Journal of Biomedical Informatics 83 (2018) 159–166

Contents lists available at ScienceDirect

Journal of Biomedical Informatics

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

Learning a single-hidden layer feedforward neural network using a rank correlation-based strategy with application to high dimensional gene expression and proteomic spectra datasets in cancer detection

Smaranda Belciug[a,*], Florin Gorunescu[b]

[a] Department of Computer Science, University of Craiova, Craiova 200585, Romania
[b] Royal Society of Medicine, United Kingdom

ARTICLE INFO

Keywords:
Extreme learning machine
Single-hidden layer feedforward neural network
Adaptive hidden nodes initialization
Automated cancer detection
Microarray
Mass spectrometry

ABSTRACT

Methods based on microarrays (MA), mass spectrometry (MS), and machine learning (ML) algorithms have evolved rapidly in recent years, allowing for early detection of several types of cancer. A pitfall of these approaches, however, is the overfitting of data due to large number of attributes and small number of instances – a phenomenon known as the 'curse of dimensionality'. A potentially fruitful idea to avoid this drawback is to develop algorithms that combine fast computation with a filtering module for the attributes. The goal of this paper is to propose a statistical strategy to initiate the hidden nodes of a single-hidden layer feedforward neural network (SLFN) by using both the knowledge embedded in data and a filtering mechanism for attribute relevance. In order to attest its feasibility, the proposed model has been tested on five publicly available high-dimensional datasets: breast, lung, colon, and ovarian cancer regarding gene expression and proteomic spectra provided by cDNA arrays, DNA microarray, and MS. The novel algorithm, called *adaptive* SLFN (aSLFN), has been compared with four major classification algorithms: traditional ELM, radial basis function network (RBF), single-hidden layer feedforward neural network trained by backpropagation algorithm (BP-SLFN), and support vector-machine (SVM). Experimental results showed that the classification performance of aSLFN is competitive with the comparison models.

1. Introduction

Gene expression arrays offer an effective solution for analyzing expression of known genes and transcripts, thus providing valuable information of gene activity in biological samples. Proteomic spectra have evolved into an essential tool for identifying proteomic patterns in serum, widely used in analyzing biological samples. Prospects for effective and reliable cancer diagnosis and treatment have improved significantly with the use of MA and MS technologies [1–8]. The massive amount of data resulting from these technologies needs to be analyzed by complex computational tools, among which the ML algorithms stand out. ML and MA technologies have been successfully used for non-small cell lung cancer [9], identification of varying genes expression in breast cancer [10], and quantitative diagnosis of breast tumors [11]. MS and ML techniques have also been fruitfully used in differentiation of benign and malignant liver tumors [12], breast cancer [13], and colorectal cancer [14].

However, there are two practical issues limiting the use of ML algorithms as classifiers for microarray data and mass spectra from proteomics. One is the '*curse of dimensionality*' because thousands or tens of thousands of features characterize these data. The other is the '*curse of dataset sparsity*' because the number of available instances is limited. In this context, the development of ML algorithms that combine fast computation speed with feature selection capability could solve these two problems, with a special focus on computer-aided medical diagnosis.

Fairly recently, extreme learning machine (ELM) has been proposed as a new learning algorithm for SLFNs [15–17]. It is noteworthy that this approach has caused some debate in the ML community [18]. Yet, various versions of ELM have been proposed [19–21], and ELM and its variants are now used in various medical applications including heart diseases [22], diabetes [23], and detection of different types of cancer (lung, breast, leukemia, and colon) [24–26].

A proper initialization of the weights in a neural network is a matter of great importance to its convergence. Although there is a rich literature on this topic, we will mention only a small part of it. In [27], an algorithm based on Cauchy's inequality and a linear algebraic method for determining the optimal initial weights of feedforward neural

* Corresponding author.
E-mail address: smaranda.belciug@inf.ucv.ro (S. Belciug).

https://doi.org/10.1016/j.jbi.2018.06.003
Received 27 July 2017; Received in revised form 5 June 2018; Accepted 7 June 2018

networks was proposed. A genetic algorithm approach was used as an alternative to the backpropagation (BP) learning algorithm for both synaptic weights initialization and optimization of a SLFN [28]. Deep and recurrent neural networks can be trained successfully by a well-designed random initialization and a particular type of slowly increasing schedule for the momentum parameter, used by the stochastic gradient descent [29]. A training technique was proposed in [30] combining error-correction learning with posterior probability distribution of weights given the error function, and Goodman–Kruskal Gamma rank correlation, assembling them in a Bayesian learning strategy. In [31], a method for weight initialization is proposed for deep net learning, Layer-sequential unit-variance (LSUV), consisting of pre-initializing weights of each convolution or inner-product layer with orthonormal matrices, and normalizing the variance of the output of each layer, from the first to the final layer.

The learning paradigm behind ELM is based on the random choice of the hidden nodes, seen as the key component of the ELM model, and the analytical calculation of the output weights. In the classical case, ELM is trained in two steps: (1) the weights connecting inputs and hidden nodes are randomly assigned and never updated, and (2) a least square solution is used for the output layer. Inspired by the way of initializing the hidden nodes of ELM, the current work proposes a novel technique to set the parameters of the hidden layer of a SLFN. One of the major debate topics regarding ELM is the random initialization, which is independent from applications. Our approach has been conceived as an alternative to this initialization. Instead of randomly generating the hidden nodes of SLFN, they are now problem-dependent and estimated using the statistical relation between attributes and class labels. In real-world applications, problems of nonlinear monotonic relationship between variables, and the existence of many tied observations in data are frequently encountered. Under these circumstances, we considered generating the hidden nodes using the non-parametric Goodman-Kruskal Gamma rank correlation between attributes and class labels, thus having a reliable quantification of the knowledge embedded in data. The current approach offers the possibility of dealing with very large datasets or dataset dimensions by substantially speeding up training time as compared to BP, and enhances classification capability due to the novel initialization of the input weights in one step, while avoiding the possible drawback generated by their random choice. However, the computation speed may decrease in case of large dataset or large dimensions, due to the fact that all the rank correlations between attributes and class labels must be calculated and used. To avoid this situation commonly encountered in practice, an embedded filtering module of each attribute's contribution has been designed. Thus, only the rank correlations that are truly significant for classification, based on the corresponding p-value, are kept for the initiation of the hidden node parameters.

The remainder of this paper is organized in five sections. Section 2 presents both the design and implementation of the novel model. Section 3 presents the benchmarking datasets and briefly summarizes the statistical framework for performance assessment. Section 4 presents the experimental results, model assessment, and corresponding discussions. Section 5 addresses the conclusions and future work.

2. Materials and methods

This paper proposes a novel initialization of a SLFN's input weights, using the knowledge embedded in the connections between attributes and class labels, expressed by the non-parametric Goodman-Kruskal Gamma rank correlation. The new algorithm has been inspired by ELM through the property that parameters of hidden nodes do not need to be updated, and the computation of the network output. A short presentation of the traditional ELM training algorithm will be given below, followed by the novel adaptive initialization of the input weight and the embedded filtering module algorithm.

2.1. Summarization of ELM algorithm

ELM represents a special case of SLFN, characterized by a single layer of hidden units. The synaptic weights connecting inputs to hidden units are randomly initialized, whereas the synaptic weights between the hidden units and the outputs are optimized by a Moore-Penrose generalized inverse. Three steps summarize the ELM training algorithm [15]:

Given the training set TS, the activation function $f(x)$, and a number $\widetilde{N}$ of hidden nodes:

Step 1: Randomly assign the input weight $\mathbf{w}_i$ and bias b_i, $i = 1, 2, \ldots, \widetilde{N}$.
Step 2: Calculate the hidden layer output matrix H.
Step 3. Calculate the output weight $\beta = H^{+} T$, where H is the hidden layer output matrix, H^{+} is the Moore-Penrose generalized inverse of H, and T is the output vector.

2.2. Knowledge embedded in data

A TS used in supervised learning contains objects characterized by inputs (features) and outputs (classes). Consider that TS contains N objects $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N$. Each object in the dataset is coded as a vector $\mathbf{x}_k = (x_1^k, \ldots x_i^k, \ldots x_p^k, y_k)$, where x_i^k, $i = 1, 2, \ldots, p$, represents the i-th feature of the object $\mathbf{x}_k$, $k = 1, 2, \ldots, N$, and y_k represents the label corresponding to object $\mathbf{x}_k$, that is the decision class C_j , $j = 1, 2, \ldots, q$. From a probabilistic point of view, one can reasonably assume that, for each $k = 1, 2, \ldots, N$, the attribute values x_i^k belonging to the attribute A_i, $i = 1, 2, \ldots, p$, are governed by a random variable (r.v.)X_i. The set $\{x_i^1, x_i^2, \ldots x_i^N\}$ represents, from a statistical point of view, a random sample of length N corresponding to the r.v. X_i. Without loss of generality, one can consider the naïve assumption that all attributes are independent of each other, i.e., the parent r.v.'s X_i, $i = 1, 2, \ldots, p$ are independent. One can also assume that, for each object $\mathbf{x}_k$, the class labels y_k, corresponding to the class labels C_j , $j = 1, 2, \ldots, q$, are governed by a categorical r.v. Y. The set $\{y_j^1, y_j^2, \ldots y_j^N\}$ represents, from a statistical point of view, a random sample of length N corresponding to the categorical r.v. Y.

A direct and simple way to discover potential information embedded in data is to highlight the statistical dependence between the parent r.v.'s X_i, $i = 1, 2, \ldots, p$, of attributes and the parent r.v. Y of the decision class. To this end, taking into account a common case in real-world applications assuming a non-linear monotonic relationships between variables and the existence of many tied observations in data, we have chosen the non-parametric Goodman-Kruskal Gamma rank correlation Γ, although there are other alternative options (e.g., Spearman rank ρ, Kendall *Tau*, etc.). The rank correlation Γ is based on the difference between concordant pairs (C) and discordant pairs (D), and computed as $\Gamma = (C-D)/(C + D)$.

2.3. Adaptive SLFN algorithm (aSLFN)

The "adaptive" attribute refers to two distinct aspects. Firstly, it is the way to initiate the hidden nodes. They are initiated based on the natural connection between attributes and classes, that is the "adaptation" of the algorithms to existing data, and not independently of data, as in the traditional ELM case. Secondly, the algorithm "adapts" further to existing data by taking into account the level of influence of each attribute on the class through the embedded filtering module.

2.3.1. Adaptive initiation of hidden nodes

Let $\widetilde{N}$ be the number of hidden nodes of the network. As it is natural for the network to adapt to the problem at hand, it has a flexible structure, tailored according to the dataset used, so the number of hidden nodes changes on a case-by-case basis. Denote by w_{ij}, $i = 1, 2, \ldots, p$, $h = 1, 2, \ldots, \widetilde{N}$, the synaptic weight connecting the input attribute