

A Rule-based Approach for Medical Decision Support

Silvia Canale, Francesco Delli Priscoli, Guido Oddi, Antonio Sassano, Silvano Mignanti

Dipartimento di Ingegneria Informatica, Automatica e Gestionale
Università degli Studi di Roma "Sapienza", Rome, Italy
{canale; dellipriscoli; oddi; sassano; mignanti}@dis.uniroma1.it

Abstract—This paper describes the medical Decision Support System (DSS) designed in the framework of the Bravehealth (BVH) project. The DSS is the heart of the data processing performed in Bravehealth, and it is aimed at enriching the medical experience to support the doctors in the decision-making processes. The paper focuses on the flexible and effective DSS architecture placed at a Remote Server side. Moreover, a Data Mining prototype algorithm, supported by the architecture, is proposed, along with encouraging test results.

Keywords—Medical Decision Support; Data Mining; Machine Learning.

I. INTRODUCTION

Recently, machine learning methods have been largely applied to a high number of medical domains. Improved medical diagnosis and prognosis have been shown to be achieved through automatic learning from past experiences, to detect and translate regularities in analytic rules that can be used to classify new patient records. Machine learning algorithms have been shown to be very successful in cardiovascular disease analysis and detection [2,6,10] and electrocardiogram (ECG) beat classification [3,4,5].

In a recent study [7], the cardiovascular diseases are indicated as the first mortality cause in women. In Europe, approximately 55 percent of women's deaths is caused by cardiovascular diseases, especially coronary disease and stroke. The Framingham heart study [8] gave a significant contribution by revealing the impact of factors as smoking, hypertension, dyslipidaemia, diabetes mellitus, obesity, male gender, and age on developing of cardiovascular disease. That was the basis for defining a classification system for identifying the cardiovascular risk class (low, medium, and high) for women on the basis of their characteristics in terms of relevant impact factors [7,9,16]. In Europe, the cardiovascular mortality and morbidity in women are some of the highest. Typically, for confirming the presence of cardiovascular disease, the patients are submitted to different tests (biochemical tests, rest ECG, stress test, echocardiography or angiography). Some of them are invasive for patients, and expensive and time consuming.

In [6], Data Mining techniques were used to identify the high risk patients and evaluate the relationships between cardiovascular risk factors and resulting cardiovascular diseases, differently by the gender of patients. The purpose of study proposed in [6] was to compare the capacity of different data mining methods: the study was conducted on an 825 people sample and the data were collected from general practitioners' files (including, for every patient, information about blood pressure, hypertension, body mass index, glycaemia, the presence or absence of cardiovascular

disease on the basis of standard medical definition, etc.). The complete sample included 825 data records of 145 attributes, which reduced, after data cleaning up process, to 303 of the initial set of patients. Two data mining algorithms were used to analyse the sample and to identify the relationships between the attributes and the label indicating the presence or absence of cardiovascular disease. The former one, the Naïve Bayes approach, provided acceptable results regarding identification of patients with coronary artery disease and acceptable results in identification of patients without stroke or without peripheral artery disease (in particular, only 62% of patients with coronary artery disease (in particular, only 62% of patients with coronary artery disease were classified). The latter was a decision tree training algorithm that succeeded to capture 72.6% of relevant information in patients with coronary artery disease but was also incapable to capture relevant information for those with strokes or peripheral artery disease (percentages being also equal to zero). These results were absolutely satisfactory if compared with the success rate achieved by data mining methods applied to different medical test (Liver diseases, Breast Cancer, etc.) and, in particular, to specific heart disease data sets (as the Cleveland HEART data set of UCI repository). Nonetheless, they were totally unsatisfactory for safe clinical protocols. This was due, in authors' opinion, to: (i) the number of patients in the data set and the cleaned data were insufficient to assess the quality of any method; (ii) the use of standard methods, taken "off the shelf" from the literature, without any specific reference to heart disease environment, was not able to produce effective classifiers.

Consequently, a double effort was necessary. On one side the Bravehealth project will be able to collect, validate and clean large amount of patient data. In this respect the idea of remotely collecting patient data directly from a so-called *Wearable Unit* (see [1] for further details) was crucial. The reported description of the logical architecture of the Bravehealth Decision Support System (DSS) highlights its capability to collect and validate large amount of data related to "real" patients. The second effort was that to devise *new classification methods*, able to cope with the large datasets and to be "tuned in" the specific medical application of Bravehealth. For this purpose, Bravehealth proposed a Boosting algorithm based on a "problem specific Kernel". The Kernel of a boosting algorithm embodies the similarity (or dissimilarity) of the different patients. One could use a simple Linear Kernel (inner product of the data vectors) or a standard Gaussian Kernel (as in many algorithms proposed in the general literature). The Bravehealth approach is to devise, on the contrary, a *specific Kernel* for the problem and data faced in Bravehealth and testing its efficiency. This definition, along with the test, could be done only using the

massive amount of patient data collected in Bravehealth. Nevertheless, the authors had already started the design of a prototype algorithm briefly described in Section IV, based on the boosting algorithm, in order to check, on well known and available test problems, the effectiveness of the boosting method. The results obtained by the proposed prototype algorithm on the widely used for new data mining methods validation *Cleveland HEART* dataset (available at the UCI Data Mining repository and comparable, in size, with the experiment in [6]) are very promising (as reported in Section IV) compared to the literature. As far as the authors know, the best results obtained by all the other research groups oscillate around 80% of accuracy (better than the results obtained by the “off the shelf” algorithms of [6]).

This paper is structured as follows. Section II illustrates the DSS architecture. Section III, IV, V and VI focus on the description of the sub-components defining the whole DSS (in particular Section IV illustrates the proposed algorithm and the results of the performed test). Finally, brief conclusions are drawn in Section V.

II. BVH DECISION SUPPORT SYSTEM (DSS)

The Bravehealth Decision Support System (DSS) has been conceived as a patient-centric, adaptive and flexible system capable to meet both patients’ and physicians’ needs, in order to support medical decisions and to account the actual expectations of both patients and physicians. Two main guidelines led to the Bravehealth DSS design and development: (i) the DSS is expected to be “close” to the patient, in the sense that the decision making process is fully affected by the patient’s actual health conditions and, in some cases, does actively involve the patient; (ii) the main users of the DSS (hospital physicians and medical researchers – hereinafter, these kind of users will be referred to as *Medical Supervisors*) must be granted the access the DSS and to insert or to update data about patients in a secure, immediate, efficient and effective way. Moreover, they expect that standard clinical models are implemented in the Bravehealth DSS ensuring that routine clinical consultations are made more consistent and informative. In addition, the DSS is supposed to support decision making by possibly providing additional information about potential new clinical models, that means useful information extracted from patients’ data by means of sophisticated data retrieval and Data Mining techniques. Taking into account these expectations, the Bravehealth DSS was designed to enhance the standard basic features of current medical DSSs.

On one side, the Bravehealth DSS is close to the physicians, in the sense of being a real decision support tool (not a “Doctor Substitution System”, as explicitly refused by physicians). Thus, the main components of the Bravehealth DSS are located at proper Remote Servers (RSs) located at the Medical Supervisors premises (e.g., in the hospitals). Hereinafter we will refer to these components as RS DSS. Using standard medical protocols, the RS DSS is able to classify patients affected by CVD into one of three categories: *High*, *Medium* or *Low Risk*. These definitions are based on rules drawn from clinical practice. Accordingly, the RS DSS can automatically generate notifications to be sent

to the physicians on the basis of deterministic rules derived by clinical practices and medical protocols. Each notification is part of a specific patient model, derived by standard Clinical Models, whose description is fully provided by the medical responsible. In addition, the Bravehealth RS DSS analyzes medical parameters and context data in order to extract useful information, in terms of rules and patterns for patient classification and profiling, by means of the Data Mining module. This additional feature is the most innovative part of the RS DSS, since advanced Data Mining algorithms, tailored to the Bravehealth environment, are adopted. These algorithms can require rather heavy processing capabilities; nevertheless, as hereafter explained, the RS DSS is organized so that the heavier calculations are performed off-line. The extracted information is real-time presented to the Medical Supervisors as suggestions.

On the other side, the Bravehealth DSS is close to the patient in the sense that a secondary subsystem, namely the Lightweight Decision Support System (LDSS), is completely dedicated to the patient care. The LDSS component is decentralized with respect to the main RS DSS components and is located at the Patient Gateway (PG): so, hereinafter we will refer to this component as PG LDSS. The main aim of the LDSS is that of filling the gap between patients and physicians when the patients are at home, especially in critical situations (emergencies, PG-RS communication link problems, RS server problems, etc.). Even the PG LDSS is supported by Data Mining algorithms; nevertheless, these algorithms have been designed with the requirement of being particularly light so that they can run even on a low processing computer implementing the PG located at the patient's premises. This paper mainly focuses on the description of the architecture, the features and the embedded algorithms of the DSS at the RS. Nevertheless, the concept of a Data Mining intelligent agent “close to the patient”, represented by the PG LDSS, is an innovative concept proposed and being developed within the Bravehealth project and further research papers will be dedicated to its architectural, algorithmic and test results.

Figure 1 shows, using the UML formalism, the functional blocks of the DSS at the Remote Server (RS DSS), and details its components and both its internal and external interfaces. The architecture components are described in detail in [1]. The following sub-sections describe in detail the subcomponents of the *Runtime Environment*, namely the core of the RS DSS, which is in charge of extracting from all the available data the useful information to be presented in real-time to the Medical Supervisors.

III. NOTIFICATION RULES ENGINE AND SUGGESTED RULE ENGINE (ON-LINE PROCESSING)

The Sensor Data Management System and the User Management System store patients’ measured data (ECG, Breath rate, SpO₂, Arterial Blood Pressure, Activity level, Fluid Index or bioimpedance, Temperature), and consolidated medical evaluation (e.g., in terms of risk classes: *Low Risk*, *Medium Risk*, *High Risk* provided and validated by doctors and physicians), respectively. All these data are provided to the Runtime Environment via the *Data*

CRUDS and the *Patient Record CRUD* interfaces, respectively. These data are properly pre-processed by an ad hoc pre-processing module as explained below.

The above-mentioned data are used, on the one hand, by the Notification Rules Engine, which is in charge of applying to these data the logic rules defined on the basis of medical protocols and standard procedures adopted by the physicians. These last rules are uploaded by the Rule Supervisors (these are particular kinds of Medical Supervisors authorized to manage the DSS rules) by means of the Rule Supervisor FE in the Medical Knowledgebase Management System (MKMS), in charge of storing the various rules. Since these rules are trusted, they are labeled in the MKMS as “Active Standard Rules”. Thus, the Notification Rules Engine uploads the Active Standard Rules from the MKMS and on-line applies each Active Standard Rule to the data acquired

IV. DATA MINING (OFF-LINE PROCESSING)

The RS Data Mining component is in charge of the main advanced features of the Bravehealth DSS. The Data Mining component is split in the following four sub-modules:

A. Pre-processing module

Data Mining algorithms cannot be fed with raw data: pre-processing of data greatly increases the reliability and the performance of the algorithms. This module is in charge of selecting, organizing and processing the available data in the most suitable way for the data analysis, performed by the Data Mining Engine. The data available in this module are: (i) *medical parameters* coming from the Wearable Units through the PGs and stored at the Sensor Data Management System; (ii) *ECG descriptors and/or other*

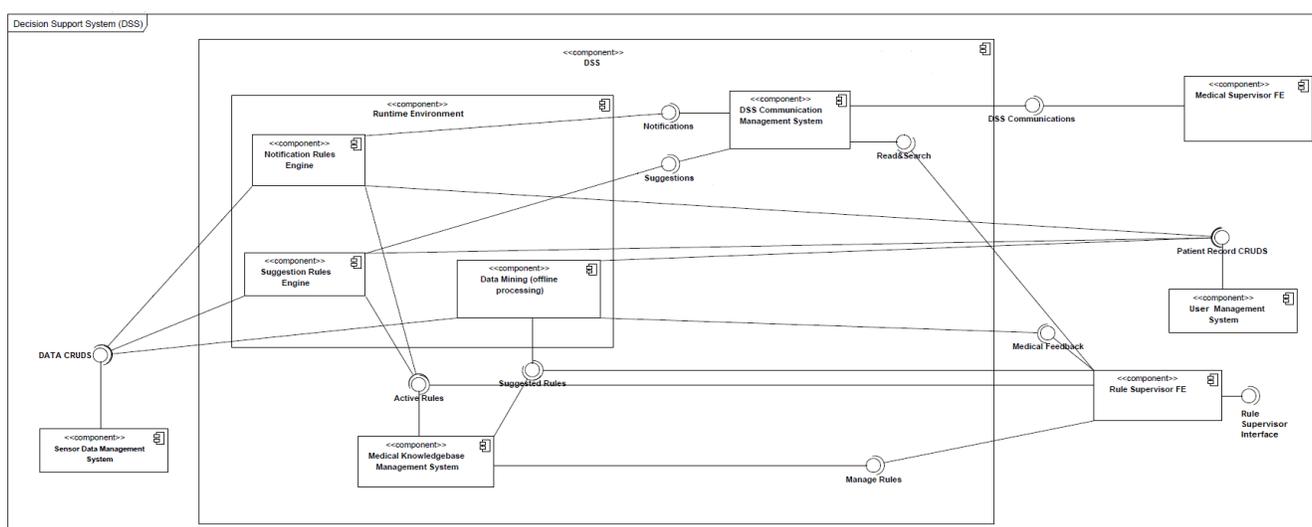


Figure 1. RS Decision Support System (DSS)

from the Sensor Data Management System and from the User Management System; the application of such rules possibly leads to “notifications”, which are sent to the Medical Supervisors. On the other hand, such acquired data are also used by the off-line Data Mining component, detailed in the next section, to produce new rules, which are stored in the MKMS being labelled as "Suggested Rules", since these rules, differently from the ones based on medical protocols, are inferred on the grounds of Data Mining techniques and therefore need to be validated by the Rule Supervisors case by case. For this reason, the Suggested Rules are not active by default. Nevertheless, as these rules are validated by the Rule Supervisors, they become “Active Inferred Rules” and can be used on-line by the Suggested Rule Engine. Thus, the Suggested Rule Engine uploads the Active Inferred Rules from the MKMS and on-line applies each Active Inferred Rule to the data acquired from the Sensor Data Management System and from the User Management System; the application of such rules possibly leads to “suggestions”, which are eventually received by the Medical Supervisors. All the rules (both the Suggested and the Active Standard/Inferred ones) are stored in the MKMS.

physiological parameters coming from the Signal Processing performed at the PG and/or at the RS and stored at the Sensor Data Management System; (iii) *context factors* elaborated at the PG and stored at the Sensor Data Management System; (iv) configuration and patient data coming from Medical Supervisors and stored in the User Management System.

The first task of the pre-processing module is to render all the data homogeneous. Then, three main pre-processing techniques are applied: (i) *Sample selection*: some data may be unreliable (e.g., because of typos in data enter, imprecise medical measurements, etc.); and an expert (doctor or medical researcher) is needed to decide their relevance for data analysis. If the sample selection is not provided, the system extracts the sample data in unsupervised way, according to the statistical distribution of the available data set. The well known structured k-fold cross validation procedure is adopted by Bravehealth system for sample validation and test. (ii) *Feature selection*: besides the sample selection, proper feature selection and extraction algorithms can be adopted in order to complete the set of significant features by means of specific indexes defined ad hoc by pre-processing environment. In the Bravehealth system, these

algorithms are based on a well known supervised machine learning model, namely the L_1 -norm Support Vector Machines [14]. (iii) *Denoising*: finally, standard denoising algorithms are applied to correct statistical errors.

B. Data Mining (DM) Engine

The Bravehealth RS DM engine is the “core” of the RS Data Mining component. It includes innovative models based on data analysis and machine learning algorithms able to infer, in an off-line fashion, new rules which, after being properly validated by the Rule Supervisors, are applied by the on-line Suggestion Rules Engine to the pre-processed data. The DM engine conceived in Bravehealth includes several machine learning based models, all tailored to the specific cardiac diseases considered in Bravehealth: all these models are simultaneously active and automatically selected. These models have to operate under the control of specialized Rule Supervisors, not only authorized to access the DSS (via the Rule Supervisor FE and the *Medical Feedback* interface), but also to manage the models in question and tune specific parameters.

The DM engine analyses all historical pre-processed data with the aim of identifying correlations, regularities and patterns in such data and to *serve as a prediction* on the patients’ health conditions. Information is extracted in the form of general patterns, such as logical rules or decision trees, that are stored in the Medical Knowledgebase Management System (MKMS) as “suggested rules” and then are studied by Rule Supervisors both to validate the suggested rules in question and to further refine the adopted models. In addition, the DM engine is able to identify abnormal behaviors or risk situations which are notified to the Medical Supervisors. The above-mentioned analysis is performed by both unsupervised (e.g., data clustering) and supervised (e.g., data classification and regression, pattern recognition) machine learning algorithms. Some of the models adopted for performing this analysis are open source implementation (e.g., WEKA), whereas other ones (e.g., exact boosting model) have been developed and implemented ad hoc for the Bravehealth purposes. Data Mining based medical models are independent of the medical protocols and standard procedures; conversely, they are totally based on proper Data Mining models, such as Decision Trees, Bayes Networks, Rule Induction Algorithms and Neural Networks, Boosting and Kernel models (e.g., Support Vector Machines). In particular, Boosting techniques have emerged in machine learning as ones of the most promising and powerful methods for supervised learning [11]. These techniques are the ones which have been selected for being designed, developed and implemented in Bravehealth. In this respect, the innovative Boosting model which has been defined and is being developed ad hoc for the Bravehealth environment, is obtained through a proper combination of a set of given base classifiers, usually called *weak learners*, to yield one classifier that is stronger than each individual base classifier. In Bravehealth we coped with the problem of combining Support Vector Machines (SVMs), properly adapting to the Bravehealth environment the approach presented in [12].

Following [14], the Boosting problem is formulated as a Linear Programming problem (LP). The dimension of the LP problem is related (via the Kernel matrix representing the similarity measure) to the number of test points (number of patient records) and hence the LP to be solved will be larger as the patient data will be collected, cleaned and stored by the DSS. The algorithm proposed in Bravehealth tackles the problem of solving LP problems with a huge number of variables by improving the solution scheme proposed in [14] and adapting to the boosting environment a standard technique used in LP theory: *Column Generation*. Column generation is a general method for solving large LP problems by iteratively solving a “reduced problem” on a subset of variables and fixing the others to zero. The solution of the “reduced problem” is optimal for the original problem if suitable values associated with the zeroed variables (the “reduced costs”) are non-negative. At each step, the reduced costs of the variables fixed to zero are evaluated, and only a limited number of “promising” variables (named entrant columns) with negative reduced cost are included in the set of variables considered in the current iteration (the so called “auxiliary problem”). Each entrant column is chosen by a “look up” procedure that automatically evaluates the reduced cost of the variables fixed to zero. The related Support Vector Machine is inserted in the subset of promising columns. By generating automatically one additional column at each iteration, the dimensions of the master problem to be solved increase slowly, and the solution algorithm is very fast. When the number of generated columns becomes considerable, the algorithm selects a subset of columns of the master problem that can be removed without affecting the current solution.

This paper shows the results obtained through the implementation of this algorithm when applied to the problem *Cleveland HEART* (303 patient data concerning heart diseases) available at the UCI data mining repository [15]. These results indicate that the “boosting + column generation” approach is capable to find very good accuracy results and ready to solve the mining problems (of increasing dimension) generated by the routine activity of Bravehealth DSS (patient data collection via Wearable Unit, in primis). A brief description of the main features of the proposed method must start from a quick sketch of the standard learning protocol. The preliminary action is the partition of the dataset in two sets: the *training* and the *test set*. The *training set* simulates the data available in the (off-line) learning phase. The classifier (its parameters) is (are) defined on the basis of the information carried by the training set, ignoring the data included in the test set. The *test set* simulates the data that will become available on line (i.e. the vital parameters measured by the Wearable Unit of a new patient and acquired by the DSS). The DM Engine uses the “boosting+column generation” approach to define a classifier which consists of a linear combination of Support Vector Machines (SVM). The classifier is defined on the basis of known and clean data represented by the training set. The classifier will subsequently be used on-line to assess the criticality of the vital parameters of unknown patients (represented by the test set in our experiment).

The main purpose of the training phase is to define a classifier which determines whether a patient is in *critical conditions* or not, under some pre-defined medical point of view, only for those patients in the training set that *it is known in advance that they were in critical conditions* for that parameter. Such a knowledge is used to assess the quality of the algorithm. The most diffuse wrong idea is to consider the “best” classifier as the one that provides the correct answer for every patient in the training set, but this simply means that the classifier is tailored for the training set and often unable to generalize its diagnosis to a new, unknown, patient. This is the so called *overtraining effect*. Conversely, the correct learning strategy is that of optimizing a functional which takes into account both the *prediction accuracy* over the training set and the capability of recognizing cases not included in the training set (*generalization*). In the proposed algorithm, this multi-objective problem is solved by maximizing the accuracy on the training set, while constraining the capabilities of the classifier (reduced set). By constraining the classifier into not performing excessively well on the training set, it should be able to generalize to the test set. In more detail, the proposed classifiers are linear combinations of Support Vector Machines (SVM) and each SVM is defined by an *hyperplane* whose variables correspond to the components of the training points (indeed this is true only if the Kernel is not used but let assume it for simplicity). The proposed solution to the “overtraining effect” is to reduce the set of SVM to be included in the linear combination (boosting) by imposing an upper bound to the norm of the coefficients of the hyperplane defining the SVM (*norm-UB*). A very low value of the upper bound produces classifiers unable to properly classify the elements of the training set, while a very high value (infinite) for the upper bound imposes no limit upon the choice of the optimal classifier and produces the feared “overtraining effect”. The optimal upper bound and hence the optimal classifiers in terms of accuracy and generalization must lay in the middle and correspond to the optimal value of the norm-UB.

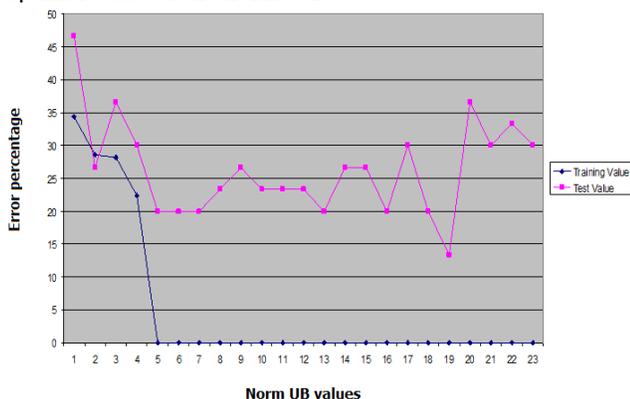


Figure 2. Value Function test results

A way to visualize the overall behavior of the learning process is by plotting the so called *value function* of the proposed optimization problem for increasing values of the norm-UB. The value function is the value of the error

percentage of the optimal classifier on the training set obtained by restricting the choice of the SVMs to those having coefficient smaller than a suitable value of the norm-UB. The value function of the tests performed by the authors is plotted (in blue) in Figure 2. The y-axis reports the percentage error while the x-axis represents 23 different and increasing values of the norm-UB (the values are not important since it is important to have an increasing series of upper bounds). The behavior of a value function for increasing values of the norm-UB is quite predictable. It starts from high values of prediction error in correspondence of the lowest values of the norm-UB and decreases to 0 (equivalently 100% prediction accuracy on the training set).

But what happens to the prediction accuracy on the *test set*? Two phases are present: a first phase in which the quality of the results on the test set follows the quality of the result in the training set and the prediction error on the test set decreases; and a second phase in which the improvement of the accuracy on the training set produces an increasing percentage of wrong answers on the test set. This second phase can be defined the *overtraining phase* and the error correspond to the fact that the algorithm is performing “too well” on the training set. The best classifiers should be searched at the interface of the two phases. This area is here informally defined as the *knee* of the curve corresponding to the accuracy error on the test set. Figure 2 reports the experimental results, in terms of prediction error in percentage constrained by the norm-UB on the x-axis, of the proposed prototype algorithm upon the Cleveland HEART data. The 303 patients of the data set have been partitioned in a training set containing 270 of them (a lower number would have made the test of the boosting+column generation procedure non significant) and leaving 33 patients unknown to the algorithm (test set). The value function starts from a 35% error for the lowest value of the norm-UB. In this case, the UB is so low that it is not possible to find a classifier which properly classifies the patients in the training set. By increasing the norm-UB the error percentages decreases until it reaches the value zero (the fifth value of the norm-UB). The error percentage remains to zero even though it is related to different classifiers for each different value of the norm UB (with different generalization capabilities).

The red plot shows the classification results on the test set, obtained by the classifier produced for each value of the norm-UB. As one can easily see the prediction error on the test set has an almost descending trend up to the 19th value of the norm UB (corresponding to a 13.33% error) and then it increases above 30% for all the subsequent values of the norm-UB. Hence, a “knee” has been found and an optimal classifier with 86.66% accuracy corresponding to the knee. As far as the authors know, this is the best classifier obtained so far for this particular problem.

V. MODEL SELECTION MODULE

As a request arrives to the Data Mining component, all the realized algorithms implementing different machine learning models are executed (Neural Networks, Decision Trees, Boosting, etc), and their outputs are automatically evaluated

in terms of accuracy and reliability. This module has the task of selecting the model (or the combination of models) which is the most appropriate to the current request on the basis of specific parameters and criteria provided by the Medical Supervisors. Driving parameter can be accuracy and reliability determined according to the examined case; model selection can be automatically performed basing those parameters. Moreover, the Bravehealth system foresees a hybrid automatic-manual model selection in which Medical Supervisors can express their preference to a particular model for each examined case.

VI. PATTERN VISUALIZATION MODULE

An important issue in Data Mining applications for medical diagnosis and risk prediction models is that the results of computer-based analysis have to be communicated to people in a clear way, to facilitate the interaction in the decision making process. The output of the DM Engine is represented by general patterns (logical rules, decision trees, etc.) that are provided to Rule Supervisors for inspection and validation. The output may not be immediately clear to non-specialized operators; therefore, a Pattern Visualization module is needed, to represent the patterns found by the DM Engine in a graphical representation, suitable for doctors and medical researchers. The Bravehealth Pattern Visualization module stores and displays data in a customizable way, offering efficient access to data and data managing tools for continuous patient monitoring.

VII. CONCLUSION AND FUTURE WORK

A key characteristic of the Bravehealth approach is that all the data processing procedures, from the data pre-processing to the output visualization, is performed according to a patient-centric vision and with tight control of doctors and medical researchers, also to encourage its use by the medical audience, usually skeptical about automatic assistance. Moreover, the Bravehealth approach includes several innovative features: (i) the use of a two-scale DSS including a *light* data processing taking place at the PG, and a more *heavy* data processing demanded to the RS; (ii) the adoption of a flexible architecture of the RS DSS based on an off-line Data Mining engine including several Data Mining models which can be adaptively selected (either in an automatic, or in an hybrid automatic/manual fashion) on the basis of the examined case for providing on-line (real-time) notifications and suggestions to the Medical Supervisors; (iii) the adoption of Data Mining models tailored to the Bravehealth environment (e.g., Boosting models based on SVMs as the proposed one); (iv) the adoption in the PG LDSS of powerful clustering algorithms tailored to the real-time classification of patient records filled with the data received from the WU.

This paper has presented the basis, along with very encouraging results of tests applied on well known available data, of the on-going Data Mining algorithms development, which, compliantly to the best practice in Data Mining, will be carefully tailored to the actual data which will be available either during the Bravehealth or other similar projects, and/or in eHealth based industrial applications. The expectation is that, thanks to specific Kernels, the proposed

boosting algorithm could represent a "quantum leap" of the capacity of (i) predicting heart diseases and (ii) providing a more accurate classification of the patients' health status.

ACKNOWLEDGMENT

The work described in this paper is partially based on the results of the ICT FP7 Integrated Project Bravehealth, under Grant Agreement no. 248694. The European Commission has no responsibility for the content of this paper. The information in this document is provided as is and no guarantee or warranty is given that the information is fit for any particular purpose. The user thereof uses the information at its sole risk and liability.

REFERENCES

- [1] S. Canale et al., "The Bravehealth Software Architecture for the Monitoring of Patients Affected by CVD", 5th eTELEMED, 2013, Nice, France.
- [2] R. B. Rao, S. Krishnan, and R. S. Niculescu, "Data mining for improved cardiac care", ACM SIGKDD Explorations Newsletter archive, Vol.8(1), 3-10, June 2006.
- [3] M. Engin, "ECG beat classification using neuro-fuzzy network", *Patt. Rec. Lett.* 25, 15, 1715-1722, 2004.
- [4] S. Barro, M. Fernandez-Delgado, J.A. Vila-Sobrino, C. V. Regueiro, and E. Sanchez, "Classifying multichannel ECG patterns with an adaptive neural network", *IEEE Eng. Med. Biol. Mag.* 17 (1), 45-55, 1998.
- [5] A. De Gaetano, S. Panunzi, F. Rinaldi, A. Risi, and M. Sciandrone, "A patient adaptable ECG beat classifier based on neural networks", *App. Math. & Comp.* 213,1,243-249, 2009.
- [6] A. V. Sitar-Taut, D. Zdrengea, D. Pop, and D. A. Sitar-Taut. "Using Machine Learning Algorithms in Cardiovascular Disease Risk Evaluation", *JACSM.* 3(5), 29-32, 2009.
- [7] M. Stramba-Badiale et al., "Cardiovascular diseases in women: a statement from the policy conference of the European Society of Cardiology", *Eur Heart J* (27), 2006.
- [8] <http://www.framinghamheartstudy.org>, accessed on 31st January 2013.
- [9] L. Mosca et al., "Evidence-Based Guidelines for Cardiovascular Disease Prevention in Women: 2007 Update", <http://circ.ahajournals.org/content/115/11/1481.full>, accessed on 31st January 2013.
- [10] H. G. Lee, K. Y. Noh, and K. H. Ryu, "A Data Mining Approach for Coronary Heart Disease Prediction using HRV Features and Carotid Arterial Wall Thickness", *Int. Conf. on BioMedical Eng. and Informatics*, Vol. 1, 200-206, 2008.
- [11] R. Schapire, "A brief introduction to boosting", 16th IJCAI-99.
- [12] C. Campbell, and Y. Ying, "Learning with Support Vector Machines", Morgan and Claypool, 2011.
- [13] K. P. Bennett, A. Demiriz, and J. Shawe-Taylor. "A Column Generation Algorithm For Boosting". *Proc 17th ICML*, 2000.
- [14] J. Zhu, S. Rosset, T. Hastie, and R. Tibshirani, "1-norm Support Vector Machines", NIPS, 2003.
- [15] UCI Repository, <http://archive.ics.uci.edu/ml/datasets.html>, accessed on 31st January 2013.
- [16] A. Pietrabissa, C. Poli, D. G. Ferriero, and M. Grigioni, "Optimal planning of sensor networks for assets tracking in hospital environments", accepted for publication in *Decision Support System* (Elsevier), 2013.