A Functional Perspective on Learning Symmetric Functions with Neural Networks

Aaron Zweig[a] and Joan Bruna *[a,b]

[a]Courant Institute of Mathematical Sciences, New York University, New York
[b]Center for Data Science, New York University

October 28, 2020

Abstract

Symmetric functions, which take as input an unordered, fixed-size set, are known to be universally representable by neural networks that enforce permutation invariance. These architectures only give guarantees for fixed input sizes, yet in many practical applications, including point clouds and particle physics, a relevant notion of generalization should include varying the input size. In this work we treat symmetric functions (of any size) as functions over probability measures, and study the learning and representation of neural networks defined on measures. By focusing on shallow architectures, we establish approximation and generalization bounds under different choices of regularization (such as RKHS and variation norms), that capture a hierarchy of functional spaces with increasing degree of non-linear learning. The resulting models can be learned efficiently and enjoy generalization guarantees that extend across input sizes, as we verify empirically.

1 Introduction

Deep learning becomes far more efficient with prior knowledge of function invariants. This knowledge underlies architectural choices that enforce the invariance or equivariance in the network, including Convolutional Neural Networks [LeC+98] which encode translation symmetries, and Graph Neural Networks [Sca+08] which encode permutation symmetries, to name a few. An important class concerns functions defined over inputs of the form $\{x_1 \dots x_N\}$, with each x_i belonging to a base space $\mathbb{I}$, that are invariant to permutations of the input elements. Several architectures explicitly encode this invariance by viewing the input as a set, while remaining universal [Zah+17; Qi+17]. However, these formulations assume a constant input size, which precludes learning an entire family of symmetric functions.

Such symmetric functions appear naturally across several domains, including particle physics, computer graphics, population statistics and cosmology. Yet, in most of these applications, the input size corresponds to a sampling parameter that is independent of the underlying symmetric function of interest. As a motivating example, consider the function family induced by the max function, where for varying N, $f_N(\{x_1 \dots x_N\}) = \max_{i \leq N} x_i$. It is natural to ask if a network can simultaneously learn all these functions.

In this work, we interpret such input sets in terms of an empirical measure defined over $\mathbb{I}$, and develop simple families of neural networks defined over the space of *probability measures of* $\mathbb{I}$, as initially suggested in [PK19; DPC19]. We identify functional spaces implementable with neural architectures and provide generalization bounds that showcase a natural hierarchy among spaces of symmetric functions. In particular, our framework allows us to understand the question of generalizing across input sizes as a corollary. Our constructions heavily rely on the theory of infinitely wide neural networks [Ben+06; Ros+07; Bac17a], and provide a novel instance of *depth separation* that leverages the symmetric structure of the input.

*This work is partially supported by the Alfred P. Sloan Foundation, NSF RI-1816753, NSF CAREER CIF 1845360, and the Institute for Advanced Study.

Summary of Contributions: We consider the infinite-width limit of neural networks taking as domain the space of probability measures in order to formalize learning of symmetric function families. We prove a necessary and sufficient condition for which symmetric functions can be learned. By controlling the amount of non-linear learning, we partition the space of networks on measures into several function classes, proving a separation result among the classes as well as proving a generalization result and empirically studying the performance of these classes to learn symmetric functions on synthetic and real-world data.

Related Work Several works consider representing symmetric functions of fixed input size with invariant neural networks, and in particular there are two main universal architectures, DeepSets [Zah+17] and PointNet [Qi+17]. These models primarily differ in their choice of pooling function. An alternative generalization of DeepSets is given in [Mar+19], which proves the universality of tensor networks invariant to any subgroup of the symmetric group, rather than the symmetric group itself. Regarding variable input size, the work from [Wag+19] proves lower bounds on representation of the max function in the DeepSets architecture with a dependency on input size.

The work most similar to ours are [PK19; DPC19], which also normalize the DeepSets architecture to define a function on measures. However, they only prove the universality of this model, while we further justify the model by classifying symmetric families that are representable and recovering generalization results. Although we motivate our work from symmetric functions on finite sets, there are applications in multi-label learning [Fro+15] and evolving population dynamics [HGJ16] that directly require functions on measures.

Our results also borrow heavily from the framework given by [Bac17a], which introduces function classes to characterize neural networks in the wide limit, and proves statistical generalization bounds to demonstrate the advantage of non-linear learning.

2 Preliminaries

2.1 Problem Setup

Let $\mathbb{I} \subseteq \mathbb{R}^d$ be a convex domain, and $N \in \mathbb{N}$. A *symmetric* function $f : \mathbb{I}^N \to \mathbb{R}$ is such that $f(x_1, \dots x_N) = f(x_{\pi(1)}, \dots x_{\pi(N)})$ for any $x \in \mathbb{I}^N$ and any permutation $\pi \in \mathcal{S}_N$. In this work, we are interested in learning symmetric functions defined independently of N. Let $\bar{\mathbb{I}} = \bigcup_{N=1}^{\infty} \mathbb{I}^N$, then $f : \bar{\mathbb{I}} \to \mathbb{R}$ is symmetric if f restricted to $\mathbb{I}^N$ is symmetric for each $N \in \mathbb{N}$. Denote by $\mathcal{F}_{\text{sym}}$ the space of symmetric functions defined on $\bar{\mathbb{I}}$. This setting is motivated by applications in statistical mechanics, particle physics, or computer graphics, where N is associated with a sampling parameter.

We focus on the realizable regression setting, where we observe a dataset $\{(\boldsymbol{x}_i, f^*(\boldsymbol{x}_i)) \in \bar{\mathbb{I}} \times \mathbb{R}\}_{i=1,\dots n}$ of n samples from an unknown symmetric function f^*, and $\boldsymbol{x}_i$ are drawn iid from a distribution $\mathcal{D}$ on $\bar{\mathbb{I}}$. The goal is to find a proper estimator $\hat{f} \in \mathcal{F}_{\text{sym}}$ such that the population error $\mathbb{E}_{\boldsymbol{x}\sim\mathcal{D}}\ell(f^*(\boldsymbol{x}), \hat{f}(\boldsymbol{x}))$ is low, where ℓ is a convex loss.

Following a standard Empirical Risk Minimisation setup [SB14; Bac17a], we will construct hypothesis classes $\mathcal{F} \subset \mathcal{F}_{\text{sym}}$ endowed with a metric $\|f\|_{\mathcal{F}}$, and consider

$$\hat{f} \in \operatorname{argmin}_{f\in\mathcal{F};\|f\|_{\mathcal{F}}\leq\delta} \frac{1}{n}\sum_{i=1}^{n} \ell(f^*(\boldsymbol{x}_i), f(\boldsymbol{x}_i)) \,, \tag{1}$$

where δ is a regularization parameter that is optimised using e.g. cross-validation. We focus on the approximation and statistical aspects of this estimator for different choices of $\mathcal{F}$; how to solve the optimization problem (1) is not the focus of the present work and will be briefly discussed in Section 7.

2.2 Symmetric Polynomials

Arguably the simplest way to approximate symmetric functions is with polynomials having appropriate symmetry. Combining Weierstrass approximation theory with a symmetrization argument, it can be seen that assuming $d = 1$, any symmetric continuous function $f : \mathbb{I}^N \to \mathbb{R}$ can be uniformly approximated by

symmetric polynomials (see [Yar18] for a proof). There are several canonical bases over the ring of symmetric polynomials, but we will consider the one given by the power sum polynomials, given by $p_k(x) = \sum_{i=1}^N x_i^k$, with $x \in \mathbb{I}^N$.

Theorem 2.1 ((2.12) in [Mac98]). *For any symmetric polynomial f on N inputs, there exists a polynomial q such that $f(x) = q(p_1(x), \dots, p_N(x))$.*

If q is linear, this theorem suggests a simple predictor for symmetric functions across varying N. If $x \in \mathbb{I}^M$, we can consider $x \mapsto \sum_{i=1}^N c_i \left(\frac{1}{M} p_i(x)\right) = \sum_{i=1}^N c_i \mathbb{E}_{y\sim\mu}(y^i)$ where $\mu = \frac{1}{M}\sum_{j=1}^M \delta_{x_j}$. The truncated moments of the empirical distribution given by x act as linear features, which yield an estimator over any input size M. We will ultimately consider a generalization of this decomposition, by moving beyond the polynomial kernel to a general RKHS (see Section 3.1).

2.3 Convex Shallow Neural Networks

By considering the limit of infinitely many neurons [Ben+06; Ros+07], [Bac17a] introduces two norms on shallow neural representation of functions ϕ defined over $\mathbb{R}^d$. For a constant $R \in \mathbb{R}$, a fixed probability measure $\kappa \in \mathcal{P}(\mathbb{S}^d)$ with full support, a signed Radon measure $\nu \in \mathcal{M}(\mathbb{S}^d)$, a density $p \in L_2(d\kappa)$, and the notation that $\tilde{x} = [x, R]^T$, define:

$$\gamma_1(\phi) = \inf\left\{ \|\nu\|_{\mathrm{TV}};\ \phi(x) = \int_{\mathbb{S}^d} \sigma(\langle w, \tilde{x}\rangle)\nu(dw) \right\},\ \text{and} \tag{2}$$

$$\gamma_2(\phi) = \inf\left\{ \|p\|_{L_2(d\kappa)};\ \phi(x) = \int_{\mathbb{S}^d} \sigma(\langle w, \tilde{x}\rangle)p(w)\kappa(dw) \right\}, \tag{3}$$

where $\|\nu\|_{\mathrm{TV}} := \sup_{|g|\le 1} \int g d\nu$ is the *Total Variation* of ν and $\sigma(t) = \max(0, t)$ is the ReLU activation [1]. These norms measure the minimal representation of ϕ, using either a Radon measure ν over neuron weights, or a density p over the fixed probability measure κ. The norms induce function classes:

$$\mathcal{F}_1 = \{\phi \in C_0(\mathbb{I}) : \gamma_1(\phi) < \infty\},$$
$$\mathcal{F}_2 = \{\phi \in C_0(\mathbb{I}) : \gamma_2(\phi) < \infty\}.$$

We also assume that the input domain $\mathbb{I}$ is bounded with $\sup_{x\in\mathbb{I}} \|x\|_2 \le R$.

These two functional spaces are fundamental for the theoretical study of shallow neural networks and capture two distinct regimes of overparametrisation: whereas the so-called *lazy* or kernel regime corresponds to learning in the space $\mathcal{F}_2$ [CB18; JGH18], which is in fact an RKHS with kernel given by $k(x, y) = \mathbb{E}_{w\sim\kappa}[\sigma(\langle w, \tilde{x}\rangle)\sigma(\langle w, \tilde{y}\rangle)]$ [Bac17a] [2], the mean-field regime captures learning in $\mathcal{F}_1$, which satisfies $\mathcal{F}_2 \subset \mathcal{F}_1$ from Jensen's inequality, and can efficiently approximate functions with hidden low-dimensional structure, as opposed to $\mathcal{F}_2$ [Bac17a].

Finally, one can leverage the fact that the kernel above is an expectation over features to define a finite-dimensional random feature kernel $k_m(x, y) = \frac{1}{m}\sum_{j=1}^m \sigma(\langle w_j, \tilde{x}\rangle)\sigma(\langle w_j, \tilde{y}\rangle)$ with $w_j \overset{i.i.d.}{\sim} \kappa$, which defines a (random) RKHS $\mathcal{F}_{2,m}$ converging to $\mathcal{F}_2$ as m increases [Bac17b; RR08]. The empirical norm $\gamma_{2,m}$ can be defined similarly to γ_2, where the density p is replaced by coefficients over the sampled basis functions $\sigma(\langle w_j, \cdot\rangle)$.

2.4 Symmetric Neural Networks

A universal approximator for symmetric functions was proposed by [Zah+17], which proved that for any fixed N and $f_N \in \mathcal{F}_{\mathrm{sym}}^N$ there must exist $\Phi : \mathbb{I} \to \mathbb{R}^L$ and $\rho : \mathbb{R}^L \to \mathbb{R}$ such that

$$f_N(x) = \rho\left(\frac{1}{N}\sum_{n=1}^N \Phi(x_n)\right). \tag{4}$$

[1] For concreteness of exposition, we will focus this work on the ReLU activation, but the essence of our results can be extended to more general families of activation functions.

[2] Or a modified NTK kernel that also includes gradients with respect to first-layer weights [JGH18]

However, universality is only proven for fixed N. Given a symmetric function $f \in \mathcal{F}_{\text{sym}}$ we might hope to learn ρ and Φ such that this equation holds for all N. Note that the fraction $\frac{1}{N}$ is not present in their formulation, but is necessary for generalization across N to be feasible (as otherwise the effective domain of ρ could grow arbitrarily large as $N \to \infty$).

Treating the input to ρ as an average motivates moving from sets to measures as inputs, as proposed in [PK19; DPC19]. Given $x \in \mathbb{I}^N$, let $\mu^{(N)} = \frac{1}{N}\sum_{i=1}^N \delta_{x_i}$ denote the empirical measure in the space $\mathcal{P}(\mathbb{I})$ of probability measures over $\mathbb{I}$. Then (4) can be written as $f_N(x) = \rho\left(\int_{\mathbb{I}} \Phi(u)\mu^{(N)}(du)\right)$.

3 From Set to Measure Functions

3.1 Neural Functional Spaces for Learning over Measures

Equipped with the perspective of (4) acting on an empirical measure, we consider shallow neural networks that take probability measures as inputs, with test functions as weights. We will discuss in Section 3.2 which functions defined over sets admit an extension to functions over measures.

Let $\mathcal{A}$ be a subset of $C_0(\mathbb{I})$, equipped with its Borel sigma algebra. For $\mu \in \mathcal{P}(\mathbb{I})$, and a signed Radon measure $\chi \in \mathcal{M}(\mathcal{A})$, define $f : \mathcal{P}(\mathbb{I}) \to \mathbb{R}$ as

$$f(\mu; \chi) = \int_{\mathcal{A}} \widetilde{\sigma}(\langle \phi, \mu \rangle)\chi(d\phi) \, . \tag{5}$$

where $\widetilde{\sigma}$ is again a scalar activation function, such as the ReLU, and $\langle \phi, \mu \rangle := \int_{\mathbb{I}} \phi(x)\mu(dx)$. Crucially, the space of functions given by $f(\cdot; \chi)$ were proven to be dense in the space of real-valued continuous (in the weak topology) functions on $\mathcal{P}(\mathbb{I})$ in [PK19; DPC19], and so this network exhibits universality.

Keeping in mind the functional norms defined on test functions in Section 2.3, we can introduce analogous norms for neural networks on measures. For a fixed probability measure $\tau \in \mathcal{P}(\mathcal{A})$, define

$$\|f\|_{1,\mathcal{A}} = \inf\left\{ \|\chi\|_{\text{TV}};\ \chi \in \mathcal{M}(\mathcal{A}),\ f(\mu) = \int_{\mathcal{A}} \widetilde{\sigma}(\langle \phi, \mu \rangle)\chi(d\phi) \right\} \text{ and} \tag{6}$$

$$\|f\|_{2,\mathcal{A}} = \inf\left\{ \|q\|_{L_2(d\tau)};\ q \in L_2(d\tau),\ f(\mu) = \int_{\mathcal{A}} \widetilde{\sigma}(\langle \phi, \mu \rangle)q(\phi)\tau(d\phi) \right\} \, . \tag{7}$$

where we take the infima over Radon measures $\chi \in \mathcal{M}(\mathcal{A})$ and densities $q \in L_2(d\tau)$. Analogously these norms also induce the respective function classes $\mathcal{G}_1(\mathcal{A}) = \{f : \|f\|_{1,\mathcal{A}} < \infty\}$, $\mathcal{G}_2(\mathcal{A}) = \{f : \|f\|_{2,\mathcal{A}} < \infty\}$. The argument in Appendix A of [Bac17a] implies $\mathcal{G}_2(\mathcal{A})$ is an RKHS, with associated kernel $k_{\mathcal{G}}(\mu, \mu') = \int_{\mathcal{A}} \widetilde{\sigma}(\langle \phi, \mu \rangle)\widetilde{\sigma}(\langle \phi, \mu' \rangle)\tau(d\phi)$.

Moving from vector-valued weights to function-valued weights presents an immediate issue. The space $C_0(\mathbb{I})$ is infinite-dimensional, and it is not obvious how to learn a measure χ over this entire space. Moreover, our ultimate goal is to understand finite-width symmetric networks, so we would prefer the function-valued weights be efficiently calculable rather than pathological. To that end, we choose the set of test functions $\mathcal{A}$ to be representable as regular neural networks.

Explicitly, using the function norms of Section 2.3, define

$$\begin{aligned} \mathcal{A}_{1,m} &:= \left\{ \phi;\ \phi(x) = \sum_{j=1}^m \alpha_j \sigma(\langle w_j, \tilde{x} \rangle) \, ,\ \|w_j\|_2 \le 1, \|\alpha\|_1 \le 1 \right\} \subset \{\phi \in \mathcal{F}_1;\ \gamma_1(\phi) \le 1\}), \\ \mathcal{A}_{2,m} &:= \{\phi \in \mathcal{F}_{2,m} :\ \gamma_{2,m}(\phi) \le 1\} \, , \end{aligned} \tag{8}$$

$\mathcal{A}_{1,m}$ thus contains functions in the unit ball of $\mathcal{F}_1$ that can be expressed with m neurons, and $\mathcal{A}_{2,m}$ contains functions in the (random) RKHS $\mathcal{F}_{2,m}$ obtained by sampling m neurons from κ. By definition $\mathcal{A}_{2,m} \subset \mathcal{A}_{1,m}$ for all m. Representational power grows with m, and observe that the approximation rate in the unit ball of $\mathcal{F}_1$ or $\mathcal{F}_2$ is in $m^{-1/2}$, obtained for instance with Monte-Carlo estimators [Bac17a; MWE19]. Hence we can also consider the setting where $m = \infty$, with the notation $\mathcal{A}_{\{i,\infty\}} = \{\phi \in \mathcal{F}_i : \gamma_i(\phi) \le 1\}$. Note

	First Layer	Second Layer	Third Layer
$\mathcal{S}_1$	Trained	Trained	Trained
$\mathcal{S}_2$	Frozen	Trained	Trained
$\mathcal{S}_3$	Frozen	Frozen	Trained

Table 1: Training for finite function approximation

also that there is no loss of generality in choosing the radius to be 1, as by homogeneity of σ any ϕ with $\gamma_i(\phi) < \infty$ can be scaled into its respective norm ball.

We now examine the combinations of $\mathcal{G}_i$ with $\mathcal{A}_i$:

- $\mathcal{S}_{1,m} := \mathcal{G}_1(\mathcal{A}_{1,m})$; the measure χ is supported on test functions in $\mathcal{A}_{1,m}$.
- $\mathcal{S}_{2,m} := \mathcal{G}_1(\mathcal{A}_{2,m})$; χ is supported on test functions in $\mathcal{A}_{2,m}$.
- $\mathcal{S}_{3,m} := \mathcal{G}_2(\mathcal{A}_{2,m})$; χ has a density with regards to τ, which is supported on $\mathcal{A}_{2,m}$.
- The remaining class $\mathcal{G}_2(\mathcal{A}_{1,m})$ requires defining a probability measure τ over $\mathcal{A}_{1,m}$ that sufficiently spreads mass outside of any RKHS ball. Due to the difficulty in defining this measure in finite setting, we omit this class.

Note that from Jensen's inequality and the inclusion $\mathcal{A}_{2,m} \subset \mathcal{A}_{1,m}$ for all m, we have the inclusions $\mathcal{S}_{3,m} \subset \mathcal{S}_{2,m} \subset \mathcal{S}_{1,m}$. And $\mathcal{S}_{3,m}$ is clearly an RKHS, since it is a particular instantiation of $\mathcal{G}_2(\mathcal{A})$. In the sequel we will drop the subscript m and simply write $\mathcal{A}_i$ and $\mathcal{S}_i$.

These functional spaces provide an increasing level of adaptivity: while $\mathcal{S}_2$ is able to adapt by selecting 'useful' test functions ϕ, it is limited to smooth test functions that lie on the RKHS, whereas $\mathcal{S}_1$ is able to also adapt to more irregular test functions that themselves depend on low-dimensional structures from the input domain. We let $\|f\|_{\mathcal{S}_i}$ denote the associated norm, i.e. $\|f\|_{\mathcal{S}_1} := \|f\|_{1,\mathcal{A}_1}$.

Finite-Width Implementation: For any m, these classes admit a particularly simple interpretation when implemented in practice. On the one hand, from (8), the spaces of test functions are implemented as a single hidden-layer neural network of width m. On the other hand, the integral representations in (6) and (7) are instantiated by a finite-sum using m' neurons, leading to the finite

For any m, these classes admit a particularly simple interpretation when implemented in practice. On the one hand, the spaces of test functions are implemented as a single hidden-layer neural network of width m. On the other hand, the integral representations in (6) and (7) are instantiated by a finite-sum using m' neurons, leading to the finite analogues of our function classes given in Table 1. Specifically,

$$f(\mu) = \frac{1}{m'} \sum_{j'=1}^{m'} b_{j'} \widetilde{\sigma} \left(\frac{1}{m} \sum_{j=1}^{m} c_{j',j} \int \sigma(\langle w_{j',j}, \tilde{x} \rangle) \mu(dx) \right)$$

One can verify [NTS15] that the finite-width proxy for the variation norm is given by

$$\|f\|_1 = \frac{1}{m'} \sum_{j'} |b_{j'}| \|\phi_{j'}\|_1 \leq \frac{1}{mm'} \sum_{j',j} |b_{j'}||c_{j',j}| \|w_{j',j}\| \,,$$

which in our case corresponds to the so-called path norm [NTS14]. In particular, under the practical assumption that the test functions $\phi_{j'}$ are parameterized by two-layer networks with shared first layer, the weight vectors $w_{j',j}$ only depend on j and this norm may be easily calculated as a matrix product of the network weights. We can control this term by constraining the weights of the first two layers to obey our theoretical assumptions (of bounded weights and test functions in respective RKHS balls), and regularize the final network weights. See Section 6 and the Appendix for practical relaxations of the constraints for our experiments.

RKHS over $\mathcal{P}(\mathbb{I})$:

$$\begin{aligned} K_m(\mu, \mu') &= \int_{\mathcal{A}_{2,m}} \widetilde{\sigma}(\langle \phi, \mu \rangle) \widetilde{\sigma}(\langle \phi, \mu' \rangle) d\tau(\phi) \\ &= \mathbb{E}_{\phi \sim \tau} \left[\widetilde{\sigma}(\langle \phi, k_m[\mu] \rangle_{\mathcal{F}_{2,m}}) \widetilde{\sigma}(\langle \phi, k_m[\mu'] \rangle_{\mathcal{F}_{2,m}}) \right] \,, \end{aligned}$$

The kernel associated with our space $\mathcal{S}_3$ with finite m is of the form

$$K_m(\mu,\mu') = \int_{\mathcal{A}_{2,m}} \widetilde{\sigma}(\langle\phi,\mu\rangle)\widetilde{\sigma}(\langle\phi,\mu'\rangle)d\tau(\phi)$$
$$= \mathbb{E}_{\phi\sim\tau}\left[\widetilde{\sigma}(\langle\phi,k_m[\mu]\rangle_{\mathcal{F}_{2,m}})\widetilde{\sigma}(\langle\phi,k_m[\mu']\rangle_{\mathcal{F}_{2,m}})\right] ,$$

where $k[\mu] := \int k(x,\cdot)\mu(dx)$ is the RKHS embedding of a probability measure [BT11; Sri+10]. It is thus a compositional kernel, akin to the hierarchical construction of Convolutional Kernel Networks [Mai+14]. When $\widetilde{\sigma}(t) = t$, the kernel K_m becomes $K_m = \mathbb{E}_{\phi\sim\tau}\langle k_m\otimes k_m, \phi\otimes\phi\rangle = \langle k_m\otimes k_m, \Sigma\rangle$, where Σ is the covariance of τ. Thus choosing k in $\mathcal{F}_2$ as a polynomial kernel amounts to a linear approximation of the underlying symmetric function in terms of finite-order moments of the measure.

3.2 Continuous Extension

In general, the functions we want to represent don't take in measures $\mu \in \mathcal{P}(\mathbb{I})$ as inputs. In this section, we want to understand when a function f defined on the power set $f : \bar{\mathbb{I}} \to \mathbb{R}$ can be extended to a continuous map $\bar{f} : \mathcal{P}(\mathbb{I}) \to \mathbb{R}$ in the weak topology, in the sense that for all $N \in \mathbb{N}$ and all $(x_1,\dots x_N) \in \mathbb{I}^N$,

$$\bar{f}\left(\frac{1}{N}\sum_{i=1}^N \delta_{x_i}\right) = f(x_1,\dots,x_N) .$$

Observe that by construction $\bar{f}$ captures the permutation symmetry of the original f. Define the mapping $D : \bar{\mathbb{I}} \to \mathcal{P}(\mathbb{I})$ by $D(x_1,\dots,x_N) = \frac{1}{N}\sum_{i=1}^N \delta_{x_i}$. Let $\hat{\mathcal{P}}_N(\mathbb{I}) := D(\mathbb{I}^N)$ and $\hat{\mathcal{P}}(\mathbb{I}) = \bigcup_{N=1}^\infty \hat{\mathcal{P}}_N(\mathbb{I})$, so that $\hat{\mathcal{P}}(\mathbb{I})$ is the set of all finite discrete measures. For $\mu \in \hat{\mathcal{P}}(\mathbb{I})$, let $N(\mu)$ be the smallest dimension of a point in $D^{-1}(\mu)$, and let x be this point (which is unique up to permutation). Then define $\hat{f} : \hat{\mathcal{P}}(\mathbb{I}) \to \mathbb{R}$ such that $\hat{f}(\mu) = f_N(x)$.

We also write $W_1(\mu,\mu')$ as the Wasserstein 1-metric under the $\|\cdot\|_2$ norm [Vil08]. The following proposition establishes a necessary and sufficient condition for continuous extension of f:

Proposition 3.1. *There exists a continuous extension $\bar{f}$ iff $\hat{f}$ is uniformly continuous with regard to the W_1 metric on its domain.*

This result formalises the intuition that in order to extend a symmetric function from sets to measures, one needs a minimal amount of regularity *across* sizes. We next show examples of eligible symmetric families that can be extended to $\mathcal{P}(\mathbb{I})$.

3.3 Examples of Eligible Symmetric Families

Moment-based Functions: Functions based on finite-range interactions across input elements

$$f(x) = \rho\left(\frac{1}{N}\sum_{i=1}^N \phi_1(x_i), \frac{1}{N^2}\sum_{i_1,i_2=1}^N \phi_2(x_{i_1},x_{i_2}),\dots,\frac{1}{N^K}\sum_{i_1,\dots,i_K=1}^N \phi_K(x_{i_1},\dots,x_{i_K})\right)$$

admit a continuous extension, by observing that

$$\frac{1}{N^k}\sum_{i_1,\dots,i_k=1}^N \phi_k(x_{i_1},\dots,x_{i_k}) = \int_{\mathbb{I}^k} \phi_k(u_1,\dots,u_k)\mu(du_1)\dots\mu(du_k) = \langle\phi_k,\mu^{\otimes k}\rangle ,$$

where $\mu = D(x)$ and $\mu^{\otimes k}$ is the k-th product measure, thus $\bar{f}(\mu) = \rho\left(\langle\phi_1,\mu\rangle,\dots,\langle\phi_k,\mu^{\otimes k}\rangle\right)$.

Ranking: Suppose that $\mathbb{I} \subseteq \mathbb{R}$. The max function $f_N(x) = \max_{i \leq N} x_i$ cannot be lifted to a function on measures due to discontinuity in the weak topology. Specifically, consider $\mu = \delta_0$ and $\nu_N = \frac{N-1}{N}\delta_0 + \frac{1}{N}\delta_1$. Then $\nu_N \rightharpoonup \mu$, but for $\hat{f}$ as in Proposition 3.1, $\hat{f}(\nu_N) = 1 \neq 0 = \hat{f}(\mu)$.

Nevertheless, we can define an extension on a smooth approximation via the softmax, namely $g_N^\lambda(x) = \frac{1}{\lambda} \log \frac{1}{N} \sum_{i=1}^N \exp(\lambda x_i)$. This formulation, which is the softmax up to an additive term, can clearly be lifted to a function on measures, with the bound $\|g_N^\lambda - f_N\|_\infty \leq \frac{\log N}{\lambda}$. Although we cannot learn the max family across all N, we can approximate arbitrarily well for bounded N.

Counterexamples: Define the map $\Delta_k : \mathbb{R}^N \to \mathbb{R}^{kN}$ such that $\Delta_k(x)$ is a vector of k copies of x. Then a necessary condition for the function $\hat{f}$ introduced in Proposition 3.1 to be uniformly continuous is that $f_N(x) = f_{kN}(\Delta_k(x))$ for any k. Intuitively, if f_N can distinguish the input set beyond the amount of mass on each point, it cannot be lifted to measures. This fact implies any continuous approximation to the family $f_N(x) = x_{[2]}$, the second largest value of x, in our formulation will incur constant error.

4 Approximation and Function Class Separation

The functional spaces $\mathcal{S}_i$ introduced in the previous section capture distinct overparameterized limits of simple neural networks for symmetric functions. This section establishes separations between these spaces in terms of approximation rates, in a similar spirit of [Bac17a], and relates a broad class of variational symmetric families to these spaces via the Laplace principle.

4.1 Approximation of single 'neurons'

In the same spirit as the "separations" between $\mathcal{F}_1$ and $\mathcal{F}_2$, we characterise prototypical functions that belong to $\mathcal{S}_i$ but have poor approximation rates in $\mathcal{S}_{i+1}$ for $i = \{1, 2\}$ in terms of the relevant parameters of the problem, the input dimensionality d and the bandwidth parameter m. Such functions are given by single neurons in a spherical input regime (details for this setting are given in the Appendix):

Theorem 4.1 (informal)**.** *Let $\widetilde{\sigma} = \sigma$ be the ReLU activation, and assume $m = \infty$. For appropriate choices of the kernel base measures κ and τ, there exist f_2 with $\|f_2\|_{\mathcal{S}_2} \leq 1$ and f_3 with $\|f_3\|_{\mathcal{S}_3} \leq 1$ such that:*

$$\inf_{\|f\|_{\mathcal{S}_3} \leq \delta} \|f - f_2\|_\infty \gtrsim d^{-1}\delta^{-3/(d-1)} ,$$
$$\inf_{\|f\|_{\mathcal{S}_2} \leq \delta} \|f - f_1\|_\infty \gtrsim |d^{-1} - \delta 2^{-d/2}| .$$

These separations use the infinity norm rather than an appropriate L_2 norm, and therefore hold in a weaker norm than separation between $\mathcal{F}_1$ and $\mathcal{F}_2$ proven in [Bac17a]. Nevertheless, these separations confirm that symmetric network expressiveness is *graded* by the degree of non-linear learning.

Both results hold in the domain $m = \infty$, so from the concentration of the empirical kernel $k_m \to k$, with high probability these approximation lower bounds will still hold for sufficiently large m. In finite-width implementations, however, m may be sufficiently small that the random kernel more explicitly determines the expressiveness of $\mathcal{S}_{i,m}$. We experimentally test the presence of these depth separations with finite m in Section 6.

4.2 Approximation of variational symmetric function via Laplace method

Consider any symmetric family $f_N(x) = \operatorname{argmin}_{t \in T} \langle \hat{\mu}_x, \phi_t \rangle$ where $\hat{\mu}_x$ is the empirical measure of x, ie, $\hat{\mu}_x = \frac{1}{N}\sum_i \delta_{x_i}$, T is a Euclidean subset, and $t \mapsto \phi_t$ is measurable. For example $T = \mathbb{R}$ and $\phi_t(x) = |t - x|$ yields f_N as the median.

Although this function family isn't necessarily uniformly continuous in the weak topology, we highlight the option of a Laplace approximation. Define $E_\mu(t) := \langle \mu, \phi_t \rangle$ and introduce the density $p_\beta(t) = \frac{1}{Z} e^{-\beta E_\mu(t)}$ where $Z = \int_T e^{-\beta E_\mu(t)} dt$ is the partition function. Then consider the Gibbs approximation:

$$g_\beta(\mu) := \mathbb{E}_{p_\beta}[t] = \frac{1}{Z}\int_T t e^{-\beta E_\mu(t)} dt\ .$$

One can verify (e.g. [RRT17]) that $g_\beta \to g$ pointwise at a rate $\sim \frac{d\log(\beta+1)}{\beta}$. As g_β is continuous, by universality it can be represented in $\mathcal{S}_i$ for all $i = \{1,2,3\}$. An approximation of g_β can be implemented as a ratio of two shallow networks $g_\beta(\mu) = \frac{\int_T t\sigma_1(\langle \mu, \phi_t\rangle)dt}{\int_T \sigma_1(\langle \mu, \phi_t\rangle)dt}$, with $\sigma_1(u) = e^{-\beta u}$. However, the approximation rates blow-up as $\beta \to \infty$ with an exponential dependency on the dimension of T.

5 Generalization and Concentration

In this section we establish generalisation guarantees for learning symmetric functions in $\mathcal{S}_1$, and provide quantitative results that control the sampling fluctuations in $\mathcal{S}_1$ (and therefore also in $\mathcal{S}_2$, $\mathcal{S}_3$).

5.1 Generalization Bounds

Despite being a larger function class than $\mathcal{F}_2$, the class $\mathcal{F}_1$ enjoys a nice generalization bound [Bac17a]. Crucially, this property is inherited when we lift to functions on measures, controlling the generalization of functions in $\mathcal{S}_1$:

Proposition 5.1. *Assume for given δ, for all y the loss function $\ell(y,\cdot)$ is G-Lipschitz on $B_0(\sqrt{2}R\delta)$, and $l(y,0) \le RG\delta$. Then with probability at least $1-t$,*

$$\sup_{\|f\|_{\mathcal{S}_1}\le\delta}\left|\mathbb{E}_{\mu\sim\mathcal{D}}\ell(f^*(\mu), f(\mu)) - \frac{1}{n}\sum_{i=1}^n \ell(f^*(\mu_i), f(\mu_i))\right| \le \frac{(2+4\max(R,R^2))G\delta}{\sqrt{n}}\left(2+\sqrt{\frac{\log 2/t}{2}}\right)\ .$$

This proposition demonstrates that learning in $\mathcal{S}_1$ is not cursed by the dimension of the underlying input space $\mathbb{I}$. In other words, the main price for learning in $\mathcal{S}_1$ is not in generalization, despite the size of this class relative to $\mathcal{S}_2$ and $\mathcal{S}_3$. In the absence of a lower bound on generalization error for the RKHS function classes, our experiments investigate the generalization of these models in practice.

Although d and N do not appear in this bound, these parameters nevertheless impact the generalization of our function classes $\mathcal{S}_i$. The input dimension controls the separation of the classes according to Theorem 4.1, and therefore larger d weakens the generalization of $\mathcal{S}_2$ and $\mathcal{S}_3$; compare Figure 1 and Figure 5 (in the Appendix) for how RKHS methods suffer in higher dimensions. Whereas large N and a natural choice of $\mathcal{D}$ make generalization for $\mathcal{S}_1$, and hence all three classes, nearly trivial, as discussed in section 5.2.

5.2 Concentration across Input Size

Consider the data distribution from which we sample, namely a measure from $\mathcal{P}\left(\overline{\mathbb{I}}\right)$ to sample finite sets. A natural way to draw data is to consider the following sampling procedure: given $\xi \in \mathcal{P}(\mathcal{P}(\mathbb{I}))$ and $\Omega \in \mathcal{P}(\mathbb{N})$, draw $\mu \sim \xi$ and $N \sim \Omega$, sample N independent points $x_i \sim \mu$, and return $\{x_1, \ldots, x_N\}$. If ξ is too peaked, this sampling process will concentrate very rapidly:

Proposition 5.2. *For $\xi = \delta_{\mu^*}$, the Rademacher complexity $R_n(B_\delta(\mathcal{S}_1)) \lesssim \delta R(n^{-1/2} + \mathbb{E}_{N\sim\Omega}[N^{-1/d}])$ where $B_\delta(\mathcal{S}_1) = \{f \in \mathcal{S}_1 : \|f\|_{\mathcal{S}_1} \le \delta\}$.*

Hence, the question of generalization across differently sized sets becomes trivial if N is large and d is small. In our experiments, $N \approx d$, so we will nevertheless choose $\xi = \delta_\mu$ for some $\mu \in \mathcal{P}(\mathbb{I})$. We consider more exotic data distributions over measures as ample territory for future work.

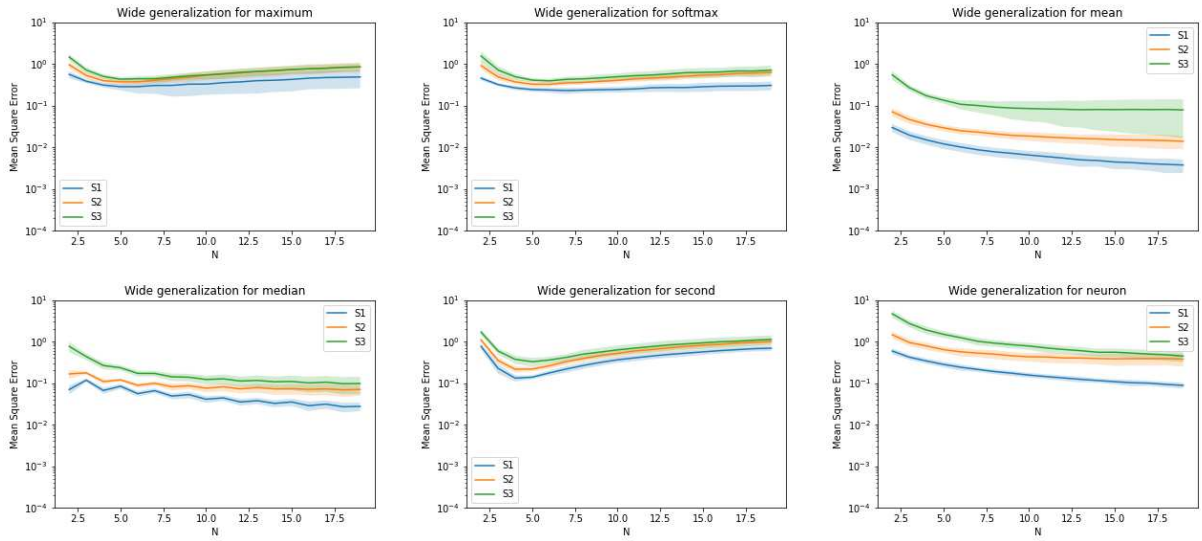

Figure 1: Test Error for $d = 10$ and $m = 100$ on the neural architectures of Section 3.1

	Error ($N = 100$)	Error ($N = 200$)
$\mathcal{S}_1$	4.96 ± 0.50	3.27 ± 0.12
$\mathcal{S}_2$	7.28 ± 1.45	5.19 ± 0.85
$\mathcal{S}_3$	11.96 ± 0.41	8.79 ± 0.17
5-Layers	2.83 ± 0.12	1.49 ± 0.06

Table 2: Classification test error on MNIST in percent, after images are compressed into sets of size N, trained with $N = 200$.

6 Experiments

Experimental Setup: We instantiate our three function classes in the finite network setting, as outlined in Table 1. To ensure a fair comparison, we choose the same architecture for all function classes, changing only which weights are frozen and which functional norm is used for regularization.

We use dimension $d = 10$, and $m = 100$ random kernel features to as a basis over test functions. Each network is trained on a batch of 100 input sets. For our data distribution we consider two settings: **narrow**, where $\mathbb{I} = [-1, 1]^d$ and ξ places all its mass on the uniform distribution $U([-1, 1]^d)$; and **wide**, where $\mathbb{I} = [-3, 3]^d$ and ξ places its mass on the uniform distribution $U([-3, 3]^d)$.

Our aim is to practically study the approximation bounds of Theorem 4.1, as well as the generalization result of Proposition 5.1. Towards the second point, we choose $\Omega = \delta_4$, i.e. all networks train on sets of size 4, and test on sets of varying size. From the results we can measure out-of-distribution generalization of finite sets.

We consider several common symmetric functions as objectives (see Figure 1). The one-dimensional symmetric functions are defined on sets of vectors by first applying norms, i.e. $f_N(x) = \max_{1 \le i \le N} \|x_i\|_2$. The only exceptions are the planted neuron and smooth neuron, given as networks with weight initializations distinct from the learned models. Further implementation details are given in the Appendix.

We also consider an applied experiment on MNIST to observe how the finite-width implementations perform on real-world data. This first requires mapping MNIST images to sets / point clouds, as described in the Appendix. Again, in order to study generalization, the training sets are of different size than the testing sets.

Discussion: We observe in Figure 1 that $\mathcal{S}_3$ performs substantially worse in several cases, consistent with this function class being the smallest of those considered. The classes $\mathcal{S}_2$ and $\mathcal{S}_1$ are competitive for some functions. In particular, for the smooth neuron function $f(\mu) = \sigma(\langle \phi^*, \mu \rangle)$, we explicitly initialize our $\mathcal{S}_2$ network so that f is exactly representable, and hence observe almost identical performance of $\mathcal{S}_1$ and $\mathcal{S}_2$ in that setting.

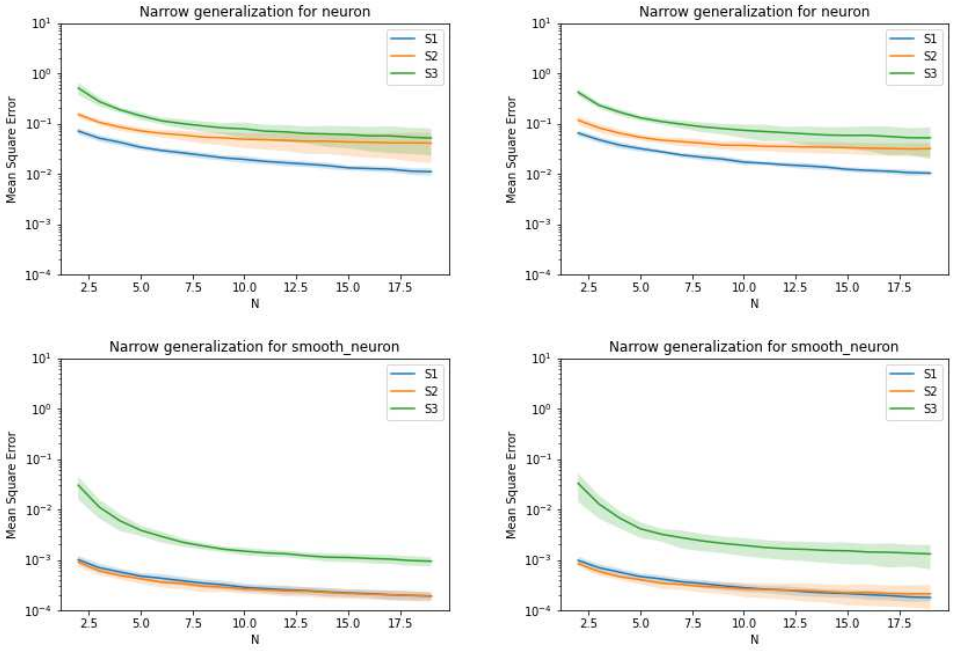

Figure 2: Planted neurons for $m = 100$ (left two) and $m = 200$ (right two). The smooth neuron has weights sampled consistently with $\mathcal{F}_{2,m}$ while regular neuron has weights sampled out-of-distribution.

The essential takeaway is the performance of the three models on the planted neuron. By using a distinct weight initialization for the planted neuron, its first layer will have little mass under κ, and its first two layers will have little mass under τ, and therefore random features will not suffice to approximate this neuron. Because all function classes are only trained on a single input size, the median function oscillates based on parity because its behavior is slightly different depending on the input parity. The second-largest-element function generalizes extremely poorly, consistent with the observation in Section 3.3 that this function family cannot be approximated without constant error. All function classes more effectively generalize across different N on the softmax than the max, following from the former's uniform continuity in measure space.

Figure 2 compares the effect of the smoothness in the planted neuron learning task: the smooth neuron is parameterized by a test function with low norm in the RKHS $\mathcal{F}_2$, while the regular neuron is parameterized by an arbitrary test function. On increasing m, $\mathcal{S}_1$ still achieves the best approximation on the neuron, but the gap between $\mathcal{S}_1$ and $\mathcal{S}_2$ shrinks as $\mathcal{S}_2$ will sometimes be fortunate with the sampled kernel features. For the smooth neuron, $\mathcal{S}_1$ and $\mathcal{S}_2$ performs comparably, but $\mathcal{S}_3$ improves with increasing m. In Figure 3 we confirm the necessity of taking averages rather than sums in the DeepSets architecture, as the unnormalized model cannot generalize outside of the value of $N = 4$ where it was trained.

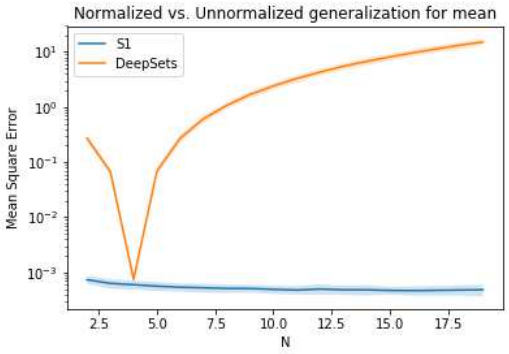

Figure 3: Test error for $\mathcal{S}_1$ versus unnormalized DeepSets architecture.

The results on MNIST, across differently-sized set representations of images, are given in Table 2. Unsurprisingly, because we compress images by mapping to small point clouds, and use the shallowest symmetric network architecture that possesses universality, the test errors are not competitive with regular applications on MNIST. Indeed, all shallow architectures are outperformed by a 5-layer symmetric network baseline. Nevertheless, we still observe the expected ordering of our functional spaces. When testing on smaller sets than training, the generalization error increases faster for S_2 and S_3 than for S_1.

7 Conclusion

In this work, we have analyzed learning and generalization of symmetric functions through the lens of neural networks defined over probability measures, which formalizes the learning of symmetric function families across varying input size. Our experimental data confirms the theoretical insights distinguishing tiers of non-linear learning, and suggests that symmetries in the input might be a natural device to study the functional spaces defined by deeper neural networks. Specifically, and by focusing on shallow architectures, our analysis extends the fundamental separation between adaptive and non-adaptive neural networks from [Bac17a] to symmetric functions, leading to a hierarchy of functional spaces $\mathcal{S}_3 \subset \mathcal{S}_2 \subset \mathcal{S}_1$, in which nonlinear learning is added into the parametrization of the network weights ($\mathcal{S}_2$), and into the parametrization of test functions ($\mathcal{S}_1$) respectively. Our results from Section 4 establish a formal separation between these spaces, and those from Section 5 provide strong statistical guarantees for learning in such non-linear spaces.

A crucial aspect we have not addressed, though, is the computational cost of learning in $\mathcal{S}_1$ through gradient-based algorithms. An important direction of future work is to build on recent advances in mean-field theory for learning shallow neural networks [CB20; MWE19; MW+20; DB20] to study learning dynamics on these symmetric function spaces, including recent studies into deeper models [Woj+20]. Another interesting question is to study the role of spiked Wasserstein models [NR19] into separating $\mathcal{S}_1$ and $\mathcal{S}_2$.

A consideration of the particular distributions over empirical measures that exist in practice, in points clouds and particle systems, motivates another exciting direction for future work in symmetric function learning. Finally, a last direction of future research is to relax the permutation invariance to smaller (discrete) symmetry groups determined from input adjacency matrices, thus providing a mean-field view on graph neural networks.

References

[Bac17a] Francis Bach. "Breaking the curse of dimensionality with convex neural networks". In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 629–681.

[Bac17b] Francis Bach. "On the equivalence between kernel quadrature rules and random feature expansions". In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 714–751.

[Ben+06] Yoshua Bengio et al. "Convex neural networks". In: *Advances in neural information processing systems*. 2006, pp. 123–130.

[BT11] Alain Berlinet and Christine Thomas-Agnan. *Reproducing kernel Hilbert spaces in probability and statistics*. Springer Science & Business Media, 2011.

[CB18] Lenaic Chizat and Francis Bach. "A note on lazy training in supervised differentiable programming". In: *arXiv preprint arXiv:1812.07956* (2018).

[CB20] Lenaic Chizat and Francis Bach. "Implicit Bias of Gradient Descent for Wide Two-layer Neural Networks Trained with the Logistic Loss". In: *arXiv preprint arXiv:2002.04486* (2020).

[DB20] Jaume de Dios and Joan Bruna. "On Sparsity in Overparametrised Shallow ReLU Networks". In: *arXiv preprint arXiv:2006.10225* (2020).

[DPC19] Gwendoline De Bie, Gabriel Peyré, and Marco Cuturi. "Stochastic deep networks". In: *International Conference on Machine Learning*. 2019, pp. 1556–1565.

[FG15] Nicolas Fournier and Arnaud Guillin. "On the rate of convergence in Wasserstein distance of the empirical measure". In: *Probability Theory and Related Fields* 162.3-4 (2015), pp. 707–738.

[Fro+15] Charlie Frogner et al. "Learning with a Wasserstein loss". In: *Advances in neural information processing systems*. 2015, pp. 2053–2061.

[He+15] Kaiming He et al. "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification". In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1026–1034.

[HGJ16] Tatsunori Hashimoto, David Gifford, and Tommi Jaakkola. "Learning population-level diffusions with generative RNNs". In: *International Conference on Machine Learning*. 2016, pp. 2417–2426.

[JGH18] Arthur Jacot, Franck Gabriel, and Clément Hongler. "Neural tangent kernel: Convergence and generalization in neural networks". In: *Advances in neural information processing systems*. 2018, pp. 8571–8580.

[KB14] Diederik P Kingma and Jimmy Ba. "Adam: A method for stochastic optimization". In: *arXiv preprint arXiv:1412.6980* (2014).

[LeC+98] Yann LeCun et al. "Gradient-based learning applied to document recognition". In: *Proceedings of the IEEE* 86.11 (1998), pp. 2278–2324.

[Mac98] Ian Grant Macdonald. *Symmetric functions and Hall polynomials*. Oxford university press, 1998.

[Mai+14] Julien Mairal et al. "Convolutional kernel networks". In: *Advances in neural information processing systems*. 2014, pp. 2627–2635.

[Mar+19] Haggai Maron et al. "On the Universality of Invariant Networks". In: *International Conference on Machine Learning*. 2019, pp. 4363–4371.

[MW+20] Chao Ma, Lei Wu, et al. "The Quenching-Activation Behavior of the Gradient Descent Dynamics for Two-layer Neural Network Models". In: *arXiv preprint arXiv:2006.14450* (2020).

[MWE19] Chao Ma, Lei Wu, and Weinan E. "Barron spaces and the compositional function spaces for neural network models". In: *arXiv preprint arXiv:1906.08039* (2019).

[NR19] Jonathan Niles-Weed and Philippe Rigollet. "Estimation of wasserstein distances in the spiked transport model". In: *arXiv preprint arXiv:1909.07513* (2019).

[NTS14] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. "In search of the real inductive bias: On the role of implicit regularization in deep learning". In: *arXiv preprint arXiv:1412.6614* (2014).

[NTS15] Behnam Neyshabur, Ryota Tomioka, and Nathan Srebro. "Norm-Based Capacity Control in Neural Networks". In: *arXiv:1503.00036 [cs, stat]* (Apr. 2015). arXiv: 1503.00036. URL: http://arxiv.org/abs/1503.00036 (visited on 12/18/2019).

[PK19] Tomas Pevny and Vojtech Kovarik. "Approximation capability of neural networks on spaces of probability measures and tree-structured domains". In: *arXiv preprint arXiv:1906.00764* (2019).

[Qi+17] Charles R Qi et al. "Pointnet: Deep learning on point sets for 3d classification and segmentation". In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 652–660.

[Ros+07] Saharon Rosset et al. "ℓ_1 regularization in infinite dimensional feature spaces". In: *International Conference on Computational Learning Theory*. Springer. 2007, pp. 544–558.

[RR08] Ali Rahimi and Benjamin Recht. "Random features for large-scale kernel machines". In: *Advances in neural information processing systems*. 2008, pp. 1177–1184.

[RRT17] Maxim Raginsky, Alexander Rakhlin, and Matus Telgarsky. "Non-convex learning via stochastic gradient Langevin dynamics: a nonasymptotic analysis". In: *arXiv preprint arXiv:1702.03849* (2017).

[SB14] Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

[Sca+08] Franco Scarselli et al. "The graph neural network model". In: *IEEE Transactions on Neural Networks* 20.1 (2008), pp. 61–80.

[Sri+10] Bharath K Sriperumbudur et al. "Hilbert space embeddings and metrics on probability measures". In: *Journal of Machine Learning Research* 11.Apr (2010), pp. 1517–1561.

[Sri+14] Nitish Srivastava et al. "Dropout: a simple way to prevent neural networks from overfitting". In: *The journal of machine learning research* 15.1 (2014), pp. 1929–1958.

[Vil08] Cédric Villani. *Optimal transport: old and new*. Vol. 338. Springer Science & Business Media, 2008.

[Wag+19] Edward Wagstaff et al. "On the Limitations of Representing Functions on Sets". In: *International Conference on Machine Learning*. 2019, pp. 6487–6494.

[Woj+20] Stephan Wojtowytsch et al. "On the Banach spaces associated with multi-layer ReLU networks: Function representation, approximation theory and gradient descent dynamics". In: *arXiv preprint arXiv:2007.15623* (2020).

[Yar18] Dmitry Yarotsky. "Universal approximations of invariant maps by neural networks". In: *arXiv preprint arXiv:1804.10306* (2018).

[Zah+17] Manzil Zaheer et al. "Deep sets". In: *Advances in neural information processing systems*. 2017, pp. 3391–3401.

A Omitted Proofs

A.1 Proof of Proposition 3.1

Proof. For the forward implication, if $\bar{f}$ is a continuous extension, then clearly $\bar{f} = \hat{f}$ restricted to $\hat{\mathcal{P}}(\mathbb{I})$.

Furthermore, continuity of $\bar{f}$ and compactness of $\mathcal{P}(\mathbb{I})$ implies $\bar{f}$ is uniformly continuous, and therefore $\hat{f}$ is as well.

For the backward implication, we introduce $\hat{f}_\epsilon(\mu) = \sup_{\nu \in B_\epsilon(\mu) \cap \hat{\mathcal{P}}(\mathbb{I})} \hat{f}(\nu)$ where the ball $B_\epsilon(\mu)$ is defined with the Wasserstein metric, and define $\bar{f}(\mu) = \inf_{\epsilon > 0} \hat{f}_\epsilon(\mu)$. Density of the discrete measures and uniform continuity of $\hat{f}$ guarantees that $\bar{f}$ is well-defined and finite.

Uniform continuity implies if $\mu \in \hat{\mathcal{P}}(\mathbb{I})$ then $\bar{f}(\mu) = \hat{f}(\mu)$. Furthermore, let (x, N) realize the definition of $\hat{f}$ for μ, i.e. $\hat{f}(\mu) = f_N(x)$. Consider any $y \in \mathbb{I}^M$ such that $\mu = D(y)$, and define a sequence $y^i = (z_i, y_2, \ldots, y_M)$ where $z_i \to y_1$ and all z_i are distinct from elements of y. Every point $y^i \in \mathbb{I}^M$ has a unique coordinate and therefore $\hat{f}(D(y^i)) = f_M(y^i)$. Because $D(y^i) \rightharpoonup D(y)$, continuity implies $\hat{f}(D(y)) = f_M(y)$. Thus, for any $y \in \mathbb{I}^M$, $\bar{f}(D(y)) = f_M(y)$, which implies $\bar{f}$ is an extension.

Suppose $\mu_n \rightharpoonup \mu$. By the density of discrete measures, we can define sequences $\mu_n^m \rightharpoonup \mu_n$ where $\mu_n^m \in \hat{\mathcal{P}}(\mathbb{I})$. In particular, we may choose these sequences such that for all n, $W_1(\mu_n^m, \mu_n) \leq \frac{1}{m}$. Then for any $\epsilon > 0$,

$$|\bar{f}(\mu) - \bar{f}(\mu_n)| \leq |\bar{f}(\mu) - \hat{f}_\epsilon(\mu)| + |\hat{f}_\epsilon(\mu) - \hat{f}(\mu_n^n)| + |\hat{f}(\mu_n^n) - \hat{f}_\epsilon(\mu_n)| + |\hat{f}_\epsilon(\mu_n) - \bar{f}(\mu_n)| \,. \tag{9}$$

Consider the simultaneous limit as $n \to \infty$ and $\epsilon \to 0$. On the RHS, the first term vanishes by definition, and the fourth by uniform continuity. For any $\nu \in B_\epsilon(\mu) \cap \hat{\mathcal{P}}(\mathbb{I})$, $W_1(\nu, \mu_n^n) \leq W_1(\nu, \mu) + W_1(\mu, \mu_n) + W_1(\mu_n, \mu_n^n) \to 0$ in the limit. So the second term vanishes as well by uniform continuity of $\hat{f}$. Similarly, for any $\nu \in B_\epsilon(\mu_n) \cap \hat{\mathcal{P}}(\mathbb{I})$, $W_1(\nu, \mu_n^n) \leq W_1(\nu, \mu_n) + W_1(\mu_n, \mu_n^n) \to 0$, and the third term vanishes by uniform continuity. This proves continuity of $\bar{f}$.

□

A.2 Proof of Proposition 5.1

Proof. We can decompose the generalization error:

$$\begin{aligned}
&\mathbb{E} \sup_{\|f\|_{\mathcal{S}_1} \leq \delta} \left| \mathbb{E}_{\mu \sim \mathcal{D}} \ell(f^*(\mu), f(\mu)) - \frac{1}{n} \sum_{i=1}^n \ell(f^*(\mu_i), f(\mu_i)) \right| \\
&\leq 2\mathbb{E} \sup_{\|f\|_{\mathcal{S}_1} \leq \delta} \left| \frac{1}{n} \sum_{i=1}^n \epsilon_i \ell(f^*(\mu_i), f(\mu_i)) \right| \\
&\leq 2\mathbb{E} \sup_{\|f\|_{\mathcal{S}_1} \leq \delta} \left| \frac{1}{n} \sum_{i=1}^n \epsilon_i \ell(f^*(\mu_i), 0) \right| + 2\mathbb{E} \sup_{\|f\|_{\mathcal{S}_1} \leq \delta} \left| \frac{1}{n} \sum_{i=1}^n \epsilon_i (\ell(f^*(\mu_i), 0) - \ell(f^*(\mu_i), f(\mu_i))) \right| \\
&\leq \frac{2RG\delta}{\sqrt{n}} + 2\sqrt{2}RG\mathbb{E} \sup_{\|f\|_{\mathcal{S}_1} \leq \delta} \left| \frac{1}{n} \sum_{i=1}^n \epsilon_i f(\mu_i) \right| \,,
\end{aligned}$$

where the second step uses symmetrization through the Rademacher random variable ϵ, and the fourth is by assumption on the loss function ℓ, from the fact that $\|f\|_{\mathcal{S}_1} \leq \delta$ implies $\|f\|_\infty \leq \sqrt{2}R\delta$. We decompose the

Rademacher complexity (removing the absolute value by symmetry):

$$\begin{aligned}\mathbb{E}\left[\sup_{\|f\|_{\mathcal{S}_1}\leq\delta}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i f(\mu_i)\right] &= \mathbb{E}\left[\sup_{\substack{\chi\in\mathcal{M}(\mathcal{A})\\ \|\chi\|_{TV}\leq\delta}}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\int\sigma(\langle\phi,\mu_i\rangle)\chi(d\phi)\right]\\ &= \delta\mathbb{E}\left[\sup_{\gamma_1(\phi)\leq 1}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\sigma(\langle\phi,\mu_i\rangle)\right]\\ &\leq \delta\mathbb{E}\left[\sup_{\gamma_1(\phi)\leq 1}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\langle\phi,\mu_i\rangle\right],\end{aligned}$$

where the last step uses the contraction lemma and that σ is 1-Lipschitz.

Now, using the neural network representation of ϕ:

$$\begin{aligned}\mathbb{E}\left[\sup_{\|f\|_{\mathcal{S}_1}\leq\delta}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i f(\mu_i)\right] &\leq \delta\mathbb{E}\left[\sup_{\|\nu\|_{TV}\leq 1}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\int_{\mathbb{R}^d}\int_{\mathbb{S}^d}\sigma(\langle w,\tilde{x}_i\rangle)\nu(dw)\mu_i(dx_i)\right]\\ &\leq \delta\mathbb{E}\left[\sup_{\|w\|_2\leq 1}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\mathbb{E}_{\mu_i}[\sigma(\langle w,\tilde{x}_i\rangle)]\right]\\ &\leq \delta\mathbb{E}_{\mu_1,\ldots,\mu_n}\left[\mathbb{E}\left[\sup_{\|w\|_2\leq 1}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\sigma(\langle w,\tilde{x}_i\rangle)\middle| x_1,\ldots,x_n\right]\right],\end{aligned}$$

where the last step uses Jensen's inequality and Fubini's theorem. The conditional expectation is itself a Rademacher complexity, so after again peeling the σ activation and using the variational definition of the l_2 norm we have the bound:

$$\mathbb{E}\left[\sup_{\|f\|_{\mathcal{S}_1}\leq\delta}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i f(\mu_i)\right] \leq \frac{\delta\sqrt{2}R}{\sqrt{n}}.$$

The high probability bound then follows from McDiarmid's inequality.

□

A.3 Proof of Proposition 5.2

Proof. We appeal to the following concentration inequality for empirical measures under the Wasserstein metric:

Theorem A.1 (Theorem 1 in [FG15]). *Let $\hat{\mu}_N = \frac{1}{N}\sum_{j=1}^{N}\delta_{X_j}$ where $X_i \sim \mu \in \mathcal{P}(\mathbb{I})$ iid. Then $\mathbb{E}[W_1(\hat{\mu}_N,\mu)] \lesssim N^{-1/d}$ where $d > 2$ is the dimension of $\mathbb{I}$.*

It's easy to see that any $\phi \in \mathcal{A}_2$ has Lipschitz constant bounded above by 1, and therefore $\sup_{\phi\in\mathcal{A}_2}|\langle\phi,\mu-\mu^*\rangle| \leq W_1(\mu,\mu^*)$. Therefore

$$\begin{aligned}\mathbb{E}\left[\sup_{\phi\in\mathcal{A}}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\langle\phi,\mu_i\rangle\right] &\leq \mathbb{E}\left[\sup_{\phi\in\mathcal{A}}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\langle\phi,\mu^*\rangle\right] + \mathbb{E}\left[\sup_{\phi\in\mathcal{A}}\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\langle\phi,(\mu^*-\mu_i)\rangle\right]\\ &\leq \sqrt{2}R\mathbb{E}\left[\left|\frac{1}{n}\sum_{i=1}^{n}\epsilon_i\right|\right] + \mathbb{E}[W_1(\mu_i,\mu^*)]\\ &\lesssim R(n^{-1/2} + \mathbb{E}_{N\sim\Omega}[N^{-1/d}]).\end{aligned}$$

The conclusion then follows from the same Rademacher decomposition as in Proposition 5.1.

□

A.4 Proof of Theorem 4.1

Recall the definitions of planted neurons: As in the vector setting, such functions are given by single neurons: $f_i(\mu) = \widetilde{\sigma}(\langle \phi_i^*, \mu \rangle)$, where $\phi_1^*(x) = \sigma(\langle w^*, x \rangle)$ and $\phi_2^* \in \mathcal{A}_2$. Let $\widetilde{\sigma}(t) = \sigma(t) = \max(0, t)$.

For simplicity, we consider spherical inputs rather than Euclidean inputs, so we consider $k(x, y) = \int_{\mathbb{S}^d} \sigma(\langle w, x \rangle)\sigma(\langle w, y \rangle)\kappa(dw)$ without the $\tilde{x}$ bias terms, and assume $x \in \mathbb{S}^d$. Note that the Euclidean inputs may be seen as a restriction of the spherical inputs to an appropriate spherical cap, see [Bac17a] for details of this construction.

In order to define $\mathcal{S}_3$ when $m = \infty$, it's necessary to define the base measure τ over test functions $\phi \in \mathcal{A}_2$. Since we are only interested in lower bounds, it will suffice to assume that our input measures $\mu \in \mathcal{P}(\mathbb{I})$ are restricted to the convex hull $\mathrm{ConvHull}(\delta_{\boldsymbol{e}_1}, \ldots, \delta_{\boldsymbol{e}_{d+1}})$, where $\{\boldsymbol{e}_i\}_{i=1}^{d+1}$ are the standard basis in $\mathbb{R}^{d+1}$. Therefore we can consider τ as a measure that only depends on the function evaluations $\alpha_\phi := (\phi(\boldsymbol{e}_1), \ldots, \phi(\boldsymbol{e}_{i+1}))$.

Let $\Psi : \mathbb{S}^d \to \mathcal{F}_2$ be any measureable map such that for all i, $\Psi(\alpha)(\boldsymbol{e}_i) = \alpha_i$. As long as the base kernel measure κ has full support on $\mathbb{S}^d$, we can always define some such Ψ. Then we will define τ as the pushforward of the uniform distribution on $\mathbb{S}^d$ by Ψ.

A.4.1 Part I

Theorem A.2. *Consider $m = \infty$. Let κ have full support on $\mathbb{S}^d$ and τ be as above. Then $\inf_{\|f\|_{\mathcal{S}_3} \le \delta} \|f - f_2\|_\infty \gtrsim d^{-1}\delta^{-3/(d-1)}$.*

Proof. For any $f \in \mathcal{S}_3$, let us remind the form of our functions:

$$f_2(\mu) = \sigma(\langle \phi^*, \mu \rangle)$$
$$f(\mu) = \int_{\mathcal{A}_2} \sigma(\langle \phi, \mu \rangle) q(\phi) \tau(d\phi)$$

Let $\alpha^* = (\phi^*(\boldsymbol{e}_1), \ldots, \phi^*(\boldsymbol{e}_{d+1})$, and consider $\mu_z = \sum_{i=1}^{d+1} z_i \delta_{\boldsymbol{e}_i}$ for some z in the simplex Δ^{d+1}. Then from our choice of τ, we may simplify:

$$f_2(\mu_z) = \sigma(\langle \alpha^*, z \rangle)$$
$$f(\mu_z) = \int_{\mathcal{F}_2} \sigma(\langle \alpha_\phi, z \rangle) q(\phi) \tau(d\phi)$$
$$= \int_{\mathbb{S}^d} \sigma(\langle \alpha, z \rangle) q'(\alpha) \tau'(d\alpha)$$

where by definition of the push-forward, τ' is uniform on $\mathbb{S}^d$, and q' is a density such that $\|q'\|_{L_2(d\tau')} = \|q\|_{L_2(d\tau)}$. We will omit the prime notation below.

If z could span all of $\mathbb{S}^d$, we'd be finished (by the separation result cited at the end of the proof). However, we only have that $z \in \Delta^{d+1}$. It remains then to restrict the inputs to a lower dimensional sphere, and prove that f and f_2 retain similar structure. So suppose $\|f - f_2\|_\infty < \epsilon$ for some fixed ϵ. Below, we will use $x \approx y$ to indicate $|x - y| \lesssim \epsilon$, i.e. that $|x - y| \le C\epsilon$ where C is some universal constant independent of the problem parameters.

We may write $g(z) := f(\mu_z)$ and $g_2(z) := f_2(\mu_z)$, defined only for $z \in \Delta^{d+1}$. Let Q be an orthogonal matrix such that $Q(\frac{1}{d+1}\mathbf{1}) = \beta := \frac{1}{\sqrt{d+1}}\boldsymbol{e}_1$. Then by renaming $\alpha^* \mapsto Q\alpha^*$ and $q \mapsto q \circ Q$, we may equivalently assume $z \in Q\Delta^{d+1}$. Note that this operation will not change the L_2 norm of q, nor the infinity norm between g and g_2, due to the rotation invariance of τ.

We take a preparatory step to control the density q. Define a truncated density $\hat{q}_\gamma$ such that for $\alpha \in \mathcal{S}^d$, $\hat{q}_\gamma(\alpha) = q(\alpha)$ when $|\alpha_1| > \gamma$ and 0 otherwise. Then clearly $\|\hat{q}_\gamma\|_{L_2} \le \|q\|_{L_2}$, and by Cauchy-Schwartz:

$$\left| \int_{\mathbb{S}^d} \sigma(\langle \alpha, z \rangle) q(\alpha) - \sigma(\langle \alpha, z \rangle) \hat{q}_\gamma(\alpha) \right| \tau(d\alpha) \le \int_{\mathbb{S}^d} \|z\| |q(\alpha) - \hat{q}_\gamma(\alpha)| \tau(d\alpha)$$
$$= \|z\| \int_{|\alpha_1| \le \gamma} q(\alpha) \tau(d\alpha)$$
$$\le \|z\| \sqrt{\tau(\{\alpha : |\alpha_1| \le \gamma\})} \|q\|_{L_2}$$

Because $\tau(\{\alpha : |\alpha_1| \le \gamma\}) \to 0$ as $\gamma \to 0$, we can choose γ depending on ϵ and q such that the difference above is bounded by ϵ. Thus, it only introduces another additive ϵ error term to assume that q is zero on this narrow band around the equator some sufficiently small γ.

Let $B \subset Q\Delta^{d+1}$ be the $d-1$ sphere centered at β of radius $\frac{1}{d+1}$. Further let $B_0 = B - \beta$ be the shifted sphere centered at the origin, such that $B_0 = \{0\} \times \frac{1}{d+1}\mathbb{S}^{d-1}$. Then for $y \in B_0$, we have $y + \beta$ as a valid input to g and g_2. Finally, we choose $\alpha^* \in \{0\} \times \mathbb{S}^{d-1}$.

We now exploit the homogeniety of σ. For any $y \in B_0$, because B_0 is a centered ball, it follows $y/c \in B_0$ for any $c \ge 1$. Thus, from the assumption of small infinity norm:

$$\begin{aligned} g(y+\beta) \approx g_2(y+\beta) &= \sigma(\langle \alpha^*, y+\beta\rangle) \\ &= c\sigma(\langle \alpha^*, y/c+\beta\rangle) \\ &= cg_2(y/c+\beta) \approx cg(y/c+\beta) \end{aligned}$$

where we use in the second line that $\alpha^* \perp \beta$. Therefore we must have $g(y+\beta) \approx cg(y/c+\beta)$ for all $c \ge 1$. We will use this fact to characterize the density q. Explicitly,

$$\begin{aligned} cg(y/c+\beta) &= c\int_{\mathbb{S}^d} \sigma(\langle \alpha, y/c+\beta\rangle) q(\alpha)\tau(d\alpha) \\ &= \int_{\mathbb{S}^d} \sigma(\langle \alpha, y\rangle + c\langle \alpha, \beta\rangle) q(\alpha)\tau(d\alpha) \end{aligned}$$

To simplify further, we may rewrite in spherical coordinates. Let $\alpha = (\cos\theta, \bar{\alpha}\sin\theta)$ for $\bar{\alpha} \in \mathbb{S}^{d-1}$ and $\theta \in [0,\pi]$. Then $\langle \alpha, \beta\rangle = \frac{1}{\sqrt{d+1}}\cos\theta$, and $\langle \alpha, y\rangle = \langle \bar{\alpha}, \bar{y}\rangle \sin\theta$ where $\bar{y} = (y_2, \dots, y_{d+1}) \in \frac{1}{d+1}\mathbb{S}^{d-1}$. Thus, letting $\bar{\tau}$ be uniform on $\mathbb{S}^{d-1}$, we have:

$$\begin{aligned} cg(y/c+\beta) &= \int_{\mathbb{S}^{d-1}} \int_0^\pi \sigma\left(\langle \bar{\alpha}, \bar{y}\rangle \sin\theta + \frac{c}{\sqrt{d+1}}\cos\theta\right) q(\alpha) d\theta \bar{\tau}(d\bar{\alpha}) \\ &= H_1 + H_2 + H_3 \end{aligned}$$

where each H_i integrates over the corresponding set in the following partition: $P_1 = \{(\bar{\alpha}, \theta) : \theta \in [\pi/2, \pi]\}$, $P_2 = \{(\bar{\alpha}, \theta) : \langle \bar{\alpha}, \bar{y}\rangle \le 0, \theta \in [0, \pi/2]\}$ and $P_3 = \{(\bar{\alpha}, \theta) : \langle \bar{\alpha}, \bar{y}\rangle \ge 0, \theta \in [0, \pi/2]\}$. Finally, we introduce $A_c = \{(\bar{\alpha}, \theta) : \langle \bar{\alpha}, \bar{y}\rangle \sin\theta + \frac{c}{\sqrt{d+1}}\cos\theta \ge 0\}$ as the set where the neuron is active. We consider these H_i terms as $c \to \infty$. The term $\sin\theta$ is always positive in the range $[0, \pi]$, so we focus on the signs of the other terms:

Bounding H_1: On P_1, $\cos\theta \le 0$. Letting λ denote the Lebesgue measure on $\mathbb{R}$, we have $(\bar{\tau} \times \lambda)(A_c) \to 0$ as $c \to \infty$. Furthermore, because $|\langle \bar{\alpha}, \bar{y}\rangle| \le 1$, this convergence is uniform over all $y \in B_0$. So by Cauchy-Schwartz:

$$H_1 \le \sqrt{(\bar{\tau} \times \lambda)(A_c)}\|q\|_{L_2} \to 0$$

Bounding H_2: On P_2, $\cos\theta \ge 0$ but $\langle \bar{\alpha}, \bar{y}\rangle \le 0$. From the fact that $\langle \bar{\alpha}, \bar{y}\rangle \ge -1$, $A_c \subseteq \{(\bar{\alpha}, \theta) : \theta \in [\pi/2 - \arctan(c/\sqrt{d+1}), \pi/2]\}$. By choosing c large enough, it follows $A_c \subseteq \{(\bar{\alpha}, \theta) : |\cos\theta| \le \gamma\}$ where γ is the truncation threshold we defined for the density q. The neuron only activates in this region where the density is zero, so we have $H_2 = 0$.

Bounding H_3: On P_3, the neuron is always active, so in fact we may write:

$$H_3 = \int_{\langle \bar{\alpha}, \bar{y} \rangle \geq 0} \int_0^{\pi/2} \left(\langle \bar{\alpha}, \bar{y} \rangle \sin\theta + \frac{c}{\sqrt{d+1}} \cos\theta \right) q(\alpha) d\theta \bar{\tau}(d\bar{\alpha})$$
$$= \int_{\langle \bar{\alpha}, \bar{y} \rangle \geq 0} \int_0^{\pi/2} \langle \bar{\alpha}, \bar{y} \rangle \sin\theta \; q(\alpha) d\theta \bar{\tau}(d\bar{\alpha}) + c \int_{\langle \bar{\alpha}, \bar{y} \rangle \geq 0} \int_0^{\pi/2} \frac{1}{\sqrt{d+1}} \cos\theta \; q(\alpha) d\theta \bar{\tau}(d\bar{\alpha})$$

The second integral either limits to $\pm\infty$ or exactly equals 0, because the former would contradict the infinity norm approximation the latter must hold. Rewriting the first yields:

$$H_3 = \int_{\mathbb{S}^{d-1}} \sigma(\langle \bar{\alpha}, \bar{y} \rangle) \left(\int_0^{\pi/2} \sin\theta \; q(\alpha) d\theta \right) \bar{\tau}(d\bar{\alpha})$$
$$= \int_{\mathbb{S}^{d-1}} \sigma(\langle \bar{\alpha}, \bar{y} \rangle) t(\bar{\alpha}) \bar{\tau}(d\bar{\alpha})$$

where we've introduced a density t over $\mathbb{S}^{d-1}$. Putting everything together yields the following for all $\bar{y} \in \frac{1}{d+1}\mathbb{S}^{d-1}$:

$$\sigma(\langle \bar{\alpha^*}, \bar{y} \rangle) = \sigma(\langle \alpha^*, y \rangle) = g(y+\beta) \approx g_2(y+\beta) = \int_{\mathbb{S}^{d-1}} \sigma(\langle \bar{\alpha}, \bar{y} \rangle) t(\bar{\alpha}) \bar{\tau}(d\bar{\alpha})$$

Finally, we bound the norm of the density t using Cauchy-Schwartz:

$$\int_{\mathbb{S}^{d-1}} t(\bar{\alpha})^2 \bar{\tau}(\bar{\alpha}) = \int_{\mathbb{S}^{d-1}} \left(\int_0^{\pi/2} \sin\theta \; q(\alpha) d\theta \right)^2 \bar{\tau}(\bar{\alpha})$$
$$\leq \int_{\mathbb{S}^{d-1}} \frac{\pi}{2} \int_0^{\pi/2} (\sin\theta)^2 \; q(\alpha)^2 d\theta \bar{\tau}(\bar{\alpha})$$
$$\lesssim \|q\|^2$$

We've arrived at the conclusion that there exists a density t of controlled norm which approximate a neuron on $\mathbb{S}^{d-1}$ in the infinity norm to within $O(\epsilon)$. By the approximation bound of Appendix D.5 in [Bac17a], $\sup_{\bar{y} \in \frac{1}{d+1}\mathbb{S}^{d-1}} |\sigma(\langle \bar{\alpha^*}, \bar{y} \rangle) - \int_{\mathbb{S}^{d-1}} \sigma(\langle \bar{\alpha}, \bar{y} \rangle) t(\bar{\alpha}) \bar{\tau}(d\bar{\alpha})| \gtrsim \frac{1}{d+1} \|t\|_{L_2}^{-3/(d-1)}$ for any density t. We therefore derive a contradiction if $\epsilon \lesssim \frac{1}{d+1} \|q\|_{L_2}^{-3/(d-1)}$. Equivalently stated via the functional norms and rounding the fractional term, we conclude $\inf_{\|f\|_{\mathcal{S}_3} \leq \delta} \|f - f_2\|_\infty \gtrsim d^{-1} \delta^{-3/(d-1)}$.

□

A.4.2 Part II

We let κ be the uniform distribution on $\mathbb{S}^d$. We will make frequent use of the identity $2 \int_{\mathbb{S}^d} x_1 \sigma(x_1) \kappa(dx) = \int_{\mathbb{S}^d} x_1^2 \kappa(dx) = \frac{1}{d}$.

For a fixed $w^* \in \mathbb{S}^d$, we define $K_0 \in \mathbb{R}$ as

$$K_0 = \int_{\mathbb{S}^d} \sigma(\langle w^*, x \rangle) \kappa(dx)$$

We may calculate the first spherical harmonic of $\sigma(\langle w^*, x \rangle)$, let Q be orthogonal such that $Qw^* = e_1$, then

$$d\int_{\mathbb{S}^d} x\sigma(\langle w^*, x\rangle)\kappa(dx) = dQ^{-1}\int_{\mathbb{S}^d} x\sigma(x_1)\kappa(dx) = \frac{1}{2}w^*$$

We then define $\phi_1^*(x) = \sigma(\langle w^*, x\rangle) - \frac{1}{2}\langle w^*, x\rangle$. Note that $\gamma_1(\phi_1^*) \le 2$.

Theorem A.3. *Consider $m = \infty$. Let $f_1(\mu) = \sigma(\langle \phi_1^*, \mu\rangle)$, then for sufficiently large d,*

$$\inf_{\|f\|_{\mathcal{S}_2}\le\delta} \|f - f_1\|_\infty \gtrsim |d^{-1} - \delta 2^{-d/2}|\,,$$

Proof. From [Bac17a], Appendix D.5, we verify that $\phi_1^* - K_0$ has no zeroth or first spherical harmonic, and any ϕ with $\gamma_2(\phi) \le \delta$ will have a correlation $\langle \phi, \phi_1^* - K_0\rangle_{L^2}$ of at most $\simeq \delta 2^{-d/2}$.

We introduce the input measure with density:

$$\mu^*(dx) = \frac{2\phi_1^*(x) + 1}{2K_0 + 1}\kappa(dx)\,.$$

We verify that μ^* is a probability measure, following from the fact that

$$\int \phi_1^*(x)\kappa(dx) = K_0 \quad \text{and} \quad \sup_x |\phi_1^*(x)| = \frac{1}{2}\,.$$

We will use μ^* and κ as inputs to distinguish f_1 from any $f \in \mathcal{S}_2$ with bounded norm. First, note that

$$\begin{aligned} f_1(\mu^*) &= \sigma(\langle \phi_1^*, \mu^*\rangle) = \frac{2\|\phi_1^*\|_{L_2}^2 + K_0}{2K_0 + 1}\,, \\ f_1(\kappa) &= \sigma(\langle \phi_1^*, \kappa\rangle) = K_0\,. \end{aligned}$$

We calculate by rotation invariance

$$\begin{aligned} \|\phi_1^*\|_{L_2}^2 &= \int_{\mathbb{S}^d} \left(\sigma(\langle w^*, x\rangle) - \frac{1}{2}\langle w^*, x\rangle\right)^2 \kappa(dx) \\ &= \int_{\mathbb{S}^d} \left(\sigma(x_1) - \frac{x_1}{2}\right)^2 \kappa(dx) \\ &= \int_{\mathbb{S}^d} \sigma(x_1)^2 - x_1\sigma(x_1) + \frac{x_1^2}{4}\kappa(dx) \\ &= \frac{1}{4d}\,, \end{aligned}$$

By rotation invariance and Lemma A.4, $K_0 \sim \frac{1}{\sqrt{2\pi d}}$. Therefore for sufficiently large d, we can lower bound the distance between the test points under the planted neuron, $|f_1(\kappa) - f_1(\mu^*)| \gtrsim |\|\phi_1^*\|_{L_2}^2 - K_0^2\| \gtrsim d^{-1}$.

Meanwhile,

$$\begin{aligned} f(\mu^*) &= \int \sigma\left(\frac{2}{2K_0 + 1}\langle \phi, \phi_1^* - K_0\rangle_{L_2} + \langle \phi, \kappa\rangle\right)\chi(d\phi) \\ f(\kappa) &= \int \sigma(\langle \phi, \kappa\rangle)\chi(d\phi)\,. \end{aligned}$$

By definition, χ is only supported on ϕ such that $\gamma_2(\phi) \le 1$, so it follows $|\langle \phi, \phi_1^* - K_0\rangle_{L_2}| \lesssim 2^{-d/2}$, hence because σ is Lipschitz it follows $|f(\mu^*) - f(\kappa)| \lesssim \|f\|_{\mathcal{S}_2} 2^{-d/2}$.

Finally, we conclude from the triangle inequality that $\inf_{\|f\|_{\mathcal{S}_2}\le\delta} \sup_\mu |f(\mu) - f_1(\mu)| \gtrsim |d^{-1} - \delta 2^{-d/2}|$.

We also note that the same lower bound holds for $\phi_1^*(x) = \sigma(\langle w^*, x\rangle)$. Suppose not, then $f \in \mathcal{S}_2$ could efficiently approximate $\sigma(\langle w^*, x\rangle)$ and $\sigma(\langle -w^*, x\rangle)$ (following from rotational symmetry on $\mathbb{S}^d$). But $\sigma(\langle w^*, x\rangle) - \frac{1}{2}\langle w^*, x\rangle = \sigma(\langle w^*, x\rangle) - \frac{1}{2}(\sigma(\langle w^*, x\rangle) - \sigma(\langle -w^*, x\rangle))$, which would imply this function is also well approximated by $\mathcal{S}_2$.

□

Lemma A.4. *Let κ be the uniform distribution on $\mathbb{S}^d$ and $w \sim \kappa$, then $\mathbb{E}[\sigma(w_1)]$ scales asymptotically as $\frac{1}{\sqrt{2\pi d}}$.*

Proof. Introducing a Gaussian random variable $g \sim \mathcal{N}(0, I_{d+1})$, we have:

$$\begin{aligned}\mathbb{E}[\sigma(w_1)] &= \frac{1}{2}\mathbb{E}[|w_1|] \\ &= \frac{1}{2}\mathbb{E}\left[\left|\frac{g_1}{\|g\|}\right|\right] .\end{aligned}$$

One can verify that the probability density function for $\left|\frac{g_1}{\|g\|}\right|$ is

$$p(x) = \frac{2(1-x^2)^{\frac{d-2}{2}}}{B(\frac{d}{2}, \frac{1}{2})} ,$$

where B denotes the Beta function. Then integrating yields

$$\begin{aligned}\mathbb{E}[\sigma(w_1)] &= \frac{1}{2}\int_0^1 \frac{2x(1-x^2)^{\frac{d-2}{2}}}{B(\frac{d}{2}, \frac{1}{2})} \\ &= \frac{1}{B(\frac{d}{2}, \frac{1}{2})d} \\ &= \frac{\Gamma(\frac{d+1}{2})}{\Gamma(\frac{d}{2})\Gamma(\frac{1}{2})d}\end{aligned}$$

The gamma function satisfies $\Gamma(\frac{1}{2}) = \sqrt{\pi}$ and $\frac{\Gamma(\frac{d+1}{2})}{\Gamma(\frac{d}{2})} \sim \sqrt{\frac{d}{2}}$ so the asymptotic bound follows. □

B Experimental Details and Additional Data

Synthetic Details: For all experiments we use the same architecture. Namely, for an input set $x = (x_1, \ldots, x_N)$, the network is defined as $f_N(x) = w_3^T \sigma(W_2 \frac{1}{N}\sum_{i=1}^N \sigma(W_1 \tilde{x}_i))$, where we choose the architecture as $W_1 \in \mathbb{R}^{h_1 \times d}$, $W_2 \in \mathbb{R}^{h_2 \times h_1}$, and $w_3 \in \mathbb{R}^{h_2}$ where $h_1, h_2 = 100$. The weights are initialized with the uniform Kaiming initialization [He+15] and frozen as described in Table 1.

We relax the functional norm constraints to penalties, by introducing regularizers of the form $\lambda \|f_N\|_{\mathcal{S}_i}$ for λ a hyperparameter. Let $K(\cdot)$ map a matrix to the vector of row-wise norms, and let $|\cdot|$ denote the element-wise absolute value of a matrix. Then we calculate the functional norms via the path norm as follows:

- For $\mathcal{S}_1$, $\|f_N\|_{\mathcal{S}_1} = |w_3|^T |W_2| K(W_1)$
- For $\mathcal{S}_2$, we explicitly normalize the frozen matrix W_1 to have all row-wise norms equal to 1, then $\|f_N\|_{\mathcal{S}_2} = |w_3|^T K(W_2)$
- For $\mathcal{S}_3$, we normalize the rows of W_1 and W_2, which simply implies $\|f_N\|_{\mathcal{S}_3} = \|w_3\|_2$

We optimized via Adam [KB14] with an initial learning rate of 0.003, for 5000 iterations. Under this architecture, all $\mathcal{S}_1$ and $\mathcal{S}_2$ functions achieved less than 10^{-7} mini-batch training error without regularization on all objective functions (listed below) on training sets of 100 samples. The $\mathcal{S}_3$ functions achieved less than 10^{-2} training error, although we note that without exponentially large width, this error is lower bounded following from Theorem 4.1.

We use the following symmetric functions for our experiments:

- $f_N^*(x) = \max_i(\|x_i\|_2)$
- $f_N^*(x) = \lambda \log\left(\sum_{i=1}^N \exp(\|x_i\|_2/\lambda)\right)$ for $\lambda = 0.1$

- $f_N^*(x) = \text{median}(\{\|x_i\|_2\}_{i=1}^N)$
- $f_N^*(x) = \text{second}_i(\|x_i\|_2)$ i.e. the second largest value in a given set
- $f_N^*(x) = \frac{1}{N}\sum_{i=1}^N(\|x_i\|_2)$
- $f_N^*(x)$ is an individual neuron, parameterized the same as f_N but with $h_1 = 100$, $h_2 = 1$. Additionally, for the "neuron" we initialize W_1 elementwise from the distribution $U(-5.0, 5.0)$, and for the "smooth_neuron" we initialize W_2 from $U(-0.3, 0.3)$.

Note that in order to guarantee the "smooth_neuron" is representable by our finite-width networks, we explicitly set W_1 in the $\mathcal{S}_2$ and $\mathcal{S}_3$ models to equal the W_1 matrix of the "smooth_neuron".

For each model in each experiment, λ was determined through cross validation over $\lambda \in [0, 10^{-6}, 10^{-4}, 10^{-2}]$ using fresh samples of training data, and choosing the value of λ with lowest generalization error, which was calculated from another 1000 sampled points.

Then, with determined λ, each model was trained from scratch over 10 runs with independent random initializations. The mean and standard deviation of the generalization error, testing on varying values of N, are plotted in Figure 1.

Application Details: For the MNIST experiment, we follow a similar setup to [DPC19]. From an image in $\mathbb{R}^{28\times 28}$, we produce a point cloud by considering a set of tuples of the form (r, c, t), which are the row, column and intensity respectively for each pixel. We restrict to pixels where $t > 0.5$, and select the pixels with the top 200 intensities to comprise the point cloud (if there are fewer than 200 pixels remaining after thresholding, we resample among them). Furthermore, we normalize the row and column values among all the points in the cloud. This process maps an image to a set $S \subseteq \mathbb{R}^3$ such that $|S| = 200$.

For this dataset we consider $h_1 = 500$ and $h_2 = 1000$ for our $\mathcal{S}_i$ finite-width architectures. For the 5-layer network baseline, we use the architecture $f_N(x) = \rho\left(\frac{1}{N}\sum_{i=1}^N \Phi(x_i)\right)$ where $\Phi(x_i) = \sigma(W_2\sigma(W_1\tilde{x}))$ and $\rho(z) = w^T\sigma(W_4\sigma(W_3 z))$, where the hidden layers have width 500 in Φ and 1000 in ρ. We also apply Dropout [Sri+14] after the activations in ρ with probability 0.6.

We perform cross-validation by setting aside 10% of the data as a validation set, and calculate the mean and standard deviation of the generalization error over five runs. In order to study generalization in this setting, we test on point clouds of different size, 100 and 200, and show the results in Table 2. The starting learning rate is 0.001. Otherwise, all other experimental details are the same as above.

Additional Experiments In Figure 4 we plot all the symmetric objectives over both the narrow and wide distribution.In Figure 5 we consider higher dimensional vectors for our set inputs to the symmetric models.

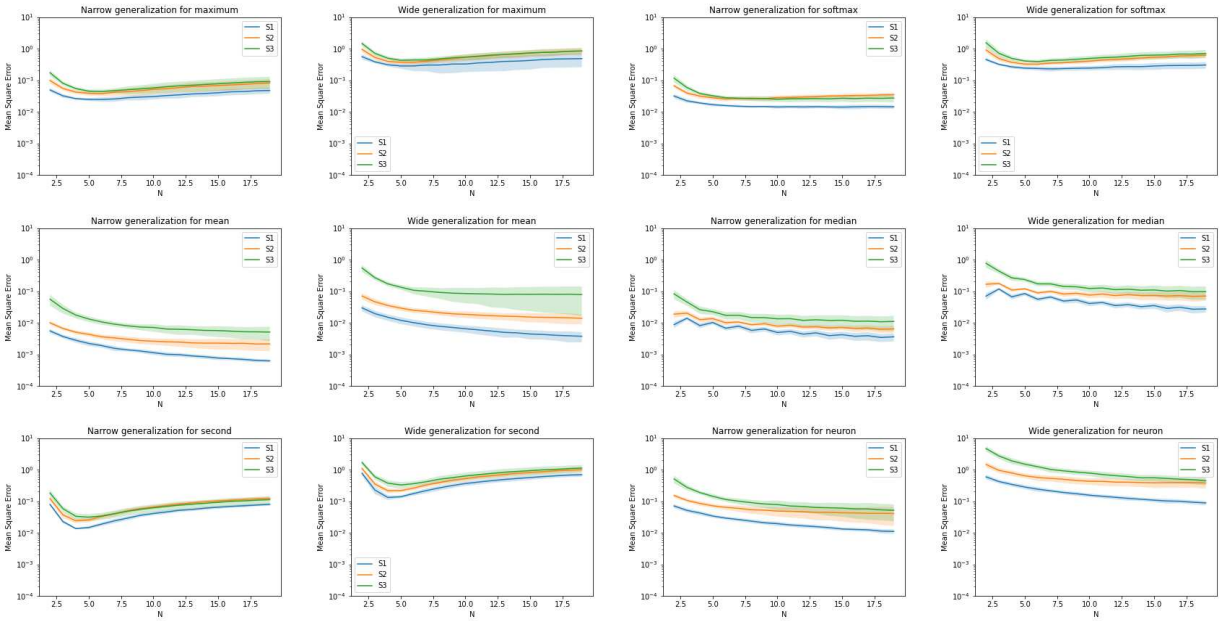

Figure 4: Test Error for $d = 10$ and $m = 100$ on the neural architectures of Section 3.1

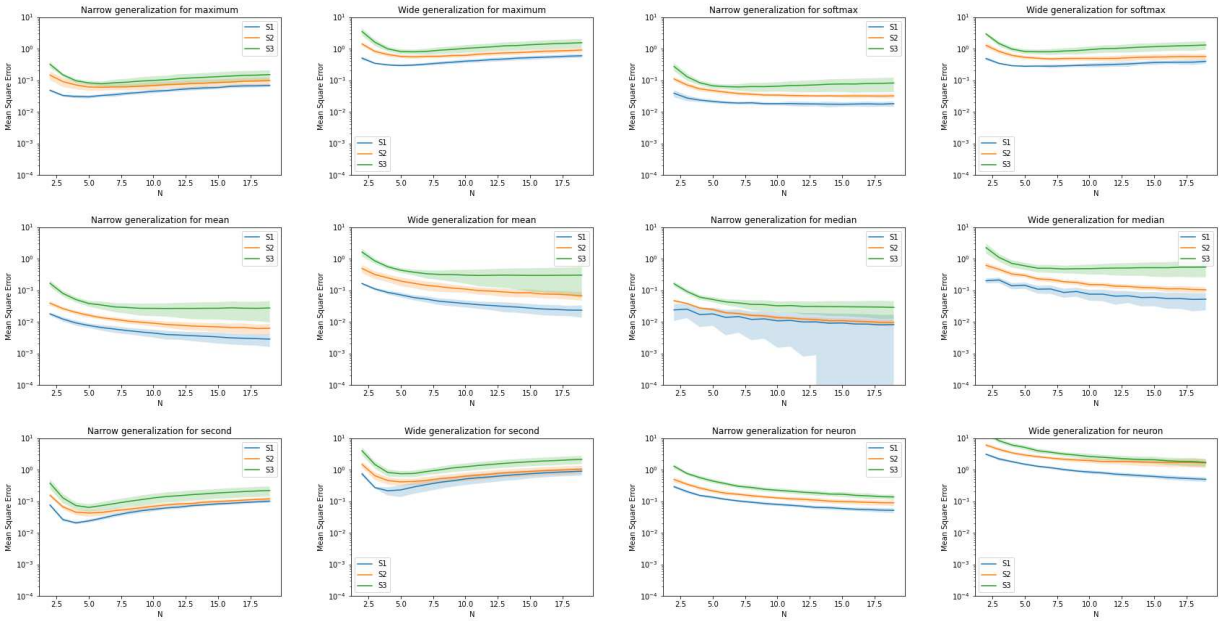

Figure 5: Test Error for $d = 20$ and $m = 100$