European Journal of Applied Mathematics (2026), **37**, pp. 238–258
doi:10.1017/S0956792525100181

CAMBRIDGE UNIVERSITY PRESS

PAPER

Do stable neural networks exist for classification problems? – A new view on stability in AI

David Liu[1] and Anders Hansen[2]

[1]CCIMI, University of Cambridge, Cambridge, UK
[2]DAMTP, University of Cambridge, Cambridge, UK
Corresponding author: Anders Hansen; Email: ach70@cam.ac.uk

Received: 07 June 2024; **Revised:** 05 September 2025; **Accepted:** 08 September 2025;
First published online: 20 November 2025

Keywords: stability; neural networks; measure theory; robustness of AI; universal approximation theorem; adversarial attacks

2020 Mathematics Subject Classification: 41Axx (Primary); 28A20, 68T07, 46Nxx (Secondary)

Abstract

In deep learning (DL), the instability phenomenon is widespread and well documented, and the most commonly used measure of stability is the Lipschitz constant. While a small Lipchitz constant is traditionally viewed as guarantying stability, it does not capture the instability phenomenon in DL for classification well. The reason is that a classification function – which is the target function to be approximated – is necessarily discontinuous, thus having an 'infinite' Lipchitz constant. As a result, the classical approach will deem every classification function unstable, yet basic classification functions a la 'is there a cat in the image?' will typically be locally very 'flat' – and thus locally stable – except at the decision boundary. The lack of an appropriate measure of stability hinders a rigorous theory for stability in DL, and consequently, there are no proper approximation theoretic results that can guarantee the existence of stable networks for classification functions. In this paper, we introduce a novel stability measure $\mathcal{S}(f)$, for any classification function f, appropriate to study the stability of classification functions and their approximations. We further prove two approximation theorems: First, for any $\epsilon > 0$ and any classification function f on a *compact set*, there is a neural network (NN) ψ, such that $\psi - f \neq 0$ only on a set of measure $< \epsilon$; moreover, $\mathcal{S}(\psi) \geq \mathcal{S}(f) - \epsilon$ (as accurate and stable as f up to ϵ). Second, for any classification function f and $\epsilon > 0$, there exists a NN ψ such that $\psi = f$ on the set of points that are at least ϵ away from the decision boundary.

1. Introduction

With the rise of adversarial attacks in deep learning (DL) for image classification, the universal instability of DL methods across various scientific fields has become evident [5, 8, 20, 24, 36, 54, 55, 56, 63, 67, 69]. This underscores the urgent need to investigate the stability properties of neural networks (NN). Traditionally, the size of the Lipschitz constant has been a common metric for such investigations [14, 17, 33, 51]. While this approach is useful in many scenarios, it falls short for discontinuous functions, which have 'infinite' Lipschitz constants. Consequently, expecting a NN to accurately approximate a classification function with a 'small' Lipschitz constant is unrealistic, given that the target function is inherently unstable. This issue is particularly problematic for DL, whose major strength lies in image recognition [35, 36, 54, 59] – an inherently discontinuous task. Empirical observations of instabilities and hallucinations in image recognition further highlight this problem [6, 9, 11, 43, 48, 50, 57, 64, 66, 68, 75, 76]. The instability issue in DL is considered one of the key problems in modern AI research, as pointed out by Y. Bengio: 'For the moment, however, no one has a fix on the overall problem of brittle AIs' (from 'Why deep-learning AIs are so easy to fool' [48]). This leads to the key problem addressed in this paper:

Do stable neural networks exist for classification problems?

© The Author(s), 2025. Published by Cambridge University Press. This is an Open Access article, distributed under the terms of the Creative Commons Attribution licence (https://creativecommons.org/licenses/by/4.0/), which permits unrestricted re-use, distribution and reproduction, provided the original article is properly cited.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

Check for updates

Conceptually, there is a lack of a comprehensive theory for the stability of classification functions. While it might be tempting to categorise all classification functions as unstable, this overlooks the varying degrees of instability among discontinuous functions. For instance, the Heaviside step function intuitively appears more stable than the Dirichlet function, which is nowhere continuous. To address this issue, we introduce a new stability measure called *class stability*. This measure is designed to study the stability of discontinuous functions and their approximations by extending classical measure theory. The proposed stability measure focuses on the closest points with different functional values, capturing the phenomenon more effectively. This concept aligns with the emerging notion of the 'margin' in the machine learning community, which is a local measure of stability [51]. Our concept of class stability extends this notion to the entire function across its domain, allowing for a comparison of the stability of different discontinuous functions. We provide two working definitions of class stability: one based on an analytic distance metric, and an alternative defined in a measure theoretic way.

Finally, in the spirit of existing approximation papers [2–4, 12, 15, 19, 21, 27–31, 34, 40, 42, 44, 47, 49, 53, 60–62], we prove the existence of NNs with class stabilities approximating the target function. Using results from approximation theory, analysis and measure theory, we prove two major theorems. The first one states that NNs are able to interpolate on sets that have a class stability of at least $\epsilon > 0$, thereby proving that NNs can approximate any 'stable' function (see Lemma 2.3). The second is regarding the ability for NNs to approximate any function, such that the class stability of the NN is at most $\epsilon > 0$ smaller than the class stability of the target function. These results demonstrate that the class stability is appropriate to study stability for classification functions.

2. Main result

Our main contribution in this paper is the introduction of 'class stability' and two corresponding stability theorems for NNs. The class stability is defined in (2.3) in Section 4. Intuitively, class stability represents the average distance to the decision boundaries of the function. The first of the two theorems addresses the restriction of classification functions to sets where the classification functions have a class stability of at least $\epsilon > 0$.

To state the main theorems, we need the following five concepts that will be formally defined later in the paper:

(I) (Classification function). We call $f : \mathcal{M} \to \mathcal{Y}$, where $\mathcal{M} \subset \mathbb{R}^d$ is the *input domain* and $\mathcal{Y} \subset \mathbb{Z}^+$ a finite subset, a *classification function*. This is the function we are typically trying to learn.

(II) (Extension of a classification function). Given a classification function $f : \mathcal{M} \to \mathcal{Y}$, we define its extension to $\mathbb{R}^d$ as $\bar{f} : \mathbb{R}^d \to \overline{\mathcal{Y}}$ such that

$$\bar{f}(x) = \begin{cases} f(x) & \text{if } x \in \mathcal{M}, \\ -1 & \text{otherwise}, \end{cases} \tag{2.1}$$

where $\overline{\mathcal{Y}} = \mathcal{Y} \cup \{-1\}$.

(III) (Distance to the decision boundary). Given the extension of a classification function $\bar{f} : \mathbb{R}^d \to \overline{\mathcal{Y}}$ and a real number $1 \leq p \leq \infty$, we define $h^p_{\bar{f}} : \mathbb{R}^d \to \mathbb{R}^+$, the ℓ^p-distance to the decision boundary, as

$$h^p_{\bar{f}}(x) = \inf\{\|x - z\|_p : \bar{f}(x) \neq \bar{f}(z),\ z \in \mathbb{R}^d\}. \tag{2.2}$$

(IV) (Class stability). If $\mathcal{M} \subset \mathbb{R}^d$ is compact, then, we define the ℓ_p-*stability of* $\bar{f}$ to be

$$\mathcal{S}^p_{\mathcal{M}}(\bar{f}) = \int_{\mathcal{M}} h^p_{\bar{f}}\, d\mu, \tag{2.3}$$

where μ is the Lebesgue measure on $\mathbb{R}^d$. We will reference this as the **class stability** of the function $\bar{f}$.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

(V) (Class prediction function). For a given $n \in \mathbb{N}$, we define the *class prediction function* $p_n : \mathbb{R}^n \to \{1, \dots, n\}$ as

$$p_n(x) = \min\{i : x_i \geq x_j, \forall j \in \{1, \dots, n\}\}. \tag{2.4}$$

The class prediction function has the same function as the 'argmax' function in, for example, the numpy library of python. This function takes a vector and returns the index of the element that has the highest value of all elements. If there are multiple such indices that satisfy the maximality, we return the first index.

We can now state the first of our main theorems.

Theorem 2.1 (Interpolation theorem for stable sets). *Let $\mathcal{M}, \mathcal{K} \subset \mathbb{R}^d$, where $\mathcal{K}$ is compact, and $f : \mathcal{M} \to \mathcal{Y} \subset \mathbb{Z}^+$ be a non-constant classification function where $\mathcal{Y}$ is finite. Define*

$$\mathcal{M}_\epsilon := \{x \mid x \in \mathcal{M}, h^p_{\bar{f}}(x) > \epsilon\}, \quad \epsilon > 0, \tag{2.5}$$

as the ϵ-stable set of $\bar{f}$, where $h^p_{\bar{f}}$ is the ℓ^p-distance to the decision boundary defined in (2.2). Then, for any $\epsilon > 0$ and any continuous non-polynomial activation function ρ, which is continuously differentiable at least at one point with non-zero derivative at that point, we have the following:

(1) There exists one hidden layer (see Lemma 5.1) NN $\Psi_1 : \mathcal{K} \to \overline{\mathcal{Y}}$, with an activation function ρ, that interpolates f on $\mathcal{M}_\epsilon$, in particular

$$p_q(\Psi_1(x)) = f(x) \quad \forall x \in \mathcal{M}_\epsilon \cap \mathcal{K}, \tag{2.6}$$

where p_q is the class prediction function, given by Eq. (2.4), that 'rounds' to discrete values and $q = |\mathcal{Y}|$.

(2) There exists a neural network $\Psi_2 : \mathcal{K} \to \overline{\mathcal{Y}}$, using the activation function ρ, with fixed 'width' (see Definition 5.1) of $d + q + 2$, that interpolates f on $\mathcal{M}_\epsilon$, in particular

$$p_q(\Psi_2(x)) = f(x) \quad \forall x \in \mathcal{M}_\epsilon \cap \mathcal{K}. \tag{2.7}$$

Remark 2.2 (Deep and Shallow neural networks). By a shallow network, we mean a NN Lemma 5.1 with one hidden layer, while the width of $d + q + 2$ refers to a NN with hidden layers of size less than or equal to $d + q + 2$.

Remark 2.3 (Interpretation of Lemma 2.1). This theorem says that NNs are able to interpolate any classification function restricted to compact sets on which the classification function attains some minimal class stability. In a simplified way, one can say that NNs can interpolate on stable sets $\mathcal{M}_\epsilon$, which are essentially the original set $\mathcal{M}$ but with a small strip of width ϵ removed from the boundary of the set. This way we ensure that we are left with points that are at least ϵ away from the decision boundary, and then we simply interpolate on these sets. It is also important to mention that the approximation theorems utilised here do allow for arbitrary width in the shallow NN case and for arbitrary depth in the deep NN case.

The second theorem relates to the ability of NNs to approximate the stability of the original classification function. The advantage of this theorem is that it also applies to the stability measure in a measure theoretic frameworks and is in a sense a generalisation of the first theorem. To state the second theorem, we need to introduce the measure theoretic versions of the distance to the decision boundary and the class stability:

(VI) (Measure theoretic distance to the decision boundary). For an extension of a classification function $\bar{f} : \mathbb{R}^d \to \overline{\mathcal{Y}}$ and a real number $p \geq 1$, we define $\tau^p_{\bar{f}} : \mathbb{R}^d \to \mathbb{R}^+$ the *l^p-distance to the decision boundary* as

$$\tau^p_{\bar{f}}(x) = \inf\left\{ r : \int_{\mathcal{B}^p_r(x)} \mathbb{1}_{\bar{f}(z) = \bar{f}(x)}\, d\mu \neq \int_{\mathcal{B}^p_r(x)} d\mu, r \in [0, \infty) \right\}.$$

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

Here, μ denotes the Lebesgue measure and $\mathcal{B}_r^p(x)$ the unit closed ball with p-norm, and $\mathbb{1}$ is the indicator function.

(VII) (Class stability (measure theoretic)). If $\mathcal{M} \subset \mathbb{R}^d$ is a compact set, we define the (measure theoretic) ℓ_p-*stability of* $\bar{f}$ to be

$$\mathcal{T}_{\mathcal{M}}^p(\bar{f}) = \int_{\mathcal{M}} \tau \quad {}_{\bar{f}}^{p}(x)\, d\mu. \tag{2.8}$$

Theorem 2.4 (Universal stability approximation theorem for classification functions). *For any Lebesgue measurable classification function $f : \mathcal{M} \subset \mathbb{R}^d \to \mathcal{Y}$, where $\mathcal{M}$ is compact, and $q = |\mathcal{Y}|$; any set $\{(x_i, f(x_i))\}_{i=1}^k$ with $\tau_{\bar{f}}^p(x_i) > 0$ for all $i = 1, \ldots, k$; and any $\epsilon_1, \epsilon_2 > 0$, there exists a NN $\psi \in \mathcal{NN}(\rho, d, q, 1, \mathbb{N})$ (see Lemma 5.1) such that we have the following. The class stability (as defined above in Eq. (2.3)) of the NN satisfies*

$$\mathcal{T}_{\mathcal{M}}^p(\overline{p_q(\psi)}) \geq \mathcal{T}_{\mathcal{M}}^p(\bar{f}) - \epsilon_1, \tag{2.9}$$

we can interpolate on the set

$$p_q(\psi) = f(x_i) \quad i = 1, \ldots, k\,, \tag{2.10}$$

where p_q is the class prediction function, given by Eq. (2.4), that 'rounds' to discrete values, and

$$\mu(R) < \epsilon_2, \quad R := \{x \,|\, f(x) \neq p_q(\psi), x \in \mathcal{M}\}, \tag{2.11}$$

where μ denotes the Lebesgue measure.

Remark 2.5 (Interpretation of Lemma 2.4). This theorem proves that if one wants to use a NN to approximate any fixed classification function, it is possible to achieve with a close to ideal stability, perfect precision (described by the second property) and an arbitrarily good accuracy (third property).

2.1. Computability and GHA vs existence of NNs – Can the brittleness of AI be resolved?

While our results produce a new framework for studying stability of NNs for classification problems and provide theoretical guaranties for the existence of stable NNs for classification functions, the key issue of computability of such NNs is left for future papers. Indeed, as demonstrated in [27, 38], based on the phenomenon of generalised hardness of approximation (GHA) [7, 9] in the theory of the Solvability Complexity Index (SCI) hierarchy [12, 13, 25, 26, 45, 46], there are many examples where one can prove the existence of NNs that can solve a desired problem, but they cannot be computed beyond an approximation threshold $\epsilon_0 > 0$. Thus, what is needed is a theory that combines our existence theorems with GHA for which one can determine the approximation thresholds ϵ_0 that will dictate the accuracy for which the NNs can be computed. This is related to the issue of NN dependency on the input.

Remark 2.6 (Non-compact domains and dependency on the inputs). Note that our results demonstrate that on compact domains, one can always find a NN ϵ-approximation ψ to the desired classification function f, where the stability properties of ψ are ϵ close to the stability properties of f. However, if the domain is not compact, this statement seizes to be true. The effect of this is that stable and accurate NN approximations to the classification function f (on a non-compact domain) can still be found; however, the NN ψ may have to depend on the input. Indeed, by choosing a compact domain K_x based on the input x, one may use our theorem to find a NN ψ_x such that $\psi_x(x) = f(x)$ and ψ_x is stable on K_x. However, ψ_x may have to change dimensions as a function of x. Moreover, if it is possible to make the mapping $x \mapsto \psi_x$ recursive is a big open problem. In particular, resolving the brittleness issue of moderns AI hinges on this question. We mention in passing that there are papers in the machine learning community that deal with local decision boundary estimates in terms of certificates [76], that potentially provide a step towards computing class stable NNs.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

2.2. Related work

Instability in AI: Our results are intimately linked to the instability phenomenon in AI methods – which is widespread [5, 8, 11, 20, 24, 36, 54, 55, 56, 63, 67, 69] – and our results add theoretical understandings to this vast research programme. Notably, our work shares significant connections with the investigations conducted by F. Voigtlaender et al. [19], which also deals with classification functions and their approximations via NNs. There has been significant work done on adversarial attacks by S. Moosavi-Dezfooli, A. Fawzi, and P. Frossard et al. [55, 56]. See also recent developments by D. Higham, I. Tyukin regarding vulnerabilities of neural networks et al. [10, 69]. Furthermore, our research aligns with the exploration of robust learning pursued by L. Bungert, G.Trillos et al. [18] as well as by S. Wang, N. Si, J. Blanchet [71]. The stability problem in NN has also been extensively investigated by V. Antun et al. [27], see also the work by B. Adcock and N. Dexter [2].

Existence vs computability of stable NNs: There is a substantial literature on existence results of NNs [16, 61, 74], see, for example, the aforementioned work by F. Voigtlaender et al. [70], review papers by A. Pinkus [62] and the work by R. DeVore, B. Hanin and G. Petrova [31] and the references therein. For recent results, see the work by G. D'Inverno, S. Brugiapaglia and M. Ravanelli [32], by N. Franco and S. Brugiapaglia [37] and by B. Adcock, S. Brugiapaglia, N. Dexter and S. Morage [1]. Our work also utilises the approximation theorems obtained by P. Kidger and T. Lyons [52]. However, as established in [27] by M. Colbrook, V. Antun et al., only a small subset of the NNs than can be proven to exist can be computed by algorithms. We also need to point out that following the framework of A. Chambolle and T. Pock [22, 23], the results in [27] demonstrate how – under specific assumptions – stable and accurate NNs can be computed. See also the work by P. Niyogi, S. Smale and S. Weinberger [58] on existence results of algorithms for learning.

3. Motivation for new stability measure

In this section, we will motivate the need for a new stability measure for classification functions. We will first discuss the classical approach to stability in NNs, which is based on the Lipschitz continuity and having a bounded Lipschitz constant. We will then demonstrate that the Lipschitz constant is not a suitable measure for classification functions, and introduce the class stability as a new measure for stability.

3.1. Classification functions and Lipschitz continuity

The Lipschitz constant is a standard measure of stability in NNs [14, 17, 33, 51]. While it is suitable to use the Lipschitz constant for continuous functions, it is not appropriate for classification functions. The main problem is summarised in the following proposition.

Proposition 3.1 (Unbounded Lipschitz continuity for classification functions). *Let $\mathcal{M}$ be a connected subset of $\mathbb{R}^d$ and $f : \mathcal{M} \to \mathcal{Y}$ be a classification function that is not a constant function a.e. on $\mathcal{M}$. Then, f is not Lipschitz continuous.*

The proof is elementary and simply follows from the fact that any non-constant discrete function on a connected domain has a discontinuity. This proposition is nothing novel and there are certain methods that researchers have used to deal with the issues caused by the discontinuities. One common assumption that is made is that the classes are separated by some minimal distance, as demonstrated in [73]. This is essentially dropping the connectedness from our assumptions. Furthermore, the issue of isolating the Lipschitz constant is highlighted by the fact that the classes themselves can be labelled by arbitrary numbers. This causes a problem for approaches such as the one in [73] where the distance between any

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

two examples from different classes is assumed to be at least 2r, for some fixed value r . As an example take the following functions

Example 3.2. *Fix an* $\epsilon > 0$. *Let* $H_1 : [-1, -\epsilon] \cup [\epsilon, 1] \to \{0, 1\}$ *defined by*

$$H_1(x) = \begin{cases} 1 & x > 0, \\ 0 & x < 0. \end{cases}$$

Similarly, we define the function $H_2 : [-1, -\epsilon] \cup [\epsilon, 1] \to \{0, 1000\}$ *defined by*

$$H_2(x) = \begin{cases} 1000 & x > 0, \\ 0 & x < 0. \end{cases}$$

These two examples illustrate two separate problems with using Lipschitz continuity for classifications functions. First, both functions are examples of separating different classes of a Heaviside step function by a small interval $(-\epsilon, \epsilon)$, *thereby leading to a finite Lipschitz constant. However, the value of the constant depends on the value of* ϵ, *and diverges as* $\epsilon \to 0$. *The implication of this is that in a machine learning setting, the more data we gather about the target function, the smaller we would expect the minimal distance between different classes to be, which corresponds to a smaller* ϵ. *As the target function in common machine learning tasks is discrete, this would lead to an unbounded Lipschitz constant. Second, the two functions demonstrate that the Lipschitz constant is not invariant under rescaling of the inputs. The function* H_2 *has a much bigger Lipschitz constant than* H_1, *even though they are describing the same classification problem. This showcases that the arbitrary choice of representing different classes as integers, has also an effect on the Lipschitz stability of the function, which we argue is not a desired property.*

3.2. A spectrum of discrete instabilities

Next, we will give examples of functions that all have an unbounded Lipschitz constant, yet somehow one could consider them to have different 'stability'. These examples will also be used to demonstrate desired properties of a more general stability measure.

Example 3.3. *Let* $f_1, f_2, f_3 : [-1, 1] \to \{-1, 1\}$ *be defined by:* $f_1(x) = sgn(x)$,

$$f_2(x) = \begin{cases} -sgn(x) & \text{if } x \in \{-0.5, 0.5\}, \\ sgn(x) & \text{otherwise}, \end{cases}$$

and

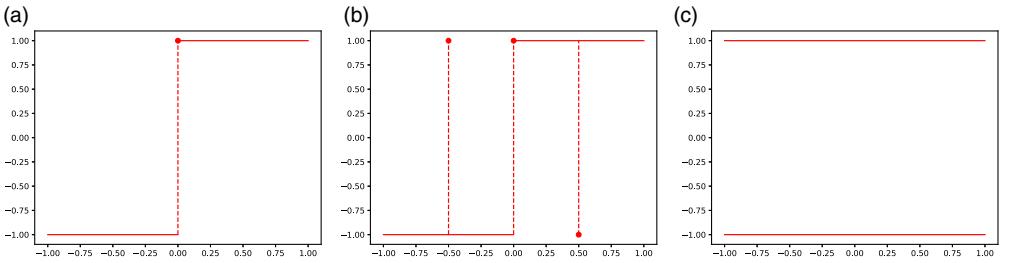
$$f_3(x) = \begin{cases} sgn(x) & \text{if } x \in \mathbb{Q}, \\ -sgn(x) & \text{if } x \in \mathbb{R} \setminus \mathbb{Q}. \end{cases}$$

Here, the function $sgn : \mathbb{R} \to \{-1, 1\}$ *is the sign function (for the sake of the argument, we will assign 0 as positive), that is,*

$$sgn(x) = \begin{cases} 1 & \text{if } x \geq 0, \\ -1 & \text{if } x < 0. \end{cases}$$

All three functions take discrete values, and as such have an unbounded Lipschitz constant. However, one could argue that f_1 *is more stable than* f_2, *which in turn is more stable than* f_3. *The function* f_2 *is just a more unstable version of* f_1, *with* f_3 *being a 'minefield' of instabilities, as any open interval contains points of different labels. This motivates us to define a local measure which takes into account the discontinuities but also the position of them, since a point close to the discontinuity would be more unstable in the sense of 'What is the smallest perturbation needed to change the output of the function?'. The three functions are displayed in Figure 1.*

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press



Standard step function.

Step function with extra discontinuities, making it more 'unstable'.

A function that seems as unstable as it is possible.

Figure 1. *Different classes of unstable classification functions.*

4. Class stability as a measure for 'robustness'

In light of the previous examples, we would like to now define a stability measure that is capable of discerning functions such as f_1, f_2, f_3, while yielding the same stability for H_1 and H_2. First, we will remind the reader about the definition of the distance to the decision boundary as stated in the second section.

Definition 4.1 (Distance to the decision boundary). *For the extension of a classification function* $\overline{f}: \mathbb{R}^d \to \overline{\mathcal{Y}}$ *and a real number* $1 \leq p \leq \infty$, *we define* $h^p_{\overline{f}}: \mathbb{R}^d \to \mathbb{R}^+$ *the* ℓ^p*-distance to the decision boundary as*

$$h^p_{\overline{f}}(x) = \inf\{\|x - z\|_p : \overline{f}(x) \neq \overline{f}(z),\ z \in \mathbb{R}^d\}.$$

It is easy to check that this definition indeed captures the intuitive notion of the 'distance to the decision boundary'. Indeed, the decision boundary is really just the closest points where the label flips. Having the local stability measure, we can now proceed to defining a global measure which would help us differentiate the different types of stabilities of, for example, functions f_1, f_2 and f_3. To assess the stability of a compact set $A \subset \mathbb{R}^d$, we define the stability of a function $\overline{f}$ to be the following:

Definition 4.2 (Class stability of discrete function). *Let* $\overline{f}: \mathbb{R}^d \to \overline{\mathcal{Y}}$ *be a extension of a classification function and* $A \subset \mathbb{R}^d$ *a compact set. Then, for a real number* $1 \leq p \leq \infty$, *we define the* ℓ_p*-stability of* $\overline{f}$ *on A to be*

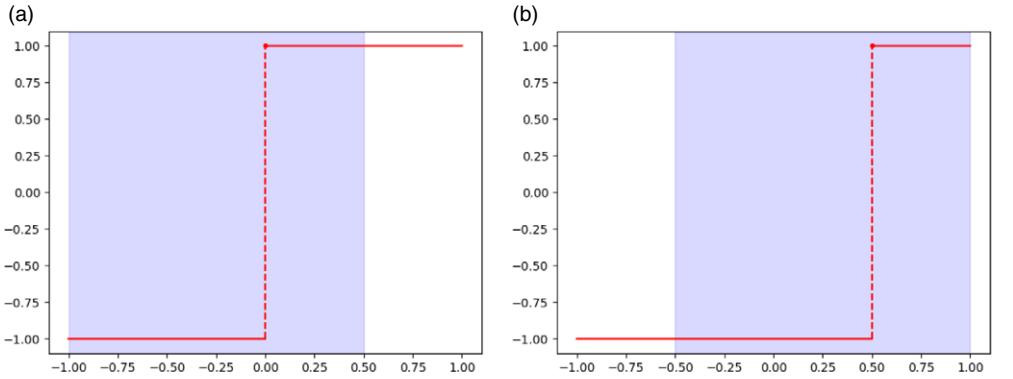
$$h^p_{\overline{f}}(A) = \int_A h^p_{\overline{f}}\, d\mu.$$

We call this stability measure the ***class stability*** *of the function* $\overline{f}$ *on the set A.*

This measure is a generalisation of the local stability measure, as it takes into account the stability of the function on the whole set. If the original classification function was defined on a compact set $\mathcal{M} \subset \mathbb{R}^d$, then $\mathcal{S}^p(\overline{f})$ the ℓ^p-class stability of $\overline{f}$ Eq. (2.3) is well defined.

Let us now examine the ℓ^1-stability of the functions $\overline{f_1}$, $\overline{f_2}$ and $\overline{f_3}$ on the compact set $\mathcal{M} = [-1, 1]$. For f_1, the distance to the decision boundary for a point x is given by $h^1_{\overline{f}}(x) = |x|$. A straightforward calculation yields $\mathcal{S}^1(\overline{f_1}) = 1$. Similarly, we can compute the other values, obtaining $\mathcal{S}^1(\overline{f_2}) = 0.5$ and $\mathcal{S}^1(\overline{f_3}) = 0$. While the specific values depend on the ℓ^p norm chosen, the usefulness of this measure lies in its ability to quantify $\overline{f_3}$ as completely unstable. In fact, $\overline{f_3}$ is deliberately selected to represent one of the worst cases, where any perturbation can cause an extreme change.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press



f_1, the original step function. f_4, the shifted step function.

Figure 2. *Step functions with differently placed steps.*

4.1. Properties of the class stability

Consider two classification functions $f_1, f_4 : \mathcal{M} = [-1, 1] \to \{-1, 1\}$ where

$$f_1(x) = \operatorname{sgn}(x), \quad f_4(x) = \operatorname{sgn}\big(x + \frac{1}{2}\big).$$

The ℓ_1 class stability of these functions on $\mathcal{M}$ are 1 and $\frac{5}{4}$ correspondingly. In fact, it is true for any $p > 0$ that the ℓ_p norm of f_1 is lower than for f_4. We can see from Figure 2 that in both functions, there is a region (shaded blue) for which the points have the exact same stability properties as the relative distance to the decision boundary remains the same. For the remaining points, we can see that the remaining portion f_4 is more stable than the remaining portion of f_1. This property makes sense in the context of how the average stability of the function looks like. If the instability is hidden away from most points, then, in some sense, this is more beneficial to the overall stability.

5. Definitions

In order to prove our main theorems, we will need to define some basic concepts.

Definition 5.1 (Neural network). *Let $\mathcal{NN}^{\rho}_{\mathbf{N},L,d}$ where $\mathbf{N} = (N_L = |\mathcal{Y}|, N_{L-1}, \ldots, N_1, N_0 = d)$ denote the set of all L-layer NNs. That is, all mappings $\phi : \mathbb{R}^d \to \mathbb{R}^{N_L}$ of the form:*

$$\phi(x) = W_L(\rho(W_{L-1}(\rho(\ldots \rho(W_1(x)) \ldots)))), \quad x \in \mathbb{R}^d,$$

where $W_l : \mathbb{R}^{N_{l-1}} \to \mathbb{R}^{N_l}$, $1 \le l \le L$ is an affine mapping and $\rho : \mathbb{R} \to \mathbb{R}$ is a function (called the activation function) which acts component-wise (Note that $W_L : \mathbb{R}^{N_{L-1}} \to \mathbb{R}^{|\mathcal{Y}|}$). Typically this function is given by $\rho(x) = \max\{0, x\}$. L is also referred to as the number of hidden layers.

We will also need to define specific sets of NNs as they are crucial to approximation theorems. To this end, we will use the following notation.

Definition 5.2 (Class of neural networks). *Let $\mathcal{NN}(\rho, n, m, D, W)$ denote the set of NNs $\mathcal{NN}^{\rho}_{\mathbf{N},L,d}$ with an activation function ρ, input dimension n, output dimension m, depth D and width W. In relation to the previous definition, this means*

$$\rho = \rho, \quad L = D, \quad d = n, \quad N_L = m, \quad \max_{i=1,\ldots,L-1} N_i = W.$$

We will also denote the NN class with unbounded depth by $\mathcal{NN}(\rho, n, m, \mathbb{N}, W)$, and similarly the NN class with unbounded width by $\mathcal{NN}(\rho, n, m, D, \mathbb{N})$.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

Definition 5.3 (Class prediction function). *For a given $n \in \mathbb{N}$, we define the class prediction function $p_n : \mathbb{R}^n \to \{1, \dots, n\}$ as*

$$p_n(x) = \min\{i : x_i \geq x_j, \forall j \in \{1, \dots, n\}\}. \tag{5.1}$$

The class prediction function has the same function as the 'argmax' function in for example the numpy library of python. This function takes a vector and returns the index of the element that has the highest value of all elements. If there are multiple such indices that satisfy the maximality, we return the first index.

Remark 5.4 (Training a neural network on a classification task). By training a NN on a classification task we mean that we want to approximate a classification function f, more precisely, its extension. To illustrate why we want the extension, imagine something simple as MNIST. We have 10 target classes, hence $\mathcal{Y} = \{1, 2, \dots, 10\}$ ('zero' is represented by 10 and each other number is represented by itself). Then, we either want to learn f which labels well-defined images correctly, while labelling undefined images randomly, or we want to learn $\overline{f}$ where we label undefined images as -1. Here, f is the ground truth (might be debatable whether it actually exists, but for the purpose of the argument assume it does).

6. Proof of Lemma 2.1

We are now equipped to prove our first main result. Our proof relies on the following two approximation results, the first being the classical approximation theorem for single layer NNs.

Theorem 6.1 (Universal approximation theorem [62]). *Let $\rho \in C(\mathbb{R})$ (continuous functions on $\mathbb{R}$) and assume ρ is not a polynomial. Then, $\mathcal{NN}(\rho, n, m, 1, \mathbb{N})$ (the class of single layer NNs with an activation function of ρ) is dense in $C(\mathbb{R}^n; \mathbb{R}^m)$.*

The second theorem is a newer result that proves the universal approximation property for fixed width NNs.

Theorem 6.2 (Kidger and Lyons [52]). *Let $\rho : \mathbb{R} \to \mathbb{R}$ be any non-affine continuous function which is continuously differentiable at at least one point, with non-zero derivative at that point. Let $\mathcal{K} \subset \mathbb{R}^n$ be compact. Then, $\mathcal{NN}(\rho, n, m, \mathbb{N}, n + m + 2)$(the class of NNs with input dimension n, output dimension m and width of at most $n + m + 2$) is dense in $C(K; \mathbb{R}^m)$ with respect to the uniform norm.*

Before we prove Lemma 2.1, we will first prove a lemma. We start by defining the following functions. For each $i \in \overline{\mathcal{Y}}$, let us define the functions $H_i : \mathcal{M} \to \mathbb{R}$ as:

$$H_i(x) = \begin{cases} h^p_{\bar{f}}(x) & \bar{f}(x) = i, \\ 0 & \text{otherwise.} \end{cases} \tag{6.1}$$

This function can be thought of as an element-wise version of the distance to the decision boundary Eq. (2.2).

Lemma 6.3. *H_i is continuous for all $i \in \overline{\mathcal{Y}}$.*

Proof. Let $\{x_m\}_{m=0}^{\infty}$ be a sequence in $\mathcal{K}$ with $x_m \to x'$ as $m \to \infty$, where $x' \in \mathcal{K}$. First, we take care of the simple case where $\overline{f}(x') \neq i$. Then, we know that $H_i(x') = 0$ and that for x_m we have $0 \leq H_i(x_m) \leq \|x_m - x'\|_p$. Thus, $H_i(x_m) \to H_i(x')$ as $m \to \infty$. Therefore, we can assume $\overline{f}(x') = i$ in which case we distinguish three cases.

Case 1 : $\exists j \in \mathbb{N}$ *such that* $\overline{f}(x_m) = i$, $\forall m > j$. Pick an $\epsilon > 0$. Then, there exists a $l \in \mathbb{N}$ such that $\|x_m - x'\|_p < \epsilon/2$ for all $m > l$. As $\overline{f}(x') = i$, it follows by the definition of $h^p_{\bar{f}}$, that there must exist a sequence of $\{z'_\alpha\}_{\alpha=0}^{\infty}$ such that

$$\|x' - z'_\alpha\|_p \to h^p_{\bar{f}}(x') \quad \text{as } \alpha \to \infty, \text{ with } \overline{f}(z'_\alpha) \neq i.$$

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

This also means that there exists a $\beta' \in \mathbb{N}$ such that $\|x' - z'_\alpha\|_p < h^p_{\bar{f}}(x') + \epsilon/2$, $\forall \alpha > \beta'$, hence

$$h^p_{\bar{f}}(x_m) \leq \|x_m - z'_\alpha\|_p \leq \|x_m - x'\|_p + \|x' - z'_\alpha\|_p < h^p_{\bar{f}}(x') + \epsilon \quad \forall \alpha > \beta' \text{ and } m > l.$$

Notice that since $f(x_m) = i$, we also have a sequence $\{z_\alpha\}_{\alpha=0}^{\infty}$ such that

$$\|x_m - z_\alpha\|_p \to h^p_{\bar{f}}(x_m) \quad \text{as } \alpha \to \infty,$$

$\forall m > l$. This also means that there exists a $\beta \in \mathbb{N}$ such that

$$\|x_m - z_\alpha\|_p < h^p_{\bar{f}}(x_m) + \epsilon/2 \quad \forall \alpha > \beta \text{ and } m > l,$$

hence

$$h^p_{\bar{f}}(x') \leq \|x' - z_\alpha\|_p \leq \|x_m - x'\|_p + \|x_m - z_\alpha\|_p < h^p_{\bar{f}}(x_m) + \epsilon \quad \forall \alpha > \beta \text{ and } m > l.$$

Putting these together, we obtain $|h^p_{\bar{f}}(x') - h^p_{\bar{f}}(x_m)|_p < \epsilon \quad \forall m > l, \epsilon > 0$. Thus $h^p_{\bar{f}}(x_m) \to h^p_{\bar{f}}(x')$ as $m \to \infty$ and therefore $H_i(x_m) \to H_i(x')$ as $m \to \infty$.

Case 2: $\exists j \in \mathbb{N}$ *such that* $\bar{f}(x_m) \neq i$, $\forall m > j$. In this case $h^p_{\bar{f}}(x') = 0$, since the subsequence has only points containing points that do not map to label i, whereas $\bar{f}(x') = i$. Similarly, $\|x_m - x'\|_p$ serves as an upper bound for $h^p_{\bar{f}}(x_m)$ for all $m > j$, but since $x_m \to x'$ as $m \to \infty$, we must also have $h^p_{\bar{f}}(x_m) \to h^p_{\bar{f}}(x')$.

Case 3: $\forall j \in \mathbb{N} \quad \exists m, l > j$ *such that* $\bar{f}(x_m) = i$ *and* $\bar{f}(x_l) \neq i$. In this case, there exists a subsequence $\{x_{h_k}\}_{k=1}^{\infty}$ such that $\bar{f}(x_{h_k}) \neq i$ for all $k \in \mathbb{Z}$ and $x_{h_k} \to x'$ as $k \to \infty$. This means that $h^p_{\bar{f}}(x') = 0$. To show that $h^p_{\bar{f}}(x_m) \to 0$ as $m \to \infty$, we use the fact that the sequence is also a Cauchy sequence, and that elements that map to label i and ones that do not map to label i occur infinitely many times in the sequence.

Combining these gives us $H_i(x_m) \to H_i(x')$ as $m \to \infty$ as required.

With this lemma, we are now ready to prove our first main result Lemma 2.1.

Proof of Theorem 2.1. The proof will rely on two steps. First, we show that we can find a continuous function $g : \mathcal{K} \to [0, 1]^q$ that satisfies

$$p_q \circ g(x) = f(x) \quad \forall x \in \mathcal{M}_\epsilon \cap \mathcal{K}.$$

Then, we apply the corresponding form of the universal approximation theorem to find an approximator, which we will show will also be an interpolator.

By the lemma 6.3, we know that $H_i : \mathcal{K} \to \mathbb{R}^q$ (defined in Eq. (6.1)) are all continuous; hence, we can proceed to define the following vector valued function $H : \mathcal{K} \to \mathbb{R}^q$

$$H(x) = (H_1(x), H_2(x), \ldots, H_q(x)), \tag{6.2}$$

which must be continuous. Note that $p_q \circ H(x) = \bar{f}(x)$ *for* $x \in \mathcal{M}_\epsilon$. As our activation function is a continuous non-polynomial, we can apply the universal approximation theorem [62] on the function H. This guarantees us a single layer NN $\Psi : \mathcal{K} \to \mathbb{R}^q$ such that $\sup_{x \in \mathcal{K}} \|H(x) - \Psi(x)\|_p < \epsilon/2$. We will show that

$$p_q \circ \Psi(x) = \bar{f}(x) \quad \forall x \in \mathcal{M}_\epsilon \cap \mathcal{K}. \tag{6.3}$$

Observe that on the sets $\mathcal{M}_\epsilon$ the function H is of the form $H(x) = \lambda * e_{\bar{f}(x)}$ where $\lambda \in \mathbb{R}$, $\lambda > \epsilon$ and $e_k \in \mathbb{R}^q$ is a k'th unit vector. Therefore, $\Psi(x) = (\psi_1(x), \psi_2(x), \ldots, \psi_q(x))$ such that

$$\psi_i(x) < \epsilon/2 \quad \text{if } i \neq \bar{f}(x), \quad \psi_i(x) > \epsilon/2 \quad \text{if } i = \bar{f}(x).$$

The result (6.3) follows immediately from this. This proves part (2.6).

For the (2.7), we recall Theorem 6.2. As our activation function was is non-polynomial, therefore, it must also be non-affine, it satisfies all the conditions of Theorem 6.2 and the rest proceeds as in the shallow network case. □

Remark 6.4. There are slightly stronger versions of this theorem. If the activation function is only continuous and non-polynomial, then there exists a shallow NN that interpolates f on $\mathcal{M}$. On the other

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

hand, if the activation function is non-affine continuous that is continuously differentiable at at least one point, with non-zero derivative at that point, then there exists a deep NN with finite with that interpolates f on $\mathcal{M}$.

An interesting note here is that one can notice that the function H is in fact 1-Lipschitz, so the proof also shows that there exists a NN that is stable in the Lipschitz framework. The caveat, however, is that in practice, the loss function is minimising the difference between Ψ and $\bar{f}$, not $p_q \circ \Psi$ with $\bar{f}$, which means that the algorithms usually do not converge at H.

Proposition 6.5. *For the norm $\|\cdot\|_p$ where $1 \le p \le \infty$, the function $H : \mathbb{R}^d \to \mathbb{R}^q$ has Lipschitz constant 1.*

Proof. We want to show that $\|H(x) - H(y)\|_p \le \|x - y\|_p$. Recall that H is defined as the vector that consists of H_i Eq. (6.2). From the Eq. (6.1), we see that $H(x)$ will have elements equal to 0, unless the index i is equal to $\bar{f}(x)$. Given this, we can distinguish two cases.

Case 1. $\bar{f}(x) = \bar{f}(y)$ We know that there is a sequence $\{z_i\}_{i=1}^{\infty}$ such that

$$\|y - z_i\|_p \to h_{\bar{f}}^p(y), \quad \text{where } \bar{f}(z_i) \neq \bar{f}(y). \tag{6.4}$$

Furthermore, $\|x - z_i\|_p \ge h_{\bar{f}}^p(x)$, as $\bar{f}(x) = \bar{f}(y)$. Without the loss of generality let us assume that $h_{\bar{f}}^p(x) \ge h_{\bar{f}}^p(y)$. Since x, y have the same label, we obtain from (6.4) that for any $\epsilon > 0$

$$\begin{aligned} \|H(x) - H(y)\|_p &= |h_{\bar{f}}^p(x) - h_{\bar{f}}^p(y)| = h_{\bar{f}}^p(x) - h_{\bar{f}}^p(y) \\ &\le \|x - z_i\|_p - \|y - z_i\|_p + \epsilon \le \|x - y\|_p + \epsilon \quad \forall i \in \mathbb{N}. \end{aligned}$$

Taking $\epsilon \to 0$, we obtain the desired result.

Case 2. $\bar{f}(x) \neq \bar{f}(y)$ In this case, let us look at the line segment

$$\mathcal{L} = \{tx + (1-t)y : t \in [0,1]\},$$

and consider the following two points w_1, w_2

$$w_1 = t_1 x + (1 - t_1)y \quad t_1 = \inf\{t : \bar{f}(tx + (1-t)y) \neq \bar{f}(y)\}, \tag{6.5}$$

$$w_2 = t_2 x + (1 - t_2)y \quad t_2 = \sup\{t : \bar{f}(tx + (1-t)y) \neq \bar{f}(x)\}. \tag{6.6}$$

By linearity, we have $\frac{w_1+w_2}{2} = \frac{t_1+t_2}{2}x + (1 - \frac{t_1+t_2}{2})y$. Clearly $t_1 \le t_2$, because otherwise $t_2 < \frac{t_1+t_2}{2} < t_1$ and by the definitions (6.5) , (6.6)

$$\bar{f}\left(\frac{w_1+w_2}{2}\right) = \bar{f}\left(\frac{t_1+t_2}{2}x + (1 - \frac{t_1+t_2}{2})y\right) = \bar{f}(y) \quad \text{as } \frac{t_1+t_2}{2} < t_1,$$

$$\bar{f}\left(\frac{w_1+w_2}{2}\right) = \bar{f}\left(\frac{t_1+t_2}{2}x + (1 - \frac{t_1+t_2}{2})y\right) = \bar{f}(x) \quad \text{as } \frac{t_1+t_2}{2} > t_2.$$

This is a contradiction with $\bar{f}(x) \neq \bar{f}(y)$. Therefore, $t_1 \le t_2$ and hence

$$\begin{aligned} \|H(x) - H(y)\|_p &= (|h_{\bar{f}}^p(x)|^p + |h_{\bar{f}}^p(y)|^p)^{1/p} \le (|\|x - w_1\|_p|^p + |\|y - w_2\|_p|^p)^{1/p} \\ &\le \|x - w_1\|_p + \|y - w_2\|_p \le \|x - y\|_p. \end{aligned}$$

□

Note that we could have also proven the theorem using Urysohn's lemma, and we would obtain the same result. Using Urysohn's lemma, we would construct a continuous function $H^* : \mathcal{K} \to \mathbb{R}^q$ such that $p_q \circ H^*(x) = f(x)$, for all $x \in \mathcal{M}_\epsilon \cap \mathcal{K}$. This would be done by applying Urysohn's lemma for indicator functions $\mathbb{1}_i : \mathcal{K} \to \{0, 1\}$ for each label $i \in \overline{\mathcal{Y}}$

$$\mathbb{1}_i(x) = \begin{cases} 1 & \text{if } f(x) = i, \\ 0 & \text{if } f(x) \neq i. \end{cases}$$

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

on disjoint subsets of $\mathcal{M}_\epsilon$, call this function obtained from Urysohn's lemma $U_i : \mathcal{K} \to [0, 1]$. Then, the final function H^* would simply just be $H^*(x) = (U_1(x), U_2(x), \ldots, U_q(x))$. The drawback here is that this function does not necessarily have a bounded Lipschitz constant. In the following examples, we will illustrate that there are certain cases where the two functions H and H^* have different Lipschitz constants, yet their class stability is the same.

Example 6.6. *Consider the classification function* f_l:$[0, 2] \to \{0, 1\}$ *where*

$$f_l = \begin{cases} 0 & \text{if } x < 1, \\ 1 & \text{if } x \geq 1. \end{cases}$$

The $\mathcal{M}_\epsilon$ *set for* $\epsilon < 1$ *here would therefore be the set* $[0, 1 - \epsilon) \cup (1 + \epsilon, 2]$. *As we have shown in Lemma 6.5, the function H will always have a Lipschitz constant of 1. However, the function* H^* *will satisfy*

$$H^*(x) = \begin{cases} (1, 0) & \text{if } x < 1 - \epsilon, \\ (0, 1) & \text{if } x > 1 + \epsilon. \end{cases}$$

This means that we have a lower bound on the Lipschitz constant L by

$$L \geq \frac{\|(1, -1)\|_p}{2\epsilon}.$$

As this expression diverges as $\epsilon \to 0$, *we see that the Lipschitz constant diverges as well. However, for both functions, we have*

$$p_q \circ H(x) = p_q \circ H^*(x) = f_l(x) \quad \forall x \in \mathcal{M}_\epsilon.$$

Thus, $p_q \circ H$ *and* $p_q \circ H^*$ *have the same class stability.*

7. Stability revised

One relevant question one might have when talking about the class stability is how that relates to measure theory. In fact, if we were to look at the class stability from that point of view, one might argue that of the functions mentioned in Section 3, function f_3 might be considered the most stable and f_1, f_2 equally stable since the unstable points have measure 0. We can define the class stability in the following sense to keep consistency.

Definition 7.1 (Measure theoretic distance to the decision boundary). *For an extension of a classification function* $\bar{f} : \mathbb{R}^d \to \overline{\mathcal{Y}}$ *and a real number* $p \geq 1$, *we define* $\tau^p_{\bar{f}} : \mathbb{R}^d \to \mathbb{R}^+$ *the* l^p*-distance to the decision boundary as*

$$\tau^p_{\bar{f}}(x) = \inf \left\{ r : \int_{\mathcal{B}^p_r(x)} \mathbb{1}_{\bar{f}(z) = \bar{f}(x)} \, d\mu \neq \int_{\mathcal{B}^p_r(x)} d\mu, r \in [0, \infty) \right\}.$$

Here, μ *denotes the Lebesgue measure and* $\mathcal{B}^p_r(x)$ *the unit closed ball with p-norm, and* $\mathbb{1}$ *is the indicator function.*

Correspondingly, we can define the class stability in the following way.

Definition 7.2 (Class stability (measure theoretic)). *If* $\mathcal{M} \subset \mathbb{R}^d$ *is a compact set, we define the (measure theoretic)* ℓ_p*-stability of* $\bar{f}$ *to be*

$$\mathcal{T}^p_{\mathcal{M}}(\bar{f}) = \int_{\mathcal{M}} \tau^p_{\bar{f}}(x) \, d\mu. \tag{7.1}$$

Remark 7.3 (Properties of the measure theoretic distance to the decision boundary). One unfortunate thing for this definition is that the function is no longer continuous as can be seen by looking at the following function f_2 at the point $1/2$. The stability of that point is 0, whereas now its neighbourhood

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

has a non-zero stability as $1/2$ is an isolated point with a different label. Fortunately, we can show that the stability remains measurable if f itself was measurable.

Lemma 7.4 (Measurability of stability). *Let $f : \mathcal{M} \to \mathcal{Y}$ be a measurable classification function. Then, the measure theoretic distance to the decision boundary τ_f^p is measurable.*

Proof. To show that τ_f^p is measurable, it suffices to show that for every real number $\alpha \geq 0$, the set $\{x \in \mathcal{M} : \tau_f^p(x) < \alpha\}$ is measurable. We will show this by showing that the set $\{x \in \mathcal{M} : \tau_f^p(x) < \alpha\}$ is a countable union of measurable sets. Let $\alpha \geq 0$ be fixed. Then, we know that

$$\{x \in \mathcal{M} : \tau_f^p(x) < \alpha\} = \bigcup_{q \in \mathbb{Q}, 0 \leq q < \alpha} \{x \in \mathcal{M} : \mu\left(\mathcal{B}_q^p(x) \cap \{z \in \mathbb{R}^d : \bar{f}(z) \neq \bar{f}(x)\}\right) > 0\}. \tag{7.2}$$

Therefore, all we need to show is that the function $\phi_q(x) = \mu\left(\mathcal{B}_q^p(x) \cap \{z \in \mathbb{R}^d : \bar{f}(z) \neq \bar{f}(x)\}\right)$ is measurable for every non-negative $q \in \mathbb{Q}$. Clearly for $q = 0$, the function is constant and hence measurable. Hence, we will only consider $q > 0$. The function ϕ_q can be rewritten as a integral:

$$\phi_q(x) = \int_{\mathbb{R}^d} \mathbb{1}_{\mathcal{B}_q^p(x)}(z) \mathbb{1}_{\{z \in \mathbb{R}^d : \bar{f}(z) \neq \bar{f}(x)\}}(z)\, \mathrm{d}z. \tag{7.3}$$

We will finish of the proof by showing that the integrand is measurable with respect to the product σ-algebra $\sigma(\mathbb{R}^d) \otimes \sigma(\mathbb{R}^d)$, as the measurability of ϕ_q follows by Fubini's theorem [72]. We will look at the two parts of the integrand separately. In both cases, we will show that the underlying set of the indicator function is measurable.

For the first term is the indicator function of the set $A = \{(x, z) \in \mathbb{R}^d \times \mathbb{R}^d : z \in \mathcal{B}_q^p(x)\}$. This set is measurable as it is the preimage of $(-\infty, q]$ under the continuous (therefore measurable) function $h : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$ given by $h(x, z) = \|z - x\|_p$.

The second term is the indicator function of the set $B = \{(x, z) \in \mathbb{R}^d \times \mathbb{R}^d : \bar{f}(z) \neq \bar{f}(x)\}$. This set can be written as the finite union of sets

$$B = \bigcup_{i,j \in \overline{\mathcal{Y}}, i \neq j} \{(x, z) \in \mathbb{R}^d \times \mathbb{R}^d : \bar{f}(x) = j \text{ and } \bar{f}(z) = i\}$$

For each label $k \in \overline{\mathcal{Y}}$, let $C_k = \{x \in \mathbb{R}^d : \bar{f}(x) = k\}$. Since the classification function $\bar{f}$ is measurable, each set C_k is measurable in $\mathbb{R}^d$. Therefore, the set $\{(x, z) : \bar{f}(x) = y_i \text{ and } \bar{f}(z) = y_j\}$ is simply the Cartesian product $C_i \times C_j$, which is measurable in the product σ-algebra. Since B is a finite union of such measurable sets, it is measurable. Therefore, the integrand is measurable with respect to the product σ-algebra $\sigma(\mathbb{R}^d) \otimes \sigma(\mathbb{R}^d)$, and hence the function ϕ_q is measurable. □

For the rest of the document, we will always assume f to be measurable.

8. Proof of Lemma 2.4

We are now set to prove our next main result Lemma 2.4. To prove this theorem, we will first show the following theorem.

Proposition 8.1. *Let $f : \mathcal{M} \to \mathcal{Y}$ be a measurable classification function. Then, for any set of pairs $\{(x_i, f(x_i))\}_{i=1}^k$ such that $\tau_f^p(x_i) > 0$ for all $i = 1, \ldots, k$ (the distance to the decision boundary Eq. (2.2) is non-zero) and $\epsilon_1, \epsilon_2 > 0$, there exists a continuous function $g : \mathcal{M} \to \mathbb{R}$ such that the class stability Eq. (2.3) satisfies*

$$\mathcal{T}_{\mathcal{M}}^p(\overline{\lfloor g \rceil}) \geq \mathcal{T}_{\mathcal{M}}^p(\bar{f}) - \epsilon_1 \tag{8.1}$$

and the functions agree on the set $\{x_i\}_{i=1}^k$, i.e.:

$$f(x_i) = g(x_i) \quad i = 1, \ldots, k, \tag{8.2}$$

and

$$\mu(R) < \epsilon_2, \quad R := \{x \,|\, f(x) \neq g(x), x \in \mathcal{M}\}, \tag{8.3}$$

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

where μ denotes the Lebesgue measure and $\lfloor\cdot\rceil$ is the function that rounds to the nearest integer.

Note that the class stability of $\lfloor g \rceil$ is well defined as it is a discrete function defined on a compact set $\mathcal{M}$.

Proof of Lemma 8.1. We define the following disjoint sets, based on the distance to the decision boundary function Lemma 7.1: For $\xi > 0$, let

$$S_\xi := \{x \mid \tau_{\bar{f}}^p(x) \geq \xi, x \in \mathcal{M}\}, \quad U_\xi := \{x \mid \tau_{\bar{f}}^p(x) < \xi, x \in \mathcal{M}\},$$
$$U := \{x | \tau_{\bar{f}}^p(x) = 0\,, x \in \mathcal{M}\}.$$

First, notice that for any $\xi_1 < \xi_2$, we have $U_{\xi_1} \subset U_{\xi_2}$ and that for any $\eta > 0$ the following holds true

$$\bigcap_{\xi < \eta} U_\xi = U. \tag{8.4}$$

Since $\tau_{\bar{f}}^p$ is measurable and we can write $U = \{x \mid \tau_{\bar{f}}^p(x) \leq 0\}$ as $\tau_{\bar{f}}^p$ is non-negative, we know that the set U is measurable. In fact, by the same reasoning, all three sets are.

Consider the closure $\overline{S_\xi}$ of the set S_ξ, and the adjusted sets $U'_\xi = U_\xi - \overline{S_\xi}$ and $U^0_\xi = U - \overline{S_\xi}$. As $\overline{S_\xi}$ is closed, it must be measurable and also the difference of two measurable sets is measurable, thus $\overline{S_\xi}$, U'_ξ, U^0_ξ are all measurable.

Claim 1: $\mu(U \cap \overline{S_\xi}) = 0$. To show the claim, we will start by considering the collection $\{B^p_{\xi/2}(x) \mid x \in S_\xi\}$ of open balls or radius ξ in the p-norm, and noting that it is an open cover of $\overline{S_\xi}$. Therefore, since $\overline{S_\xi} \subset \mathcal{M}$, which is bounded, and since $\overline{S_\xi}$ is closed, there must exist a finite subcover, in particular there must exist a finite subset $S^* \subset S_\xi$ such that $\overline{S_\xi} \subset \bigcup_{x \in S^*} B^p_{\xi/2}(x)$. Now, suppose that $\mu(U \cap \overline{S_\xi}) > 0$, then we would neccesarily have

$$\mu(U \cap (\bigcup_{x \in S^*} B^p_{\xi/2}(x))) > 0, \text{ hence } \mu(\bigcup_{x \in S^*} (U \cap B^p_{\xi/2}(x))) > 0. \tag{8.5}$$

By subadditivity (as S^* is finite), there must exist a point x_0 such that $\mu(U \cap B^p_{\xi/2}(x_0)) > 0$. Recall that $x_0 \in S_\xi$ means $\tau_{\bar{f}}^p(x_0) \geq \xi$ which implies

$$\inf\{r \in [0, \infty) \mid \int_{\mathcal{B}^p_r(x_0)} \mathbb{1}_{\bar{f}(z) = \bar{f}(x_0)}\, d\mu \neq \int_{\mathcal{B}^p_r(x_0)} d\mu\} \geq \xi. \tag{8.6}$$

Thus, the function $\bar{f}$ is constant on $B^p_{\xi/2}(x_0)$ almost everywhere and any point z of the set

$$L_{x_0,\xi/2} := \{z \mid z \in B^p_{\xi/2}(x_0), \bar{f}(z) = \bar{f}(x_0)\} \tag{8.7}$$

satisfies $\tau_{\bar{f}}^p(z) \geq \xi/2$ as x_0 satisfies $\tau_{\bar{f}}^p(x_0) \geq \xi$. This means that $\mu(U \cap L_{x_0,\xi/2}) = 0$ as all $z' \in U$ have $\tau_{\bar{f}}^p(z') = 0$. Finally, from the fact that $\bar{f}$ is constant on $B^p_{\xi/2}(x_0)$ almost everywhere, we must have $\mu(B^p_{\xi/2}(x_0) - L_{x_0,\xi/2}) = 0$, which means that we cannot have $\mu(U \cap B^p_{\xi/2}(x_0)) > 0$, giving us the required contradiction and we have shown Claim 1.

Claim 2: *$\bar{f}$ is continuous on S_ξ and there exists a unique continuous extension of $\bar{f}$ to $\overline{S_\xi}$.* We start by showing that $\bar{f}$ is continuous on S_ξ. For any $x_0 \in S_\xi$, consider the neighbourhood $B^p_{\xi/2}(x_0)$ as before and recall that $\bar{f}$ is constant on this ball almost everywhere, with the constant being $\bar{f}(x_0)$. Suppose now that there is a $z \in S_\xi \cap B^p_{\xi/2}(x_0)$ such that $\bar{f}(x_0) \neq \bar{f}(z)$. As $z \in S_\xi$ (recall (8.6)), we must also have that $\bar{f}$ constant on $B^p_{\xi/2}(z)$ almost everywhere, with the constant being $\bar{f}(z)$. However, as $B^p_{\xi/2}(x_0)$ and $B^p_{\xi/2}(z)$ intersect, we obtain our contradiction. The second part of this claim follows a similar argument. Let x^* be a limit point of S_ξ. Consider the set $B^p_{\xi/2}(x^*) \cap S_\xi$. By arguing as in the first part of the proof of the claim, no two points in this set can have different labels. Thus, this means that any sequence $x_i \to x^*$ as $i \to \infty$ with $x_i \in S_\xi$ we have $x_i \in B^p_{\xi/2}(x^*) \cap S_\xi$ for all large i, and thus all the labels will eventually have to be the same. Therefore, there is a unique way of defining the extension of $\bar{f}$ to $\overline{S_\xi}$, which proves Claim 2. We will call this unique extension

$$\overline{f^*} : \overline{S_\xi} \to \overline{\mathcal{Y}}. \tag{8.8}$$

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

Claim 3: *Consider any $x_0 \in S_\xi$, and define $a = \tau^p_{\bar{f}}(x_0) - \xi$. We claim that $B^p_a(x_0) \subset \overline{S_\xi}$.* We first show that $\tau^p_{\bar{f}} \geq \xi$ on $B^p_a(x_0)$ almost everywhere for any fixed $x_0 \in S_\xi$. As before, it suffices to only consider the points $z \in B^p_a(x_0)$ such that $\bar{f}(z) = \bar{f}(x_0)$, as $\bar{f}$ is constant almost everywhere on this set. Suppose there exists $z \in L_{x_0,a}$ (as defined in Eq. (8.7)) such that $\tau^p_{\bar{f}}(z) < \xi$. The ball centred at x_0 with a radius $\|x_0 - z\|_p + \tau^p_{\bar{f}}(z)$ has to contain the ball centred at z with a radius of $\tau^p_{\bar{f}}(z)$. Thus, by the definition of the distance to the decision boundary, we must have $\tau^p_{\bar{f}}(x_0) \leq \|x_0 - z\|_p + \tau^p_{\bar{f}}(z) < a + \xi = \tau^p_{\bar{f}}(x_0)$, which gives the contradiction. Therefore, $\tau^p_{\bar{f}} \geq \xi$ on $B^p_a(x_0)$ almost everywhere and hence

$$L_{x_0,a} \subset S_\xi. \tag{8.9}$$

Now consider any $x \in B^p_a(x_0)$. Since the ball is open, there exists a $\delta_0 > 0$, such that $B^p_\delta(x) \subset B^p_a(x_0)$ for all $\delta < \delta_0$. Moreover, as $\mu(B^p_\delta(x)) > 0$ for any $\delta > 0$, there must be a sequence $\{x_i\}_{i=1}^\infty \subset L_{x_0,a}$ such that $x_i \to x$ as $i \to \infty$, as $L_{x_0,a} \subset B^p_a(x_0)$ and $\mu(B^p_a(x_0) - L_{x_0,a}) = 0$. This means that $x \in \overline{L_{x_0,a}}$ the closure of $L_{x_0,a}$ and from Eq. (8.9) we obtain $x \in \overline{S_\xi}$ for all $x \in B^p_a(x_0)$. Therefore $B^p_a(x_0) \subset \overline{S_\xi}$ which proves Claim 3.

Claim 4: $\mu(\overline{S_\xi} - S_\xi) = 0$. To see this, we first show that for any $x \in \overline{S_\xi} - S_\xi$ we have $\tau^p_{\bar{f}}(x) = 0$. Since $x \notin S_\xi$, we must have $\tau^p_{\bar{f}}(x) < \xi$. Suppose $\tau^p_{\bar{f}}(x) = \kappa$, where $\xi > \kappa > 0$. From the definition of the measure theoretic distance to the decision boundary, we have that

$$\inf\{r \in [0,\infty) \mid \int_{\mathcal{B}^p_r(x)} \mathbb{1}_{\bar{f}(z)=\bar{f}(x)}\, d\mu \neq \int_{\mathcal{B}^p_r(x)} d\mu\} = \kappa > 0. \tag{8.10}$$

As a consequence, we must have

$$\int_{\mathcal{B}^p_{\frac{1}{2}\kappa}(x)} \mathbb{1}_{\bar{f}(z)=\bar{f}(x)}\, d\mu = \int_{\mathcal{B}^p_{\frac{1}{2}\kappa}(x)} d\mu. \tag{8.11}$$

Furthermore, since $x \in \overline{S_\xi}$ there must be a sequence $\{x_i\}_{i=1}^\infty \subset S_\xi$ such that $x_i \to x$ as $i \to \infty$. Pick an $j \in \mathbb{N}$,such that $x_j \in \mathcal{B}^p_{\frac{1}{2}\kappa}(x)$. Then, by the definition of the measure theoretic distance to the decision boundary, we must have that $\tau^p_{\bar{f}}(x_j) \geq \xi$. This means that

$$\int_{\mathcal{B}^p_{\frac{1}{2}\xi}(x_j)} \mathbb{1}_{\bar{f}(z)=\bar{f}(x_j)}\, d\mu = \int_{\mathcal{B}^p_{\frac{1}{2}\xi}(x_j)} d\mu. \tag{8.12}$$

However, as $x_j \in \mathcal{B}^p_{\frac{1}{2}\kappa}(x)$, we must have that $\mathcal{B}^p_{\frac{1}{2}\xi}(x_j) \cap \mathcal{B}^p_{\frac{1}{2}\kappa}(x) \neq \emptyset$. Combining this with the fact that $\mathbb{1}_{\bar{f}(z)=\bar{f}(x)} + \mathbb{1}_{\bar{f}(z)=\bar{f}(x_j)} \leq 1$, we must have that

$$\begin{aligned}
\int_{\mathcal{B}^p_{\frac{1}{2}\xi}(x_j)\cap\mathcal{B}^p_{\frac{1}{2}\kappa}(x)} d\mu &\geq \int_{\mathcal{B}^p_{\frac{1}{2}\xi}(x_j)\cap\mathcal{B}^p_{\frac{1}{2}\kappa}(x)} \mathbb{1}_{\bar{f}(z)=\bar{f}(x)} + \mathbb{1}_{\bar{f}(z)=\bar{f}(x_j)}\, d\mu \\
&= \int_{\mathcal{B}^p_{\frac{1}{2}\xi}(x_j)\cap\mathcal{B}^p_{\frac{1}{2}\kappa}(x)} \mathbb{1}_{\bar{f}(z)=\bar{f}(x)}\, d\mu + \int_{\mathcal{B}^p_{\frac{1}{2}\xi}(x_j)\cap\mathcal{B}^p_{\frac{1}{2}\kappa}(x)} \mathbb{1}_{\bar{f}(z)=\bar{f}(x_j)}\, d\mu \\
&= 2\int_{\mathcal{B}^p_{\frac{1}{2}\xi}(x_j)\cap\mathcal{B}^p_{\frac{1}{2}\kappa}(x)} d\mu.
\end{aligned} \tag{8.13}$$

As the $\int_{\mathcal{B}^p_{\frac{1}{2}\xi}(x_j)\cap\mathcal{B}^p_{\frac{1}{2}\kappa}(x)} d\mu > 0$, we obtain our contradiction. Hence, $\tau^p_{\bar{f}}(x) = 0$ for all $x \in \overline{S_\xi} - S_\xi$. This is equivalent to saying that for any $x \in \overline{S_\xi} - S_\xi$, we have $x \in U$. Therefore, for any $x \in \overline{S_\xi} - S_\xi$, we have $x \in U \cap \overline{S_\xi}$, which by Claim 1 implies that $\mu(\overline{S_\xi} - S_\xi) = 0$. This proves Claim 4.

Next, we apply Lusin's Theorem for the function $\bar{f}$ on the set U^0_ξ and obtain, for any $\alpha > 0$, a closed set $U^\alpha_\xi \subset U^0_\xi$ such that

$$\mu(U^0_\xi - U^\alpha_\xi) < \alpha, \quad \bar{f} \text{ is continuous on } U^\alpha_\xi. \tag{8.14}$$

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

We can now define $g_{\alpha,\xi} : \overline{S_\xi} \cup U_\xi^\alpha \to [a, b]$, where $a := \min\{\mathcal{Y}\}$ and $b := \max\{\mathcal{Y}\}$, where

$$g_{\alpha,\xi}(x) = \begin{cases} \overline{f^*}(x) & \text{if } x \in \overline{S_\xi}, \\ \overline{f}(x) & \text{if } x \in U_\xi^\alpha. \end{cases}$$

Finally, as both sets $\overline{S_\xi}$ and U_ξ^α are compact, since they are closed and subsets of $\mathcal{M}$, which is compact, we can apply Tietze's extension theorem. More precisely, we will use Tietze's extension theorem to extend the restriction of the function $g_{\alpha,\xi} : \overline{S_\xi} \cup U_\xi^\alpha \to [a, b]$, to a continuous function on the whole set $\mathcal{M}$. Then, by Tietze's extension theorem, we obtain a continuous function $g^*_{\alpha,\xi} : \mathcal{M} \to [a, b]$ such that

$$g^*_{\alpha,\xi}(x) = g_{\alpha,\xi}(x) \quad x \in \overline{S_\xi} \cup U_\xi^\alpha.$$

Having constructed the function, all we need to do is to check that the properties (8.1) (8.2) and (8.3) are satisfied for some particular choices of α and ξ. Let us first estimate the loss in class stability for the rounded function $\lfloor g^*_{\alpha,\xi} \rceil$. For any fixed ξ, we can bound the stability by:

$$\mathcal{T}^p_{\lfloor g^*_{\alpha,\xi} \rceil} = \int_{\mathcal{M}} \tau^p_{\lfloor g^*_{\alpha,\xi} \rceil}\, d\mu = \int_{\overline{S_\xi} \cup U'_\xi} \tau^p_{\lfloor g^*_{\alpha,\xi} \rceil}\, d\mu.$$

We know that $\overline{f^*}$ (defined in Eq. (8.8)) and $g^*_{\alpha,\xi}$ agree on $\overline{S_\xi}$, hence $\lfloor g^*_{\alpha,\xi} \rceil$ agrees with $\overline{f^*}$ as well. From Claim 3, we know that for any point $x_0 \in S_\xi$, $B^p_a(x_0) \subset \overline{S_\xi}$, where $a = \tau^p_{\overline{f}}(x_0) - \xi$, while from Claim 2, we know that $\overline{f^*}$ is continuous on $\overline{S_\xi}$, therefore $\overline{f^*}$ is constant on $B^p_a(x_0)$ as it is a discrete function. Thus, we must have $\tau^p_{\lfloor g^*_{\alpha,\xi} \rceil}(x_0) \geq \tau^p_{\overline{f}}(x_0) - \xi$ for all $x_0 \in S_\xi$. This means that

$$\begin{aligned} \mathcal{T}^p_{\lfloor g_{\alpha,\xi} \rceil} &= \int_{\overline{S_\xi} \cup U'_\xi} \tau^p_{\lfloor g_{\alpha,\xi} \rceil}\, d\mu \geq \int_{S_\xi \cup U'_\xi} \tau^p_{\lfloor g_{\alpha,\xi} \rceil}\, d\mu \geq \int_{S_\xi} \tau^p_{\overline{f}} - \xi\, d\mu \\ &= \int_{\mathcal{M} - U_\xi} \tau^p_{\overline{f}}\, d\mu - \xi\mu(S_\xi) = \mathcal{T}^p(f) - \int_{U_\xi} \tau^p_{\overline{f}}\, d\mu - \xi\mu(S_\xi) \\ &> \mathcal{T}^p(f) - \xi\mu(U_\xi) - \xi\mu(S_\xi) = \mathcal{T}^p(f) - \xi\mu(\mathcal{M}). \end{aligned}$$

The last inequality comes from the fact that $\tau^p_{\overline{f}}(x) < \xi$ for $x \in U_\xi$. By choosing $\xi \leq \frac{\epsilon_1}{\mu(\mathcal{M})}$, we obtain Eq. (8.1).

To ensure (8.2), we simply need to guarantee that the set $\{x_i\}_{i=1}^k$, from the statement of the proposition, satisfies $\{x_i\}_{i=1}^k \subset S_\xi$. This can be achieved by choosing $\xi < \min_{i=1,\ldots,k}\{\tau^p_{\overline{f}}(x_i)\}$.

Finally, we observe that $R \subset \left(U'_\xi - U^\alpha_\xi\right) + \left(\overline{S_\xi} - S_\xi\right)$, where we recall R from Eq. (8.3). Therefore, we have

$$\begin{aligned} \mu(R) &\leq \mu\left(U'_\xi - U^\alpha_\xi\right) + \mu\left(\overline{S_\xi} - S_\xi\right) = \mu\left(U'_\xi - U^\alpha_\xi\right) \leq \mu(U'_\xi - U^0_\xi) + \mu(U^0_\xi - U^\alpha_\xi) \\ &< \mu(U'_\xi - U^0_\xi) + \alpha = \mu((U_\xi - \overline{S_\xi}) - (U - \overline{S_\xi})) + \alpha = \mu(U_\xi - U) + \alpha. \end{aligned} \tag{8.15}$$

Thus, to establish Eq. (8.3), it suffices to show that $\mu(U_\xi) \to \mu(U)$ as $\xi \to 0$, and then by setting $\alpha = \epsilon_2/2$ we could choose a small enough ξ to finally obtain (8.3). Thankfully, this is true as we have shown that U_ξ is decreasing in ξ and since $U_\xi \subset \mathcal{M}$, we know that the measure $\mu(U_\xi) \leq \mu(\mathcal{M})$. Therefore, $\mu(U_\xi)$ is bounded and because of Eq. (8.4) we can apply Theorem 3.26 from [72] to obtain $\mu(U_\xi) \to \mu(U)$ as $\xi \to 0$. □

Proof of Lemma 2.4. Using Lemma 8.1, we construct a continuous function $g : \mathcal{M} \to \mathbb{R}$ that satisfies the conditions. Next, we construct a continuous function $G : \mathcal{M} \to \mathbb{R}^q$ such that

$$\mathcal{T}^p_{\mathcal{M}}(p_q(G)) \geq \mathcal{T}^p_{\mathcal{M}}(\bar{f}) - \epsilon_1, \tag{8.16}$$

we can interpolate on the set

$$p_q(G) = f(x_i) \quad i = 1, \ldots, k, \tag{8.17}$$

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

and

$$\mu(R) < \epsilon_2, \quad R := \{x \,|\, f(x) \neq p_q(G), x \in \mathcal{M}\}, \tag{8.18}$$

where μ denotes the Lebesgue measure. Recall from the proof of Lemma 8.1 that g is constant on $\overline{S_\xi} \cup U_\xi^\alpha$ for $\xi > 0$. Furthermore, from the proof it is clear that any function that agrees with g on the set $\overline{S_\xi} \cup U_\xi^\alpha$ will also have to satisfy all three conditions of the theorem. Therefore, it is enough to construct G such that $p_q(G)$ agrees with g on $\overline{S_\xi} \cup U_\xi^\alpha$. To construct the function G, consider the function $\omega : \mathbb{R} \to \mathbb{R}$ defined by

$$\omega_i(x) = \begin{cases} 0 & x \leq i-1, \\ x-(i-1) & i-1 < x \leq i, \\ (i+1)-x & i < x \leq i+1, \\ 0 & i+1 \leq x. \end{cases} \tag{8.19}$$

Having this, we can simply define $G(x) = (\omega_1(g(x)), \ldots, \omega_q(g(x)))$, which will be continuous as ω is continuous. Furthermore, $p_q(G)$ agrees with g on $\overline{S_\xi} \cup U_\xi^\alpha$ and thus satisfies all three conditions of the theorem. We now just need to apply the universal approximation theorem on the function G to obtain a NN $\psi : \mathcal{M} \to \mathbb{R}^q$ that differs from G in the uniform norm by less than $1/2$. This NN will give the same labels on $\overline{S_\xi} \cup U_\xi^\alpha$ as G and thus must satisfy all three conditions of the theorem, thereby completing the proof. □

9. Emprical estimation of the class stability

Having established the theoretical results, we conclude this paper with a discussion on how one might determine the class stability of a NN in practice. Both versions of the distance to the decision boundary (Eqs. (2.2) and (7.1)) are in practice extremely difficult to compute. To remedy this, we will propose an empirical method to estimate the class stability using a NN.

Instead of calculating the distance to the decision boundary, we can use adversarial attacks to estimate the distance to the decision boundary. More specifically, we can use adversarial attack algorithms to find the smallest perturbation that changes the label of a data point. This perturbation will then be an upper bound on the actual distance to the decision boundary. To highlight the fact that this estimate is contingent on the adversarial attack algorithm used, we will index the estimate with the name of the algorithm.

For the numerical examples, we will use the MNIST dataset and a few NNs with different architectures but similar performance. The models used are two custom networks, a fully connected network (FCNN) and a convolutional network (CNN), a ResNet18 [47] and a VGG16 [65]. The algorithms used to estimate the distance to the decision boundary are Fast Gradient Sign Method (FGSM) [41], DeepFool (DF) [56], Projected Gradient Descent (PGD) [54] and L-infinity Projected Gradient Descent (LinfPGD) [39]. The documentation for the code can be found at https://github.com/zhenningdavidliu/paper_measure_code.

The precise method to estimate the class stability is as follows.

(1) Select a problem (e.g. MNIST) and a NN (e.g. a VGG16).
(2) Train the NN on the problem.
(3) Select an adversarial attack algorithm (e.g. PGD).
(4) For each data point in the dataset, use the adversarial attack algorithm to find the smallest perturbation that changes the label of the data point.
(5) Use the perturbation to estimate the distance to the decision boundary.
(6) Take the sample mean of the estimated distances to obtain an estimate of the class stability.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

Table 1. *Stability and performance metrics for different models. We have tested two custom networks, a ResNet18 and a VGG16. The custom networks are simple implementations of a fully connected network and a convolutional network, respectively. The algorithms used to estimate the distance to the decision boundary are F: FGSM, D: DPG, P: PGD, and L: LinfPGD. The results suggests that VGG16 is the most stable model, according to the definition of class stability*

Model	Name (f_i)	Params	Accuracy	$\mathcal{S}^2_{\mathcal{M},F}(\bar{f}_i)$	$\mathcal{S}^2_{\mathcal{M},D}(\bar{f}_i)$	$\mathcal{S}^2_{\mathcal{M},P}(\bar{f}_i)$	$\mathcal{S}^2_{\mathcal{M},L}(\bar{f}_i)$
Custom FCNN	f_1	101,770	95.48 %	7.25	3.18	4.14	4.17
Custom CNN	f_2	1,625,866	98.67 %	19.65	4.28	4.92	4.95
ResNet18	f_3	11,181,642	97.97 %	5.90	3.27	4.09	4.31
VGG16	f_4	134,301,514	98.8 %	56.00	13.74	17.30	16.70

In other words, we will estimate $h^p_{\bar{f}}(x)$ by $h^p_{f,PGD}(x)$ for the PGD attack, where $h^p_{f,PGD}(x)$ is the empirical estimate of the distance to the decision boundary for the PGD attack for the data point x. We will then use this estimate to estimate the class stability by

$$\mathcal{S}^p_{\mathcal{M}}(\bar{f}) \approx \frac{1}{k}\sum_{i=1}^{k} h^p_{f,PGD}(x_i), \tag{9.1}$$

where k is the number of data points in the dataset. To have consistent notation for our tables, we will reference the empirical estimate of the class stability as $\mathcal{S}^p_{\mathcal{M},\Gamma}(\bar{f})$, where Γ is the name of the adversarial attack algorithm used. For example, $\mathcal{S}^p_{\mathcal{M},PGD}(\bar{f})$ is the empirical estimate of the class stability for the PGD attack.

9.1. Empirical estimation of class stability for neural networks

The empirical class stability provides a way to measure robustness of a model with respect to adversarial attacks. One of the main advantages of this approach is the simplicity of the method, as it only requires running existing adversarial attack algorithms on models, without the need for additional training or optimisation. To demonstrate this, we will use the MNIST dataset and a few NNs with different architectures but similar performance. We use several adversarial attack algorithms to estimate the distance to the decision boundary for each data point in the dataset. We then use the estimated distances to estimate the class stability using the method described above. Table 1 shows the performance and stability of the different models. The higher the score for the stability, the more stable the model is, as it is more difficult to find adversarial examples. The final column shows the minimum ϵ for the aggregate of all the adversarial attack algorithms we used. This is an estimate of the distance to the decision boundary, and thus the higher the score, the more stable the model is.

Funding Statement. ACH acknowledges support from the Simons Foundation Award No. 663281 granted to the Institute of Mathematics of the Polish Academy of Sciences for the years 2021-2023, from a Royal Society University Research Fellowship, and from the Leverhulme Prize 2017.

Competing interests. None.

References

[1] Adcock, B., Brugiapaglia, S., Dexter, N. & Morage, S. (2022) Deep neural networks are effective at learning high-dimensional hilbert-valued functions from limited data. In: Bruna, J., Hesthaven, J. & Zdeborova, L. (eds.) *Proceedings of the 2nd Mathematical and Scientific Machine Learning Conference, volume 145 of Proceedings of Machine Learning Research*. PMLR, pp. 1–36.

[2] Adcock, B. & Dexter, N. (2021) The gap between theory and practice in function approximation with deep neural networks. *SIAM J. Math. Data Sci.* **3**(2), 624–655.

[3] Adcock, B. & Hansen, A. C. (2021). *Compressive Imaging: Structure, Sampling, Learning*. Cambridge University Press.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

[4] Adcock, B. & Huybrechs, D. (2020) Approximating smooth, multivariate functions on irregular domains. *Forum Math., Sigma* **8**, e26.
[5] Akhtar, N. & Mian, A. (2018) Threat of adversarial attacks on deep learning in computer vision: A survey. *IEEE Access* **6**, 14410–14430.
[6] Antun, V., Renna, F., Poon, C., Adcock, B. & Hansen, A. C. (2020) On instabilities of deep learning in image reconstruction and the potential costs of ai. *Proc. Natl. Acad. Sci.* **117**(48), 30088–30095.
[7] Bastounis, A., Cucker, F. & Hansen, A. C. (2023) When can you trust feature selection? – i: A condition-based analysis of lasso and generalised hardness of approximation. arXiv: 2312.11425.
[8] Bastounis, A., Hansen, A. C. & Vlacic, V. (2021) The mathematics of adversarial attacks in AI – why deep learning is unstable despite the existence of stable neural networks. arXiv: 2109.06098.
[9] Bastounis, A., Hansen, A. C. & Vlačić, V. (2021) The extended Smale's 9th problem – On computational barriers and paradoxes in estimation, regularisation, computer-assisted proofs and learning. arXiv: 2110.15734.
[10] Beerens, L. & Higham, D. J. (2023) Adversarial ink: Componentwise backward error attacks on deep learning. *IMA J. Appl. Math.* hxad017.
[11] Belthangady, C. & Royer, L. A. (2019) Applications, promises, and pitfalls of deep learning for fluorescence image reconstruction. *Nat. Methods* **16**(12), 1215–1225.
[12] Ben-Artzi, J., Colbrook, M. J., Hansen, A. C., Nevanlinna, O. & Seidel, M. (2020) Computing spectra – On the solvability complexity index hierarchy and towers of algorithms. arXiv: 1508.03280.
[13] Ben-Artzi, J., Hansen, A. C., Nevanlinna, O. & Seidel, M. (2015) New barriers in complexity theory: On the solvability complexity index and the towers of algorithms. *C. R. Math.* **353**(10), 931–936.
[14] Béthune, L., González-Sanz, A., Mamalet, F. & Serrurier, M. (2021) The many faces of 1-lipschitz neural networks. *CoRR* **2104**(05097).
[15] Binev, P., Cohen, A., Dahmen, W., DeVore, R. & Temlyakov, V. (2005) Universal algorithms for learning theory part i : Piecewise constant functions. *J. Mach. Learn. Res.* **6**(44), 1297–1321.
[16] Bölcskei, H., Grohs, P., Kutyniok, G. & Petersen, P. (2019) Optimal approximation with sparsely connected deep neural networks. *SIAM J. Math. Data Sci.* **1**(1), 8–45.
[17] Bubeck, S. & Sellke, M. (2021) A universal law of robustness via isoperimetry. In *NeurIPS 2021*.
[18] Bungert, L., Trillos, N. García & Murray, R. (2023) The geometry of adversarial training in binary classification. *Inf. Inference: J. IMA* **12**(2), 921–968.
[19] Caragea, A., Petersen, P. & Voigtlaender, F. (2022) Neural network approximation and estimation of classifiers with classification boundary in a barron class. arXiv: 2011.09363.
[20] Carlini, N. & Wagner, D. (2018). Audio adversarial examples: Targeted attacks on speech-to-text. In *2018 IEEE Security and Privacy Workshops (SPW)*. IEEE, pp. 1–7.
[21] Celledoni, E., Ehrhardt, M. J., Etmann, C., et al. (2021) Structure-preserving deep learning. *Eur. J. Appl. Math.*, **32**, 1–49.
[22] Chambolle, A. (2004) An algorithm for total variation minimization and applications. *J. Math Imaging Vis.* **20**(1), 89–97.
[23] Chambolle, A. & Pock, T. (2011) A first-order primal-dual algorithm for convex problems with applications to imaging. *J. Math Imaging Vis.* **40**(1), 120–145.
[24] Choi, C. Q. (2021) 7 revealing ways ais fail: Neural networks can be disastrously brittle, forgetful, and surprisingly bad at math. *IEEE Spectrum* **58**(10), 42–47.
[25] Colbrook, M. (2024) On the computation of geometric features of spectra of linear operators on hilbert spaces. *Found. Comp. Math.* **24**(3), 723–804.
[26] Colbrook, M. & Hansen, A. C. (2023) The foundations of spectral computations via the solvability complexity index hierarchy. *J. Eur. Math. Soc.* **25**(12), 4639–4718.
[27] Colbrook, M. J., Antun, V. & Hansen, A. C. (2022) The difficulty of computing stable and accurate neural networks: On the barriers of deep learning and smale's 18th problem. *Proc. Natl. Acad. Sci.* **119**(12), e2107151119.
[28] Dahl, G. E., Yu, D., Deng, L. & Acero, A. (2011) Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition. *IEEE Trans. Audio Speech Lang. Process.* **20**(1), 30–42.
[29] Daubechies, I., DeVore, R., Dym, N., et al. (2023) Neural network approximation of refinable functions. *IEEE Trans. Inform. Theory* **69**(1), 482–495.
[30] Daubechies, I., DeVore, R., Foucart, S., Hanin, B. & Petrova, G. (2022) Nonlinear approximation and (deep) relu networks. *Constr. Approx.* **55**(1), 127–172.
[31] DeVore, R., Hanin, B. & Petrova, G. (2021) Neural network approximation. *Acta Numer.* **30**, 327–444.
[32] D'Inverno, G. A., Brugiapaglia, S. & Ravanelli, M. (2023) Generalization limits of graph neural networks in identity effects learning. arXiv preprint arXiv: 2307.00134.
[33] Ducotterd, S., Goujon, A., Bohra, P., Perdios, D., Neumayer, S. & Unser, M. (2022) Improving lipschitz-constrained neural networks by learning activation functions. arXiv: 2210.16222.
[34] Elbrächter, D., Perekrestenko, D., Grohs, P. & Bölcskei, H. (2021) Deep neural network approximation theory. *IEEE Trans. Inform. Theory* **67**(5), 2581–2623.
[35] Ferreira, A., Silva, L., Renna, F., et al. (2020) Deep learning-based methods for individual recognition in small birds. *Methods Ecol. Evol.* **11**, 07.
[36] Finlayson, S. G., Bowers, J. D., Ito, J., Zittrain, J. L., Beam, A. L. & Kohane, I. S. (2019) Adversarial attacks on medical machine learning. *Science* **363**(6433), 1287–1289.
[37] Franco, N. R. & Brugiapaglia, S. (2024) A practical existence theorem for reduced order models based on convolutional autoencoders. arXiv preprint arXiv: 2402.00435.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

[38] Gazdag, L. E. & Hansen, A. C. (2023) Generalised hardness of approximation and the SCI hierarchy – on determining the boundaries of training algorithms in AI. arXiv: 2209.06715.
[39] Geisler, S., Wollschläger, T., Abdalla, M. H. I., Gasteiger, J. & Günnemann, S. (2024) Attacking large language models with projected gradient descent. arXiv: 2402.09154.
[40] Girshick, R., Donahue, J., Darrell, T. & Malik, J. (2014) Rich feature hierarchies for accurate object detection and semantic segmentation. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 580–587.
[41] Goodfellow, I. J., Shlens, J. & Szegedy, C. (2015) Explaining and harnessing adversarial examples. International Conference on Learning Representations (ICLR).
[42] Gorban, A., Golubkov, A., Grechuk, B., Mirkes, E. & Tyukin, I. (2018) Correction of ai systems by linear discriminants: Probabilistic foundations. *Inform. Sci.* **466**, 303–322.
[43] Gottschling, N. M., Antun, V., Hansen, A. C. & Adcock, B. (2025) The troublesome kernel – on hallucinations, no free lunches and the accuracy-stability trade-off in inverse problems. *SIAM Review* **67**(1), 73–104.
[44] Gribonval, R., Kutyniok, G., Nielsen, M. & Voigtlaender, F. (2022) Approximation spaces of deep neural networks. *Constr. Approx.* **55**, 259–367.
[45] Hansen, A. C. (2011) On the solvability complexity index, the *n*-pseudospectrum and approximations of spectra of operators. *J. Am. Math. Soc.* **24**(1), 81–124.
[46] Hansen, A. C. & Nevanlinna, O. (2016) Complexity issues in computing spectra, pseudospectra and resolvents. *Banach Cent.* **112**, 171–194.
[47] He, K., Zhang, X., Ren, S. & Sun, J. (2016). Deep residual learning for image recognition. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778.
[48] Heaven, D. (2019) Why deep-learning AIs are so easy to fool. *Nature* **574**(7777), 163–166.
[49] Hinton, G. & and, etal (2012) Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups. *IEEE Signal Process. Mag.* **29**(6), 82–97.
[50] Hoffman, D. P., Slavitt, I. & Fitzpatrick, C. A. (2021) The promise and peril of deep learning in microscopy. *Nat. Methods* **18**(2), 131–132.
[51] Huang, Y., Zhang, H., Shi, Y., Kolter, J. Z. & Anandkumar, A. (2021). Training certifiably robust neural networks with efficient local lipschitz bounds. In: Ranzato, M., Beygelzimer, A., Dauphin, Y., Liang, P. & Vaughan, J. W. (eds.), *Advances in Neural Information Processing Systems*, Vol. **34**. Curran Associates, Inc., pp. 22745–22757.
[52] Kidger, P. & Lyons, T. (2020) Universal approximation with deep narrow networks. In: Abernethy, J. & Agarwal, S. (eds.), *Proceedings of 33rd Conference on Learning Theory, Volume 125 of Proceedings of Machine Learning Research* , Vol. **125**. PMLR, pp. 2306–2327.
[53] Kutyniok, G. (2022) The mathematics of artificial intelligence. arXiv: 2203.08890.
[54] Madry, A., Makelov, A., Schmidt, L., Tsipras, D. & Vladu, A. (2018). Towards deep learning models resistant to adversarial attacks. International Conference on Learning Representations (ICLR).
[55] Moosavi-Dezfooli, S., Fawzi, A., Fawzi, O. & Frossard, P. (2017) Universal adversarial perturbations. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1765–1773.
[56] Moosavi-Dezfooli, S., Fawzi, A. & Frossard, P. (2016) Deepfool: A simple and accurate method to fool deep neural networks. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2574–2582.
[57] Neyra-Nesterenko, M. & Adcock, B. (2022) Nestanets: Stable, accurate and efficient neural networks for analysis-sparse inverse problems. *Sampling Theory, Signal Processing, and Data Analysis* **21**(1), 4.
[58] Niyogi, P., Smale, S. & Weinberger, S. (2011) A topological view of unsupervised learning from noisy data. *SIAM J. Comput.* **40**(3), 646–663.
[59] Oliveira, J., Renna, F., Costa, P. D., et al. (2021) The circor digiscope dataset: From murmur detection to murmur classification. *IEEE J. Biomed. Health* **26**, 2524–2535.
[60] Perekrestenko, D., Grohs, P., Elbrächter, D. & Bölcskei, H. (2018) The universal approximation power of finite-width deep relu networks. arXiv: 1806.01528.
[61] Petersen, P. & Voigtlaender, F. (2018) Optimal approximation of piecewise smooth functions using deep relu neural networks. *Neural Netw.* **108**, 296–330.
[62] Pinkus, A. (1999) Approximation theory of the mlp model in neural networks. *Acta Numer.* **8**, 143–195.
[63] Qin, C., Martens, J., Gowal, S., et al. (2019) Adversarial robustness through local linearization. arXiv: 1907.02610.
[64] Raj, A., Bresler, Y. & Li, B. (2020). Improving robustness of deep-learning-based image reconstruction. In: *International Conference on Machine Learning*. PMLR, pp. 7932–7942.
[65] Simonyan, K. & Zisserman, A. (2015) Very deep convolutional networks for large-scale image recognition.
[66] Sokolić, J., Giryes, R., Sapiro, G. & Rodrigues, M. R. D. (2017) Robust large margin deep neural networks. *IEEE Trans. Signal Process.* **65**(16), 4265–4280.
[67] Szegedy, C., Zaremba, W., Sutskever, I., et al. (2014) Intriguing properties of neural networks. arXiv: 1312.6199.
[68] Tsipras, D., Santurkar, S., Engstrom, L., Turner, A. & Madry, A. (2019) Robustness may be at odds with accuracy. arXiv: 1805.12152.
[69] Tyukin, I. Y., Higham, D. J., Bastounis, A., Woldegeorgis, E. & Gorban, A. N. (2024) The feasibility and inevitability of stealth attacks. *IMA J. Appl. Math.* **89**(1), 44–84.
[70] Voigtlaender, F. (2023) The universal approximation theorem for complex-valued neural networks. *Appl. Comput. Harmon. A* **64**, 33–61.
[71] Wang, S., Si, N., Blanchet, J. & Zhou, Z. (2023) On the foundation of distributionally robust reinforcement learning. arXiv: 2311.09018.

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press

[72] Wheeden, R. (2015). *Measure and Integral: An Introduction to Real Analysis*, 2nd edn. Chapman & Hall/CRC Pure and Applied Mathematics. CRC Press.
[73] Yang, Y., Rashtchian, C., Zhang, H., Salakhutdinov, R. & Chaudhuri, K. (2020) Adversarial robustness through local lipschitzness. *CoRR*. arXiv: 2003.02460.
[74] Yarotsky, D. (2018). Optimal approximation of continuous functions by very deep relu networks. In: *Conference on Learning Theory*. PMLR, pp. 639–649.
[75] Zhang, B., Cai, T., Lu, Z., He, D. & Wang, L. (2021) Towards certifying l-infinity robustness using neural networks with l-inf-dist neurons neurons. *CoRR* arXiv: 2102.05363.
[76] Zhang, B., Jiang, D., He, D. & Wang, L. (2022). Rethinking lipschitz neural networks and certified robustness: A boolean function perspective. In: Oh, A. H., Agarwal, A., Belgrave, D. & Cho, K. (eds.), *Advances in Neural Information Processing Systems*.

Cite this article: Liu D. and Hansen A. (2026). Do stable neural networks exist for classification problems? – A new view on stability in AI. *European Journal of Applied Mathematics*, **37**, 238–258. https://doi.org/10.1017/S0956792525100181

https://doi.org/10.1017/S0956792525100181 Published online by Cambridge University Press