Using Machine-Assisted Learning to Help Medical Experts Improve Their Knowledge

Vili Podgorelec
Institute of Informatics
University of Maribor, FERI
Smetanova 17, SI-2000 Maribor
Slovenia
vili.podgorelec@uni-mb.si

Marjan Heričko
Institute of Informatics
University of Maribor, FERI
Smetanova 17, SI-2000 Maribor
Slovenia
marjan.hericko@uni-mb.si

Ivan Rozman
Institute of Informatics
University of Maribor, FERI
Smetanova 17, SI-2000 Maribor
Slovenia
i.rozman@uni-mb.si

Abstract – **A large amount of various machine learning algorithms have already proved their usefulness in everyday work in a variety of disciplines including medicine. Machine learning does not only help in a routine task such as classification, it can also help in learning new patterns, which are hoped for to reveal a whole new perspective onto areas not even imaginable so far. In helping experts to learn new knowledge, algorithms should assist them in a learning process in order to enhance their "information processing strength". On the other hand, the experts can "assist" the algorithm to learn what is important. This process we have called machine-assisted learning.**

In the paper we present an approach to enhance the machine learning algorithm with the help of experts' assessment of the progressing solution. The evolutionary method for automatic extraction of rules is presented that is based on the evolutionary induction of decision trees and automatic programming. The method is evaluated in a case study using a medical database. The obtained rules are assessed by physicians to evaluate the strength of the developed knowledge discovery method.

I. INTRODUCTION

The general idea of discovering knowledge in large amounts of data is both appealing and intuitive, but technically it is significantly challenging and difficult, especially in the field of medicine where really huge amounts of row data have been collected over last decades or even centuries. The medicine, generally speaking, is a conservative science, where every new knowledge has to be fully acknowledged and well understood. In order to fulfill this additional constraint of the knowledge discovery process one would do better when helping experts to learn instead of trying to "produce" the knowledge from data fully automatically.

In the present paper we attack the above challenge and introduce a new approach to rules induction from medical datasets with the help of physicians' assessment of the progressing solution in order to find new patterns in the available data. In this way the knowledge discovery algorithm is only a tool that enhances the experts' learning capabilities rather than being a "knowledge generator". A case study performed should help in assessing this approach that we call the machine-assisted learning.

II. KNOWLEDGE DISCOVERY METHOD

Almost since their introduction decision trees (DTs) have been exhaustively used as a classification method, showing a great potential in several domains. Their efficiency and classification accuracy have surprised many experts, but their most important advantage is the transparency of the classification process that one can easily interpret, understand and criticize. However, the classical induction approach of DTs, that has not changed much since the introduction, contains several disadvantages, like 1) poor processing of incomplete, noisy data, 2) inability to build several trees for the same dataset, 3) inability to use the preferred attributes, etc.

For all those reasons a need for an approach that would preserve the positive aspects of DTs and avoid their disadvantages emerged. Encouraged by the success of evolutionary algorithms for optimization tasks a series of attempts occurred to induce a DT-like models with evolutionary methods [1, 9, 10, 11]. Despite their high classification accuracy, GPs have proven extremely difficult to interpret – this is a major obstacle to their use in data mining problems. Nevertheless, many researchers have tried to use the power of GAs/GPs for the problem of data mining and knowledge discovery [12, 13].

As we learn from the history, the ideal choice for effective, accurate and efficient data mining algorithm would be to combine the power of GA/GP with the interpretability of DTs (or rule sets) in a successful way. In this paper we present one such approach that shows a great potential both in data mining (classification) and knowledge discovery.

As a knowledge discovery system we developed a rule extraction method called AREX, that is able to induce a set of classification rules based on the given dataset. For the purpose of searching for new knowledge in medical datasets, the developed algorithms had to be modified in some aspects.

In the first run the AREX algorithm generates a set of rules (see bellow), which are then assessed by physicians according to medical relevance and originality. The rules, which are at the same time relevant and represent some new knowledge, are stored in the database of good rules. In the next runs AREX give the bonus preference to rules that use the same attributes and parts of the rules stored in the database. After several runs the quality of the rules produced by AREX is significantly improved regarding the preference criteria given by physicians and the process eventually leads to some new knowledge. The outline of our approach is presented in figure 1.

A. Automatic rules extracting algorithm AREX

Knowledge discovered with data mining algorithms should ideally give an in-depth explanation of a problem domain along with a good classification [3,6,7]. Experts are performing exhaustive analyses of the results, which are the output of knowledge discovery tools, in order to extract the useful knowledge. To make a part of their work as easy as possible the best way is to present the results in a form of set of classification rules, which are clear and straightforward to understand, accept or reject. Ideal system would therefore include:

- accuracy – classification with minimal error rate,
- compactness – use of a minimum number of rules, and
- simplicity – single rules are not complex.

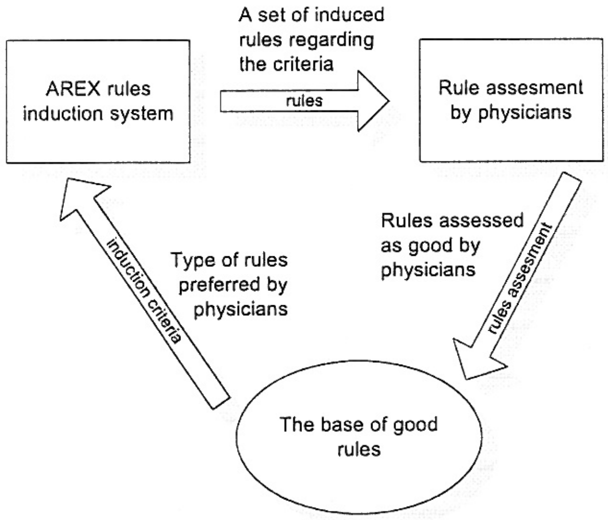

Fig. 1. The outline of our approach to the discovery of new knowledge.

Data to use as the source for knowledge discovery system are represented with the set of training objects o_1, ..., o_N. Each training object o_i is described with the values of attributes a_{i1}, ..., a_{ik} and the accompanied decision ω_i from the set of m possible decisions $[\Omega_1, ..., \Omega_m]$. Attributes can be either numeric (value is a numbers from a continues interval) or discrete (value is one from the discrete set of all possible values). Of course it is not necessary for all values to be known, as the algorithm can work also with missing values.

Knowledge that is discovered with the help of our algorithm is represented with a set of classification rules. Each single rule in a set is in the following form:

`if <condition> then <decision>`, where

`<condition> :=` $<c_1>$ and ... and $<c_d>$, and
`<decision> :=` ω, $\omega \in [\Omega_1, ..., \Omega_m]$

In this manner the rules are clear and easy enough for further analyses and in the same time their functional power is strong enough for successful classification. A set of classification rules can lose on the understandability if the number of rules in a set is too high. On the other hand the classification accuracy would decrease if the number of classification rules in a set is too low. Following this statement all the rules are arranged regarding their importance into several levels to form the hierarchical classification model (figure 2).

An approach to the distribution of data objects into subsets that will be covered by the rules on a specific level is defined first. For this purpose an algorithm for the construction of decision trees was defined [1] – with the help of classification by decision trees it can be determined which objects are appropriate to be used on a specific level of classification.

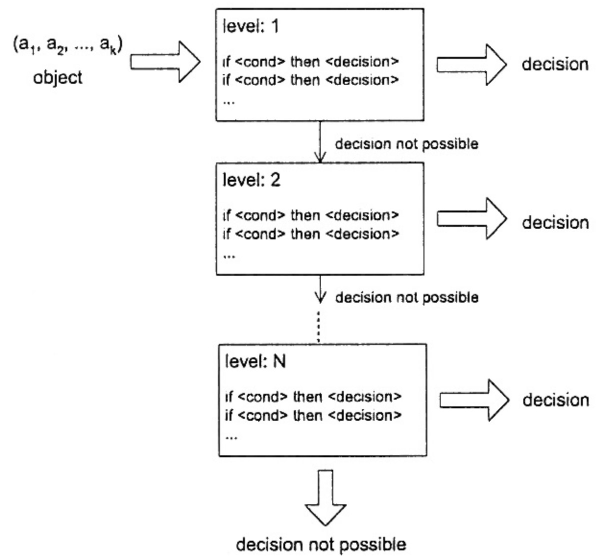

Fig. 2. Multi-level hierarchical classification model.

The complete rules extracting algorithm has been called AREX (*Automatic Rules EXtractor*). AREX uses as the input a training set of objects and based on those objects classification rules are built, hierarchically distributed over several levels. Algorithm AREX includes a hybrid system of two basic algorithms: 1) an evolutionary algorithm for the construction of decision trees [1] that is used to distribute the objects between levels and to build a part of the initial set of classification rules, and 2) *proGenesys* system that allows automatic evolution of programs in an arbitrary programming language [2] and is used for the construction of classification rules on a single level. The basic outline of the AREX algorithm is illustrated in figure 3 and can be described in the following steps:

1. input: a set of training objects S and clearness tolerance t
2. build N evolutionary decision trees upon objects from S
3. copy all objects from S, classified in leaves in accordance with clearness tolerance t in at least $\lceil N/2 \rceil + 1$ trees, to a set S^* and clear from a set S
4. from all N decision trees create M initial classification rules (all leaves in accordance with clearness tolerance t are used)
5. with the *proGenesys* system create another M classification rules randomly; initial population now contains $2 \times M$ classification rules
6. using the *proGenesys* system evolve the final set of rules for classifying objects from S^* at the current level
7. find the optimal final set of rules using simple optimization

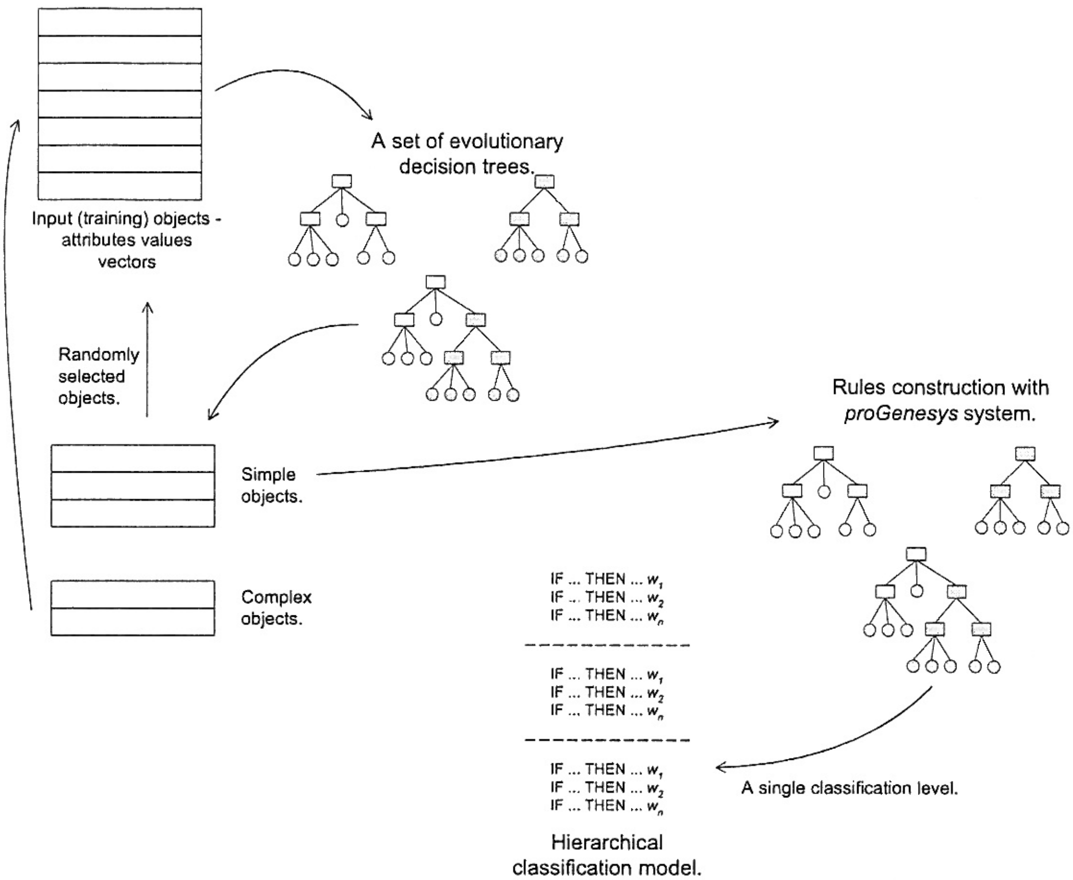

Fig. 3. The diagram of the AREX algorithm.

8. if S is not empty (there are still some unclassified objects for the next level): 1) add $|S|$ randomly chosen objects from S^* to S; 2) increase clearness tolerance $t=2{\times}t$; 3) repeat the whole procedure from step 2
9. finish if S is empty

A. Genetic algorithm for construction of decision trees

First step of the genetic algorithm is the induction of the initial population. A random decision tree is constructed based on the following algorithm:

1. input: number of attribute nodes N that will be in the tree
2. select an attribute X_i from the set of all possible attributes and set it as a rood node t
3. in accordance with the selected attribute's X_i type (discrete, continuous) define a test for this node t: 1) for continuous attributes in a form of $f_t(X_i) < \phi_i$, where $f_t(X_i)$ is the attribute value for a data object and ϕ_i is a split constant; 2) for discrete attributes two disjunctive sets of all possible attribute values are randomly defined
4. connect empty leaves to both new branches from node t
5. randomly select an empty leaf node t (the probability of selecting an empty leaf is decreased with the depth of the leaf in a growing tree)
6. randomly select an attribute X_i from the set of all possible attributes (the probability of choosing an attribute depends on a number of previous uses of that attribute in a tree – in this manner unused attributes have better chances to be selected)
7. replace the selected leaf node t with the attribute X_i and go to step 3
8. finish when N attribute nodes has been created

For each empty leaf the following algorithm determines the appropriate decision class: let S be the training set of all training objects N with K possible decision classes $d_1, .., d_K$ and N_i is the number of objects within S of a class d_i. Let S^t be the sample set at node t (an empty leaf for which we are trying to select a decision class) with N^t objects; N_i^t is the number of objects within S^t of a decision class d_i. Now we can define a function that measures a potential percentage of correctly classified objects of a class d_i:

$$F(t,i) = \frac{N_i^t}{N_i} \qquad (1)$$

Decision d_i^t for the leaf node t is then set as a decision d_i, for which $F(t,i)$ is maximal.

The ranking of an individual DT within a population is based on the local FF:

$$LFF = \sum_{i=1}^{K} w_i \cdot (1 - acc_i) + \sum_{i=1}^{N} c(t_i) + w_u \cdot nu \qquad (2)$$

where K is the number of decision classes, N is the number of attribute nodes in a tree, acc_i is the accuracy of classification of objects of a specific decision class d_i, w_i is the importance weight for classifying the objects of a decision class d_i, $c(t_i)$ is the cost of using the attribute in a node t_i, nu is number of unused decision (leaf) nodes, i.e. where no object from the training set fall into, and w_u is the weight of the presence of unused decision nodes in a tree.

B. System proGenesys for automatic evolution of rules

For constructing classification rules we developed a system for the evolution of programs in an arbitrary programming language, described with BNF productions – *proGenesys* (program generation based on genetic systems) [2]. In our approach an individual is represented with a syntax tree (a derivation tree). To get the final solution this tree (genotype) is transformed into a program (phenotype) [8]. In the case of rule induction, a single rule is represented as a simple program (i.e. a derivation tree), that evolves through generations.

A good classification rule should simultaneously be clear (most of the objects covered by the rule should fall into the same decision class) and general (it covers many objects – otherwise it tends to be too specific). Those two criteria can be measured with the following formulas:

$$generality = \frac{num.\ of\ classified\ objects - 1}{num.\ of\ objects\ of\ this\ decision\ clas} \qquad (3)$$

$$clearness = 1 - \frac{\frac{\omega_2}{\Omega_2}}{\frac{\omega_1}{\Omega_1}} \qquad (4)$$

where ω_1 is number of objects covered by the rule that belong to the most frequent decision class, ω_2 is number of objects covered by the rule that belong to the second most frequent decision class, Ω_1 is number of all objects in the training set that belong to the most frequent decision class of the rule, and Ω_2 is number of all objects in the training set that belong to the second most frequent class of the rule. Now a fitness function can be defined as

$$FF = clearness \times generality + \sum_{i=1}^{N} c(t_i) \qquad (5)$$

where the last part represents a cost of the use of specific attributes, the same as in the local fitness function *LFF* in building decision trees.

C. Finding the optimal set of rules

System *proGenesys* is used to evolve single rules, whereas for the classification of all objects on a specific level a set of rules is required. For this purpose between all the evolved rules a set of rules should be found that together classify all the objects – with high classification accuracy and a small number of rules. A problem is solved with a simple genetic algorithm that optimizes the following fitness function:

$$FF = \sum_{i=1}^{K} w_i \cdot (1 - acc_i) + \sum_{i=1}^{N} c(t_i) + w_m \cdot nm + w_u \cdot nu \qquad (6)$$

where the first two parts are the same as in the LFF for building decision trees, and instead of a penalty for unused decision nodes penalties for multiple classified objects and non-classified objects are used.

III. CASE STUDY

The early and accurate identification of cardiovascular problems in children patients is of vital importance. In order to help the clinicians in diagnosing the type of the cardiovascular problems in young patients, computerized data mining and decision support tools are used which are able to help clinicians to process a huge amount of data available from solving previous cases and suggest the probable diagnosis based on the values of several important attributes. Clearly, black-box classification methods (neural networks for example) are not appropriate for this kind of task, because the clinical experts need to evaluate and validate the decision making process, induced by those tools, before there is enough trust to use the tools in practice.

On the other hand, the evaluation of the induced classification rules produced by the computerized tools by a clinical expert can be an important source of new knowledge on how to make a diagnosis based on the available attributes. In order to achieve this goal, the classification process should be easily understandable and straightforward. Different kinds of knowledge discovery methods are therefore appropriate to do the job, and we decided to use the developed knowledge discovery method presented above.

A. Data sets

Two cardiovascular datasets have been composed to be used for the knowledge discovery process. Each of them contains data from 100 patients from Maribor Hospital. A protocol has been defined to collect the important data. The attributes include general data (age, sex, etc.), a health status (data from family history and child's previous illnesses), a general cardiovascular data (blood pressure, pulse, chest pain, etc.) and more specialized cardiovascular data – data from child's cardiac history and clinical examinations (with findings of ultrasound, ECG, etc.).

In the first dataset three different diagnoses are possible: innocent heart murmur, congenital heart disease, palpitations with chest pain. In the second dataset five different diagnosis were possible: innocent heart murmur, congenital heart disease with left to right shunt, aortic valve disease with aorta coarctation, arrhythmias, and chest pain.

Because the first database is actually the simplified version of the second one, we decided to use in our study only the second one.

B. Results

With our algorithm several rule-sets were evolved, which should provide some information on how the diagnosis of the cardiovascular problems can be made. Some interesting patterns have been discovered, especially for those decisions, which were not very often between the patients. This fact speaks in favor to our hypothesis that new patterns can be found in the cardiovascular data with the use of the appropriate knowledge discovery algorithm. An example of the induced rule-set is presented in table 1, together with the physician's evaluation of each rule.

IV. CONCLUSIONS

The results of searching for patterns in cardiovascular data, obtained with the presented approach to knowledge discovery with the help of physicians' assessment, turned out to be very good. The physician's evaluation of the final solution – a set of rules obtained through an iterative rule induction process – show that our method AREX produces mostly correct and reliable rules. What is especially important is the fact that between the rules in the induced ruleset some rules show a potential to be new knowledge. The assessment of physicians regarding the "goodness" of induced rules during the process defines the preferences criteria for inducing the following generations of rules. In this manner, more and more rules are induced, which comply with the given preferences by physicians. We may say that the obtained results satisfy our intentions and, more importantly, equip the physicians with a powerful technique to 1) confirm their existing knowledge about some medical problem, and 2) enable searching for new facts, which should reveal some new interesting patterns and possibly improve the existing medical knowledge.

REFERENCES

[1] V. Podgorelec, P. Kokol, "Evolutionary induced decision trees for dangerous software modules prediction," *Information Processing Letters*, vol. 82(1), pp. 31-38, 2002.

[2] V. Podgorelec, "ProGenesys – program generation tool based on genetic systems," in *Proceedings of the ICAI'99*, 1999, pp. 299-302.

[3] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, Inc., 2000.

[4] H.R. Anderson et al, *Clinicians Illustrated Dictionary of Cardiology*, London: Science Press, 1991.

[5] P. Kokol et al, "Decision Trees and Automatic Learning and Their Use in Cardiology," *Journal of Medical Systems*, vol. 19(4), 1994.

[6] U.M. Fayad, et al, *Advances in Knowledge Discovery and Data Minning*, AAAI, 1996.

[7] S.M. Weiss, N. Indurkhya, *Predictive Data Mining*, Morgan–Kaufman, 1998.

[8] C. Ryan, "Shades – a polygenic inheritance scheme," in *Proceedings of Mendel'97 Conference*, 1997, pp. 140-147.

[9] K.A. DeJong, "Hybrid learning using genetic algorithms and decision trees for pattern classification," in *Proceedings of the IJCAI Conference*, 1995.

[10] P.D. Turney, "Cost-sensitive classification: empirical evaluation of a hybrid genetic decision tree induction algorithm," *Journal of Artificial Intelligence Research*, vol. 2, pp. 369-409, 1995.

[11] A. Papagelis, D. Kalles, "Breeding decision trees using evolutionary techniques," in *Proceedings of the ICML'01*, 2001.

[12] V. Podgorelec, P. Kokol, "Towards more optimal medical diagnosing with evolutionary algorithms," *Journal of Medical Systems*, vol. 25(3), pp. 195-220, 2001.

[13] A.A. Freitas, "A survey of evolutionary algorithms for data mining and knowledge discovery," *Advances in Evolutionary Computation*, Springer-Verlag, pp. 819-845, 2002.

TABLE I

THE INDUCED SET OF RULES FOR THE DATASET "DATA FROM CLINICAL EXAMINATIONS AND CHILD'S HISTORY", TOGETHER WITH THE PHYSICIAN'S EVALUATION OF EACH INDUCED RULE. THE DECISION CLASSES ARE: CLASS 0 -> INNOCENT HEART MURMUR; CLASS 1 -> CONGENITAL HEART DISEASE WITH LEFT TO RIGHT SHUNT; CLASS 2 -> AORTIC VALVE DISEASE WITH AORTA COARCTATION; CLASS 3 -> ARRHYTHMIAS; AND CLASS 4 -> CHEST PAIN.

Induced rule	**Physician's evaluation of the rule**

Rule 1:
`IF Heart_murmur = yes`
`   ECHO = NotPerformed`
`THEN  class 0`
	<u>Possible</u>
	The most likely diagnosis is innocent heart murmur, because it was decided on the basis of ECG and clinical examinations that echocardiography is not necessary. There is a possibility that a congenital heart disease was missed.

Rule 2:
`IF Heart_murmur = yes`
`   ECHO = Normal`
`THEN  class 0`
	<u>Very good</u>
	With a classical relevant diagnostic method, like echocardiography, a structurally normal heart was proved. Therefore, a heart murmur could only be classified as an innocent heart murmur.

Rule 3:
`IF Heart_murmur = no`
`   Stenocardy = no`
`   Arrhythmias = no`
`THEN  class 0`
	<u>Not correct</u>

Rule 4:
`IF Dystrophy = yes`
`   Stenocardy = no`
`   ECHO = Pathological`
`THEN  class 1`
	<u>Very good</u>
	With the echocardiography confirmed congenital heart disease, that manifests clinically in small children with physical growth retardation and dystrophy. Those children do not have chest pains in the form of stenocardy.

Rule 5:
`IF Stenocardy = no`
`   RR_Diastolic <= 55`
`   ECHO = Pathological`
`THEN  class 1`
	<u>Possible – potentially new knowledge</u>
	How the hemodynamic changes in patients with congenital heart disease with left to right shunt influence to the possible changes of systemic arterial blood pressure?

Rule 6:
`IF Heart_murmur = yes`
`   Stenocardy = yes`
`   ECHO = Pathological`
`THEN  class 2`
	<u>Correct, logical rule</u>
	In the case of aortic valve disease it can come to the overload of left heart and with effort often to the stenocardy pain.

Rule 7:
`IF Dystrophy = no`
`   RR_Diastolic > 55`
`   ECHO = Pathological`
`THEN  class 2`
	<u>Correct, logical rule</u>
	Medical knowledge acknowledges the changes of systemic arterial blood pressure in the case of aortic valve disease.

Rule 8:
`IF Heart_murmur = no`
`   Arrhythmias = yes`
`THEN  class 3`
	<u>Correct</u>
	In the case of arrhythmias usually there is no evidence of heart murmur. Therefore, in this case the primary rhythm disturbances without congenital heart disease is the obvious diagnosis.

Rule 9:
`IF Heart_murmur = no`
`   Stenocardy = yes`
`   Arrhythmias = no`
`THEN  class 4`
	<u>Correct</u>
	No physical heart disease; neuro-muscular etiology of precordial pain.