Journal of Intelligent & Fuzzy Systems 23 (2012) 297–304
DOI:10.3233/IFS-2012-0521
IOS Press

On sampling strategies for small and continuous data with the modeling of genetic programming and adaptive neuro-fuzzy inference system

S. Sen[a,*], E.A. Sezer[a], C. Gokceoglu[b] and S. Yagiz[c]
[a]*Department of Computer Engineering, Hacettepe University, Ankara, Turkey*
[b]*Department of Geological Engineering, Hacettepe University, Ankara, Turkey*
[c]*Department of Geological Engineering, Pamukkale University, Denizli, Turkey*

Abstract. Sampling strategies which have very significant role on examining data characteristics (i.e. imbalanced, small, exhaustive) have been discussed in the literature for the last couple decades. In this study, the sampling problem encountered on small and continuous data sets is examined. Sampling with measured data by employing k-fold cross validation, and sampling with synthetic data generated by fuzzy c-means clustering are applied, and then the performances of genetic programming (GP) and adaptive neuro fuzzy inference system (ANFIS) on these data sets are discussed. Concluding remarks are that when the experimental results are considered, fuzzy c-means based synthetic sampling is more successful than k-fold cross validation while modeling small and continous data sets with ANFIS and GP, so it can be proposed for these type of data sets. Additionally, ANFIS shows slightly better performance than GP when sytnthetic data is employed, but GP is less sensitive to data set and produces ouputs that are narrower range than ANFIS's outputs while k-fold cross validation is employed.

Keywords: Sampling strategies, small and continuous data, genetic programming, adaptive neuro-fuzzy inference system

1. Introduction

Prediction and classification are well-known objectives of data mining, soft computing and machine learning areas. Both objectives generally require collection of data, selection of train and test samples from the data, building a model and, training and testing of the model. In the literature, there are many methods employed for prediction and/or classification, and some of them (e.g. artificial neural networks, support vector machines, decision trees, genetic programming and neuro fuzzy systems) have much more applications in many complex domains such as medicine, earth science, economics, and the like. However the characteristics of data used in these methods (or more) may affect the performance of the model created. For example, Yen and Lee [34] highlight the problem of imbalanced data set and present that many applications such as fraud detection, intrusion prevention, risk management, medical research often have the imbalanced class distribution [34]. Imbalanced data set problem occurs when a class has much more samples than another class. However, modeling methods usually expect an uniform distribution of data. To overcome this problem some under and oversampling strategies are listed in Liao [20]. While imbalanced data deal with the

*Corresponding author. S. Sen, Department of Computer Engineering, Hacettepe University, Ankara, Turkey. E-mails: ssen@cs.hacettepe.edu.tr (S. Sen), ebru@hacettepe.edu.tr (E.A. Sezer), candan_gokceoglu@yahoo.co.uk (C. Gokceoglu) and syagiz@pau.edu.tr (S. Yagiz).

1064-1246/12/$27.50 © 2012 – IOS Press and the authors. All rights reserved

minority of some classes in a data set, this study focuses on the minority of the whole data, in other words, modeling with the small-sized data sets. Because the availability of data sets for solving relevant problems and the development of a prediction model are the major obstacles in many research areas. To obtain widespread data sets could be difficult or almost impossible due to many reasons such as environmental factors, cost, data security, and the like.

Vladimir [31] states a sample size as small if the ratio of the number of training samples to the Vapnik-Chervonenkis dimensions (VC dimensions) of a learning machine function is less than 20 [31]. Furthermore, Chao et al. [10] emphasize that theories focus on general machine learning with a large number of training samples, cannot be applied to practical cases with the small data set. In reality the continuity characteristic of small data sets may cause some problems due to overlapping ranges of input variables which result in different output values. Because, this situation increases the VC dimension of the learning function, and so small-sized train samples become insufficient for learning process. Expansion of data set with some synthetic data is a promising approach to solve this problem. This approach is similar with the oversampling methods employed in imbalanced problem. Bootstrapping [15] which means re-sampling of data set with replacement is utilized in some studies [15, 17, 30] to produce synthetic data. In fact bootstrapping is based on the assumption that the available data set is a particular manifestation of some unknown probability distribution [29]. This assumption fits on the problem of small data set for modeling.

In many research areas different techniques are employed in order to generate synthetic data. In some cases they are used to generate test data which meets specific requirements that may not be exist in the original data. In many other cases, they are employed to generate balanced training data for artificial intelligence based approaches. There are many applications in computer security research area that use synthetic data [5, 22]. It is usually generated by simulating user profiles and the system. The aim is to generate a test data which include some key properties and attacks not available in the original data. Using synthetic/partially synthetic data because of privacy issues are also discussed in some other researches [1]. Software testing is also a promising area that automate software testing in order to reduce the high cost of manual software testing and increase the reliability of the testing process [14]. The reader may refer to [14] for a detailed review of test data generation in software engineering. Evolutionary computation techniques have also been used extensively for test data generation in software engineering [24]. Fuzzy c-regression technique is recently proposed for preserving privacy as well [34]. There are also other techniques that are employed in order to generate synthetic data. A popular generation algorithm based on statistical techniques is given in [2]. In [9] how random forests can be adapted to generate partially synthetic data for categorical variables is discussed. A recent research which generates multi-dimensional data with specific visual properties is proposed in [3]. Other techniques such as based on the posterior predictive distribution [27], data complexity [23], support vector machines [12], and many others are also exist in the literature.

In this study, the production of synthetic data by using real data set is applied and, the usage of fuzzy c-means clustering technique for this purpose is investigated. To obtain the aim, the performance of genetic programming and adaptive neuro-fuzzy inference system (ANFIS) on small and continuous dataset, and also on synthetic dataset generated by fuzzy c-means (FCM) clustering are evaluated and compared.

In this research, rock brittleness data set compiled by Yagiz [32] is used to explore these issues, since obtaining the dataset from nature is rather difficult and limited for such engineering projects. The same data was used by Yagiz and Gokceoglu [33] to construct a Mamdani fuzzy inference system for predicting rock britleness. Due to the availability and the acceptability of the data set in the literature, sampling strategies for small and continuous data is examined using this data, and then the reliability of the developed models on it is discussed herein.

2. Data source and data structure

Rock brittleness that is the combination of rock properties rather than only simple properties is one of the main researches for earth science engineers. In this research, we use a data set obtained from 48 different tunnel sites most of which excavated in the USA [32]. Utilized dataset composed of various rock types including sedimentary (17 tunnel cases), metamorphic (15 cases), and igneous (16 cases) rocks. Further, the data comprises density, compressive and tensile strengths as well as brittleness of selected rock types as given in Table 1.

In this study, the obtained dataset composed of only 48 cases was utilized for both training and testing

Table 1
Descriptive statistics of utilized database [32]

Variables	Minimum	Maximum	Mean	Std. Deviation
σ_c	9.50	327.00	126.39	70.26
σ_t	2.30	17.80	7.815	3.411
ρ_r	20.51	28.90	25.47	2.124
BI_m	10.00	45.00	27.45	9.413

purposes. Solving encountered problem by small and continuous dataset may not be possible with artificial intelligence based techniques which require wide dataset rather than limited one in order to generate reliable models. So, "how to analyze and develop prediction models with small and continuous dataset" is one of the considerable problems in natural sciences and some engineering disciplines like rock mechanics and tunneling. As the dataset is small and continuous, some overlaps may usually occur among the range of inputs in the dataset, so this makes problem even more complicated and so, difficult. That is the case herein to be solved via some further modeling techniques. In this study, two problem solving techniques including k-fold cross validation and FCM are used to deal with such matter. Later on, these techniques are evaluated and two models, namely ANFIS and GP are developed for predicting rock brittleness.

2.1. K-fold cross validation

In real applications we usually encounter with small sample sizes and high dimensional data. To evaluate our model realistically we usually partition this limited data into two sets: training and testing, instead of using all data for training purposes. In some approaches, another data set namely validation data is also used for tuning the parameters of a classifier, however it is out of scope here.

Many re-sampling methods are proposed and evaluated in the literature so far. Holdout is the simplest technique which uses a part of the data for training and the rest for testing. One of the most popular technique is cross validation. The simplest type of cross-validation is random sub-sampling in which the sample is divided into k subsamples. In each subsample, a fixed number of test cases where the model is evaluated on is picked randomly. Then the average of the k separate estimates are used. However with this technique some part of the data may not be used, or may be used more than once for testing. k-fold cross validation overcome this issue by using all samples for both training and testing. Another approach leave-one-out cross validation is a version of k-fold cross validation in which k is the sample size. Hence in each subset only one example is employed for testing. Other variations of cross validation is also exist in the literature.

k-fold cross validation has been employed to many different research areas from medicine to finance [11, 21, 28] so far. It has also been employed with various artificial intelligence techniques such as support vector machines, decision trees, neural networks, and the like.

In k-fold cross validation, the dataset is divided randomly into k different training and test sets. In model development, a part of the dataset is picked randomly for training and the rest of it is employed for testing. Therefore, a training algorithm is run on each training dataset and, the result of the algorithm is evaluated k times. Afterwards, these obtained k results from the folds can be averaged to produce a single output. In practice, the choice of the number of folds depends on the size of the dataset. While small k size is generally good enough for large datasets, common practice is to use 5 or 10 fold cross validation. In this study, we utilize 5-fold cross for solving the problem by using 80% of the data as training and 20% of it as testing at each fold.

2.2. Fuzzy clustering

Fuzzy c-means (FCM) algorithm is first improved by Bezdek [6], and it reflects the properties of fuzzy logic. It is an unsupervised learning algorithm being applicable to many research areas. Fuzzy c-means is very similar to k-means clustering. In this method, each data point can be member of more than one cluster with different membership degrees. In other words, membership degrees of inner points of the clusters are higher than the degrees of outer points in the clusters. In fuzzy c-means, the centroids of clusters correspond to the mean of all points which are weighted according to their membership degrees. In FCM, $\mu_{Si}(x_j)$ denotes the membership degree of j^{th} case x to the i^{th} cluster S and sum of membership degree of each cases to the fuzzy clusters is equal to 1. The general algorithm of the FCM is given below.

choose number of clusters
initialize fuzzy clusters randomly
do
  compute centroids of each cluster
  compute membership degrees of each case
while *convergence criteria is not satisfied*

Computation of cluster centroids are implemented with the use of Equation 1 where *N* is the number of

cases, x_j is the j^{th} case, c_i is the centroid of i^{th} cluster and m is the constant for the adjustment of fuzziness degree of the clustering.

$$ci = \frac{\sum_{j=1}^{N} \left(\mu_{s_i}(x_j)\right)^m x_j}{\sum_{j=1}^{N} \left(\mu_{s_i}(x_j)\right)^m} \quad (1)$$

The value of m is advised as 2 and in the range of [1.5, 2] in [7]. The membership degree of j^{th} case to c_i fuzzy cluster is calculated by Equation 2 where C is the number of centroids.

$$\mu_{s_i}(x_j) = \frac{1}{\sum_{k=1}^{C} \left(\frac{||x_j - c_i||}{||x_j - c_k||}\right)^{2/(m-1)}} \quad (2)$$

The convergenge criteria which should be met to stop FCM clustering is the ε value denoting the change in the membership degrees of cases to fuzzy clusters. In other words, FCM clustering stops when the change in all membership degrees of cases to the clusters are less than the specified ε value. In general definition of FCM, euclidian distance measure is used, but replacement of it with another distance measurement method is possible.

In this study, FCM is used to produce train data set for solving a problem with small and continuous data set. The reason for the selection of this method is to produce train data set which contains meaningful synthetic data. In the literature, most of the supervised methods use 80% of the original data as the training purpose. In small sets, 80% of the data is nearly original data, there is a very little part of the data that may be used for testing (i.e. as original dataset size is 20, test set includes only 4 cases). For this reason, using synthetic data which can represent the input space and testing the model with the original data seems more plausible than the very small data to examine the performance of the model. In addition, when the continuity characteristic of the data is considered, producing synthetic data with the fuzzy approach is more suitable than the crisp approach because of the interpretation of the distances between the cases and the centroids. FCM is implemented with MATLAB R2009a (version 7.8.0.347). FCM requires the specification of the cluster number. The number of clusters are specified and used as 12, 24 and 36 in this study. The reason for this type of specification is to generate the train set with size of 25%, 50% and 75% of the whole data size. The centroids of the clusters are used as train datasets with the size of 12, 24, 36 cases, respectively. As a result of using synthetic data for training, entire established original data set is considered for testing.

3. Adaptive neuro fuzzy interference model

Adaptive neuro fuzzy inference system (ANFIS) which is an attractive modeling method for complex problems that cannot be solved with the binary logic and require uncertainty handling is presented by Jang [18]. ANFIS takes more than one input with the specification of their fuzzy set numbers and produce only one output. It is a supervised method that tries to combine the advantages of fuzzy interference system (FIS) and artificial neural networks (ANN) methods. In other words, it takes the expert knowledge from the user (lingustic variables, number of fuzzy sets), generates rules automatically within the first order Sugeno Type (if x is A and y is B than $z = mx + ny + k$) and adjusts the rule weights and ranges of fuzzy sets by training and learning. This approah requires less expert knowledge than fuzzy inference system and tries to fill lack of expert knowledge by learning from data. In other words, ANFIS is a special type ANN model which aims to learn ranges of fuzzy sets and coefficient of function placed in the consequent part of the rules.

An ANFIS model uses a hybrid learning algorithm that combines the least squares estimator and the gradient descent method [18]. In each epoch, least squares estimator is employed in forward pass, and gradient descent method is employed in backward pass. In backward pass, ranges of parameters of membership function is adjusted and, in forward pass, coefficients in the polynomial expression are tuned. Negnevitsky [25] presents that when the input-output data set is relatively small, membership fuctions can be described by a human expert. Training of the model continues until the stopping criteria is met or desired epoch numbered is reached. A typical ANFIS architecture consists of six layers and their responsibilities are distinct from each other [8]. The computation steps of the model process as follows:

- Layer 0: it presents the inputs to the layer 1.
- Layer 1: it fuzzifies the inputs according to the selected membership function and mostly bell type MF is employed however it is optional. In this study, bell shaped function is used.
- Layer 2: it includes specific node for each rule and each node calculates the firing strength of the associated rule.

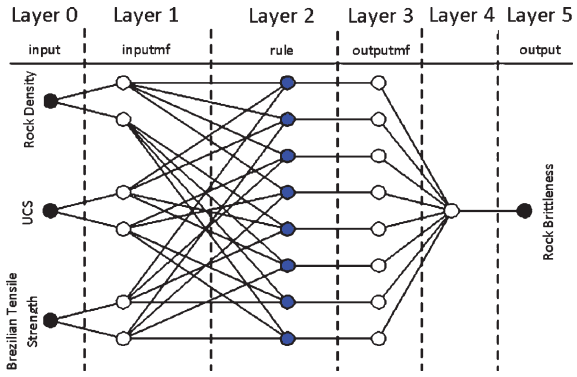

Fig. 1. The structure of implemented ANFIS.

- Layer 3: it normalizes the firing strengths of the each rule coming from the layer 2.
- Layer 4: it calculates the contribution of each rule toward the overall output.
- Layer 5: it contains only one node and computes the output.

In this study, three input parameters are used and Fig. 1 illustrates the whole architecture of the ANFIS model employed in this study. Each input parameter has 2 fuzzy sets such as low and high, and 8 Sugeno type rules are extracted. Hybrid learning is utilized with 100 epochs for each training process. The number of 100 is tested visually on the learning rate graphic of the training phase and overfitting is not observed. ANFIS is implemented with MATLAB R2009a.

4. Genetic Programming model

Genetic Programming (GP) is one of the most widely used evolutionary computation techniques in the literature. It has been named and popularised significantly by Koza [19], since then it has been employed to many problems in industry and academia. Even though GP has not reached the popularity of other machine learning methods (support vector machines, artificial neural networks, etc.), it has been shown to exceed the performance these methods in many applications [26]. It is also claimed that GP has evolved better programs than the best programs written by people in many applications [4].

GP is inspired by biological evolution. It is loosely based on the process of Darwinian survival of the fittest, where individuals are competing with each other for survival and reproduction in an environment that can only host limited number of individuals [16]. Evolutionary computation techniques, including GP, uses this aproach to solve hard problems automatically where candidate solutions correspond to the individuals, and the best solutions correspond to the fittest individuals in a population. GP tries to evolve better solutions by employing genetic operators (such as selection, mutation, crossover) iteratively until a termination creterion is satisfied. The general steps in evolutionary computation are outlined below.

initialize population
while *termination criterion not satisfied* ***do***
 execute and evaluate fitness value of each individual
 apply genetic operators to the individuals
 create new population
end while

Table 2
GP parameter settings

Objective	Find a regression model to estimate rock brittleness
Terminal set	Rock density, Uniaxial compressive strength, Brazilian tensile strength, and Constant values
Function set	+, *, −, /, sin, cos, log, exp
Population size	200
Generations	2000
Crossover probability	0.9
Reproduction probability	0.1

In this research, the usage of GP to predict the rock brittlenes from available dataset automatically is examined. The aim of this study is to find a mathematical expression representing the relationship between the rock brittleness, and the relevant rock parameters including density, uniaxial compressive and Brazilian tensile strength of rock. ECJ 19 toolkit [13] is used for GP implementation. GP parameters used in this research are given in Table 2. The parameters not listed here are the default parameters of the toolkit. The fitness function is very important in GP, since it evaluates how well individuals solve the problem. The fitness function used in these experiments is the sum of the absolute error that is the difference between the measured and the predicted values for each cases in the training dataset. In each iteration, GP algorithm tries to minimize the fitness function that is defined as:

$$\text{FITNESS} = \sum |\text{measured rock brittleness} - \text{predicted rock brittleness}| \quad (3)$$

GP algorithm is performed thirty times for each training dataset separately in order to generate the best accurate mathematical expression to estimate rock brittleness from the relevant rock features. The best result among thirty runs is chosen. The same technique is also employed to the synthetic training data generated by FCM. Afterwards, the performance of the evolved best expressions is analysed and discussed herein.

5. Analysis of the results

Metrics coefficient of determination (R^2), variance account for (VAF) and mean square error (RMSE) are used to analyze cross-correlations between measured and predicted rock brittleness values. R^2 shows how well the model represents the data. If R^2 is 1, the model would be able to explain all variation. If VAF is 100 and RMSE is 0, the model accomplishes the perfect result fit.

The performances of both techniques ANFIS and GP, trained on the data sets created by 5-fold cross validation and synthetic data produced with FCM clustering are presented in Tables 3 and 4 respectively.

Table 3
The performance of anfis with k-fold and FCM

ANFIS	Test			Train		
	R^2	*VAF*	*RMSE*	R^2	*VAF*	*RMSE*
Train-1	0.83	81.95	3.70	0.91	90.70	2.84
Train-2	0.91	89.18	3.24	0.90	89.72	2.90
Train-3	0.96	96.15	2.74	0.90	89.69	2.97
Train-4	0.94	93.84	3.58	0.89	89.40	2.89
Train-5	0.78	69.29	5.56	0.91	91.05	2.72
AVG. K-fold	**0.88**	**86.08**	**3.76**	**0.90**	**90.11**	**2.87**
Set-12	0.87	86.63	3.41	0.99	99.42	0.70
Set-24	0.90	87.78	3.28	0.96	96.68	1.77
Set-38	0.88	89.72	2.99	0.94	93.54	2.46
AVG. FCM	**0.88**	**88.04**	**3.23**	**0.96**	**96.55**	**1.64**

Table 4
The performance of gp with k-fold and FCM

GP	Test			Train		
	R^2	*VAF*	*RMSE*	R^2	*VAF*	*RMSE*
Train-1	0.84	86.29	3.31	0.94	94.48	2.24
Train-2	0.87	81.79	4.18	0.94	94.06	2.25
Train-3	0.87	86.65	4.83	0.97	97.28	1.58
Train-4	0.88	87.93	4.20	0.93	93.38	2.39
Train-5	0.90	94.34	3.19	0.85	87.36	3.39
AVG. K-fold	**0.87**	**87.40**	**3.94**	**0.93**	**93.31**	**2.37**
Set-12	0.81	86.29	3.53	0.98	97.99	1.32
Set-24	0.86	80.53	4.15	0.96	96.03	1.78
Set-38	0.90	90.33	2.90	0.96	96.41	1.83
AVG. FCM	**0.86**	**85.72**	**3.53**	**0.97**	**96.81**	**1.65**

When the performances of ANFIS and GP trained on real data set produced by 5-fold are compared, it could be seen that there is slightly difference between obtained average values. However, ANFIS gives fairly better results of R^2 and RMSE than GP model. Moreover ANFIS results measured in testing and training phases are more close to each other than GP results. It could be stated that the capability of ANFIS for prediction is higher than GP in general overview. However, this statement may not be right because ANFIS is comparably more affected from sampling ability of training data. Minimum-maximum performance ranges of ANFIS and GP for each training dataset are given in Tables 3 and 4.

As the performances of ANFIS and GP trained on synthetic data set are compared, the ANFIS model shows better performance result. It is the effect of FCM based sampling which increases the quality of samples in ANFIS model. FCM reflects the fuzziness of the input space and adjust the centroids by taking into account of the fuzziness. This result can be clearly seen in Table 4 considering average values of k-fold cross validation and FCM. In addition, the minimum-maximum performance range of ANFIS become relatively narrow in FCM based sampling. As a result of this findings, FCM based sampling can be suggested for ANFIS as processing small and continuous data sets.

For GP, the performance of the model gets better when the training data size increases. In fact, it is a natural behavior of any machine learning technique. When we increase the training data size and range, the model could have a better understanding of the system. So, the best model is obtained by using 38 cases as training data herein.

6. Conclusion

The prediction approaches to estimate unknowns from known parameters become very common in the literature. Establishment of perfect and widespread data set is usually not possible in engineering practice. As a result of the limited data, the prediction ability of these algorithms should be taken into account. In this study, sampling strategies for small and continuous data to develop two models, namely Genetic Programming and ANFIS are discussed. When small data sets divided into two (train-test) or three data sets (train-test-validation),

the size of these sets becomes very small. Furthermore when the data considered is continuous, the size of the data set may be inadeqaute to train and test the model properly due to the overlaping ranges of inputs that produce different outputs. The rock brittleness data set in the literature [32] are used as an exemplar data herein. In this study, the production of sythetic data from original data is aimed to overcome issues sourced from a small and continuous data set, and FCM is employed for this purpose. Additionally, 5-fold cross validation is implemented to make a comparison with FCM. The performance of GP and ANFIS modeling techniques on the data sets obtained from FCM based synthetic data production and 5 fold cross validation are examined. The results are concluded that; when the dataset is small and continuous, using fuzzy c-means clustering to produce sythetic data is more effective with ANFIS and GP techniques than 5 fold cross validation. Addionally, ANFIS has a bit more success than GP when FCM based synthetic data is used. When the results obtained from 5 fold cross validation is considered stand alone, the performance of GP is more succesful than the ANFIS's results. Because GP is less affected by data set than ANFIS, the performance values of ANFIS has wider range than the GP's ones. To sum up, fuzzy c-means based synthetic sampling is proposed for small and continous data sets, and applied to a data set with these characteristics from the literature successfully in this study.

References

[1] J.M. Abowd and J.I. Lane, New approaches to confidentiality protection: Synthetic data, remote access and research data centers, *Privacy in Statistical Databases*, *Springer-Verlag* (2003), 282–289.

[2] R. Agrawal and R. Srikant, Fast algorithms for mining association rules, in: *Proceedings of the 20th International Conference on Very Large Data Bases* (1994), 487–499.

[3] G. Albuquerque, T. Lowe and M. Magnor, Synthetic generation of high-dimensional datasets, *IEEE Transactions on Visualization and Computer Graphics* **17**(12) (2011), 2317–2324.

[4] W. Banzhaf, P. Nordin, R.E. Keller and F.D. Francome, Genetic programming: An introduction on the automatic evolution of computer programs and its applications, *Morgan Kaufman Publishers*, 1998.

[5] E.L. Barse, H. Kvarnstrom and E. Jonsson, Synthesizing test data for fraud detection system, in: *Proceedings of Computer Security Applications Conference* (2003), 384–394.

[6] J.C. Bezdek, Pattern recognition with fuzzy objective function algorithms, *Plenum*, 1981.

[7] J. Bezdek, J. Keller, R. Krishnapuram and T. Pal, Fuzzy models and algorithms for pattern recognition and image processing, *Kluwer Academic Publishers*, 1999.

[8] E. Buyukbingol, A. Sisman, M. Akyildiz, F.N. Alparslan and A. Adejared, Adaptive Neuro-FuzzyInference System (ANFIS): A new approach to Predictivemodelling in QSAR applications: A study of Neuro-FuzzyModelling of PCP-based NMDA receptor, Antagonists, *Bioorganic & MedicalChemistry* **15** (2007), 4265–4282.

[9] G. Cailo and J.P. Reiter, Random forests for generating partially synthetic, categorical data, *Transactions on Data Privacy* **3** (2010), 27–42.

[10] G. Chao, T. Tsai, T. Lu, H. Hsu, B. Bao, W. Wu, M. Lin and T. Lu, A new approach to prediction of radiotherapy of bladder cancer cells in small dataset analysis, *Expert System with Applications* **38** (2011), 7963–7969.

[11] D. Delen, G. Walker and A. Kadam, Predicting breast cancer survivability: A comparison of three data mining methods, *Artificial Intelligence in Medicine* **34**(2) (2005), 113–127.

[12] J. Drechsler, Using Support Vector Machines for Generating Synthetic Datasets, *LNCS 6344* Springer (2011) 148–161.

[13] ECJ 19: A Java-basedEvolutionary Computation Research System [Online], Available: http://cs.gmu.edu/∼eclab/projects/ecj/.

[14] J. Edvardsson. A survey on automatic test data generation, in: *Proceedings of the 2nd Conference on Computer Science and Engineering* (1999), 21–28.

[15] B. Efron and R. Tibshirani, An introduction to the bootstrap, *Chapman and Hall*, 1993.

[16] A.E. Eiben and J.E. Smith, Introduction to evolutionary computing, *Springer*, 2003.

[17] V.C. Ivanescu, J.W.M. Bertrand, J.C. Fransoo and J.P.C. Kleijnen, Bootstrapping to solve the limited data problem in production control: An application in batch process industries, *Journal of the Operational Research Society* **57** (2006), 2–9.

[18] J.-S.R. Jang, ANFIS: Adaptive-Network Based Fuzzy Inference Systems, *IEEE Transactions on Systems, Man, Cybernetics* **23** (1993), 665–685.

[19] J.R. Koza, Genetic programming: On the programming of computers by means of natural selection, *MIT Press*, 1992.

[20] T.W. Liao, Classification of weld flaws with imbalanced class data, *Expert System with Applications* **35** (2008), 1041–1052.

[21] C.-H. Luang, M.-C. Chen and C.-J. Wang, Credict scoring with a data mining approach based on support vector machines, *Expert Systems with Applications* **33**(4) (2007), 847–856.

[22] E. Lundin, H. Kvarnstrom and E. Jonsson, A synthetic fraud data generation methodology, in: *Proceedings of ICICS, LNCS* **2513** (2002), 265–277.

[23] N. Macia, E. Bernado-Mansilla and A. Orriols-Puig, Preliminary approach on synthetic data sets generation based on class separability measure, in: *Proceeding of the 19th International Conference on Pattern Recognition*, 2008, 1–4.

[24] T. Mantere and J.T. Alander, Evolutionary software Engineering, a review, *Applied Soft Computing* **5** (2005), 315–331.

[25] M. Negnevitsky, Artificial intelligence a guide to intelligent systems, *Addison Wesley*, 2002.

[26] M. O'neill, L. Vanneschi, S. Gustafson and W. Banzhaf, Open Issues in Genetic Programming, *Genetic Programming and Evolvable Machines* **11** (2010), 339–363.

[27] J.W. Sakshaug and T.E. Raghunathan, Synthetic data for small data estimation, *LNCS 6344* Springer (2011), 162–173.

[28] S.K. Shevade and S.S. Keerthi, A simple and efficient algorithm for gene selection using sparse logistic regression, *Bioinformatics* **19**(17) (2003), 2246–2253.

[29] M. Talebizadeh and A. Moridnejad, Uncertainty analysis for the forecast of lake level fluctuations using ensembles of ANN and ANFIS models, *Expert Systems with Applications* **38** (2011), 4126–4135.

[30] T.I. Tsai and D.C. Li, Utilize bootstrap in small data set learning for pilot run modeling of manufacturing systems, *Expert Systems with Applications* **35** (2008), 293–1300.

[31] N.V. Vladimir, The nature of statistical learning theory, *Springer*, 1995.

[32] S. Yagiz, Assessment of brittleness using rock strength and density with punch penetration test, *Tunnelling and Underground Space Technology* **24** (2009), 66–74.

[33] S. Yagiz and C. Gokceoglu, Application of fuzzy inference system and nonlinear regression models for predicting rock brittleness, *Expert Systems with Applications* **37** (2010), 2265–2272.

[34] S.J. Yen and Y.S. Lee, Cluster-based under-sampling approaches for imbalanced data distributions, *Expert System with Applications* **36** (2009), 5718–5727.

Copyright of Journal of Intelligent & Fuzzy Systems is the property of IOS Press and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.