A Data Mining Approach to Discover Genetic and Environmental Factors involved in Multifactoral Diseases

L. Jourdan, C. Dhaenens, E.G. Talbi, S. Gallina

KNOSYS1341

ELSEVIER

Knowledge-Based Systems 00 (2001) 000–000

Knowledge-Based SYSTEMS

www.elsevier.com/locate/knosys

A data mining approach to discover genetic and environmental factors involved in multifactorial diseases

L. Jourdan[a,*], C. Dhaenens[a], E.-G. Talbi[a], S. Gallina[b]

[a]*LIFL, Batiment M3, Cité Scientifique, 59655 Villeneuve d'Ascq Cedex, France*
[b]*Biological Institute, Multifactorial Disease Laboratory, 1 rue du Professeur Calmette, B.P. 245, 59019 Lille Cedex, France*

Received 16 March 2001; revised 22 April 2001; accepted 31 May 2001

Abstract

In this paper, we are interested in discovering genetic and environmental factors that are involved in multifactorial diseases. Experiments have been achieved by the Biological Institute of Lille and many data has been generated. To exploit these data, data mining tools are required and we propose a two-phase optimisation approach using a specific genetic algorithm. During the first step, we select significant features with a specific genetic algorithm. Then, during the second step, we cluster affected individuals according to the features selected by the first phase. The paper describes the specificities of the genetic problem that we are studying, and presents in detail the genetic algorithm that we have developed to deal with this very large size feature selection problem. Results on both artificial and real data are presented.

Keywords: Data mining; Clustering; Genetic algorithm; Feature selection; Multifactorial disease

1. Introduction

Common diseases such as Type 2 diabetes, obesity, asthma, hypertension and certain cancers represent a major public health concern (around 160 million people have Type 2 diabetes). These complex diseases have a multifactorial aetiology in which environmental factors (for example: Body Mass Index, which is a measure of obesity, or the age at onset, which is the age of the individual when diabetes was diagnosed), as well as genetic factors (genetic markers) contribute to the pathogenesis of the disease. In order to localise the involved factors, we must be able to detect complex interactions such as [(gene A and gene B) or (gene C and environmental factor D)] in one or more populations. Classical methods of genetic analysis test models with one genetic factor. Some methods have been adapted to test the interaction of a second factor, once a first major factor has been detected. These methods are not intended to test a huge amount of genetics models combining more than two genes. So, they have only a limited capacity to dissect the genetics of complex disease traits.

In order to elucidate the molecular bases of Type 2 diabetes and obesity, the Multifactorial Disease Laboratory at the Biological Institute of Lille had performed large analyses on collections of affected families from different populations. Since the precise molecular mechanisms leading to these diseases are largely unknown, a genome-wide scan strategy was used to localise the genetic factors. This strategy requires no presumptions about interesting loci (a locus is a specific portion of DNA, located without any ambiguity on one chromosome). Seeking for interaction patterns in the huge amount of data generated during this first step should increase the power to detect regions containing genes with no individual major effect, but whose interaction leads to the disease. The biological analysis already performed consists of four steps.

1. Collect families with at least two or three affected members.
2. Extract DNA of parents and offspring from a blood sample.
3. Characterise each DNA sample at 400 loci spread over the 23 chromosomes.
4. Compute the genetic similarities for each pair of relatives at each locus.

Loci are polymorphous, so that parents may have different variants, called alleles. Each individual has two alleles for each locus, one inherited from his father and the other from

* Corresponding author. Tel.: +33-320337186.
E-mail addresses: jourdan@lifl.fr (L. Jourdan), talbi@lifl.fr (E.-G. Talbi).

0950-7051/01/$ - see front matter
PII: S0950-7051(01)00145-9

his mother. So we can calculate a genetic similarity for a pair of relatives (for example two brothers), based on the number of common alleles they have. For example, at a given locus, two brothers will have the probability: 25% to share no alleles, 50% to share one allele, 25% to share two alleles.

The genetic similarity for each pair is calculated with multi-point methods, in order to have extrapolated values at regular positions between loci. That leads to 3652 values corresponding to 3652 points of comparison on the 23 chromosomes. Then, a comparison is made between the genetic similarity observed and what would be expected regarding the statistical sharing probabilities. This leads to a binary matrix, where a 1 indicates that the observed similarity is greater than the expected probability.

The method must detect a combination of factors, for which a subset of lines share a common pattern of values. However, the method should not converge to a unique solution, but must produce several solutions in order to take into account the heterogeneity of the populations. Moreover, only very few factors (less than 5%) should be relevant.

This is an unsupervised clustering problem, where 3654 features[1] (3652 comparison points and two environmental factors) have to be considered, but where biologists have in mind that only very few features are relevant.

We would like here to point some specifications of the problem regarding usual data mining problems.

- A lot of features have to be considered (up to 3654).
- Very few significant features (less than 5%).
- Only few data (number of pairs of individuals), compared to the large number of features, are available.
- We can only work with affected people because those who are not affected, when extracting their DNA, may become affected.
- The objective is to discover not only a single association, but several associations of genes and environmental factors, and to group affected people according to these associations.

To deal with an unsupervised clustering problem on such a large number of features, it is not possible to apply directly classical clustering algorithms. Algorithms, such as *k*-means algorithm, dedicated for clustering, are not able to deal with so many features. Their executions would be time consuming and results obtained would not be exploitable. Therefore, we adapt a two phase approach.

- A feature selection phase using a genetic algorithm.
- A clustering phase.

For the first phase, a feature selection, we developed a genetic algorithm to extract most influential features and in particular influential associations of features. This heuristic approach has been chosen as the number of features is large. For this specific problem, some advanced mechanisms have been introduced in the genetic algorithm such as some dedicated genetic operators, the sharing, the random immigrant, and a particular distance operator has been defined. Then, the second phase is a clustering based on the features selected during the previous phase. The clustering algorithm used is *k*-means.

This paper is organised as follows. First, we give some mathematical background. Then, our approach, an adaptation of a genetic algorithm for this particular feature selection problem, is detailed. Section 4 presents the clustering phase. In Sections 5 and 6, we report results on experiments realised with data from GAW11, a workshop on genetic analysis and with real datasets.

2. Mathematical background

In this section, we give definitions of concepts and functions used in the rest of the paper.

Unsupervised classification (or clustering or learning from observations). This is a data mining task in which the system has to classify a set of objects without any information on the characteristics of classes. It has to find its own classes (clusters).

Feature (or attribute). It is a quantity describing an instance. Here, a feature may be an environmental factor or a genetic comparison point.

Support. The notion of support in data mining denotes the number of times (the number of rows) a set of features is met over the number of times at least one member of the set is met.

Here is a formal definition of the support: let R be a set, r be a binary database over R and $X \subseteq R$ be a set of items. The item set X matches a row $t \in R$ if $X \subseteq t$. The set of rows in r matches by X is denoted by $|\{t \in R/X \subseteq t\}|$ and the $support = |\{t \in r/X \subseteq t\}|/|\{t \in r/(\exists x_i \in X/x_i \subseteq t)\}|$.

Scattering criterion. To measure the quality of clustering we use the classical scattering criterion based on statistical concepts. It is defined as follows:

Given c clusters, n features,
Let χ_j be the j-th cluster with $j = 1, \ldots, c$,
Let m be the mean vector ($m = (m_1 \quad m_2 \quad \cdots \quad m_n)$ and m_j the vector of means for the j-th cluster $m_j = (m_{j_1} \quad m_{j_2} \quad \cdots \quad m_{j_n})$.
X_i is an element of the cluster χ_j and $(X_i - m_j)^t$ is the transposition of the vector $(X_i - m_j)$.
The scatter matrix for the j-th cluster is then: $P_j = \sum_{X_i \in \chi_j} (X_i - m_j)(X_i - m_j)^t$. We can define the intra

[1] The term feature will now be used according to the data mining terminology. It is defined in Section 2.

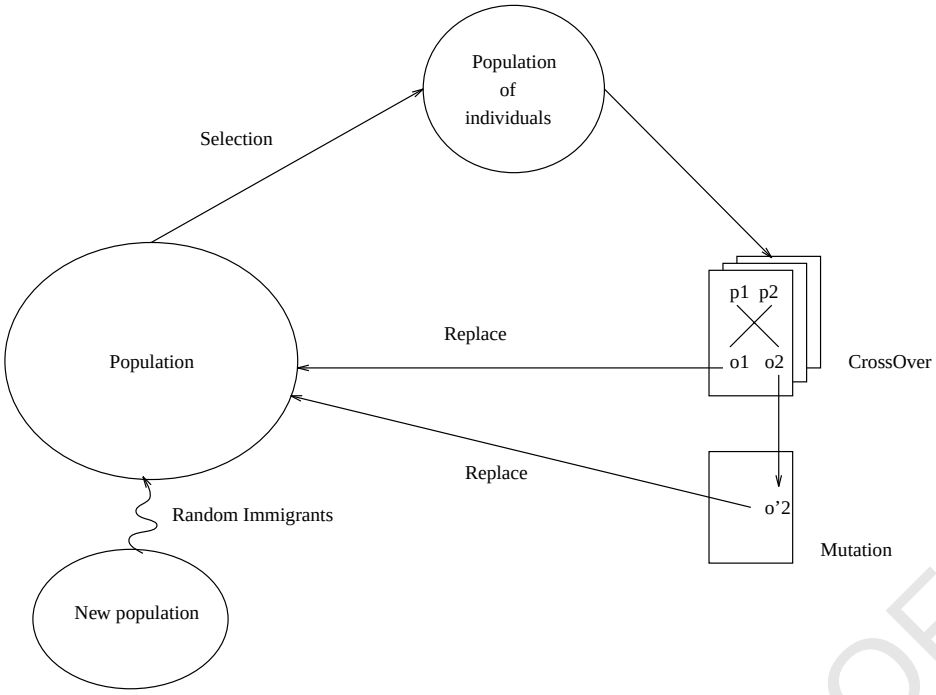

Fig. 1. Our genetic algorithm: the different phases.

cluster scatter matrix for c clusters:

$$P_W = \sum_{j=1}^{c} P_j$$

and the inter cluster scatter matrix between clusters:

$$P_B = \sum_{j=1}^{c} \left(m_j - m\right)\left(m_j - m\right)^t$$

Then, the scattering criterion is: $tr\left(P_W^{-1} P_B\right)$ where tr is the trace. The better $tr\left(P_W^{-1} P_B\right)$ is, the better the ratio between the intra cluster scatter and the inter cluster scatter is and the better is the quality of the clustering.

3. The feature selection phase

The first phase of our algorithm deals with isolating the very few relevant features from a large set. This is not exactly the classical feature selection problem known in data mining as, in Ref. [1], for example around 50% of features are selected. Here, we have the idea that less than 5% of features have to be selected. However, this problem is very close to the classical feature selection problem, and we take inspiration from the literature about this subject.

3.1. Description of the data to be analysed for interaction patterns

The genetic similarities are merged in a matrix together with environmental factors. In this matrix, lines represent pairs of relatives and columns, the genetic and environmental factors. Each factor will be called a feature. This matrix has no missing value for genetic data since we use extrapolation. The size of the matrix will be from 500 to 1000 lines, depending on the number of families, and from 400 to 3654 columns depending on the extrapolation function. The size of this matrix is unusual for data mining because the number of columns (i.e. the features) is of the same order than the number of lines.

3.2. Feature subset selection

The feature subset selection problem refers to the task of identifying and selecting a useful subset of features from a large set of redundant, perhaps irrelevant, features [1].

Two models exist depending on whether the selection is coupled with the classification scheme or not. The first one, the filter model, which realises the feature subset selection and the classification in two separate phases, uses a measure that is simple and fast to compute. The second one, the wrapper method, which realises the feature subset selection and the classification in the same process, engages a learning algorithm to measure the classification accuracy.

Classically, feature selection is treated as an optimisation problem: finding the minimal number of bits that have value 1 with which a given criterion is satisfied.

Feature selection and extraction can optimise classification performance and even affect classifier design [2]. Researchers show that optimal feature selection is NP-hard [3]. Therefore, only heuristics, and in particular

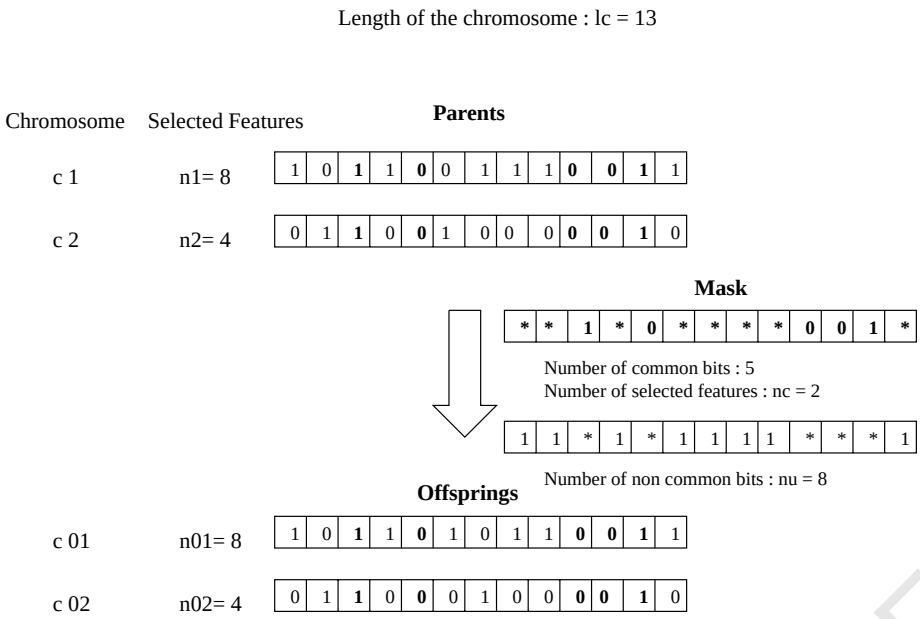

Fig. 2. The SSOCF crossover operator.

meta-heuristics such as genetic algorithms [1,2], are able to deal with large size problems.

For this phase, we decide to use a genetic algorithm as the literature shows they are well adapted for problems with a large number of features [2,4,5].

A genetic algorithm (GA) works by repeatedly modifying a population of artificial structures through the application of genetic operators. The goal is to find the best possible solution or good solutions for the problem. For this particular feature selection problem, we have developed a particular GA (see Fig. 1). We present here the main characteristics and adaptations that we made to deal with the particular problem.

3.3. The encoding

The genetic algorithm uses the concept of chromosome to encode and manipulate solutions. Each chromosome will define an individual of the population of the genetic algorithm. Here, we use a binary representation: a chromosome is a string of bits whose size corresponds to the number of features. A 0 or 1, at position i, indicates whether the feature i is selected (1) or not (0).

3.4. The chromosomal distance (a distance adapted to the problem)

We noted that features are correlated. A gene and its neighbours on the same chromosome have similar influence. We decided to create a distance to be able to compare two individuals and to recognise two similar individuals of our GA. This distance takes into account this correlation.

Biologist experts indicate that the length of correlation is 20 c-Morgan (a measure unit): i.e. a gene is correlated with the genes that are around it. This correlation stops at the chromosomal cut.

3.5. The fitness function

As we deal with an unsupervised classification (clustering), it is impossible to evaluate the accuracy of the feature selection phase according to the clustering and we need to build a particular objective function and to adapt a filter model.

The fitness function we developed refers to the support notion (see Section 2) of an association. The function is composed of two parts. The first one favours for a small support a small number of selected features and the second one, the most important (multiplied by 2), favours for a large support a large number of features. What is expected is to favour good associations (in terms of support) with as many features as possible.

$$F = \left((1 - S) \times \frac{\frac{T}{10} - 10 \times SF}{T} \right) + 2 \times \left(S \times \frac{\frac{T}{10} - 10 \times SF}{T} \right)$$

where: *Support*; $S = |A \cap B \cap C \cap \cdots| / |A \cup B \cup C \cup \cdots|$; A, B, C, … are the selected features; $|A \cap B \cap C \cap \cdots|$ is the number of rows for which features A, B, c, … are all equal to 1; and $|A \cup B \cup C \cup \cdots|$ is the number of rows for which at least one feature A, B, … is equal to 1. T = total number of features and SF = number of selected significant

features (selected features that are not too close in terms of the chromosomal distance.

3.6. The genetic operators

These operators allow GAs to explore the search space. However, operators typically have destructive as well as constructive effects. They must be adapted to the problem.

Crossover

We use the *subset size-oriented common feature crossover operator* (SSOCF) [6] that keeps useful informative blocks and produces offspring which have the same distribution than the parents (see Fig. 2).

The shared features are kept by offspring and the non-shared features are inherited by offspring corresponding to the i-th parent with the probability $(n_i - n_c/n_u)$ where n_i is the number of selected features of the i-th parent, n_c is the number of commonly selected features across both mating partners and n_u is the number of non-shared selected features.

Mutation

The mutation is an operator which allows diversity. During the mutation stage, a chromosome has a probability p_{mut} to mutate. If a chromosome is selected to mutate, a number n of bits to be flipped is chosen then n bits are randomly chosen and flipped. In order to create a large diversity, we set p_{mut} around 10%.

Offspring are kept, only if they fit better than the least good individual of the population.

3.7. Selection

We implement a probabilistic binary tournament selection. Tournament selection holds n tournaments to choose n individuals. Each tournament consists of sampling two individuals of the population and choosing the fittest one with a probability $p \in [0.5, 1]$.

3.8. Sharing

To avoid premature convergence and to discover different good solutions (different relevant associations of features), we use a niching mechanism. A comparison of such mechanisms has been done in Ref. [7]. The objective is to boost the selection chance of individuals that lie in less crowded areas of the search space. We use a niche count that measures how crowded the neighbourhood of a solution is. The distance D is the chromosomal distance adapted to our problem presented before. The fitness of individuals in high concentrated search space regions is degraded and a new fitness value is used, in place of the initial value of the fitness, for the selection. The sharing fitness $f_{sh}(i)$ of an individual i, where n is the size of the population, $\alpha_{sh} = 1$ and $\sigma_{sh} = 3$, is:

$$f_{sh}(i) = \frac{F(i)}{\sum_{j=1}^{n} Sh\left(D_{i,j}\right)}$$

where

$$A_{i,j} = \left(\frac{D_{i,j}}{\sigma_{sh}}\right)^{\alpha_{sh}}$$

$$Sh\left(D_{i,j}\right) = \begin{cases} 1 - A_{i,j} & \text{if } D_{i,j} < \sigma_{sh} \\ 0 & \text{else} \end{cases}$$

3.9. Random immigrant

Random immigrant is another method that helps to maintain diversity in the population. It should also help to avoid premature convergence [8]. We use random immigrant as follows. When the best individual is the same during N generations, all the individuals of the population, whose fitness, $f_{sh}(i)$, is under the mean, are replaced by new individuals randomly generated.

4. The clustering phase

The second phase of our method is a clustering phase using features selected during the first phase.

4.1. Clustering

Clustering is used to identify classes of objects sharing common characteristics. Clustering methods can be applied to many human activities and particularly to the problem of taking automatic decisions, for example in the establishment of a medical diagnosis from the clinical description of a patient. The classification, or data clustering, can isolate similarities and differences in the database and make groups of similar data which are called classes or groups. The clustering methods make classes of homogeneous objects [9]. Widely used clustering methods are: K nearest neighbour, k-means [10], density clustering [11]. However, most of the time, only a few features are considered by these methods. In our case, we need to consider 3654 features and clustering methods have difficulties dealing with such a large number, that is why we needed to pre-select some of the most relevant features.

Much research has been done on clustering problems, but as our main work is around the feature selection process, we do not describe all of the clustering works. The interested reader may refer to the tutorial [12].

4.2. Objective of the clustering phase

The objective of the clustering phase is to group together pairs of individuals that share the same genetic particularities. A feature selected during the first phase indicates that

Table 1

*E*1	*A*	*B*	*C*	*D*	*F*	*G*	*H*	*I*	*J*	*K*
1	0	0	1	0	0	0	0	0	0	0
1	0	0	1	0	1	0	0	0	0	0
0	0	0	1	0	1	0	1	0	0	0
1	0	0	1	0	1	0	0	0	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	1	0	0	0	1	1	0
0	1	0	0	0	0	0	0	1	1	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	0	0	0	0	1	0	1	0

alone or with other features, it is often shared by affected people.

Now, the clustering should indicate what are the most significant associations.

Two pairs that would be grouped in a same cluster indicate that the concerned people share similar genetic particularity. Hence, these particularities may explain their affect and a rule such as 'The disease may be explained thanks to features *A*, *B* and *C*' may be expressed. Each class will represent a rule, so the final result could ideally be 'This disease may be explained thanks to association of features *A*, *B*, *C* or *D*, *E*, *F*, *G* or *H*, *I*'.

4.3. Application of k-means

The feature selection has selected significant features. We can now use a classical algorithm: the *k*-means algorithm.

The *k*-means algorithm is an iterative procedure which requires an initial classification of the data. The *k*-means algorithm proceeds as follows: it computes the centre of each cluster, then computes a new partition by assigning each object to the cluster whose centre is the closest to that object. This cycle is repeated during a given number of iterations or until the assignment has not changed during one iteration [13].

Since the number of features is now very small, we implement a classical version of the *k*-means algorithm by randomly selecting initial centres.

5. An illustrative example

In this part, we give a short example of the two phases of our method.

If we consider the matrix shown in Table 1, where *E*1 is an environmental factor and *A*, *B*, *C*, …, *K* are genetic factors, the best solutions are $f_{E1C} = 271/120, f_{CF} = 271/120, f_{IJ} = 112/75, f_{E1CF} = 7/5$.

The genetic algorithm will select the features *E*1, *C*, *F*, *I*, *J*.

Table 2
Associations discovered by the GA with the artificial data

Association	*A* + *B*	*A* + *D*	*B* + *D*	*C* + *E*1
Frequency (%)	100	50	20	10

Table 3
Clusters obtained and their occurrences with the artificial data

Cluster	Occurrence
(*A B D*), (*E*1 *C*)	4
(*A B D*), (*E*1 *C D*)	1
(*A B*), (*E*1 *D C*)	1
(*A B*), (*E*1)	2
(*E*1 *A D B*), (*E*1 *C B*)	1
(*A B D*), (*E*1 *D*)	1

Then, the *k*-means algorithm will make the following clusters:

- cluster1: *E*1 *C F*; and
- cluster2: *I J*

6. Experimental results

6.1. Experiments on artificial data

Experiments have been first executed on an artificial database, constructed for the workshop GAW11 (Genetic Analysis Workshop) which was held in 1998 and was organised by Dr David A. Greenberg.[2] This base is public and is used to evaluate different data mining methods. This is an artificial database, but constructed in order to be very similar to real databases, and we know, by construction, the relevant associations of features (associations of loci (*A*, *B*, *C*, *D*) and environmental factors (*E*1, *E*2)) which can influence the disease. Results to obtain are associations $A + B + D$ and $E1 + C$.

This test base is composed of 491 features and 165 pairs of individuals.

For 10 runs, we wanted to know how many times associations were discovered by the genetic algorithm. Table 2 presents the results.

The first phase was able to discover real interactions of loci. Some of them are more difficult to find than others.

Then, we ran the *k*-means algorithm with the results of the GA, in order to validate results.

We gave 11 features selected among the 491 thanks to the first phase, (only 2.24% of the initial number of features), to the *k*-means algorithm. The *k*-means algorithm helps us to discover associations between genes and association between genes and environmental factors. We had previously experimented with the classical *k*-means algorithm without any feature selection. The execution time was very large (over 7500 min) and results could not be interpreted (we did not know which were the features involved in the disease). With the feature selection, the executive time of *k*-means has decreased to 1 min and the

[2] Dr D.A. Greenberg, Department of Psychiatry, Box 1229, Mt Sinai Medical Center, 1 G. Levy Place, New York, NY 10029, USA.

Table 4
Associations discovered by the GA with the real data

Association	$B + C$	$D + G$	$A + F$	$A + D$	$E1 + H$
Frequency (%)	50	100	50	40	10

results are exploitable. We ran k-means (with $k = 2$) 10 times with the 11 selected features. We present in Table 3 clusters obtained and their number of occurrences. This table shows that the k-means algorithm using results of the GA is able to construct clusters very closely related to the solution presented in results of the workshop. Moreover, this solution has been exactly found in 40% of the executions.

6.2. Experiments on real data

Experiments have been executed on real data provided by the Biological Institute of Lille (B.I.L.) for the study of diabetes. First, the dataset was composed of 543 pairs of individuals who have diabetes. Biologists examined 385 points of comparison and two environmental factors (age at onset, the age of the individual when diabetes was diagnosed and BMI Body Mass Index, which is a measure of obesity).

Now, experiments are currently done on a more complete database composed of 1075 pairs of individuals, 3652 comparison points and two environmental factors. As data and results are confidential, we are not allowed to detail all the biological results, but we present here some aspects.

Performance of the method

The proposed approach must be able to deal with a very large dataset so we first evaluate its performance. Therefore, we made some tests to determine limits of our application. The genetic algorithm is the longest phase of our approach, so we tested its ability to deal with large databases. Firstly, we tested it for different numbers of features. We obtained the following results: 1000 features, 60 min; 2000 features, 140 min; 3000 features, 207 min; 3654 features, 255 min. So, the genetic algorithm grows linearly in time in function of the number of features.

Secondly, we tested it for different numbers of pairs of individuals. We obtained the following results: 200 pairs, 56 min; 400 pairs, 64 min; 600 pairs, 74 min; 800 pairs, 76 min.

Hence, these experiments show that this algorithm may be applied to very large datasets (with a large number of features) and is able to give results in reasonable time.

Table 5
Clusters obtained and their occurrences with the real data

Cluster	Occurrence (%)	Cluster	Occurrence (%)
(*A F H*)		(*A H*)	
(*B C D*)	75	(*B C D*) (%)	25
(*E*1 *G*)		(*E*1 *F G*)	

Results

For the first real dataset (453 pairs and 387 features) experiments have been achieved. We present here results for 10 executions of the genetic algorithm. During the 10 executions, a total of seven features corresponding to seven different locations of chromosomes (that we cannot reveal) (A, B, C, D, F, G, H) and an environmental factor ($E1$) have been selected in different associations. Table 4 indicates the number of times each association has been found.

This table shows that some associations are easy to find (e.g. association of features D and G is found in 100% of the executions) and others are more difficult (association $E1 + H$ is found in 10% of the executions).

Then, in order to validate these results, and to be able to propose interesting and complete associations, we ran the k-means algorithm with the eight selected features.

As we do not know how many clusters (i.e. how many associations will be found), we ran k-means with several values of k. The best results were found with $k = 3$ according to the scattering criteria. Table 5 reports the number of times each association was found with $k = 3$ when the scattering criterion was good.

Now, experiments are done with the complete database, as the performance study showed that the method was able to deal with such a large database. Those results are confidential and we cannot give any information about them. What we can say is that the results of two executions are similar, and it allows us to express assumptions that biologists have now to confirm.

The important aspect here is the ability of the method to find several associations. The method is a tool that must help biologists to identify interesting areas of chromosomes that they will be able to study in detail.

7. Conclusions and discussion

This paper presents an optimisation method developed to deal with a data mining problem. The objective is to provide a tool to help biologists to find interactions between genes involved in multifactorial diseases. However, these interactions are difficult to find and statistical methods are mostly based on single-gene models. We model this problem as a particular feature selection problem, where we want to find different combinations of genes. We solve this problem using a two phase method, where the first phase selects the most relevant associations of features and the second phase validates the feature selection by grouping affected people according to the selected features, using the classical k-means algorithm. For the first phase, we propose an adapted genetic algorithm. We have tested our approach on constructed data provided for GAW11 and compared our results with other works. Now, this method is used by

biologists of B.I.L. in order to exploit data they have collected to study diabetes and obesity. This approach gave promising results (it is able to identify interesting associations of genes and/or environmental factors) that biologists have to confirm with biological experiments. Complex diseases are a good challenge for geneticists. These diseases are characterised by etiologic heterogeneity and with data mining approaches, such as the approach proposed here, biologists are able to discover interactions between genes and between genes and environmental factors.

The perspectives for this project may be the following.

- Taking into account the family aspect in our data for the evaluation function.
- Working on different clustering methods (*k*-nearest neighbours, dedicated methods, etc.) because the *k*-means algorithm is not able to give good solutions for each run.

References

[1] J. Yang, V. Honoavar, Feature subset selection using a genetic algorithm, in: H. Liu, H. Motoda (Eds.), Feature Extraction, Construction and Selection: a Data Mining Perspective, Kluwer Academic, MA, 1998, pp. 117–136.

[2] M. Pei, E.D. Goodman, W.F. Punch, Feature extraction using genetic algorithms. Technical Report, Michigan State University, GARAGe, June 1997.

[3] P.M. Narendra, K. Fukunaga, A. branch, and bound algorithm for feature subset selection, IEEE Trans. Computer C-26 (9) (1977) 917–922.

[4] M. Pei, E.D. Goodman, W.F. Punch, Y. Ding, Genetic algorithms for classification and feature extraction, in: Annual Meeting: Classification Society of North America, June 1995.

[5] M. Pei, E.D. Goodman, W.F. Punch, Pattern discovery from data using genetic algorithm, in: Proceedings of the First Pacific-Asia Conference on Knowledge Discovery and Data Mining, February 1997.

[6] C. Emmanouilidis, A. Hunter, J. MacIntyre, A multiobjective evolutionary setting for feature selection and a commonality-based crossover operator, in: Congress on Evolutionary Computing 2000, vol. 2, CEC, 2000, pp. 309–316.

[7] S.W. Mafhoud, Niching method for genetic algorithms, PhD thesis, University of Illinois, 1995.

[8] C. Bates Congdon, A comparison of genetic algorithm and other machine learning systems on a complex classification task from common disease research, PhD thesis, University of Michigan, 1995.

[9] T. Mitchell, Machine Learning, McGraw-Hill, 1997.

[10] K. Fukunaga, Introduction to Statistical Pattern Recognition, Academic Press, San Diego, CA, 1990.

[11] R. Lefebvre, G. Venturi, Le Data Mining, Eyrolles Informatique, 1998.

[12] A. Hinneburg, D.A. Keim, Tutorial: clustering techniques for large datasets from the past to the future, in: PKDD 2000, September, 2000, p. 65, url: http://hawaii.informatik.uni-halle.de/hinnebur/ClusterTutorial/

[13] N. Monmarché, M. Slimane, G. Venturini, Antclass: discovery of cluster in numeric data by a hybridization of an ant colony with the kmeans algorithm. Technical Report 213, Ecole d'Ingénieurs en Informatique pour l'Industrie (E3i), Université de Tours, January 1999.