The Use of Nature Inspired Methods in Electrocardiogram Analysis

Miroslav Burša, Michal Huptych, Lenka Lhotská, *Member IEEE*

***Abstract*—Holter ECG analysis (long-term ECG monitoring over more than 24 hours) requires a huge amount of computational resources to process such recordings and evaluate them in acceptable time. Promising approach is to perform automatic clustering and present to the cardiologist only several representative cardiac beats contained in the data. Thus the interpretation can be performed much faster. Exhaustive clustering methods are very resource demanding (computer time, memory), thus heuristics should be used, which would speed up the process having acceptable resource consumption and/or yielding better results. The nature inspired methods are currently subject of intensive research. This paper presents comparison of the use of nature inspired methods (Ant Colony Clustering and Kohonen self-organizing neural network) with traditional methods (k-nearest neighbor and k-means) and compares the robustness of such methods and their ability to cope with real data obtained from ECG recordings on the MIT-BIH database. The both Ant Colony Clustering and Kohonen self-organizing neural network achieved better results than k-means algorithm.**

I. INTRODUCTION

Nature inspired methods play an important role in artificial intelligence domain. These methods have their origins in nature (in the behavior or life of animals, insect or other species, originating in physiological, genetic, communication, information processing and other processes). Remember that the well-known and widely used genetic algorithms have their origin in the nature as well.

The nature inspired methods have received special attention from the research community over the recent years. It is because these methods are particularly suitable to perform exploratory data analysis, and also because there is still a lot of investigation to perform on this field – the research nowadays concentrates on improving performance, stability, convergence, speed, robustness and other key features that would allow using these methods in real applications. The main research on the nature inspired methods does not focus on the strict modeling of the natural processes; it merely focuses on using the best ideas to improve the convergence and accuracy of such methods.

Manuscript received June 30, 2006. This work has been supported by the research program "Information Society" under grant No. 1ET201210527 "Knowledge-based support of diagnostics and prediction in cardiology".

M. Burša is a PhD. student on the Czech Technical University in Prague, Czech Republic, dept. of Cybernetics, Gerstner laboratory. (contact e-mail: bursam@fel.cvut.cz).

M. Huptych is a PhD. student on the Czech Technical University in Prague, Czech Republic, dept. of Cybernetics, Gerstner laboratory.

L. Lhotská is an assoc. professor on the Czech Technical University in Prague, Czech Republic, dept. of Cybernetics, Gerstner laboratory.

This paper provides an evaluation of the use of such nature inspired methods for electrocardiogram (ECG) classification. ECG is a body surface recording of the electric heart signal. ECG signal analysis plays an important role in patient diagnosis process (and of course in maintaining healthy lifestyle and heart disease prevention). The complexity of the signal (inhering many factors) makes an automated analysis of the signal very difficult to perform. Although it is possible to perform the analysis in real-time, for long-term monitoring, the speed of analysis is crucial. (More than five minutes of processing a 24-hour recording is considered unacceptable for physicians in the clinical praxis.)

The objective of clustering algorithm is to find similar classes in spatially distributed data. There are no efficient solutions known to clustering problem and some formulations of the problem are even NP-hard [4]. All the clustering methods compared in this paper have its origin based on the observation of nature. In this paper we perform comparison of two main clustering methods: Ant Colony Clustering algorithm (also known as "Brood sorting" or "Clustering algorithm based on Swarm Intelligence (CSI)") and Kohonen self-organizing neural network (self-organizing map, SOM).

The nature inspired methods are evaluated and compared with classical methods of clustering (k-means, k-nearest neighbor). The task of this work is to apply all the clustering methods to real ECG data and to evaluate their performance in ECG data analysis.

II. METHODS

In all the clustering methods described, the Euclidean metrics (also called L_2 norm) is used. For any two vectors $P = (p_0, p_1, \ldots, p_n)$ and $Q = (q_0, q_1, \ldots, q_n)$ of the same dimension n (where every (i-th) component $p_i \in \mathbf{R}$ and $q_i \in \mathbf{R}$), the Euclidean distance can be computed as follows:

$$d = \sqrt{\sum_{i=0}^{n} (p_i - q_i)^2} \quad (1),$$

The following described methods are the methods for performing data clustering. Its task is to arrange input vectors (clustered data) into disjoint subsets where the data inside the subset are similar to each other as much as possible (in the terms of Euclidean metrics (1)). The subsets

should be different from each other as much as possible. These groups are called clusters. Note that the number of natural clusters which are present in the data is not known in advance. This means that for example data of normal heart beats of each usually form a natural cluster in the data.

A. Ant Algorithms

M. Dorigo presented in [2] an Ant System and Ant Colony Optimization which is a meta heuristic approach based on the foraging behavior (a positive motivation) of real ants. It is based on the parameterized probabilistic model – the pheromone model. Deneubourg et al. [1] proposed the basic approach for Ant Clustering approach.

For example, the *Messor sancta* ants organize dead corpses into clusters; brood sorting has been studied in ant colony of *Leptothorax unifasciatus*.

While the behavior of individual ant is very primitive, the resulting behavior on the colony level can be quite complex. The Ant Colony methods use so called (positive) stigmergic cooperation, which is a form of information shared by the agents. This information can be stored in either in the form of pheromone trails (Ant System) or directly in the environment (Ant Colony Clustering).

Both the methods show an agent-like approach with stigmergic cooperation.

The Ant Colony Clustering works as follows. First the data vectors are randomly scattered onto a two-dimensional grid (usually toroid one) (analogy of the real world of an ant). Ants (agents) are then also randomly placed onto the two-dimensional grid. In each iteration step an ant searches its neighborhood and computes a probability of picking up a vector (if the ant is unloaded and steps onto a vector) or of dropping down a vector (if the ant is loaded and steps onto a free grid element). The ant moves randomly.

The probability of the ant to pick up an object is given by following equation:

$$P_p = \left(\frac{k_1}{k_1 + f} \right)^2 \quad (2),$$

where f is the perceived fraction of items in the neighborhood of the agent (4), and k_1 is a threshold constant. If f is much greater than k_1, the probability is close to 1, thus the probability of picking up the item is high.

The probability of dropping down the carried object is given by following equation:

$$P_d = \left(\frac{f}{k_2 + f} \right)^2 \quad (3),$$

where k_2 is another threshold constant. Again with high values of f (much greater than k_2) the probability is close to 1, thus the probability of dropping the vector is high.

The function f (might be called a similarity measure) and is computed as follows: Assume agent located at site r finding a vertex v_i at that site.

$$f(v_i) = \begin{cases} \frac{1}{s^2} \sum_{v_j \in Neigh(s \times s)(r)} \left[1 - \frac{d(v_i, v_j)}{\alpha} \right] & \text{if } f > 0 \\ 0 & \text{otherwise} \end{cases} \quad (4),$$

where $f(v_i)$ is a measure of the average distance within the graph of element v_i to other elements v_j present in the neighborhood of v_i. Parameter α defines the scale for dissimilarity. If (for example) the neighborhood contains maximum of vectors identical to v_i, then the function $f(v_i)=1$ thus the vertex should not be picked up. On the other hand, when the surrounding contains no vertices, the vector will be picked up with high probability ($f(v_i)=0$. For ant carrying a data vector, the same equation (4) is used. The v_i represents the vector being carried by the ant.

B. SOM neural networks

The Kohonen self-organizing network (SOM) [6] is a single layer feed-forward network where the output neurons are arranged in low dimensional (usually 2D, sometimes 3D) structure that can be easily visualized. Each input of the network is distributed to all output neurons. There is a weight vector $\overline{W}(t)$ attached to each neuron having the same dimensionality as the input vectors. The number of input dimensions is usually much greater than the output grid dimension. Kohonen self-organizing networks are mainly used for dimensionality reduction rather than expansion, working similarly to PCA (primary component analysis) [7] or ICA (independent component analysis) [8].

The goal of the learning process in the self-organizing map is to associate different parts of the SOM lattice to respond similarly to certain input patterns. This is partly motivated by how visual, auditory or other sensory information is handled in separate parts of the cerebral cortex in human brain. Unlike the most of artificial neural networks, this neural network is trained by unsupervised training. This means that the neural network is given no information about the correct class of the input vector.

The learning algorithm is an iterative learning algorithm, which works as follows. The state before the learning process starts is illustrated in Fig. 1. In each iteration, a vector is presented to the network input and the best matching unit (BMU) with the lowest Euclidean distance (1) is determined. Then the weight vector $\overline{W}(t)$ of the BMU and all neurons in its neighborhood are updated as follows:

$$\overline{W}(t+1) = \overline{W}(t) + \overline{\Theta}(v,t)\alpha(t)d(\overline{D}(t), \overline{W}(t)) \quad (5),$$

where $\alpha(t)$ α(t) is monotonically decreasing learning

coefficient (decreasing with iterations) and $\overline{D}(t)$ is the input vector. The neighborhood function $\overline{\Theta}(v,t)$ depends on the lattice distance between the BMU and neuron *v*. Note that function $d(\overline{D}(t),\overline{W}(t))$ computes Euclidean distance between the two vectors (1). In the simplest form it gives value of one (1) for all neurons close enough to BMU and zero for others, but a Gaussian function is also an usual choice too. The neighborhood function shrinks with time (iterations). At the beginning when the neighborhood is broad, the self-organizing takes place on the global scale. When the neighborhood has shrunk to just a couple of neurons, the weights are converging to local estimates. The convergence result is illustrated on Fig. 2. Note, that the number of neurons remains the same through the whole learning process.

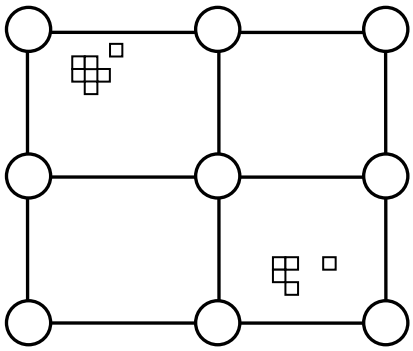

Fig. 1 Kohonen self-organizing neural network (SOM) with nine neurons before the learning process starts. Circles represent neurons, squares represent data vectors.

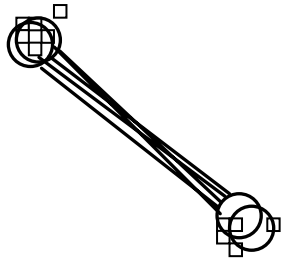

Fig. 2 Kohonen self-organizing neural network (SOM) after the learning process has been performed. Circles represent neurons, squares represent data vectors. The number of neurons stays the same; they are only visualized one above another one.

C. K-means

K-means is an algorithm for partitioning *n* data points into *K* disjoint subsets S_j containing N_j clusters. The goal is to divide the objects into K clusters such that some metric relative (Euclidean distance (1) in our case) to the centroids of the clusters is minimized. The formula to be minimized is the following one:

$$Err = \sum_{j=1}^{K} \sum_{m \in S_j} \left| x_m - \mu_j \right|^2 \quad (3),$$

where x_m is a vector representing the *m*-th data point and μ_j is the geometric centroid of the data points in S_j. In general, the algorithm does not achieve a global minimum of the error function *Err* over the assignments. In fact, since the algorithm uses discrete assignment rather than a set of continuous parameters, the "minimum" it reaches cannot even be properly called a local minimum. Despite these limitations, the algorithm is used fairly frequently as a result of its ease of implementation.

In the initial phase, the centroids are positioned randomly. In the first part of iterative step, all the input vectors are processed and every vector is assigned to the nearest centroid. The second part of the iterative step consists of recomputing mean value of each dataset S_j and moving the centroid to this new position. Then these two substeps are iteratively repeated until no change occurs (or until a predefined iteration limit is reached). The centroid is then designed to the major class contained in the cluster (if known).

The process looks similar to Kohonen networks processing (see Fig. 1 and Fig. 2) except that there is no connection between centers and no neural network approach is used.

One disadvantage of this process is that the number of clusters *K* has to be known *a priori*. For different number of clusters *K* the algorithm has to be rerun.

D. Nearest neighbor (k-NN)

The nearest neighbor method (k-NN) is a hierarchical clustering method. During the clustering process a hierarchy of nearest-neighbors is created.

The hierarchy is created as follows. In the first step every input data vector represents one cluster. In the next step, the two most similar clusters are merged. The similarity (or least distance) is computed via equation (1) (Euclidean distance). This merging step is repeated until desired number of clusters is reached.

If a history of these cluster hierarchy during the whole process is conserved, the different number of clusters can be used for classification with no need to be recomputed.

III. Experiment

A. Preliminary parameter estimation

For basic parameter estimation and implementation testing the well-known iris-dataset has been used (publicly available). This dataset contains three classes, each containing 50 vectors. Contains one well separable class, others are more difficult to separate.

B. Input Data

The features extracted are the basic ECG heart action parameters. Input signals are taken from widely used MIT-BIH database [3], which contains annotated records. In [3], certain description of the data can also be found (together with some basic anonymous description of the patients). For the sake of simplicity, only two distinct classes have been used: normal cardiac action and abnormal cardiac action. The classification into more classes is nearly impossible due to lack of the data (mainly abnormal heart action signal) in some signals. In this way (by merging all the abnormal heart actions into one class) more records from the MIT-BIH database can be processed.

As an input record, the record No. 106 has been selected. This record contains 1505 instances of normal heart action and 518 instances of abnormal heart action. For the method comparison a random selection from this file has been used and for the comparison a file of 500 instances for both classes has been evaluated. The file contains more independent natural clusters.

C. Extracted Features

From the ECG signal, the following eight features have been automatically extracted (see [5]): amplitudes of Q, R, S, positive T and negative T wave, amplitude ratio of Q/R, R/S and R/T waves. For processing, three features have been normalized.

IV. Results

Table I shows an average sensitivity and specificity for all methods evaluated. All results have been first clustered into four classes (the data contains four natural classes); classification of each class has been determined by the major class representation in the cluster.

The best results have been achieved by the k-NN method, which is the only method not using centroid approach. The nature inspired methods, however, outperforms the k-means algorithm both in specificity and sensitivity and achieves more stable results (in term of standard deviation).

TABLE I
METHOD COMPARISON

Method \ Result	Sensitivity	Specificity
K-means	77 %	65 %
K-NN	96 %	74 %
Ant Colony Clustering	79 %	68 %
Kohonen SOM	79 %	66 %

V. Discussion

Both the k-means algorithm and Kohonen neural network can cope with one new unknown input vector added. After the iterative (learning) algorithm has stopped, these algorithms can classify the new vector without any modification. For the other methods, ACO Clustering and k-nearest neighbor, this is a problem. The new vector cannot be easily classified by these algorithms. The algorithm must be restarted or there must be some heuristics used (like the comparison with the average sample of the cluster, etc.).

VI. Conclusion

The results have shown that the performance of the nature inspired method is comparable and in some cases outperforms the classical methods, but it depends on the type of clustering method. The centroid methods can get stuck in the local minima. The nature inspired methods however demand a lot computing time, which makes it difficult for these methods to be more widely used.

The advantage of Ant Clustering and Kohonen network is that it can determine natural clusters within data. The drawback (mainly for Kohonen network) is its high computing resources consumption.

VII. Future work

Future work might be focused on the automatic setting of nature inspired methods parameters. This process would surely make these methods less complicated to use. Additional evaluation of best number of clusters should be performed. This means that first the number of natural clusters in the data should be evaluated, and then the classification using such number of clusters should be performed; finally the classification would be performed.

References

[1] J. L. Deneubourg, S. Goss, N. Franks, A. Sendova-Franks, C. Detrain, L. Chrétien. The Dynamics of Collective Sorting: Robot-like Ants and Ant-like Robots. From Animals to Animats: Proc. of the 1st International Conference on Simulation of Adaptive Behaviour 1990:356-353.

[2] Dorigo, M., Di Caro, G., Gambarella, L. M. Ant Algorithms for Discrete Optimization. Artificial Life 5, 1999:137-172.

[3] Goldberger AL, Amaral LAN, Glass L, Hausdorff JM, Ivanov PCh, Mark RG, Mietus JE, Moody GB, Peng CK, Stanley HE. PhysioBank, PhysioToolkit, and PhysioNet: Components of a New Research Resource for Complex Physiologic Signals. Circulation 101(23):e215-e220 (http://circ.ahajournals.org/cgi/content/full/101/23/e215]; 2006)

[4] E. Forgey. A Cluster Analysis of Multivariate Data: Efficiency vs. Interpretability of Classification, Biometrics, 1965:21, p. 768

[5] Chudáček V, Lhotská L., Unsupervised Creation of Heart Beats Classes from Long-term ECG Monitoring, Biennial International EURASIP Conference BIOSIGNAL, 18:199–201, 2006

[6] T. Kohonen, Self-organised Formation of Topologically Correct Feature Maps. Biol. Cybernet. 43 (1982), pp. 59–69.

[7] R. O. Duda, P. E. Hart, and D. G. Stork. Pattern Classification. John Wiley and Sons, 2001

[8] A. Hyvärinen. Survey on Independent Component Analysis. Neural Computing Surveys, 2:93–128, 1999