Approximation with Random Bases: Pro et Contra

Alexander N. Gorban[a], Ivan Yu. Tyukin[a,b,1,*], Danil V. Prokhorov[c],
Konstantin I. Sofeikov[a,1]

[a]*University of Leicester, Department of Mathematics, University Road, Leicester, LE1 7RH, UK*
[b]*Department of Automation and Control Processes, St. Petersburg State University of Electrical Engineering, Prof. Popova str. 5, Saint-Petersburg, 197376, Russian Federation*
[c]*Toyota Research Institute NA, Ann Arbor, MI 48105, USA*

Abstract

In this work we discuss the problem of selecting suitable approximators from families of parameterized elementary functions that are known to be dense in a Hilbert space of functions. We consider and analyze published procedures, both randomized and deterministic, for selecting elements from these families that have been shown to ensure the rate of convergence in L_2 norm of order $O(1/N)$, where N is the number of elements. We show that both strategies are successful providing that additional information about the families of functions to be approximated is provided at the stages of learning and practical implementation. In absence of such additional information one may observe exponential growth of the number of terms needed to approximate the function and/or extreme sensitivity of the outcome of the approximation to parameters. Implications of our analysis for applications of neural networks in modeling and control are illustrated with examples.

Keywords: Random bases, measure concentration, neural networks, approximation

1. Introduction

The problem of efficient representation and modeling of data is important in many areas of science and engineering. A typical problem in this area involves constructing quantitative models (maps) of the type

$$x_1, x_2, \ldots, x_d \mapsto f(x_1, x_2, \ldots, x_d),$$

where $x_1, x_2, \ldots, x_d$, $x_i \in \mathbb{R}$, $i = 1, \ldots, d$ are variables and $f : \mathbb{R}^d \to \mathbb{R}$ is an unknown functional relation between the variables. The total number, d, of variables, determining input data may be large, and physical models of such relations $f(\cdot)$ are not always available.

In absence of acceptably detailed prior knowledge of "true" models, $f(\cdot)$, a commonly used alternative is to express the function $f(\cdot)$ as a linear combination of known functions, $\varphi_i(\cdot)$, $\varphi : \mathbb{R}^d \to \mathbb{R}$:

$$f(x) \simeq f_N(x) = \sum_{i=1}^{N} c_i \varphi_i(x), \ c_i \in \mathbb{R}. \tag{1}$$

Numerous classes of functions $\varphi_i(\cdot)$ in (1) that can be used for this purposes have been proposed and analysed to date, starting from $\sin(\cdot)$, $\cos(\cdot)$ and polynomial functions featured in classical Fourier, Fejer, and Weierstrass results, wavelets [32], [28], and reaching out to linear combinations of sigmoids [5]

$$f_N(x) = \sum_{i=1}^{N} c_i \frac{1}{1 + e^{(w_i^T x + b_i)}}, \ w_i \in \mathbb{R}^d, \ b_i \in \mathbb{R}, \tag{2}$$

radial basis functions [23]

$$f_N(x) = \sum_{i=1}^{N} c_i e^{(-\|w_i^T x + b_i\|^2)} \ w_i \in \mathbb{R}^d, \ b_i \in \mathbb{R}, \tag{3}$$

and other general functions [8] that are often used in neural networks literature [12]. Some times the values of parameters w_i, b_i may be pre-selected on the basis of additional prior knowledge, leaving only linear weights c_i for training. If no prior information is available then both nonlinear (w_i and b_i) and linear (c_i) parameters, or weights, are typically subject to training on data specific to the problem at hand (full network training).

One special feature that makes full training of approximators (2), (3) particularly attractive is that in addition to their universal approximation capabilities [5], [14], [23] and their homogenous structure, they are reported to be efficient when the dimension d of input data is relatively high. In particular, if *all parameters* w_i, c_i, b_i are allowed to vary, the order of convergence rate of the approximation error of a sufficiently smooth function $f(\cdot) \in \mathcal{C}^0[0,1]^d$ as a function of N (the number of elements in the network) is shown to be independent of the input dimension d [16], [4]. Furthermore, the achievable rate of convergence of the L_2-norm of $f(\cdot) - f_N(\cdot)$ is shown to be of order $O(1/N^{1/2})$. This contrasts sharply with the rate of order $O(d^{-1}/N^{1/d})$ corresponding to the worst-case estimate inherent to linear combinations (1) with $\varphi_i(\cdot)$ given or (2), (3) with w_i, b_i fixed. In particular, it is shown in [4] that if only linear parameters of (2), (3) are adjusted the approximation error *cannot be made* smaller than $Cd^{-1}/N^{1/d}$, where C is independent on N, uniformly for functions satisfying the same smoothness constraints.

2

Favorable independence of the order of convergence rates on the input dimension of the function to be approximated, however, comes at a price. Construction of such efficient models of data involves a nonlinear optimization routine searching for the best possible values of w_i, b_i. The necessity to adjust nonlinear parameters of (2), (3) restricts practical application of these models in relevant problems in which implementation of such optimization is practically challenging (see e.g., [25], [35], [33], [21] for an overview of potential issues).

An alternative to adjustment of nonlinear parameters of (2), (3) has been proposed in [31], [22], [15] and further developed in [26], [27] (see also earlier work by F. Rosenblatt [29] in which he discussed perceptrons with random weights). In these works the nonlinear parameters w_i and b_i are proposed to be set randomly at the initialization, rather than through training. The only trainable parameters are those which enter the network equation linearly, c_i. Random selection of weights in the hidden layers is supposed to generate a set of (basis) functions that is sufficiently rich to approximate a given function by mere linear combinations from this set. This crucially simplifies training in systems featuring such approximators, and renders an otherwise computationally complex nonlinear optimization problem into a much simpler linear one. Moreover, the rate of error convergence is argued to be of the order $O(1/N^{1/2})$ [15], [26], [27].

This brings us to a strange and a rather paradoxical contradiction: on the one hand the reported convergence rate of order $O(1/N^{1/2})$ which a randomized approximator is supposed to achieve contradicts to the earlier worst-case estimate $O(d^{-1}/N^{1/d})$ obtained in [4]. On the other hand numerous studies report successful application of such randomization ideas (see also comments [19]) in a variety of intelligent control applications [13], [18], [30], [17], [20] as well as in machine learning (see e.g. [1], [36] and references therein). The question, therefore, is if it is possible to resolve such an apparent controversy? Answering to this question is the main purpose of this contribution.

The paper is organized as follows. We begin the analysis with Section 3 in which review basic reasoning in [15] and compare these results with [16], [4]. We show that, although these results may seem inconsistent, they have been derived for different performance criteria. The worst-case estimate in [4] is "crisp", whereas the convergence rate in [15] is probabilistic. That is it involves a measure function with respect to which the rate of $O(1/N^{1/2})$ is assured. Introduction of measure into the problem brings out a range of interesting consequences that is discussed and analyzed in Section 4. There we approach data approximation problem as that of representation of a given vector by linear combinations of randomly chosen vector in high dimensions. A simple logic suggests that if one takes $m \leq n$ random vectors in $\mathbb{R}^n$ then he can expect that with probability one these vectors will be linearly independent because the set of linearly dependent corteges $\{x_1, \ldots, x_m\}$ ($m \leq n$) is a proper algebraic subset in the space of corteges $(\mathbb{R}^n)^m$. We can select n random vectors, with probability one it will be a basis. Every given data vector y can be represented by coordinates in this basis. If these n vectors are (accidentally) too close to dependence we can generate few more vectors that will enable us to represent the data vector

y. We will show, however, that this simple (and correct!) reasoning looses its credibility in high dimensions. We show that in an n-dimensional unite cube $[-1,1]^n$ a randomly generated vector x will be almost orthogonal to a given data vector y (the angle between x and y will be close to $\pi/2$ with probability close to one). To compensate for this *waist concentration* effect [11], [10], [9] one needs to generate exponentially many random vectors. Typicality of such exponential growth is an inherent feature of high-dimensional data representation, including problems of data approximation and modelling. Moreover, for high-dimensional data representation the following two seemingly contradictory situations are typical in some sense:

- *with probability one linear combinations of $n-k$ random vectors approximate any normalized vector with accuracy ε if $k \ll n$ and no constraints on the values of coefficients in linear combinations are imposed;*

- *with probability close to one an exponentially large number N of random vectors are pairwise almost orthogonal and do not span an arbitrarily selected normalized vector if coefficients in linear combinations are not allowed to be arbitrarily large.*

Implications of our analysis are illustrated in Section 5 followed by few examples presented in Section 6. Section 7 concludes the paper.

2. Notation

Throughout the paper the following notational agreements are used

- $\mathbb{R}$ denotes the field of real number;

- $\mathbb{R}^n$ stands for the n-dimensional linear space over the field of reals;

- let $x \in \mathbb{R}^n$, then $\|x\|$ is the Euclidean norm of x: $\|x\| = \sqrt{x_1^2 + \cdots + x_n^2}$;

- if x, y are two non-zero vectors from $\mathbb{R}^n$ then $\angle(x,y)$ denotes an angle between these vectors;

- symbol $|\cdot|_{\mathbb{R}^n}$ is reserved to denote an arbitrary norm in $\mathbb{R}^n$;

- $\mathcal{S}^{n-1}(R)$ denotes an $n-1$-sphere of radius R centred at 0: $\mathcal{S}^{n-1}(R) = \{x \in \mathbb{R}^n |\ \|x\| = R\ \}$

- μ is the normalized Lebesgue measure on $\mathcal{S}^{m-1}(1)$: $\mu(\mathcal{S}^{m-1}(1)) = 1$.

- $\mathcal{B}^n(R)$ denotes a n-ball of radius R centered at 0: $\mathcal{B}^n(R) = \{x \in \mathbb{R}^n |\ \|x\| \leq R\}$

- $\mathcal{V}(\Xi)$ is the Lebesgue volume of $\Xi \subset \mathbb{R}^n$.

- $\mathcal{D}^{n-1}(R)$ stands for a $n-1$-disc in the n-ball $\mathcal{B}^n(R)$ corresponding to its largest equator, and $\mathcal{D}_\delta^{n-1}(R)$ is its δ-thickening.

- Let $f : [0, 1]^d \to \mathbb{R}$ be a continuous function, then

$$\|f\|^2 = \langle f, f \rangle = \int_{[0,1]^d} f(x)f(x)dx,$$

denotes the L_2-norm of f.

3. Preliminary analysis and motivation

In this section we review and compare the so-called *greedy approximation* upon which the famous Barron's construction is based [4] and the Random Vector Functional-Link (RVFL) network [15] in which the basis functions are randomly chosen, and only their linear parameters are optimized. As we show below, despite the error convergence rates appear to be of the same order, they are in essence dramatically different. One rate is crisp in the sense that it is a worst-case estimate for all functions from a given class. The other one is probabilistic in nature, and hence holds in probability. The consequences of the latter are further investigated in Section 4.

Consider the following class of problems. Suppose that $g : \mathbb{R} \to \mathbb{R}$ be a function such that

$$\|g\| \leq M, \ M \in \mathbb{R}_{>0},$$

and

$$\mathcal{G} = \{g(w^T x + b)\}, \ w \in \mathbb{R}^d, \ b \in \mathbb{R},$$

be a family of parameterized functions $g(\cdot)$. Let $f \in \mathcal{C}^0([0, 1]^n)$, and let f belong to a convex hull of $\mathcal{G}$. In other words there is a sequence of w_i, b_i, and c_i such that

$$f(x) = \sum_{i=1}^{\infty} c_i g(w_i^T x + b_i), \ \sum_{i=1}^{\infty} c_i = 1, \ c_i \geq 0.$$

Let

$$f_N(x) = \sum_{i=1}^{N} c_i g(w_i^T x + b_i) \tag{4}$$

be a superposition of functions $g(w_i^T x + b_i)$. In practice the values of c_i, w_i, b_i are not always unknown for a given f. Furthermore, retaining infinitely many elements the sums is not feasible either. We are thus interested in finding an acceptably possible approximation of f by sums of mere N elements from $\mathcal{G}$. It is therefore important to know how approximation errors, defined in some meaningful sense, may decay with N, and what are the computational costs for achieving this rate of decay?

3.1. Greedy approximation and Jones Lemma

In order to answer the question above one needs first to determine the error of approximation. It is natural for functions from L_2 to define the approximation error as follows:

$$e_N = \|f_N - f\| \tag{5}$$

5

The classical Jones iteration [16] (refined later by Barron [4]) provides us with the following estimate of achievable convergence rate:

$$e_N^2 \le \frac{M'^2 e_0^2}{Ne_0^2 + M'^2}, \ M' > \sup_{g \in \mathcal{G}} \|g\| + \|f\|. \tag{6}$$

The rate of convergence depends on d only through the L_2-norms of f_0, g, and f. The iteration itself is deterministic and can be described as follows:

$$f_{N+1}(x) = (1 - \alpha_N)f_N(x) + \alpha_N g(w_N^T x + b_N)$$
$$\alpha_N = \frac{e_n^2}{M''^2 + e_n^2}, \ M'' > M' \tag{7}$$

where parameters W_N, b_N of g is chosen such that the following condition holds

$$\langle f_N - f, g(w_N^T \cdot + b_N) - f \rangle < \frac{((M'')^2 - (M')^2)e_N^2}{2(M'')^2}, \tag{8}$$

This choice is always possible (see [16] for details) as long as the function f is in the convex hull of $\mathcal{G}$.

According to (6) the rate of convergence of such approximators is estimated as

$$\|e_N\|^2 = O(1/N).$$

This convergence estimate is *guaranteed* because it is the upper bound for the approximation error at the nth step of iteration (7).

3.2. Approximation by linear combinations of functions with randomly chosen parameters

We now turn our attention to the result in [15]. In this approximator the original function $f(\cdot)$ is assumed to have the following integral representation[2]

$$f(x) = \lim_{\alpha \to \infty} \lim_{\Omega \to \infty} \int_{W^d} F_{\alpha,\Omega}(\omega)g(\alpha w^T x + b)d\omega, \tag{9}$$

where $g : \mathbb{R} \to \mathbb{R}$ is a non-trivial function from L_2:

$$0 < \int_{\mathbb{R}} g^2(s)ds < \infty, \tag{10}$$

where $\omega = (y, w, u) \in \mathbb{R}^d \times \mathbb{R}^d \times [-\Omega, \Omega]$, $\Omega \in \mathbb{R}_{>0}$, $W^d = [-2\Omega; 2\Omega] \times I^d \times V^d$, $V^d = [0; \Omega] \times [-\Omega; \Omega]^{d-1}$, $b = -(\alpha w^T y + u)$ and

$$F_{\alpha,\Omega}(\omega) \sim \frac{\alpha \prod_{i=1}^{d} w_i}{\Omega^d 2^{d-1}} f(y) \cos_\Omega(u), \ \cos_\Omega(u) = \begin{cases} \cos(u), \ u \in [-\Omega, \Omega] \\ 0, \ u \notin [-\Omega, \Omega] \end{cases}$$

---

[2]We keep the original notation of [15] which uses both ω and w for the sake of consistency.

(see [15], [19] for more detailed description). Function $g(\cdot)$ induces a parameter-ized basis. Indeed if we were to take integral (9) in quadratures for sufficiently large values of α and Ω, we would then express $f(x)$ by the following sums of parameterized $g(\alpha w^T x + b)$ [15]:

$$f_N(x) \simeq \sum c_i g(\alpha w_i^T x + b_i), \ b_i = -\alpha(w_i^T y_i + u_i) \tag{11}$$

The summation in (11) is taken over points ω_i in W^d, and c_i are weighting coefficients. Variables α in (9) and α_N in (7) play different roles in each approx-imations schemes. In (7) the value of α_n is set to ensure that the approximation error is decreasing with every iteration, and in (9) it stands for a scaling factor of random sampling.

The main idea of [15] is to approximate integral representation (9) of $f(x)$ using the Monte-Carlo integration method as

$$
\begin{aligned}
f(x) &\simeq \frac{4\Omega^d}{N} \lim_{\alpha \to \infty} \lim_{\Omega \to \infty} \sum_{k=1}^{n} F_{\alpha,\Omega}(\omega_k) g(\alpha w_k^T x + b_k) \\
&= \lim_{\alpha \to \infty} \lim_{\Omega \to \infty} \sum_{k=1}^{N} c_{k,\Omega}(\alpha, \omega_k) g(\alpha w_k^T x + b_k) \\
&= f_{N,\omega,\Omega}(x),
\end{aligned}
\tag{12}
$$

where the coefficients $c_k(\alpha, \omega_k)$ are defined as

$$c_{k,\Omega}(\alpha, \omega_k) = \frac{1}{N} \frac{4\alpha \prod_{i=1}^{d} w_{k,i}}{2^{d-1}} \cos_\Omega(u_k) f(y_k) \tag{13}$$

and $\omega_k = (y_k, w_k, u_k)$ are randomly sampled in W^d (domain of parameters, i.e., weights and biases of the network).

When the number of samples, N, i.e., *the network size*, is large, then

$$E_\omega(N) = \sqrt{E_\omega \int_K |f(x) - f_{N,\omega,\Omega}(x)|^2 dx}, \ K \subset [0,1]^d, \tag{14}$$

converges to zero asymptotically for large N. In particular:

Theorem 1 (Igel'nik and Pao [15]). *For any compact K, $K \subset [0,1]^d$, $K \neq [0,1]^d$ and any absolutely integrable function g satisfying (10) there exists a sequence of $f_{N,\omega,\Omega}$ and a sequence probability measures $\{\mu_{N,\Omega,\alpha}\}$ such that*

$$\lim_{N \to 0} E_\omega(N) = 0.$$

Furthermore, under some additional restrictions on the functions g, f the ex-pectation $E_\omega(N)$ is shown to obey (Theorem 3 in [15]):

$$E_\omega(N) \leq \frac{C}{N^{1/2}}. \tag{15}$$

7

Similar results have been reported in [27]. There the authors look at the following classes of functions:

$$\mathcal{F}_p = \{f(x) = \int_\Omega \alpha(\omega)\phi(x,\omega)d\omega \mid |\alpha(\omega)| \leq Cp(\omega)\}, \ \sup_{x,w}|\phi(x,w)| \leq 1,$$

where p is a distribution on Ω, and

$$\mathcal{F}_\omega = \{f(x) = \sum_{k=1}^N \alpha_k\phi(x,\omega_k) \ ||\alpha_k| \leq C/N\}.$$

For the chosen classes of functions they establish the following

Theorem 2 (cf. Lemma 1 in [27]). *Let f be a function from $\mathcal{F}_p$ and $w_1,\ldots,w_N$ be drawn iid from p. Then for any $1 > \delta > 0$ with probability at least $1 - \delta$ over $w_1,\ldots,w_N$ there exists a function f_N from $\mathcal{F}_\omega$ so that*

$$\|f - f_N\| \leq \frac{C}{\sqrt{N}}\left(1 + 2\sqrt{\log\frac{1}{\delta}}\right). \tag{16}$$

At the first glance errors (5), (14), (16) and their respective convergence rates look very similar. Yet, they are fundamentally different in that (5) is deterministic and its convergence rate is crisp, whereas the other two estimates have an additional element - a probability measure - characterizing their convergence rate. Introduction of this measure enables to "ignore" worst-case elements corresponding to the rate of convergence $O(d^{-1}/N^{1/d})$. At the same time, it imposes some limitations too since there always is a non-zero probability of an unlucky draw from the probability distribution which will require re-initialization at some stage. Furthermore, explicit presence of $1/\delta$ in (16)[3] suggest that this rapid convergence of order $1/N^{1/2}$ is assured only up to a given and fixed tolerance. These features of randomized approximators should thus be considered with special care in applications.

There is one additional point that is inherent to all randomized approximators and yet it is rarely addressed in practice. This is a measure concentration effect that we will present and consider in detail in the next section.

4. Randomized Data Approximation and Measure Concentration

So far we considered the problem of data approximation and modelling from the function approximation point of view. Let us, however, look at this problem from a slightly different perspective (cf. [36]). Suppose that the data points, x,

---

[3]The very same term is implicit in the error rate derived for Monte-Carlo inspired approach in [15]. This follows immediately from the Chebyshev inequality. Indeed, if $X_1,\ldots,X_N$ are iid random variables with the same mean, μ and variance σ^2 then the probability that $|(X_1 + \cdots + X_N)/N - \mu| \geq \delta$ is assured to be at most $\sigma^2/(\delta^2 N)$.

are labeled by real numbers t_i so that each i-th data point is uniquely determined by its label t_i. This means that the problem can be rephrased as that of representing a combined vector $y = (f(x(t_1)), f(x(t_2)), \ldots, f(x(t_n)))$ by linear combinations of $h_i = (g(w_i^T x(t_1) + b_i), g(w_i^T x(t_2) + b_i), \ldots, g(w_i^T x(t_n) + b_i))$ so that the error

$$y - \sum_{i=1}^{m} c_i h_i$$

is minimized. If our choice of vectors $h_1, \ldots, h_m$ is good enough to represent an arbitrary vector y with acceptable precision then we can be assured that the above approximation model is viable. The questions, however, are

1) if the choice of h_i is randomized then how many vectors one must choose to ensure desired universality and accuracy?

2) how robust such representations are with respect to small perturbation of data?

These questions are addressed in the next sections. For simplicity we ignore that the vectors h_i are in fact generated by functions $g(\cdot)$ and proceed assuming that they are simply sampled randomly in a hypercube.

4.1. Linear dependence and independence and related elementary properties
    Let us first recall standard notions of linear dependence and independence.

Definition 1 (Linear Dependence). *A system of vectors $h_1, h_2, \ldots, h_m$, $h_i \in \mathbb{R}^n$, $i = 1, \ldots, m$ is said to be linearly dependent iff there exist $c_1, c_2, \ldots, c_m$, $c_i \in \mathbb{R}$, $i = 1, \ldots, m$ such that*

$$h_1 c_1 + h_2 c_2 + \cdots + c_m h_m = 0$$

and at least one of $c_1, c_2, \ldots, c_m$ is not equal to zero.

The same definition can be rewritten in the vector-matrix notation as

$$\exists\, c \in \mathbb{R}^m, \ c \neq 0 : Hc = 0, \tag{17}$$

where

$$H = (h_1 \ h_2 \cdots \ h_m)$$

is an $n \times m$ matrix formed by $h_1, \ldots, h_m$.

Definition 2 (Linear Independence). *A system of vectors $h_1, h_2, \ldots, h_m$, $h_i \in \mathbb{R}^n$, $i = 1, \ldots, m$ is said to be linearly dependent if it is not linearly independent. In other words,*

$$Hc \neq 0 \ \forall\, c \in \mathbb{R}^m : \ c \neq 0. \tag{18}$$

    A very simple fact follows immediately from Definition 2

Proposition 1. *Consider a system of vectors $h_1, \ldots, h_m$, and let $H = (h_1 \ h_2 \ \cdots \ h_m)$. Let $|\cdot|_{\mathbb{R}^n}$ be a norm on $\mathbb{R}^n$, then the system $h_1, \ldots, h_m$ is linearly independent iff there exists an $\varepsilon > 0$ such that*

$$|Hx|_{\mathbb{R}^n} > \varepsilon \ \forall\, x \in \mathcal{S}^{m-1}(1). \tag{19}$$

Proof. Suppose first that the system $h_1, \ldots, h_m$ is linearly independent. Hence (18) holds, and given that $x \neq 0$ for all $x \in \mathcal{S}^{m-1}(1)$ we thus obtain that $|Hx|_{\mathbb{R}^n} > 0$ for all $x \in \mathcal{S}^{m-1}(1)$. Noticing that $|\cdot|_{\mathbb{R}^n}$ is continuous and $\mathcal{S}^{m-1}(1)$ is compact we conclude that $|\cdot|_{\mathbb{R}^n}$ takes minimal and maximal values on $\mathcal{S}^{m-1}(1)$. Let

$$\varepsilon = \min_{x \in \mathcal{S}^{m-1}(1)} |Hx|_{\mathbb{R}^n} .$$

The minimum of $|\cdot|_{\mathbb{R}^n}$ on $\mathcal{S}^{m-1}(1)$ is separated away from 0 since otherwise there will exist a $x \in \mathcal{S}^{m-1}(1)$ such that $Hx = 0$. Thus (19) holds.

Let us now show that (19) implies (18). Indeed, for any $c \in \mathbb{R}^m$, $c \neq 0$ there is an $x \in \mathcal{S}^{m-1}(1)$ and an $\alpha \in \mathbb{R}$, $\alpha \neq 0$ such that $c = \alpha x$. Hence $|Hc|_{\mathbb{R}^n} = |\alpha| \, |Hx|_{\mathbb{R}^n} > |\alpha|\varepsilon > 0$, which automatically assures that (18) holds. $\square$

4.2. Quantification of linear independence

Standard notions of linear dependence and independence are, unfortunately, not always easy to assess numerically when the values of ε in (19) are small. Further to this, checking that for a given system of vectors $h_1, \ldots, h_m$, some ε, and all $x \in \mathcal{S}^{m-1}(1)$ the following holds

$$|Hx|_{\mathbb{R}^n} > \varepsilon \tag{20}$$

may not always be practically feasible or desirable.

Two ways to relax and quantify the conventional notion of linear independence are obviated in Proposition 1. These are 1) the value of ε in (19), and 2) a possibility of introducing a finite measure on $\mathcal{S}^{m-1}(1)$ that determines a proportion of x from $\mathcal{S}^{m-1}(1)$ which satisfy (20).

Definition 3 (Almost Linear Independence). *Let $h_1, \ldots, h_m$ be a system of m normalized vectors from $\mathbb{R}^n$: $|h_i|_{\mathbb{R}^n} = 1$, $i = 1, \ldots, m$. We will say that the system is (ε, θ)-linearly independent (almost linearly independent) with respect to μ if*

$$\mu(\{x \in \mathcal{S}^{m-1}(1) \mid |Hx|_{\mathbb{R}^n} \geq \varepsilon\}) \geq 1 - \theta. \tag{21}$$

Similarly, one can define an alternative quantification of linear independence:

Definition 4 (Almost Linear Dependence). *Let $h_1, \ldots, h_m$ be a system of m normalized vectors from $\mathbb{R}^n$: $|h_i|_{\mathbb{R}^n} = 1$, $i = 1, \ldots, m$. We will say that the system is (ε, θ)-linearly dependent (almost linearly independent) with respect to μ if*

$$\mu(\{x \in \mathcal{S}^{m-1}(1) \mid |Hx|_{\mathbb{R}^n} \leq \varepsilon\}) \geq 1 - \theta.$$

Notice that the notions of almost linear independence and almost linear dependence introduced in Definitions 3, 4 are consistent with conventional notions in the sense that the latter can be viewed as limiting cases of the former. Indeed, if μ is a surface area then setting $\theta = 0$ in Definition 3 and picking ε small enough one obtains an equivalent of Definition 2.

The above measure, or "probabilistic", quantification of linear independence has significant implications for data representation in applications. As we shall see in the next sections two seemingly exclusive extremes are likely to hold in higher dimensions. First, almost all points of an n-ball concentrate in an ϵ-thickening of an $n-1$-disc. This means that for m sufficiently large a family of randomly and independently chosen vectors $h_1, h_2, \ldots, h_m$ becomes almost linearly dependent. This phenomenon is well-known in the literature (see e.g. [10], [11], [9] as well as classical works of Maxwell, Levy, and Gibbs; in Data Mining applications measure concentration effects have been discussed in [24]). Yet, the values of m for which almost linear independence persists may be exponentially large. Furthermore, the latter situation holds with a probability that is close to one. This second extreme can be rephrased as that the number of almost orthogonal vectors grows exponentially with dimension. More formal statements and justifications are provided in Propositions 2, 3 below.

4.3. Measure concentration

We begin with the following statement

Proposition 2. *Let $\mathcal{B}^n(R)$ be an n-ball of radius R in $\mathbb{R}^n$, and $0 < \delta < R$ be a non-negative number. Then, for $n \gg 1$ the following estimate holds:*

$$\frac{\mathcal{V}(\mathcal{B}^n(R) - \mathcal{B}^n(R - \delta))}{\mathcal{B}^n(R)} \sim 1 - e^{-\frac{n\delta}{R}} \tag{22}$$

Proof. Noticing that $\mathcal{V}(\mathcal{B}^n(R)) = C_n R^n$, where C_n is a constant independent on R we conclude that

$$\frac{\mathcal{V}(\mathcal{B}^n(R)) - \mathcal{V}(\mathcal{B}^n(R - \delta))}{\mathcal{V}(\mathcal{B}^n(R))} =$$

$$1 - \frac{(R - \delta)^n}{R^n} = 1 - \left(1 - \frac{\delta}{R}\right)^n.$$

Given that

$$\left(1 - \frac{\delta}{R}\right)^n \sim e^{-\frac{n\delta}{R}}$$

for $n \gg 1$, we conclude that the statement holds. $\square$

In accordance with Proposition 2 the volume of an n-ball of radius R, for n sufficiently large, is concentrated in a thin layer around its surface. Furthermore, in this thin layer the volume of an n-ball is concentrated around the largest equator of the corresponding $n-1$ sphere, $\mathcal{S}^{n-1}(R)$.

Proposition 3. *Let $\mathcal{B}^n(R)$ be an n-ball of radius R in $\mathbb{R}^n$, and $0 < \delta < R$ be a non-negative number. Let $\mathcal{D}_\delta^{n-1}(R)$ be a δ-thickening of an $n-1$-disc $\mathcal{D}^{n-1}(R)$. Then for $n \gg 1$*

$$\frac{\mathcal{V}(\mathcal{B}^n(R)) - \mathcal{V}(\mathcal{D}_\delta^{n-1}(R))}{\mathcal{V}(\mathcal{B}^n(R))} \lesssim 1 - e^{-\frac{n\delta^2}{2R^2}}.$$
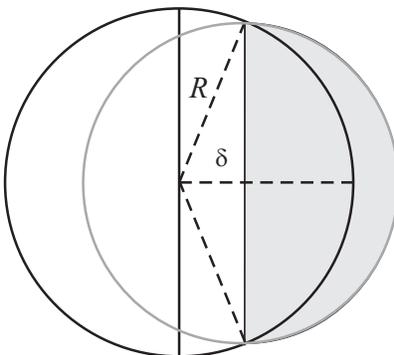
Figure 1: Illustration to the estimate of the 2δ width of the waist of the sphere

Proof. Consider first

$$\frac{\mathcal{V}(\mathcal{B}^n(R)) - \mathcal{V}(\mathcal{B}^n(\sqrt{R^2 - \delta^2}))}{\mathcal{V}(\mathcal{B}^n(R))} =$$

$$1 - \left(1 - \frac{\delta^2}{R^2}\right)^{\frac{n}{2}}$$

Let $n \gg 1$, then

$$1 - \left(1 - \frac{\delta^2}{R^2}\right)^{\frac{n}{2}} \sim 1 - e^{-\frac{n\delta^2}{2R^2}}.$$

Noticing that

$$\mathcal{V}(\mathcal{B}^n(R)) - \mathcal{V}(\mathcal{D}_\delta^{n-1}(R)) \le \mathcal{V}(\mathcal{B}^n(\sqrt{R^2 - \delta^2}))$$

the result follows. $\qquad\square$

In high dimension the volume of the ball is concentrated in a thin layer near the sphere. Therefore, the estimate of the volume of the disk automatically provides an estimate of the surface of the corresponding waist of the sphere. Let us produce this estimate explicitly. The proportion of $\mathcal{S}^{n-1}(1)$ belonging to the cap (shaded part of the sphere in Fig. 1) equals the proportion of the solid ball that lies in the corresponding spherical cone (cf. [3], Fig. 11). The latter consist of two parts: one is the cone of height δ and radius of the base $\sqrt{1 - \delta^2}$. The volume of the second part can be bounded from above by the half of the volume of the ball with radius $\sqrt{1 - \delta^2}$. If we use the Stirling formula for the volume of high-dimensional ball $V_n(R) \sim \frac{1}{\sqrt{n\pi}} \left(\frac{2\pi e}{n}\right)^{n/2} R^n$ then we obtain that the fraction of the waist of the width 2δ is $1 - (1 + O(\delta/\sqrt{n}))e^{-\frac{n\delta^2}{2}}$. This estimate improves the textbook estimate $1 - 2e^{-\frac{n\delta^2}{2}}$ for large n (see [3] for further details).

The concentration effect described in Proposition 3 implies that in high dimensions nearly all independently and randomly drawn vectors are almost

linearly dependent. On the other hand a set of $n-1$ vectors with probability close to one spans almost all vectors. We note that the latter property holds for systems of $n-k$, $k > 1$ vectors too which follows immediately from [2]:

Theorem 3 (Theorem 3.1 in [2]). *Let E_k be a k-dimensional subspace of $\mathbb{R}^n$, and denote by $\mu((E_k)_\varepsilon)$ the Haar measure on the sphere $\mathcal{S}(1)$ of the set of points within a geodesic distance smaller than ε of E_k. We write $k = \lambda n$. Fix $0 < \varepsilon < \pi/2$ and $0 < \lambda < 1$. The following estimates hold as $n \to \infty$*

(i) If $\sin^2 \varepsilon > 1 - \lambda$, then

$$\mu((E_k)_\varepsilon) \simeq 1 - \frac{1}{\sqrt{n\pi}} \frac{\sqrt{\lambda(1-\lambda)}}{\sin^2 \varepsilon - (1-\lambda)} e^{\frac{n}{2} u(\lambda, \varepsilon)}.$$

(ii) If $\sin^2 \varepsilon < 1 - \lambda$, then

$$\mu((E_k)_\varepsilon) \simeq \frac{1}{\sqrt{n\pi}} \frac{\sqrt{\lambda(1-\lambda)}}{(1-\lambda) - \sin^2 \varepsilon}$$

where $u(\lambda, \varepsilon) = (1 - \lambda) \log \frac{1-\lambda}{\sin^2 \varepsilon} + \lambda \log \frac{\lambda}{\cos^2 \varepsilon}$.

Since nearly all vectors in $\mathcal{B}^n(1)$ are concentrated in an ε-disc it is interesting to know how many pairwise almost orthogonal vectors can be found in this set. It turns out that this number is exponentially large. Detailed analysis and derivations are provided in the next section.

4.4. Almost orthogonality in high dimensions

Select two small positive numbers, ϵ and ϑ. Let us generate randomly and independently N vectors $x_1, \ldots, x_N$ on $\mathcal{S}^{n-1}(1)$. We are interested in the probability P that all N random vectors are pairwise ϵ-orthogonal, i.e. $|(x_i, x_j)| < \epsilon$ for $i, j = 1, \ldots, N$, $i \neq j$. For which N this $P > 1 - \vartheta$?

Propositions 2, 3 suggest that for $n \gg 1$ almost all volume of an n-ball $\mathcal{B}^n(1)$ is concentrated in an ϵ-thickening of its largest equator. This implies that for an arbitrarily chosen point p on the surface of $\mathcal{B}^n(1)$ almost all points of the ball will belong to a $\mathcal{D}_\epsilon^{n-1}(1)$. On the other hand, for the given choice of p and $\mathcal{D}_\epsilon^{n-1}(1)$ the following estimates holds

$$|p^T x| \leq \epsilon \; \forall \; x \in \mathcal{D}_\epsilon^{n-1}(1).$$

The latter property is equivalent to that the vector p is almost orthogonal to the rest of points in $\mathcal{B}^n(1)$. Indeed, in accordance with Proposition 2, almost all points of the ball $\mathcal{B}^n(1)$ are concentrated around a ε-thickening of surface. Lengths of such points, x, satisfy $1 - \varepsilon \leq \|x\| \leq 1$, and hence

$$\cos(\angle(p, x)) \leq \frac{|p^T x|}{1 - \varepsilon} \leq \frac{\epsilon}{1 - \varepsilon}.$$

Let us determine the number of independent random vectors which are pairwise ϵ-orthogonal with probability $1 - \vartheta$. The volume taken by all vectors that are almost orthogonal to a given vector on a unit sphere can be estimated from Proposition 3 (up to a small correction term $O(\delta e^{-\frac{n\delta^2}{2}}/\sqrt{n})$). Consider the following products

$$P(\epsilon, N) = \prod_{k=1}^{N} \left(1 - ke^{-\frac{n\epsilon^2}{2}}\right). \tag{23}$$

The value of $P(\epsilon, N)$ is an estimate from below of the probability of a set of $N+1$ independent random vectors to be pairwise ϵ-orthogonal. Indeed, for one vector h_1 the fraction of the vectors which are *not* ϵ-orthogonal to h_1 is evaluated as $e^{-n\delta^2}$. Therefore, for k vectors $h_1, \ldots, h_k$, the fraction of the vectors which are not ϵ-orthogonal to $h_1, \ldots, h_k$ is at most $ke^{-n\delta^2}$. The probability to select randomly a vector h_{k+1}, which is ϵ-orthogonal to $h_1, \ldots, h_k$, is higher than $1 - ke^{-n\delta^2}$. The vectors are selected independently, therefore we have the estimate (23).

For $N \gg 1$, the value of $P(\epsilon, N)$ in (23) can be estimated as follows. For $Ne^{-\frac{n\epsilon^2}{2}} < 1$:

$$P(\epsilon, N) > \left(1 - Ne^{-\frac{n\epsilon^2}{2}}\right)^N \sim e^{-N^2 e^{-\frac{n\epsilon^2}{2}}}. \tag{24}$$

According to (24), if $P(\epsilon, N)$ is set to be exceed a certain value, $1 - \vartheta$, the number of pairwise almost orthogonal vectors in $\mathcal{B}^n(1)$ will have the following asymptotic estimate: For

$$\boxed{N \le e^{\frac{\epsilon^2 n}{4}} \left[\log\left(\frac{1}{1 - \vartheta}\right)\right]^{\frac{1}{2}}} \tag{25}$$

random vectors $h_1, \ldots, h_{N+1}$ are pairwise ϵ-orthogonal with probability $P > 1 - \vartheta$.

Estimate (24) of (23) can be refined if we apply log to the right hand side of (23)

$$\log P(\epsilon, N) = \sum_{k=1}^{N} \log\left(1 - ke^{-\frac{n\epsilon^2}{2}}\right),$$

end estimate the above sum using the integral

$$J(z) = \int_0^z \log(1 - xr)dx = \frac{rz - 1}{r}\log(1 - rz) - z, \ r = e^{-\frac{n\epsilon^2}{2}}$$

Since $\log P(\epsilon, N)$ is monotone for $e^{\frac{n\epsilon^2}{2}} > N \ge 1$ we can conclude that $J(N+1) \le \log P(\epsilon, N) \le J(N)$. Furthermore, given that

$$\log(1 - x) \le -\frac{2x}{2 - x} \ \text{for all } x \in [0, 1]$$

14

the following holds

$$J(z) \geq \frac{2z(rz-1)}{rz-2} - z$$

for all $zr \in [0,1]$. Hence

$$\frac{2(N+1)(r(N+1)-1)}{r(N+1)-2} - (N+1) \leq P(\epsilon, N) = \log(1-\vartheta).$$

Transforming this equation into quadratic

$$r(N+1)^2 - \log(1-\vartheta)r(N+1) + 2\log(1-\vartheta) \leq 0$$

and solving for N results in

$$\boxed{N \leq \sqrt{\frac{\log^2(1-\vartheta)}{4} + 2\log\frac{1}{1-\vartheta}e^{\frac{n\epsilon^2}{2}}} + \frac{\log(1-\vartheta)}{2}} \tag{26}$$

Notice that the refined estimate (26) has asymptotic exponential rate of order $e^{\epsilon^2 n/4}$ which is identical to the one derived in (25).

5. Discussion

Estimate (25), (26) derived in the previous section suggests that, for ϑ sufficiently small a set of N randomly and independently chosen vectors in $\mathcal{B}^n(1)$ will be pairwise ϵ-orthogonal with probability $1 - \vartheta$ for

$$N \lesssim e^{\frac{\epsilon^2 n}{4}} \vartheta^{\frac{1}{2}}$$

Such exponential bound enables us to explain some apparent controversy [34] regarding convergence rates of approximation schemes based on iterative greedy approximation [16], [4], and randomized choice of functions advocated in [15], [26], [27].

Both greedy approximation and systems of randomized basis functions enjoy the convergence rate of order $1/N^{1/2}$ in L_2-norm irrespective of dimensionality of the domain on which a function that is being approximated is defined. Greedy approximation, however, requires solving nonlinear optimization problems at each step. In randomized approached to approximation parameters of the basis functions/kernels are randomly drawn from a given set. This transforms the original problem into a much simpler linear one.

Notice, however, that the constant C in the error rate $O(C/N^{1/2})$ can become rather large in these schemes. If, for example kernels $\phi(x, \omega)$ in Theorem 2 approach Dirac delta-functions, then the constant C in (16) becomes arbitrarily large. The same problem may occur for the error rate (15). This means that practical application of randomized approximation schemes is to be preceded by additional analysis of the class of functions being approximated. Second, achievable rate in these schemes is in general dependent on the values of δ,

15

determining probability of success (cf. (16); see our comment on Monte-Carlo convergence in probability too). The smaller the value of δ the higher is the probability of achieving the rate $O(C/N^{1/2})$. That rate, however, is proportional to a strictly monotone function of $1/\delta$. This might be acceptable in many practical applications. Nevertheless, if particularly high accuracy is desirable one needs to take this into account.

An insight into the slowing-down of convergence of randomized approximators due to either increased C or due to the need to keep δ small can be gained by considering measure concentration effects discussed in Propositions 2, 3, Theorem 3 and in Section 4.4. Indeed, in accordance with the analysis presented a system of $n - k$, $k \geq 1$ vectors spans almost all vectors in a ball $\mathcal{B}^n(1)$. This system, however, is almost linearly dependent. This means that the coefficients in corresponding linear combinations could become arbitrary large. Large coefficients, in turn, make the representation sensitive to small inaccuracies which is of course generally not very desirable. This observation of typicality of large coefficients in randomized choice of relatively short bases is consistent with our earlier remarks about growth of C and influence of $1/\delta$. On the other hand, if a representation of data is needed in which all coefficients are to be within certain bounds then the number of randomized basis vectors may become exponentially large.

To overcome potential danger of exponential growth of the number of elements needed to achieve acceptable level of performance of a randomized approximator one may constrain dimensionality of randomization by "binning" or partitioning the space of original problem into a set of spaces of smaller dimension. An example of such an optimization would be to use e.g. multi-scale basis functions, followed by randomization at each scale. Cascading, multi-grid, frequency separation, localization could all be used for the same purposes. The achieved efficiency will of course be determined by suitability of each of these "binning" approaches for a problem at hand.

6. Examples

6.1. Error convergence rate. Deterministic vs Randomized approaches

In order to illustrate the main difference between greedy and RVFL approximators, we consider the following example in which a simple function is approximated by both methods, greedy approximation (4)–(7) and approximation based on randomized choice of bases. Let $f(x)$ be defined as follows:

$$f(x) = 0.2e^{-(10x-4)^2} + 0.5e^{-(80x-40)^2} + 0.3e^{-(80x-20)^2}$$

The function $f(x)$ is shown in Fig. 2, top panel. Clearly, $f(\cdot)$ belongs to the convex hull of G, and hence to its closure.

First, we implemented greedy approximation (4)–(7) in which we searched for g_n in the following set of functions

$$G = \{e^{-(w^T x + b)^2}\},$$

16

where $w \in [0, 200]$, $b \in [-100, 0]$. The procedure for constructing f_n was as follows. Assuming $f_0(x) = 0$, $e_0 = -f$ we started with searching for w_1, b_1 such that

$$
\begin{aligned}
\langle 0 - f(x), g(w_1 x + b) - f(x) \rangle = \\
- \langle f(x), g(w_1 x + b) \rangle + \|f(x)\|^2 < \varepsilon.
\end{aligned}
\tag{27}
$$

where ε was set to be small ($\varepsilon = 10^{-6}$ in our case). When searching for a solution of (27) (which exists because the function f is in the convex hull of G [16]), we did not utilize any specific optimization routine. We sampled the space of parameters w_i, b_i randomly and picked the first values of w_i, b_i which satisfy (27). Integral (27) was evaluated in quadratures over a uniform grid of 1000 points in $[0, 1]$.

The values of α_1 and the function f_1 were chosen in accordance with (7) with $M'' = 2$, $M' = 1.5$ (these values are chosen to assure $M'' > M' > \sup_g \|g\| + \|f\|$). The iteration was repeated, resulting in the following sequence of functions

$$
f_n(x) = \sum_{i=1}^{n} c_i g(w_i^T x + b_i),
$$

$$
c_i = \alpha_i (1 - \alpha_{i+1})(1 - \alpha_{i+2}) \cdots (1 - \alpha_n)
$$

Evolution of the normalized approximation error

$$
\bar{e}_n = \frac{e_n^2}{\|f\|^2} = \frac{\|f_n - f\|^2}{\|f\|^2}
\tag{28}
$$

for 100 trials is shown in Fig. 2 (middle panel). Each trial consisted of 100 iterations (4)–(7), thus leading to the networks of 100 elements at the 100th step. We observe that the values of $\bar{e}_n$ monotonically decrease as $O(1/n)$, with the behavior of this approximation procedure consistent across trials.

Second, we implemented an approximator based on the Monte-Carlo integration. At the nth step of the approximation procedure we pick randomly an element from G, where $w \in [0, 200]$, $b \in [-200, 200]$ (uniform distribution). After an element is selected, we add it to the current pool of basis functions

$$
P_{n-1} = \{g(w_1^T x + b_1), \ldots, g(w_{n-1} x + b_{n-1})\}.
$$

Then the weights c_i in the superposition

$$
f_n = \sum_{i=1}^{n} c_i g(w_i^T x + b_i)
$$

are optimized so that $\|f_n - f\| \to \min$. Evolution of the normalized approximation error $\bar{e}_n$ (28) over 100 trials is shown in Fig. 2 (panel (c)). As can be observed from the figure, even though the values of $\bar{e}_n$ form a monotonically decreasing sequence, they are far from $1/n$, at least for $1 \leq n \leq 100$. Behavior across trials is not consistent, at least for the networks smaller than 100 elements, as indicated by a significant spread among the curves.
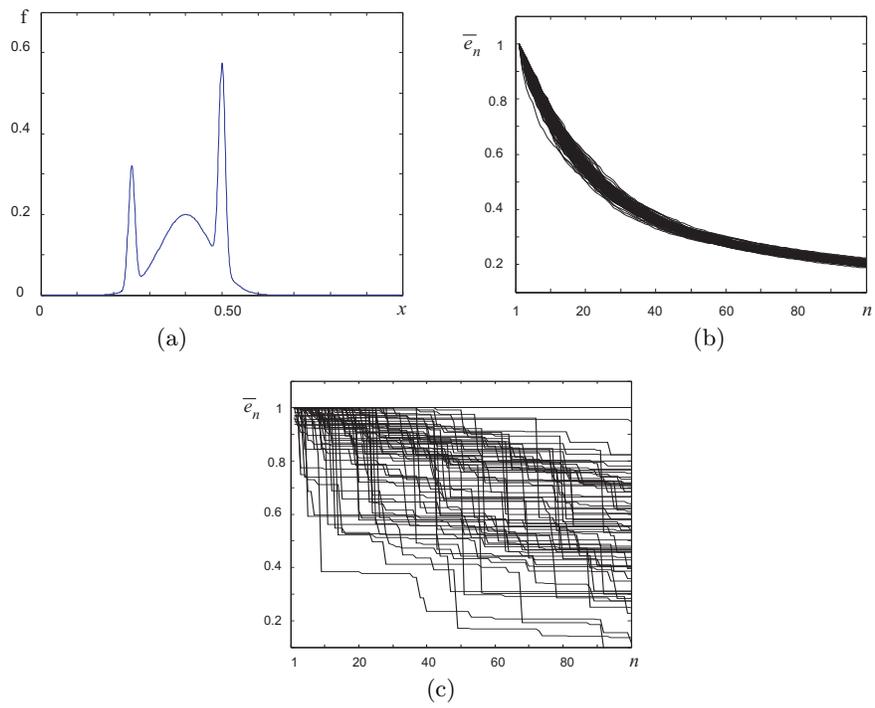
Figure 2: Practical speed of convergence of function approximators that use greedy algorithm (panel (b)) and Monte-Carlo based random choice of basis functions (panel (c)). The target function is shown in panel (a).
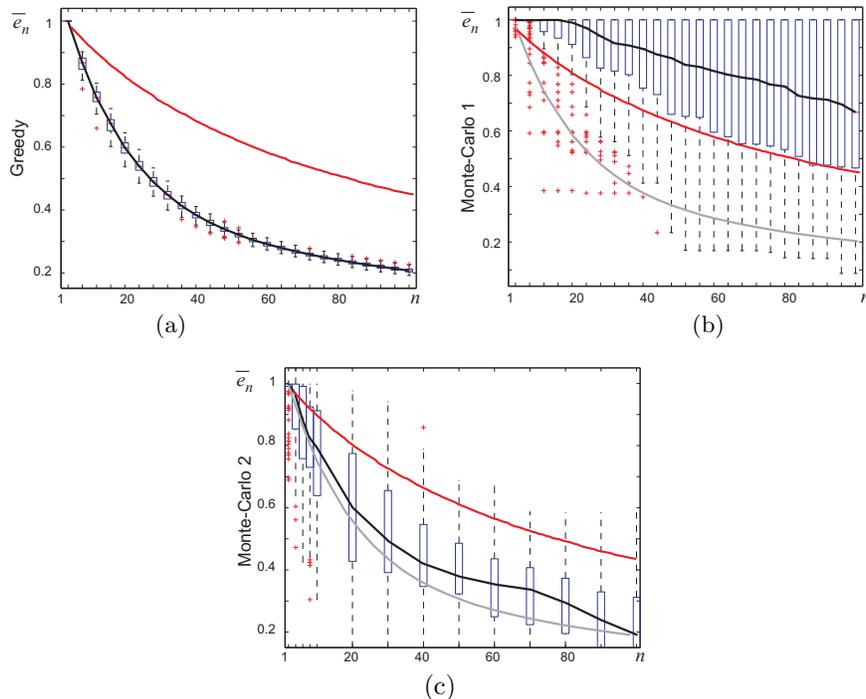
18

Figure 3: Box plots of convergence rates for function approximators that use greedy algorithm (panel (a)) and Monte-Carlo random choice of basis functions (panels (b) and (c)). Panel (b) corresponds to the case in which the basis functions leading to ill-conditioning were discarded. Panel (c) shows performance of the MLP trained by the method in [7] which is effective at counteracting ill-conditioning while adjusting the linear weights only. The red curve shows the upper bound for $\bar{e}_n$ calculated in accordance with (6). We duplicated the average performance of the greedy algorithm (grey solid curve) in the middle and bottom panels for convenience of comparison.

Overall comparison of these two methods is provided in Fig. 3, in which the errors $\bar{e}_n$ are presented in the form of a box plot. Black solid curves depict the median of the error as a function of the number of elements, n, in the network; blue boxes contain 50% of the data points in all trials; "whiskers" delimit the areas containing 75% of data, and red crosses show the remaining part of the data. As we can see from these plots, random basis function approximators, such as the RVFL networks, mostly do not match performance of greedy approximators for networks of reasonable size. Perhaps, employing integration methods with variance minimization could improve the performance. This, however, would amount to using prior knowledge about the target function f, making it difficult to apply the RVFL networks to problems in which the function f is uncertain.

Now we demonstrate performance of an MLP trained to approximate this target function. The NN is trained by a gradient based method described in

[7]. At first, the full network training is carried out for several network sizes $n = 20, 40, 60, 80$ and 100 and input samples randomly drawn from $x \in [0, 1]$. The values of $\bar{e}_n$ are $1.5 \cdot 10^{-4}$ for all the network sizes (as confirmed in many training trials repeated to assess sensitivity to weight initialization). This suggests that training and performance of much smaller networks should be examined. The networks with $n = 2, 4, 6, 8, 10$ are trained, resulting in $\bar{e}_n = 0.5749, 0.1416, 0.0193, 0.0011, 0.0004$, respectively, averaged over 100 trials per the network size. Next, we train only the linear weights (c_i in (2)) of the MLP, fixing the nonlinear weights w_i and b_i to random values. The results for $\bar{e}_n$ averaged over 100 trials are shown in Fig. 3, bottom panel (black curve). Remarkably, the results of random basis network with $n = 100$ are worse than those of the MLP with $n \geq 4$ and full network training. These results indicate that both the greedy and the Monte-Carlo approximation results shown in Fig. 3 are quite conservative. Furthermore, the best of those two, i.e., the greedy approximation's, can be dramatically improved by a practical gradient based training.

6.2. Measure concentration effects

Measure concentration effects, as presented in Section 4, have been discussed for idealized objects such as $\mathcal{S}^{n-1}(R)$ and $\mathcal{B}^n(R)$. The phenomenon, however, broadly applies to other objects whose geometric and formal description is not limited to the former.

In order to illustrate this point we analysed a database of HOG feature vectors [6] containing representations of images of faces[4] as well as the negatives (non-faces). Each feature vector has 1920 components, and hence belongs to $\mathbb{R}^n$ with $n = 1920$. Vectors of each classes have been centered and normalized so that they belong to the hypercube $[-1, 1]^n$. Fig. 4 shows distributions of angles between a randomly chosen vector (1-st) and that of the rest in their respective classes. As one can see from this figure, the angles concentrate in a vicinity of $\pi/2$ which is consistent with our derivations for points in $\mathcal{B}^n(1)$.

Another interesting effect of measure concentration is exponential growth of the lengths of chains of randomly chosen vectors which are pairwise almost orthogonal. In order to illustrate and assess validity of our estimates (25), (26) the following numerical experiments have been performed. A point is first randomly selected in a hypercube $[-1, 1]^n$ of some given dimension. The second point is randomly chosen in the same hypercube. These two points correspond to two vectors randomly drawn in $[-1, 1]^n$. If the angle between the vectors was within $\pi/2 \pm 0.037\pi/2$ then the vector was retained. At the next step a new vector is generated in the same hypercube, and its angles with the previously generated vectors are evaluated. If these angles are within $0.037\pi/2$ of $\pi/2$ then the vector is retained. The process is repeated until the chain of almost orthogonality breaks, and the number of such pairwise almost orthogonal vectors

---

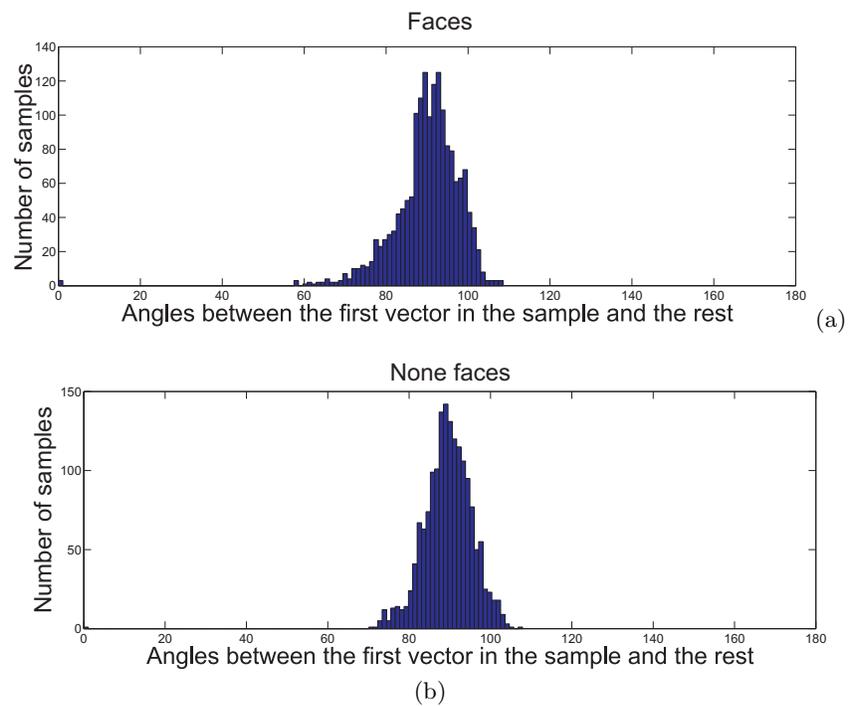[4]The database has been developed by Apical LTD.

Figure 4: Measure concentration in high dimensions. Panel (a) shows histogram of angles between a randomly chosen feature vector in the set of "faces" and the rest of the vectors in the class. Panel (b) shows histogram of angels between a randomly chosen feature vector in the set of "non-faces" (negatives) and the rest of vectors in this class.
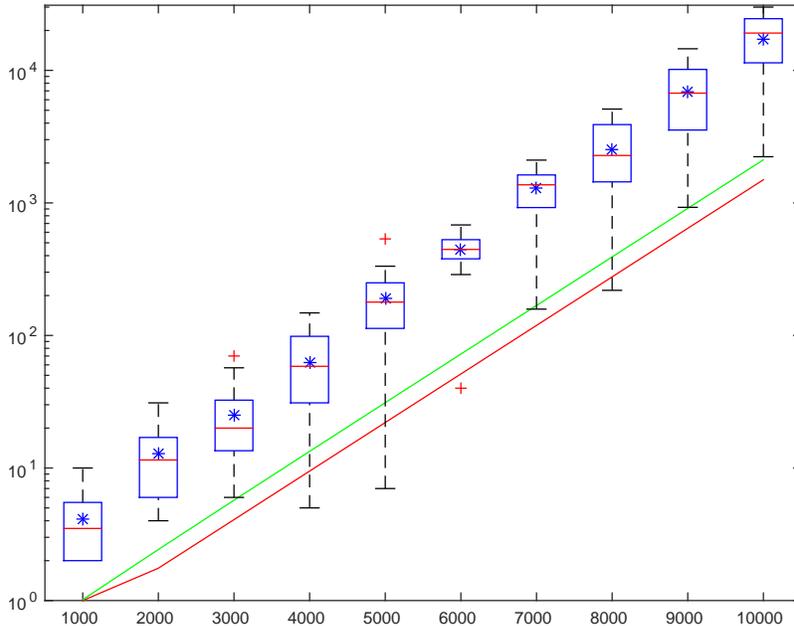
Figure 5: Lengths N of pairwise almost orthogonal chains of vectors that are independently randomly sampled from $[-1, 1]^n$ as a function of dimension, n. For each n 20 pairwise almost orthogonal chains where constructed numerically. Boxplots show the second and third quartiles of this data for each n, red bars correspond to the medians, and blue stars indicate means. Red curve shows theoretical bound (25), and green curve shows refined estimate (26).

(length of the chain) is recorded. Results are shown in Figure 5. Red line corresponds to the conservative theoretical estimate (25), green curve shows refined estimate, (26), and box plot shows lengths of pairwise almost orthogonal chains as a function of dimension. The value of ϑ was set to 0.1 for both theoretical estimates, and our choice of precision margins $\pi/2 \pm 0.037\pi/2$ correspond to $\varepsilon = \cos(0.963\pi/2) = 0.0581$. As we can see from this figure our empirical observations are well aligned with theoretical predictions.

6.3. Approximation of a constant: dimensionality blowup

Another, and perhaps, more interesting illustration of measure concentration and orthogonality effects belongs to the field of function approximation. Let us suppose that we are to approximate a given continuous function defined on an interval $[0, 1]$ by linear combinations of the following type

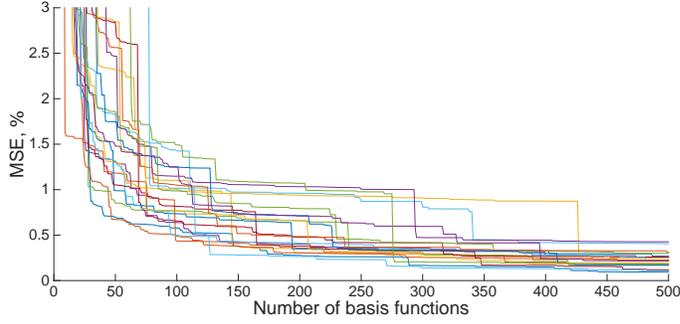$$f_N(x) = \sum_{i=1}^{N} c_i \varphi(a_i, \sigma_i, x), \tag{29}$$

22

Figure 6: Errors of approximation of $f^*(x) = 1$ by linear combinations $f_N(x)$, (29), as functions of N.

where the function φ is defined as follows

$$\varphi(a, \sigma, x) = \begin{cases} 0, & x > a + \sigma/2 \\ 1, & a - \sigma/2 \leq x \leq a + \sigma/2 \\ 0, & x < a - \sigma/2 \end{cases}$$

For simplicity we suppose that the function to be approximated, f^* is a constant:

$$f^*(x) = 1 \ \forall \ x \in [0, 1].$$

Linear combinations of $\varphi(a_i, \sigma_i, \cdot)$ can uniformly approximate every continuous function on $[0, 1]$. Furthermore the chosen function f^* can be represented by just a single element with $a = 0.5, \sigma = 0.5$: $f^*(x) = \varphi(0.5, 0.5, x)$. Since we assumed no prior knowledge of these parameters, we approximated the function f^* with linear combinations (29) in which the values of a_i, σ_i were chosen randomly in the interval $[0, 1]$, and the values of c_i were chosen as follows

$$c_1, \ldots, c_N = \arg \min_{c_1, \ldots, c_N} \int_0^1 (f^*(x) - f_N(x))^2 dx. \tag{30}$$

In order to evaluate performance of approximation as a function of N the following iterative procedure has been used. On the first step the values of a_1 and σ_1 are randomly drawn from the interval $[0, 1]$. This is followed by finding the value of c_1 in accordance with (30) (it is clear though that $c_1 = 1$). Next the values of a_2, σ_2 are drawn from $[0, 1]$ followed by determination of optimal weights c_1, c_2. The L_2 error of approximation is recorder for each step. The process repeats until $N = 500$.

Fig. 6 presents 20 different error curves corresponding to different growing systems of functions $\{\varphi(a_i, \sigma_i, \cdot\}_{i=1}^N$. Despite that problem is both a) simple and b) admits explicit solutions with respect to the values of $c_1, \ldots, c_N$, performance

23

of such approximations in terms of convergence rates is far from ideal. One can observe that initially there is a significant drop of the approximation error for $N \leq 100$. After this value, however, the error decays very slowly and its rate of decay nearly stalls for N large.

One of potential explanation of this effect is as follows. Fig. 7 shows functions $f^*(x) - f_N(x)$ for $N = 5$, 50, and 500 along a single typical curve from Fig. 6. As we can see from the these figures the error functions become more and more pitchy or spiky with N. The individual spikes are randomly distributed on $[0, 1]$, and their thickness is converging to zero. To compensate for such errors one needs to be able to generate a very narrow $\varphi(a_i, \sigma_i, \cdot)$ which, in addition, is to be placed in the right location. Effectively, if error reduction at each step is sought for, this is equivalent to dimensionality growth of the problem at each step. However, in accordance with (25) to overcome negative effect of measure concentration, one needs to accumulate exponentially large number of samples. This is reflected in very slow convergence at the end of the process.

7. Conclusion

In this work we demonstrate that, despite increasing popularity of random basis function networks in control literature, especially in the domain of intelligent/adaptive control, one needs to pay special attention to practical aspects that may affect performance of these systems in applications.

First, as we analyzed in Section 3 and showed in our examples, although the rate of convergence of the random basis function approximator is qualitatively similar to that of the greedy approximator, the rate of the random basis function approximator is *statistical*. In other words, small approximation errors are guaranteed here in *probability*. This means that, in some applications such as e.g. practical adaptive control in which the RVFL networks are to model or compensate system uncertainties, employment of a re-initialization with a supervisory mechanism monitoring quality of the RVFL network is necessary. Unlike network training methods that adjust both linear and nonlinear weights of the network, such mechanism may have to be made robust against numerical problems (ill-conditioning).

Our conclusion about the random basis function approximators is also consistent with the following intuition. If the approximating elements (network nodes) are chosen at random and not subsequently trained, they are usually not placed in accordance with the density of the input data. Though computationally easier than for nonlinear parameters, training of linear parameters becomes ineffective at reducing errors "inherited" from the nonlinear part of the approximator. Thus, in order to improve effectiveness of the random basis function approximators one could combine unsupervised placement of network nodes according to the input data density with subsequent supervised or reinforcement learning values of the linear parameters of the approximator. However, such a

24
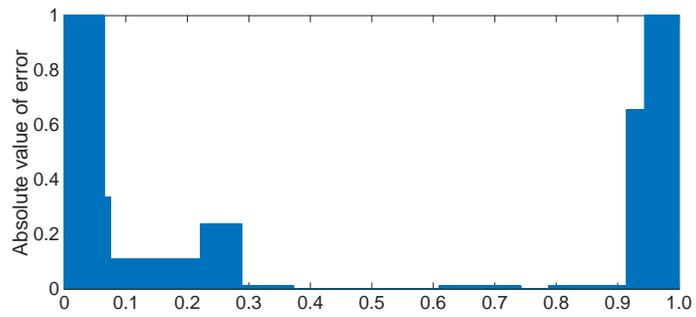
(a)
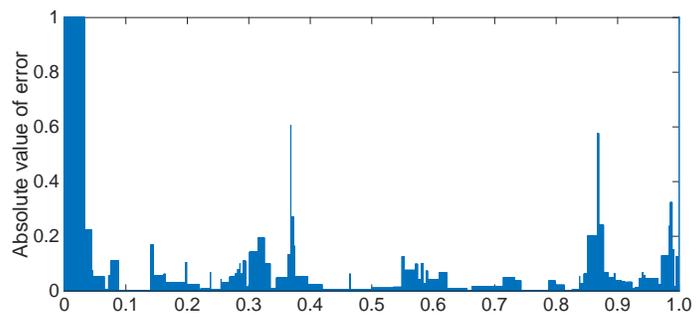
(b)
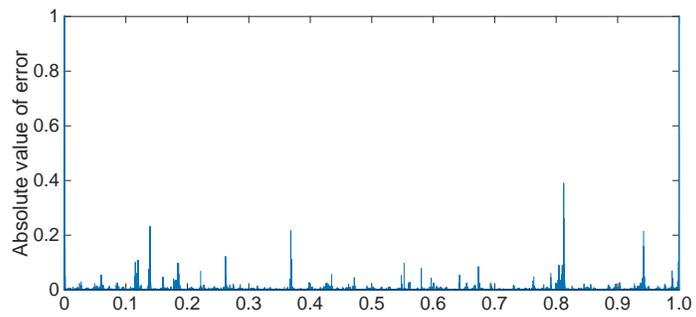
(c)

Figure 7: Error function $f^*(x) - f_N(x)$ for $N = 5$ (panel (a)), $N = 50$ (panel (b)), and $N = 500$ (panel (c)) for a typical curve from Fig. 6.

25

combination of methods is not-trivial because in adaptive control and modeling one often has to be able to allocate approximation resources adaptively – and the full network training seems to be the natural way to handle such adaptation.

Second, we showed that in high dimensions exponentially large number of randomly and independently chosen vectors are almost orthogonal with probability close to one. This implies that in order to represent an element of such high-dimensional space by linear combinations of randomly and independently chosen vectors, it may often be necessary to generate samples of exponentially large length if we use bounded coefficients in linear combinations. On the other hand, if coefficients with arbitrarily large values are allowed, the number of randomly generated elements that are sufficient for approximation is even less than dimension. In the latter case, however, we have to pay for such a significant reduction of the number of elements by ill-conditioning of the approximation problem.

A simple numerical example that illustrates such behavior has been provided. Not only this example demonstrates the effect of measure concentration in a simple approximation problem, it also highlights practical implications for other approximation schemes that are based on randomization.

References

[1] M. Alhamdoosh and D. H. Wang. Fast decorrelated neural network ensembles with random weights. *Information Sciences*, 264:104–117, 2014.

[2] S. Artstein. Proportional concentration phenomena of the sphere. *Israel Journal of Mathematics*, 132:337–358, 2002.

[3] K. Ball. *An Elementary Introduction to Modern Convex Geometry*, volume 31. Flavors of Geometry, MSRI Publications, 1997.

[4] A. R. Barron. Universal approximation bounds for superposition of a sigmoidal function. *IEEE Trans. on Information Theory*, 39(3):930–945, 1993.

[5] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Math. of Control, Signals and Systems*, 2:303–314, 1989.

[6] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Proceedings of the IEEE Computer Vision and Pattern Recognition Conference*, pages 886 – 893. 2005.

[7] L.A. Feldkamp, D.V. Prokhorov, C.F. Eagen, and F. Yuan. Enhanced multi-stream kalman filter training for recurrent networks. In J. Suykens and J. Vandewalle, editors, *Nonlinear Modeling: Advanced Black-Box Techniques*, pages 29–53. Kluwer Academic Publishers, 1998.

[8] A.N. Gorban. Approximation of continuous functions of several variables by an arbitrary nonlinear continuous function of one variable, linear functions, and their superpositions. *Appl. Math. Lett.*, 11(3):45–49, 1998.

[9] A.N. Gorban. Order-disorder separation: Geometric revision. *Physica A*, 374:85–102, 2007.

[10] M. Gromov. *Metric Structures for Riemannian and non-Riemannian Spaces. With appendices by M. Katz, P. Pansu, S. Semmes. Transpated from the French by Sean Muchael Bates.* Birkhauser, Boston, MA, 1999.

[11] M. Gromov. Isoperimetry of waists and concentration of maps. *GAFA, Geomteric and Functional Analysis*, 13:178–215, 2003.

[12] S. Haykin. *Neural Networks: A Comprehensive Foundation.* Prentice Hall, 1999.

[13] P. He and S. Jagannathan. Reinforcement learning based output feedback control of nonlinear systems with input contraints. *IEEE Trans. Systems, Man and Cybernetics*, 51(1):150–154, 2005.

[14] K. Hornik, K. Stinchcombe, and H. White. Universal approximation of an unknown function and its derivatives using multilayer neural networks. *Neural Networks*, (3):551–560, 1990.

[15] B. Igelnik and Y.-H. Pao. Stochastic choice of basis functions in adaptive function approximation and the functional-link net. *IEEE Trans. Neural Networks*, 6(6):1320–1329, 1995.

[16] L. K. Jones. A simple lemma on greedy approximation in Hilbert sapce and convergence rates for projection pursuit regression and neural network training. *The Annals of Statistics*, 20(1):608–613, 1992.

[17] F. Lewis, J. Campos, and R. Selmic. *Neuro-fuzzy control of industrial systems with actuator nonlinearities.* SIAM, 2002.

[18] F. Lewis and S. Ge. Neural networks in feedback control systems. In M. Kutz, editor, *Mechanical Engineers' Handbook*, volume 2. Wiley, third edition, 2006.

[19] J.Y. Lin and W.S. Chow. Comments on "stochastic choice of basis functions in adaptive function approximation and the functional-link net". *IEEE Trans. Neural Networks*, 8(2):452–454, 1997.

[20] W. Liu, J. Sarangapani, G. Venayagamoorthy, L. Liu, D. Wunsch, M. Crow, and D. Cartes. Decentralized neural network-based excitation control of large-scale power systems. *International Journal of Control, Automation, and Systems*, 5(5):526–538, 2007.

[21] Ai-Poh Loh, A.M. Annaswamy, and F.P. Skantze. Adaptation in the presence of general nonlinear parameterization: An error model approach. *IEEE Trans. on Automatic Control*, 44(9):1634–1652, 1999.

[22] Y.H. Pao, G.H. Park, and D.J. Sobajic. Learning and generalization characteristics of the random vector functional-link net. *Neurocomputing*, 6(2):163–180, 1994.

[23] J. Park and I. W. Sandberg. Approximation and radial basis function networks. *Neural Computation*, 5(2):305–316, 1993.

[24] V. Pestov. Is the k-nn classifier in high dimensions affected by the curse of dimensionality? *Computers & Mathematics with Applications*, 65:1427–1437, 2013.

[25] D.V. Prokhorov, V.A. Terekhov, and I.Yu. Tyukin. On the applicability conditions for the algorithms of adaptive control in nonconvex problems. *Automation and Remote Control*, 63(2):262–279, 2002.

[26] A. Rahimi and B. Recht. Uniform approximation of functions with random bases. In *Proceedings of the 46th IEEE Annual Allerton Conference on Communication, Control and Computing*, pages 555 – 561. 2008.

[27] A. Rahimi and B. Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. In *Advances in Neural Information Processing Systems (NIPS)*. 2008.

[28] H. L. Resnikoff and R. O. Wells. *Wavelet Analysis*. Springer, 2002.

[29] F. Rosenblatt. *Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms*. Spartan Books, 1962.

[30] J. Sarangapani. *Neural Network Control of Nonlinear Discrete-Time Systems*. CRC Press, 2006.

[31] W. Schmidt, M. Kraaijveld, and R. Duin. Feedforward neural networks with random weights. In *Proceedings of 11th IAPR International Conference on Pattern Recognition Methodology and Systems*, pages 1–4. 1992.

[32] L. N. Trefethen. *Approximation Theory and Approximation Practice*. SIAM, 2013.

[33] I.Yu. Tyukin, D. Prokhorov, and C. van Leeuwen. Adaptation and parameter estimation in systems with unstable target dynamics and nonlinear parametrization. *IEEE Trans. on Automatic Control*, 52(9):1543–1559, 2007.

[34] I.Yu. Tyukin and D.V. Prokhorov. Feasibility of random basis function approximators for modeling and control. In *Proceedings of the IEEE International Symposium on Intelligent Control ISIC'2009*, pages 1391 – 1396. 2009.

[35] I.Yu. Tyukin, D.V. Prokhorov, and V.A. Terekhov. Adaptive control with nonconvex parameterization. *IEEE Trans. on Automatic Control*, 48(4):554–567, 2003.

[36] B. Widrow, A. Greenblatt, Y. Kim, and D. Park. The no-prop algorithm: A new learning algorithm for multilayer neural networks. *Neural Networks*, 37:182–188, 2013.