



ELSEVIER

Contents lists available at ScienceDirect

Engineering Applications of Artificial Intelligence

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

DENSA: An effective negative selection algorithm with flexible boundaries for self-space and dynamic number of detectors

Sajjad Fouladvand^{a,*}, Alireza Osareh^b, Bita Shadgar^c, Mario Pavone^d, Siyamack Sharafi^a

^a Academic Center for Education, Culture and Research, Lorestan, Iran

^b Department of Computing Science & Information Systems, Langara College, Canada

^c School of Computing Science, Simon Fraser University, Canada

^d Department of Mathematics and Computer Science, University of Catania, Italy

ARTICLE INFO

Article history:

Received 22 February 2016

Received in revised form
29 July 2016

Accepted 26 August 2016

Keywords:

Anomaly detection
Gaussian Mixture Model
Artificial Immune System
Negative selection algorithm
DENSA

ABSTRACT

The negative selection algorithm is an anomaly detection technique inspired by the self-nonsel self discrimination behavior observed in the Biological Immune System. The most controversial question of these algorithms is their poor performance on real world applications. To overcome such limitation this research work focuses on generating more efficient detectors through a more flexible boundary for self-patterns. Rather than applying conventional affinity measures, the detectors are generated benefiting from a Gaussian Mixture Model (GMM) fitted on normal space. From the GMM capabilities the algorithm is able to dynamically determine efficient subsets of detectors. In order to evaluate the efficiency and robustness of the proposed algorithm, different data sets have been examined as benchmark, including 2D synthesis data sets. Furthermore, for evaluating the capability and effectiveness of the proposed algorithm on real-world problems, it has been performed and tested for detecting anomalies in archaeological sites located in Lorestan, Iran. The experimental results prove how the proposed approach helps the negative selection algorithm to improve its detection capability, because the detectors can be efficiently distributed into the non-self space. It is important to note how this research work presents also a first analysis of the anomaly detection capabilities in the field of archaeology, introducing a novel application method, which can be efficiently used by the archaeologists for interpreting their growing amount of data and draw valuable conclusions about the historical past. Finally, in order to analyse the convergence and the running time of the proposed algorithm, a study has been conducted using the classical Time-To-Target plots, which present a standard graphical methodology for data analysis based on the comparisons between the empirical and theoretical distributions.

© 2016 Elsevier Ltd. All rights reserved.

1. Introduction

The immune system is the best defence system existing in the world able to protect organisms against invaders and diseases, and comprising many biological structures, and processes, made up of special cells, proteins, tissues, and organs. It is then able to detect and recognize a wide variety of agents (pathogens, viruses, fungi, parasitic worms, etc.); and distinguish between foreign agents (*nonself*), and own ones (*self*). Its main role is managed by white blood cells – B-cells, and T-cells – which are produced in the bone marrow. Here, some of them born and mature (B cells), whilst others migrate and mature in the thymus (T cells). The T-cell maturation undergoes a selection process in the thymus that is called *Negative Selection*: this process eliminates all T-cells that

react strongly with self-cells. A full explanation of such a complex system is outside the scope of this paper, and detailed review on its functionalities may be found in [Goldsby et al. \(2003\)](#) and [Aickelin and Dasgupta \(2010\)](#).

What makes the immune system very challenging, and source of inspiration it is to be the best recognition system, as well as its high ability in learning; memory usage; self-regulation; associative retrieval; threshold mechanism, and its ability in performing parallel, and distributed cognitive tasks ([Dasgupta, 2014](#)). In light of this, the Artificial Immune Systems (AIS) nowadays represent an efficient paradigm in bio-inspired computation, successfully applied in many real-world applications ([Costanza et al., 2015](#); [Cutello et al., 2011a, 2011b, 2007c](#)). The immune-inspired heuristics are primarily focused on four main theories: (1) negative selection ([Poggiolini and Engelbrecht, 2013](#)); (2) clonal selection ([Pavone et al., 2012](#); [Cutello et al., 2007a, 2007b](#)); (3) immune networks ([Smith and Timmis, 2008](#)); and (4) danger theory ([Aickelin and Cayzer, 2002](#); [Aickelin et al., 2003a, 2003b](#)). Such algorithms have

* Corresponding author.

E-mail address: sjfd.fouladvand@gmail.com (S. Fouladvand).

been successfully employed in a variety of different application areas.

One of the most widely used and developed algorithms as recognition system is the negative selection algorithm (NSA), which was proposed by Forrester et al. (1994), and takes inspiration by the negative selection process in order to achieve an efficient method for anomaly detection. This algorithm consists of two phases: (i) generating detectors and (ii) monitoring the occurrence of anomalies. In the first phase, the algorithm generates a number of random patterns – called detectors – that do not match any self-sample. During the second phase, if a detector pattern matches any new sample this situation is regarded as a possible anomaly occurrence.

Binary NSAs, and Real-Valued NSAs (RVNSAs) (Gonzalez et al., 2003, 2002) are the two main models of the NSAs. Unlike the binary NSA, the RVNSAs have attracted considerable attention in various fields. RVNSAs operate on a unitary hypercube $[0,1]^n$ such that each self-sample and nonself sample have a center and a constant radius. Thus, each detector represents a hypersphere: if an element lies within a detector hypersphere then it will be considered as an anomaly. Although different NSAs, especially RVNSAs, have shown promising performances in several fields, they have proved limitation, and poor performances in real world applications, and this has extensively reduced their use.

One of the most controversial questions for NSA is its innate limitation in detecting foreign patterns as anomalies that lead to an increase in False Positive Rate (FPR) (Kim et al., 2007; Aickelin et al., 2003a, 2003b). Many studies have been conducted for overcoming such shortcomings, which have led to several extensions that focus on generating more efficient detectors (Wang and Luo, 2009; Zhang and Luo, 2014), or on designing more efficient matching functions (Poggiolini and Engelbrecht, 2013; Luo et al., 2006; Ayara et al., 2002; Balthrop et al., 2002). In D'haeseleer et al. (1996) a binary detector generator was introduced using a linear-time algorithm in order to overcome the exponential cost of the original NSA. In this work the authors proposed a greedy algorithm that places the detectors as far apart as possible from each other in order to cover the nonself space more efficiently by using a more impact detector set. In Singh (2002), and Wierzchon (2000) a greedy algorithm was proposed for removing redundant detectors; whilst in Ayara et al. (2002) were investigated various algorithms for generating detectors, including the greedy algorithm proposed in D'haeseleer et al. (1996), and were compared in terms of time and space complexity, as well as the Detection Rate (DR) coverage of the final detector set. Thus, a new algorithm was proposed and called *NSMutation*, Negative Selection with Mutation, which was designed on the basic concepts of the original NSAs except for the elimination of the redundancy, and a better tuning of the parameters that improved the performances. In Ayara et al. (2002) was proved that there is no single algorithm able to produce, always, accurate detectors for any domain, because different domains show different constraints to satisfy. In addition to the linear-time algorithm, and greedy algorithms, several ways of evolving detectors have been proposed by Ayara et al. (2002), Gonzalez and Cannady (2004), and Hang and Dai (2004). In particular, in Gonzalez and Cannady (2004) a new NSA was presented, as well as the self-adaptive methods were developed that outperform *NSMutation* in term of higher detection rates; lower false detection rates; and computational time. In Gomez et al. (2003) was introduced a more flexible boundary between self and nonself space making use of fuzzy rules. In the research work by Ji (2006), instead, a RVNSA with variable-sized detectors (*v-detector*) was proposed, which benefits by detectors with variable sizes, and estimates coverage of an arbitrary number of detectors. In Stibor et al. (2005) a comparison between the real valued negative selection, and the *v-detector* algorithms was

presented via anomaly detection statistical techniques, proving the not competitive of NSA in real world, and high-dimensional applications. An extension of the NSA was also developed by Chmielewski and Wierzchon (2012) with the aim to increase its efficiency and coping with high-dimensional data. They simultaneously applied both binary and real valued detectors.

Unlike the *v-detector* that uses variable-sized detectors, Zhang and Luo (2011) and Zeng et al. (2012) have used, instead, self-samples with variable sizes. In particular in Zeng et al. (2012), the authors introduced a new scheme, called *VSRNSA*, for representing the self-space, where the self-samples have variable-sized radiuses that is based on the total distance from each sample to the other self-samples. A new affinity matching function was presented in Poggiolini and Engelbrecht (2013), called feature-detection rule, that defines a more operative matching function for NSA, which benefits from the inter relationship between adjacent and non-adjacent features of a particular problem domain.

The limitations of NSA in detecting foreign patterns as anomalies (Kim et al., 2007; Aickelin et al., 2003a, 2003b), and its drawbacks in real world applications, mainly on the ones with high dimensions Stibor et al. (2005) have motivated us to apply a more flexible boundary between self and non-self space. The NSA primary assumption that considers a sharp distinction between self and non-self space has, indeed, prevented from being used efficiently in real applications (Gomez et al., 2003). In Fouladvand et al. (2015) an improved version of the NSA was developed, called *DENSA* – Distribution Estimation based Negative Selection Algorithm, which uses a Gaussian Mixture Model (GMM) that attempts to create a more flexible, and efficient boundary for self-samples. GMMs, with strong statistical background, can cover the self-space efficiently, and it can also flexibly choose components' distributions, especially when full covariance matrixes are used. In this way, generating detectors through this sophisticated model can help in design to efficient NSA even in real-world contexts. Incorporating GMMs into a negative selection algorithm provides good opportunities to develop an efficient, and robust algorithm overcoming its limitations.

In this paper, we present an extended and revised version of the one already proposed in Fouladvand et al. (2015), which is basically improved in order to make it more efficient in real world applications. The probabilities of the Gaussian Mixture Model are used to distribute detectors on the non-self space more efficiently, which together with a suitably defined objective function – that tries to cover the non-self space – enable *DENSA* to be more efficient own in real world applications. Thus, for proving this last statement, *DENSA* has been applied in the field of Archaeology for interpreting the growing amount of archaeological data, and detect anomalous sites.

The archaeologists are still looking for efficient techniques that enable them to inspect and analyse archaeological sites through Artificial Intelligence approaches, which are increasingly employed to create new knowledge from archaeological data. Nowadays, the best known artificial intelligence approaches in archaeology are represented by artificial neural networks (ANN) and expert systems (Vitali, 1991; Voorrips, 1990; Richards, 1998). Deravignone and Jánica (2006) studied the basic concepts required to bring artificial intelligence methods into archaeological research, investigating, in particular, the application of Artificial Neural Networks in a raster GIS environment with the aim to create archaeological predictive models. A review on the implication to use computational intelligence models in archaeology was presented in Barceló (2010); this paper describes how artificial intelligence models are feasible in archaeological recognition systems just like other sciences. Puyol-Gruaiart (1999) has considered the possibility of using more recent subfields including Knowledge Discovery in Databases (KDD), Visual Information

Management (VIM) and Multi-agent Systems (MAS) in archaeological research. One of the most challenging tasks for archaeologists is to find anomalies in a set of archaeological sites as it enables them to draw valuable conclusions about the events that caused those anomalies in human settlements in the past. In this paper, the environmental variables of archaeological sites of Silakhor, which is an ancient place located in west Iran, are extracted using ArcGIS 10.1. After that, these data are plugged into the proposed algorithm to evaluate the capabilities of DENSA in detecting anomalous sites. This experiment is a first analysis of the anomaly detection capabilities of a negative selection algorithm in the field of archaeology. The results show that DENSA can be easily utilized to help archaeologists to detect anomalous sites and consequently it helps them to interpret their growing amount of data more efficiently. Finally, with respect to the DENSA's original version (Fouladvand et al., 2015), in this work we have included also an analysis on the convergence and running time of DENSA using the classical Time-To-Target plots, which compares the empirical and theoretical distributions.

The paper is structured as follows: in Section 2 GMM concepts are presented, and we explain how they can be optimally fitted on a self-space; in Section 3 are introduced and described the training and testing phases, as well as the objective function used in DENSA; in Section 4 we present the experimental results obtained on 2D synthesis data sets, another real world data set, and on archaeological data set; finally, in Section 5 we give the conclusions, including some beneficial and new ideas for future research.

2. Gaussian Mixture Model of self space

The Gaussian Mixture Models (GMMs) are widely used in applications where data can be viewed as a combination of different populations mixed in varying proportions. In a mixture model, a probability density function is expressed as a linear combination of basis functions. A general model can be shown as follows:

$$p(x) = \sum_{j=1}^M p(j)p(x|j) \quad (1)$$

where M is the number of Gaussian components, and $p(j)$ is the mixing coefficient, which corresponds to the prior probability that the D -dimensional feature vector x is generated by the M components. The parameters of the component density function $p(x|j)$ typically vary with j (Nabney, 2004). The mean, the covariance matrix and the mixing coefficient parameterize each component density. Basically, there are three forms of covariance matrices: spherical, diagonal and full. These kinds of covariance matrices have higher flexibility in estimating the underlying distributions (Osareh, 2004). Thus, a full covariance matrix for each mixture component is benefitted in this paper. The full covariance matrix can be any positive definite ($d \times d$) matrix, and the density function is described as

$$p(x|j) = \frac{1}{(2\pi)^{d/2} |\Sigma_j|^{1/2}} \exp\left\{-\frac{1}{2}(x-\mu_j)^T \Sigma_j^{-1}(x-\mu_j)\right\} \quad (2)$$

where Σ_j is the covariance matrix, and μ is the mean vector of d dimensions for a class. The method for determining the parameters of a GMM from a data set is based on maximizing the data likelihood. One of the most widely used approaches to maximize the likelihood is the Expectation-Maximization (EM) algorithm, which is based on data likelihood maximization. Therefore, becomes more convenient rewrite the problem in its equivalent form of minimizing the negative log likelihood of the data. Thus, the error function is defined as (Nabney, 2004)

$$E = -L = - \sum_{n=1}^N \log p(x^n) \quad (3)$$

where E is an error function, which is minimized within the training phase. The EM algorithm modifies the GMM parameters (E-step), and decreases E (m -step) until a minimum is reached. The EM algorithm generates a sequence of estimations for parameters $w^{(m)}$ starting from the initial parameter set $w^{(0)}$. The E step includes calculation of a function Q that is defined as

$$Q(ww^n) = E(\log p(yw))p(z^n x^n, w^{(m)}) \quad (4)$$

For each data point x^n , there is a corresponding random variable z^n , which is a hidden variable. This hidden variable determines a Gaussian component that is related to the variable x^n . In fact, the complete data point form is considered as: $y^n = (x^n, z^n)$. To calculate the new set of parameters in M-Step, the function $Q(ww^n)$ should be optimized as

$$w^{(m+1)} = \operatorname{argmax}_w Q(ww^{(m)}) \quad (5)$$

The EM algorithm needs to be initialized by making some initial guesses for the parameters of the model. In this work, a K-means clustering algorithm was used to initialize the EM algorithm, which uses the Euclidean distance. Remains only to decide the number of K-means clusters for representing the initial Gaussian components for the GMM. The number of mixture components in GMMs relies on a combination of good modelling, a sensible number of components, and avoid a highly complex model. In this paper, the right number of components is chosen repeating the density model estimation, and evaluating a criterion by varying the number of components. The evaluation measurement is the Bayesian Information Criterion (BIC) principle (Osareh, 2004), and the number of components is defined as

$$\text{BIC}(K) = -l(\theta) + \frac{1}{2}\gamma(K)\log\gamma N \quad (6)$$

where N indicates the number of data points; $l(\theta)$ is the data log likelihood; K is the number of Gaussian components in a GMM; and $\gamma(K)$ demonstrates the number of free parameters in a mixture model. There are $d(d+1)/2$ free parameters for each full covariance matrix component, where d is the dimensionality of the feature space. Each mean vector μ has d free parameters, and the mixing parameters $P(j)$ require another $K-1$ free parameters. Thus, the dimension of a GMM may be written as

$$\gamma(K) = K \left(\frac{d(d+1)}{2} + d \right) + K - 1 \quad (7)$$

As mentioned above, the optimum number of Gaussian components for the self-space is obtained using Eq. (6), and varying the number of components within the range $[1, M]$. It is important to make use of a reasonable number of components either for minimizing the computational costs, and because the EM algorithm could failing. In a nutshell, EM algorithm cannot proceed if Gaussian components include only a few samples.

An efficient heuristic is to observe the different BIC values and to select the number of components corresponding to the first decisive local minimum of the BIC values (Fraleigh and Raftery, 1998). In light of this, the number of Gaussian components is varied within the range $[1, 20]$ in order to select the first local minimum value. As a result, the BIC function, which corresponds to the Pentagram-big, Cross-thick, Triangle-big, Stripe-thick and Ring-thick distributions, reaches the minimum values respectively to $K=8$, $K=6$, $K=6$, $K=7$ and $K=5$. It should be noted that this results are the average of the outputs obtained performing 100 independent runs on each data set, and therefore, the mean value

of K obtained for each data set is rounded.

3. Distribution Estimation based Negative Selection Algorithm (DENSE)

Similar to the NSA, the goal of DENSE is to generate a set of detectors that covers the non-self space efficiently. The training phase of the proposed algorithm can be summarized as in Fig. 1. DENSE first maps the data in the interval between $[0, 1]^n$; then, it compares the randomly generated samples with a GMM, which is suitably fitted on normal space to generate a predefined number of detectors. In a nutshell, instead of comparing a randomly generated pattern to all self-samples, DENSE compares the randomly generated sample with a couple of flexible Gaussian components, which are representative for the self-space. As DENSE generates the detectors, it calculates an objective function for the future decision on the optimum number of detectors. This objective function selects the optimum number of detectors based on estimating efficacy of the detector set. It is fully described in Section 3.1.

The test stage of DENSE, instead, is accomplished via the two following stages:

- calculating the Euclidean distance of an input sample from the nearest detector;
- classifying the input sample as *normal* or *abnormal*, based on the calculated distance in previous step and a threshold termed *Threshold₂*.

Threshold₁ and *Threshold₂* are important parameters, which have major impacts on the accuracy of DENSE. *Threshold₁* is a parameter used in the training phase for evaluating if a randomly generated sample is far enough from the Gaussian components in the normal space in order to be considered as a detector. Whilst the *Threshold₂* is another control parameter used in the testing phase, which evaluates if a test sample is close enough to a detector to be considered as an anomaly, or not. As the *Threshold₂* increases, more test samples are considered anomalies, and therefore, the rates of false positive, and true positive increase too.

3.1. An objective function for selecting optimum number of

generated detectors

In order to have an efficient objective function, the distribution of the detectors must be able to well cover the non-self space using a number of detectors as small as possible. In this work is considered an objective function based on three main items: a first term for evaluating the detection rate; a second one for estimating the false positive rate; and a last one for keeping the size of the detector set as small as possible. Thus, the objective function used in order to satisfy these constraints is defined as follows:

$$\text{Objective}(i) = \left(W_1 \times \text{DetectionRate}(S, G, D) + W_2 \times \frac{1}{1 + \text{FalsePositiveRate}(S, G, D)} + W_3 \times \frac{1}{|D|} \right) \quad (8)$$

where S is the self-space, G is the Gaussian Mixture Model of the normal space, and D is the detector set. The parameters W_1 , W_2 and W_3 represent instead the weighted values of the single sub-objectives, whose aim is to increase the flexibility of the objective function, and cope to different applications and/or data sets. In this research work W_1 , W_2 and W_3 have been set to 1. Besides, $\text{DetectionRate}(S, G, D)$ and $\text{FalsePositiveRate}(S, G, D)$ respectively denote the estimated detection rate and false positive rate of the anomaly detection system using a validation set and the current set of detectors D .

It is easy to check how the optimum number of detectors is obtained by maximizing the objective function in Eq. (8). This means maximize the estimated detection rate, and minimize the estimated false positive rate, as well as the size of the detector set. In fact, as the number of efficient detectors increases, increase also of the first two terms of (8), and consequently will increase the whole value of the objective function. On the other hand, by increasing the number of detectors, the last term of (8) decreases, and consequently the whole objective function value will decrease. Fig. 2 indicates the objective function values versus the number of generated detectors for all 2-D synthesis data sets. In these plots, the horizontal and vertical axes represent respectively the number of generated detectors, and the corresponding objective function values; each plotted value is averaged over 50 independent runs. It

1. Perform Min-Max normalization and map the data in the interval between $[0, 1]^n$.
2. Find the optimum number of GMM components for self-data using BIC.
3. Fit a GMM on the training data using the EM algorithm.
4. Find the optimum value for *Threshold₁* using a k-fold cross-validation.
5. Repeat
 - 5.1. Generate a random pattern x_i in the interval between $[0, 1]^n$.
 - 5.2. If $p(x_i) < \text{Threshold}_1$ then add x_i to the detector set D .
 - 5.3. Calculate the objective function value considering the newly added detector.
6. Until ($|D| > \text{a predefined number of detectors}$)
7. Investigate the objective values and select the optimum subset of generated detectors

Fig. 1. DENSE training phase.

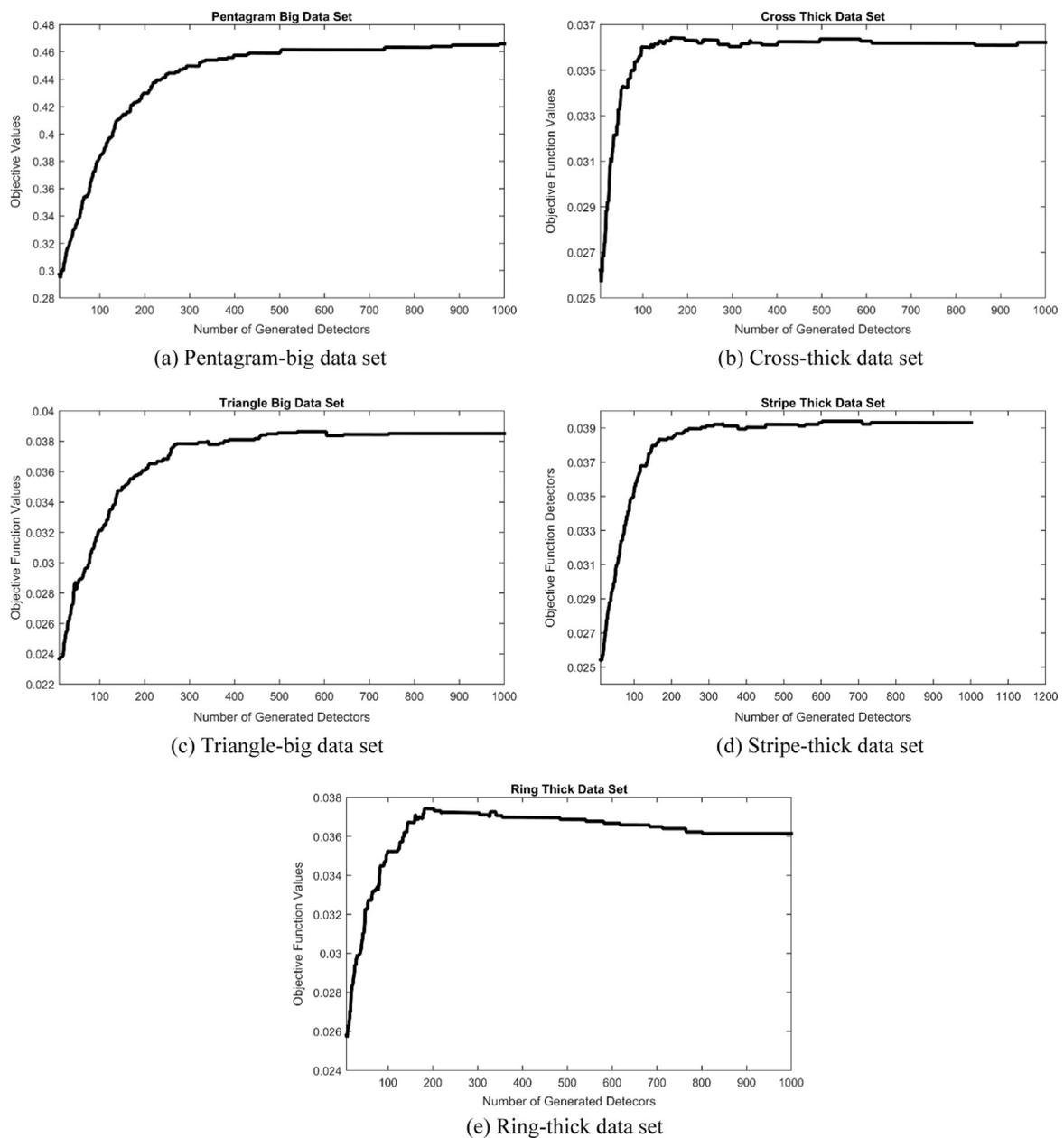


Fig. 2. Objective function values versus the number of generated detectors.

is worth noting that *Threshold₂* has been set to 0.05, as a default value, for all experiments shown in Fig. 2. Inspecting each plots of this figure, the optimum number of detectors for Pentagram-big, Cross-thick, Triangle-big, Stripe-thick and Ring-thick data sets, is reached respectively to 989, 163, 540, 603 and 181.

3.2. Time-To-Target analysis

The Time-To-Target plots (Aiex et al., 2002; Feo et al., 1994) are a standard graphical methodology for data analysis (Chambers et al., 1983) to compare the empirical and theoretical distributions, and, nowadays, they represent an easy way to characterize the running time of a generic stochastic algorithm in order to solve a given combinatorial optimization problem. They display the probability that an algorithm will find a solution as good as a target within a given running time.

Through the Time-To-Target analysis two kinds of plots are produced: QQ-plot with superimposed variability information;

and superimposed empirical and theoretical distributions. For our study we have used the Perl program proposed in (Aiex et al., 2007), which represents a useful tool for the comparisons of different stochastic algorithms or, in general, strategies for solving a given problem. Such program can be downloaded from <http://mauricio.resende.info/ttplots/>.

For this study we performed DENSA on two different instances of the Pentagram-big dates, with different features and complexity (of course with not simple targets), where the detector set always correctly detects the targets. For these experiments the termination criterion was properly set until finding the target. Besides, taking into account that larger is the number of runs closer is the empirical distribution to the theoretical distribution, we performed our experiments on 200 independent runs: for each run the random number generator has been initialized with a distinct seed.

The plots produced on the two different data sets are shown in Figs. 3 and 4. The left plots show the comparisons between the

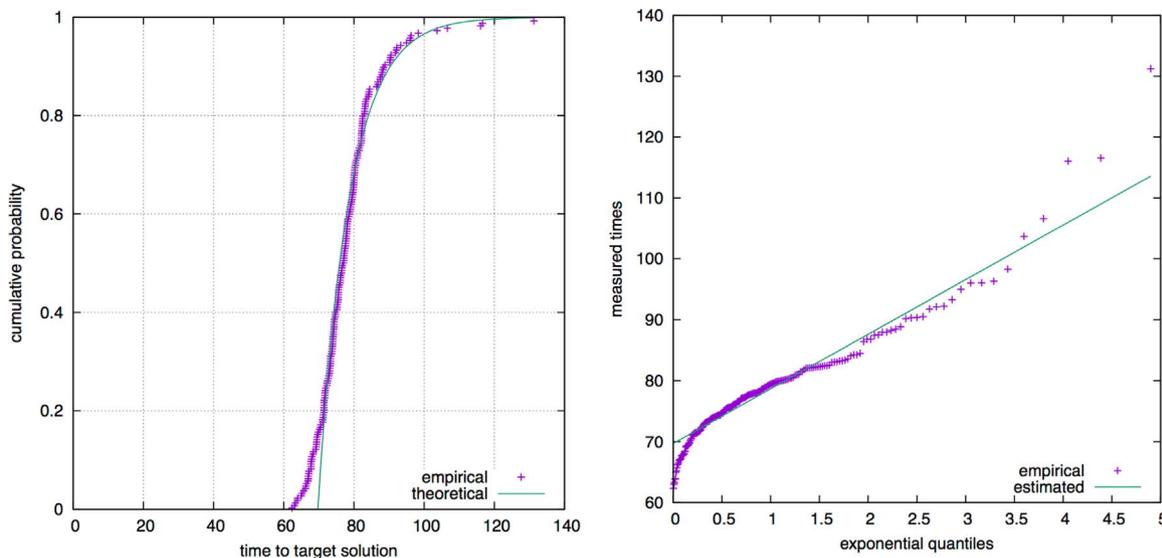


Fig. 3. Time-To-Target plots on the first data set: empirical versus theoretical distributions (left plot), and QQ-plots with variability information (right plot).

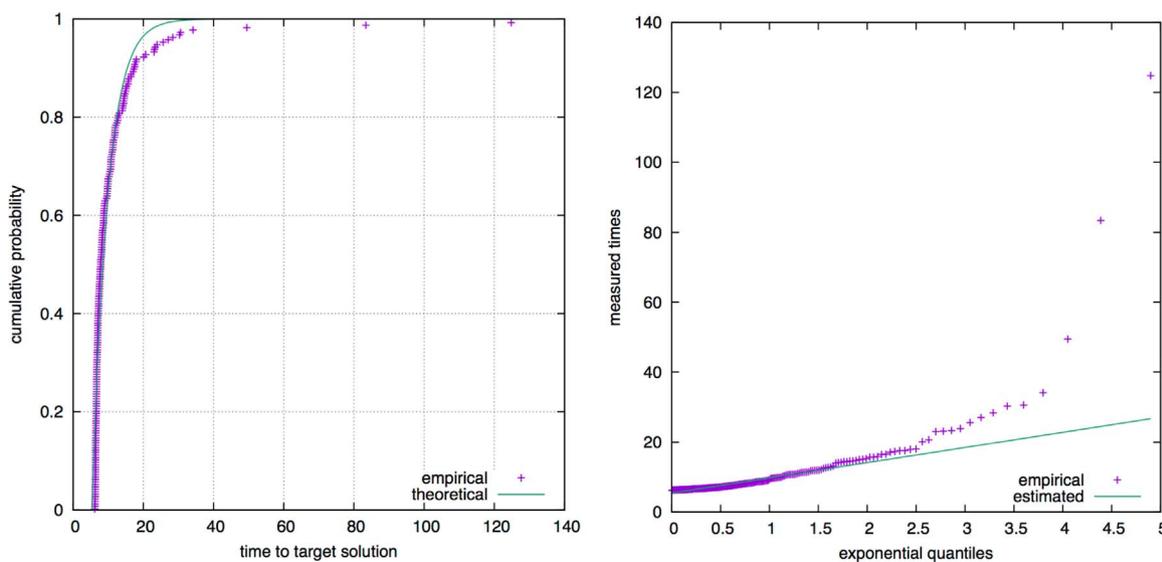


Fig. 4. Time-To-Target plots on the second data set: empirical versus theoretical distributions (left plot), and QQ-plots with variability information (right plot).

empirical and theoretical distributions, whilst in right ones the QQ-plots with variability information. Analysing the left plot of Fig. 3, it is possible to see how the curve of the empirical distribution matches almost always with the theoretical one, except for very few cases. Same behavior is shown in the left plot of Fig. 4. Instead, inspecting the Quantile-Quantile plots (right plots of both figures) is possible to see how often the curve produced by DENSA is the equal, or however very closer, to the estimated one, except for the last points in the second data set.

From these figures we can assert how DENSA shows to be robust, since the empirical points are very close to theoretical and estimated points.

4. Experimental results and discussion

In order to visualize how GMM can properly fit on the normal space, and investigate the performance of the proposed algorithm, the Fig. 5 represents, respectively, the self-space (red plots), the Gaussian components fitted on self-space (black ovals), and the generated detectors by DENSA (blue stars) in five different

2-dimensional synthetic data sets: (a) Pentagon-big; (b) Cross-thick; (c) Triangle-big; (d) Stripe-thick; and (e) Ring-thick self-regions over the whole space. Each data set has a geometric shape and includes 2000 real valued samples: 1000 samples for training purpose, and another 1000 samples for testing. In this research work, all 1000 normal samples in the training set were considered as train set, and the test set, which includes 1000 normal and abnormal samples, was used to conduct a K fold-cross validation (with $K=3$ as a default value). The 3-fold cross validation includes one main loop. In this loop the test data that are a mixture of 1000 normal and abnormal samples, are partitioned into 3 parts of equal sizes for guaranteeing that the ratio between self and non-self samples is kept yet; hence, one of them is used to validate the GMM of normal space (i.e. optimise the $Threshold_1$), and tested on the remaining two parts. This procedure is repeated for all three possible choices for the held-out part, and the performance scores from the 100 runs are averaged. As a result, the values for $Threshold_1$ have been set to 1.54, 0.51, 1.21, 0.74 and 0.49, respectively, for (a) Pentagon-big, (b) Cross-thick, (c) Triangle-big, (d) Stripe-thick, and (e) Ring-thick data sets. It is worth noting that $p(x_i)$ is the probability of the density function that is not negative,

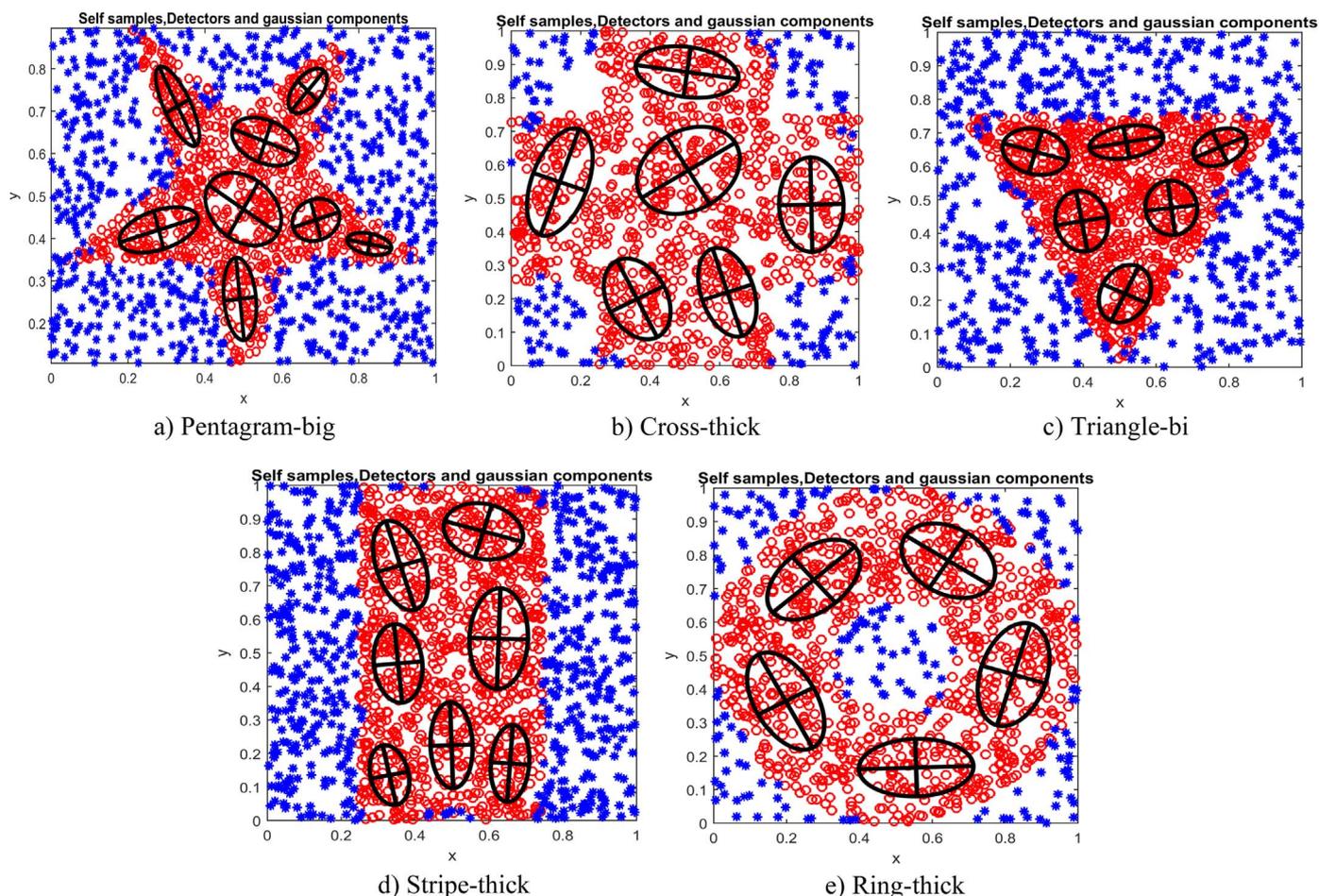


Fig. 5. Normal space (red dots), Gaussian components (black ovals) and detectors (blue stars). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and its full integral is 1; therefore, its integral over any region is less than 1. However, there is no upper bound to the value of the probability density function, and they can exceed 1.

It should also be noted that the optimum number of Gaussian components (black ovals in Fig. 5) was computed using BIC (Eq. (8)). Moreover, using the objective function described in Eq. (8), the optimal numbers of detectors for Pentagram-big, Cross-thick, Triangle-big, Stripe-thick and Ring-thick data sets are respectively: 499, 384, 599, 524 and 540.

Fig. 6 shows the complete trend of the effect of *Threshold_2* for the values from 0.001 up to 0.1, performed on the data sets: (a) Pentagram-big; (b) Cross-thick; (c) Triangle-big; (d) Stripe-thick; and (e) Ring-thick. All the results shown in this figure have been obtained using the 3-fold cross validation described above, and have been averaged over 100 independent runs. The accuracy; true positive rate (detection rate); and false alarm rate are defined as follows:

$$\text{Accuracy} = \frac{\text{TPs} + \text{TNs}}{\text{TPs} + \text{TNs} + \text{FPs} + \text{FNs}} \quad (9)$$

$$\text{True Positive Rate (Detection Rate)} = \frac{\text{TPs}}{\text{TPs} + \text{FNs}} \quad (10)$$

$$\text{False Positive Rate (False Alarm Rate)} = \frac{\text{FPs}}{\text{FPs} + \text{TNs}} \quad (11)$$

where TPs (true positives) indicates the number of abnormal

samples, which are correctly classified as anomalous; TNs (True Negatives) means the number of normal samples, which are correctly classified as normal; FPs (False Positives) and FN (False Negatives) are respectively the number of normal samples that are incorrectly classified as anomalous, and the number of anomalous samples that are incorrectly classified as normal.

Analysing Figs. 5 and 6 is possible to see that DENSA is able to efficiently generate detectors on the nonself space, and consequently, performs an efficient, and robust anomaly detector on these synthetic data sets. Further, according to the plots in Fig. 6, as soon as *Threshold_2* increases, more test samples are considered as anomalies, and then both rates of false positive, and true positive increase too.

4.1. A real world application: anomaly detection in computer networks

In an attempt to evaluate the efficiency of DENSA's performance in a more practical context, and examine its feasibility, DENSA has been also performed on the NSL-KDD data set, which is a modified data set for KDDCup 1999 data set (Tavallaei et al., 2009). This data set is commonly used to train and evaluate network intrusion detection systems (IDSs), and presents 125,973 records in the train set (67,343 normal samples and 58,630 abnormal samples), and 22,544 records in the test set (9711 normal samples and 12,833 abnormal samples). Moreover, to study the property, and the possible advantages of DENSA, several experiments have been also carried out to compare the results of DENSA with the ones obtained by the V-detector algorithm. For both

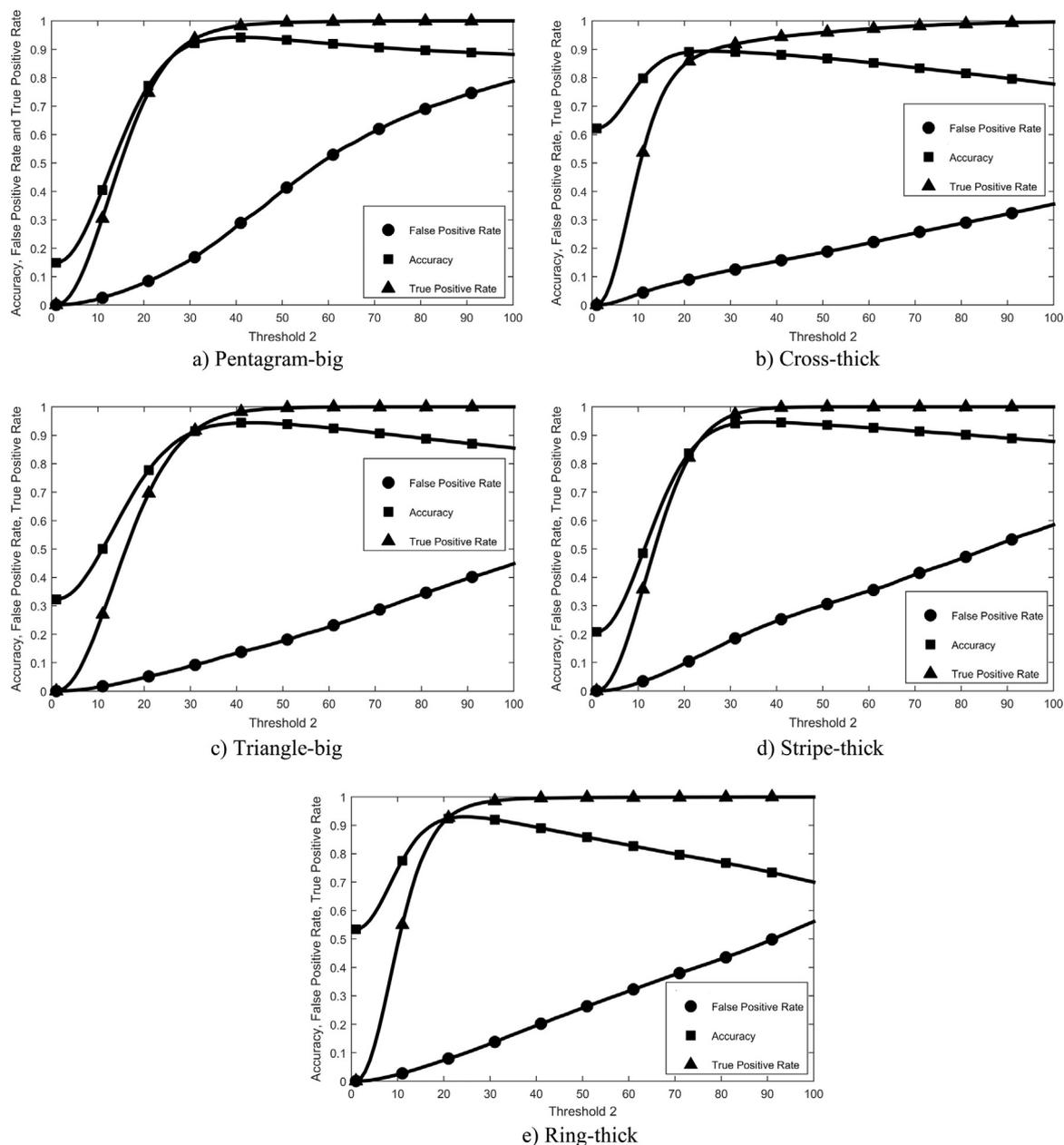


Fig. 6. The effect of different values of the $Threshold_2$ on the results.

algorithms, the self set includes all 67,343 normal samples of NSL-KDD training set, whilst the test set comprises 22,544 NSL-KDD test samples. In Fig. 7 it is shown the comparison of the DENSA's efficacy versus V-detector, over the whole range of thresholds. The experiments shown in Fig. 7 have been performed using 20 different threshold values: for V-detector are the various values of self-radius, whilst for DENSA represents the $Threshold_2$ values. Besides, for the V-detector algorithm have been fixed the parameters as follow: $T_{max}=60,000$ (almost equal to the number of self-samples), $C_0=0.99$, and $C_{max}=0.99$; for DENSA instead the optimum value for $Threshold_1$ has been set through a 3-fold cross validation, and using the NSL-KDD train set.

As it is illustrated in Fig. 7, it is easy to see how generate detectors using an optimally fitted GMM that covers the self-space properly, and may flexibly choose component distributions on self-space, leads to an efficient NSA properly able to detect intrusions. Indeed, DENSA generates detectors based on smooth, and optimally fitted boundary on the self space, unlike V-detector that

generates them based on a distinct sharp between self, and non-self spaces. As a result, DENSA yields a higher accuracy than the V-detector method, as shown in Fig. 7(a). It can be also noticed from the Fig. 7(c) as DENSA is more inclined to a significant increase in the false positive rate than V-detector. However, the main control parameter $Threshold_2$ can be used for balancing between high rates of true positives, and low rates of the false positives. Is possible indeed to observe from the Fig. 7 how the $Threshold_2$ values affect on the DENSA results. In fact, high $Threshold_2$ values would result to high true positive rates, but also high false positive rates would be achieved. On the other hand, smaller $Threshold_2$ values would result to low false positive rates, but also to low true positive rates. Thus, the $Threshold_2$ parameter can be suitably, and easily tuned for having an efficient anomaly detection system, which has shown better accuracy with respect V-detector (Fig. 7(a)).

4.2. A real world application: anomalous archaeological site

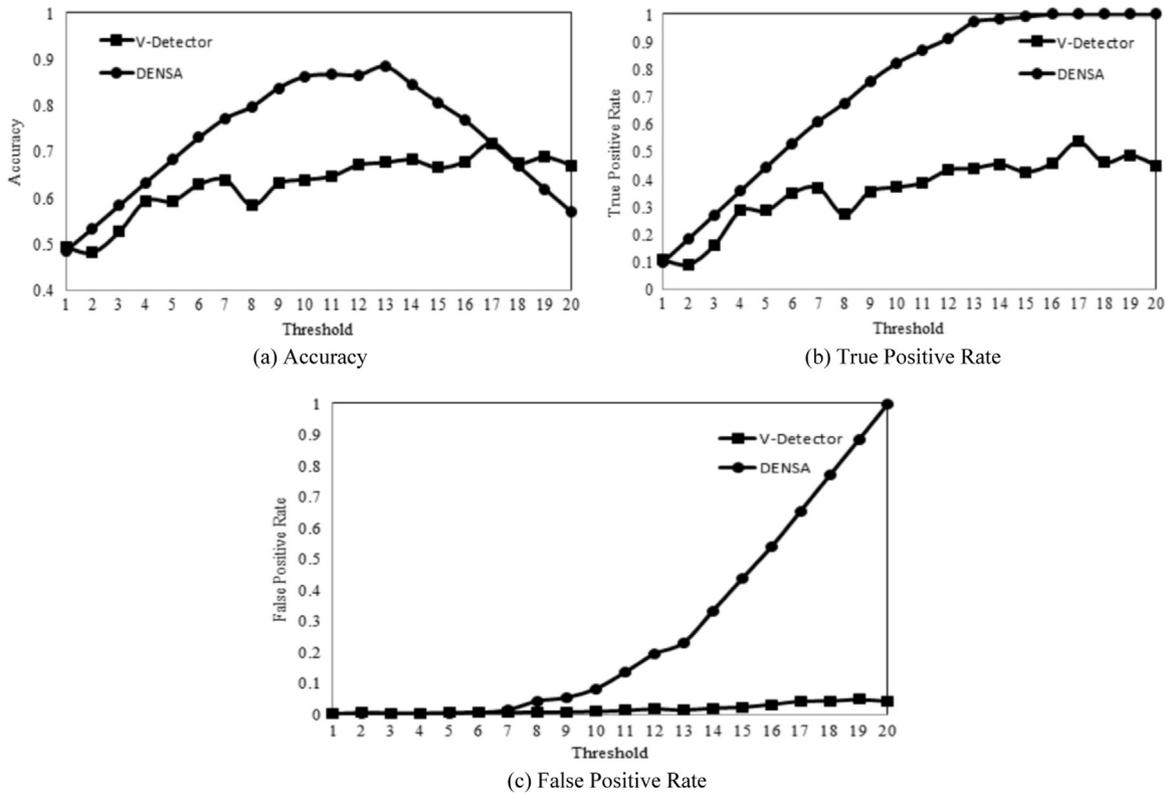


Fig. 7. Classification performance of DENSA and V-Detector on NSL-KDD data set.

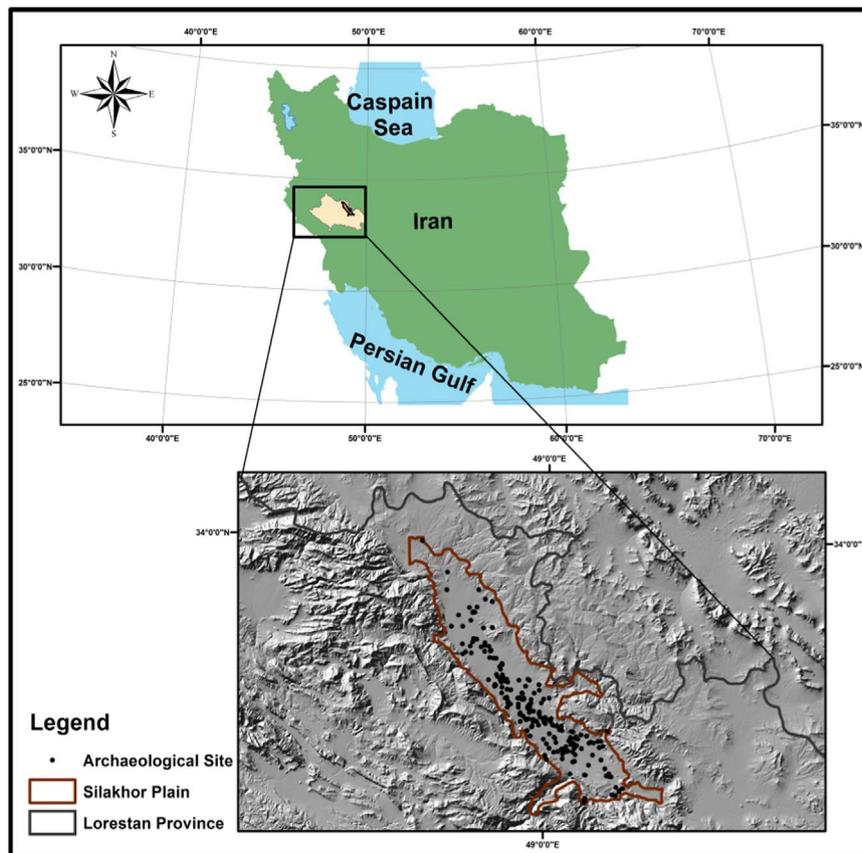


Fig. 8. Location of Silakhor plain in Iran.

detection

Once evaluated the good performances of DENSA on the used test-beds (see sections above), and having shown good detection accuracy, we have evaluated also its efficacy on a real-world problem in archaeology by analysing its anomaly detection capabilities. Detect anomalies in archaeological human settlements is an essential task for archaeologists. The anomalous archaeological sites are all those that do not follow the normal settlement patterns as the other sites while having same circumstances, such as same history period, or same locations. Thus, being able to detect these anomalies enables the archaeologists to lead more studies onsite; scrutinize the possible reasons and causes that have made those sites anomalous, and consequently, helping them in reveal more information about the past events, such as the occurrence of ancient battles, which not allowed to the site to follow the usual pattern of other sites in its neighbourhood.

This research paper is therefore a first analysis of the anomaly detection capabilities of the Artificial Intelligence methods for

detecting anomalous archaeological sites. Thus, DENSA has been tested, and performed in detecting anomalies on the archaeological sites of the Silakhor plain, which is located in Lorestan province, in the west Iran: it lies within E 48 35" to E 35 48" and N 33 47" to N 33 59", as shown in Fig. 8. It is one of the oldest residential plains, occupied after the prevalence of agriculture during the first, and fourth millennium BC. There have been many mounds in this plain that not only enabled the human to be protected against wild animals, and other humans, but also to have easy access to their agricultural lands. Besides, water resources, fertile soil, and a flat topography indicate the existence of numerous cultures and archaeological sites own on this area (Hole, 1970); this, together with its compactness, has led the plain to become an important focus for the archaeological investigations. There are over 201 archaeological sites in this ancient location that are used as a data set in this paper. These 201 archaeological sites were detected in 2000 by the Cultural Heritage of Lorestan province.

The archaeological sites of Silakhor originate from a period of

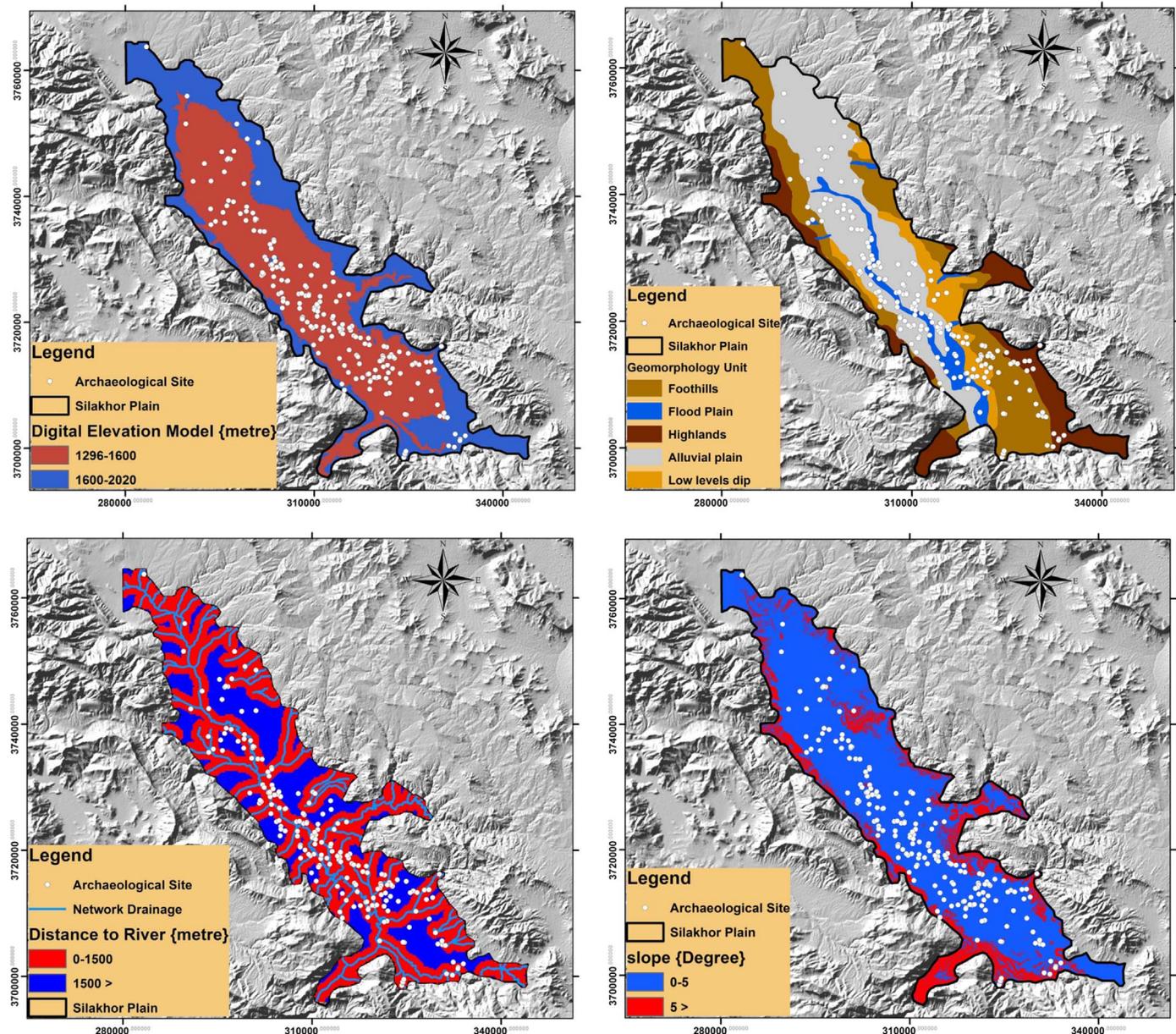


Fig. 9. Environmental-based raster layers used in the models.

one millennium (Achaemenian, Parthian and Sassanid), thus they did not experienced significant environmental changes, and they should follow a normal behavior of settlement pattern. However, according to archaeological field studies (Maghsoudi et al., 2014) there are some anomalies in the settlement patterns of these sites, which may have been caused by unnatural events, such as security matters, battles or political issues.

The methodology used in this work can be divided into two main stages: the environmental factors (Maghsoudi et al., 2014) of all 201 archaeological sites have been collected using ARC GIS 10.1; and later such results have been considered as the input for the proposed anomaly detector algorithm (DENSEA). The environmental factors of the 201 archaeological sites have been prepared, which include Geomorphic Unit (1:250,000); slope (1:25,000); distance to water resources (1:50,000); elevation; and altitude (1:25000), and after the raster layers of these factors have been generated via ArcGIS 10.1. Fig. 9 shows these raster layers in which the sizes of each cell is $20 \times 20 \text{ m}^2$. These digital values once extracted for each factor, they were exported into Microsoft Excel 2010; thus, the Min-Max normalization (Han et al., 2006) was performed on the data set in order to reduce the effect of the measurement unit on the learning process of the algorithms. As outcome, a digital database of the environmental characteristics is extracted for 201 historical sites. Hence, each sample is labelled as *normal* or *abnormal* based on the values of their variables (Maghsoudi et al., 2014). In this way, a data set with 201 samples (48 abnormal and 153 normal) is achieved, and used then for evaluating the performance of DENSEA as anomalous sites detector.

As mentioned before, one of the open questions for the negative selection algorithms is their efficiency in dealing with real-world applications (Stibor et al., 2005), whose issue is the core of this research work. Moreover, the advantage of the Gaussian Mixture Models is define more flexible boundaries for the self space, and consequently, more efficient detectors. However, in dealing the anomalous archaeological sites detection, becomes important also the location of the detectors inside the feature space. Therefore, becomes crucial carefully select the detectors in such way to cover efficiently the non-self space. Thus a novel method based on GMM probabilities is employed to select detectors, which are efficiently scattered over the non-self space; in this way, the detectors are distributed more efficiently in the whole non-self space.

In order to achieve this goal, a function, $\text{dis}(d_j)$, is defined, which calculates the distance of a generated detector to the GMM of normal space:

$$\text{dis}(d_j) = 1 - \left(\frac{P(d_j)}{\text{Threshold}_1} \right) \quad (12)$$

where $P(d_j)$ denotes the probability of j th detector generated by the GMM of normal data. Fig. 10 shows the graph of the function in Eq. (12). According to this diagram, as the probability of a detector increases, the value of $\text{dis}(d_j)$ decreases. Indeed, this function returns lower values for those detectors that are nearer to the normal space (their probabilities is closer to the Threshold_1). Follows then that this function can be used to distribute more efficiently detectors on the non-self space.

In real world applications, the traditional real valued negative selection algorithm cannot be efficiently applied due to the high dimensionality of the data sets; in fact, generating random detectors in a high dimensional feature space will lead to poor results or even a disproportionate number of detectors. In this research work, the function in Eq. (12) has been used to generate detectors distributed over all the non-self space. Fig. 11 shows the DENSEA pseudo code, properly modified to make it more efficient in detecting archaeological sites; with respect to the ones shown in

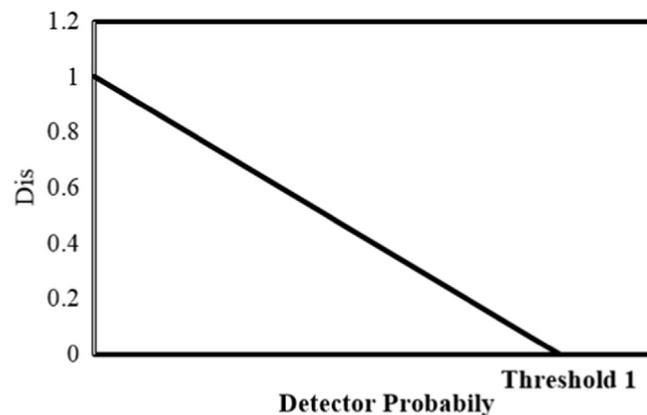


Fig. 10. The graph of the function shown in Eq. (12).

Fig. 1, the lines 5.1 and 5.2 have been replaced with the algorithm shown in Fig. 11 in order to distribute the detectors more efficiently on the archaeological site feature space. The algorithm tries to divide the non-self space into four different regions by controlling the GMM probabilities of the detectors (the number 4 is selected as a default value, and it can be changed in different applications).

This is due because, when a random pattern is generated (line 9.1 in Fig. 11), this can be anywhere in the feature space: (i) inside the self-space (high values of $P(x)$); (ii) somewhere outside of the normal space, but not so far from the self space (low values of $P(x)$); (iii) outside of the self-space, and away from the normal space (very low values of $P(x)$). If, however, $P(x)$ is laid into the range $[0, \text{Threshold}_1]$, this means that the pattern is located in the non-self space, and in particular with $P(x)$ near to 0 it is away from boundary, whilst with $P(x)$ near to Threshold_1 it is located near the boundary of the self space. Thus, inspecting $P(x)$ is possible to guess how far is a random pattern from the normal space. In light of this, because we need to cover the whole non-self space, including spaces near the self boundaries, we have divided the range $[0, \text{Threshold}_1]$ into four sub intervals, and we have generated patterns in all sub-intervals to be sure that the detectors are fairly scattered on the non-self space.

To see the basic property and the visual effect of the proposed approach, DENSEA has been employed to generate detectors for the Pentagon-big data set (see Fig. 12). Fig. 12(a) shows only the detectors with the values of $\text{dist}(d_j)$ greater than 0.99, whilst the plot (b) those whose values of $\text{dist}(d_j)$ are less than 0.25. As it can be seen from Fig. 12, by changing the sub-intervals, and controlling the values of GMM probabilities of the detectors it is easy to distribute them in the non-self space, and keep their balance, so that they cover the non-self space more efficiently.

For the archaeological data set, the optimum value of Threshold_1 was set to 13.99, which was computed using a k-fold cross validation method, with $k=3$ as a default value; whilst, in our experiments, the variables MAX1, MAX2, MAX3 and MAX4 have been considered to be 200. Moreover, the non-self space has been divided into four equal sub-regions by setting the variables first_sub_regions , $\text{second_sub_regions}$, third_sub_regions and $\text{fourth_sub_regions}$ to 0.25, 0.5, 0.75 and 1, respectively. In addition to the above adjustments, the optimum number of Gaussian components for the archaeological site data set is 9, which is obtained using Eq. (6). Fig. 13 shows the performance of DENSEA in detecting anomalous sites of Silakhor. These results have been obtained on the test sets, and have been averaged on 100 independent runs. It can be noticed in Fig. 13 that the main control parameter, Threshold_2 , can be used to balance the rates between higher true positive, and lower false positive. High Threshold_2

```

1. MAX1= predefined maximum number of detectors with the  $\text{dis}(d_i)$  values for them is between  $[0, \text{first\_sub\_region})$ .
2. MAX2= predefined maximum number of detectors with the  $\text{dis}(d_i)$  values for them is between  $[\text{first\_sub\_region}, \text{second\_sub\_region})$ .
3. MAX3= predefined maximum number of detectors with the  $\text{dis}(d_i)$  values for them is between  $[\text{second\_sub\_region}, \text{third\_sub\_region})$ .
4. MAX4= predefined maximum number of detectors with the  $\text{dis}(d_i)$  values for them is between  $[\text{third\_sub\_region}, \text{fourth\_sub\_region})$ .
5. first_sub_detector_numbers=0;
6. second_sub_detector_numbers=0;
7. third_sub_detector_numbers=0;
8. fourth_sub_detector_numbers=0;
9. Repeat
  9.1. Generate a random pattern  $x_i$  in the interval between  $[0,1]^2$ .
  9.2. Distance=  $1 - (p(x_i)/\text{Threshold}_1)$ 
  9.3. If  $(p(x_i) < \text{Threshold}_1 \text{ AND } 0 < \text{Distance} < \text{first\_sub\_region} \text{ AND } \text{first\_sub\_detector\_numbers} < \text{MAX1})$ 
    9.3.1. Add  $x_i$  to the detector set  $D$ .
    9.3.2. first_sub_detector_numbers ++
  9.4. Else If  $(p(x_i) < \text{Threshold}_1 \text{ AND } \text{first\_sub\_region} < \text{Distance} < \text{second\_sub\_region} \text{ AND } \text{second\_sub\_detector\_numbers} < \text{MAX2})$ 
    9.4.1. Add  $x_i$  to the detector set  $D$ .
    9.4.2. second_sub_detector_numbers ++
  9.5. Else If  $(p(x_i) < \text{Threshold}_1 \text{ AND } \text{second\_sub\_region} < \text{Distance} < \text{Third\_sub\_region} \text{ AND } \text{Third\_sub\_detector\_numbers} < \text{MAX3})$ 
    9.5.1. Add  $x_i$  to the detector set  $D$ .
    9.5.2. third_sub_detector_numbers ++
  9.6. Else  $(p(x_i) < \text{Threshold}_1 \text{ AND } \text{third\_sub\_region} < \text{Distance} < \text{fourth\_sub\_region} \text{ AND } \text{fourth\_sub\_detector\_numbers} < \text{MAX4})$ 
    9.6.1 Add  $x_i$  to the detector set  $D$ .
    9.6.2. fourth_sub_detector_numbers ++
  9.7. end
10. Until  $((\text{first\_sub\_detector\_numbers}=\text{MAX1}) \text{ AND } (\text{second\_sub\_detector\_numbers}=\text{MAX2})$ 
    AND  $(\text{third\_sub\_detector\_numbers}=\text{MAX3}) \text{ AND } (\text{fourth\_sub\_detector\_numbers}=\text{MAX4}))$ 

```

Fig. 11. The DENSA pseudocode properly modified to make it more efficient in detecting archaeological sites: a real-world problem.

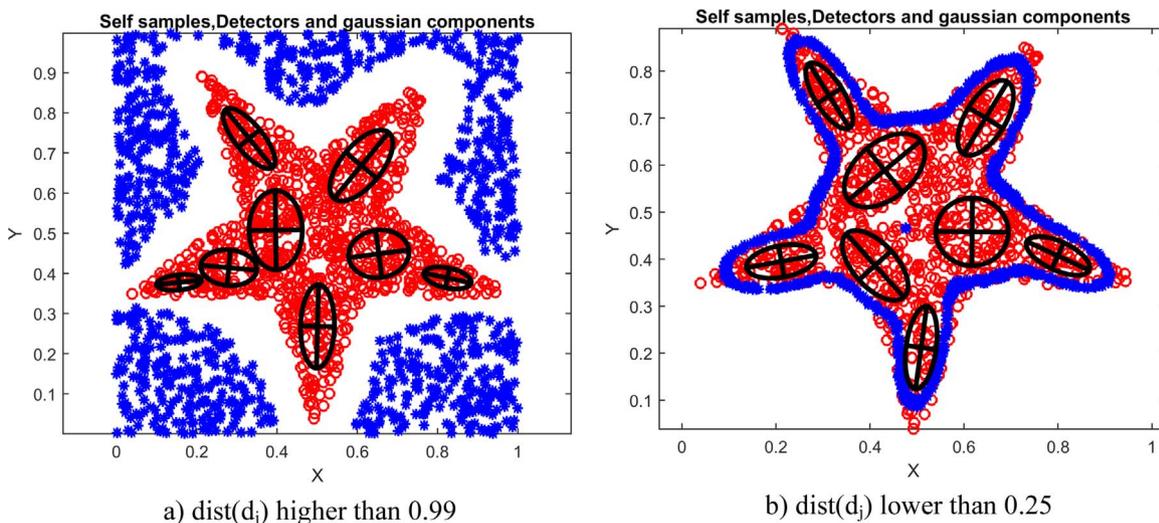


Fig. 12. Effect of the method described in Fig. 11 on the DENSA (stars are detectors and dots are self-samples).

value would result in high true positive rates, but in the same way, high false positive rates are attained. On the other hand, small Threshold_2 value would result low false positive rates, but low true positive rates too. Thus, it is easy to check that Threshold_2 can be easily tuned using a validation set for creating an efficient anomalous archaeological site detection system, which is a desirable system for archaeologists.

The promising results that are represented in Fig. 13 can be culturally interpreted, based on the history of the study area. Most

of the archaeological sites in the Silakhor Plain have a cultural sequence, and they have been used as settlements for human. This cultural sequence shows that the environmental potentials in the Silakhor Plain have led to an almost common settlement pattern among people in different periods that makes it possible to efficiently model the archaeological sites in this area by artificial intelligence algorithms. Artificial Intelligence is attracting widespread interest in many sciences because of its emerging robust detective capabilities. AI enables archaeologist to more fully

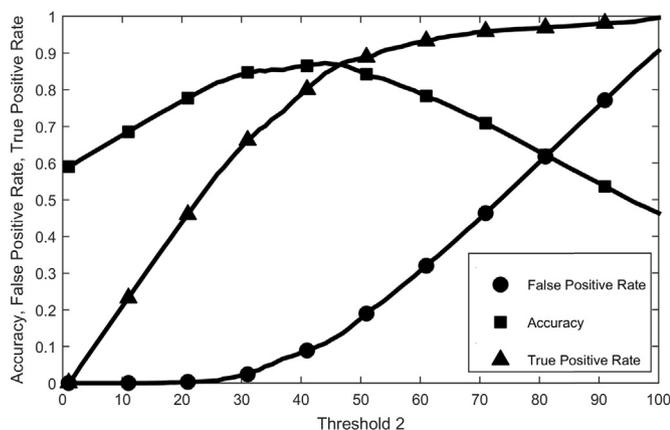


Fig. 13. Efficacy of the proposed method in detecting anomalous archaeological sites.

exploit knowledge from extensive amount of archaeological data and assists archaeologists in reasoning and making decisions that range from appropriate conservation and protection strategies to where best to excavate in a complex cultural landscape. The experimental results of this paper provide clear evidence that the application of negative selection algorithms, and specifically DENSA, represent an efficient potential for robust AI applications as detector of anomalous archaeological sites. It can ensure that archaeologist avoids time consuming, and expensive efforts to survey and excavate more archaeologically limited landscape areas for detecting anomalies. Although in this study the proposed algorithm represents a reasonable performance, we also highlight that there is no single method that for any data set represents the most accurate method (the No Free Lunch Theorem; [Alpaydin, 2009](#)). However, while DENSA may be successfully applied as an anomaly detector model in other semi-arid area, a deep investigation, as well as testing of a range of anomaly detection algorithms (with selection based on best performance against a test data set) should be conducted.

5. Conclusion and future works

This paper has introduced a new NSA termed DENSA that fitted a GMM on the normal space and modelled the self-space with a couple of flexible Gaussian components and generated dynamic number of detectors based on this GMM. DENSA compared randomly generated patterns to the GMM of self-space and then kept random patterns with low probabilities as detectors. The proposed algorithm was evaluated using different data sets. The experimental results, showed the efficiency of DENSA even in a real world application. Using GMM probabilities the detectors can be efficiently distributed in the non-self space, which led to efficient anomaly detection system in real world applications. Finally, in order to analyse the convergence and the running time of the proposed algorithm, a study has been conducted using the classical Time-To-Target plots, which present a standard graphical methodology for data analysis based on the comparisons between the empirical and theoretical distributions. Furthermore, this paper introduced a novel application of NSA in archaeology. Employing NSA in detecting anomalous archaeological sites can help archaeologist to detect anomalous sites, which do not follow normal behavior of other sites in their vicinities.

Trying to use other distribution estimation techniques such as One-Class SVM with a suitable objective function can be an interesting idea for future works. Another idea could be investigating the application of DENSA in generating artificial outliers for

one-class classification tasks where the outlier samples are rare, or using DENSA for systems that need to generate detectors for some certain reason (e.g., to distribute them or to initialize a learning classifier system).

Acknowledgment

The authors wish to thank the anonymous reviewers for having provided helpful and valuable feedback.

S.F. would like to express his warm thanks to Dr. Wenjian Luo from University of Science and Technology of China for his valuable, and inspiring comments.

References

- Aickelin, U., Bentley, P., Cayzer, S., Kim, J., McLeod, J., 2003a. Danger theory: the link between AIS and IDS? In: Proceedings of the 2nd ICARIS. Edinburgh, pp. 147–155.
- Aickelin, U., Bentley, P., Cayzer, S., Kim, J., McLeod, J., 2003b. Danger theory: the link between AIS and IDS? *Lect. Notes Comput. Sci.* 2787, 147–155.
- Aickelin, U., Cayzer, S., 2002. The danger theory and its application to artificial immune systems. In: Proceedings of the First International Conference on Artificial Immune Systems (ICARIS). Canterbury, pp. 141–148.
- Aickelin, U., Dasgupta, D., 2010. Artificial immune systems. In: Burke, E.K., Kendall, G. (Eds.), *Search Methodologies: Introductory Tutorials in Optimization and Decision Support Techniques*. Springer, New York, pp. 375–401.
- Aiex, R.M., Resende, M.G.C., Ribeiro, C.C., 2002. Probability distribution of solution time in GRASP: an experimental investigation. *J. Heuristics* 8, 343–373.
- Aiex, R.M., Resende, M.G.C., Ribeiro, C.C., 2007. TTTPLOTS: a perl program to create time-to-target plots. *Optim. Lett.* 1, 355–366.
- Alpaydin, E., 2009. *Introduction to Machine Learning*, second edition. MIT Press, Cambridge.
- Ayara, M., Timmis, J., deLemos, R., deCastro, L.N., Duncan, R., 2002. Negative selection: how to generate detectors. In: Proceedings of the 1st International Conference on Artificial Immune System (ICARIS'02). Canterbury, pp. 89–98.
- Balthrop, J., Esponda, F., Forrest, S., Glickman, M., 2002. Coverage and generalisation in an artificial immune system. In: Proceedings of the Genetic and Evolutionary Computation Conference. New York, pp. 3–10.
- Barceló, J.A., 2010. Computational intelligence in archaeology. State of the Art. In: Frischer, B., Webb, C.J., Koller, D. (Eds.), *Making History Interactive. Computer Applications and Quantitative Methods in Archaeology (CAA)*. Archaeopress, Williamsburg, pp. 11–21.
- Chambers, J.M., Cleveland, W.S., Kleiner, B., Tukey, P.S., 1983. *Graphical Models for Data Analysis*. Chapman & Hall, UK.
- Chmielewski, A., Wierzchon, S.T., 2012. Hybrid negative selection approach for anomaly detection. *Comput. Inform. Syst. Ind. Manag.* 7564, 242–253.
- Costanza, J., Cutello, V., Pavone, M., 2015. An immunological algorithm for combinatorial optimization: the fuel distribution problem as case study. *Int. J. Swarm Intell. Evolut. Comput.* 4 (118), 2–9.
- Cutello, V., Krasnogor, N., Nicosia, G., Pavone, M., 2007a. Immune algorithm versus differential evolution: a comparative case study using high dimensional function optimization. In: Proceedings of the International Conference on Adaptive and Natural Computing Algorithms, vol. LNCS 4431, pp. 93–101.
- Cutello, V., Morelli, G., Nicosia, G., Pavone, M., Scollo, G., 2011a. On discrete models and immunological algorithms for protein structure prediction. *Nat. Comput.* 10, 91–102.
- Cutello, V., Nicosia, G., Pavone, M., 2007b. An immune algorithm with stochastic aging and Kullback entropy for the chromatic number problem. *J. Comb. Optim.* 14, 9–33.
- Cutello, V., Nicosia, G., Pavone, M., Prizzi, I., 2011b. Protein multiple sequence alignment by hybrid bio-inspired algorithms. *Nucleic Acids Res.* 39, 1980–1992.
- Cutello, V., Nicosia, G., Pavone, M., Timmis, J., 2007c. An immune algorithm for protein structure prediction on lattice models. *IEEE Trans. Evolut. Comput.* 11, 101–117.
- Dasgupta, D., (Ed.), 2014. *Artificial Immune Systems and Their Applications*. Springer, Berlin.
- Deravignone, L., Jánica, G.M., 2006. Artificial neural networks in archaeology. *Archaeol. Calcolatori* 17, 121–136.
- D'haeseleer, P., Forrest, S., Helman, P., 1996. An immunology approach to change detection: algorithm, analysis and implications. In: Proceedings of the 1996 IEEE Symposium on Computer Security and Privacy. Los Alamitos, CA, pp. 110–119.
- Feo, T.A., Resende, M.G.C., Smith, S.H., 1994. A greedy randomized adaptive search procedure for maximum independent set. *Oper. Res.* 42, 860–878.
- Forrest, S., Perelson, A., Allen, L., Cherukuri, R., 1994. Self-nonspecific discrimination in a computer. In: Proceedings of the IEEE Symposium on Research in Security and Privacy. Oakland, pp. 16–18.
- Fouladvand, S., Osareh, A., Shadgar, B., 2015. Distribution Estimation Based Negative

- Selection Algorithm (DENSA). In: Proceedings of the International Congress on Systems Immunology, Immunoinformatics & Immune-computation. Taormina, Italy, pp. 1–7.
- Fraley, C., Raftery, A.E., 1998. How many clusters? Which clustering method? Answers via model-based cluster analysis. *Comput. J.* 41, 578–588.
- Goldsby, R., Kindt, T., Osborne, B., Kuby, J. (Eds.), 2003. *Immunology, fifth edition*. W. H. Freeman, New York.
- Gomez, J., Gonzalez, F., Dasgupta, D., 2003. An immuno-fuzzy approach to anomaly detection. In: Proceedings of the 12th IEEE International Conference on Fuzzy Systems. Missouri, pp. 25–28.
- Gonzalez, F., Dasgupta, D., Kozma, R., 2002. Combining negative selection and classification techniques for anomaly detection. In: Proceedings of the 2002 CEC. Honolulu, pp. 12–17.
- Gonzalez, F., Dasgupta, D., Nino, L. F., 2003. A randomized real-valued negative selection algorithm. In: Proceedings of the 2nd International Conference on Artificial Immune Systems (ICARIS). Edinburgh, pp. 261–272.
- Gonzalez, L.J., Cannady, J., 2004. A self-adaptive negative selection approach for anomaly detection. In: Proceedings of the 2004 Congress of Evolutionary Computation (CEC-2004). Portland, pp. 1561–1568.
- Han, J., Kamber, M., Pei, J., 2006. *Data Mining: Concepts and Techniques, third edition*. Morgan Kaufmann, Burlington.
- Hang, X., Dai, H., 2004. Constructing detectors in schema complementary space for anomaly detection. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO), Lecture Notes in Computer Science, vol. 3102, pp. 275–286.
- Hole, F., 1970. The Palaeolithic culture sequence in Western Iran. *Actes du VII Congrès International des Sciences Préhistoriques et Protohistoriques*. Pruge, pp. 286–292.
- Ji, Z., 2006. *Negative Selection Algorithms: From the Thymus to V-Detector* (Ph.D. dissertation). Department of Computer Science, University of Memphis, Memphis.
- Kim, J., Bentley, P.J., Aickelin, U., Greensmith, J., Tedesco, G., Twycross, J., 2007. Immune system approaches to intrusion detection – a review. *Nat. Comput.* 6, 413–466.
- Luo, W., Zhang, Z., Wang, X., 2006. A heuristic detector generation algorithm for negative selection algorithm with hamming distance partial matching rule. *Lect. Notes Comput. Sci.* 4163, 229–243.
- Maghsoudi, M., Sharafi, S., Sharafi, F., 2014. Natural factors affecting the distribution pattern of archaeological sites in the Silakhor Plain, Lorestan Province. *Geogr. Reg. Dev. J.* 12 (22).
- Nabney, I.T., 2004. *Netlab Algorithms for Pattern Recognition, third edition*. Springer, London.
- Osareh, A., 2004. Automated Identification of Diabetic Retinal Exudates and the Optic Disc (Ph.D. dissertation). Department of Computer Science, University of Bristol, Bristol.
- Pavone, M., Narzisi, G., Nicosia, G., 2012. Clonal selection: an immunological algorithm for global optimization over continuous spaces. *J. Glob. Optim.* 53, 769–808.
- Poggiolini, M., Engelbrecht, A., 2013. Application of the feature-detection rule to the negative selection algorithm. *Expert Syst. Appl.* 40, 3001–3014.
- Puyol-Gruaiart, J., 1999. Computer science, artificial intelligence and archaeology. In: Barceló, J.A., Briz, I., Vila, A. (Eds.), *New Techniques for Old Times. CAA98. Computer Applications and Quantitative Methods in Archaeology, Proceedings of the 26th Conference, Barcelona, March 1998 (BAR International Series 757)*. Archaeopress, Oxford, pp. 19–28.
- Richards, J.D., 1998. Recent trends in computer applications in archaeology. *J. Archaeol. Res.* 6, 331–382.
- Singh, S., 2002. Anomaly detection using negative selection based on the r-continuous matching rule. In: Proceedings of the 1st International Conference on Artificial Immune Systems (ICARIS'02). Canterbury, pp. 99–106.
- Smith, S., Timmis, J., 2008. Immune network inspired evolutionary algorithm for the diagnosis of Parkinsons disease. *Biosystems* 94, 34–46.
- Stibor, T., Timmis, J., Eckert, C., 2005. A comparative study of real-valued negative selection to statistical anomaly detection techniques. In: Proceedings of the ICARIS. Banff, pp. 262–275.
- Tavallaee, M., Bagheri, E., Lu, W., Ghorbani, A.A., 2009. A detailed analysis of the KDD cup 99 data set. In: Proceedings of the 2nd IEEE Conference on CISDA. Ottawa, pp. 8–10.
- Vitali, V., 1991. Formal methods for the analysis of archaeological data: Data analysis vs expert systems. In: Lockyear, K., Rahtz, S.P.Q. (Eds.), *Computer Applications and Quantitative Methods in Archaeology. BAR International Series, Oxford*, pp. 207–209.
- Voorrips, A., 1990. New tools from mathematical archaeology. In: Presented at the 5th International Symposium on Data Management and Mathematical Methods in Archaeology. Scientific Information Centre of the Polish Academy of Sciences, Warsaw, Poland.
- Wang, Y., Luo, W., 2009. PTS-RNSA: A novel detector generation algorithm for real-valued negative selection algorithm. In: Proceedings of the IJCBS. Shanghai, pp. 577–583.
- Wierzchon, S.T., 2000. *Generating Optimal Repertoire of Antibody Strings in an Artificial Immune System. Intelligent Information System, Advances in Soft Computing Series of Physica Verlag. Physica-Verlag, Heidelberg, New York*, pp. 119–133.
- Zhang, J., Luo, W., 2011. On the self representation based on the adaptive self radius. In: Proceedings of the 4th IWACI. Hubei, pp. 157–163.
- Zhang, J., Luo, W., 2014. EvoSeedRNSAI: an improved evolutionary algorithm for generating detectors in the real-valued negative selection algorithms. *Appl. Soft Comput.* 19, 18–30.
- Zeng, J., Tang, W., Liu, C., Hu, J., Peng, L., 2012. Real-valued negative selection algorithm with variable-sized self radius. *Inform. Comput. Appl.* 7473, 229–235.