International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:1, No:8, 2007

Genetic Algorithm for Feature Subset Selection with Exploitation of Feature Correlations from Continuous Wavelet Transform: a real-case Application

G. Van Dijck, M. M. Van Hulle, and M. Wevers

***Abstract*—**A genetic algorithm (GA) based feature subset selection algorithm is proposed in which the correlation structure of the features is exploited. The subset of features is validated according to the classification performance. Features derived from the continuous wavelet transform are potentially strongly correlated. GA's that do not take the correlation structure of features into account are inefficient. The proposed algorithm forms clusters of correlated features and searches for a good candidate set of clusters. Secondly a search within the clusters is performed. Different simulations of the algorithm on a real-case data set with strong correlations between features show the increased classification performance. Comparison is performed with a standard GA without use of the correlation structure.

***Keywords*—**Classification, genetic algorithm, hierarchical agglomerative clustering, wavelet transform.

I. Introduction

In the design of a pattern recognition system possibly a large set of features is computed. Many of the computed features can be irrelevant to distinguish between the different classes. Selection of the most relevant features can increase the interpretability of the data under study. Moreover, given a finite learning sample size for a classification algorithm, the choice of a too high dimensional feature vector can seriously degrade the performance of a classification algorithm [1], [2]. This paper describes a GA based algorithm to select important features when some features are highly correlated. Especially features derived from the continuous wavelet transform (CWT) [3] are candidates for the proposed GA based feature subset selection algorithm. In the computation of features from the CWT for pattern recognition a continuous series of scale parameters is considered. Differences in wavelet coefficients from neighboring scales are typically relatively small. Therefore features that exploit the different scales are potentially highly correlated. The proposed algorithm is demonstrated on a real-case application [4]. In [4] three different sets of features are computed to characterize different corrosion processes from electrochemical noise measurements. Two sets of features are computed based on the continuous wavelet transform. A third set is computed based on the power spectral density (PSD) of electrochemical voltage time series.

Manuscript received November 19, 2004. This work was supported by the Institute for the Promotion of Innovation through Science and Technology in Flanders (IWT-Vlaanderen).

G. Van Dijck is with the Computational Neuroscience Research Group of the Laboratory of Neurophysiology and the Department of Metallurgy and Materials Engineering (MTM), K.U. Leuven, 3000 Leuven, Belgium (phone: +32 16 32 12 47; fax: +32 16 32 19 90; e-mail: gert.vandijck@mtm.kuleuven.ac.be).

M.M. Van Hulle is with the Computational Neuroscience Research Group of the Laboratory of Neurophysiology, K.U. Leuven, 3000 Leuven, Belgium, (e-mail: marc@neuro.kuleuven.ac.be).

He is supported by research grants received from the Belgian Fund for Scientific Research -- Flanders (G.0248.03 and G.0234.04), the Flemish Regional Ministry of Education (Belgium) (GOA 2000/11), and the European Commission (IST-2001-32114 and IST-2002-001917).

M. Wevers is with the Department of Metallurgy and Materials Engineering (MTM), K.U. Leuven, 3000 Leuven, Belgium, (e-mail: martine.wevers@mtm.kuleuven.ac.be).

She is supported by research grants received from the Belgian Fund for Scientific Research -- Flanders (G.0248.03).

II. Related work

Feature subset selection algorithms can be classified into two main categories: the w*rapper* approach and the *filter* approach [5], [6]. In the filter approach the feature selection is done independently of the learning algorithm of the classifier. This is computationally more efficient but ignores the fact that an optimal selection of features may be dependent on the learning algorithm. On the other hand the wrapper method is computationally more involved, but takes the dependence of feature subset on the leaning algorithm of the classifier into account. The proposed algorithm in this paper belongs to the class of wrapper methods by using the learning algorithm in the evaluation of the performance of the feature subset.

GA's belong to the class of randomized heuristic search techniques. In feature subset selection problems they perform a population-based probabilistic or randomized sampling in feature space. Different approaches in literature have been proposed for feature subset selection based on GA's and are often tailored towards specific problem domains [7]-[9]. This paper envisages pattern recognition applications where features are derived based on the CWT and therefore are potentially highly correlated. GA's that do not take the

correlation structure of the data into account, are potentially more time-consuming due to the addition or replacement of highly correlated features in a subset of features.

III. The Algorithm

A. Fitness function in the GA's

The quality of a particular selection of features, further called genotype [10], is determined by its classification performance. Therefore it is sound for the fitness function to be a function of the classification performance. To increase selective pressure [10] compared to the classification performance as a fitness function, a cosh function of the averaged classification performance is taken:

$$f(i) = (\cosh(ACP(i)) - 1)^n \quad (1)$$

where $f(i)$ is the fitness value for genotype 'i', $ACP(i) \in [0,1]$ the averaged classification performance of genotype 'i' and $'n' \in \mathbb{R}$ is a parameter that controls selective pressure. A higher $'n'$ increases selective pressure. The classification performance $CP(i)$ is determined by: the classification algorithm, the learning set and test set. Therefore $CP(i)$ is a random variable. As a validation rule 10-fold crossvalidation was chosen. The average of $CP(i)$ over different runs of 10-fold crossvalidation is $ACP(i)$. In this paper a Bayesian classifier is opted for. In a Bayesian classifier observation $\mathbf{x_k}$ is assigned to the class with maximum posterior class probability:

$$j = \arg\max_i P(C_i \mid \mathbf{x_k}) = \frac{P(\mathbf{x_k} \mid \boldsymbol{\theta}, C_i) P(C_i)}{P(\mathbf{x_k})}. \quad (2)$$

Observation $\mathbf{x_k}$ takes concrete values for genotype k. C_i represents the class and $\boldsymbol{\theta}$ is the set of parameters that determine the class conditional density $P(\mathbf{x_k} \mid \boldsymbol{\theta}, C_i)$.

Density estimations are obtained by the minimum message length (MML) criterion for Gaussian mixture models (GMM) [11]. More details on the use of the GMM in the design of a classifier can be found in [4]. In a naïve-Bayesian classifier density estimations are factorized under assumption of independence of features. In this paper correlations between features are assumed. Therefore the assumptions for a naïve-Bayesian classifier are strictly speaking violated.

B. Description of the algorithm

An overview of the different steps in the algorithm is presented in figure 1.

1) Forming clusters of features

The proposed algorithm exploits the correlations between features in a first step by forming clusters of features.

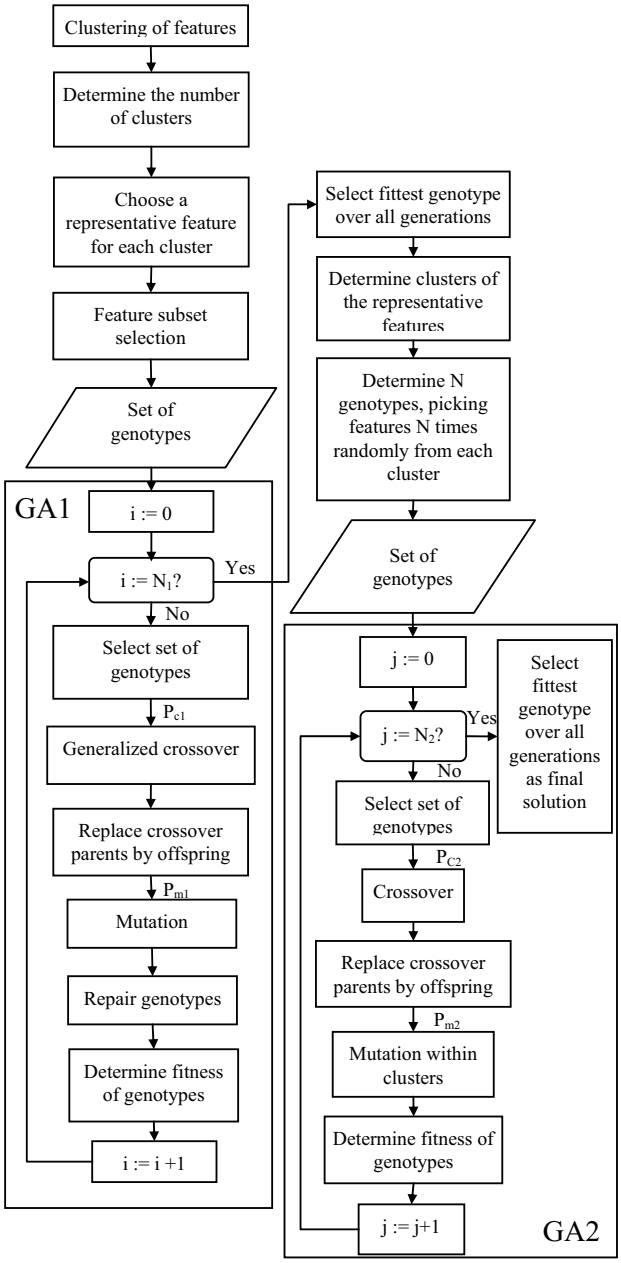

Fig. 1. Flowchart of the 2-stage GA based feature subset selection algorithm, GAFCO (**G**enetic **A**lgorithm for **F**eature Subset Selection with Exploitation of Feature **Co**rrelations). In a first step clusters of features are formed. Highly correlated features end up in the same cluster. In the first GA, GA1, a search is made among good candidate clusters. Only the best set of clusters found in N_1 generations is considered in the second GA, GA2. The second GA searches within the best set of clusters for good candidate features. In the second GA only 1 feature from every cluster is considered.

The distance between features F_α and F_β is computed as:

$$dist(F_\alpha, F_\beta) = 1 - \sum_i \frac{(F_\alpha(i) - mean(F_\alpha))}{std(F_\alpha)} \cdot \frac{(F_\beta(i) - mean(F_\beta))}{std(F_\beta)} \quad (3)$$

where $F_\alpha(i)$ represents the feature value of feature α for observation i, "mean" denotes the mean and "std" the standard deviation of the feature values. From (3) one can notice that highly correlated features are near in distance.

In the application of the algorithm to the real-case data set, hierarchical agglomerative clustering was used [12]. A hierarchical method allows a user of the algorithm to decide interactively upon the number of clusters. In order to automate finding the number of clusters, different criteria have been investigated [13]. These criteria, however, often include user-defined parameters and performance is highly dependent on their choice. These criteria are used in retrieval of the number of 'naturally' underlying clusters such as different populations. In this case clustering is applied to limit search space, see (5), and not to determine the 'natural' underlying populations, which would rather correspond to the number of different computations needed to extract features e.g. the wavelet transform, Fourier transform, AR-modelling, … . Use of these criteria therefore is not advised. Our criterion is considered in paragraph V.

2) Cluster selection with a first GA

In a first GA a good combination of clusters is searched for. For numeric evaluation of the 'goodness' of a combination of clusters a decision upon a feature representing a cluster needs to be taken. In the application the feature closest to the cluster center is chosen.

Note that a high computational cost is involved in determining the fitness values of genotypes: GMM-based density estimation for different classes, repeated ten times for 10-fold crossvalidation. This is repeated several times to compute the average classification performance ACP. Instead of re-running the GA for different numbers of combinations of clusters, the GA decides upon the number of clusters to be considered. The number of clusters is considered implicitly in the fitness evaluation. A small number of combinations of clusters is insufficient to express the differences between the different classes, and therefore has a lower classification performance and hence a lower fitness value. However choosing too many features decreases classification performance. Addition of features to an existing set of features can decrease the asymptotic probability of misclassification (PMC). The asymptotic PMC P_∞ is the probability of misclassification given an infinite learning sample size. Addition of features however requires more parameters to be estimated. Inaccurate estimation of parameters increases the classification error. If the increase of error is larger than the decrease in error by the addition of new features, the net effect is a decrease of classification performance.

The first GA starts from a set of random combinations of clusters.

In a first step of GA1 the genotypes are chosen with probability P_{c1} to be used in the crossover operator. Note that possibly genotypes of different lengths, namely a different number of features, are paired. Therefore a generalized crossover operator is introduced. First the smallest genotype is shifted a random number of positions against the longer genotype. A shift of 0 positions results in an alignment of the first feature of the smaller genotype with the first feature of the longer genotype. A maximum shift of L2-L1positions results in an alignment of the last features from both genotypes, where L2 is the length of the longest genotype and L1 of the smaller genotype. A crossover position is chosen randomly within the smaller genotype. This generalized crossover operation results in 2 offspring with potentially different lengths from the parents. The parents are replaced by their offspring.

Next a mutation operator is applied to the genotypes. Every feature of a genotype has a probability P_{m1} to be mutated. After mutation and crossover a genotype possibly contains the same cluster multiple times. This does not result in additional information to separate the classes. Neighboring clusters then replace the repeated clusters. Additionally the features are sorted so that every combination of clusters has a unique representation. The fitness of the genotypes is determined by (1). This implies several runs of 10-fold crossvalidation.

Roulette wheel selection [10] is applied to determine which genotypes pass to the next generation. The elitist model [10] is applied to preserve the best genotype.

When the limiting number of generations N_1 is reached, the best combination of clusters found so far is considered for further investigation in a second GA, GA2.

3) Search for features within clusters with a second GA

In GA2 the number of clusters and which ones to consider are fixed. GA2 starts from N genotypes. Each genotype consists of exactly 1 feature from each cluster found in GA1. These features are chosen randomly from the clusters for initialization.

Because all genotypes have the same length there is no need for a generalized crossover operation.

In the mutation operation, every feature is only allowed to be mutated into a feature from the same cluster. This implies that every cluster is preserved within each genotype. Note that there is no need for a repair method. Once sorted, the features stay sorted, and no multiple occurrences of the same feature can be present, while only 1 feature from every cluster is allowed.

The fitness evaluation is performed in the same manner as in GA1. When N_2 generations are reached the best solution found is the final solution.

IV. Discussion

The proposed algorithm first searches between clusters of features. This implies a coarse grain sampling in feature space. No time is lost in a search for features that are highly correlated, when an appropriate choice of the number of

clusters is chosen. In a 2nd phase the clusters are opened and a local search within the clusters is performed. This implies a fine grain sampling in feature space. A very similar search strategy is performed in simulated annealing.

The number of clusters is an important parameter to be considered.

The higher the number of clusters the more representative the representative features become to represent a cluster.

However when the number of clusters is equal to the number of features, the algorithm degenerates to a standard GA (SGA) with variable length of genotypes. In that case the correlation structure between features is not exploited.

The proposed algorithm limits the number of possible solutions. In the standard GA (SGA), the number of possible subsets c_{SGA} equals:

$$c_{SGA} = \sum_{i=1}^{k} \binom{f_c}{i} \tag{4}$$

where it is assumed that the initial population contains genotypes of length 1 until length k. f_c is the total number of features and $\binom{f_c}{i}$ the number of combinations of i out of f_c. In the 2-stage GA combinations of features within a cluster are not possible and therefore the number of possible outcomes equals:

$$c_{GAFCO} = \sum_{i=1}^{k} \binom{f_c}{i} - \sum_{p=1}^{CC} \sum_{\substack{j=2 \\ j<=N_p}}^{k} \binom{N_p}{j} \tag{5}$$

where CC is the cluster count, and N_p the number of features in cluster 'P'.

V. Application to a Real-Case Data Set

Details of the data set can be found in [4]. Four different classes are considered in the data set for which a (sub-)optimal set of features has to be determined to classify the time series f(t). A first set of features is derived from the continuous wavelet transform:

$$x(k) = \frac{std_{F(b)_{acc,k}<0}(F(b)_{acc,k})}{std_{F(b)_{acc,k}>0}(F(b)_{acc,k})}, \tag{6}$$

$$\text{with } F(b)_{acc,k} = \sum_{a=1}^{k} F(a,b), \tag{7}$$

$$F(a,b) = \int_{-\infty}^{\infty} f(t)\overline{\psi_{a,b}(t)}dt, \tag{8}$$

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right), \text{ with } a \neq 0 \text{ and } a,b \in \mathbb{R} \tag{9}$$

where $\psi(t)$ is the mother wavelet function.

A second set of features, also derived from the continuous wavelet transform, is computed as:

$$y(k) = \frac{mean(F(b)^2_{acc,k,\max})}{mean(F(b)^2_{acc,k,\min})} \tag{10}$$

where $F(b)_{acc,k,\max}$ and $F(b)_{acc,k,\min}$ represent the maxima and minima extracted from $F(b)_{acc,k}$. A third set of features is computed based on the power spectral density (PSD) of the signals f(t) that need to be classified. The PSD parameters are slopes of line fits on different decades of the log PSD of f(t) vs. log frequency plot. A motivation for the derivation of these features can be found in [4]. A plot of the x(k) values for different k values is shown in figure 2.

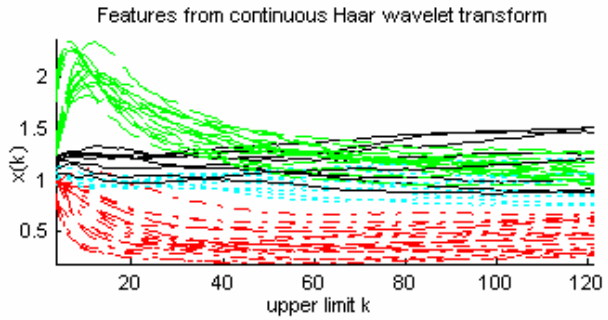

Fig. 2. Plot of wavelet features for different accumulation levels k. Each plot represents the x(k) parameters derived for a different time series. Parameters derived from time series from 4 different classes are shown. Different classes are encoded in different gray values and in different line styles: '- -', '-', '- ·' and '···'. For reasons of visibility the complete data set is not shown. Different regions of k values can be distinguished where a class becomes separated from the other classes.

A similar plot is obtained for the y(k) parameters.

In total 263 features are derived: 7 PSD features, 128 x(k) features and 128 y(k) features. In total 579 time series spread over 4 classes are considered. An agglomerative hierarchical clustering technique is used in the formation of the clusters. The farthest-neighbor algorithm is used in merging the clusters [7]. In this algorithm the distance between clusters is computed as:

$$d_{\max}(D_i, D_j) = \max_{\substack{\mathbf{x}\in D_i \\ \mathbf{x}\in D_j}} \|\mathbf{x} - \mathbf{x}'\|. \tag{11}$$

When the data is gathered in 3 main clusters then 1 cluster is formed by the PSD features, 1 by the x(k) features and 1 by the y(k) features. So 3 would rather correspond with the number of 'naturally' underlying clusters. In GA1 each representative feature of a cluster represents a set of features in the same cluster. Therefore correlations between features within the same cluster may not be too small. We propose that the number of clusters is defined by a user-defined minimum correlation constant. That is the number of clusters is equal to the minimum number of clusters where correlations between any features in the same cluster are not smaller than the minimum correlation constant.

For this application, the constant is set equal to 0.8. This leads to 16 clusters: 6 clusters are formed from the x(k) features, 6 from the y(k) features and 4 from the PSD features. In this case, the smallest correlation between any of the features in the same cluster is equal to 0.82. The performance of the proposed GA is compared with a standard GA (SGA) with

variable length of genotypes without exploitation of feature correlations. Results for 4 subsequent simulations are summarized in table I.

TABLE I
SIMULATIONS OF GA ON REAL-CASE DATA SET

	Simulation 1	Simulation 2	Simulation 3	Simulation 4
SGA generartion	26 of 50	25 of 25	15 of 25	24 of 25
SGA performance	95.82%	95.39%	95.23%	95.42%
GA1 generation	2 of 5	5 of 5	4 of 5	5 of 5
GA1 performance	95.60%	96.40%	95.73%	96.20%
GA2 generations	3 of 10	1 of 10	6 of 10	9 of 10
GA2 performance	96.27%	96.40%	96.20%	96.41%

The SGA is basically GA1 where the number of clusters equals the number of features.
The SGA and the 2-stage GA were initialized from the same sets of genotypes. In each simulation a different setting of P_{c1} and P_{m1} was chosen. These parameters were set equal for GA1 and SGA. The parameters P_{m2} and P_{C2} were always set to 0.3.

The table shows how many generations are considered in each simulation and in which generation the best solution is found.

From the table we note that the 2-stage GA finds better solutions (about 1%), given a predefined number of generations. Note that all solutions, independently from the initial set of genotypes, found in the 2-stage GA perform better than the best solution found in SGA (95.82%). Note also that fewer generations are needed to find the best genotypes. The best genotype found (96.41%) contains 5 features: 1 PSD feature, 2 features from x(k) and 2 features from y(k).

The results in table I are obtained by averaging the performances of 10 runs of 10-fold crossvalidation. Due to the variance of these performances, one needs to consider whether the difference in performance between the SGA and the 2-stage GA are statistically significant. Hypothesis testing, with level of significance α equal to 0.05, has shown that the differences are statistically significant.

VI. CONCLUSIONS

A new 2-stage GA based feature subset selection algorithm was proposed in which the correlation structure of the features is exploited. Simulations on a real-case data set with correlated features show that the 2-stage GA finds better solutions in fewer generations compared to a standard GA in which the correlation structure is not exploited.

REFERENCES

[1] S. Raudys, and V. Pikelis, "On dimensionality, sample size, classification error, and complexity of classification algorithm in pattern recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. PAMI-2, no. 3, pp. 242-252, May 1980.
[2] S. Raudys, and K. Jain, "Small sample size effects in statistical pattern recognition: recommendations for practitioners," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, no. 3, pp. 252-264, March 1991.
[3] I. Daubechies, *Ten Lectures On Wavelets*, CBMS Regional Conference Series in Applied Mathematics # 61, SIAM, 1992.
[4] G. Van Dijck, M. Wevers, and M. Van Hulle, "Corrosion time series classification using the continuous wavelet transform and MML density estimation," submitted for publication, International Conference on Computational Intelligence, ICCI 2004.
[5] G. John, R. Kohavi, and K. Pfleger, "Irrelevant features and the subset selection problem," in *Machine Learning: Proc. of the Eleventh Int. Conf.*, Morgan Kauffman, 1994, pp. 121-129.
[6] R. Kohavi, and G. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, spec. issue on relevance, pp. 273-324, Dec. 1997.
[7] L. Citi, R. Poli, and F. Sepulveda, "An evolutionary approach to feature selection and classification in P300-based BCI," in *Proc. of the 2nd Int. Brain-Computer Interface Workshop and Training Course*, Graz, 2004
[8] E. Kalapanidas, and N. Avouris, "Feature selection using a genetic algorithm applied on an air quality forecasting problem," *AI communications*, vol. 16, no. 4, pp. 235-251, 2003.
[9] J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm," *IEEE Intelligent Systems*, vol. 13, nr. 2, pp. 44-49, 1998.
[10] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer-Verlag, 3rd edition, 1999.
[11] M. A.T. Figueirido, and A.K. Jain, "Unsupervised learning of finite mixture models," *IEEE Trans. On Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp.381-396, March 2002.
[12] R. O. Duda, P.E. Hart, and D.G. Stork, *Pattern Classification*, 2nd ed., Wiley-Interscience, 2000, pp. 550-559.
[13] G. W. Milligan, and M.C. Cooper, "An examination of procedures for detecting the number of clusters in a data set," *Psychometrika* 50(2), pp. 159-179, 1985.