Information Sciences 382–383 (2017) 170–178

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Insights into randomized algorithms for neural networks: Practical issues and common pitfalls

Ming Li, Dianhui Wang*

Department of Computer Science and Information Technology, La Trobe University, Melbourne, VIC 3086, Australia

ARTICLE INFO

Article history:
Received 15 August 2016
Revised 2 December 2016
Accepted 4 December 2016
Available online 7 December 2016

Keywords:
Randomized algorithms
Neural networks
Incremental learning
Function approximation

ABSTRACT

Random Vector Functional-link (RVFL) networks, a class of learner models, can be regarded as feed-forward neural networks built with a specific randomized algorithm, i.e., the input weights and biases are randomly assigned and fixed during the training phase, and the output weights are analytically evaluated by the least square method. In this paper, we provide some insights into RVFL networks and highlight some practical issues and common pitfalls associated with RVFL-based modelling techniques. Inspired by the folklore that "all high-dimensional random vectors are almost always nearly orthogonal to each other", we establish a theoretical result on the infeasibility of RVFL networks for universal approximation, if a RVFL network is built incrementally with random selection of the input weights and biases from a fixed scope, and constructive evaluation of its output weights. This work also addresses the significance of the scope setting of random weights and biases in respect to modelling performance. Two numerical examples are employed to illustrate our findings, which theoretically and empirically reveal some facts and limits of such class of randomized learning algorithms.

1. Introduction

Universal approximation capability of neural networks as the theoretical foundation for data modelling has been extensively studied in literature [5,6,8,17]. However, there are few works asserting theoretical bounds on the number of hidden nodes required [13,18]. In other words, a neural network model with fewer number of hidden nodes cannot ensure modelling performance whilst a larger number of hidden nodes may lead to over-fitting phenomenon which implies poor generalization performance.

To resolve this problem, constructive schemes, which starts with a small size of network then incrementally generates hidden nodes and output weights until an acceptable learning performance is achieved, have received considerable attention in the past years [1,12]. Although some convergence properties of these methods can be theoretically established, there exist some limitations in practice due to the extensive search for the hidden parameters. Randomized learning techniques for neural networks become popular in recent years because of their good potential in dealing with large-scale data analysis, fast dynamic modelling, and real-time data processing [4,14,19,20,23]. To the best of our knowledge, researches on randomized algorithms for training neural networks can be tracked back to the 1980s [2]. For single hidden layer feed-forward neural networks, Schmidt et al. tried to randomly assign the input weights and biases from $[-1,1]$ and calculate the output

* Corresponding author.
E-mail address: dh.wang@latrobe.edu.au (D. Wang).

http://dx.doi.org/10.1016/j.ins.2016.12.007
0020-0255/

weights by solving a linear least square problem [21]. Unfortunately, such an attempt on the scope setting lacks scientific justification and it could not ensure the universal approximation ability of the resulted model. In [15,16], Pao and Takefji proposed Random Vector Functional-link (RVFL) networks where the input weights and biases are randomly generated and then fixed. RVFL networks have direct links from the input layer to the output layer, its output weights can be calculated by using a pseudo-inverse of the hidden output matrix. In 1995, Igelnik and Pao established some significant results on the approximation capability of RVFL networks [10]. As indicated in their proofs, however, the scope of the input weights and biases is specified and data dependent. In [22], Ivan and his co-workers compared RVFL approximators with Barron's greedy learning framework [1], and examined the feasibility of RVFL networks. To simplify the specification of the random parameters of in RVFL networks, Husmeier suggested to use symmetric and adjustable intervals for approximating a class of nonlinear maps [9]. For more details about the history and recent developments of randomized method for training neural networks, readers may refer to an informative editorial [23].

In this paper, we address some practical issues and common pitfalls in RVFL-based data modelling. Specifically, we look into some impacts of the scope of random parameters on the model's performance, and empirically show that a widely-used setting (e.g $[-1, 1]$) is misleading. Two illustrations are presented to clarify some incorrect perceptions about the way to randomly assign the input weights and biases. Also, we provide a theoretical verification about the infeasibility of a class of incremental RVFL (IRVFL) networks for universal approximation. Simulation results align with the theoretical analysis, and demonstrate that IRVFL networks have technical limits to model nonlinear maps with arbitrary accuracy.

The remainder of the paper is organized as follows. Section 2 briefly reviews constructive neural networks and presents some practical considerations on RVFL networks. Section 3 provides our theoretical analysis that shows IRVFL networks with specific coefficients will not universally approximate a given map. Section 4 reports some numerical results, revealing and correcting some common pitfalls in randomized learning techniques. Section 5 concludes this paper with some remarks.

2. Related work

2.1. Constructive neural networks

Let $L_2(D)$ denote the space of all Lebesgue-measurable functions f: $R^d \to R$ on a compact set $D \subset R^d$, with the L_2 norm defined as $\| f\|_2 := \sqrt{\langle f, f\rangle} = (\int_D |f(x)|^2 dx)^{1/2}$ where the inner product of f_1 and f_2 is defined as $\langle f_1, f_2\rangle = \int_D f_1(x) f_2(x) dx$. For a target function f: $R^d \to R$, assume a single layer feed-forward network (SLFN) with $L-1$ hidden nodes has already been constructed, i.e., $f_{L-1}(x) = \sum_{j=1}^{L-1} \beta_j g_j(w_j^{\mathrm{T}} x + b_j)$ $(f_0 = 0)$. If the current residual error, denoted as $e_{L-1} = f - f_{L-1}$, is still unacceptable, the problem of incremental learning becomes how to add β_L, g_L (w_L and b_L) leading to $f_L = f_{L-1} + \beta_L g_L$ until the residual error $e_L = f - f_L$ reaches a pre-defined tolerance ϵ for the given task, i.e., $\|e_L\|_2 \le \epsilon$.

In [1], Barron provided the framework of greedy approximation by using the classical Jones iteration method [11]. For a target function f that belongs to closure of the convex hull of a given Hilbert space G, i.e., $f \in \overline{\mathrm{conv}(G)}$, Barron's constructive scheme aims at finding α_L and g_L that minimize $\|\alpha_L f_{L-1} + (1-\alpha_L) g_L - f\|_2$ at each step. However, this strategy is only applicable to $f \in \overline{\mathrm{conv}(G)}$ rather than all Lebesgue-measurable functions $f \in L_2(D)$. Besides, the ability of f_L may be constrained because it is generated via a convex combination of f_{L-1} and g_L, rather than optimizing the output weights over the parameter space.

In [12], the authors proposed an approach by optimizing certain objective functions. Their proposed schemes can ensure the convergence property of the constructed model, provided that g_L, which maximizes $\langle e_{L-1}, g_L\rangle^2 / \|g\|_2^2$, can be found in each incremental step. However, it is not easy to obtain an optimal g_L^* as the optimization process might be frequently plagued by local minima when performing gradient ascent methods. The optimization process required for building a new hidden node is very inefficient, as mentioned in [12]. For many real world applications with large scale dataset and/or with real-time data processing, it is impractical to iteratively find out the parameters of the hidden nodes. Thus, fast algorithms for generating a new hidden node (basis function) are being expected.

2.2. Some practical considerations on RVFL networks

As indicated in [7] and [22], some additional conditions on the families of functions to be approximated are requested to ensure successful data modelling with a class of randomized approximators. It should be noticed that the universal approximation theorem presented in [10] holds for certain appropriate range of random parameters in the hidden nodes. Some theoretical analysis on the feasibility of randomized basis function approximators is given in [7], which proved that in the absence of certain additional conditions one may observe an exponential growth of the number of terms needed to approximate a nonlinear map, and the resulted model may be very sensitive to the random parameters. These work motivates us to highlight some 'risky' aspects caused by the randomness in RVFL networks. In particular, the illogical way of simply selecting a trivial range $[-1, 1]$ for random assignment of the input weights and biases should be questioned and corrected.

3. Theoretical analysis

This part aims to provide a theoretical analysis on the infeasibility of IRVFL networks for universal approximation. For a target function f: $R^d \to R$, IRVFL networks start from generating one (hidden) node $g_1 = g_1(w_1^{\mathrm{T}} x + b_1)$ (g is the activation