Int. J. Math. And Appl., 6(3)(2018), 215–225

ISSN: 2347-1557

Available Online: http://ijmaa.in/

International Journal of *Mathematics And its Applications*

Evolutionary Vertical Size Reduction: A novel Approach for Big Data Computing

M. Najarian[1,*], Z. Sarmast[2], S.M. Ghasemi[2] and S. Sarmadi[2]

1 Department of Industrial Engineering, University of Houston, Houston, USA.

2 Department of Mathematics, University of Houston, Houston, USA.

Abstract: Complexity of model based machine learning techniques depends mainly on the size of feature vector, so various feature selection and feature extraction methods are widely employed for size reduction. Although horizontal size reduction approaches can exponentially subside the complexity of learning algorithms, utilizing all instances to build and train any models can potentially be problematic. As illustration, not only problematic data can lead for over-fitting and over-generalization, but using noisy data can cause no proper decision makings. Furthermore, performance and complexity of instance/memory based classifiers (e.g. k-NN) are highly depends on train samples. In this paper, evolutionary vertical size reduction approach is introduced to identify and filter problematic and noisy data, not only for enhancing the performance and robustness of machine learning techniques but for ebbing the needs for remarkable system resources. To this end, genetic algorithms are proposed to label problematic instances in datasets. To evaluate the performance of proposed approach, benchmark classification datasets are employed to quantify the impacts of filtering noisy and problematic data in classification applications. Despite the fact that proposed vertical size reduction approach enhances machine learning techniques (mainly supervised learning methods), real world applications which contain big data would require remarkable system resources. To address this drawback, cloud computing frameworks (such as MapReduce on Hadoop) is recommended to make the proposed vertical size reduction more applicable for big data processing.

MSC: 62-07, 68P05.

Keywords: Big Data, Data-Mining, Gene p53 Mutant, High-Performance Computing, Instance Selection, Size Reduction.

1. Introduction

Owing to wide variety of real world applications, various classification and pattern recognition approaches have been developed which are generally sorted out into model-based and instance/memory-based categories [1]. Model-based classifiers process train data to build and tune the model parameters up such that can imitate the functionality and behavior of prior experiences properly [2, 3]. On the other hand, instance or memory-based methods take the advantages of using large amount of memory to store train data for decision making purposes [1, 4]. In some sense, model-based classifiers spend time for building the model, and memory-based classifiers spend time during the test [5]. Admittedly, model-based techniques seem to be efficient when offline processes are affordable as well as real-time decision makings [6]. On the other hand, instance-based techniques are desirable, when memory is available enough or offline process is not affordable for model training [1, 7].

In many machine learning applications, noisy instances in datasets bias the models towards the falsie behaviors. Machine learning applications are generally sorted out into three main categories: regression, classification and clustering [8, 9].

[*] *E-mail: Mnajarian@uh.edu*

From regression perspective, time series analysis are very vulnerable against the noisy records [10]. In this sense, the performance of many statistical and stochastic regression techniques highly depends on the accuracy of the observations [11, 12]. In supervised machine learning applications, all pattern recognition approaches rely on the annotations of the previous observations [9, 13]. As illustration, all techniques for training the neural networks measure the accuracy of the trained model based on the target values [9, 13]; hence, small noises in the dataset can cause major drawbacks in delivering robust and reliable predictions (or decisions) [8, 14]. In the same way, clustering applications are also very vulnerable against the noisy records [9, 12], and problematic instances can cause meaningless results [8, 15]. Acknowledging the fact that high-performance computing (HPC) can significantly accelerate machine learning tools, response time is still an issue in many real-world applications [16]. From computation complexity point of view, the less operations that we need to perform the less response time a machine learning method needs to return the results [17, 18]. In some sense, both train and test processes in many supervised and unsupervised machine learning approaches mainly depend on the size of the input dataset [19].

Not only training a model using machine learning algorithms is knotty but allocating memory and making timely decisions seems to be involute for instance-based methods [1, 20]. To reduce computational complexities and to resolve memory storage issues in large-scale corpses, sparsity is applied in extensive intimate applications [21, 22]. This feature is theoretically investigated for many problems employed in numerous contexts such as compressive sensing [23, 24]. Nowadays, real world problems come with big data, so it is difficult not only to record and maintain the records but also process the big datasets. In some sense, dealing with the volume, velocity and veracity of big data have become as the major concerns for designing and developing many applications. For example, when it comes to process high-resolution images [25, 26], or video records [27], size reduction techniques are needed to ebb the memory related issues. Similarly, data from sensors come with high velocity [28, 29], so it is crucial to filter the input data timely [30, 31].

In lights of size reduction approaches, vast various feature selection and feature extraction methods have been used to collapse feature vectors. From feature selection perspective, the dataset (matrix) is processed to find the most informative features and eliminate columns with less importance [1, 32]. However, feature extraction algorithms, take the advantages of linear and nonlinear mappings to squeeze original feature space, so the essence of features in new dataset would not include same information [1, 33]. Despite the fact that both feature selection and feature extraction approach exponentially reduce the complexity of many machine learning algorithms, using all recorded instances in a dataset (more narrowly big data) might not make sense in multitude real world applications. From another attitude, not only using all available data would increase the train and decision-making time in model-based and instance-based classifiers respectively, but it also leads to allocate substantial memory for instance-based pattern recognitions [32, 34]. Furthermore, employing illusory instances can potentially cause improper judgments [1].

This paper introduces vertical size reduction concept to perform complementary functionality for feature selection and feature extraction strategies which both horizontally squeeze the dataset matrix. Hence, genetic algorithms are used label and eliminate problematic, noisy, obscure and replicated data.

2. Evolutionary Big Data Squeezing Approach

Genetic algorithm (GA) is a population based meta-heuristic which requires no specific knowledge about the problem but generally serves desirable functionality in optimization applications [34, 35]. In this study, vertical size reduction is looked as an optimization problem in which, the objective is identifying and filtering problematic, obscure and replicated data to improve both performance and complexity of machine learning algorithm. In some sense, this size reduction problem seems to be multi-objective because, instead of a singular solution, GA oversees finding Pareto set in which, all solutions satisfy the trade-off between different objective functions optimally. It is worth noting that although there are some meta-heuristics that can outperform genetic algorithms in specific applications, they are not applicable in the proposed vertical size reduction model. For example, particle swarm optimization (PSO) has faster convergence; however, it is mainly applicable on continues objective functions [36]. Similarly, memetic algorithms can find higher quality solutions (compare to meta-heuristics such as genetic algorithms); however, due to their running time complexity, they are not recommended for big data applications [37, 38].

In this paper, GA is employed to stochastically explore and exploit the search space and label all identifiable problematic samples in a given dataset. To this end, state space of the problem is represented as binary alleles of genes. In other words, each gene includes two alleles: active or passive. In this formulation, each gene is in charge of labeling a particular instance in the dataset, and passive allele in a gene indicates that the corresponding instances is problematic (or vice versa). Consequently, genome of each particular chromosome (sequence of allele of genes) includes all information which are enough to represent a potential solution for vertical size reduction problem. From another perspective, GA tries to solve a transpose feature selection problem in which, instead of columns, rows of dataset matrix are selected/eliminated. Thus, chromosome size is determined by number of instances in the dataset.

All evolutionary algorithms need to have at least one genetic and one evolution operator. Proposed GA takes the advantages of crossover and mutation (as genetic operators) to exploit and explore the search space respectively. In addition, a selection mechanism (here tournament selection) is used as evolution operator to guarantee the convergence. Owing to the essence of machine learning problems, datasets mainly includes significant number of instances, so chromosomes are expected to be long enough to make standard crossovers inefficient. On this basis, proposed GA employs multitude cutting points for crossover. From implementation perspective, if chromosome size is N, proposed GA generates offspring chromosomes using $\frac{N}{10}+1$ random cutting point crossover. In the same way, due to large number of genes in chromosomes, filliping the value of one randomly selected gene would not be enough to prevent premature convergence. To empower the exploration characteristic of proposed GA, all genes are allowed to change their allele probabilistically. From another perspective, values of all genes are flipped with the probability of $P_{mutation}$. This probability is empirically small to push the GA back from random searching.

As an evolutionary algorithm, proposed GA is guided through fitness function toward Pareto set. In this study, fitness function is formulated to satisfy two objective functions: (1) eliminating more instances from primary dataset vertically squeeze the dataset matrix as much as possible; (2) enhancing the performance of pattern recognition from accuracy perspective. It seems that these objectives can potentially have conflict. Hence, first objective function which is in charge of conducting the population toward smaller matrices is formulated as Equation (1).

$$f_1 = \frac{n}{N} \tag{1}$$

Where, N is total number of instances in dataset and n denotes total number of genes with active allele in the chromosome. In the same way, second objective function tries to conduct the population towards more accurate pattern recognition, is formulated as Equation (2).

$$f_2 = re_{train} + (1-r)\, e_{test} \tag{2}$$

Where, r is a random variable with uniform distribution $(0 < r < 1)$, e_{train} and e_{yrsy} are train and test errors respectively. To compute these errors, a temporary dataset is created which only contains instances with active allele in the chromosome. To this end, $\frac{1}{3}$ of new dataset is excluded to create test data. Henceforth, both train and test errors are computed using a supervised classifier (such as neural networks, k-NN, etc.). According to peculiar architecture of meta-heuristics, fitness function would be computed multiple times. Consequently, instance-based classifiers are recommended for this GA to omit several model trainings-essentially, k-NN seems to fit in the proposed architecture well.

Moreover, to deal with stochastic behavior of almost machine learning algorithms (for example, and excluding test data randomly), bagging strategy is used. More preciously, after eliminating problematic data (instance with passive allele in chromosome), process of calculating train and test errors is repeated, but with different train and test sets arrangement. Thereupon, mean of both and train and test errors are used in Equation (2). Proposed GA is supposed to find optimum trade-off between minimizing the number of active alleles (eliminating more instances from dataset), and enhancing the performance of pattern recognition method (reducing classification errors). Thus, both objective functions are in minimization direction, so the fitness function for quantifying the quality of chromosomes is formulated as Equation (3).

$$f = wf_1 + (1 - w)\, f_2 \tag{3}$$

Where, w is uniform random variable $(0 < w < 1)$. It's worth noting that using initiating r in Equation (2) and w in Equation (3) would lead the GA to conduct chromosomes of the population to be converged to Pareto set; however, in this study, these parameters are fixed during the GA's iterations. In conclusion, instead of looking for Pareto set, proposed GA will be converged to only one solution on the Pareto line. As a summary, GA receives a dataset as input, assigns a gene for each instance, and tries to label problematic samples to vertically squeeze the dataset matrix.

3. Experimental Analysis

In this section, performance of proposed approaches for vertical size reduction is discussed. To confront stochastic demeanor of employed machine learning techniques in proposed architecture, and collating with randomness of proposed vertical size reduction approach (ranging from initialization to test data selection), 10-fold cross-validation is used to provide more valid and meaningful evaluation. To this end, main dataset is partitioned in to two parts (train and test sub-dataset). In each try, $\frac{1}{3}$ of data are randomly put in test set and remaining are added into train set. Train dataset is processed using proposed vertical size reduction method, and afterwards, two classifiers are trained using both primary and squeezed train set but are validated using same test set. Then, both train and test (validation) errors are measured, and finally, mean and variance of errors of all 10 folds are calculated. It's worth mentioning that partitioning the data into train and test sets in validation phase is different from calculating train and test errors in fitness function. From a macro perspective, the evaluation process takes the advantages of 10-fold cross-validation; however, the fitness function uses bagging strategy.

3.1. Experiment I

Since 1988, Iris is being used as standard benchmark to evaluate machine learning algorithms. This dataset includes 150 instances which are defined in R^4 feature space. In this experiment, 50 instances are randomly selected to create test (validation) set. Afterward, remaining instances are labeled using proposed GA to identify problematic samples. Henceforth, problematic data are filtered before training the classifier (here k-NN). In addition, to deal with stochastic behavior of proposed method, 10-fold cross-validation is employed to see how robust this approach is. Therefore, proposed GA is run 10 times over 10 randomly selected train sets. Filtered data are used as train data for a k-NN to measure the train and test error. To make the compressions more meaningful, mean and variance of both train and test errors are calculated over same train and test sets arrangements.

Figure 1 illustrates the convergence of GAs in search for problematic data inside Iris dataset. It seems that proposed GA is converged to a local optimum after few iterations. Therefore, higher mutation probability can empower the exploration in GA not to have premature convergence. Table 1 also demonstrates the minimum, average and variance of both train and test errors before and after vertical size reduction (+VSR). Table 2 also contains the information regarding the filtering rate. Statistical parameters from TABLE I and II reveal that proposed vertical size reduction approach filters about 55 percent of from train set; however, performance of k-NN is not affected. In other words, vertical size reduction on Iris dataset identify 55 percent of instances as problematic (noisy or unnecessary to use).

In addition, significant improvement in train error is observed after employing vertical size reduction, but these train errors are calculated using different train sets (original and squeezed/filtered). On this basis, comparing train errors would not provide meaningful information. Moreover, it seems that vertical size reduction reduces the robustness of classification (increase in variance). It should be highlighted that GAs only had 50 iterations, so increasing the maximum number of generations in GA is recommended to enhance robustness and stability of the classifications using filtered dataset.

3.2. Experiment II

Wine dataset is another standard benchmark which includes 178 instances that are defined in R^{13} feature space. During last two decades, this dataset has been widely used for comparing the performance of machine learning algorithms. Same as previous experiment, in each run of GA, $\frac{1}{3}$ of instances are randomly excluded from primary dataset to be employed for validation purposes. Train dataset is processed using GA to identify and filter problematic data. Figure 2 illustrates the convergence of GAs in labeling problematic data in Wine dataset.

Table 3 and 4 are explaining statistical parameters regarding impacts of using GA for vertical size reduction on Wine dataset. Considering these statistical parameters conclude that, although proposed vertical size reduction method eliminates about 80 percent of data in train set, using squeezed (filtered) set to build k-NN provides barely accurate and significantly robust classification-compare to using original train set. Surefire, it can be concluded that using all instances in Wine dataset makes no sense, and 20 percent of data are enough to demonstrate even better classification functionality. Empowering the GA (e.g. increasing the number of iterations and/or population size) can potentially provide more desirable results.

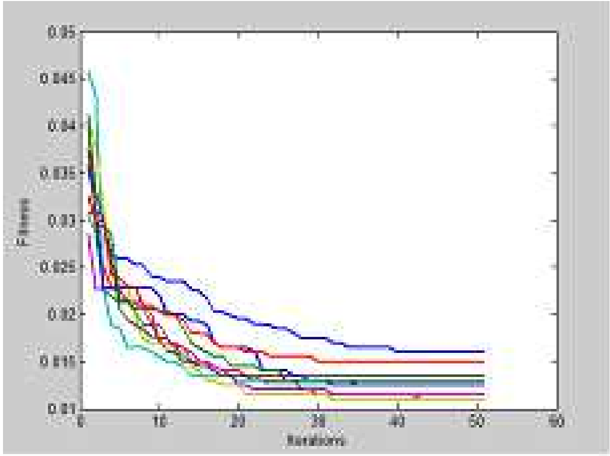

Figure 1. **Convergence of 10 GAs in filtering problematic data in Iris dataset.**

Error	min	Mean	Var
Train	0	0.039	0.00045
Train + VSR	0	0.0040	0.00007
Test	0.02	0.04	0.00062
Test + VSR	0.02	0.04	0.00089

Table 1. Statistical measurements for evaluating the performance of proposed GA in vertical size reduction (+VSR) on Iris dataset.

Dataset	Original	After VSR	Filter Ration
Iris	100	45.8	54.2 %

Table 2. Impact of using GA for vertical size reduction on Iris dataset (Comparing total number of instances in main train dataset (Original), average number of instances that are NOT labeled as problematic (After VSR) and average ratio of elimination (Ratio)).

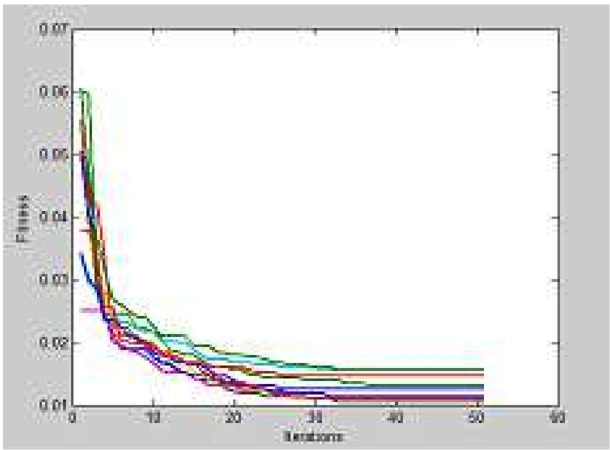

Figure 2. Convergence of 10 GAs in filtering problematic data in Wine dataset.

Error	min	Mean	Var
Train	0.0336	0.0513	0.00027
Train+VSR	0	0	0
Test	0	0.0339	0.00076
Test+VSR	0.0169	0.0336	0.00022

Table 3. Statistical measurements for evaluating the performance of proposed GA in vertical size reduction (+VSR) on Wine dataset.

Dataset	Original	After VSR	Filter Ration
Iris	119	30	74.79%

Table 4. impact of using GA for vertical size reduction on Wine dataset (Comparing total number of instances in main train dataset (Original), average number of instances that are NOT labeled as problematic (After VSR) and average ratio of elimination (Ratio)).

3.3. Experiment III

During last three decades, Glass dataset has been used as a benchmark problem for evaluating machine learning algorithms. This dataset includes 214 instances which are defined in R^9 feature space. Similar to previous experiments, $\frac{1}{3}$ of Glass dataset is excluded randomly to create test set for validating the performance of vertical size reduction. Figure 3 illustrates convergence of 10 GAs in looking for problematic data in Glass dataset. TABLE V and VI also explain the statistical parameters to evaluate performance of proposed vertical size reduction on Glass dataset. Simulation results reveal that, proposed GA vertically squeezes the Glass dataset and filters about 58 percent of instances; however, in this experiment, k-NN face improper classifications during the test. Similar to previous experiments, using squeezed train set for k-NN demonstrates more stable behavior.

3.4. Computational Complexity

Proposed GA for vertical size reduction is an iterative algorithm in which, several k-NN classifiers are built to calculate fitness values. Additionally, bagging strategy inside fitness function significantly increases complexity of the GA from CPU time perspective. On the other side, chromosomes are long enough to require outstanding memory allocation –more preciously in big data processing. To comfort this major drawback, cloud computing frameworks are recommended. As illustration, proposed method is MapReduce friendly enough to use Hadoop for distributing the needs for significant system resources (memory and CPU time) over the cluster(s). In other words, in lights of employing Hadoop framework, higher configurations of GA can be applied (e.g. more chromosomes in population, more generations of GA, using Memetic algorithms to take the advantages of heuristics beside meta-heuristics, etc.). If this is so, theoretically, more desirable can be achieved (higher filtering rates with higher performance).

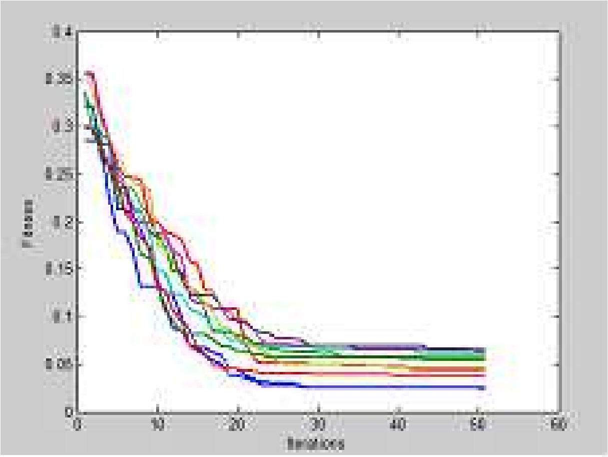

Figure 3. **Convergence of 10 GAs in filtering problematic data in Glass dataset.**

Error	min	Mean	Var
Train	0.2792	0.3483	0.00191
Train+VSR	0	0.0098	0.00007
Test	0.2535	0.3211	0.00145
Test+VSR	0.2816	0.3659	0.00243

Table 5. Statistical measurements for evaluating performance of proposed GA in vertical size reduction (+VSR) on Glass dataset.

Dataset	Original	After VSR	Ration
Glass	143	60.7	50.45%

Table 6. Impact of using GA for vertical size reduction on Glass dataset (Comparing total number of instances in main train dataset (Original), average number of instances that are NOT labeled as problematic (After VSR) and average ratio of elimination (Ratio)).

4. Squeezing Gene P53 Dataset

From biological viewpoint, cells play fundamental role in emerging living beings' behavior. From a phenotype perspective, cells collaborate with each other to demonstrate the natural functionality of an organism. And from a genotype scenery, DNA of each cell contains a predefined program in which every single action of the cell has been scheduled-for example, each knows that when it should be died. Cancer is a gene disruption-based disease in which, one (or some) genes demonstrate abnormal behavior (synthesis more or less protein than normal volume). These damaged cells are called cancerous, and a mass of cancerous cells leads tumor [39]. Breast cancer is the leading cause of mortality and morbidity among female population. Unfortunately, owing to specific anatomy of the breast, identifying malignant tumors from benign masses is still complicated –using common technologies like CT images. According to the fact that early diagnosis of malignant tumors significantly enhance the odds of surviving the patient, Thus, wide various researches have been focused on providing decision support systems for early breast cancer diagnosis applications –including but not limited to employing image processing techniques for more accurate diagnosing and using data-mining techniques for processing genomic data [39–41].

Genotype level studies have revealed that gene p53 activity is more likely to be observed in breast cancer, rather than other genes. In other words, identifying the mutation of gene p53 can significantly increase the odds of detecting malignant solid cancerous tumors. Gene p53 mutant dataset contains genetic information of about 20,000 patients. In this dataset, each instance includes 5,409 features [39]. Apparently, performing any machine learning method on this data will potentially require remarkable computational resources. According to the fact that employing size reduction methods on this dataset can provide more feasible data analytics studies, evolutionary instance selection approach was performed on p53 mutant dataset to squeeze this big dataset (vertically). There is no doubt that this study of case will require more resources (both memory and processing units). Therefore, parameters of evolutionary instance selection were retuned-up to finish the algorithm in feasible time. After running the algorithm on a multi-core i7 system, the proposed method eliminated about 13.4% of the dataset. There is no doubt that this result can potentially be enhanced using distributed computing technologies (e.g. Apache Hadoop). It is worth noting that, due to scale of this study case, cross-validation was not used for verification and validation purposes.

5. Conclusion

In this paper, the concept of vertical size reduction was introduced in order to subside the complexity of machine learning algorithms and enhance their performance. In this sense, vertical size reduction was looked as an optimization problem, so genetic algorithms (GAs) were recommended to find problematic and, obscure and noisy data. To this end, weighted linear mapping was used to combine two objective functions and provide a fitness function which tries to conduct the population toward optimum solutions. In other words, train and test errors of a k-NN classifier were used to evaluate the fitness of chromosomes. It's worth noting that, bagging strategy was utilized in calculating the fitness values. Furthermore, 10-fold cross-validation was employed to deal with stochastic behavior of proposed method in evaluation. Computational simulations were applied on some standard datasets (Iris, Wine and Glass) to show the performance of proposed vertical size reduction approach. Experimental results revealed that, although proposed GA filtered substantial portion of train data, k-NN demonstrated more accurate and more stable classification functionality-compare to using original train set.

In spite the proposed method serves desirable filtering functionality over datasets, spending outstanding memory and CPU time are major downsides of the vertical size reduction method which practically make it no applicable for real world applications which contains big data. To deal with this drawback, could computing frameworks (e.g. MapReduce on

Hadoop) was recommended to distribute required memory and computation resources over the cluster(s).

Finally, proposed big data squeezing method was performed on gene p53 mutant dataset. This data set contains genomic data about breast cancer, and can be considered as a big data problem for data analytics applications. Owing to the limitations on CPU time, the proposed evolutionary instance selection filtered about 14% of original dataset. Apparently, taking the advantages of cloud and distributed computing technologies can potentially increase this filtering ratio. Moreover, analyzing the convergence of GAs showed that, proposed GA was facing premature convergence. To prevent GA from trapping in local optimum, increasing the exploration mechanisms (such as higher mutation probability) can theoretically benefit the GA. Apparently, empowering the GA using larger population and more generations (based on cloud computing technologies) can prevent from premature convergence as well. Furthermore, employing other types of both model and memory-based pattern recognition methods to validate the proposed vertical size reduction can provide more evaluations to consider the efficiency of proposed approach. Finally, using other model-free optimization methods seems to be applicable in this particular problem. For example, reinforcement learning methods (e.g. Q-Learning) can be employed to look for problematic data, as well as learning automata and artificial immune systems.

References

[1] A. R. Webb, *Statistical pattern recognition*, John Wiley & Sons, (2003).

[2] J. Derrac, I. Triguero, S. García and F. Herrera, *A co-evolutionary framework for nearest neighbor enhancement: combining instance and feature weighting with instance selection*, In Hybrid Artificial Intelligent Systems 176-187, Springer Berlin, (2012).

[3] J. Tang and H. Liu, *CoSelect: Feature Selection with Instance Selection for Social Media Data*, In SDM, (2013), 695-703.

[4] Z. Fu and A. Robles-Kelly, *An instance selection approach to multiple instance learning*, In Computer Vision and Pattern Recognition, IEEE Conference, (2009, 911-918.

[5] J. Derrac, C. Cornelis, S. García and F. Herrera, *Enhancing evolutionary instance selection algorithms by means of fuzzy rough set based feature selection*, Information Sciences, 186(1)(2012), 73-92.

[6] K. Buza, A. Nanopoulos and L. Schmidt-Thieme, *Insight: efficient and effective instance selection for time-series classification*, In Advances in Knowledge Discovery and Data Mining 149-160, Springer Berlin, (2001).

[7] T. Jang, R. Jyh-Shing and S. Chuen-Tsai, *Neuro-fuzzy and soft computing: a computational approach to learning and machine intelligence*, Prentice-Hall, (2005).

[8] Michalski, S. Ryszard, Jaime G. Carbonell and Tom M. Mitchell, *Machine learning: An artificial intelligence approach*, Springer Science & Business Media, (2013).

[9] Robert and Christian, *Machine learning, a probabilistic perspective*, Journal Name?, (2014), 62-63.

[10] Ayanzadeh, Ramin, Azam S. Zavar Mousavi and Saeid Setayeshi, *Fossil fuel consumption prediction using emotional learning in Amygdala*, In Electrical Engineering (ICEE), 19th Iranian Conference, (2011), 1-6.

[11] Roozbeh, Mahdi and Mohammad Najarian, *Efficiency of the QR class estimator in semiparametric regression models to combat multicollinearity*, Journal of Statistical Computation and Simulation, 88(9)(2018), 1804-1825.

[12] Azencott, Robert, Viktoria Muravina, Rasoul Hekmati, Wei Zhang and Michael Paldino, *Automatic clustering in large sets of time series*, In Contributions to Partial Differential Equations and Applications, 65-75.

[13] Marzban, Forough, Ramin Ayanzadeh and Pouria Marzban. *Discrete time dynamic neural networks for predicting chaotic time series*, Journal of Artificial Intelligence, 7(1)(2014).

[14] Baghban, Alireza, Rasoul Hekmati, Mahdi Hajiali, Milad Janghorban Lariche and Majid Kamyab, *Application of MLP-*

ANN as novel tool for estimation of effect of inhibitors on asphaltene precipitation reduction, Petroleum Science and Technology, (2018), 1-6.

[15] Gholami, Ali-Asghar, Ramin Ayanzadeh and Elaheh Raisi, *Fuzzy honey bees foraging optimization: swarm intelligence approach for clustering*, Journal of Artificial Intelligence, 7(1)(2014).

[16] Mirhajianmoghadam, Hengameh, R. Mohammad, T. Akbarzadeh and Ehsan Lotfi, *A harmonic emotional neural network for non-linear system identification*, In Electrical Engineering (ICEE), 24th Iranian Conference, (2016), 1260-1265.

[17] Z. Sarmast, B. Soltanalizadeh and K. Boubaker. *A new numerical method to study a Second-order hyperbolic equation*, South Asian J. Math 6(2014), 285-96.

[18] R. Hekmati, R. Azencott, W. Zhang and M. J. Paldino. *Machine Learning to Evaluate fMRI Recordings of Brain Activity in Epileptic Patients*, q-bio Conference, (2018) June 26–29 at Rice University; Houston, TX.

[19] Z. Sarmast, Mohammad Reza Safi and Elham Farzaneh, *Post Optimality Analysis in Fuzzy Multi Objective Linear Programming*, Journal Name?, 55(2018).

[20] H. Liu, *Instance selection and construction for data mining*, Springer-Verlag, (2010).

[21] L. Kong, D. Zhang, Z. He, Q. Xiang, J. Wan and M. Tao, *Embracing big data with compressive sensing: a green approach in industrial wireless networks*, IEEE Communications Magazine, 54(10)(2016), 53-59.

[22] A. C. Gilbert, P. Indyk, M. Iwen and L. Schmidt, *Recent developments in the sparse Fourier transform: A compressed Fourier transform for big data*, IEEE Signal Processing Magazine, 31(5)(2014), 91-100.

[23] J. Shen and S. Mousavi. *Least sparsity of p-norm based optimization problems with* $p > 1$, Preprint arXiv: 1708.06055, (2017).

[24] S. Mousavi and J. Shen, *Solution uniqueness of convex piecewise affine functions based optimization with applications to constrained l_1 minimization*, Preprint arXiv: 1711.05882, (2017).

[25] Marni, Lahir, Morteza Hosseini, Jennifer Hopp, Pedram Mohseni and Tinoosh Mohsenin. *A Real-Time Wearable FPGA-based Seizure Detection Processor Using MCMC*, In Circuits and Systems (ISCAS), 2018 IEEE International Symposium, (2018), 1-4.

[26] A. Mobiny and Hien Van Nguyen, *Fast CapsNet for Lung Cancer Screening*, arXiv preprint arXiv:1806.07416, (2018).

[27] F. Shahraki, A. P. Yazdanpanah, E. E. Regentova and V. Muthukumar, *A trajectory based method of automaticcounting of cyclist in traffic video data*, International Journalon Artificial Intelligence Tools, 26(04)(2017).

[28] A. Hekmati and Rasoul Hekmati, *Optimum acoustic sensor placement for partial discharge allocation in transformers*, IET Science, Measurement & Technology, 11(5)(2017), 581-589.

[29] G. Slaughter and Zahra Ghassemi, *Dynamic Modeling of PQQ-GDH Bioanode of Self-Powered Biosensing System.*

[30] Z. Ghassemi and Gymama Slaughter, *Biological Fuel Cells and Membranes*, Membranes, 7(1)(2017).

[31] Z. Ghassemi and Gymama Slaughter, *Dynamic modeling of direct electron transfer PQQ-GDH MWCNTs bioanode function*, In Nano/Micro Engineered and Molecular Systems (NEMS), IEEE 12th International Conference, (2017), 383-387.

[32] J. R. Cano, F. Herrera and M. Lozano, *Using evolutionary algorithms as instance selection for data reduction in KDD: an experimental study*, Evolutionary Computation, IEEE Transactions, 7(6)(2003), 561-575.

[33] H. Brighton and C. Mellish, *Advances in instance selection for instance-based learning algorithms*, Data Mining and Knowledge Discovery, 6(2)(200), 153-172.

[34] M. Grochowski and N. Jankowski, *Comparison of instance selection algorithms II. Results and comments*, In Artificial Intelligence and Soft Computing, (2004), 580-585.

[35] P. Engelbrecht, *Computational intelligence: an introduction*, John Wiley & Sons, (2007).

[36] A. Khosravani-Rad, Ramin Ayanzadeh and Elaheh Raisi, *Dynamic parameters optimization for enhancing performance and stability of PSO*, Trends in Applied Sciences Research, 9(5)(2014).

[37] R. Ayanzadeh, E. Shahamatnia and S. Setayeshi, *Determining optimum queue length in computer networks by using memetic algorithms*, J. Applied Sci., 9(2009), 2847-2851.

[38] E. Shahamatnia, Ramin Ayanzadeh, Rita A. Ribeiro and Saeid Setayeshi, *Adaptive imitation scheme for memetic algorithms*, In Doctoral Conference on Computing, Electrical and Industrial Systems, (2011), 109-116.

[39] S. A. Danziger, R. Baronio, L. Ho, L. Hall, K. Salmon, G. W. Hatfield and R. H. Lathrop, *Predicting positive p53 cancer rescue regions using Most Informative Positive (MIP) active learning*, PLoS Computational Biology, 5(9)(2009).

[40] K. Barker, D. Berry and C. Rainwater, *Levelized Resiliency Assessment of Interdependent Natural Gas and Electric Power Systems*.

[41] A. Jafari, Morteza Hosseini, Adwaya Kulkarni, Chintan Patel and Tinoosh Mohsenin, *Binarized neural Network Many-core Accelerator*, In Proceedings of the 2018 on Great Lakes Symposium on VLSI, (2018), 443-446.