Classifier ensemble,
Combination rule,
Decision support system

Tomasz ORCZYK[1], Piotr PORWIK[1], Marcin BERNAŚ[1]

MEDICAL DIAGNOSIS SUPPORT SYSTEM BASED ON THE ENSEMBLE OF SINGLE-PARAMETER CLASSIFIERS

This paper presents a medical diagnosis support system based on an ensemble of single parameter *k*–NN classifiers [1]. System was verified on a database containing real blood test results of diagnosed patients with a liver fibrosis. This dataset contains problems typical to a real medical data – especially missing values.
Paper also describes the process of selecting a subset of parameters used for further evaluation (feature selection/elimination algorithm). Complete database contains many parameters, but not all are important for diagnosis, thus eliminating them is an important step.
A comparison of proposed method of classification and feature selection with methods known from literature has also been presented.

1. INTRODUCTION

Decision support systems are used in a medicine as a medical diagnosis support systems. They allow for an indirect diagnosis of a presence or a progression of many diseases. These systems may have a different architecture, but they all rely on an archival medical records and verified diagnosis correlated with this records. Internally they may either rely on a human (expert) induced rules - which we then call expert systems, or induce rules by them selves relying only on a raw data and experts' diagnosis – that is how classifiers work.
In presented example a *k*–NN classifier is used to determine a liver fibrosis stage using blood test results. Due to a relatively high number of missing data a decision has been taken to use an ensemble of single-parameter *k*–NN classifiers rather than a single, multi-parameter *k*–NN classifier. An ensemble classifier is a proven in literature [2], [3], [4] solution, but an ensemble of single parameter classifiers has a very important, yet not very emphasized elsewhere, feature – it solves the problem of missing data with no need for additional data processing.
The *k*–NN classifier has also been used for the feature elimination stage, in which the accuracy of classifiers trained on every single parameter available has been evaluated and thus parameters best differentiating the classes are chosen.

[1] University of Silesia, Institute of Computer Science, 41-200 Sosnowiec, Będzińska 39, Poland
{tomasz.orczyk, piotr.porwik, marcin.bernas}@us.edu.pl

2. DATA CHARACTERISTICS

Data used in this experiment comes from the Gastroenterology and Hepatology Department of the Independent Public Central Hospital of the Silesian Medical University and contains medical records of 290 patients infected with a hepatitis virus type C. These records consist of patients' age, routine blood test results and a liver biopsy result. Age and blood test results are used as a descriptive parameters for training a classifier ensemble, while the liver biopsy result in a form of the METAVIR Score [5] is used to create the class label field. Data characteristics are shown in Table 1.

Table 1. Data characteristics.

No (k)	Parameter [unit]	Mean	Std. deviation	Missing values
1	Age [years]	57.4	14.15	0%
2	Hemoglobin $[g/l]$	14.6	1.71	58%
3	RBC $[10^6/\mu l]$	4.8	0.62	58%
4	WBC $[10^3/\mu l]$	6.1	1.90	0%
5	PLT $[10^3/\mu l]$	197.1	59.50	0%
6	PT $[sec.]$	12.0	4.70	27%
7	PTP [%]	99.6	15.75	3%
8	APTT $[sec.]$	33.5	5.59	42%
9	INR	1.0	0.11	12%
10	ASPT $[IU/l]$	63.8	48.54	1%
11	ALAT $[IU/l]$	82.5	64.26	0%
12	ALP $[IU/l]$	80.3	29.99	5%
13	Bilirubin $[mg/dl]$	1.0	0.64	6%
14	GGT $[IU/l]$	70.9	66.15	3%
15	Creatinine $[mg/dl]$	1.0	0.35	60%
16	Glucose $[mg/dl]$	96.4	19.83	62%
17	Na $[mmol/l]$	138.3	3.10	63%
18	K $[mmol/l]$	4.3	0.46	63%
19	Cholesterol $[mg/dl]$	187.0	38.71	20%
20	Total Protein $[g/dl]$	7.5	0.64	16%
21	Albumins $[g/dl]$	0.5	0.25	29%
22	Albumins [%]	60.9	5.92	23%
23	$\alpha 1$ Globulins [%]	2.7	0.87	24%
24	$\alpha 2$ Globulins [%]	9.2	1.53	24%
25	β Globulins [%]	10.6	1.70	24%
26	γ Globulins [%]	16.4	5.09	23%

RBC-Red blood cells; WBC-White blood cells; PTL-Platelets; PT-prothrombin time; PTP-prothrombin ratio; APTT-activated partial thromboplastin time; INR-international normalised ratio; AST-aspartate aminotransferase; ALT-alanine aminotransferase; ALP-alkaline phosphatse; GGT-γ-glutamyltransferase; Na-natrium; K-kalium.

Biopsy result has originally been described according to a METAVIR scoring system, but due to relatively high uncertainty of liver biopsy (according to a different sources from 25% up to even 33% [6], [7]), after a medical consultations, the number of classes have been limited to 3, as described in Table 2. Class names depict severity of the patient's condition: L=low, M=medium, H=high).

Table 2. Classes assignment.

Class	METAVIR Score	Count (%)
L	F0, F1	129 (44.5)
M	F2, F3	102 (35.2)
H	F4	59 (20.3)
TOTAL		290 (100.0)

3. ALGORITHM DESCRIPTION

3.1. FEATURE SELECTION

Prior to the classification a feature selection stage has been introduced. It is a well know fact, that elimination of some of the features before the classification stage may gain the classification accuracy [8], [9]. In the first step all parameters that have more than 33% of missing values are rejected. The remaining parameters are used to build individual *k*–NN based classifiers (where *k*=3), which are cross validated using 10–fold CV [10]. The resulting overall classification accuracy is then used to filter-out weak classifiers. The filtering level has been experimentally set up to 45%. Remaining parameters (shown in Table 3) are used to build an ensemble of individual classifiers, described in the next section.

Table 3. Selected parameters.

Parameter name	Individual Accuracy (%)	Missing values (%)
Age	58%	0.3%
Albumins	53%	23.4%
ASPT	52%	1.4%
PLT	50%	0.3%
ALAT	48%	0.3%
GGTP	48%	2.8%
INR	48%	11.7%
γ Globulins	47%	23.4%
Cholesterol	46%	20.3%
$\alpha 1$ Globulins	45%	24.1%

3.2. ENSEMBLE OF CLASSIFIERS

Ensemble of classifiers proposed in this paper is build using classifiers of the same type (*k*–NN), but with classification models build using different attributes. In the proposed solution classification model for each classifier is build using only one attribute and a decision class, thus for each attribute a separate classification model is created. Fig. 1 shows the general idea behind this method – in this example, the training vector consists of 3 attributes (A, B, C) and a class label field. If, in training data, a parameter was missing for a given record, then obviously this record is omitted while training a classifier corresponding to this parameter, but other (non-missing) parameters from this record will still be used to train other classifiers in the ensemble.
All classifiers work in parallel and make their decision individually. For each testing vector, classifiers trained on missing-in-this-vector parameters are omitted and thus the problem of missing data is solved with no additional effort.

3.3. COMBINATION RULE FOR CLASSIFIER RESPONSES

Each of classifiers from the ensemble returns so called support values which corresponds to the probability that classified sample belongs to a given class – we will call them partial decisions. So, for a given dataset, each classifier returns 3 values in range from 0 to 1 corresponding to the 3 classes in the database. The higher value is, the decision has been taken with a higher confidence level. The sum of values for all classes for one case is always 1. As mentioned before, when there is a missing value of some attribute in a test case, then

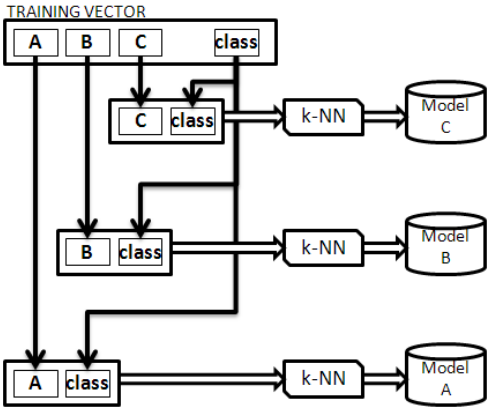

Fig. 1. Tranining of classifiers ensemble.

corresponding classifier does not participate in voting, and it simply returns a probability value equal to 0 for all classes – in other words it doesn't return any partial decision. Assuming that there are p attributes (and classifiers) and n classes, aggregated decision is achieved by summarizing values returned by individual classifiers for each class (1) and choosing a class that has the greatest summarized value (2).

$$V(j) = \sum_{i=1}^{p} v^i(j), \qquad j = 1, \ldots, n \tag{1}$$

$$N = \arg\max_{j} \{V(j) : j = 1, \ldots, n\} \tag{2}$$

Where:
$v^i(j)$ – support value of j–th class coming from i–th classifier.
$V(j)$ – summarized support value of j–th class
N – identifier of the winning class (aggregated decision)

This can be understand as a combination rule in a form of a weighted majority voting. The general idea is illustrated on Fig. 2, where data sample with one missing attribute (B) is being classified. Same scheme is used for model validation, but in that case the correct class label is known, and at the end it is compared with the aggregated decision of the ensemble classifier.

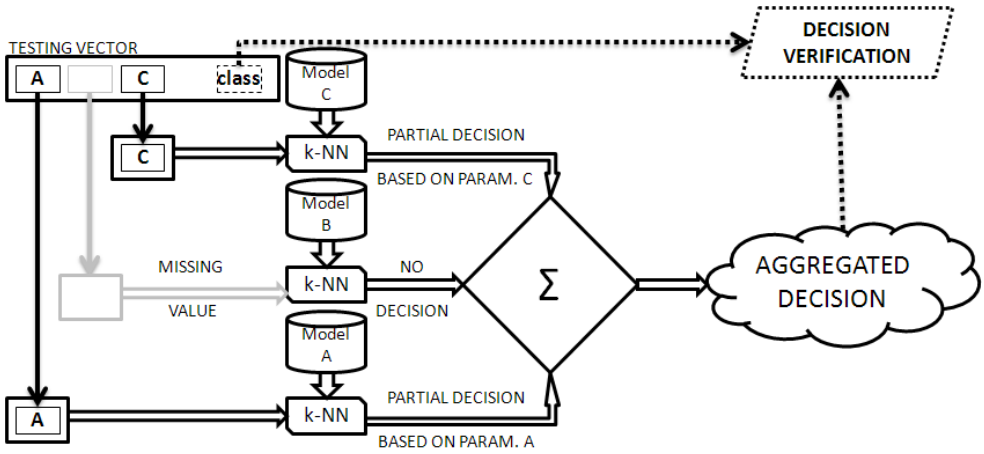

Fig. 2. Verification of classifiers ensemble (with missing attribute B).

4. EXPERIMENT SETUP

The main goal of the experiment was to test the accuracy of the presented algorithm, rather than test the complete classifier in real life environment. In order to achieve this, and due to limited amount of data, a leave one out cross validation has been used. So there were 290 ensemble classifiers, trained on 289 rows of data, and validated using a single row of data.
For the purpose of evaluating obtained results additional experiments have been executed to determine accuracy of proposed ensemble in comparison with:

- a similar ensemble using backward feature elimination [11],
- a similar ensemble using all available features,
- a single classifier (*k*–NN, Random Forest [12], Naive Bayes [13], C4.5 [14]) trained on a subset of parameters selected using proposed feature selection method,
- a single classifier (*k*–NN, Random Forest, Naive Bayes, C4.5) without any form of feature selection/elimination.

5. RESULTS

All these tests (Table 4) were performed on the same dataset and using the same cross validation method. For the backward feature elimination two results were presented: the one with the best accuracy, and the one with the same number of features as in the proposed solution. It is also worth to remember that the Random Forest classifier itself is an ensemble of a Random Trees, so in this experiment it was limited to respectively 26 and 10 trees (as there were 26 features in the dataset and 10 selected features after the feature selection) and each tree has been limited to a single feature. Removing these limitations could allow the Random Forest classifier to outperform the presented classifier, but due to random element in the algorithm its results are unstable and are changing from test to test.

Table 4. Obtained results.

Classifier name	Overall Accuracy (%)	Specifity (%)			Sensivity (%)		
		L	M	H	L	M	H
Proposed method (10 params.)	**67.6**	**68.9**	**83.5**	**94.4**	82.9	61.8	44.1
k–NN ens. w/ backward feature elimination (10 params.)	60.7	66.5	82.4	88.3	71.3	58.8	40.7
k–NN ens. w/ backward feature elimination (15 params.)	65.5	62.1	85.1	95.2	80.6	57.8	45.8
k–NN ens. w/o feature elimination	61.7	50.3	86.7	97.4	85.3	40.2	47.5
k–NN w/ proposed feature selection	54.1	71.4	74.5	83.1	63.6	47.1	45.8
Random Forest w/ proposed feature selection	66.9	68.9	82.4	94.4	**77.5**	**52.9**	**67.8**
Naive Bayes w/ proposed feature selection	61.4	57.1	84.0	94.4	83.7	41.2	47.5
C4.5 w/ proposed feature selection	63.4	68.9	81.4	90.9	74.4	50.0	62.7
k–NN w/o feature elimination	49.7	43.5	83.5	89.6	81.4	17.6	35.6
Random Forest w/o feature elimination	65.9	66.5	77.7	98.7	79.8	56.9	50.8
Naive Bayes w/o feature elimination	58.3	75.8	62.2	95.2	60.5	60.8	49.2
C4.5 w/o feature elimination	63.8	74.5	77.7	90.5	72.1	54.9	61.0

In this comparison proposed method has best overall accuracy, best average specifity and second best average sensivity in all classes.

Method performs well for data with unbalanced classes. What can be noticed from the

Table 6. Confusion table for proposed method.

Actual class \ Classified as	L	M	H
L	107	15	7
M	33	63	6
H	17	16	26

confusion table (Table 6), that there were 66 cases (which is 70% of misclassified cases) classified as more severe than they actually were, while only 28 cases (30%) were classified as a less severe than they were (according to biopsy).
Also the proposed feature selection method has significantly improved classification accuracy for 3 out of 4 tested classifiers in comparison to classification using all available features.

6. CONCLUSIONS

Conducted research have proven usefulness of the presented method in the classification of unprocessed medical data with missing values. Method itself is very simple to implement and may work with any classification algorithm. Preferably this algorithm should return a so called support values, which are probabilities that analysed sample belongs to a given class. In future it is possible to implement more sophisticated combination rules which could further improve this method. It may use more complex voting algorithm (like Copeland voting [15]) or a neural network [16] using output data of the classifiers from the ensemble (so called neural fuser).

ACKNOWLEDGEMENT

This work was supported by the Polish National Science Centre under the grant no. DEC-2013/09/B/ST6/02264.

BIBLIOGRAPHY

[1] AHA D., KIBLER D., Instance-based learning algorithms, Machine Learning, 1991, Vol. 6, pp. 37–66.
[2] WOZNIAK M., ZMYSLONY M., Combining classifiers using trained fuser—Analytical and experimental results. Neural Network World, 2010, Vol. 20(7), pp. 925–934.
[3] WOZNIAK M., KRAWCZYK B., Combined classifier based on feature space partitioning. International Journal of Applied Mathematics and Computer Science, 2012, Vol. 22(4), pp. 855–866.
[4] DOROZ R., PORWIK P., WROBEL K., Signature Recognition Based on Voting Schemes, In Biometrics and Kansei Engineering (ICBAKE), 2013, pp. 53–57.
[5] BEDOSSA P., POYNARD T., An algorithm for the grading of activity in chronic hepatitis c. The metavir cooperative study group, Hepatology, 1996, Vol. 24, pp. 289–293.
[6] REGEV A., BERHO M., JEFFERS L., MILIKOWSKI C., MOLINA E., PYRSOPOULOS N., FENG Z., REDDY Z., SCHIFF E., Sampling error and intraobserver variation in liver biopsy in patients with chronic HCV infection, American Journal of Gastroenterology, 2002, Vol. 97(10), pp. 2614-–2618.
[7] BEDOSSA P., DARGERE D., PARADIS V., Sampling variability of liver fibrosis in chronic hepatitis c, Hepatology, 2003, Vol. 38, pp. 1449-–1457.
[8] DOROZ R., PORWIK P., Handwritten signature recognition with adaptive selection of behavioral features., Communications in Computer and Information Science (CISIM), Springer, 2011, Vol. 245, pp. 128–136.
[9] PORWIK P., DOROZ R., Self-adaptive biometric classifier working on the reduced dataset, Hybrid Artificial Intelligence Systems, Lecture Notes in Computer Science, Springer International Publishing, 2014, Vol. 8480, pp. 377–388.
[10] STONE M., Cross-validatory choice and assessment of statistical predictions, J. Royal Stat. Soc., 1974, Vol. 36(2), pp. 111—147.
[11] KARNAN M., KALYANI P., Attribute reduction using backward elimination algorithm, IEEE International Conference on Computational Intelligence and Computing Research (ICCIC), 2010, pp. 1–4.
[12] BREIMAN L., Random Forests, Machine Learning, 2001, Vol. 45(1), pp. 5–32.

[13] JOHN G. H., LANGLEY P., Estimating Continuous Distributions in Bayesian Classifiers, Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, 1995, pp. 338–345.
[14] QUINLAN R., C4.5: Programs for Machine Learning, Morgan Kaufmann Publishers, San Mateo, CA, 1993.
[15] COPELAND A. H., A 'reasonable' social welfare function, Seminar on Mathematics in Social Sciences, University of Michigan, 1951.
[16] ROSENBLATT F., The Perceptron: A Probalistic Model For Information Storage And Organization In The Brain, Psychological Review, 1958, Vol. 65(6), pp. 386-–408.