Artificial Intelligence in Medicine 70 (2016) 31–40

Contents lists available at ScienceDirect

Artificial Intelligence in Medicine

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

A framework for parameter estimation and model selection in kernel deep stacking networks

Thomas Welchowski*, Matthias Schmid

Department of Medical Biometry, Informatics and Epidemiology, Rheinische Friedrich-Wilhelms-Universität Bonn, Sigmund-Freud-Str. 25, 53127 Bonn, Germany

ARTICLE INFO

Article history:
Received 9 November 2015
Received in revised form 9 March 2016
Accepted 21 April 2016

Keywords:
Deep learning
Artificial neural networks
Kernel regression
Model-based optimization

ABSTRACT

Background and objectives: Kernel deep stacking networks (KDSNs) are a novel method for supervised learning in biomedical research. Belonging to the class of deep learning techniques, KDSNs are based on artificial neural network architectures that involve multiple nonlinear transformations of the input data. Unlike traditional artificial neural networks, KDSNs do not rely on backpropagation algorithms but on an efficient fitting procedure that is based on a series of kernel ridge regression models with closed-form solutions. Although being computationally advantageous, KDSN modeling remains a challenging task, as it requires the specification of a large number of tuning parameters.
Methods and material: We propose a new data-driven framework for parameter estimation, hyperparameter tuning, and model selection in KDSNs. The proposed methodology is based on a combination of model-based optimization and hill climbing approaches that do not require the pre-specification of any of the KDSN tuning parameters. We demonstrate the performance of KDSNs by analyzing three medical data sets on hospital readmission of diabetes patients, coronary artery disease, and hospital costs.
Results: Our numerical studies show that the run-time of the proposed KDSN methodology is significantly shorter than the respective run-time of grid search strategies for hyperparameter tuning. They also show that KDSN modeling is competitive in terms of prediction accuracy with other state-of-the-art techniques for statistical learning.
Conclusions: KDSNs are a computationally efficient approximation of backpropagation-based artificial neural network techniques. Application of the proposed methodology results in a fast tuning procedure that generates KDSN fits having a similar prediction accuracy as other techniques in the field of deep learning.

1. Introduction

Deep learning methods [1,2] are among the most powerful techniques for supervised and unsupervised learning in biomedical research. Originating in the artificial intelligence and pattern recognition fields, deep learning methods have been used, among many other examples, to predict splicing patterns in tissues [3], to annotate the pathogenicity of genetic variants [4], to analyze basal cell carcinoma images [5] and to learn cellular signaling systems [6]. A notable example of the success of deep learning techniques is the Merck Molecular Activity Challenge contest hosted by Kaggle, where deep learning outperformed competing techniques to predict molecular activity from chemical structures represented by molecular descriptors [7].

Conceptually, deep learning methods can be regarded as an extension of artificial neural networks with one hidden layer ("one-hidden-layer multi-layer perceptrons"), which have become an established tool to address learning tasks in biomedical research, see, e.g. [8–10] for recent publications in this field. A key property of one-hidden-layer multi-layer perceptrons is their ability to approximate any continuous function of the input data arbitrarily well ("universal function approximators" [11]). Deep learning methods extend neural networks with one hidden layer by multiple – potentially less complex – hidden layers, so that higher-level dependencies between transformations of the input data can be represented more efficiently than in one-hidden-layer multi-layer perceptrons [1]. This is in contrast to many other regression

* Corresponding author.
E-mail addresses: welchow@imbie.meb.uni-bonn.de (T. Welchowski), schmid@imbie.meb.uni-bonn.de (M. Schmid).

http://dx.doi.org/10.1016/j.artmed.2016.04.002
0933-3657/

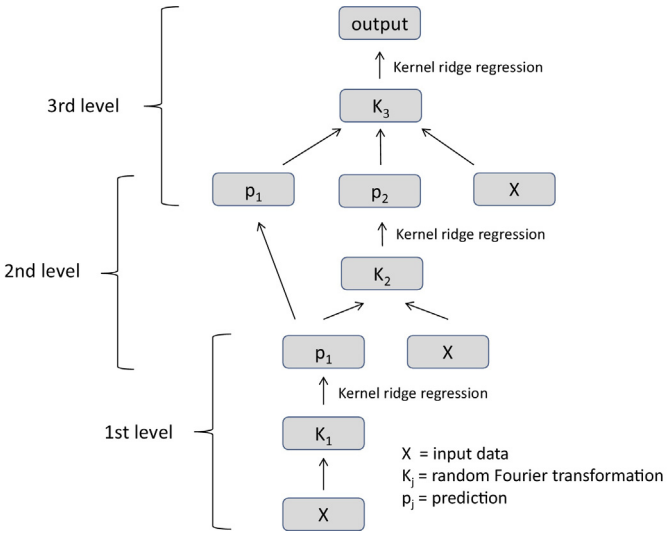

Fig. 1. Example of a KDSN with three levels.

methods (such as generalized additive models [12], finite mixture models [13] or neural networks with one hidden layer), which are based on *shallow architectures* involving at most two nonlinear transformations of the input data.

In this paper we consider *kernel deep stacking networks* (KDSNs [14,15]), which are a novel supervised deep learning method for continuous and binary outcome variables. KDSNs are defined by a series of approximations to single-layer networks (in the following termed "levels") for which the predictions obtained from lower levels are repeatedly added to the input space. Estimation of a KDSN is based on level-wise regularized regression models that are fitted to the kernel-transformed input data (see Fig. 1 for a schematic overview). This estimation procedure is a key advantage of KDSNs, as it reduces model fitting to a series of convex optimization problems with closed-form solutions, thereby replacing traditional estimation techniques such as backpropagation (which has been criticized for its slow convergence and its tendency to get stuck in local optima [16]). To increase the efficiency of KDSN fitting in high-dimensional settings, random Fourier transformations can be applied to the data [15,17].

A difficult problem arising from the flexibility of KDSNs (and of deep learning methods in general) is the specification of a large number of tuning parameters. These do not only comprise the number of KDSN levels ("model selection") but also several level-specific parameters needed for Fourier transformations and regularized kernel estimation. As a consequence, tuning KDSNs becomes an intrinsically complex and difficult task. Because traditional grid search strategies require massive computing power if applied to high-dimensional search spaces [7], many published results on deep learning are based on human-guided tuning (e.g. [18,19]), with little details on the optimization of tuning parameters being provided. Obviously, the lack of a data-driven strategy to select tuning parameters in KDSNs limits the use of the method in biomedical research.

To address this issue, we propose a fully data-driven framework for parameter estimation and model selection in KDSNs. The backbone of our method is a hill climbing algorithm (e.g. [20]) that is designed to identify the optimal number of levels in the network. Within each level, tuning parameters are determined by application of a Kriging approach for model-based optimization of prediction accuracy [21–23]. As will be demonstrated in Section 3.1, the average run-time of the proposed method is considerably shorter than grid search strategies for backpropagation-based neural network methods [24]. Regarding predictive performance (Section 3.2), KDSN modeling within the proposed framework is competitive with other state-of-the-art techniques for supervised learning. This will be illustrated by the analysis of two medical data sets from the Department of Biostatistics' data repository at Vanderbilt University (http://biostat.mc.vanderbilt.edu/DataSets) and another data set that was extracted from the Health Facts Database (Cerner Corporation, Kansas City, MO, see [25]). The latter data are used to develop a KDSN-based prediction model for hospital readmission rates of diabetes patients, which is a central issue in transitional care intervention and hospital discharge planning [26].

2. Methods

In Section 2.1 the basic notation is introduced and a formal definition of KDSNs is provided. The proposed methodology for parameter estimation and model selection is described in Section 2.2.

2.1. Kernel deep stacking networks

The goal of KDSNs is to derive a prediction rule $f(\boldsymbol{x})$ for a continuous or binary outcome variable y based on a vector of predictor variables $\boldsymbol{x} \in \mathbb{R}^d$. KDSNs are usually trained on a set of learning data that are represented by a vector of outcome values $\boldsymbol{y} \in \mathbb{R}^n$ and an input data set $\boldsymbol{X} = (\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n)^\top \in \mathbb{R}^{n \times d}$. We assume that the columns of $\boldsymbol{X}$ have been standardized before analysis by their respective medians and median absolute deviations. As shown in Fig. 1, the estimation of KDSNs (given fixed values of the tuning parameters) is carried out by solving a series of regularized kernel regression problems, emulating the fitting of a multi-layer neural network [14].

In the *first level* of a KDSN, the input data are transformed into a radial basis function (RBF) kernel matrix

$$\boldsymbol{K} = (K(\boldsymbol{x}_i, \boldsymbol{x}_j))_{i,j=1,\ldots,n} \in \mathbb{R}^{n \times n}, \tag{1}$$

where $K(\boldsymbol{x}_i, \boldsymbol{x}_j) := \exp(-\left\|\boldsymbol{x}_i - \boldsymbol{x}_j\right\|_2^2/(2\tau^2))$ denotes the RBF kernel function with scale parameter τ [27]. Because the transformation in Eq. (1) corresponds to a mapping of the input data into an infinite dimensional space [28, pp. 36–39], it can be thought of as an implicit, large hidden layer in the context of neural networks. Because radial basis functions are universal function interpolators [29], solving the regression problem of $\boldsymbol{y}$ on $\boldsymbol{K}$ is based on the same rationale as fitting a neural network with one hidden layer.

To avoid overfitting, the kernel regression problem can be solved by applying ridge-regularized estimation, giving rise to the optimization problem

$$\underset{\boldsymbol{\beta}}{\operatorname{argmin}}\left(\left(\boldsymbol{y} - \boldsymbol{K}\boldsymbol{\beta}\right)^T \left(\boldsymbol{y} - \boldsymbol{K}\boldsymbol{\beta}\right) + \eta \;\; \boldsymbol{\beta}^T\boldsymbol{K}\boldsymbol{\beta}\right) \tag{2}$$

with parameter vector $\boldsymbol{\beta} \in \mathbb{R}^n$ and tuning parameter $\eta > 0$ [14,30]. The vector $\boldsymbol{\beta}$ replaces the weights of the interconnections between the neurons of a neural network. Statistically, $\boldsymbol{\beta}$ has the same interpretation as the coefficient vector of a linear regression model with design matrix $\boldsymbol{K}$ and outcome variable y. The closed-form solution to (2) is given by

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{K} + \eta\boldsymbol{I})^{-1}\boldsymbol{y}; \quad \boldsymbol{I} \in \mathbb{R}^{n \times n}, \tag{3}$$

implying that local optima and convergence problems arising in neural networks with backpropagation estimation are avoided. Note that (3) is calculated regardless of whether $\boldsymbol{y}$ is continuous or binary.

To handle storage and memory problems in situations where n and $\boldsymbol{K}$ are large, Huang et al. [15] proposed to approximate $\boldsymbol{K}$ by an additional random Fourier transformation. This strategy, which is