4.69×10^{-4}	4.75×10^{-4}	5.53×10^{-4}	2.64×10^{-4}	-0.47 ± 0.79
$d = 2$	RaNN (π)	1.25×10^{-1}	1.94×10^{-2}	3.15×10^{-3}	1.44×10^{-3}	7.59×10^{-4}	-1.85 ± 0.37
	RaNN (t)	9.90×10^{-2}	1.52×10^{-2}	3.53×10^{-3}	1.43×10^{-3}	6.84×10^{-4}	-1.78 ± 0.23
	PINN	2.56×10^{-3}	2.32×10^{-3}	1.13×10^{-3}	8.40×10^{-4}	6.27×10^{-4}	-0.55 ± 0.80

In both dimensions the RaNN residual decays at least as fast as $N^{-1/2}$, which is what Corollary 4.1 asserts. A slope steeper than $-1/2$ is consistent with the corollary, which is an upper bound. The fully trainable network attains a smaller residual at every width, but the gap narrows as N grows: at $N = 1280$ the RaNN residual is larger by a factor of 5.9 in $d = 1$ and 1.5 in $d = 2$ (Student-t



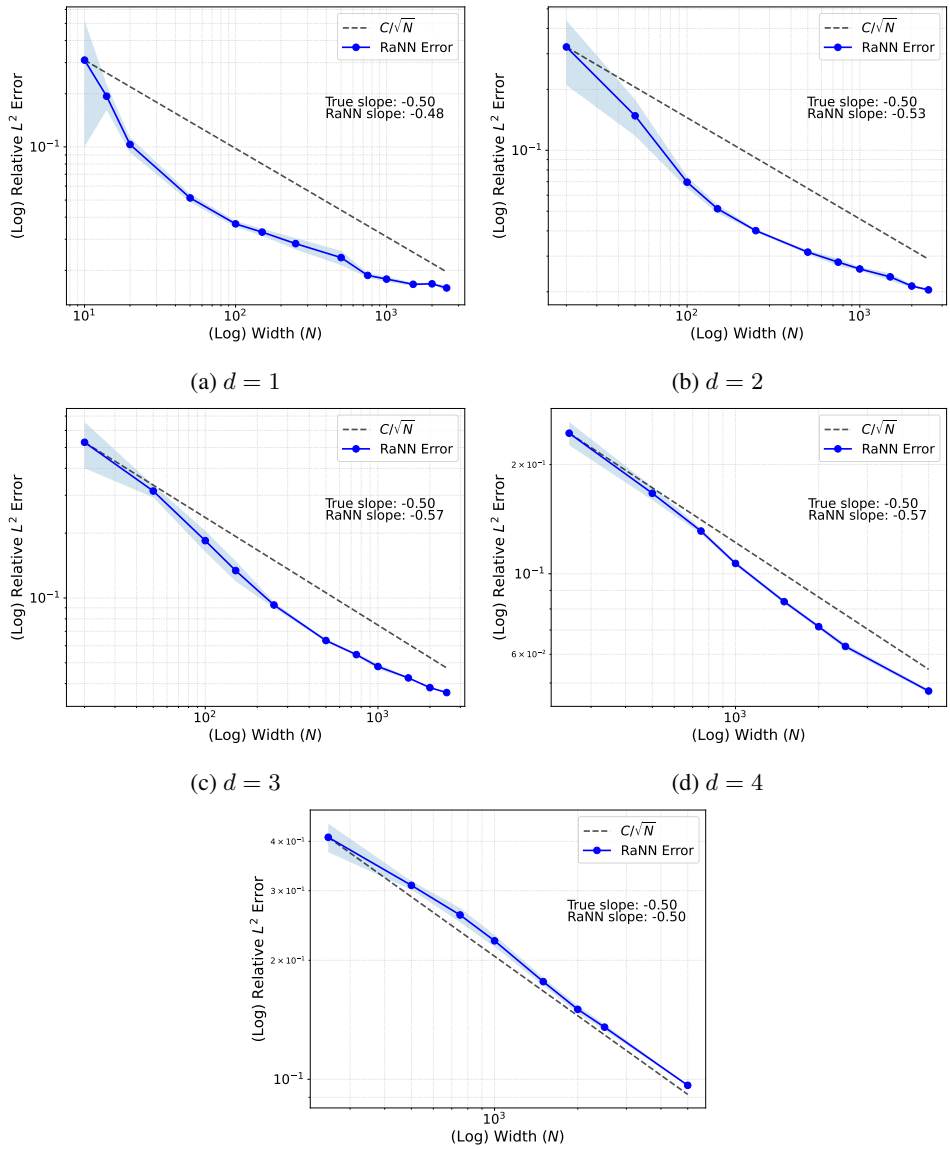
(a) $d = 1$ (b) $d = 2$

(c) $d = 3$ (d) $d = 4$

(e) $d = 5$

Figure 5: Log-log plot of the relative L^2 error versus the width N for the PME with hidden weights drawn from π. The shaded band indicates the region within one standard deviation of the mean over 3 draws.

features; 5.7 and 1.7 with π), and in $d = 2$ the relative L^2 errors are almost equal. The rate measured for the RaNN is also considerably more reproducible across draws than that of the fully trainable network (± 0.08 and ± 0.12 with Student-t features, ± 0.07 and ± 0.29 with π, against ± 0.78 and ± 0.69 for the residual). The two feature laws give essentially the same rates. This is a consequence of the a-priori construction: with the features frozen the model is linear in its trainable parameters, so the optimisation problem is better behaved than that of the fully trainable network (see also Appendix D). We stress that, unlike the experiments of Section 5, the errors reported here include optimisation and collocation errors in addition to the approximation error bounded by Theorem 3.4.



(a) $d = 1$ (b) $d = 2$

(c) $d = 3$ (d) $d = 4$

(e) $d = 5$

Figure 6: Log-log plot of the relative L^2 error versus the width N for the PME with hidden weights drawn from the multivariate Student-t variant of π. The shaded band indicates the region within one standard deviation of the mean over 3 draws.

B.3 Navier-Stokes

We now turn to the compressible Navier-Stokes system, given by (21). We restrict our attention to the one-dimensional case, where (21) is reduced to (in Lagrangian mass coordinates)

$$\begin{cases} \partial_t v - \partial_x u = 0, \\ \partial_t u + \partial_x p_\epsilon(v) - \mu \partial_x \left(\frac{1}{v} \partial_x u \right) = 0. \end{cases} \tag{105}$$

Here, $v = 1/\rho$ represents the specific volume and u the velocity. We consider the singular pressure in the work of [18].

$$p_\epsilon(v) = \frac{\epsilon}{(v-1)^\gamma}, \qquad \gamma > 0. \tag{106}$$

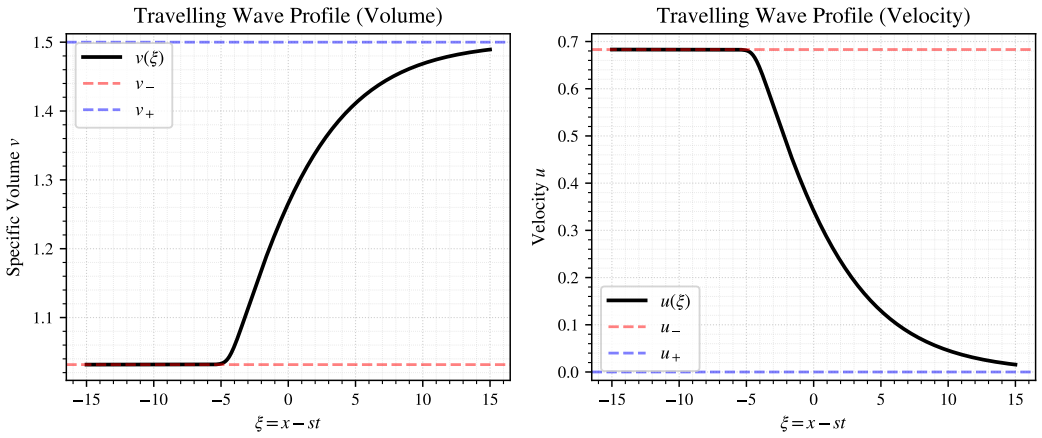


Figure 7: The travelling wave solution $(\mathfrak{v}, \mathfrak{u})$ to (105), obtained by solving the ODE (107).

In order to evaluate the performance of the randomised PINN method, we need a baseline solution analogous to the Barenblatt profile ((28)) which we used for the Porous Medium Equation. For this purpose, we consider the travelling shock-wave solutions to system (105), which were studied by [18]. Shock wave solutions to compressible Navier-Stokes models have also been studied in other works ([40, 19, 35, 65]). The travelling wave solutions to system (105) can be obtained by taking the ansatz $(v, u)(t, x) = (\mathfrak{v}, \mathfrak{u})(x - st)$, where s is the shock speed. This reduces the PDE to an ODE for $\mathfrak{v}$:

$$\mathfrak{v}' = \frac{\mathfrak{v}}{\mu s}(s^2(v_- - \mathfrak{v}) + p_\epsilon(v_-) - p_\epsilon(\mathfrak{v})). \tag{107}$$

The velocity $\mathfrak{u}$ can then be obtained from the relationship $\mathfrak{u} = s(v_+ - \mathfrak{v})$, which follows from the conservation of mass.

We consider the domain $(0, T) \times (-5, 5)$ with $T = 1.0$, $\mu = 10, \epsilon = 10^{-3}, \gamma = 2$. The shock profile connects a far-field state v_- to the far-field state v_+. We fix $v_+ = 1.5$, while $v_- < v_+$ and the shock speed s are derived from the Rankine-Hugoniot jump condition (see Proposition 1.1 of [18]).

To obtain the baseline solution $(\mathfrak{v}, \mathfrak{u})$, we numerically integrate the ODE using the scipy.integrate.solve_ivp solver on an interval $\xi \in [-15, 15]$ with 5000 points. The velocity profile $\mathfrak{u}(\xi)$ is then obtained from $\mathfrak{u} = s(v_+ - \mathfrak{v})$. The travelling wave solution can be seen below in Figure 7.

In an effort to minimise optimisation error and adhere to the setting of Theorem 3.4 as closely as possible, we choose to use a supervised learning approach when finding a Randomised Neural Network. We include Fourier features (see Appendix C for details) where frequencies ω_j are drawn from $\mathcal{N}(0, 3.5^2)$. Note that we choose to use a smaller variance than for the PME because the solution is of lower frequency than the Barenblatt profile.

We sweep across a range of widths $N \in \{50, ..., 250\}$ and aim to minimise the mean-squared error between the network and the solution to the ODE. For each width, we train the model using Ridge regression on a dataset where the sample size is $M = 2000N$ (M grows with N to avoid an underdetermined problem, ensuring the optimisation problem remains stable). For any given training point (t_i, x_i), the evaluation of the baseline solution $(\mathfrak{v}, \mathfrak{u})$ is obtained by linear interpolation on the ODE grid. We sample uniformly in time but use a mixture of uniform and importance sampling in space; 50% of points are sampled uniformly on $[-5, 5]$ whereas 50% of points are sampled from a normal distribution $N(x_0, 1)$ around the shock location $x_0(t) = x_0 + st$. The frequencies for the Fourier features ω_j are sampled from $\mathcal{N}(0, 3.5^2)$. We find the smaller variance of 3.5^2 to be effective for the simpler behaviour of a travelling wave solution.

The network is trained to minimise the L^2-regularised MSE (Ridge regression loss) :

$$\mathcal{L}(\mathbf{W}) = \frac{1}{M}\sum_{i=1}^{M} \|\hat{\mathbf{v}}(t_i, \mathbf{x}_i) - \mathbf{y}_i\|_2^2 + \lambda \|\mathbf{W}\|_2^2, \qquad \lambda = 10^{-10}, \tag{108}$$

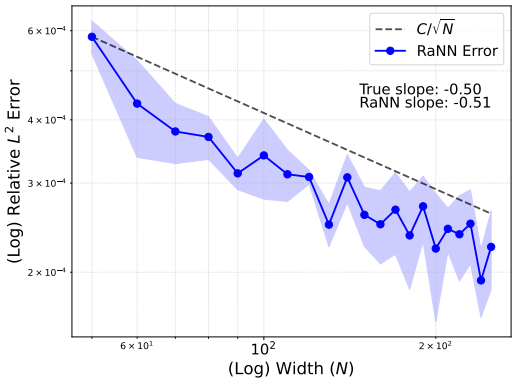


Figure 8: Approximation error of RaNNs of varying width N for solving the compressible Navier-Stokes system on $(0,1) \times [-5,5]$ in logarithmic scale.

where $\hat{\mathbf{v}}$ is the network output and $\mathbf{W}$ is the vector of output weights. The minimiser $\hat{\mathbf{W}}$ has a closed-form which can be explicitly calculated and used to generate the final network. With the final network, we compute the L^2 errors relative to the baseline solution using a set of $20,000$ (pre-generated) points. Each width N is tested five times and the mean relative error is recorded for each N. These errors are plotted against N in Figure 2. The errors are plotted on a log-log scale in Figure 8.

C Random Feature Neural Networks

It is known that classical PINNs suffer from a spectral bias phenomenon [50], which essentially means that the network is biased towards learning lower frequency functions (see[67, 68, 70]). This can be troublesome, particularly for non-linear PDEs whose solutions are often highly chaotic. [63] suggested to use Fourier features to overcome this.

To carry out the simulations in Section 5, we integrated Fourier feature embeddings into the RaNN network. We now describe the architecture of a network with Fourier feature embeddings. Instead of choosing a smooth activation such as $\tanh$, we take $\sigma(z) = \cos(z)$ and include both cosine and sine activations for symmetry. Concretely, let $\{\tau_i, \mathbf{a}_i\}_{i=1}^N \subset \mathbb{R}^{1+d}$ and $\{b_i\}_{i=1}^N \subset [0, 2\pi]$ be frozen random samples, and define

$$\phi_i(t,x) = \cos(\tau_i t + \mathbf{a}_i \cdot x + b_i), \qquad \psi_i(t,x) = \sin(\tau_i t + \mathbf{a}_i \cdot x + b_i). \tag{109}$$

We then construct the feature vector

$$\Phi(t,x) = \frac{1}{\sqrt{N}}\big(\phi_1(t,x), \dots, \phi_N(t,x), \psi_1(t,x), \dots, \psi_N(t,x)\big), \tag{110}$$

and take

$$u_W(t,x) = \beta + \sum_{i=1}^{N} \big(\alpha_i \phi_i(t,x) + \gamma_i \psi_i(t,x)\big), \tag{111}$$

where the coefficients $\{\alpha_i, \gamma_i\}_{i=1}^N$ and bias β are the trainable parameters. The prefactor $N^{-1/2}$ normalises the variance of the features, and does not affect the approximation class.

We can express the sum of sine and cosine functions as a single shifted cosine with amplitude W_i, giving us the form

$$u_W(t,x) = \beta + \sum_{i=1}^{N} W_i \cos(\tau_i t + \mathbf{a}_i \cdot \mathbf{x} + \tilde{b}_i), \tag{112}$$

where $W_i := \sqrt{\alpha_i^2 + \gamma_i^2}$, $\theta_i := \operatorname{atan2}(\gamma_i, \alpha_i)$ and $\tilde{b}_i := b_i - \theta_i$. This shows that the RaNN used in our experiments is of the same general form as (1), with smooth activation $\sigma = \cos$ (the sign of the bias is immaterial since its distribution is symmetric). Recall from Remark A.1 that $\sigma = \cos$ is an admissible choice, meaning that the approximation result of Theorem 3.4 directly applies to networks of the form (111).

D Using the bounds inside non-linear solvers

Theorem 3.4 bounds the approximation error of the drawn class of RaNNs and does not analyse a particular algorithm for determining the read-out weights. When no reference solution is available, the read-out weights are determined from the equation itself, for instance by least squares on the PDE residual as in the physics-informed extreme learning machine [25] and the random feature method [14], by reducing training to the solution of an ODE [20], or within the random deep splitting scheme [47]. Our bounds supply the approximation-error term in the error analysis of each of these methods. We outline two ways of handling the non-linearity of the PDE, and the choice of random weights across iterations.

Direct residual minimisation. With the hidden weights frozen, a RaNN is linear in its read-out weights W. For a linear PDE the residual loss is therefore a convex quadratic in W with a closed-form minimiser. For the PME with $m = 2$ the residual is quadratic in the network, so the loss is a polynomial least-squares problem in W; it is not convex, but it can be solved by Gauss–Newton methods. Neither structure is available for a fully trainable network, which is non-convex through its parametrisation regardless of the form of the residual.

Picard linearisation. Alternatively, the non-linear problem can be replaced by a sequence of linear ones. For the PME with $m = 2$, given a RaNN $u^{(n)}$ one solves

$$\partial_t u^{(n+1)} - \nabla \cdot \left(2u^{(n)} \nabla u^{(n+1)}\right) = 0, \qquad u^{(n+1)}(t_0, \cdot) = u_0,$$

for a RaNN $u^{(n+1)}$. Since the coefficient $u^{(n)}$ is known, each subproblem is linear in the read-out weights of $u^{(n+1)}$ and can be solved by regression.

Random weights across iterations. For a single regression, Theorem 3.4 together with Markov's inequality (as in the proof of Corollary 4.1) gives the error bound with probability at least $1 - \delta$ over the random weights. Over K iterations one needs the bound to hold at all K steps simultaneously: applying it at each step with δ replaced by δ/K, a union bound gives probability at least $1 - \delta$. Since the bound is proportional to $1/\delta$, the rate in N is unchanged and only the constant grows, by a factor K. This accounting is needed both when a single draw of random weights is reused at every step and when fresh weights are drawn at each step. Reusing one draw allows the feature matrices to be computed once, while redrawing avoids being tied to a draw that is poorly suited to the solution, at the cost of recomputing the feature matrices at each step. Our bounds do not control how the error made at one step propagates to the next; this would require a stability estimate for the linearised scheme and is left for future work.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction claim that we obtain a rigorous approximation bound for sufficiently smooth functions in time-dependent Sobolev spaces by randomised neural networks (RaNNs) with rate $N^{-1/2}$, where N is the width of the RaNN. In Theorem 3.4 and particularly equation (14) we see that the factor of N appearing on the RHS is N^{-1}, which when square rooted (because the square of the norm appears on the LHS) gives $N^{-1/2}$, verifying the claim that we prove an approximation result with rate $N^{-1/2}$. Furthermore, since solutions to non-linear PDEs live in time-dependent Sobolev spaces, this approximation result (14) can be applied to such solutions and in particular can be used to obtain bounds on PINN residuals. In Section 4 it is applied to two PDEs where we obtain approximation bounds on the solutions as well as the PINN residuals and carry out experiments to verify the rate.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: We note the limitations through two remarks and the conclusion. In Remark 3.5 we note that the space of admissible functions does depend on dimension. In Remark A.2 we mention that a limitation of our work is that the growth of the coefficient in (14) is difficult to understand, but we obtain a bound that grows at most polynomially in d in a simpler case. In the conclusion we highlight that future work may investigate less smooth solutions (i.e. reducing the assumptions of Theorem 3.4), using a different distribution for the weights, improving the constant of proportionality and investigating whether the spectral bias that PINNs face carries over to RaNNs.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.

- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

   Justification: Each result contains all of the necessary assumptions. The full proofs are also given in the appendices.

   Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer: [Yes]

   Justification: The methodology is fully specified in Appendix B.1.- B.3. Each of the hyperparameters is given, the loss function and the form of the solution targeted. The code attached to the submission is stand-alone and allows the experiments to be reproduced. The code is publicly available at `https://github.com/M6LI/RaNN-Expressivity-NonLinearPDEs`, with instructions in its README. The experiments do not use external datasets.

   Guidelines:

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

- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

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

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

   Answer: [Yes]

   Justification: The paper references the publicly available repository https://github.com/M6LI/RaNN-Expressivity-NonLinearPDEs where the code can be found. The code is also attached to the submission.

   Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: The training and test details are fully specified in Appendix B.1. - B.3. The full algorithm details, including each of the hyperparameters is given

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.

- The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: The plots (e.g. Figure 1) include a shaded band representing a region within one standard deviation of the mean relative L^2 error. Error bars are also in Tables 1-2.

   Guidelines:

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: We mention in Appendix B that all experiments were run on a NVIDIA RTX 3080 GPU with 10GB VRAM.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: We have adhered to the NeurIPS code of ethics in its entirety.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: There is no societal impact of the work performed.

    Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

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

Answer: [N/A]

Justification: The paper does not use existing assets.

Guidelines:

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

13. **New assets**

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

Answer: [N/A]

Justification: The paper does not release new assets.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

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

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

Answer: [N/A]

Justification: The paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

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

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: No LLMs used.

Guidelines:

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