Pattern Recognition 47 (2014) 3414–3428

Contents lists available at ScienceDirect

Pattern Recognition

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

Cross-validation based weights and structure determination of Chebyshev-polynomial neural networks for pattern classification

Yunong Zhang *, Yonghua Yin, Dongsheng Guo, Xiaotian Yu, Lin Xiao

School of Information Science and Technology, Sun Yat-sen University, Guangzhou, Guangdong 510006, PR China

A R T I C L E I N F O

Article history:
Received 28 June 2013
Received in revised form
8 April 2014
Accepted 29 April 2014
Available online 14 May 2014

Keywords:
Cross validation
Chebyshev polynomial
Neural network
Pattern classification
Robustness

A B S T R A C T

This paper first proposes a new type of single-output Chebyshev-polynomial feed-forward neural network (SOCPNN) for pattern classification. A new type of multi-output Chebyshev-polynomial feed-forward neural network (MOCPNN) is then proposed based on such an SOCPNN. Compared with multi-layer perceptron, the proposed SOCPNN and MOCPNN have lower computational complexity and superior performance, substantiated by both theoretical analyses and numerical verifications. In addition, two weight-and-structure-determination (WASD) algorithms, one for the SOCPNN and another for the MOCPNN, are proposed for pattern classification. These WASD algorithms can determine the weights and structures of the proposed neural networks efficiently and automatically. Comparative experimental results based on different real-world classification datasets with and without added noise prove that the proposed SOCPNN and MOCPNN have high accuracy, and that the MOCPNN has strong robustness in pattern classification when equipped with WASD algorithms.

1. Introduction

Pattern classification is one of the most important areas of artificial intelligence [1–7]. In recent years, artificial neural networks have become powerful tools for pattern classification because of their remarkable features, such as nonlinear system modeling, self-learning, and self-adaptive capabilities [3–7]. The back-propagation (BP) neural network first proposed by Rumelhart and McClelland [8] is one of the most widely applied neural network models [7,9]. However, BP-type neural networks have inherent weaknesses such as slow convergence [10] and the existence of local minima [11]. Different from algorithmic improvements on the BP iterative-training procedure, activation function and network structure improvements are the focus of this paper, with the goal of achieving better efficacy [12–14]; one example of such structure improvements is the use of orthogonal polynomial neural networks [13,14].

Chebyshev polynomials, a sequence of orthogonal polynomials, are frequently used in various applications. Recently, different kinds of Chebyshev-polynomial-based neural networks have been developed for function approximation [12,15–17], pattern classification [18,19], and nonlinear system identification [20]. Our previous studies have found that a Chebyshev-polynomial-based neural network performs effectively in approximation, generalization, and prediction [13,14]. Therefore, Chebyshev polynomials have been chosen as the basis for activation-function construction in this paper. In light of the theories of Bernstein polynomial [21,22] and orthogonal polynomial approximation [12,15], a group of Chebyshev-polynomial-based basis functions are constructed in this paper for data approximation. A three-layer feed-forward neural network (including the input, hidden, and output layers) can approximate nonlinear continuous functions effectively [23,24]. Thus, a new type of single-output Chebyshev-polynomial feed-forward neural network (SOCPNN) that adopts a three-layer structure is proposed in this paper for pattern classification. The hidden-layer neurons of the new SOCPNN are activated by the Chebyshev-polynomial-based basis functions. The proposed SOCPNN can achieve satisfactory prediction performance in handling high-dimensional data. Based on the new SOCPNN, a new type of multi-output Chebyshev-polynomial feed-forward neural network (MOCPNN), which is a generalized form of the SOCPNN, is constructed for pattern classification. This new MOCPNN can also achieve satisfactory prediction performance. The proposed SOCPNN and MOCPNN have low computational complexity, making them alternatives for pattern classification. Weights and structures significantly influence the neural network performances of a feed-forward neural network. Thus, one important issue is designing effective and efficient algorithms to determine the appropriate weights and structures for the SOCPNN and MOCPNN such that their superior characteristics can be fully used in pattern classification.

Two types of learning algorithms are frequently used for weights learning, namely, the BP (or termed, gradient-based) type and the gradient-free type. However, the BP-type algorithm

* Corresponding author. Tel.: +86 13060687155; fax: +86 20 39943353.
E-mail address: zhynong@mail.sysu.edu.cn (Y. Zhang).

http://dx.doi.org/10.1016/j.patcog.2014.04.026
0031-3203/

has inherent weaknesses, such as slow convergence and local minima, as mentioned previously. In addition, the efficacy of the BP-type algorithm is significantly related to the initial values of the parameters. By contrast, the gradient-free algorithm based on pseudoinverse, known as the least square (LS) method, the weights-direct-determination (WDD) method [12–14], or the extreme learning machine (ELM) [25], determines the optimal weights for the neural network efficiently with satisfactory or even superior performance. The WDD method has been proposed for neural networks activated by linearly independent functions, particularly for orthogonal polynomial neural networks, where the weights that link the input and hidden layers (i.e., input weights) and the hidden-layer biases can be fixed [12–14]. Meanwhile, the ELM has been proposed for single-hidden layer feedforward neural networks; for example, the sigmoidal-function neural networks, where the input weights and hidden-layer biases are chosen randomly [25]. Thus, the WDD method presented in [12–14] has been chosen to obtain the optimal weights between the hidden-layer and output-layer neurons of the SOCPNN and MOCPNN so that the weight-learning process can be more efficient, stable, and automatic.

As pointed out in [3], the performances of approximation and generalization are probably the most significant features of neural networks for pattern classification. Studies have shown that the performances of neural networks on approximation and generalization are considerably affected by the number of hidden-layer neurons [26–29]. On the one hand, neural networks with extremely few hidden-layer neurons may not be able to adjust themselves to the underlying complex model of real-world data. On the other hand, too many hidden-layer neurons may lead to an overfitting problem, in which these neural networks can perform excellently on approximation but perform poorly on generalization. Based on a number of numerical investigations (some of which are presented in Section 4.3), we found that the aforementioned phenomena exist in the proposed SOCPNN and MOCPNN. That is, the performances of two such neural networks are significantly affected by the number of their hidden-layer neurons. Therefore, developing an effective method is essential to determine the structures of the SOCPNN and MOCPNN with appropriate numbers of hidden-layer neurons. Recent studies have shown that the multi-fold cross-validation (MFCV) method is effective for model selection [31–33]; for example, using the MFCV method to select appropriate parameter values for radial-basis-function neural networks [32,33]. In light of the basic idea of the MFCV method [30–33], the four-fold cross-validation (4FCV) method is exploited specifically to determine the appropriate numbers of hidden-layer neurons for the SOCPNN and MOCPNN.

Based on the WDD and 4FCV methods, two weight-and-structure-determination (WASD) algorithms, one for the SOCPNN and another for the MOCPNN, are proposed. These WASD algorithms can determine the weights and structures of the neural networks efficiently and automatically. Moreover, comparative experiment results based on different real-world classification datasets with and without noise added further substantiate that the SOCPNN and MOCPNN possess high accuracy, and that the MOCPNN is robust in pattern classification when equipped with WASD algorithms.

The remainder of this paper is organized into five sections. Section 2 presents the theoretical basis for constructing the SOCPNN and MOCPNN. In Section 3, models of the SOCPNN and MOCPNN are constructed in detail. The corresponding analyses of their computational complexities are provided as well. Section 4 proposes the WASD algorithms for the SOCPNN and MOCPNN. In Section 5, comparative experimental results are presented to substantiate the efficacy and superiority of the proposed SOCPNN, MOCPNN, and WASD algorithms for pattern classification. Section 6 concludes this paper.

2. Theoretical basis

This section presents the theoretical basis for constructing the SOCPNN and MOCPNN. As basis for further discussion, the definition of Chebyshev polynomials is given as follows [15,16]:

Definition 1. For the variable $x\in[-1,1]$, Chebyshev polynomials can be defined as follows:

$$\varphi_{i+2}(x)=2x\varphi_{i+1}(x)-\varphi_i(x)\quad\text{with } \varphi_0(x)=1\quad\text{and}\quad \varphi_1(x)=x,$$

where $\varphi_i(x)$ denotes the Chebyshev polynomial of degree i (with $i=0,1,\ldots$).

Based on the theory of orthogonal polynomial approximation [12,15], an unknown target function $p(x)$ with $x\in[-1,1]$ can be approximated by a group of Chebyshev polynomials as follows:

$$p(x)\approx\sum_{i=0}^{I}a_i\varphi_i(x),\tag{1}$$

where a_i is the weight for $\varphi_i(x)$ and I is the total number of Chebyshev polynomials used to approximate the target function $p(x)$. Notably, I should be sufficiently large.

For a continuous real-valued function $f(X)$ with N variables, where $X:=[x_1\ x_2\cdots x_N]^{\mathrm{T}}\in[0,1]^{N\times1}$ with superscript $^{\mathrm{T}}$ denoting the transpose operator, the Bernstein polynomial can be constructed as follows:

$$B^f_{G,N}(X)=\sum_{g_1=0}^{G}\cdots\sum_{g_N=0}^{G}b_{g_1,\ldots,g_N}p_{g_1}(x_1)\cdots p_{g_N}(x_N),$$

where

$$b_{g_1,\ldots,g_N}=f\left(\frac{g_1}{G},\ldots,\frac{g_N}{G}\right),\qquad p_{g_n}(x_n)=\binom{G}{g_n}x_n^{g_n}(1-x_n)^{G-g_n}$$
(with $n=1,2,\ldots,N$),

and $\binom{G}{g_n}=G!/(g_n!(G-g_n)!)$ denotes a binomial coefficient [21,22].

According to (1), each function in $\{p_{g_n}(x_n)|n=1,2,\ldots,N\}$ can be approximated by Chebyshev polynomials; for example, $p_{g_n}(x_n)\approx\sum_{i_n=0}^{I_{n,g_n}}a_{i_n,n,g_n}\varphi_{i_n}(x_n)$, with $n=1,2,\ldots,N$. Thus, let $P_{g_1,\ldots,g_N}(X):=p_{g_1}(x_1)\cdots p_{g_N}(x_N)$; then, the following can be obtained:

$$\begin{aligned}P_{g_1,\ldots,g_N}(X)&\approx\left(\sum_{i_1=0}^{I_{1,g_1}}a_{i_1,1,g_1}\varphi_{i_1}(x_1)\right)\cdots\left(\sum_{i_N=0}^{I_{N,g_N}}a_{i_N,N,g_N}\varphi_{i_N}(x_N)\right)\\&=\sum_{i_1=0}^{I_{1,g_1}}\cdots\sum_{i_N=0}^{I_{N,g_N}}a_{i_1,1,g_1}\cdots a_{i_N,N,g_N}\varphi_{i_1}(x_1)\cdots\varphi_{i_N}(x_N).\end{aligned}$$

Therefore, the Bernstein polynomial $B^f_{G,N}(X)$ can be approximated by Chebyshev polynomials as follows:

$$\begin{aligned}B^f_{G,N}(X)&=\sum_{g_1=0}^{G}\cdots\sum_{g_N=0}^{G}b_{g_1,\ldots,g_N}P_{g_1,\ldots,g_N}(X)\\&\approx\sum_{i_1=0}^{I_1^{(\max)}}\cdots\sum_{i_N=0}^{I_N^{(\max)}}w_{i_1,\ldots,i_N}\varphi_{i_1}(x_1)\cdots\varphi_{i_N}(x_N),\end{aligned}$$

where $w_{i_1,\ldots,i_N}$ is the weight for $\varphi_{i_1}(x_1)\cdots\varphi_{i_N}(x_N)$, and $I_n^{(\max)}=\max\{I_{n,0},\ldots,I_{n,G}\}$ (with $n=1,2,\ldots,N$).

References [21,22] have shown that $B^f_{G,N}(X)\to f(X)$ uniformly on $X\in[0,1]^{N\times1}$ as $G\to\infty$, i.e., $f(X)=\lim_{G\to\infty}B^f_{G,N}(X)$. Based on such an important result, with G being a sufficiently large value, the following can be obtained:

$$f(X)\approx B^f_{G,N}(X)\approx\sum_{i_1=0}^{I_1^{(\max)}}\cdots\sum_{i_N=0}^{I_N^{(\max)}}w_{i_1,\ldots,i_N}\varphi_{i_1}(x_1)\cdots\varphi_{i_N}(x_N).$$

Thus, the continuous real-valued target function $f(X)$ defined at $X\in[0,1]^{N\times1}$ can be best estimated through the optimal weights $\{w_{i_1,\ldots,i_N}\}$ corresponding to the basis functions $\{\varphi_{i_1}(x_1)\cdots\varphi_{i_N}(x_N)\}$.

Among the numerous choices of total orders for polynomials in several variables [34], the graded lexicographic order is employed