Annals of the University of Craiova, Mathematics and Computer Science Series
Volume 46(1), 2019, Pages 189–202
ISSN: 1223-6934

A Bayesian framework for extreme learning machine with application for automated cancer detection

Smaranda Belciug and Renato Constantin Ivanescu

Abstract. Fairly recently, extreme learning machine (ELM) has been proposed as a single-hidden layer feedforward neural network (SLFN), where the input weights are randomly initiated and never updated, and the output weights are analytically computed. Setting the parameters of the hidden layer randomly may not be always effective if the function that is learned is not simple and the amount of labeled data is not small, even if theoretical studies have shown that ELM maintains the universal approximation capability. To address this issue, we propose a new approach inspired by the Bayesian paradigm as an alternative to the random initiation of the hidden node parameters. The idea behind this model is that we can use the information (prior knowledge) about a certain labeled data through the correlation between attributes and decision classes. The prior knowledge is acquired through the Goodman−Kruskal Gamma rank correlation between attributes and labels, assuming that the input weights should be related to the influence of attributes upon labels. Five publicly available high-dimensional datasets regarding cancer (breast, lung, colon, and ovarian) related to cDNA arrays, DNA microarray, and mass spectroscopy are used for experimentation and model assessment. We compared the performance of this classifier with that of three 'neighboring' algorithms, such as a basic ELM, a SLFN trained by backpropagation (BP) algorithm, and a radial basis function network (RBF). The experimental results undoubtedly indicated that the proposed variant of ELM is very effective and its performance is superior to that of the comparison models.

2010 Mathematics Subject Classification. Primary 60J05; Secondary 60J20.
Key words and phrases. Extreme learning machine, Bayesian decision rule, prior probabilities, hidden node initialization, automated cancer detection.

1. Introduction

Particular cases of multi-layer perceptions (MLPs) using randomly initialized input weights along with adaptable output weights have been proposed in literature, [5], [10], [40], [37], [26] in order to substantially decrease the computational time spent in adjusting the weights as occurs in the classical case.

Fairly recently, extreme learning machine (ELM) was proposed as a new learning algorithm for single-hidden layer feedforward neural networks (SLFNs) [22], [23], [24]. The learning paradigm is based on the random choice of the hidden nodes and the analytical calculation of the output weights. This approach has given rise to some debate in the Machine Learning (ML) community, mainly due to its resemblance to SFLNs with this particular learning technique [44]. In spite of this dispute, ELM has been widely used in different domains because of its better generalization ability, robustness, and fast learning speed [11], [43], [15], [25]. Recent years have seen a strong development of ELM, various versions being proposed along with applications

in a wide range of fields. State-of-the-art ELM models consist of evolutionary cost-sensitive ELM, parsimonious kernel ELM, self-adaptive ELM, robust ELM, hybrid Bat algorithm-ELM, hierarchical ELM, kernel ELM, sequential ELM, regularized ELM, fuzzy ELM, incremental-based ELM, etc. [51], [52], [44], [36], [33], [39], [50], [46], [14], [49], [7].

Because ELM requires a much shorter training time and less manual intervention, it received increased attention, becoming very popular, particularly in the analysis of a large amount of data. ELM has been successfully applied to solve a lot of real-world applications, such as traffic flow prediction [41], image classification [12], dimension reduction in image processing [30], stochastic sensitivity analysis [4], real-time tidal prediction [50], prediction of demographic attributes [32], gene expression data classification [46], etc.

For several years, ML changes the way health care professionals do their jobs, by improving diagnostics and treatments, beginning to develop personalized care, and optimizing patient management, [6]. In this context, ELM is extensively used nowadays in various medical applications, such as Alzheimer's disease detection [3], lung cancer diagnosis [14], [18], [19], [20], [21] breast cancer diagnosis and analysis [35], [9], heart disease diagnosis [28], diabetes [16], nuclear magnetic resonance imaging [44], [50], pathological brain detection [33], electromyography [2], liver fibrosis [8].

The research on Bayesian methods used to enhance the ELM performance is a current concern nowadays. Some of the most relevant studies of this kind focus on different approaches, such as: Bayesian linear regression to optimize the weights of the output layer(Bayesian ELM) [42], learning the output weights of ELM by estimating the marginal likelihood of network outputs through a sparse Bayesian approach [34], using the Bayesian posterior probability as activation function for the hidden neurons in a constrained-optimization-based ELM [48], the determination of network target vectors exploiting both training data labeling information and training data geometric information in aELM-based unsupervised subspace learning [27], the use of the Bayesian prior distribution and the variational approximation inference to compute the posterior distribution and the independent variational hyper-parameters for selecting the hidden nodes automatically [13].

This paper proposes and evaluates a novel approach, inspired by the Bayesian paradigm,to set the parameters of the ELM hidden layer. The method has been conceived as a more efficient alternative to the traditional random initialization,which is independent from applications, thus representing a major topic of debate. To the best of our knowledge, our method is completely different from the researches regarding the use of the Bayes model to ELM.

The Bayes decision rule combines both the priors and the likelihoods to achieve the minimum probability of error. In the Bayesian classification, the predicted class maximizes the posterior probability, i.e., the conditional probability of attributes, given a class label. There are often situations where we have to answer the question: *'What is a reasonable decision rule if the only available information is the prior?'*. Different from other approaches dealing with the Bayesian paradigm, the current study proposes a way of making a reasonable decision by using the only information available in datasets,that is the relation between attributes and class labels, statistically quantified by the corresponding correlation. Assuming a nonlinear monotonic relationship between variables, and the existence of many tied observations in data, which is a

frequently encountered situation in real-world applications, we considered the non-parametric Goodman-Kruskal Gamma rank correlation between attributes and class labels as prior knowledge. Instead of randomly generating the hidden nodes, they are now problem-dependent and estimated by a rank correlation between attributes and labels. In the benchmarking process, the statistical comparison indicated that ELM enhanced with this novel initialization technique of the hidden nodes outperforms the nearest conventional techniques(traditional ELM, RBF, and SLFN trained by BP algorithm (BP-SLFN with one hidden layer)) regarding both the decision accuracy and the computation speed. The contributions of the paper are twofold: mainly, to develop a novel initialization technique for ELM based on Bayesian paradigm, and secondary, to assess its effectiveness using real-world high-dimensional cancer databases.

The remainder of this paper is organized in six sections. Section 2 is devoted to a brief presentation of the Bayesian paradigm. Section 3 presents the design and implementation of the novel model. Section 4 presents the benchmarking datasets and briefly summarizes the statistical framework for performance assessment. Section 5 presents the experimental results, model assessment, and corresponding discussions. Section 6 deals with the conclusions and future work.

2. THE BAYESIAN PARADIGM: PRIOR KNOWLEDGE

A. The Bayesian model

In the Bayesian framework, the (observable) data D are assumed to be generated by hidden reasons r. The connection between data and the corresponding causes is given through the conditional probability P($D|r$)-*likelihood*, the probability P(D)-*evidence*, and the prior knowledge P(r)-*prior probability*. The Bayesian model consists in computing the conditional probability P($r|D$)-*posterior probability*, using the Bayes' theorem:

$$P(r|D) = \frac{P(D|r) \cdot P(r)}{P(D)} \tag{1}$$

Using the Bayesian terminology, equation (1) can be written as

$$posterior = \frac{likelihood \cdot prior}{evidence}.$$

Based on the Bayesian framework, a decision-making process combines prior knowledge with information extracted from observations. Formally, the posterior probability $P(h|D)$ is computed given a hypothesis h, the data D, the likelihood $P(D|h)$, the prior probability $P(h)$, and the evidence $P(D)$. From a probabilistic point of view, the Bayes' formula is given by:

$$P\{A_i|B\} = \frac{P\{B|A_i\}\, P\{A_i\}}{\sum_{i=1}^{n} P\{B|A_i\}\, P\{A_i\}} \tag{2}$$
$$P\{B\} > 0, P\{A_i\} > 0, i = 1, 2, ..., n,$$

where B is an arbitrary event and $\{A_1, A_2, ..., A_n\}$ is a partition of the sample space Ω.

In a decision-making/classification problem, an object with attributes $\{A_1, A_2, ..., A_n\}$ has to be assigned to a certain class C. Using the Bayesian model, the class $C = C_k$ that maximizes the posterior probability $P\{A_1, A_2, ..., A_n \mid C_j\}$ is chosen accordingly.

In addition, one often considers the so-called *naïve Bayes assumption (Idiot's Bayes),* stating the independence of attributes (obviously, a false assumption most of the time) for a given class C, namely $P\{A_1, A_2, ..., A_n \mid C\} = P\{A_1 \mid C\} \cdot P\{A_2 \mid C\} \cdot P\{A_n \mid C\}$.

B. Knowledge embedded in data

A dataset used in supervised learning contains objects characterized by inputs (features) and outputs (classes/categories). The intrinsic connection between features and the corresponding class labels provides valuable information, worth to be considered in the decision-making process. Considering an objective Bayesian point of view [29], the knowledge embedded in this connection, based upon available data, should lead to the same decision, regardless of the way of usage. From a statistical point of view, the prior knowledge may be measured by the prior probability $P(h)$, expressing the (objective) information about a certain object through the liaison between its attributes and decision class.

Consider that the (training) dataset contains N objects $x_1, x_2, ..., x_N$. Each object in the dataset is coded as a vector $x_k = (x_1^k, ..., x_i^k, ..., x_p^k; y_k)$, where $x_i^k, i = 1, 2, ..., p$, represents the $i - th$ feature of the object $x_k, k = 1, 2, ..., N$, and y_k represents the label corresponding to object x_k, that is the decision class $C_j, j = 1, 2, ..., q$. From a probabilistic point of view, assume that, for each $k = 1, 2, ..., N$, the attribute values x_i^k belonging to the attribute $A_i, i = 1, 2, ..., p$, are governed by a random variable (r.v.) X_i, in other words, one identifies the attribute A_i with the corresponding r.v. X_i. Statistically, the set $\{x_i^1, x_i^2, ..., x_i^n\}$ represents a random sample of length N corresponding to the r.v. X_i. For the sake of simplicity and 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 , $i = 1, 2, ..., p$ are independent.

Next, assume that, for each object x_k, the class labels y_j , $j = 1, 2, ..., q$, are governed by a categorical r.v. Y. Statistically, the set $\{y_j^1, y_j^2, ..., y_j^N\}$ represents a random sample of length N corresponding to the categorical r.v. Y.

A straightforward method to discover potential information within data is to assess the statistical dependence between the parent r.v.'s X_i , $i = 1, 2, ..., p$, of attributes and the parent r.v. Y of the decision class. Assuming a common situation in real-world applications, that is a non-linear monotonic relationships between variables and the existence of many tied observations in data, we considered the non-parametric Goodman-Kruskal Gamma rank correlation Γ, which is based on the difference between concordant pairs (C) and discordant pairs (D), and computed as $\Gamma = (C - D)/(C + D)$, although there are other alternative options (e.g., Spearman rank ρ, Kendall Tau, etc.).

3. BAYESIAN INITIALIZATION OF ELM

The proposed alternative to the basic random initialization of the connections between the input layer and the hidden neurons, called Bayesian initialization of ELM (BiELM), simply assigns the rank correlation between attributes and decision classes to the input weights. This controlled setting of the parameters of the hidden layer directly expresses the prior knowledge embedded in the training dataset upon the hidden weights.

A. Traditional ELM algorithm

ELM is a special case of SLFN with a single layer of hidden units, where the synaptic weights connecting inputs to hidden units are randomly initiated, while the synaptic weights between hidden units and outputs are optimized using a Moore-Penrose generalized inverse [24].

For a training dataset TS, containing N objects with p attributes and q decision classes, consider a standard SLFN with $\tilde{N}$ hidden nodes and activation function $g(x)$. Denote by $w_i = (w_{i2}, w_{i2}, ..., w_{ip})$ the weight vector connecting the ith hidden node and the input nodes, by $\beta_i = (\beta_{i1}, \beta_{i2}, ..., \beta_{iq})$ the weight vector connecting the i-th hidden node and the output nodes, and by b_i the threshold of the i-th hidden node. The traditional ELM learning method for SLFN can be summarized as follows [24]:

ELM algorithm: Given the training set TS, the activation function , and a number $\tilde{N}$ of hidden nodes:
Step 1: Randomly assign input weight w_i and bias b_i , $i = 1, 2, ...,\tilde{N}$.
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 Moon-Penrose generalized inverse of H, and T is the output vector.

B. Bayesian initialization of the hidden nodes. The BiELM algorithm

For the sake of simplicity, consider $\tilde{N}$ = q. Denote by w_{ij}, $i = 1, 2, ..., p$, $j = 1, 2, ..., q$, the synaptic weight connecting the input attribute x_i belonging to the feature vector x to the jth hidden neuron of the network. From a probabilistic point of view, assume that the real values of the weights w_{ij} represent the values of a r.v. denoted by $W_{ij}, i = 1, 2, ..., p, j = 1, 2, ..., q$. A synaptic weight refers to the strength of a connection between two units, perceived as a measure of this strength. Assuming that the weights w_{ij} belong to the interval [0, 1], they might be interpreted as a probability-like measure encoding the strength of the connection between attributes and class labels. The basic idea of our Bayesian approach is to transfer this measure of the connection strength from the couple formed by attribute and class label to the couple formed by input and hidden neuron. In this way, one replaces an arbitrary random initialization of the hidden nodes by a rational initialization, directly related to the purpose of the network, that is to identify the optimal connection between attributes and decision classes. Under these circumstances, the modulus of Γ might be interpreted as a probability-like measure encoding the strength of the relationship between attributes and class labels enhancing the connection between input and hidden nodes.

Assume that the weights w_{ij} are naturally related to the attributes influence on the decision class, and suppose that the events A_{ij} corresponding to the r.v. W_{ij} provide a partition of the "weight space" W. Then, the probabilities $P\{A_{ij}\}$ may be considered as priors in the Bayesian context, expressing specific information about the object $\mathbf{x}$ through the correlation between attributes X_i and decision Y. In Bayesian statistical inference, these (informative) priors reflect the prior knowledge of how likely attributes influence decision classes before the classification is taken into account. $P\{A_{ij}\}$ are expressed in probabilistic terms by the rank correlation Γ between X_i and Y by:

$$P\{A_{ij}\} = \Gamma(X_i, Y), i = 1, 2, ..., p, j = 1, 2, ..., q. \tag{3}$$

For each hidden neuron, consider the non-linear activation function given by the hyperbolic tangent $f(u) = 1.7159 \cdot tanh(2u/3)$, recommended by its fast convergence [31]. Given a decision class $C_j, j = 1, 2, ..., q$, the corresponding label y_j is encoded using the *"1-of-q"* rule for nominal/categorical data, i.e., $y_1 \sim (0, 0, ..., 1), y_2 \sim (0, 0, ..., 1, 0), ..., y_q \sim (1, 0, ..., 0)$. For each decision class $C_j, j = 1, 2, ..., q$, and for each attribute $A_i, i = 1, 2, ..., p$, one computes the corresponding mean attribute value m_i^j, numerically encoding the knowledge regarding the connection between attributes and decision classes. Then for the jth hidden unit, $j = 1, 2, ..., q$, one computes the synaptic weights w_{ij}, given by:

$$W_{ij} = \Gamma((x_i^k - m_i^j), y_k),\ i = 1, 2, ..., p,\ j = 1, 2, ..., q,\ k = 1, 2, ..., N. \tag{4}$$

BiELM algorithm: Given the training set TS, the hyperbolic tangent as activation function, and a number $\tilde{N}$=q of hidden nodes:
Step 1: For each decision class $C_j, j = 1, 2, ..., q$, and for each attribute $A_i, i = 1, 2, ..., p$, compute the corresponding mean attribute value m_i^j.
Step 2: For each synaptic weight $w_{ij}, i = 1, 2, ..., p, j = 1, 2, ..., q$, assign the Goodman-Kruskal Gamma rank correlation between attributes and decision classes, given by formula (4).
Step 3: The network output is given by $y_k = \sum \beta_k \cdot f(x_k, w_i k), k = 1, 2, ..., N, i = 1, 2, ...p$, where x_k is a sample vector, and f is the hyperbolic tangent.
Step 4: Using TS, Calculate the hidden layer output matrix H.
Step 5: Calculate the output weight $\beta = H^+Y$, where H is the hidden layer output matrix, H^+ is the Moon-Penrose generalized inverse of H, and Y is the output vector.
A Python implementation of BiELM and ELM has been performed on a 2.80 GHz Intel(R) Core(TM)2 Extreme CPU X9000, 4.00GB (RAM).

4. BENCHMARKING DATASETS. STATISTICAL ASSESSMENT

A.Benchmarking datasets

The benchmarking datasets used for experimentation and model assessment originate from the publically available Machine Learning Data Set Repository `http://mldata.org/`. The first four refer to the DNA micro-array technology, while the last one relates to proteomic spectra obtained by mass spectroscopy. All datasets are characterized by high dimensionality, ranging from 2000 to 24481 attributes, in order to highlight the huge computation power, speed and effectiveness of BiELM algorithm regardless the dataset dimension. It is noteworthy that, although the balanced datasets are desirable for classification, we considered the original (unbalanced) datasets in order to test the algorithm in real-time conditions.
1) Breast cancer Kent Ridge (BCKR) available at
`http://mldata.org/repository/data/viewslug/breast-cancer-kent-ridge-2/`.
DNA microarray analysis was used to identify a gene expression signature strongly predictive of a short interval to distant metastases for patients with primary breast tumors [Veer]. The dataset contains 97 instances with 24481 attributes and two-class decision (46-relapse vs. 51-non-relapse).
2) Colon cancer Kent Ridge(CCKR) available at
`http://mldata.org/repository/data/viewslug/colon-cancer-kent-ridge/`. The

oligonucleotide arrays (or cDNA arrays), allowing the parallel monitoring of expression level of thousands of genes, was used to discriminate between tumor biopsies and normal biopsies of colons of the same patients [1]. The dataset contains 62 instances with 2000 attributes and two-class decision (40-negative vs. 22-positive).
3) Breast cancer Duke (BCD) available at
`http://mldata.org/repository/data/viewslug/duke-breast-cancer/`.Gene expression data derived from DNA microarray analysishave the capacity to discriminate breast tumors on the basis of estrogen receptor (ER) status and also on the categorized lymph node status. The dataset contains 86 instances with 7129 attributes and two-class decision regarding the estrogen receptor (45-(ER+) vs. 41-(ER-)).
4) Lung cancer Michigan(LCM) available at
`http://mldata.org/repository/data/viewslug/lung-cancer-michigan/`. Gene expression profiles based on microarray analysis can be used to predict patient survival in early-stage lung adenocarcinomas. The dataset contains 96 instances with 7129 attributes and two-class decision (86-tumor vs. 10-normal).
5) Ovarian cancer (NCI PBSII)(OC) available at
`http://mldata.org/repository/data/viewslug/ovarian-cancer-nci-pbsii-data/`.
The proteomic spectra were generated by mass spectroscopy and used to identify proteomic patterns in serum that distinguish ovarian cancer from non-cancer [38]. The dataset contains 253 instances with 15154 continuous attributes and two-class decision (162-cancer vs. 91-normal).

B. Statistical assessment

Since the sample size is relatively small in all cases, the 10-fold cross-validation (10-fold CV) has been chosen to assess the predictive accuracy. To assess the BiELM performance and to compare it to those of ELM, RBF and BP-SLFN, each competing algorithm has been independently run 100 times in 10-fold CV cycle, representing a minimum sample size ensuring the validity of the envisaged statistical tools.

The average classification accuracy (ACC) along with the corresponding standard deviation (SD) obtained in the testing phase have been chosen as the main indicators of the classification accuracy and algorithm's stability. The area under the ROC curve (AUC), simultaneously combining the sensitivity and specificity of the classifier, has been also considered as one of the best ways to evaluate or compare by a unique value the classifiers' performance. The corresponding AUCs related to example runs of each model with similar classification accuracy to the average one have been presented to serve the reader a more comprehensive overview.The traditional evaluation system for AUC (i.e., 0.9-1.0 excellent; 0.8-0.9 good; 0.7-0.8 fair; 0.6-0.7 poor; 0.5-0.6 failure) has been used to assess the performance of each classifier [17].

Inspired by the (Lyapunov) stability of trajectories of dynamical systems, one can consider that the ACC values corresponding to a certain number of independent computer runs represent an 'orbit' related to the 'dynamics' of the stochastic algorithm (transition from one computer run to another). Accordingly, a stochastic algorithm is considered 'stable' if the ACC values stay in a small enough neighborhood of the 'point of equilibrium' irrespective of the small perturbations due to its stochastic nature (randomly assignation/initialization, etc.). The neighborhood might be considered the 95% confidence interval (95% CI) for the average accuracy obtained during a certain number of independent computer runs ('point of equilibrium'), computed

using SD. Consequently, narrow CI indicates evidence of a more stable model regardless of the number of computer runs.It is noteworthy that the distribution of ACC values would be approximately Gaussian for large enough samples (e.g., greater than or equal to 100) represented by independent computer runs showing consistency of the CI.

Besides the classification accuracy, which is the most important performance measure especially in computer-aided medical diagnosis, we also need algorithms that run quickly and use the available computing resources efficiently. NNs, generally, despite their key role in classification, face a challenging issue regarding the learning speed especially in case of very large data dimension. In this regard, we have compared the CPU time (in seconds) for each model during a complete 10-fold CV cycle for 100 independent runs. This tough enough benchmarking condition has been chosen to better highlight the very large difference regarding computation speed between the two types of ELM algorithms and the traditional NNs.

The classical one-way ANOVA technique along with the Tukey's honestly significant difference (Tukey HSD) *post-hoc* test have been used to statistically quantify the magnitude of the contrast between the corresponding classification performances.

5. RESULTS AND DISCUSSIONS

The experiments on five high-dimensional medical datasets regarding major types of cancer aimed to assess the BiELM against some traditional approaches. In this respect, a direct comparison of BiELM with results obtained by three closest algorithms, namely traditional ELM, BP-SLFN, and RBF on the same datasets has been performed.

A. Experimental results

The classification performance indicators, in terms of ACC, SD, and AUC are displayed in Table 1.

Dataset	ACC/SD(%)	AUC
BCKR	53.33/8.58	0.612
CCKR	71.27/12.34	0.724
BCD	72.63/4.69	0.732
LCM	92.58/4.21	0.995
OC	80.15/7.21	0.857

TABLE 1. Experimental results: ACC/SD and AUC for BiELM.

We observe that ACC as well as SD strongly depend on the dataset used. For the dataset (BCKR) having the largest dimensionality (24481), ACC as well as AUC are very unsatisfactory from a medical point of view (53.58% and 0.612), indicating a poor classification performance. However, it seems that the classification performance is not correlated with the data dimension. Thus, although BCD and LCM have the same dimension (7129), the performance difference is significant, that is 72.63% vs. 92.58% and 0.732 vs. 0.995, respectively, ranging from fair to excellent. Moreover, although OC has 15154 attributes as against CCKR with only 2000, ACC = 80.15%

(AUC = 0.857) vs. ACC = 71.27% (AUC = 0.724). To conclude, BiELM is not directly influenced by data dimension but by dataset itself. It is noteworthy that the same conclusions can be drawn regarding the algorithm's stability indicated by SD(see, for instance, the case of BCKR and CCKR).

B. Evaluation of BiELM performance

The simulations regarding RBF and BP-SLFN have been carried out in STATISTICA 7 (StatSoft, Inc.) environment, run on a 2.80 GHz Intel(R) Core(TM)2 Extreme CPU X9000, 4.00GB (RAM). The experimental results displayed in Table 2 refer to ACC (%) and CPU time (seconds) obtained by each competing model during 100 independent computer runs (complete 10-fold CVcycle).

Dataset	BiELM	ELM	RBF	BP-SLFN
	ACC(%)	ACC(%)	ACC(%)	ACC(%)
	CPU(sec)	CPU(sec)	CPU(sec)	CPU(sec)
BCKR	53.33	42.16	46.75	50.81
	1751	761	4500	43020
CCKR	71.27	58.53	69.00	68.40
	14	7	27	374
BCD	72.63	67.67	68.09	51.38
	151	63	367	3567
LCM	92.58	87.10	80.95	89.91
	173	79	305	4440
OC	80.15	75.18	65.42	75.20
	1415	643	2296	363600

TABEL 2. Comparing models performance (ACC, CPU TIME).

From Table 2 one can notice, as before, that the performance of all classifiers,in terms of ACC, strongly depends on the specific dataset, thus confirming the traditional findings.It is important to note that, regardless of the dataset, BiELM has the best classification performance. Regarding the competitors, we notice that the hierarchy changes depending on the datasets. Thus, in three out of five datasets (BCKR, LCM, OC), BP-SLFN is the most performant algorithm, pointing out that it is about the datasets with the largest dimension. ELM outperforms RBF in just two cases (LCM, OC), and BP-SLFN in just one case (BCD). It follows that there is need of improving the classical ELM. In conjunction with Table 2, Fig. 1 illustrates the performance comparison regarding the stability to several computer runs, expressed by the 95% CI computed via SD. Excepting BCKR, with the poorer stability of BiELM, i.e., SD = 8.58% (BiELM) vs. 4.60% (ELM), 5.88% (RBF), and 5.50% (BP-SLFN), and, partially, CCKR with SD = 12.34% (BiELM) vs. 11.45% (ELM), 8.96% (RBF), and 26.99% (BP-SLFN), in the other cases BiELM was relatively more stable than its competitors. Thus, on BCD we obtained SD = 4.69% (BiELM) vs. 8.95% (ELM), 11.36% (RBF), and 19.76% (BP-SLFN); on LCM, SD = 4.21% (BiELM) vs. 4.71% (ELM), 11.52% (RBF), and 9.70% (BP-SLFN); and OC, SD = 7.21% (BiELM) vs. 9.63% (ELM), 6.20% (RBF), and 9.85% (BP-SLFN)).

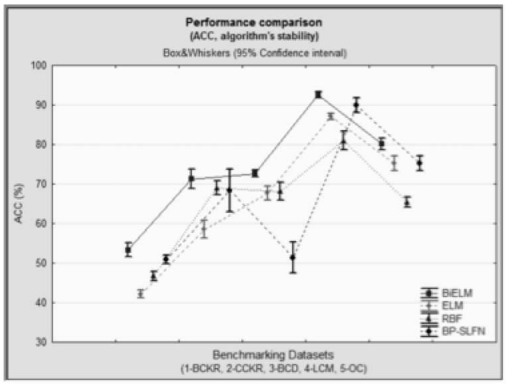

FIGURE 1. Performance comparison (Box&Whiskers plot).

The CPU time analysis explicitly highlighted the advantage of using ELM-like algorithms, especially for huge dimensions. The fastest algorithm, regardless the dataset, is ELM, closely followed by BiELM and, at considerable distance, by RBF. Regarding BP-SLFN, it was by far the slowest algorithm, despite its relatively high classification accuracy. These findings are fully justified if we take into consideration their learning paradigms [17]. The BP algorithm is relatively time consuming because it uses two distinct phases of computation. In the forward pass, the synaptic weights remain unaltered and the network function signals are computed on a neuron-by-neuron basis. In the backward pass, the error signals are sent back layer-by-layer through the network and the local gradient for each neuron is computed. Faster than the BP algorithm, training RBF consists in determining the parameters of the (Gaussian) basis functions, followed by the computation of the synaptic weights. While in the traditional ELM, only the output weights are learned in a single step, BiELM improves the learning process by controlling the initialization of the input weights, losing thus computation speed, but gaining accuracy.

The performance comparison displayed in Table 2 revealed primary information about the behavior of each algorithm involved in the benchmarking process. Deepening the comparison, the one-way ANOVA technique along with the Tukey's honestly significant difference(Tukey HSD) post-hoc test have been used to highlight statistically significant differences regarding ACCs between BiELM and the competitors. The one-way ANOVA analysis is used to determine whether there is a significant difference between the means of two or more independent groups. In our case a difference between the BiELM and all the other competitors for each dataset.The ANOVA output, consisting of (combined) sums of squares (SS), degrees of freedom (df), mean squares (MS), F-value, and p-level (contrasts: quadratic polynomial), is presented in Table 3.

Dataset	SS	df	MS	F-value	p-level
BCKR	7167.58	3	2389.19	59.88	0.00
CCKR	9568.90	3	3189.63	11.68	0.00
BCD	24823.72	3	8274.57	52.22	0.00
LCM	7708.53	3	2569.51	40.20	0.00
OC	11237.32	3	3745.77	53.13	0.00

TABEL 3. One-Way ANOVA test BiELM vs. competitiors on each dataset.

The post-hoc Tukey HSD test has revealed the following statistically significant differences in classification performance (p-level < 0.05):
- On BCKR dataset, BiELM vs. ELM (mean diff. = 12.73, std. err. = 2.34), BiELM vs. RBF (mean diff. = 6.58, std. err. = 0.89), BiELM vs. MLP (mean diff. = 2.52, std. err. = 0.89);
- On CCKR dataset, BiELM vs. ELM (mean diff. = 11.17, std. err. = 0.89);
- On BCD dataset BiELM vs. ELM (mean diff. = 4.87, std. err. = 1.78), BiELM vs. RBF (mean diff. = 4.73, std. err. = 1.78), BiELM vs. MLP (mean diff. = 20.78, std. err. = 1.78);
- On LCM dataset, BiELM vs. ELM (mean diff. = 5.53, std. err. = 1.13), BiELM vs. RBF (mean diff. = 11.84, std. err. = 1.13);
- On OC dataset, BiELM vs. ELM (mean diff. = 4.98, std. err. = 1.19), BiELM vs. RBF (mean diff. = 14.64, std. err. = 1.19), BiELM vs. MLP (mean diff. = 4.92, std. err. = 1.19).

The one-way ANOVA test along with the Tukey's post-hoc test provided the objective analytical confirmation regarding the statistically difference concerning ACC between the competitor algorithms presented in Table 2. More importantly though is the fact that, regardless the benchmarking dataset, BiELM had a better classification performance,statistically proven, than the traditional model.

C. Discussion

It can be noticed that the idea of replacing the randomness specific to the traditional ELM by the embedded knowledge in data, quantified by the correlation between attributes and class labels, has proven fruitful. The benchmarking results from Table 2 in terms of ACC, along with the ANOVA/$post-hoc$ Tukey HSD comparison tests from Table 3, undoubtedly showed a significant gain in accuracy. In addition, this novel approach brought a gain in terms of algorithm stability quantified by the 95% CI as illustrated in Fig. 1.

As regards the computational speed, although ELM remains the fastest algorithm, BiELM proved surprisingly fast despite the relatively large volume of computations needed for initialization. The significant gain in decision-making accuracy, of particular interest in medical diagnosis, is fully offset by an about two times lower computing speed. As expected, BiELM took advantage both of the well-known parent ELM algorithm's speed, and the gain in classification accuracy obtained by using the new initialization approach, based on extracting knowledge from data.

6. CONCLUSIONS

The prior knowledge embedded in the connections between attributes and class labels can successfully assist a traditional ELM algorithm in learning better samples in a dataset. This paper explores the potential to enhance the classification performance of a traditional ELM by replacing the simple random initiation of the hidden node parameters by the Goodman-Kruskal Gamma rank correlation between attributes and class labels. Based on the Bayesian model, by directly using the informative priors, which measure the influence of attributes upon labels through the corresponding rank correlation, BiELM uses the potential prior knowledge provided by the training dataset to initialize the connections between the input layer and the hidden neurons. The current algorithm offers the possibility of dealing with very large datasets due to its ELM inheritance, combined with an enhanced classification capability due to the novel initialization of the input weights.

The proposed approach can be improved in attempting to enhance the computation speed. A task for future work concerns the design of a filtering mechanism of the rank correlations, inspired by the partially connected models of neural networks. In this way, only those that are truly significant for classification should be chosen for the initiation of the hidden node parameters.

References

[1] U. Alon, N. Barkai, D.A. Notterman, et al., Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays, in: *Proc. Annals Nat Acad Sci USA* **96** (1999), 6745–6750.

[2] K. Anam, A. Al-Jumaily, Evaluation of extreme learning machine for classification of individual and combined finger movements using electromyography on amputees and non-amputees, *Neural Networks* **85** (2017), 51–68.

[3] A. Aswatha Kumar, B.S. Mahanand, Alzheimer's Disease Detection Using Minimal Morphometric Features with an Extreme Learning Machine Classifier, *Proc. International Conference on Advances in Computing* **174**, Advances in Intelligent Systems and Computing series (2012), 753–762.

[4] D. Becerra-Alonso, M. Carbonero-Ruz, A.C. Martinez-Estudillo, et al., Stochastic sensitivity analysis using extreme learning machine, in: *Extreme Learning Machine 2013: Algorithms and Applications, Adaptation, Learning and Optimization* **16** (F. Sun, K.-A. Toh, M. Romay, K. Mao, Eds.), Springer (2014), 1–12.

[5] S. Belciug, F. Gorunescu, Error-correction learning for artificial neural networks using the Bayesian paradigm. Application to automated medical diagnosis, *Journal of Biomedical Informatics* **52** (2014), 329–337.

[6] S. Belciug, F. Gorunescu, *Intelligent Decision Support SystemsA Journey to Smarter Healthcare*, Springer Nature Switzerland AG, 2020.

[7] S. Belciug, F. Gorunescu, Learning a single-hidden layer feedforward neural network using rank correlation-based strategy with application to high dimenional gene expression and proteomic spectra datasets in cancer detection, *Journal of Biomedical Informatics* **83** (2018), 159–166.

[8] S. Belciug, M. Lupsor, R. Badea, Feature selection approach for non-invasive evaluation of liver fibrosis, *Annals of the University of Craiova, Mathematics and Computer Science Series* **35** (2018), 15–20.

[9] S. Belciug, F. Gorunescu, M. Gorunescu, A. Salem, Assessing performances of unsupervised and supervised neural networks in breast cancer detection, *The 7th International Conference on Informatics and Systems* (2010), 1–8.

[10] D.S. Broomhead, D. Lowe, Multivariable functional interpolation and adaptive networks, *Complex Syst.* **2** (1988), 321–355.

[11] E. Cambria, G.-B. Huang, L. Lekamalage, et al., Extreme Learning Machines [Trends Controversies], *IEEE Intelligent Systems* **28** (2013), no. 6, 30–59.
[12] J. Cao, K. Zhang, M. Luo, et al. Extreme learning machine and adaptive sparse representation for image classification *Neural Networks* **81 (C)** (2016), 91–102.
[13] Y. Chen, J. Yang, C. Wang, et al., Variational Bayesian extreme learning machine, *Neural Comput Applic.* **27** (2016), no. 1, 185–196.
[14] M.R. Daliri, A hybrid automatic system for the diagnosis of lung cancer based on genetic algorithm and fuzzy extreme learning machines *J Med Syst* **36** (2012), no. 2, 1001–1015.
[15] S. Ding, H. Zhao, Y. Zhang, Extreme learning machine: algorithm, theory, and applications, *Artificial Intelligence Review* **44** (2015), no. 1, 103–115.
[16] E. Georga, V.C. Protopappas, D. Polyzos, et al., Online prediction of glucose concentration in type 1 diabetes using extreme learning machines, in: *Proc. IEEE Conf Eng Med Biol Soc* (2015), 3262–3265.
[17] F. Gorunescu, Data Mining: Concepts, Models and Techniques, *Berlin Heidelberg: Springer-Verlag*, 2011.
[18] F. Gorunescu,S. Belciug, Boosting backpropagation algorithm by stimulus-sampling: Application in computer-aided medical diagnosis, *Journal of Biomedical Informatics* **63** (2016), 74–81.
[19] F. Gorunescu, S. Belciug, M. Gorunescu, R. Badea, Intelligent decision-making for liver fibrosis stadialization based on tandem feature selection and evolutionary-driven neural network, *Expert Syst Appl* **39** (2012), 12824–12833.
[20] F. Gorunescu, S. Belciug, Evolutionary strategy to develop learning-based decision systems. Application to breast cancer and liver fibrosis stadialization, *J Biomed Inform* **49** (2014), 112–118.
[21] F. Gorunescu, M. Gorunescu, E. El-Darzi, S. Gorunescu, A statistical framework for evaluationg neural networks to predict reccurent events in breast cancer, *International Journal of General Systems* **39** (2010), no. 5, 471–488.
[22] G.-B. Huang, Q.-Y. Zhu, C.-K. Siew, Extreme Learning Machine: A New Learning Scheme of Feedforward Neural Networks, in: *Proc. Intl. Joint Conf. Neur. Net.* (2006), 985–990.
[23] G.-B. Huang, L. Chen, C.-K. Siew, Approximation Using Incremental Constructive Feedforward Networks with Random Hidden Nodes, *IEEE Trans. Neural Networks* **17** (2006), no. 4, 879–892.
[24] G.-B. Huang, X. Ding, H. Zhou, Optimization Method Based Extreme Learning Machine for Classification, *Neurocomputing* **74** (2010), 155–163.
[25] G.-B. Huang, H. Zhou, X. Ding, et al., Extreme Learning Machine for Regression and Multiclass Classification,, *IEEE Trans. Systems, Man, and Cybernetics* **42** (2011), no. 2, 513–529.
[26] D. Husmeier, Random vector functional link (RVFL) networks. Neural Networks for Conditional Probability Estimation (Perspectives in Neural Computing Series), *London: Springer* (1999), 87–97.
[27] A. Iosifidis, M. Gabbouj, A Bayesian approach for extreme learning machine-based subspace learning, in: *Proc. 23rd IEEE European Signal Processing Conference (EUSIPCO)* (2015), 2401–2405.
[28] S. Ismaeel, A. Miri, D. Chourishi, Using the Extreme Learning Machine (ELM) technique for heart disease diagnosis, in: *Proc. IEEE Canada International Humanitarian Technology Conference (IHTC2015)* (2015), 1–3.
[29] S. James Press, *Subjective and Objective Bayesian Statistics. Principles, Models, and Applications* (2nd ed.) Wiley, 2010, [Online]. Available: (http://onlinelibrary.wiley.com/doi/10.1002/9780470317105.fmatter/pdf.
[30] L.L. Kasun, Y. Yang, G.B. Huang, et al., Dimension Reduction With Extreme Learning Machine, *IEEE Trans Image Process* **25** (8) (2016), 3906–3918.
[31] Y. LeCun, L. Bottou, G. Orr, K.-L. Muller Efficient BackProp. Neural Networks: Tricks of the Trade, *Lecture Notes in Computer Science (Springer)* **7700** (2012), 9–48.
[32] Y. Liu, T. Ye, G. Liu, et al., Demographic attributes prediction using extreme learning machine, in: *Extreme Learning Machine 2013: Algorithms and Applications, Adaptation, Learning and Optimization* **16** (F. Sun, K.-A. Toh, M. Romay, K. Mao, Eds.) Springer (2014), 145–165.
[33] S. Lu, X. Qiu, J. Jianpin Shi, et al., A Pathological Brain Detection System based on Extreme Learning Machine Optimized by Bat Algorithm, *CNS. Neurol Disord Drug Targets* **15** (2016).

[34] J. Luo, C.M. Vong, P.K. Wong, Sparse Bayesian extreme learning machine for multi-classification, *IEEE Trans Neural Netw Learn Syst* **25** (2014), no. 4, 836–43.
[35] A. Malik, J. Iqbal, Extreme learning machine based approach for diagnosis and analysis of breast cancer *J. Chinese Inst. Eng.* **39** (1) (2015), 74–78.
[36] D. Mesquita, A. Araujo Neto, J. Queiroz Neto, et al., Using Robust Extreme Learning Machines to Predict Cotton Yarn Strength and Hairiness, in: *Proc. European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning* (2016), 65–70.
[37] Y.-O. Pao, G.-H. Park, D. Sobajic, Learning and generalization characteristics of the random vector Functional-link net, *Neurocomputing* **6** (1994), 163–180.
[38] E. Petricoin, A. Ardekani, B. Hitt, et al., Use of proteomic patterns in serum to identify ovarian cancer, *Lancet* **359** (2002), 572–577.
[39] M.N. Qureshi, B. Beomjun Min, H.J. Jo, et al., Multiclass Classification for the Differential Diagnosis on the ADHD Subtypes Using Recursive Feature Elimination and Hierarchical Extreme Learning Machine: Structural MRI Study, *PLoS ONE* **11** (2016), no. 8.
[40] W. Schmidt, M. Kraaijveld, R. Duin, Feed forward neural networks with random weights, in: *Proc. 11th IEEE IAPR International Conference Pattern Recognition Methodology and Systems* (1992), 1–4.
[41] Q. Shang, C. Lin , Z. Yang, et al., A Hybrid Short-Term Traffic Flow Prediction Model Based on Singular Spectrum Analysis and Kernel Extreme Learning Machine, *PLoS ONE* **11** (2016), no. 8.
[42] E. Soria-Olivas, J. Gomez-Sanchis, J. Martin, et al., BELM: Bayesian Extreme Learning Machine, *IEEE Trans Neural Netw Learn Syst* **22** (2011), no. 3, 505–509.
[43] F. Sun, K.-A. Toh, M. Romay, K. Mao, *Extreme Learning Machine 2013: Algorithms and Applications. Adaptation, Learning and Optimization,* Springer, 2014.
[44] Z. Wang, J. Xin, Z. Wang, et al., Single NMR image super-resolution based on extreme learning machine *Phys Med* **32** (2016), no. 10, 1331–1338.
[45] G.-G. Wang, M. Lu, Y.-Q. Dong, et al., Self-adaptive extreme learning machine, *Neural Computing and Applications* **27** (2016), no. 2, 291–303.
[46] S. Wei, H. Lu, Y. Lu, et al., An improved weight optimization and Cholesky decomposition based regularized extreme learning machine for gene expression data classification, in *Extreme Learning Machine 2013: Algorithms and Applications, Adaptation, Learning and Optimization* **16** (F. Sun, K.-A. Toh, M. Romay, K. Mao, Eds.) Springer (2014), 55–66.
[47] M. West, C. Blanchette, H. Dressman, et al., Predicting the clinical status of human breast cancer by using gene expression profiles , in: *Proc. Natl Acad Sci USA* **98** (2001), no. 20, 11462–11467.
[48] S.H. Wong, K.S. Yap, H.J. Yap, Constrained-optimization-based Bayesian posterior probability extreme learning machine for pattern classification, *Neural Information Processing* **1** (2014), 466–473.
[49] Y. Ye, S. Squartini, F. Piazza, Incremental-based extreme learning machine algorithms for time-variant neural networks, in: *Proc. 6th international conference on Advanced intelligent computing theories and applications: intelligent computing* (2010), 9–16.
[50] J.-C. Yin, G.-S. Li, J.-Q. Hu, A modular prediction mechanism based on sequential extreme learning machine with application to real-time tidal prediction, in *Extreme Learning Machine 2013: Algorithms and Applications, Adaptation, Learning and Optimization **16*** (F. Sun, K. -A. Toh, M. Romay, K. Mao, Eds.), Springer (2014), 35–54.
[51] L. Zhang, D. Zhang, Evolutionary Cost-Sensitive Extreme Learning Machine, *IEEE Transactions on Neural Networks and Learning Systems* **PP** (99) (2016), 1–16.
[52] Y.-P.Zhao, Parsimonious kernel extreme learning machine in primal via Cholesky factorization, *Neural Networks* **80** (C) (2016).

(Smaranda Belciug, Renato Constantin Ivanescu) Department of Computer Science,
University of Craiova, 13 A.I. Cuza Street, Craiova, 200585, Romania
E-mail address: sbelciug@inf.ucv.ro, ivanescurenato@yahoo.com