Neural Computing and Applications
https://doi.org/10.1007/s00521-018-3680-1

S.I. : EANN 2017

Limitations of shallow networks representing finite mappings

Věra Kůrková[1]

Received: 8 January 2018 / Accepted: 9 August 2018

Abstract
Limitations of capabilities of shallow networks to efficiently compute real-valued functions on finite domains are investigated. Efficiency is studied in terms of network sparsity and its approximate measures. It is shown that when a dictionary of computational units is not sufficiently large, computation of almost any uniformly randomly chosen function either represents a well-conditioned task performed by a large network or an ill-conditioned task performed by a network of a moderate size. The probabilistic results are complemented by a concrete example of a class of functions which cannot be efficiently computed by shallow perceptron networks. The class is constructed using pseudo-noise sequences which have many features of random sequences but can be generated using special polynomials. Connections to the No Free Lunch Theorem and the central paradox of coding theory are discussed.

Keywords Shallow and deep networks · Sparsity · Variational norms · Functions on large finite domains · Finite dictionaries of computational units · Pseudo-noise sequences · Perceptron networks

1 Introduction

To identify and explain efficient network designs, it is necessary to develop a theoretical understanding to the influence of a proper choice of network architecture and a type of units on reducing network complexity. Bengio and LeCun [6], who recently revived the interest in deep networks, conjectured that "most functions that can be represented compactly by deep architectures cannot be represented by a compact shallow architecture". On the other hand, a recent empirical study demonstrated that shallow networks can learn some functions previously learned by deep ones using the same numbers of parameters as the original deep networks [1].

While experimental research of deep networks is rapidly evolving, theoretical analysis complementing the experimental evidence is still in its early stages. There are fundamental wide open questions related to the role of depth of network architectures asking: Why should deep networks be better than shallow ones and under which conditions?

Bianchini and Scarselli [8] proposed a promising approach to investigation of complexity of shallow and deep networks based on topological characteristics of input–output functions using the concept of the Betti Numbers from algebraic topology. Mhaskar et al. [34, 35] suggested that due to their hierarchical structure, deep networks could outperform shallow networks in visual recognition of pictures with objects of different scales and compared VC-dimensions of shallow and deep networks.

Generally, derivation of lower bounds on network complexity is much more difficult than estimates of upper ones. Poggio et al. [37] proposed as a potential tool for comparison of deep and shallow networks an application of the topological approach for obtaining lower bounds on complexity of shallow networks exhibiting the "curse of dimensionality" (i.e., an exponential dependence on the number of parameters [5]) from [12]. However, its applicability is limited to classes of networks where best or near-best approximation of functions can be obtained by a continuous selection of network parameters. We proved in [19–21] that in many common classes of networks such continuous selection is not possible due to their nonlinear and non-convex nature.

This work was partially supported by the Czech Grant Foundation Grants GA15-18108S, GA18-23827S and institutional support of the Institute of Computer Science RVO 67985807.

✉ Věra Kůrková
vera@cs.cas.cz

[1] Institute of Computer Science, Czech Academy of Sciences, Pod Vodárenskou věží 2, 182 07 Prague, Czech Republic

Published online: 17 August 2018

In [7], it was suggested that a cause of large model complexities of shallow networks might be in the "amount of variations" of functions to be computed. As an example of a highly-varying function, the parity function on the Boolean cube was presented and it was proven that classification of points from the d-dimensional Boolean cube by Gaussian SVM requires at least 2^{d-1} support vectors. In [29], we showed that the concept of a highly-varying function has to be studied in dependence on a type of computational units. We proposed to formalize using a concept of variational norm tailored to a type of computational units, which has been used as tool for estimates of rates of approximation by neural networks [3, 23, 24]. Using probabilistic arguments based on Chernoff–Hoeffding bound, we derived in [29] lower bounds on variational norms and in [31] on errors of approximation of binary-valued functions (representing binary classification tasks) by shallow networks. In [26] we complemented probabilistic results by constructing binary-valued functions with large variational norms with respect to the dictionary of perceptrons.

It has long been known that from the point of view of expressibility one hidden layer is sufficient. Shallow networks with merely one hidden layer formed by computational units of many common types can approximate within any accuracy any reasonable function on a compact domain and can exactly compute any function on a finite domain (see, e.g., [18, 36]). Theorems proving such universality type results do not imply any estimates of network complexity as they assume that numbers of network units are potentially infinite or, in the case of finite domains, at least as large as sizes of the domains. However, a proper choice of a network architecture together with a type of computational units could considerably reduce network complexity. Various measures of network simplicity have been promoted by regularization techniques, such as the weight-decay (see, e.g., [15]).

In this paper, we investigate efficiency of computation of real-valued functions on finite domains by shallow networks with units from finite dictionaries. In practical applications, domains of functions to be computed are finite (such as discretized cubes, pixels of images), but their sizes and/or input dimensions d can be quite large. Also dictionaries are formed by parameterized families of functions with finite sets of parameters. As minimization of the numbers of nonzero output weights representing the basic measure of network sparsity is a difficult non-convex task, we investigate minima of its convex approximation by l_1-norm. Large lower bounds on this minima imply a large model complexity or non-stability of a computation caused by ill-conditioning. Both are not desirable as networks with large numbers of units might require too large resources for an implementation, while large output weights might amplify small changes of inputs and thus lead to non-stability of computation.

We derive lower bounds on l_1-norms of output-weight vectors of shallow networks in terms of variational norms tailored to dictionaries of computational units. Combining a geometrical characterization of variational norms with the properties of high-dimensional Euclidean spaces, we show that on large domains most functions have large variations with respect to some common dictionaries (such as signum perceptrons and kernel units used in SVM). More generally, we prove that this holds for all dictionaries of sizes bounded by $e^{p(\ln m)}$, where p is a polynomial and m is the size of the domain. Our results extend to real-valued functions lower bounds which we derived in [29] for binary classification using probabilistic arguments based on the Chernoff Bound on sums of independent random variables. We illustrate our probabilistic result by a concrete class of functions constructed using circulant matrices generated by pseudo-noise sequences. We discuss the effect of pseudo-randomness on network complexity. The paper is an extended version of a conference paper [25].

The paper is organized as follows. Section 2 contains basic concepts on shallow networks and dictionaries of computational units. In Sect. 3, sparsity is investigated in terms of l_1-norm and norms tailored to computational units. In Sect. 4, lower bounds on sparsity are derived in terms of sizes of dictionaries and sizes of finite domains of functions to be computed. In Sect. 5, probabilistic results are complemented by constructive ones. Section 6 contains a discussion.

2 Preliminaries

A *one-hidden-layer (shallow) network with a single linear output* computes input–output functions belonging to the set of the form

$$\operatorname{span}_n G := \left\{ \sum_{i=1}^{n} w_i g_i \,|\, w_i \in \mathbb{R},\ g_i \in G \right\},$$

where the coefficients w_i are called *output weights*, n denotes the number of network units, and G is a parameterized set of computational units called a *dictionary*. Dictionaries are parameterized families of functions of the form

$$G_\phi(X, Y) := \{\phi(\cdot, y) : X \to \mathbb{R} \,|\, y \in Y\},$$

where $\phi : X \times Y \to \mathbb{R}$ is a function of two variables: an input vector $x \in X \subseteq \mathbb{R}^d$ and a parameter vector

$y \in Y \subseteq \mathbb{R}^s$. When the set of parameters is the whole $\mathbb{R}^s$, we write shortly $G_\phi(X)$.

A common type of a computational unit is *perceptron*, which computes functions of the form $\sigma(v \cdot . + b) : X \to \mathbb{R}$, where $\sigma : \mathbb{R} \to \mathbb{R}$ is an *activation function*. It is called *sigmoid* when it is monotonic increasing and $\lim_{t\to-\infty} \sigma(t) = 0$ and $\lim_{t\to\infty} \sigma(t) = 1$. Important types of activation functions are the *Heaviside function* defined as

$$\vartheta(t) := 0 \quad \text{for } t<0 \quad \text{and} \quad \vartheta(t) := 1 \quad \text{for } t \geq 0$$

and the *signum function* $\mathrm{sgn} : \mathbb{R} \to \{-1, 1\}$, defined as

$$\mathrm{sgn}(t) := -1 \text{ for } t<0 \quad \text{and } \mathrm{sgn}(t) := 1 \quad \text{for } t \geq 0.$$

We denote by $G_P(X)$ the dictionary of functions on X computable by *signum perceptrons*, i.e.,

$$G_P(X) := \{\mathrm{sgn}(v \cdot . + b) : X \to \{-1, 1\} \,|\, v \in \mathbb{R}^d, b \in \mathbb{R}\}. \tag{1}$$

Note that from the point of view of the number of network units, there is only a minor difference between networks with signum and Heaviside perceptrons as

$$\mathrm{sgn}(t) = 2\vartheta(t) - 1 \quad \text{and} \quad \vartheta(t) = \frac{\mathrm{sgn}(t) + 1}{2}. \tag{2}$$

It is more convenient to consider the dictionary of signum perceptrons instead of Heaviside ones because all signum perceptrons have the same norms equal to $\sqrt{m}$, where m is the size of the domain X.

Another important class of dictionaries is formed by sets of kernel units. For $X, U \subseteq \mathbb{R}^d$ and a symmetric positive semidefinite kernel $K : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}$, we denote by

$$G_K(X, U) := \{K(., u) : X \to \mathbb{R} \,|\, u \in U\}$$

the *dictionary of kernel units on X with parameters (centers) in U*. When $X = U$, we write shortly $G_K(X)$. In the support vector machine (SVM) algorithm, the set $U = \{u_i, \,|\, i = 1, \ldots, l\}$ is the set of points to be classified, among which some play the role of support vectors. The number of units in the trained network is equal to the number of support vectors.

For a domain $X \subset \mathbb{R}^d$ we denote by

$$\mathcal{F}(X) := \{f \,|\, f : X \to \mathbb{R}\}$$

the *set of all real-valued functions on X*.

It is easy to see that when X is finite with $\mathrm{card}\, X = m$ and $X = \{x_1, \ldots, x_m\}$ is a linear ordering of X, then the mapping $\imath : \mathcal{F}(X) \to \mathbb{R}^m$ defined as $\imath(f) := (f(x_1), \ldots, f(x_m))$ is an isomorphism. So, on $\mathcal{F}(X)$ we have the Euclidean inner product and the norm defined as

$$\langle f, g\rangle := \sum_{u\in X} f(u)g(u) \qquad \|f\| := \sqrt{\langle f, f\rangle}. \tag{3}$$

Table of symbols

$\mathbb{R}^d$	d-dimensional Euclidean space
X	Finite subset of $\mathbb{R}^d$
$\mathcal{F}(X)$	Set of real-valued functions on X
$\langle ., . \rangle$	Inner product on $\mathcal{F}(X)$
$S_1(X)$	Unit ball in $\mathcal{F}(X)$
$C(g, \eta)$	Polar cap centered at g with angle $\arccos \eta$
μ	Uniform probability measure on $S_1(X)$
l_1	l_1-norm
l_0	l_0-pseudo-norm
ϑ	Heaviside function
sgn	Signum function
G	Dictionary of computational units
span G	Linear span of G
conv G	Convex hull of G
$G^\perp$	Orthogonal complement of G
$\|.\|_G$	G-variation
$G_\phi(X, Y)$	Dictionary of computational units on X computing ϕ with parameters in Y
$G_P(X)$	Dictionary of signum perceptrons on X
$G_K(X, U)$	Dictionary of kernel units on X with parameters in U
$L_k(\alpha)$	Matrix induced by a kth-order pseudo-noise sequence

3 Sparsity of shallow networks

It has long been known that shallow networks with many types of computational units have the *universal representation property*, i.e., they can exactly compute any function on a finite domain. Ito [18] proved a mild condition sufficient for this property.

Theorem 1 *Let $X = \{x_1, \ldots, x_m\} \subset \mathbb{R}^d$ and $G_\phi(X, Y) = \{\phi(\cdot, y) : X \to \mathbb{R} \,|\, y \in Y\}$ be a dictionary such that there exist $y_1, \ldots, y_m \in Y$ for which the $m \times m$ matrix Φ defined for every $i, j = 1, \ldots, m$ as $\Phi_{i,j} := \phi(x_i, y_j)$ is non-singular. Then, every $f : X \to \mathbb{R}$ can be expressed as $f(x) = \sum_{j=1}^m \phi(x, y_j)$.*

In [18], it was proven that dictionaries of perceptrons with any sigmoidal activation function satisfy the assumptions of Theorem 1. It is easy to verify that this condition holds for all strictly positive-definite kernel networks and RBF networks of a variety of types. Argument of Theorem 1 is based on a solution of a family of linear equation, and thus it assumes that the number of network

units is potentially as large as the size $\operatorname{card} X = m$ of the domain.

However, a proper choice of network units can reduce this number considerably. For example, computation of parities on d-dimensional Boolean cubes $\{0,1\}^d$ by Gaussian SVM networks requires at least 2^{d-1} units (support vectors) [7], while parities (as well as generalized parities and symmetric functions) can be expressed by shallow networks with Heaviside or signum perceptrons with merely d units [29].

If a dictionary G satisfies assumptions of Theorem 1, then for any finite domain $X \subset \mathbb{R}^d$ with $\operatorname{card} X = m$, any function $f : X \to \mathbb{R}$ can be expressed as

$$f = \sum_{i=1}^{m} w_i g_i \tag{4}$$

where $g_1, \ldots, g_m \in G$ and $w_1, \ldots, w_m \in \mathbb{R}$. Typically, a representation (4) of a function on a finite domain as an element of $\operatorname{span}_m G$ is not unique, and there are several *functionally equivalent networks*. In contrast, on infinite domains often uniqueness (up to permutations of units or sign flipping) holds, see, e.g., [28] and references therein. Equality of two functions on a finite set is a much weaker requirement than equality on $\mathbb{R}^d$ or on its infinite compact subsets.

It is desirable that networks are chosen in such a way that they can compute given tasks using reasonably small numbers of units. In applied mathematics, the number of nonzero entries of a vector $w \in \mathbb{R}^m$ is called "l_0-*pseudo-norm*" and denoted $\|w\|_0$. The quotation marks are used because $\|w\|_0$ is neither a norm nor a pseudo-norm. It lacks the homogeneity property. The basic measure of sparsity of a representation of a function f as an input–output function of a shallow network with units from a dictionary G is the minimum of the number of nonzero of output weights over all representations of f in the form (4), i.e.,

$$\min\left\{ \|w\|_0 \,|\, f = \sum_{i=1}^{m} w_i g_i, w_i \in \mathbb{R}, g_i \in G \right\}. \tag{5}$$

On can easily check that for a finite dictionary, the infimum of $\|w\|_0$ is achieved and thus we write min instead of inf.

As the "unit ball" in "l_0-pseudo-norm" is non-convex and unbounded, minimization of "l_0-pseudo-norms" of output-weight vectors over all representations of a function f in the form (4) is a difficult non-convex optimization problem. In some cases studied in signal processing, it was proven that minimization of l_0 is NP-hard [41].

However, l_0 can be approximated by l_p-functionals. Indeed,

$$\lim_{p \to 0} \|w\|_p = \|w\|_0 ,$$

where

$$\|w\|_p{}^p = \sum_{i=1}^{m} |w_i|^p .$$

For $p < 1$, unit balls of l_p-functionals are concave, and thus the smallest p for which the unit ball is convex is $p = 1$ (see Fig. 1). Optimization problems based on representation with minimal l_1-norms are much easier to handle than the one related to "l_0-pseudo-norm" [14, 17]. In some cases, even a solution with the minimal l_1-norm is the sparsest solution [13].

In neurocomputing, l_1-norm has been used as a stabilizer in weight-decay regularization techniques [15, 16, 27, 42]. A network with a large l_1-norm of its output-weight vector must have either a large number of units or some output weights must be large. Both of these properties are not desirable as they imply either a large model complexity or non-stability of the computation caused by ill-conditioning. When some of the output weights of a network are large, small errors in input data or small change of parameters of hidden units can lead to large differences in the network output.

Thus instead of minimization of the form (5), we focus on the minimization

$$\min\left\{ \|w\|_1 \,|\, f = \sum_{i=1}^{m} w_i g_i, w_i \in \mathbb{R}, g_i \in G \right\}. \tag{6}$$

An advantage of this minimum is that it can be studied in terms of the Minkowski functional of the convex body $\operatorname{conv}(G \cup -G)$ (where conv denotes the convex hull). It is easy to verify the following proposition.

Proposition 1 *Let G be a finite subset of $\mathcal{F}(X)$ with $\operatorname{card} G = k$. Then for every $f \in \mathcal{F}(X)$*

$$\min\left\{ \|w\|_1 \,|\, f = \sum_{i=1}^{m} w_i g_i, w_i \in \mathbb{R}, g_i \in G \right\}$$
$$= \min\{c > 0 \,|\, f/c \in \operatorname{conv}(G \cup -G)\}.$$

The Minkowski functional of a symmetric convex set generates a norm. For a general normed linear space $(\mathcal{X}, \|.\|)$ and its bounded subset G, the norm generated by $\operatorname{cl}\operatorname{conv}(G \cup -G)$ is called G-variation and denoted $\|.\|_G$. So

$$\|f\|_G := \inf\left\{ c \in \mathbb{R}_+ \;\middle|\; f/c \in \operatorname{cl}_{\mathcal{X}} \operatorname{conv}(G \cup -G) \right\},$$

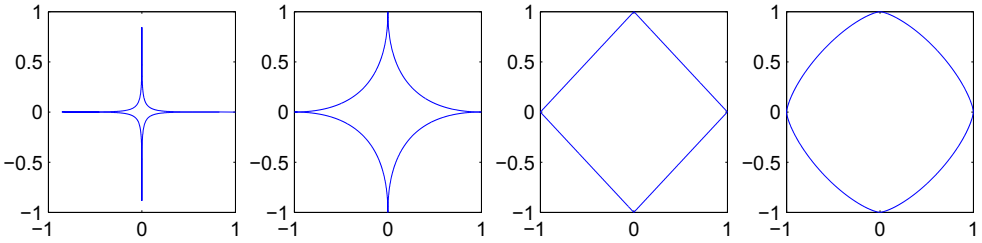

Fig. 1 Balls in l_p

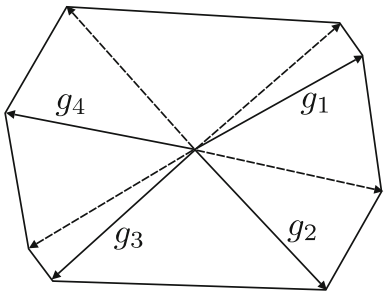

Fig. 2 Unit ball in G-variation

where $-G := \{-g \mid g \in G\}$, $\mathrm{cl}_{\mathcal{X}}$ denotes the closure with respect to the topology induced by the norm $\|\cdot\|_{\mathcal{X}}$ (see Fig. 2).

Variation with respect to the dictionary of Heaviside perceptrons (called *variation with respect to half-spaces*) was introduced by Barron [3], and we extended it to general sets in [23]. It plays an important role in nonlinear approximation theory (see, e.g., [22]).

As G-variation is a norm, it can be made arbitrarily large by multiplying a function by a scalar. Also in theoretical analysis of approximation capabilities of shallow networks, it has to be taken into account that the approximation error $\|f - \mathrm{span}_n G\|$ in any norm $\|.\|$ can be made arbitrarily large by multiplying f by a scalar. Indeed, for every $c > 0$, $\|cf - \mathrm{span}_n G\| = c\|f - \mathrm{span}_n G\|$. Thus, both G-variation and errors in approximation by $\mathrm{span}_n G$ have to be studied either for sets of normalized functions or for sets of functions of a given fixed norm.

A small l_1-norm of an output-weight vector guarantees that an input–output function of a network can be well approximated by input–output functions computable by networks with small "l_0-pseudo-norms". This follows from the Maurey–Jones–Barron Theorem [4]. Here, we state a version of its reformulation from [23, 24] in terms of G-variation for the special case of the finite-dimensional Hilbert space $\mathcal{F}(X)$ with the Euclidean norm. By G^o is denoted the set of normalized elements of G, i.e., $G^o = \left\{\frac{g}{\|g\|} \mid g \in G\right\}$.

Theorem 2 *Let $X \subset \mathbb{R}^d$ be finite, G be a finite subset of $\mathcal{F}(X)$, $s_G = \max_{g\in G} \|g\|$, and $f \in \mathcal{F}(X)$. Then for every n,*

$$\|f - \mathrm{span}_n G\| \leq \frac{\|f\|_{G^o}}{\sqrt{n}} \leq \frac{s_G \|f\|_G}{\sqrt{n}}.$$

Theorem 2 implies that there exists an input–output function $f_n = \sum_{i=1}^n w_i g_i$, i.e., $\|w\|_0 \leq n$, such that $\|f - f_n\| \leq \frac{s_G \|f\|_G}{\sqrt{n}}$.

4 Probabilistic lower bounds on variation

In this section, we derive lower bounds on variational norms of functions on finite domains in terms of sizes of the domains and sizes of dictionaries. The main theorem of this section is obtained by combining a geometric characterization of the variational norm with rather counter-intuitive geometrical properties of high-dimensional Euclidean spaces.

The following theorem based on a separation of a function from a convex set by a linear functional (the Hahn–Banach Theorem) shows that functions which are "nearly orthogonal" to all elements of a dictionary G have large G-variations (see Fig. 3). Here, we state a special case following from general versions proven in [24, 30] for the finite-dimensional space $\mathcal{F}(X)$. By $G^\perp$ is denoted the orthogonal complement of G, i.e., $G^\perp = \{f \in \mathcal{F}(X) \,|\forall g \in G\ \langle f, g\rangle = 0\}$.

Theorem 3 *Let $G \subset \mathcal{F}(X)$ be bounded. Then for every $f \in \mathcal{X} \setminus G^\perp$,*

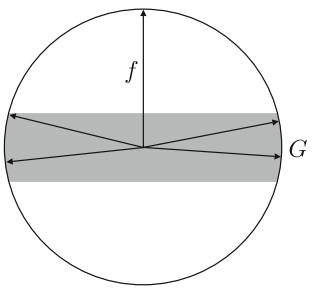

Fig. 3 A function nearly orthogonal to elements of G

$$\|f\|_G \geq \frac{\|f\|^2}{\sup_{g\in G} |g \cdot f|}.$$

Assume that both G and functions to be investigated have Euclidean norms equal to 1 (they can be normalized). Then, Theorem 3 implies that functions not contained in any of the "polar caps" (see Fig. 4) of an angle α with centers in elements of G have G-variation at least $\frac{1}{\arccos \alpha}$.

For large dimensions, such "polar caps" are small, and their relative measures with respect to the whole area of the sphere S^{m-1} decrease exponentially fast with increasing dimension m. Most of areas of high-dimensional spheres lie very close to their "equators". More precisely, let μ be a uniform probabilistic measure on the unit sphere $S^{m-1} := \{h \in \mathbb{R}^d \,|\, \|h\| = 1\}$, and for $g \in S^{m-1}$ and $\eta > 0$ let

$$C(g,\eta) := \left\{h \in S^{m-1} \,|\, |\langle h, g\rangle| \geq \eta\right\}$$

denote the *spherical cap* formed by all vectors within the angular distance $\alpha = \arccos \eta$ from g (see Fig. 4). Then

$$\mu(C(g,\eta)) \leq e^{-\frac{m\eta^2}{2}} = e^{-\frac{m(\cos\alpha)^2}{2}} \tag{7}$$

(see, e.g., [2, p. 11]).

The following theorem estimates uniform probability measures of sets of functions with large variations with respect to finite dictionaries.

Theorem 4 *Let d be a positive integer, $X \subset \mathbb{R}^d$ with* $\operatorname{card} X = m$, *$r > 0$, μ be a uniform probabilistic measure on $S_r(X)$, $b > 0$, and $G(X)$ a finite subset of $\mathcal{F}(X)$ with card $G(X) = k$ such that for all $g \in G(X)$, $\|g\| \geq r$. Then*

$$\mu(\{f \in S_r(X) \,|\, \|f\|_{G(X)} \geq b\}) \geq 1 - 2k\, e^{-\frac{m}{2b^2}}.$$

Proof As $f \in S_r(X)$ and for all $g \in G$, $\|g\| \geq r$, we have by Theorem 3,

$$\|f\|_{G(X)} \geq \frac{\|f\|^2}{\max_{g\in G} |\langle f, g\rangle|} \geq \frac{1}{\max_{g\in G} |\langle f^o, g^o\rangle|}.$$

Hence

$$\{f \in S_r(X) \,|\, \|f\|_{G(X)} \geq b\} \supseteq S_r(X) \setminus \bigcup_{g\in G\cup -G} \bar{C}(g, 1/b),$$

where $\bar{C}(g, 1/b) = \{f \in S_r(X) \,|\, |\langle f^o, g^o\rangle \geq \frac{1}{b}\}$. Setting $X = \{x_1, \ldots, x_m\}$, let $\imath : \mathcal{F}(X) \to \mathbb{R}^m$ be defined as $\imath(f) := (f(x_1), \ldots, f(x_m))$. As $\imath$ is an isometry between $\mathcal{F}(X)$ and $\mathbb{R}^m$, by the inequality (7)

$$\mu^o\left(\left\{h \in S_1(X) \,|\, |\langle h, g^o\rangle| \geq \frac{1}{b}\right\}\right) \leq e^{-\frac{m}{2b^2}},$$

where μ^o is the uniform probability measure on $S_1(X)$ defined as $\mu^o(A^o) = \mu(A)$ for all $A \subset S_r(X)$. So the statement follows. □

The proof of Theorem 4 is based on a comparison of the measure of the whole sphere with its part formed by the union of "polar caps" centered at elements of a dictionary. When the dictionary is not sufficiently large, this union covers only a small fraction of the sphere (see Fig. 5).

With a proper choice of b, for example b such that $b^4 = m = \operatorname{card} X$, we obtain from Theorem 4 large lower bounds on variational norms for functions on large domains. Theorem 4 implies existence of a function with G-variation at least $m^{1/4}$ for a dictionary of size smaller than $\frac{1}{2}e^{\frac{\sqrt{m}}{2}}$. When the size of a dictionary is bounded by $e^{p(\ln m)}$, this lower bound even holds for most functions on X with the fixed norm r.

Corollary 1 *Let d be a positive integer, $X \subset \mathbb{R}^d$ with* $\operatorname{card} X = m$, *$r > 0$, p be a polynomial, $G(X) \subset \mathcal{F}(X)$ with $\|g\| \geq r$ for all $g \in G$ and* $\operatorname{card} G(X) \leq e^{p(\ln m)}$, *$\mu$ be a uniform probabilistic measure on $S_r(X)$, and $b > 0$. Then*

$$\mu(\{f \in S_r(X) \,|\, \|f\|_{G(X)} \geq b) \geq 1 - 2e^{-(\frac{m}{2b^2} - p(\ln m))}.$$

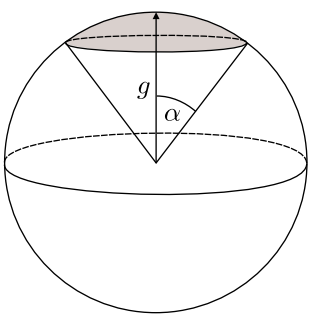

Fig. 4 Spherical cap

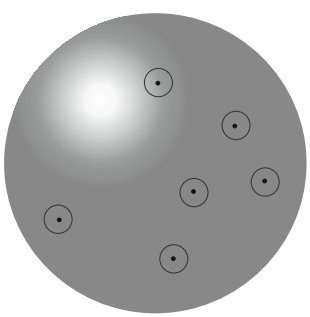

Fig. 5 Complement of polar caps around elements of a dictionary

Combining Corollary 1 with Proposition 1, we obtain a lower bound on l_1-norm of output-weight vectors of shallow networks with units from dictionaries of sizes which do not outweight the factor $e^{-\frac{\operatorname{card} X}{2b^2}}$.

Corollary 2 *Let d be a positive integer, $X \subset \mathbb{R}^d$ with $\operatorname{card} X = m$, $r > 0$, p be a polynomial, $G(X) \subset \mathcal{F}(X)$ with $\|g\| \geq r$ for all $g \in G$ and $\operatorname{card} G(X) \leq e^{p(\ln m)}$, μ be a uniform probabilistic measure on $S_r(X)$, and $b > 0$. Then, the output-weight vector w of any shallow network with units from $G(X)$ computing a uniformly randomly chosen function $f \in S_r(X)$ satisfies $\|w\|_1 \geq b$ with probability at least $1 - 2e^{-(\frac{m}{2b^2} - p(\ln m))}$.*

For example, when the domain X is the d-dimensional cube, then Corollary 2 implies for almost all functions in $S_r(X)$ a lower bound $2^{d/4}$ on l_1-sparsity of all shallow networks with units from dictionaries of sizes bounded by $e^{\frac{1}{2}2^{d/2}}$ computing these functions. Such networks must have either at least $2^{d/4}$ hidden units or absolute values of some output weights have to be greater or equal to $2^{d/4}$, which might lead to non-stability of computation.

Examples of rather small dictionaries are dictionaries of kernel units used in SVM. They contain kernel units parameterized by vectors which belong to the set of data to be classified or to data used for learning with generalization. Thus $\operatorname{card} X = \operatorname{card} G$. SVM algorithm assigns non-zero output weights merely to those units which correspond to support vectors. A solution with a large "l_0-pseudo-norm" corresponds to a large number of support vectors. A solution with a large l_1-norm has a large number of support vectors or it is unstable because some output weights are large.

Theorem 4 implies that for a dictionary G_K of kernel units with centers in the domain X,

$$\mu(\{f \in S_r(X) \mid \|f\|_{G_K(X)} \geq b\}) \geq 1 - 2m\, e^{-\frac{m}{2b^2}}.$$

Also the dictionary of signum perceptrons $G_P(X)$ on a finite domain X in $\mathbb{R}^d$ is small enough to satisfy assumptions of Corollary 2. An upper bound on its size follows from estimates of numbers of linearly separable dichotomies on sets of m points in $\mathbb{R}^d$. Such dichotomies were already studied in the nineteenth century by Schläfli [39]. Their sizes grow only polynomially with m [11]. The degree of the polynomial is equal to the dimension d. The next theorem estimates probability distributions of functions with large variations with respect to signum perceptrons on finite domains.

Theorem 5 *Let d be a positive integer, $X \subset \mathbb{R}^d$ with $\operatorname{card} X = m$, μ a uniform probability measure on $S_{\sqrt{m}}(X)$, and $b > 0$. Then*

$$\mu\left(\left\{f \in S_{\sqrt{m}}(X) \mid \|f\|_{P_d(X)} \geq b\right\}\right) \geq 1 - 4\frac{m^d}{d!}e^{-\frac{m}{2b^2}}.$$

Proof By [11, p. 330], for every d and every $X \subset \mathbb{R}^d$ such that $\operatorname{card} X = m$, $\operatorname{card} G_P(X) \leq 2\sum_{i=0}^{d}\binom{m-1}{i} \leq 2\frac{m^d}{d!}$. Combining this bound with an upper bound on partial sum of binomials, we obtain an upper bound on $\operatorname{card} G_P(X)$. The statement then follows from Theorem 4. □

Applying Theorem 5 to functions on the Boolean cube $\{0,1\}^d$, we obtain a lower bound on measures of sets of functions having variations with respect to signum perceptrons bounded from below by a given bound b. For example, for $b = 2^{d/4}$, we get a lower bound

$$1 - 4\frac{2^{d^2}}{d!}e^{-(2^{d/2}-1)}$$

on the probability that a uniformly randomly chosen function from $\mathcal{F}(\{0,1\}^d)$ with the norm $2^{d/2}$ has variation with respect to signum perceptrons greater or equal to $2^{d/4}$. Thus, a computation of almost any uniformly randomly chosen function from the set of functions with norms equal to $2^{d/2}$ on the d-dimensional Boolean cube $\{0,1\}^d$ by a shallow signum perceptron network would need either $2^{d/4}$ units or computation would be unstable as absolute values of some output weights would be at least $2^{d/4}$.

5 Construction of functions with large variations

In this section, we complement probabilistic estimates of variational norm by concrete examples of functions with large variations built using pseudo-noise sequences, which play an important role in coding [33] and acoustics [40].

It is not difficult to construct an example of a class of functions with large variations with respect to the dictionary of Gaussian kernel units $G_K(x,y) = e^{-a\|x-y\|^2}$ with centers in $\{0,1\}^d$. Let $p_d : \{0,1\}^d \to \{-1,1\}$ denote the *parity function* defined as

$$p_d(v) = -1^{v \cdot u},$$

where $u = (1,\ldots,1)$. In [29], we proved a lower bound $2^{d/2}$ on variation of parities on $\{0,1\}^d$ with respect to the dictionary of Gaussian kernel units of any fixed width $a > 0$.

In [26], we derived a lower bound on variation with respect to the dictionary of signum perceptrons $G_P(X)$ holding for functions on square domains. Any function f on a square domain $X = \{x_1,\ldots,x_n\} \times \{y_1,\ldots,y_n\}$ can be represented by an $n \times n$ matrix $M(f)$ defined as

$$M(f)_{i,j} = f(x_i, y_j).$$

On the other hand, an $n \times n$ matrix M induces a function f_M on X such that

$$f_M(x_i, y_j) = M_{i,j}.$$

Recall that an $n \times n$ matrix is called *Hadamard* when its entries are in $\{-1, 1\}$ and all pairs of its distinct rows (or equivalently columns) are orthogonal. The following theorem from [26] gives a lower bound on variation with respect to signum perceptrons for functions induced by Hadamard matrices.

Theorem 6 *Let M be an $n \times n$ Hadamard, $d = d_1 + d_2$, $\{x_i \,|\, i = 1, \ldots, n\} \subset \mathbb{R}^{d_1}$, $\{y_j \,|\, j = 1, \ldots, m\} \subset \mathbb{R}^{d_2}$, $X = \{x_i \,|\, i = 1, \ldots, n\} \times \{y_j \,|\, j = 1, \ldots, n\} \subset \mathbb{R}^d$, and $f_M : X \to \mathbb{R}$ be defined as $f_M(x_i, y_j) = M_{i,j}$. Then*

$$\|f_M\|_{G^P(X)} \geq \frac{\sqrt{n}}{\lceil \log_2 n \rceil}.$$

Theorem 6 shows that shallow perceptron networks computing functions generated by $n \times n$ Hadamard matrices must have l_1-norms bounded from below by $\frac{\sqrt{n}}{\lceil \log_2 n \rceil}$. In particular, when the domain is the $2d$-dimensional Boolean cube $\{0,1\}^{2d} = \{0,1\}^d \times \{0,1\}^d$, then the lower bound is $\frac{2^{d/2}}{d}$. So the lower bounds grow with d exponentially.

An interesting class of functions with large variations with respect to perceptrons can be obtained by applying Theorem 6 to a class of circulant matrices with rows formed by shifted segments of pseudo-noise sequences. These sequences are deterministic but exhibit some properties of random sequences.

An infinite sequence $a_0, a_1, \ldots, a_i, \ldots$ of elements of $\{0, 1\}$ is called kth-*order linear recurring sequence* if for some $h_0, \ldots, h_k \in \{0, 1\}$

$$a_i = \sum_{j=1}^{k} a_{i-j} h_{k-j} \bmod 2$$

for all $i \geq k$. It is called kth-*order pseudo-noise (PN) sequence* (or *pseudo-random sequence*) if it is kth-order linear recurring sequence with minimal period $2^k - 1$. PN-sequences are generated by *primitive polynomials*. A polynomial

$$h(x) = \sum_{j=0}^{m} h_j x^j$$

is called *primitive polynomial of degree m* when the smallest integer n for which $h(x)$ divides $x^n + 1$ is $n = 2^m - 1$.

PN-sequences have many useful applications because some of their properties mimic those of random sequences. A *run* is a string of consecutive 1's or a string of consecutive 0's. In any segment of length $2^k - 1$ of a kth-order PN-sequence, one-half of the runs have length 1, one quarter have length 2, one-eighth have length 3, and so on. In particular, there is one run of length k of 1's, one run of length $k - 1$ of 0's. Thus every segment of length $2^k - 1$ contains $2^{k/2}$ ones and $2^{k/2} - 1$ zeros [33, p. 410].

An important property of PN-sequences is their low autocorrelation. The *autocorrelation* of a sequence $a_0, a_1, \ldots, a_i, \ldots$ of elements of $\{0, 1\}$ with period $2^k - 1$ is defined as

$$\rho(t) = \frac{1}{2^k - 1} \sum_{j=0}^{2^k - 1} -1^{a_j + a_{j+t}}. \tag{8}$$

For every PN-sequence and for every $t = 1, \ldots, 2^k - 2$,

$$\rho(t) = -\frac{1}{2^k - 1} \tag{9}$$

[33, p. 411].

Let $\tau : \{0, 1\} \to \{-1, 1\}$ be defined as

$$\tau(x) = -1^x$$

(i.e., $\tau(0) = 1$ and $\tau(1) = -1$). We say that a $2^k \times 2^k$ *matrix $L(\alpha)$ is induced by a kth-order PN-sequence* $\alpha = (a_0, a_1, \ldots, a_i, \ldots)$ when for all $i = 1, \ldots, 2^k$, $L_{i,1} = 1$, for all $j = 1, \ldots, 2^k$, $L_{1,j} = 1$, and for all $i = 2, \ldots, 2^k$ and $j = 2, \ldots, 2^k$

$$L(\alpha)_{i,j} = \tau(A_{i-1,j-1})$$

where A is the $(2^k - 1) \times (2^k - 1)$ circulant matrix with rows formed by shifted segments of length $2^k - 1$ of the sequence α. The next proposition following from Eqs. (8) and (9) shows that for any PN-sequence α the matrix $L_k(\alpha)$ has orthogonal rows.

Proposition 2 *Let k be a positive integer, $\alpha = (a_0, a_1, \ldots, a_i, \ldots)$ be a kth-order PN-sequence, and $L_k(\alpha)$ be the $2^k \times 2^k$ matrix induced by α. Then, all pairs of rows of $L_k(\alpha)$ are orthogonal.*

Applying Theorem 6 to the $2^k \times 2^k$ matrices $L_k(\alpha)$ induced by a kth-order PN-sequence α we obtain a lower bound of the form $\frac{2^{k/2}}{k}$ on variation with respect to signum perceptrons of the function induced by the matrix $L_k(\alpha)$. So in any shallow perceptron network computing this function, the number of units or sizes of some output weights depends on k exponentially.

6 Discussion

We investigated limitations of efficiency of shallow networks to represent real-valued functions on finite domains. As minimization of the number of network units computing a given input–output function is a difficult non-convex optimization problem, we focused on approximate measure of sparsity expressed in terms of the l_1-norm of output-weight vectors. The concept of l_1-norm plays an important role in several fields. It has been used as a stabilizer in weight-decay regularization techniques to improve stability of solutions which leads to better generalization. Also, it was used in compressed sensing [9]. Classes of functions defined by bounds on their l_1-norms represent a similar type of a concept as classes of functions defined by bounds on both numbers of gates and sizes of output weights studied in theory of circuit complexity [38].

We derived lower bounds on l_1-norms of output-weight vectors from lower bounds on variational norms tailored to dictionaries of computational units. Using geometric properties of variational norms and of high-dimensional spheres, we proved probabilistic lower bounds on variational and l_1-norms. We showed that almost any uniformly randomly chosen function on a domain of a large size m has variation at least $m^{1/4}$ with respect to any dictionary of size bounded by $e^{p(\ln m)}$, where p is a polynomial.

Our results hold for almost any uniformly randomly chosen function on a large finite domain and can be applied to finite dictionaries (such as signum and Heaviside perceptrons and dictionaries of kernel units used in SVM). Character of our results resembles the No Free Lunch Theorem which also assumes the uniform distribution. However, in real applications, classes of functions of interest are likely non-uniformly distributed. Some of them might belong to those small fractions of sets of all functions on given finite domains which can be computed by reasonably sparse shallow networks. This can explain capabilities of shallow networks to perform efficiently in many practical applications. Investigation of variational norms and l_1-sparsity of functions selected from non-uniform distributions is subject of our future research.

We illustrated our general results by an example of a class of functions generated by matrices constructed using pseudo-noise sequences. These deterministic sequences mimic some properties of random sequences. We showed that shallow perceptron networks, which compute functions constructed using these sequences, must have either large numbers of hidden units or some of their output weights must be large.

There is an interesting analogy with the central paradox of coding theory. This paradox is expressed in the title of the article "Any code of which we cannot think is good" [10]. It was proven there that any code which is truly random (in the sense that there is no concise way to generate the code) is good (it meets the Gilbert–Varshamov bound on distance versus redundancy). However, despite sophisticated constructions for codes derived over the years, no one has succeeded in finding a constructive procedure that yields such good codes. Similarly, computation of "any function of which we cannot think" (truly random) by shallow perceptron networks might be untractable. Our results show that computation of functions exhibiting some randomness properties by shallow perceptron networks is difficult in the sense that it requires networks of large complexities. Such functions can be constructed using deterministic algorithms and have many applications. Properties of pseudo-noise sequences were exploited for constructions of codes, interplanetary satellite picture transmission, precision measurements, acoustics, radar camouflage, and light diffusers. These sequences permit designs of surfaces that scatter incoming signals very broadly making reflected energy "invisible" or "inaudible" [40].

Investigation of sparsity of artificial neural networks has also a biological motivation. Laughlin and Sejnowski [32] concluded from a number of studies that "the brain is organized to reduce wiring costs". They described both sparse activity (only a small fraction of neurons have a high rate of firing at any time) and sparse connectivity (each neuron is connected to only a limited number of other neurons). Our research was focused on investigation of sparse connectivity between hidden units and network outputs.

Acknowledgements This work was partially supported by the Czech Grant Foundation Grant GA15-18108S and institutional support of the Institute of Computer Science RVO 67985807.

Compliance with ethical standards

Conflict of interest The author declares that she has no conflict of interest.

References

1. Ba LJ, Caruana R (2014) Do deep networks really need to be deep? In: Ghahrani Z (ed) Advances in neural information processing systems, vol 27. MIT Press, Cambridge, pp 1–9
2. Ball K (1997) An elementary introduction to modern convex geometry. In: Levy S (ed) Flavors of geometry. Cambridge University Press, Cambridge, pp 1–58
3. Barron AR (1992) Neural net approximation. In: Narendra KS (ed) Proceedings of 7th Yale workshop on adaptive and learning systems. Yale University Press, New Haven, pp 69–72
4. Barron AR (1993) Universal approximation bounds for superpositions of a sigmoidal function. IEEE Trans Inf Theory 39:930–945

5. Bellman R (1957) Dynamic programming. Princeton University Press, Princeton
6. Bengio Y, LeCun Y (2007) Scaling learning algorithms towards AI. In: Bottou L, Chapelle O, DeCoste D, Weston J (eds) Large-scale kernel machines. MIT Press, Cambridge
7. Bengio Y, Delalleau O, Roux NL (2006) The curse of highly variable functions for local kernel machines. In: Weiss Y, Schölkopf B, Platt J (eds) Advances in neural information processing systems, vol 18. MIT Press, Cambridge, pp 107–114
8. Bianchini M, Scarselli F (2014) On the complexity of neural network classifiers: a comparison between shallow and deep architectures. IEEE Trans Neural Netw Learn Syst 25:1553–1565
9. Candès EJ (2008) The restricted isometric property and its implications for compressed sensing. C R Acad Sci Paris I 346:589–592
10. Coffey JT, Goodman RM (1990) Any code of which we cannot think is good. IEEE Trans Inf Theor 36:1453–1461
11. Cover T (1965) Geometrical and statistical properties of systems of linear inequalities with applictions in pattern recognition. IEEE Trans Electron Comput 14:326–334
12. DeVore RA, Howard R, Micchelli C (1989) Optimal nonlinear approximation. Manuscr Math 63:469–478
13. Donoho D (2006) For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution. Commun Pure Appl Math 59:797–829
14. Donoho DL, Tsaig Y (2008) Fast solution of 1-norm minimization problems when the solution may be sparse. IEEE Trans Inf Theory 54:4789–4812
15. Fine TL (1999) Feedforward neural network methodology. Springer, Berlin
16. Gnecco G, Sanguineti M (2009) The weight-decay technique in learning from data: an optimization point of view. Comput Manag Sci 6:53–79
17. Gribonval R, Nielsen M (2003) Sparse representations in unions of bases. IEEE Trans Inf Theory 49:3320–3325
18. Ito Y (1992) Finite mapping by neural networks and truth functions. Math Sci 17:69–77
19. Kainen PC, Kůrková V, Vogt A (1999) Approximation by neural networks is not continuous. Neurocomputing 29:47–56
20. Kainen PC, Kůrková V, Vogt A (2000) Geometry and topology of continuous best and near best approximations. J Approx Theory 105:252–262
21. Kainen PC, Kůrková V, Vogt A (2001) Continuity of approximation by neural networks in L_p-spaces. Ann Oper Res 101:143–147
22. Kainen PC, Kůrková V, Sanguineti M (2012) Dependence of computational models on input dimension: tractability of approximation and optimization tasks. IEEE Trans Inf Theory 58:1203–1214
23. Kůrková V (1997) Dimension-independent rates of approximation by neural networks. In: Warwick K, Kárný M (eds) Computer-intensive methods in control and signal processing. Birkhäuser, Boston, pp 261–270 The Curse of Dimensionality
24. Kůrková V (2012) Complexity estimates based on integral transforms induced by computational units. Neural Netw 33:160–167
25. Kůrková V (2017) Sparsity of shallow networks representing finite mappings. In: Boracchi G (ed) Engineering applications of neural networks, vol CCIS 744. Springer, Berlin, pp 337–348
26. Kůrková V (2018) Constructive lower bounds on model complexity of shallow perceptron networks. Neural Comput Appl 29:305–315
27. Kůrková V, Sanguineti M (2008) Approximate minimization of the regularized expected error over kernel models. Math Oper Res 33:747–756
28. Kůrková V, Kainen PC (2014) Comparing fixed and variable-width Gaussian networks. Neural Netw 57:23–28
29. Kůrková V, Sanguineti M (2016) Model complexities of shallow networks representing highly varying functions. Neurocomputing 171:598–604
30. Kůrková V, Savický P, Hlaváčková K (1998) Representations and rates of approximation of real-valued Boolean functions by neural networks. Neural Networks 11:651–659
31. Kůrková V, Sanguineti M (2017) Probabilistic lower bounds for approximation by shallow perceptron network. Neural Netw 91:34–41
32. Laughlin SB, Sejnowski TJ (2003) Communication in neural networks. Science 301:1870–1874
33. MacWilliams F, Sloane NA (1977) The theory of error-correcting codes. North Holland Publishing Co., New York
34. Mhaskar H, Liao Q, Poggio T (2016) Learning functions: when is deep better than shallow. CBMM Memo No. 045, May 31, 2016. https://arxiv.org/pdf/1603.00988v4.pdf. Accessed 29 May 2016
35. Mhaskar H, Liao Q, Poggio T (2016) Learning real and Boolean functions: when is deep better than shallow. CBMM Memo No. 45, March 4, 2016. https://arxiv.org/pdf/1603.00988v1.pdf. Accessed 3 Mar 2016
36. Pinkus A (1999) Approximation theory of the MLP model in neural networks. Acta Numer 8:143–195
37. Poggio T, Mhaskar H, Rosasco L, Miranda B, Liao Q (2017) Why and when can deep-but not shallow-networks avoid the curse of dimensionality: a review. Int J Autom Comput. https://doi.org/10.1007/s11633-017-1054-2
38. Roychowdhury V, Siu KY, Orlitsky A (1994) Neural models and spectral methods. In: Roychowdhury V, Siu K, Orlitsky A (eds) Theoretical advances in neural computation and learning. Springer, New York, pp 3–36
39. Schläfli L (1901) Theorie der Vielfachen Kontinuität. Zürcher & Furrer, Zürich
40. Schroeder M (2009) Number theory in science and communication. Springer, Berlin
41. Tillmann A (2015) On the computational intractability of exact and approximate dictionary learning. IEEE Signal Process Lett 22:45–49
42. Vaiter S, Peyre G, Dossal C, Fadili J (2013) Robust sparse analysis regularization. IEEE Trans Inf Theory 59:2001–2016