Expert Systems with Applications 39 (2012) 7004–7014

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

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

A data clustering algorithm for stratified data partitioning in artificial neural network

Ajit K. Sahoo[a], Ming J. Zuo[a,*], M.K. Tiwari[b]

[a] Department of Mechanical Engineering, University of Alberta, Edmonton, Canada
[b] Department of Industrial Engineering and Management, Indian Institute of Technology, Kharagpur, India

ARTICLE INFO

Keywords:
Artificial neural network
Data partitioning
Data clustering
Self organizing map
Fuzzy clustering
Genetic algorithm
Custom design clustering algorithm

ABSTRACT

The statistical properties of training, validation and test data play an important role in assuring optimal performance in artificial neural networks (ANNs). Researchers have proposed optimized data partitioning (ODP) and stratified data partitioning (SDP) methods to partition of input data into training, validation and test datasets. ODP methods based on genetic algorithm (GA) are computationally expensive as the random search space can be in the power of twenty or more for an average sized dataset. For SDP methods, clustering algorithms such as self organizing map (SOM) and fuzzy clustering (FC) are used to form strata. It is assumed that data points in any individual stratum are in close statistical agreement. Reported clustering algorithms are designed to form natural clusters. In the case of large multivariate datasets, some of these natural clusters can be big enough such that the furthest data vectors are statistically far away from the mean. Further, these algorithms are computationally expensive as well. We propose a custom design clustering algorithm (CDCA) to overcome these shortcomings. Comparisons are made using three benchmark case studies, one each from classification, function approximation and prediction domains. The proposed CDCA data partitioning method is evaluated in comparison with SOM, FC and GA based data partitioning methods. It is found that the CDCA data partitioning method not only perform well but also reduces the average CPU time.

1. Introduction

Artificial neural networks (ANNs) are considered as one of the most widely reported data driven techniques in the last couple of decades. The performance of ANN model depends on quality of input data and network parameters. However, most of the published researches are exclusively focused on design and implementation of ANN models (Yu, Wang, & Lai, 2007). The process in which the raw experimental data is transformed into quality input data is described as data preparation or preprocessing. Data preparation in ANN mainly comprises three steps: (1) data cleaning which includes removing incomplete data, outliers and random noise; (2) input and output feature selection; and (3) data partitioning into three sub-groups, namely, training, validation and test datasets, in predefined proportions (De Noord, 1994; Joo, Choi, & Park, 2000; Nedeljkovic & Milosavljevic, 1992; Nguyen & Chan, 2004; Sjoberg, 1992; Stein, 1993a, 1993b). Yu et al. (2007) observe that 50–70% of the time and effort is spent in data preparation in complex data analysis projects. Although there is a significant amount of study on the first two stages (Chen, Sugi, Shirakawa, Zou, & Nakamura, 2009; He, Huang, Zeng, & Lu, 2008; Nguyen & Torre, 2010; Park, Shin, & Jang, 2010; Xu, Zhang, & Yang, 2010; Yen & Lin, 2000; Yoon & Bae, 2010; Zhang & Sun, 2008), the third stage has not received adequate attention. This paper focuses on the third stage, namely, data partitioning.

Data partitioning (DP) is also known as data splitting (May, Maier, & Dandy, 2010) or data division (Bowden, Maier, & Dandy, 2002; Samanta, Bandopadhyay, Ganguli, & Dutta, 2004a; Samanta, Bandopadhyay, & Ganguli, 2004b; Shahin, Maier, & Jaksa, 2004) in the literature. It constitutes the activities in which input dataset is sampled into three sub-groups called training, validation and test dataset. Training dataset is used to train the network; validation data is used to prevent over-fitting (cross-validation); and test data measures the accuracy of the network. ANNs perform best when they do not extrapolate beyond the extreme values of the data used for training (Minns & Hall, 1996; Tokar & Johnson, 1999). If the test data contains data samples that are beyond the range of training data, the model cannot be expected to perform well.

When cross-validation is used as the stopping criterion, validation data have to be in statistical agreement with the training data to ensure optimal learning. In the case of arbitrary data sampling, it is not possible to guarantee this agreement among the sub-groups.

* Corresponding author.
E-mail addresses: sahoo@ualberta.ca (A.K. Sahoo), ming.zuo@ualberta.ca (M.J. Zuo), mkt09@hotmail.com (M.K. Tiwari).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
doi:10.1016/j.eswa.2012.01.047

Nomenclature

Notation	Definition
d	territory size or the Euclidian distance between the representative object n_k and its boundary where k is the kth cluster
$\Re^l$	l-dimensional Euclidian space
n_k	representative object of cluster k, where $k = 1,2,\ldots,m$ and m is the number of clusters
$\mathbf{x}^i$	data point i, where $i = 1,2,\ldots,r$, where r is total number of data points in the entire dataset
$\mathbf{w}_k^{(q)}$	qth weight vector of representative object n_k, where $0 \leqslant q \leqslant p$ and p is the total number of data points in cluster k
t	updating coefficient
$\tilde{s}$	average silhouette width
SC	silhouette coefficient or maximum average silhouette width
$s(\mathbf{x}^i)$	silhouette value of data point $\mathbf{x}^i$
$a(\mathbf{x}^i)$	average Euclidian distance of data point $\mathbf{x}^i$ to all other data points of the same cluster
$b(\mathbf{x}^i)$	lowest average inter-cluster Euclidian distance of data point $\mathbf{x}^i$
$d(\mathbf{x}^i)$	average inter-cluster Euclidian distance of data point $\mathbf{x}^i$ from any cluster k

Maier and Dandy (2000) found that in most of the cases, the input data is divided on an arbitrary basis, without any consideration given to their statistical properties. Statistical properties such as mean and standard deviation of the three sub-groups must be close enough to ensure that each sub-group represents the same statistical population.

In recent years, researchers have made successful attempts to address the issue of efficient data partitioning using optimized data partitioning (ODP) based on genetic algorithm (GA) (Shahin et al., 2004). Stratified data partitioning (SDP) based on self organizing map (SOM) (May et al., 2010; Samanta et al., 2004a) and fuzzy clustering (FC) (Bowden et al., 2002) are also proposed. ODP based on GA may not be suitable for large datasets since the total number of combination of data split can be a gigantic task to explore. For an input dataset only having 60 data points to be divided into 40 training, 10 validation, 10 test datasets, there will be

$$\frac{60!}{40!10!10!} = 7.7 \times 10^{20}$$

ways of arranging the data points. In the case of SDPs based on SOM and FC, strata are formed using the corresponding clustering algorithm in Step 1 shown in Fig. 1. It is assumed that data points in any individual stratum are in close statistical agreement. In Step 2 data points are sampled into three sub-groups from each of these clusters. Allocation rules like equal allocation (Bowden et al., 2002), proportional allocation (May et al., 2010) and Neyman allocation (Cochran, 1977; May et al., 2010) are employed in Step 2. Although the SDPs have better performance and are faster as compared to ODP, they can be computationally expensive too. This is because SOM and FC are based on the principle of iterative learning and optimization. More descriptions on these algorithms are given in Hagan, Demuth, and Beale (1996), Bezdec (1981) respectively.

Standard classification or clustering algorithms like SOM (Kohonen, 2001), fuzzy clustering (Bezdec, 1981; Kaufman & Rousseeuw, 1990), K-mean (MacQueen, 1967), and vector quantization (Linde, Buzo, & Gray, 1980) are designed to form natural clusters at their best performance. The stratified data space in Fig. 1 illustrates natural clusters. Some of these natural clusters can be big in the case of large multivariate datasets, such that the extreme data points are statistically far away from the mean. This invalidates the assumptions made in Step 1 that data points are in close statistical agreement inside the stratum.

The goal of this paper is to report a custom design clustering algorithm (CDCA) taking the following facts into account: (1) to obtain customized clusters having close statistical agreement; (2) to work faster in terms of CPU time as compared to its competitors like SOM and FC; (3) to successfully integrate CDCA in the overall data partitioning process; and (4) to have better performance relative to the reported DP schemes. A comparative analysis is made with the reported DP schemes including GA, SOM and FC. Three benchmark datasets are selected to validate the proposed methodology. For function approximation, (1) Friedman regression function from May et al. (2010); for prediction, (2) housing dataset from Blake and Merz (1998); for classification, and (3) ultrasonic scanning data from Sahoo, Zhang, and Zuo (2008); is considered.

The remainder of the paper is organized as follows: in Section 2 we explain the CDCA based data partitioning scheme which includes the proposed CDCA data clustering algorithm. Section 3

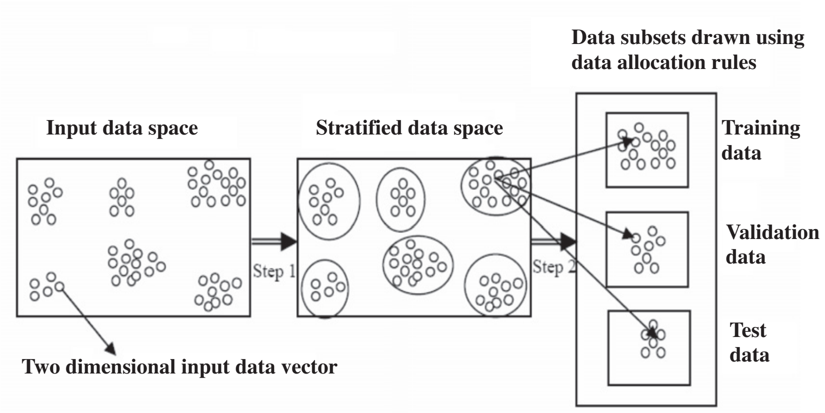

Fig. 1. Schematic diagram of SDP data partitioning process.