Engineering Applications of Artificial Intelligence 59 (2017) 236–243

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

A PSO based virtual sample generation method for small sample sets: Applications to regression datasets

Zhong-Sheng Chen[a], Bao Zhu[b], Yan-Lin He[a,*], Le-An Yu[b]

[a] College of Information Science & Technology, Beijing University of Chemical Technology, Beijing 100029, China
[b] School of Economics and Management Science, Beijing University of Chemical Technology, Beijing 100029, China

ARTICLE INFO

Keywords:
Small sample set
Virtual sample generation
Information expanded
Regression

ABSTRACT

In the early period of process industries, it is an intractable challenge to build an accurate and robust forecasting model using the collected scared samples. The information derived from small sample sets is unreliable and weak. Thus, the models established based on the small sample sets are inefficient. Virtual sample generation (VSG) is a promising technology which can be used to generate plenty of new virtual samples by the information acquired from small sample sets, aiming at improving the accuracy of forecasting models. To capture the tendency of the raw sample set and reduce information gaps among individuals, an information-expanded function based on triangular membership (TMIE) is developed to asymmetrically expand the domain range in each attribute in this paper. A novel particle swarm optimization based VSG (PSOVSG) approach is proposed to iteratively generate the most feasible virtual samples over the search-space. The effectiveness of PSOVSG is tested against other three methods of VSG over two real cases: multi-layer ceramic capacitors (MLCC) and purified Terephthalic acid (PTA). The simulation results show the proposed PSOVSG achieves better performance than other methods.

1. Introduction

With data-driven modeling methods having been widely used to build forecasting models in many fields, lots of algorithms are conceived to learn the data tendency using the structural dataset collected from a specific field. Such algorithms are data-dependent, i.e., sufficient data and a good distribution assumption are the two necessary conditions to ensure a more accurate model in the applications of classification and regression. Small sample sets problems (Zhu et al., 2016; Chang et al., 2014a, 2015; Li et al., 2012, 2014; Li and Lin, 2013) refer to the case of small amount of samples, where the number of samples is less than 50 in respect to engineering applications or less than 30 in regard to academic researches. Such small number of sample sets cannot completely reveal the whole features of a population due to the insufficient information (Li and Fang, 2009). For instance, a data-driven soft sensor is utilized to build a regression model between the difficult-to-measure variables and the easy-to-measure variables. If the soft sensor is established based on a small quantity of labeled samples, then the model will achieve a low accuracy (Ge, 2014). In spite of the coming of the big data era, the number of samples is still limited due to high consumptions of efforts and time, especially in the early period of the production processes (Li et al., 2006, 2012; Lin and Li, 2010; Li and Lin, 2006, 2008) and chemical process transitions (Zhu et al., 2014). The problem of small sample sets causes serious concerns on the scientific community and industrial circles. Thus, up to date, solving the problem of small sample sets is still a longstanding challenge.

To circumvent this challenge, several relevant researchers have designed and developed varieties of methods to improve the accuracy of the forecasting models based on small sample sets. In a nutshell, the current methods to tackle this issue can be separated into three categories:

(1) Gray forecasting model: this model uses the accumulating generation operator (AGO) of the gray theory to deal with raw sample set for improving the accuracy, such as BGM (1,1) (Chang et al., 2015), GBM (Wang et al., 2014), ANGM (1,1) (Chang et al., 2014b).
(2) Virtual sample generation (VSG): this method fills the information gaps among each raw sample in order to stabilize the forecasting performance through adding newly generated virtual samples, including MTD (Li et al., 2007), GTD (Lin and Li, 2010), GAVSG (Li and Wen, 2014), and Gaussian distribution based VSG (Yang et al., 2011), etc.

* Corresponding author.
E-mail address: heyl@mail.buct.edu.cn (Y.-L. He).

http://dx.doi.org/10.1016/j.engappai.2016.12.024
Received 25 April 2016; Received in revised form 16 October 2016; Accepted 28 December 2016
Available online 13 January 2017
0952-1976/

(3) Feature extraction: this method is an available dimension reduction based method. In this method, the useful attributes or features of the subset are selected to enhance the analytical performance. This method is suitable for medicine cases where size of sample set is small but has fairly high dimensionality (also known as large p small n datasets). Typical applications can be found in the literatures (Espezua et al., 2015; Dernoncourt et al., 2014; Liao, 2011).

Nevertheless, the VSG technology is the most outstanding and popular among the aforementioned three state-of-the-art technologies. The VSG technology can be used to generate a great number of new virtual samples with the information hidden in the small number of raw sample sets. The accuracy of the forecasting models can be improved by means of the expanded information of generated virtual samples. The original idea of VSG stems from the work (Niyogi et al., 1998), where Niyogi et al. used the prior information derived from a given small training set to generate virtual samples for improving image recognition performance. In details, a mathematical transformation was employed to generate new views of a given 3D-object from any other rotation. It has been mathematically proved that the procedure of VSG is equivalent to incorporating the prior knowledge in their work. Following the work, a functional virtual population was proposed by Li et al. (Li et al., 2003) to acquire the robust knowledge of manufacturing systems under dynamic environments using the VSG method. An internalized kernel density estimator (IKDE) (Li and Lin, 2006) was adopted to estimate the population density and carry out the procedure of VSG for building up management knowledge in the early manufacturing stages. In order to deal with dependent datasets, an extend version of IKDE, named GKIDE (Li and Lin, 2008), was developed to extract extra information for expediting the learning ability based on a small sample set. For the purpose of obtaining the scheduling knowledge in the early flexible manufacturing system (FMS), Li et al. successively proposed a mega-fuzzification involving an adaptive-network-based fuzzy inference system (ANFIS) (Li et al., 2006) and a mega-trend-diffusion (MTD) (Li et al., 2007). In the MTD method, the concept of information diffusion was considered (Huang and Moraga, 2004) to accelerate the learning ability using the extra information of virtual samples. To avoid the disadvantages of a single distribution, a multi-distribution MTD was proposed by Zhu et al. to generate virtual samples based on the uniform distribution and triangular distribution (Zhu et al., 2016). Other practical VSG methods based on the principle of information diffusion have been successfully applied in the fields of medicine (Li et al., 2007, 2009; Chao et al., 2011). Aiming at improving the accuracy of the nonlinear function recognition, a combination of segmentation techniques and artificial samples was employed to decrease the errors between real values and estimated values (Tsai and Li, 2008a), the results of which showed the learning accuracy can be significantly improved over a small sample set.

Back-propagation neural networks (BPNNs) are efficient and powerful nonlinear modeling tools and widely used to tackle many real problems encountered in engineering. Unfortunately, the gradient descent-based algorithms like BP algorithm used in single-hidden layer feedforward neural networks (SLFNs) has some inherent flaws, such as expensive consumption in the training stage, local minima, and so on. These flaws greatly restrict the application of the algorithms. To avoid the above demerits of BPNNs, extreme learning machine (ELM) was proposed in literature (Guang-Bin et al., 2004). ELM is a fast learning algorithm for SLFNs. In ELM, the input weights are randomly assigned and the output weights are analytically determined. The performance of ELM has been rigorously proved in literature (Huang et al., 2006). Recently, an extreme learning machine with hierarchical structure (HELM) was successfully employed to handle high-dimensional datasets with noise (He et al., 2014). In other work, a novel double parallel ELM with Pearson correlation coefficient based independent subnets was well-designed to deal with the highly nonlinear data gathered from complex chemical processes (He et al., 2015). In our study, the traditional ELM is adopted to construct the forecasting model.

Inspired by swarm intelligence, particle swarm optimization (PSO) is adopted as a prevalent population-based global stochastic optimization technology to solve nonlinear, multimodal and non-differentiable optimization problems. Unlike genetic algorithm (GA), PSO does not carry out crossover and mutation (Garg, 2016). Meanwhile, PSO is not sensitive to the swarm size as well (Marini and Walczak, 2015). It is worth mentioning that PSO is derivative-free and works well in handling continuous, discrete and integer variables. It could also achieve a better performance against many other conventional optimization technologies due to its intrinsic advantages, namely, comparative potency, flexibility, autobiographical memory, easy implementation, fast convergence and simple principle (Zhao and Zhou, 2015; Kennedy and Eberhart, 1995; Perez and Behdinan, 2007). Recent years, PSO has attracted an increasing attention from researchers and been successfully applied to diverse fields, such as optimal design (Sadeghierad et al., 2010), weights optimization in neural networks (Garg, 2016), trajectory optimization (Zhao and Zhou, 2015), inverse boundary design (Payan et al., 2015), function optimization (Chen et al., 2015), large-scale optimization problems (Zhang and Hui, 2013) and so on.

The main contribution of this paper is that it takes the integrated effects of attributes into consideration and avoids the normal distribution assumption. A novel particle swarm optimization based VSG (PSOVSG) approach is proposed. Different from other VSG approaches, the proposed PSOVSG generates more effective virtual samples to improve the accuracy of the ELM forecasting model. Two real case studies are carried out to demonstrate the effectiveness of the proposed method.

The rest of this paper is organized as follows. Section 2 presents a brief overview on extreme learning machine and constrained particle swarm optimization. In Section 3, a detailed description of the proposed PSOVSG is presented. Two real cases are studied and the results are shown in Section 4, followed by Section 5 as conclusions.

2. Preliminaries

In this paper, we make an attempt to build an accurate and robust extreme learning machine (ELM) model based on the proposed PSOVSG method. In this section, a brief overview of ELM is introduced. Then, the general PSO methodology as well as its constraints handling is presented.

2.1. Extreme learning machine modeling

The basic ELM originated from the literature (Guang-Bin et al., 2004) and was rigorously proved in literature (Huang et al., 2006). Unlike the gradient descent-based single-hidden layer feedforward neural networks (SLFNs), ELM can random assign the input weights, and the output weights are analytically determined using the least square method. Simple in theory and fast in implementation, ELM becomes a competitive machine learning technology to deal with many sophisticated problems, like high-dimensional datasets with noise (He et al., 2014) and complex chemical processes modeling (He et al., 2015).

Considering that there is a training sample set with N distinct samples $(\boldsymbol{x}_i, \boldsymbol{y}_i)$, where $\boldsymbol{x}_i = [x_{i1}, x_{i1}, \ldots, x_{ip}]^T \in \mathbb{R}^p$ and $\boldsymbol{y}_i = \left[y_{i1}, y_{i1}, \ldots, y_{iq}\right]^T \in \mathbb{R}^q$. For regression tasks, q equals to 1, while for classification tasks, q indicates the number of labels or categories. The output of ELM with Nh hidden nodes is given by

$$y_i = \sum_{j=1}^{Nh} \beta_i f(\mathbf{w}_i \cdot \mathbf{x}_j + \mathbf{b}_i) \tag{1}$$

where $|\bullet|$ denotes inner product operation, and $f(x)$: $\mathbb{R} \rightarrow \mathbb{R}$ is activa-

tion function. The above equation can be written as the following matrix format:

$$\mathbf{Y} = \mathbf{H}\boldsymbol{\beta} = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_{Nh}]\boldsymbol{\beta} = \begin{bmatrix} f(\mathbf{w}_1 \cdot \mathbf{x}_1 + \mathbf{b}_1) & \cdots & f(\mathbf{w}_{Nh} \cdot \mathbf{x}_1 + \mathbf{b}_{Nh}) \\ \cdots & f(\mathbf{w}_i \cdot \mathbf{x}_j + \mathbf{b}_i) & \cdots \\ f(\mathbf{w}_1 \cdot \mathbf{x}_N + \mathbf{b}_1) & \cdots & f(\mathbf{w}_{Nh} \cdot \mathbf{x}_N + \mathbf{b}_{Nh}) \end{bmatrix} \times \begin{bmatrix} \beta_1 \\ \cdots \\ \beta_N \end{bmatrix} \quad (2)$$

where $\mathbf{w}_i = [w_{i1}, w_{i2}, \ldots, w_{iq}]^T$, $i = 1, 2, \ldots, Nh$ denotes the weight vector connecting the i-th hidden neuron and the input neurons, $\boldsymbol{\beta} = \left[\beta_{i1}, \beta_{i2}, \ldots, \beta_{iq}\right]^T$ is the vector connecting the i-th hidden neuron and the output neurons, and $\mathbf{H}$ is the hidden layer output matrix. Any bounded non-constant piecewise continuous function can be chosen as the activation function, such as the sigmoid function, the Gaussian function, sine, and so on (Huang et al., 2006). In this paper, the sigmoid function is used as the activation function, which is defined as

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

Assume that the input weights $\mathbf{w}$ and the biases of the hidden neuron $\mathbf{b}$ are fixed. The sole parameters of liner system $\mathbf{Y} = \mathbf{H}\boldsymbol{\beta}$ are shown in Eq. (2). The weights $\boldsymbol{\beta}$ can be analytically determined by finding a least-squares solution $\hat{\boldsymbol{\beta}}$ of the system

$$\hat{\boldsymbol{\beta}} = \min_{\beta|w,b} \|H\boldsymbol{\beta} - Y\| \quad (4)$$

where $\|\bullet\|$ denotes l_2- norm. Therefore, the optimal solution is

$$\hat{\boldsymbol{\beta}} = \mathbf{H}^{\dagger}\mathbf{Y} \quad (5)$$

where $\mathbf{H}^{\dagger}$ is Moore-Penrose generalized inverse of $\mathbf{H}$, which can be computed via the orthogonal projection: the singular value decomposition (SVD).

2.2. Constrained particle swarm optimization (PSO)

Stemmed from the natural analogue, PSO was developed by Kennedy and Eberhart in 1995 (Kennedy and Eberhart, 1995) and mimics the foraging behavior and collective collaboration of swarms like fish schooling and bird flocking. In PSO, solutions of optimization problems can be represented as points (particles) in a d-dimension space. Each particle is characterized by a position vector and a velocity vector. A particle represents a potential solution and is able to move through the parameter space shaped via constrains belonging to a specific optimization problem. Essentially, for finding the best position, PSO makes use of not only individual particle's experiences but also that of the whole population rather than derivative information to change its positions. During the optimization process, the algorithm starts from a random swarm initialization. The positions and velocities of the particles of the swarm are generated randomly as start point. Then the fitness of each particle in the swarm is evaluated at each iteration to find the personal best position (called **pbest**, the best position of a particle that it has explored so far) and global best position (called ***gbest***, the best position of the whole swarm that it has explored so far) (Zhao and Zhou, 2015; Payan et al., 2015). As soon as one or more user-predefine termination criteria are met, the iterative search process is terminated (Marini and Walczak, 2015).

Let design variable vector $\boldsymbol{x}_i = [x_{i1}, x_{i1}, \ldots, x_{id}]^T \in \mathbb{R}^d$, $i = 1, 2, \ldots, m$ denotes the current position of the i-th particle and $\mathbf{v}_i = [v_{i1}, v_{i1}, \ldots, v_{id}]^T \in \mathbb{R}^d$ denote the current velocity of the i-th particle in the d-dimension search space, where m is the swarm (population) size, $\boldsymbol{x}_i$ is associated with a potential solution of a particular optimization problem. The initial position $\boldsymbol{x}_{i0}$ is uniformly distributed in the domain range $[x_{i,max}, x_{i,min}]$, where $x_{i,max}$ and $x_{i,min}$ indicate the upper and lower boundaries of the i-th design variable, respectively. The particles in the swarm evolve at $(k + 1)$-th iteration according to the position update equation

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \mathbf{v}_i^{k+1} \quad (6)$$

and the velocity update equation,

$$\mathbf{v}_i^{k+1} = w^k\mathbf{v}_i^k + c_1r_1(\boldsymbol{pbest}_i^k - \mathbf{x}_i^k) + c_2r_2(\mathbf{gbest}_i - \mathbf{x}_i^k) \quad (7)$$

where c_1 and c_2 are non-negative constants, known as cognitive coefficient and social coefficient respectively, which meet $0 < c_1+c_2 < 4$, usually set $c_1=c_2 = 2$ (Perez and Behdinan, 2007), r_1 and r_2 are uniformly random numbers independently distributed between 0 and 1, and w^k is inertia weight, which is linearly decreased from maximum inertia weight to minimum inertia weight as follow.

$$w^k = w_{max} - \frac{w_{max} - w_{min}}{K}k \quad (8)$$

Several works support that the combination $w_{max} = 0.9$ and $w_{min} = 0.4$ can achieve the best performances, and a detail dynamic adjustment strategy for inertia weight can be found in literature (Marini and Walczak, 2015).

In order to make PSO have ability to cope with the problem immanent equality and inequality constraints, a parameter-less adaptive penalty scheme (Perez and Behdinan, 2007) is used, depicted in Eq. (9). The scheme is used to reduce the constrained problem to an unconstrained one.

$$f_{new}(x) = \{f(x) \text{if } x \text{ is feasible}, f(x) + \sum_{i=1}^{n_c} k_i\overline{g}_i(x) \quad otherwise \quad (9)$$

The penalty weight is computed based on the average of the objective function and the level of violation of each constraint,

$$k_i = |\overline{f}(x)| = \frac{\overline{g}_i(x)}{\sum_{l=1}^{n_c}\overline{g}_l^2(x)}$$
$$\overline{f}(x) = \frac{1}{m}\sum_{j=1}^{m} f_{new}(x)$$
$$\overline{g}_i(x) = \frac{1}{m}\sum_{l=1}^{m} g_l(x) \quad (10)$$

where $f(x)$ denotes the fitness function, $\overline{f}(x)$ denotes the average of the fitness function values in the current swarm, $g_i(x)$ denotes a pre-define violated constraint value, $\overline{g}_i(x)$ is the average of violation of i-th constraint over the current swarm, n_c denotes the number of constraints, and m is the swarm size, which is problem-dependent and generally kept in between 20 and 50 (Garg, 2016).

3. The proposed PSO-based virtual sample generation (PSOVSG)

The aim of this paper is to improve accuracy of forecasting model by adding virtual samples. In other words, we attempt to design a unique mechanism of virtual sample generation. The mechanism can generate distinct virtual samples. In this section, the understandable deduction of TMIE function is given firstly. In addition, the detailed description of the proposed PSOVSG is presented in the following subsection (shown as Fig. 1).

3.1. Asymmetric acceptable domain range expansion

For the sake of capturing data tendency, an appropriate estimation of the underlying population distribution for small sample sets plays a significant role in virtual sample generation. Inspired by mega-trend-diffusion (MTD) (Li et al., 2007) and driven by estimating data tendency, TMIE function (depicted in Fig. 2) is developed to roughly determine the acceptable domain range and plausibly describe the underlying population distribution. Similarly, TMIE can be used to asymmetrically expand the acceptable domain range by using a triangular structure. The triangular structure of TMIE is similar to that of MTD. Notably, quite different from MTD, TMIE does not undergo virtual attribute creation and is deduced from the similar triangles.

Suppose $\boldsymbol{X} = \{x_i | x_i \in \mathbb{R}\}_{i=1}^n$, denote a set of observations with small size $n(n \leq 50)$. The central location (CL) of the observations $\boldsymbol{X}$ is estimated by the following formula.

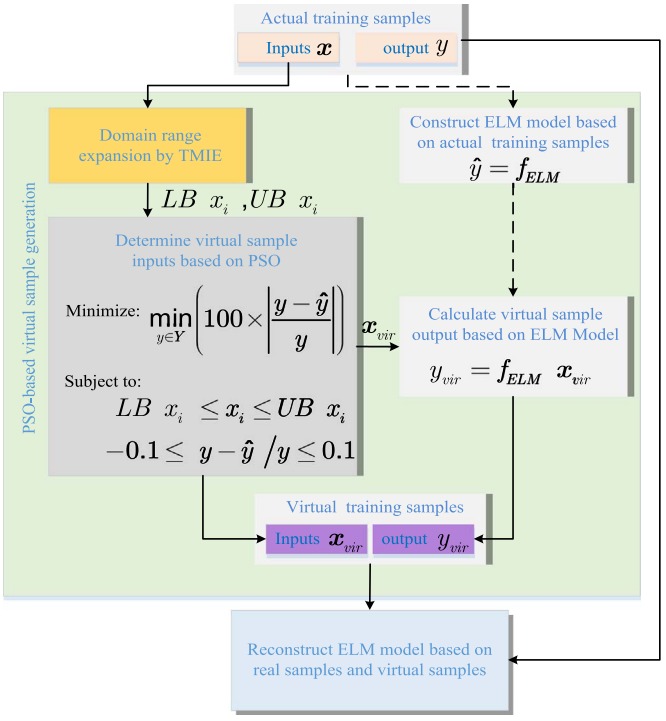

Fig. 1. Flowchart of the proposed PSOVSG.

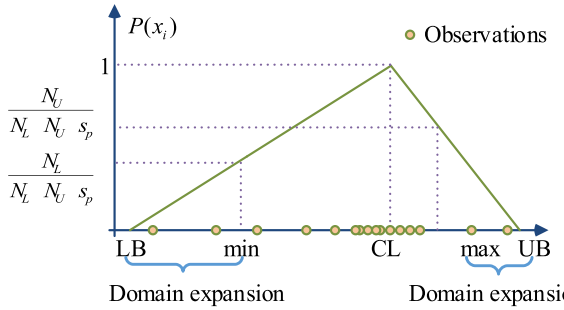

Fig. 2. The developed information-expanded based on triangular membership (TMIE) function.

$$CL = \frac{1}{n} \sum_{i=1}^{n} x_i \tag{11}$$

Following the concept applied to the MTD approach, the occurrence possibility value rather than probability is employed in TMIE function to avoid the normal distribution assumption. In Fig. 2, it is straightforward to show that the degree of skewness of triangle shape of TMIE function is related to the relative amount of observations located on both side of CL, where the abscissa represents the different observations, and the ordinate represents the occurrence possibility values of the observations. Several well-known measures of skewness, such as the standardized third moment, Bowley's skewness, etc. are failed to obtain a robust estimation because such measures require a large amount of observations (Ekström and Jammalamadaka, 2012). Accordingly, the left and right skewness magnitudes, Sk_L and Sk_U are regarded as a measure of the asymmetry of the population distribution, defined in Eqs. (12) and (13), respectively.

$$Sk_L = \frac{N_L}{N_L + N_U + s_p} \tag{12}$$

$$Sk_U = \frac{N_U}{N_L + N_U + s_p} \tag{13}$$

where N_L (N_U) is the number of observations smaller (greater) than CL,

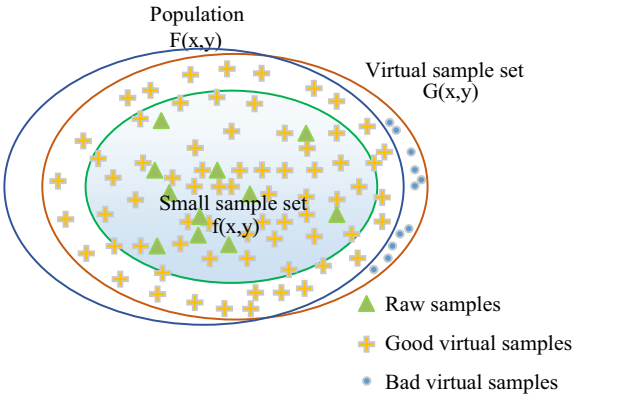

Fig. 3. The relationship among the population, virtual sample set and small sample set. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

s_p is unknown shape parameter to slightly adjust the degree of skewness, taken as 1 in this paper. According to the similar triangles portrayed in Fig. 2, the asymmetric domain range starts from the lower boundary LB and ends with the upper boundary UB, described mathematically as

$$\begin{aligned} LB &= CL - \tfrac{1}{Sk_U} \times (CL - min) \\ UB &= CL + \tfrac{1}{Sk_L} \times (max - CL) \end{aligned} \tag{14}$$

where min denotes the minimum of the observations, max denotes the maximum of the observations.

3.2. Virtual samples generation by the proposed PSOVSG

In this subsection, the theoretical foundation of virtual sample generation and the implementation of the proposed PSOVSG are demonstrated as follows.

The feature of small sample set is illustrated in Fig. 3. It vividly reveals that the small sample set (green circle) consisting of only few raw samples (green triangles) is a subset of the population (blue circle). The overlap between the population and the virtual sample set (brown circle) is the informative area which is filled of a large number of virtual samples and a small number of raw samples. Obviously, the virtual samples are embedded in the gaps among raw samples, reducing the information gaps of the small sample set. It is clear that the population is a neighbor of the small sample set. Thus, the hyperplane $G(x, y)$ corresponding to small sample set moves towards the hyperplane $F(x, y)$ corresponding to the population after addition of virtual samples. Therefore, the process of virtual sample generation makes a contribution to accuracy improvement. It should be noticed that not all generated virtual samples would help to enrich information derived from the small sample set. The generated virtual sample can be divided into two types: the good virtual samples (yellow cross) and the bad virtual samples (blue dot), as shown in Fig. 3. The good virtual samples indicate the virtual samples which have a positive impact on accuracy of forecasting model while the bad ones indicate the virtual samples which have a negative impact on accuracy of forecasting model.

The trick of the proposed PSOVSG is that it tries to find a new combination of inputs to minimize the smallest relative percent error between the prediction value and the real value through optimization process and ensure better virtual samples generation (shown in Fig. 3). In this paper, the virtual samples generation for small sample set is considered as a nonlinear constrained optimization in a certain sense, mathematically described as follows,

$$\begin{aligned} &\text{Minimize} \quad f(x) \\ &\text{Subject to} \quad LB_i \le x_i \le UB_i \;\; ; \; i = 1, 2, \ldots, n_a \\ &\qquad\qquad G_k(x) \le 0 \qquad ; \; k = 1, 2, \ldots, n_c \end{aligned} \tag{15}$$

where $f(x)$ denotes the fitness function,$x = [x_1, x_2, \ldots, x_{n_a}]$ denotes the n_a-dimensional vector of decision variables, $LB(UB)$ is the lower (upper) boundary of decision variable x_i, $G_k(x)$ is the k-th nonlinear inequality, n_a is the number of decision variables, and n_c is the number of the nonlinear inequality. Each nonlinear inequality $G_k(x)$ is usually derived from a specific application. This nonlinear constrained optimization problem can be solved by constrained particle swarm optimization described in Section 2.2.

Assume that there is an available small sample set $\mathbf{D}_{small} = \{(\mathbf{x}_i, y_i)|\mathbf{x}_i = [x_{i1}, x_{i2}, \ldots, x_{in_a}] \in \mathbb{R}^{n_a}, y_i \in \mathbb{R}\}_{i=1}^{n_o}$, where n_o is the number of observations, and n_a is the number of inputs. The raw small sample set $\mathbf{D}_{small}$ is randomly divided into two subsets: training set $\mathbf{D}_{train} = \{[\mathbf{x}_i, y_i]\}_{i=1}^{n_{train}}$ and testing set $\mathbf{D}_{test} = \{[\mathbf{x}_i, y_i]\}_{i=1}^{n_{test}}$. Many experimental results confirm that an unacceptable prediction performance is obtained when $\mathbf{D}_{train} = \{[\mathbf{x}_i, y_i]\}_{i=1}^{n_{train}}$ is directly used to build up forecasting model. Although the forecasting model is fragile, it has acquired partial relationship between input variable $\mathbf{x}$ and y. To make the best use of the knowledge acquired by the forecasting model before virtual sample addition, the forecasting model with a 10% mean absolute percentage error (MAPE) is suggested to calculate the virtual output when a new virtual input is generated (Li and Lin, 2006; Li and Wen, 2014). Thus, the fitness function $f(x)$ is defined as the following formula.

$$f(x) = \min_{y \in \mathbf{Y}} \left(100 \times \left| \frac{y - \hat{y}}{y} \right| \right) \tag{16}$$

where y denotes the real value, $\mathbf{Y}$ denotes a vector consisting of y, and $\hat{y}$ denotes the prediction values. Considering feasibility of generated virtual samples, we impose a nonlinear inequality that increases quality of virtual samples, thus, the optimization described Eq.(15) can be further revised as

$$\begin{aligned} &\text{Minimize} \quad f(x) \\ &\text{Subject to} \quad LB_i \le x_i \le UB_i \qquad ; i = 1, 2, \ldots, n_a \\ &\qquad\qquad G_k(x) \le 0 \qquad\qquad ; k = 1, 2, \ldots, n_c \\ &\qquad\qquad -0.1 \le (y - \hat{y})/y \le 0.1 \end{aligned} \tag{17}$$

The population distribution is generally unknown but could be roughly estimated by the developed TMIE approach and has a great impact on virtual sample generation. The process of virtual sample generation is illustrated by Fig. 4. First of all, the rectangle in $x_1 - x_2$ plane is formed search-space. The sides on each dimension is treated as domain ranges calculated by Eq. (14). In addition, a constrained particle swarm optimization (PSO) process starting from the vertices of the rectangle is performed over the search-space to solve the nonlinear constrained described in Eq. (15) until the one of the following termination criteria: the maximum number of iterations K, is reached, or the desired value of fitness ε_{tol} is met, or $|f^{k+\Delta k} - f^k| < \varepsilon_{tol}$ with $\Delta k = 50$ is achieved. A virtual sample is generated around real samples in acceptable prediction area as soon as each iterative search process is terminated. Finally, repeat the above process of virtual sample generation until the desired number of virtual samples n_{vir} is satisfied. Thus, virtual sample set $\mathbf{D}_{vir}$ with size n_{vir} is artificially generated.

To sum up, the proposed approach is illustrated in Fig. 1 and presented in following four steps.

Step 1. An forecasting model $\hat{H}_{ELM}$ is directly constructed by ELM. The model is trained on $\mathbf{D}_{train}$ and tested on $\mathbf{D}_{test}$. It is possible to ensure that the MAPE of the model is less than 10% when a proper the number of hidden neurons Nh is assigned based on trial and error.

Step 2. Asymmetric acceptable domain range of each input is calculated by Eq. (14) and considered as boundary constraint for the optimization problem described in Eq. (17).

Step 3. Generate n_{vir} virtual samples through the aforementioned process of virtual sample generation, where m vertices is set as start point for PSO. As shown in Fig. 4, the number of all the possible vertices is 2^n when dimension is n. Usually, $2^n > m$ holds.

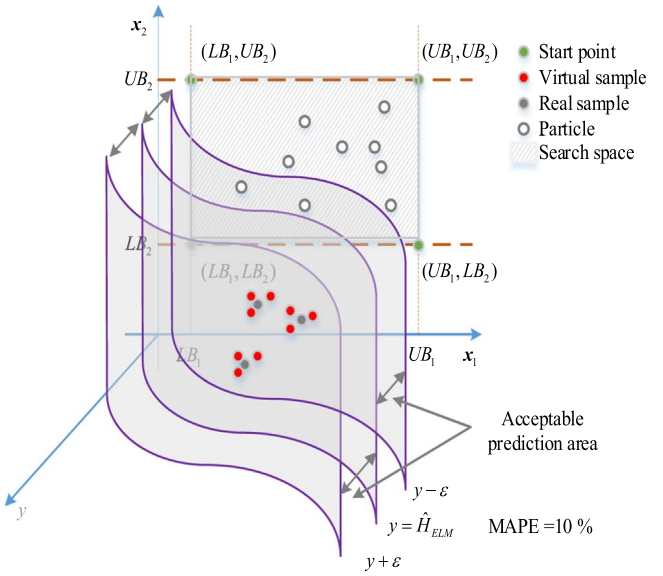

Fig. 4. The process of virtual sample generation.

Step 4. Form a synthetic sample set $\mathbf{D}_{syn}$ by combining the original training samples $\mathbf{D}_{train}$ with the generated virtual samples $\mathbf{D}_{vir}$. A final forecasting ELM model is obtained by training on synthetic sample set $\mathbf{D}_{syn}$ and validating on the same testing set $\mathbf{D}_{test}$.

4. Case study

In this section, two real case, multi-layer ceramic capacitors (MLCC) and purified Terephthalic acid (PTA) are examined to evaluate the proposed PSOVSG. Two metrics of forecasting performance, the average MAPE (AveMAPE) and average error improving rate (AveEIR) of 30 independent runs, are employed to demonstrate the effectiveness of the proposed approach. The following experimental results with different size of training set show that the proposed approach has a better forecasting performance than that of other methods. All experiments are carried out using the same test set and parameter settings for PSO, listed in Table 1. The AveMAPE and AveEIR are computed by the following equations.

$$\begin{aligned} MAPE &= \frac{1}{n_o} \sum_{n=1}^{n_o} \left| \frac{y - \hat{y}}{y} \right| \times 100\% \\ AveMAPE &= \frac{\sum_i^{30} MAPE_i}{30} \end{aligned} \tag{18}$$

$$\begin{aligned} EIR &= \frac{MAPE_{before} - MAPE_{after}}{MAPE_{before}} \times 100\% \\ AveEIR &= \frac{\sum_{i=1}^{30} EIR_i}{30} \end{aligned} \tag{19}$$

Table 1
The parameter settings for PSO.

Notation	Value	Parameter
m	50	Population size
w^k	$w^k = w_{\max} - \frac{w_{\max} - w_{\min}}{K} k$ where $w_{\max} = 0.9$ and $w_{\min} = 0.4$	Inertia weight
c_1	2	Cognitive coefficient
c_2	2	Social coefficient
K	500	Maximum number of iteration
ε_{tol}	10^{-6}	Desired value of fitness

4.1. Case 1: MLCC

A multi-layer ceramic capacitor is a kind of passive component. It is composed of ceramic ponder and is able to store and release an electric charge in a short time (Li et al., 2012). However, it is difficult to obtain stable physical characteristics of ceramic ponder in a batch of production. Relative permittivity (*RK*) is considered as the most significant characteristic. To cut down the related costs and shorting inspection cycle in incoming quality control, *RK* requires a fast, accurate and stable prediction owing to. The MLCC sample set is consisting of 44 samples, with the twelve key factors affecting the *RK* of AD143 ceramic powder, provided by a manufacturer in Taiwan. The datasheet and the detailed descriptions of the twelve key factors can be found in Tsai and Li (2008b).

In order to explore the influence of size of training set on the proposed approach, we perform a sensitivity analysis for size of training set. Specifically, we randomly draw a certain number of samples from the raw sample set as the training set $\mathbf{D}_{train}$, the remainder is treated as the testing set $\mathbf{D}_{test}$. The size of training sample is set to 5, 10, 15, 20, 25, sequentially. For each size of training set, 30 independent runs are carried out. The computational results before adding virtual samples show low accuracy, as presented in Fig. 5. The number of hidden neurons *Nh* of ELM is roughly determined based on trial and error, assigned to 85. Obviously, along with the increase of the size of training set, the AveMAPE of ELM monotonically decreases, while the AveMAPE of BPNN has a noticeable fluctuation. Compared to Tsai et al. (Tsai and Li, 2008b), ELM has a better performance than BPNN in small sample case. To confirm a significant improvement between the original ELM model and the final ELM model retrained after adding 100 virtual samples, the further analytic comparisons are shown in Figs. 6 and 7 and exhibited in Table 2. It is apparent that the original ELM has a notable improvement after virtual sample addition. The AveMAPE achieve the minimum, 3.518%, as the size of training set is 30. Meanwhile the AveEIR reaches the maximum, 34.280%, presented in Table 2. The comparisons among our approach, MTD, and the two previous methods reported in other papers (Li et al., 2012; Tsai and Li, 2008b), summarized in Fig. 8 and Table 3. It is found that the proposed approach outperforms the other three methods in different size of training set except the smallest training set size (i.e. 5). Therefore, the conclusions are two-folds. On one hand, the PSOVSG is reasonable and effective to improve forecasting performance for small sample set. One the other hand, the AveMAPE of the forecasting model decreases when the size of the training set increases and virtual samples are added to original training set as well.

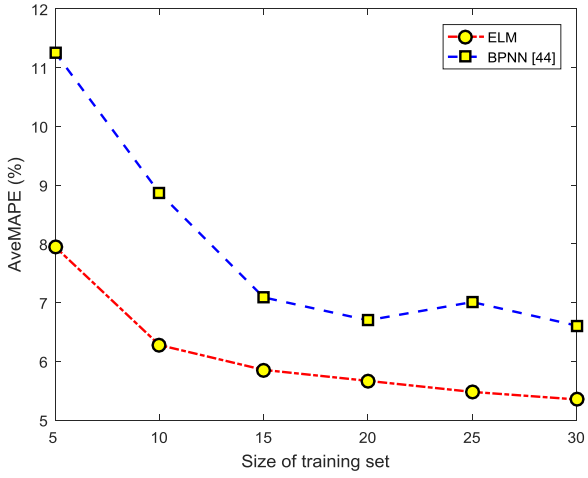

Fig. 5. Comparisons of computational results before adding virtual samples.

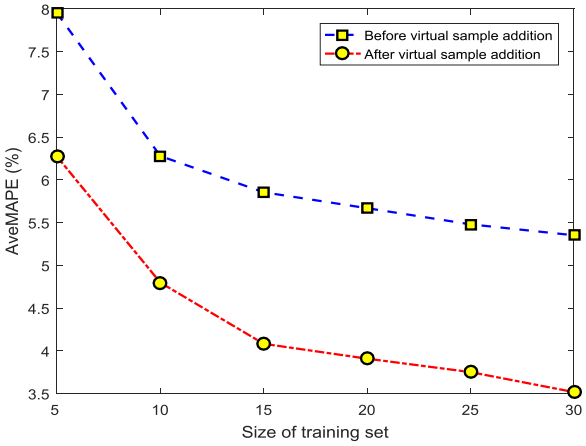

Fig. 6. The average accuracy of the proposed PSOVSG before and after adding 100 virtual samples.

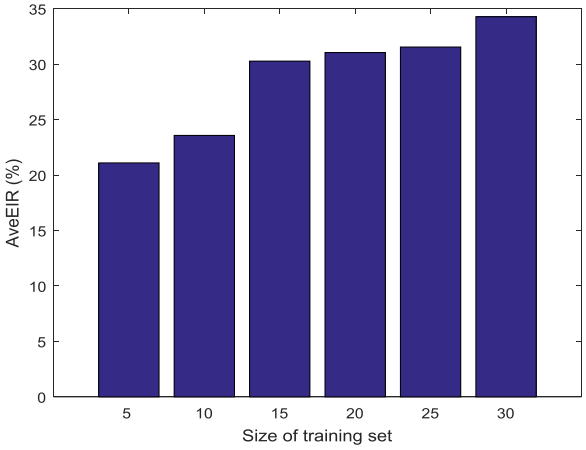

Fig. 7. The accuracy improvements of the proposed PSOVSG after adding 100 virtual samples.

Table 2
The average accuracy of the proposed PSOVSG before and after adding 100 virtual samples.

Size	Before (Average MAPE) (%)	After (Average MAPE) (%)	Improvement (Average EIR) (%)
5	7.951	6.273	21.099
10	6.279	4.798	23.578
15	5.855	4.082	30.277
20	5.669	3.909	31.047
25	5.480	3.751	31.548
30	5.352	3.518	34.280

4.2. Case 2: PTA

Purified Terephthalic acid is one of the most significant organic raw materials, widely used in the manufacture of polyester films and fibers, molded plastics, chemical and pharmaceutical industries. The consumption of the acetic acid is considered as a main indicator among the indicators of PTA plant assessment but is difficult to accurately measure (He et al., 2015). Therefore, an accurate and robust forecasting of consumption of the acetic acid is required to make contributions to reducing production cost and improving enterprise competiveness. The PTA sample set is collected from a real production process of PTA solvent system, including 260 samples with seventeen input attributes and one output attribute. The seventeen factors are selected through expert's analyses, which affect the acetate consumption. We randomly extract 50 samples from the PTA sample set. Then the 50 samples are

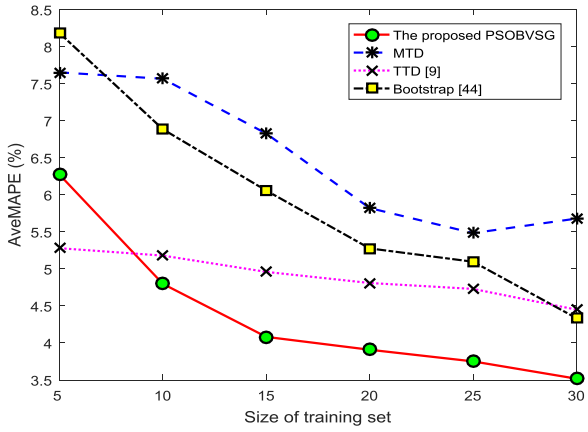

Fig. 8. Comparisons of computational results among the proposed PSOVSG, MTD, and the other two methods.

Table 3
The summary of the results using different methods with various size of training set after adding 100 virtual samples.

Method	Size					
	5	10	15	20	25	30
PSOVSG (%)	6.273	4.798	4.082	3.909	3.951	3.518
MTD (%)	7.654	7.568	6.828	5.343	5.014	4.286
TTD (Li et al., 2012) (%)	5.280	5.180	4.960	4.810	4.730	4.450
Bootstrap (Tsai and Li, 2008b) (%)	8.185	6.886	6.058	5.277	5.097	4.335

Table 4
The average accuracy of the proposed approach before and after adding virtual samples with different size.

Size	Before (Average MAPE) (%)	After (Average MAPE) (%)	Improvement (Average EIR) (%)
100	1.553	1.309	15.715
150	1.553	1.219	21.540
200	1.553	1.122	27.794
250	1.553	1.110	28.560
300	1.553	1.088	29.977
350	1.553	1.022	34.182
400	1.553	0.971	37.473

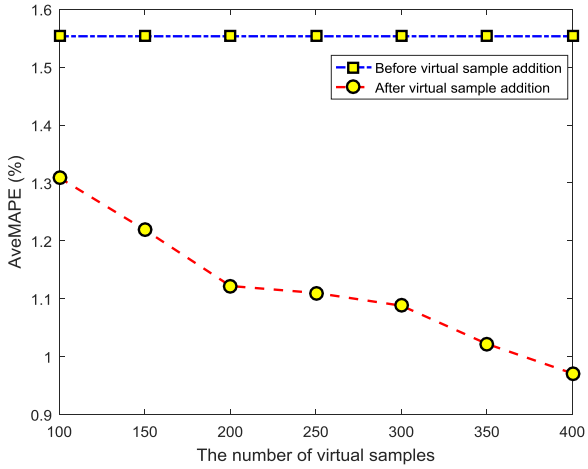

Fig. 9. The average accuracy of the proposed PSOVSG before and after adding virtual samples with different size.

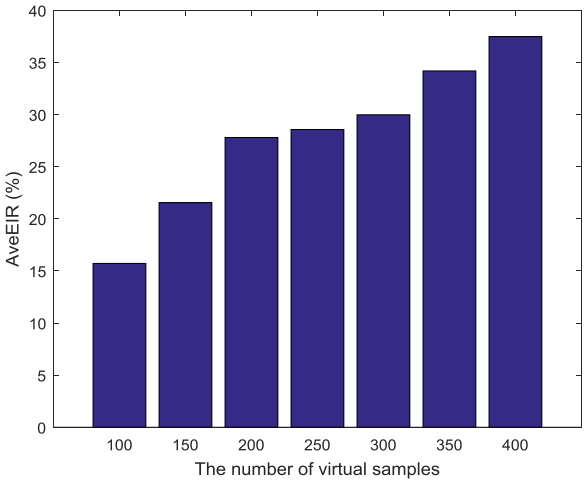

Fig. 10. The accuracy improvements of the proposed PSOVSG before and after adding virtual samples with different size.

Table 5
Cmparisons of forecasting accuracy among different number of virtual samples.

Method	The number of virtual samples n_{vir}.						
	100	150	200	250	300	350	400
PSOVSG (%)	1.309	1.219	1.122	1.110	1.088	1.022	0.971
MTD (%)	1.482	1.468	1.446	1.456	1.454	1.433	1.590
TTD (%)	1.498	1.429	1.431	1.408	1.393	1.384	1.378
Bootstrap (%)	1.419	1.407	1.398	1.377	1.302	1.290	1.326

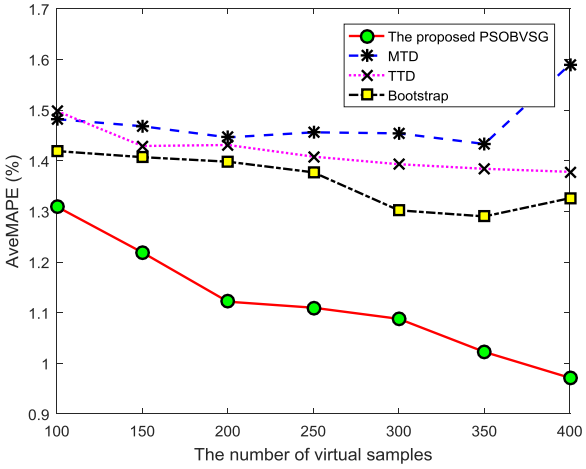

Fig. 11. Comparisons of computational results among the proposed PSOVSG and the other three methods.

partitioned into two parts in a random manner: training set $\mathbf{D}_{train}$ with 30 samples and testing set $\mathbf{D}_{test}$ with 20 samples. In order to investigate the influence of the number of virtual samples on the accuracy of forecasting model, we carry out experiments with different n_{vir}. For each experiment, 30 independent runs are carried out. The number of hidden neurons Nh of ELM is taken as 125 based on the rule of trial and error. Table 4 summarizes the experimental results, indicating that the AveMAPE of the final forecasting model decreases when the number of virtual samples increases. Figs. 9 and 10 exhibit that there are significant accuracy improvements before and after adding virtual samples with different size. The best improvement occurs when n_{vir} equals to 400 (shown in Fig. 10) and the AveMAPE reaches by 0.971% (shown in Table 4). The comparisons among the proposed PSOVSG and the other three methods are shown in Table 5 and Fig. 11, manifesting the proposed PSOVSG has a better performance with

regard to accuracy. On the whole, PSOVSG is a reasonable and effective tool for forecasting tasks with small samples sets.

4.3. Discussion

Since the training samples are insufficient, the performance of the original model is poor. After executing the proposed PSOVSG, a considerable number of virtual samples are generated in a feasible region. The virtual samples could fill the information gaps caused by sparsity of the training samples. Therefore, the performance of the final model retrained using the original training samples and the generated virtual samples is dramatically improved.

Previous methods pay little attention to the integrated effects of attributes and are hard to avoid the normal distributions. As the number of generated virtual samples increases, the performance is slightly improved. In the previous methods, the virtual samples are not restricted to the feasible region. Thus, the virtual samples might deviate from the actual ones. When the number of generated virtual samples grows larger (i.e. 400), the cumulative effects of the deviation might even degrade the performance of the final model.

In a nutshell, the computational results exhibited earlier are thus not fluke at all. The proposed PSOVSG is suitable to solve small sample sets problems.

5. Conclusion

Faced with the low accuracy of data-driven modeling methods due to insufficient training samples, a novel PSOVSG is proposed to generate new virtual samples for improving accuracy of forecasting model. Two real cases are engaged to verify the effectiveness of the proposed PSOVSG method. Computational results show the generated virtual samples by PSOVSG enlarge the original training set so that the enriched information of raw small sample set reduces the information gaps among individuals. The accuracy of foresting model is improved after adding virtual samples. To sum up, PSOVSG is a reasonable and effective tool for forecasting tasks with small samples sets.

The number of virtual samples has an influence on the accuracy of forecasting model. Furfure researches might more focus on finding an effective method to determine the optimal number of virtual samples. Besides, the shape parameter of the TMIE function will be optimized.

Acknowledgment

This research was partly funded by National Natural Science Foundation of China (No. 61533003 and 61473026).

References

Chang, C.J., Li, D.C., Dai, W.L., Chen, C.C., 2014a. A latent information function to extend domain attributes to improve the accuracy of small-data-set forecasting. Neurocomputing 129, 343–349.
Chang, C.J., Li, D.C., Chen, C.C., Chen, C.S., 2014b. A forecasting model for small non-equigap data sets considering data weights and occurrence possibilities. Comput. Ind. Eng. 67, 139–145.
Chang, C.J., Li, D.C., Huang, Y.H., Chen, C.C., 2015. A novel gray forecasting model based on the box plot for small manufacturing data sets. Appl. Math. Comput. 265, 400–408.
Chao, G.-Y., Tsai, T.-I., Lu, T.-J., Hsu, H.-C., Bao, B.-Y., Wu, W.-Y., Lin, M.-T., Lu, T.-L., 2011. A new approach to prediction of radiotherapy of bladder cancer cells in small dataset analysis. Expert Syst. Appl. 38, 7963–7969.
Chen, D., Chen, J., Jiang, H., Zou, F., Liu, T., 2015. An improved PSO algorithm based on particle exploration for function optimization and the modeling of chaotic systems. Soft Comput. 19, 3071–3081.
Dernoncourt, D., Hanczar, B., Zucker, J.D., 2014. Analysis of feature selection stability on high dimension and small sample data. Comput. Stat. Data Anal. 71, 681–693.
Ekström, M., Jammalamadaka, S.R., 2012. A general measure of skewness. Stat. Probab. Lett. 82, 1559–1568.
Espezua, S., Villanueva, E., Maciel, C.D., Carvalho, A., 2015. A projection pursuit framework for supervised dimension reduction of high dimensional small sample datasets. Neurocomputing 149, 767–776.
Garg, H., 2016. A hybrid PSO-GA algorithm for constrained optimization problems. Appl. Math. Comput. 274, 292–305.
Ge, Z.Q., 2014. Active learning strategy for smart soft sensor development under a small number of labeled data samples. J. Process Control 24, 1454–1461.
H. Guang-Bin, Z. Qin-Yu, S. Chee-Kheong, 2004. Extreme learning machine: a new learning scheme of feedforward neural networks. Vol. 2, pp. 985–990.
He, Y.L., Geng, Z.Q., Zhu, Q.X., 2015. Data driven soft sensor development for complex chemical processes using extreme learning machine. Chem. Eng. Res. Des. 102, 1–11.
He, Y.-L., Geng, Z.-Q., Xu, Y., Zhu, Q.-X., 2014. A hierarchical structure of extreme learning machine (HELM) for high-dimensional datasets with noise. Neurocomputing 128, 407–414.
Huang, C.F., Moraga, C., 2004. A diffusion-neural-network for learning from small samples. Int. J. Approx. Reason. 35, 137–161.
Huang, G.B., Zhu, Q.Y., Siew, C.K., 2006. Extreme learning machine: theory and applications. Neurocomputing 70, 489–501.
Kennedy, J., Eberhart, R., 1995. Particle swarm optimization. In: Neural Networks, 1995. Proceedings of IEEE International Conference. pp. 1942–1948 vol.1944.
Li, D.C., Lin, Y.S., 2006. Using virtual sample generation to build up management knowledge in the early manufacturing stages. Eur. J. Oper. Res. 175, 413–434.
Li, D.C., Lin, Y.S., 2008. Learning management knowledge for manufacturing systems in the early stages using time series data. Eur. J. Oper. Res. 184, 169–184.
Li, D.C., Chen, L.S., Lin, Y.S., 2003. Using functional virtual population as assistance to learn scheduling knowledge in dynamic manufacturing environments. Int. J. Prod. Res. 41, 4011–4024.
Li, D.C., Chang, C.C., Liu, C.W., 2012. Using structure-based data transformation method to improve prediction accuracies for small data sets. Decis. Support Syst. 52, 748–756.
Li, D.C., Lin, L.S., Peng, L.J., 2014. Improving learning accuracy by using synthetic samples for small datasets with non-linear attribute dependency. Decis. Support Syst. 59, 286–295.
Li, D.C., Wu, C.S., Tsai, T.I., Chang, F.M.M., 2006. Using mega-fuzzification and data trend estimation in small data set learning for early FMS scheduling knowledge. Comput. Oper. Res. 33, 1857–1869.
Li, D.C., Wu, C.S., Tsai, T.I., Lina, Y.S., 2007. Using mega-trend-diffusion and artificial samples in small data set learning for early flexible manufacturing system scheduling knowledge. Comput. Oper. Res. 34, 966–982.
Li, D.C., Fang, Y.H., Lai, Y.Y., Hu, S.C., 2009. Utilization of virtual samples to facilitate cancer identification for DNA microarray data in the early stages of an investigation. Inf. Sci. 179, 2740–2753.
Li, D.-C., Fang, Y.-H., 2009. A non-linearly virtual sample generation technique using group discovery and parametric equations of hypersphere. Expert Syst. Appl. 36, 844–851.
Li, D.-C., Lin, L.-S., 2013. A new approach to assess product lifetime performance for small data sets. Eur. J. Oper. Res. 230, 290–298.
Li, D.-C., Wen, I.H., 2014. A genetic algorithm-based virtual sample generation technique to improve small data set learning. Neurocomputing 143, 222–230.
Li, D.-C., Chen, C.-C., Chang, C.-J., Lin, W.-K., 2012. A tree-based-trend-diffusion prediction procedure for small sample sets in the early stages of manufacturing systems. Expert Syst. Appl. 39, 1575–1581.
Li, D.-C., Hsu, H.-C., Tsai, T.-I., Lu, T.-J., Hu, S.C., 2007. A new method to help diagnose cancers for small sample size. Expert Syst. Appl. 33, 420–424.
Liao, T. Warren, 2011. Diagnosis of bladder cancers with small sample size via feature selection. Expert Syst. Appl. 38, 4649–4654.
Lin, Y.S., Li, D.C., 2010. The generalized-trend-diffusion modeling algorithm for small data sets in the early stages of manufacturing systems. Eur. J. Oper. Res. 207, 121–130.
Marini, F., Walczak, B., 2015. Particle swarm optimization (PSO). Tutor. Chemom. Intell. Lab. Syst. 149, 153–165.
Niyogi, P., Girosi, F., Poggio, T., 1998. Incorporating prior information in machine learning by creating virtual examples. Proc. IEEE 86, 2196–2209.
Payan, S., Farahmand, A., Sarvari, S.M. Hosseini, 2015. Inverse boundary design radiation problem with radiative equilibrium in combustion enclosures with PSO algorithm. Int. Commun. Heat. Mass Transf. 68, 150–157.
Perez, R.E., Behdinan, K., 2007. Particle swarm approach for structural design optimization. Comput. Struct. 85, 1579–1588.
Sadeghierad, M., Darabi, A., Lesani, H., Monsef, H., 2010. Optimal design of the generator of microturbine using genetic algorithm and PSO. Int. J. Electr. Power Energy Syst. 32, 804–808.
Tsai, T.I., Li, D.C., 2008a. Approximate modeling for high order non.-linear functions using small sample sets. Expert Syst. Appl. 34, 564–569.
Tsai, T.I., Li, D.C., 2008b. Utilize bootstrap in small data set learning for pilot run modeling of manufacturing systems. Expert Syst. Appl. 35, 1293–1300.
Wang, Y.Q., Wang, Z.Y., Sun, J.Y., Zhang, J.J., Mourelatos, Z., 2014. Gray bootstrap method for estimating frequency-varying random vibration signals with small samples. Chin. J. Aeronaut. 27, 383–389.
Yang, J., Yu, X., Xie, Z.-Q., Zhang, J.-P., 2011. A novel virtual sample generation method based on Gaussian distribution. Knowl.-based Syst. 24, 740–748.
Zhang, H., Hui, Q., 2013. Parallel multiagent coordination optimization algorithm: implementation. Eval. Appl..
Zhao, J., Zhou, R., 2015. Particle swarm optimization applied to hypersonic reentry trajectories. Chin. J. Aeronaut. 28, 822–831.
Zhu, B., Chen, Z.S., Le, Y., 2016. A novel mega-trend-diffusion for small sample. CIESC Journal 67, 820–826.
Zhu, J.F., Shu, Y.D., Zhao, J.S., Yang, F., 2014. A dynamic alarm management strategy for chemical process transitions. J. Loss Prev. Process Ind. 30, 207–218.