SIGNIFICANCE LEVEL BASED CLASSIFICATION WITH MULTIPLE TREES

INFORMATICA 1/91

Keywords: machine learming, artificial intelligence, level, ASSISTANT.

Aram Karalič, Vlado Pirnat
Jožef Stefan Institute, Ljubljana

ABSTRACT: Usually, algorithms for machine learning during the classification return a single class for a given object. Many of the systems do not estimate a reliability of their answer. In the article a method is presented that returns multiple classes as possible. The method also gives the user an estimation of the answer's reliability. Additionally, the method enables also classification in domains where one example can belong to more than one class. The described ideas are tested on a real medical domain — rheumatology. The results are compared with the results of the classical algorithms for machine learning and with the results of general practitioners.

POVZETEK: KLASIFIKACIJA Z VEČ DREVESI BAZIRANA NA STOPNJI ZAUPANJA. Algoritmi za avtomatsko učenje ponavadi pri klasifikaciji neznanega primera podajo samo en razred. Mnogo sistemov ne oceni zanesljivosti svojega odgovora. V članku je podana metoda, ki poda več razredov kot možne. Metoda poda tudi zanesljivost svojega odgovora. Dodatna prednost opisanega pristopa je, da omogoča učenje tudi v domenah, kjer en učni primer lahko pripada več razredom hkrati. Opisane ideje so preverjene na realni medicinski domeni — revmatologiji. Podana je tudi primerjava rezultatov metode z rezultati splošnih zdravnikov.

1 INTRODUCTION

The task of machine learning from examples is usually defined as follows:

> Given: A set of learning examples, described in terms of attributes and their values. Every example belongs to one class. Attributes have symbolic values (discrete attributes) or real values (continuous attributes).
>
> Find: A decision rule that fits the learning set and maps every (previously unseen) example into probability distribution:
>
> $$y = (p_1, p_2, \ldots, p_n) \quad (1)$$
>
> where component p_i of the vector $y \in \Re^n$ is an estimation of the probability that the example belongs to class C_i.

An algorithm for machine learning gives an estimation of the probability distribution (1) over classes. It also assumes that one example belongs to exactly one class.

A decision rule usually consists of a *knowledge base* and a *classification algorithm*. The knowledge base is constructed by a *learning algorithm* in a process of learning. The classification algorithm *uses* the knowledge base to obtain a class-probability distribution of an unseen example. Thus, with different classification algorithms the same knowledge base can be interpreted in several ways.

In this article only the algorithms that construct knowledge base in the form of decision trees (e.g. ASSISTANT [5,3,7], ID3 [8], etc.) are discussed. Nevertheless, the described ideas can be generalized also to algorithms with different knowledge representations.

The two situations that can often occur in the context of machine learning in real-world domains and that are not treated properly by the existing learning systems are the following:

1. It is often the case that class probabilities have to be estimated from relatively small number of examples. Let us consider the case with 3 classes. It is possible that when classifying an unseen example the algorithm classifies it in the leaf which corresponds to three learning examples, all belonging to class C2. The system's answer is: (0.00 1.00 0.00), saying that our new example belongs to class C2 with probability 1. But it is obvious that this probability estimate is extremely inaccurate, because it was calculated only from three learning examples. It was also stated in [2] that the probability estimation is one of the crucial tasks in certain subareas of machine learning.
2. In many of the problems suitable for the application of machine learning example can belong to more than one class. In medicine, for example, a patient can have more than one disease simultaneously. In such domains systems act with incorrect assumption, that the classes are disjoint, which can decrease the performance of the system.

To solve the above situations we propose the following:

1. We believe that it would be more convenient for a user if a system, in the case of inability to give an accurate answer, explicitly answers "I don't know" instead of giving inaccurate information. An appropriate measure of the answer's reliability is necessary to accomplish this task.
2. In domains where classes are not necessary disjoint it is reasonable to drop the assumption of disjoint classes. This means that the relation

$$p_1 + \ldots + p_n = 1$$

no longer holds. The system should rather return the probability for each class independently.

In this article a method called MULTI-α is described. It was designed so as to overcome the above mentioned weak points. It is described in section 2. Section 3 describes experiments in the real-life medical domain — rheumatology. The results of computer-based diagnosing are compared with those achieved by general practitioners. The testing of performance of physicians is also described in section 3. In section 4 advantages of MULTI-α method are summarized and some ideas presented for its practical use.

2 DESCRIPTION OF MULTI-α METHOD

2.1 Learning Algorithm

The basic idea of our approach is to generate a decision tree (decision rule) with classes "C_i" and "$\neg C_i$" for each class C_i separately. That is how the assumption about disjoint classes can be avoided. In fact, the problem is divided into n (n is the number of classes) subproblems. Knowledge about the domain now consists of n decision trees $T_1, \ldots, T_n$. Every decision tree tells something about one of the classes. Similar approaches to building a knowledge base are described in [1,6]. Our approach differs from the above-mentioned ones in a way of how it classifies unseen examples.

2.2 Classification Algorithm

A version of learning algorithm should be used, which produces trees, whose leaves (besides probability distribution) also include the number of learning examples. When classifying example E each class C_i is marked with $\oplus$, $\ominus$ or $\odot$. The meaning of this labels is the following:

- $\oplus$ — "It is possible that the example belongs to this class.",
- $\ominus$ — "It is not possible that the example belongs to this class.",
- $\odot$ — "Nothing can be accurately said about this class.".

This three claims are made with certain degrees of reliability. The whole answer is composed of answers of individual trees, and looks as follows:

$$(x_1, x_2, \ldots, x_n), \alpha$$

Where n is number of classes, $x_n \in \{\oplus, \ominus, \odot\}$ and α is level of significance, described later. Such an answer enables a user to judge the value of the system's answer, which is especially important in "soft" domains like, for example, medicine.

The assignment of $\oplus$, $\ominus$ or $\odot$ to class C_i is made by the following algorithm:

- Classify example with a tree T_i.
- In a leaf, in which an example E is classified, let k be the number of examples belonging to class "C_i" and let the number of all examples in the leaf be N. The relative frequency R_i of the class C_i is $R_i = k/N$ and the system's answer is:

$$(N|R_i, 1 - R_i)$$

- If $R_i > 0.5$ we suspect that the example belongs to class C_i. Let p be the actual probability of class C_i in a given population. The hypothesis $H_\ominus$:"$p < 0.5$" is formulated, saying: "probability of the example belonging to the class is less than 0.5". Interpretation of $H_\ominus$ could be: "the example probably doesn't belong to the class". Now, we try to reject the hypothesis. If we succeed in rejecting it we can, with a certain level of significance, believe that the example belongs to the class. In this case we mark the class with $\oplus$. If we can not reject the hypothesis we believe that we can not make any statistically significant claim about the example belonging to the class. We inform the user of this by marking the class with a $\odot$.
- If $R_i < 0.5$ we formulate the hypothesis $H_\oplus$:"$p > 0.5$" (meaning: "example belongs to the class") and try to reject it. If we succeed it means that we can (with certain level of significance) claim that the example does not belong to the class, so we mark the class with $\ominus$. If we can not reject the hypothesis we can not make any statistically significant claim about the example belonging to the class and inform the user of this by marking the class with a $\odot$
- If $R_i = 0.5$ the class is simply marked with $\odot$.

Let us now explain the statistical test that is based on the principles of statistical tests explained in [4]. The test is explained only for the case when $R_i > 0.5$. The other situation ($R_i < 0.5$) is handled analogously.

Recall that $R_i = k/N$. The leaf can be seen as a series of N experiments, where event C_i occured k times (C_i = "example belongs to class C_i"). We formulate a hypothesis about the probability of event C_i:

$$H_\ominus : p(C_i) < 0.5$$

and try to reject it.

Let R be a random variable denoting a relative frequency of event C_i in a series of experiments (R_i is its value in our series of experiments). The hypothesis can be rejected with risk α when:

$$p(R \geq R_i / H_\ominus) < \alpha$$

Where "/" indicates conditional probability. Let us calculate the value of $p(R \geq R_i/H_\ominus)$:

$$\begin{aligned} p(R \geq R_i/H_\ominus) &= p(R \geq R_i/(p < 0.5)) \\ &= \sum_{i=k}^{N} \binom{N}{i} p^i (1-p)^{N-i} \end{aligned}$$

The actual probability p is not known. But we know, that it is between 0 and 0.5. Considering that

$$(0 \leq p < 0.5) \wedge (N/2 < i \leq N) \Rightarrow$$

$$\begin{aligned} p^i(1-p)^{N-i} &= p^{2i-N}p^{N-i}(1-p)^{N-i} \\ &= [p(1-p)]^{N-i}p^{2i-N} \\ &\leq [1/4]^{N-i}p^{2i-N} \\ &< (1/2)^{2N-2i}(1/2)^{2i-N} \\ &= (1/2)^N \end{aligned}$$

$p(R \geq R_i/H_\Theta)$ can be estimated in the following way:

$$\begin{aligned} \sum_{i=k}^{N}\binom{N}{i}p^i(1-p)^{N-i} &< \sum_{i=k}^{N}\binom{N}{i}0.5^N \\ &= \sum_{i=k}^{N}\binom{N}{i}0.5^i(1-0.5)^{N-i} \\ &= p(R \geq R_i/(p=0.5)) \end{aligned}$$

The last expression can be easily calculated as follows:

$$p(R \geq R_i/(p=0.5)) = 2^{-N}\sum_{i=k}^{N}\binom{N}{i}$$

So, we computed the upper bound for $p(R \geq R_i/H_\Theta)$. Let's denote it with $p'(R \geq R_i/H_\Theta)$:

$$p(R \geq R_i/H_\Theta) < p'(R \geq R_i/H_\Theta) = 2^{-N}\sum_{i=k}^{N}\binom{N}{i}$$

If $p'(R \geq R_i/H_\Theta) \leq \alpha$ (which means $p(R \geq R_i/H_\Theta) < \alpha$) then we reject the hypothesis and declare the diagnosis as "possible" (mark it with $\oplus$).

3 EXPERIMENTS

ASSISTANT [5,3,7], a system for inductive learning of decision trees, was used for generating the classification trees. ASSISTANT classifies an example into class C_i with the highest p_i in the corresponding probability distribution (1). An experiment with the described method of classification was performed and the obtained results were compared with the results of the classical use of ASSISTANT. Parameter settings for the ASSISTANT, which are described in [3], are displayed in Table 1. When describing experiments in medicine a term "patient" is often used instead of "example" and the a term "diagnosis" instead of "class".

ALL Instances Selected	
Instances for Testing	: 30 %
Pruning Factor	: 3.0 x
Best Class Threshold	: 100 %
Weight Threshold	: 0 %
Post Pruning	: YES

Table 1: ASSISTANT's learning parameters.

3.1 Domain Description And Experimental Data

The data for the patients were collected at the Rheumatological Clinic of the University Clinical Center in Ljubljana. If a diagnosis after the first examination of a patient was unclear the patient was re-examined many times during one year to obtain the reliable diagnosis.

Experiments were performed on four different diagnostic problems. The first (and the easiest) problem is to classify a patient into one of the three possible diagnoses. The other three problems have six, eight and twelve possible diagnoses. The percentage of majority class for each problem is given in Table 2. Each example is described in terms of 16 anamnestical attributes, 37 clinical, 4 laboratory and 1 radiological attribute.

number of diagnoses	percentage of majority class
3	66.45
6	61.90
8	34.20
12	34.20

Table 2: Percentage of majority class for each diagnostic problem.

The data for 462 patients were collected. Data for anamnestical and clinical attributes were missing only for 10 attribute values. Laboratory data were missing in 44 cases and radiological in 211 cases.

Experiments were made for all four diagnostic problems (classification into one of 3, 6, 8 and 12 diagnostic groups). For every problem one tree was built for each diagnosis. Learning was performed on 70% of the patients, the rest (30%) was used for testing. Each experiment was repeated 10 times, so the final results are averages of 10 runs.

3.2 Testing Of MULTI-α Method

On the basis of the results of the statistical tests a set of possible diagnoses (SPD) was constructed. An answer of the system was defined as correct if SPD included the patient's diagnosis. SPD was constructed using three different strategies, called A, B and C.

When using strategy A all diagnoses marked with $\oplus$ or $\odot$ were included into SPD. When using strategy B diagnoses marked with $\oplus$ were included into SPD and when using strategy C only the diagnose marked with $\oplus$ whose hypothesis was rejected with the smallest α was included into SPD (this corresponds to the most probable diagnosis). In all cases the SPD was checked for emptiness. If it was found empty, the "most probable" diagnose was added to the SPD.

Strategy A typically produces the largest SPD, resulting in the highest percentage of correct answers. Strategy C is the most similar to the classic use of the ASSISTANT (select the most probable diagnosis). They differ only when several diagnoses were assigned the same risk factor. The SPD then contains more than one element. Strategy B is a compromise between the strategies A and C. If there are many examples in the domain that belong to more than one class, it is wiser to use strategy A; otherwise one should use strategies B or C.

We refer to strategies A, B and C also as MULTI/A, MULTI/B and MULTI/C respectively and to all three of them collectively as MULTI.

Three different levels of significance were used: 1%, 5% and 10%, but the results differed only slightly so only results for $\alpha = 5\%$ are presented in the article.

When comparing the results of ASSISTANT and MULTI-α, a problem can arise due to a fact that ASSISTENT always classifies in one class. The measure of success of ASSISTANT classifi-

cation is classification accuracy, which is defined with an assumption, that the system always suggests only one diagnosis. But in MULTI-α classification the system can suggest more diagnoses to be the possible ones. So, the classification accuracy as defined in ASSISTANT classification does not exist. Therefore one can not make any exact comparison (e.g. with t-test), between the results. Nevertheless, results of ASSISTANT classification can be compared with the results of C strategy of the MULTI classification (which is the most similar to ASSISTANT classification) just to get the impression about the performance.

To further reduce the dissimilarity between the two measures of performance, each correct answer was weighted with a $1/|SPD|$. This technique is also implemented in original ASSISTANT [5].

In the experiments the *weighted* percentage of correct answers, which we denoted with *acc*, the size of the SPD (denoted with *siz*) and the percentage of cases when |SPD| was more than one (denoted with *MT1*) were measured. The mean values and standard deviations were measured for each parameter. Results of the experiments are presented in Table 3.

strategy		3 diagnoses	6 diagnoses	8 diagnoses	12 diagnoses
A	acc	70.5 ± 2.7	64.8 ± 2.7	48.9 ± 2.0	47.9 ± 2.1
	siz	1.2 ± 0.1	1.2 ± 0.0	1.4 ± 0.1	1.4 ± 0.1
	MT1	20.0 ± 9.4	15.1 ± 4.2	30.1 ± 7.8	29.2 ± 5.3
B	acc	71.7 ± 3.2	65.7 ± 3.6	51.2 ± 3.0	50.85 ± 3.5
	siz	1.1 ± 0.1	1.1 ± 0.0	1.2 ± 0.1	1.2 ± 0.0
	MT1	12.5 ± 6.1	8.9 ± 3.4	17.7 ± 4.0	16.4 ± 2.3
C	acc	71.6 ± 3.4	65.5 ± 3.5	51.4 ± 3.5	51.2 ± 3.7
	siz	1.1 ± 0.1	1.1 ± 0.0	1.2 ± 0.1	1.2 ± 0.0
	MT1	11.8 ± 6.4	7.6 ± 3.2	14.1 ± 3.8	13.8 ± 2.1

Table 3: Multiple tree classification.

The comparison between ASSISTANT and MULTI/C is summarized in Table 4. It can be seen that the weighted number of correct answers using MULTI/C method is typically larger than the number of correct answers using ASSISTANT.

number of diagnoses	ASSISTANT	MULTI/C
3	70.4 ± 3.2	71.6 ± 3.4
6	64.7 ± 3.0	65.5 ± 3.5
8	49.9 ± 3.5	51.4 ± 3.5
12	47.8 ± 3.6	51.2 ± 3.7

Table 4: Comparison between ASSISTANT and MULTI/C method.

The classification accuracy on testing examples was compared with the classification accuracy on learning examples. The results are summarized in Table 5. The Table shows that the classification on learning examples is better than on the testing examples. This indicates that the system did not have enough learning examples to accurately learn decision rules. With increasing number of learning examples we believe that the system could achieve similar results on testing examples as on learning examples.

number of diagnoses	MULTI/C, $\alpha = 0.05$		
	Lrn	Tst	Lrn/Tst
3	78.8 ± 2.2	71.6 ± 3.4	1.10
6	72.9 ± 2.2	65.5 ± 3.5	1.11
8	71.2 ± 2.0	51.4 ± 3.5	1.39
12	69.8 ± 1.9	51.2 ± 3.7	1.36

Table 5: Number of correct answers when classifying learning (Lrn) and testing (Tst) examples.

3.3 Comparing MULTI-α Method With Physicians

Because we wanted to compare the results of our system with the results that can be obtained from general practitioners 10 general practitioners were tested. Their task was to classify 30 randomly chosen patients from our set of patients. For each patient, the practitioners were presented with a description of the patient (attribute values). They had to assign one of the numbers 1, 2, 3, 4, 5 to every possible diagnose. Interpretation of number 1 was that this diagnose was surely incorrect, number 2, that there was a small possibility of this diagnose. Number 3 meant, that the doctor couldn't say anything accurate about the diagnose. Interpretation of number 4 was that this diagnose was very probable and number 5 that this diagnose was certain. Numbers 1 and 2 were mapped to $\ominus$, number 3 to $\odot$ and numbers 4 and 5 to $\oplus$. Patients were chosen so that their class distribution was as close as possible to the distribution of the whole set of patients. Our system classified the same 30 patients with knowledge, learned from the *remaining* 432 patients.

The results of the testing of physicians' performance are summarized in Table 6. Columns marked with *physicians* are the results of the physicians and columns marked with *MUL* are the results of our method with $\alpha = 0.05$. The computer-based classification always outperformed the practitioners.

strategy		3 diagnoses		6 diagnoses		8 diagnoses		12 diagnoses	
		physicians	MUL	physicians	MUL	physicians	MUL	physicians	MUL
A	acc	48.6 ± 14.8	61.7	51.9 ± 13.5	58.8	30.8 ± 7.1	41.7	17.4 ± 6.9	88.8
	siz	1.4 ± 0.4	1.1	2.8 ± 0.8	1.1	2.7 ± 1.0	1.8	3.4 ± 1.4	1.4
	MT1	85.7 ± 28.1	18.8	60.8 ± 84.4	10.0	74.0 ± 26.2	26.7	84.0 ± 19.7	80.0
B	acc	49.7 ± 14.8	61.7	57.9 ± 11.8	58.8	36.1 ± 7.4	41.7	22.7 ± 7.1	88.9
	siz	1.8 ± 0.2	1.0	1.6 ± 0.8	1.1	1.7 ± 0.4	1.8	1.7 ± 0.4	1.4
	MT1	25.0 ± 19.4	8.8	82.7 ± 21.9	6.7	46.0 ± 18.0	26.7	46.0 ± 17.8	80.0
C	acc	49.9 ± 14.8	61.7	58.1 ± 11.9	58.8	36.4 ± 7.2	44.7	28.1 ± 7.6	48.9
	siz	1.8 ± 0.2	1.0	1.8 ± 0.4	1.1	1.8 ± 0.4	1.3	1.4 ± 0.3	1.8
	MT1	24.0 ± 19.2	8.8	26.7 ± 19.8	6.7	36.7 ± 16.9	16.7	20.8 ± 10.0	20.0

Table 6: Results of the testing of physicians' performance.

4 CONCLUSIONS

We developed a method which enables learning in domains where one example can belong to *several classes*. We also improved the weak points of the systems that do not *estimate reliability of their answers*. Additional advantage of the method is that it does not contain any *probability estimates* and therefore avoids all problems arising when estimating probabilities from small samples. Let us at this place point out that parameter α is the *upper bound* for a probability of making an improper decision *not an approximation* of the probability. Only exact statistical assertions are used in our approach. The method was developed on the basis of the system ASSISTANT that expresses the induced knowledge in the form of a decision tree. However, the method can be used by a wide number of today known systems for empirical learning.

Results of the physicians clearly showed that with equal information about the patient general practitioners correctly diagnose less patients than our method. However, in real life physicians have much more information about the patient than just the 58 attributes that were used for our experiments. But in spite of this we think that physicians could benefit from computer-based classification. The discrepancy between system's and physician's classification could serve as a warning to the physician to re-examine the patient and possibly take some additional tests.

To the general practitioner, the most interesting problem is our first diagnostic problem (3 diagnoses). General practitioner examines the patient and sends him to the rheumatologist, to the orthopedist or to another specialist. Some of the general practi-

tioner's decisions are not correct and the patient has to queue for the improper doctor and wait for some time. Meanwhile, a disease can make a considerable progress. The use of an expert system to assist the general practitioner's decisions would decrease the number of such cases which would result in better functioning of medical systems as well as in smaller medical expenses.

ACKNOWLEDGMENTS

We would like to thank all members of Artificial intelligence laboratory at Jožef Stefan Institute, Ljubljana and Faculty of Electrical and Computer engineering, Ljubljana, especially Igor Kononenko and Bojan Cestnik. Thanks to Dunja Mladenić for incorporating extensions to system ASSISTANT, which enabled us to perform experiments. Thanks also to Renata Zupanc and Alen Varšek for careful proofreading.

REFERENCES

[1] Catlett, J., Christopher, J.: *"Is it better to learn each class separately?"*, technical report, University of Sydney, Australia, 1987.

[2] Cestnik, B.: "Estimating Probabilities: A Crucial Task in Machine Learning", *Proceedings of ECAI-90*, Stockholm, Sweden, august 1990.

[3] Cestnik, B., Kononenko, I., Bratko, I.: "ASSISTANT 86: A Knowledge elicitation tool for sophisticated users", in: Bratko, I., Lavrač, N. (eds.): *Progress in Machine Learning*, Sigma Press, Wilmslow, 1987.

[4] Ferguson, G.A., *"Statistical Analysis in Psychology and Education"*, Chapter 11, McGraw-Hill, London, 1959.

[5] Kononenko, I.: *"The design of the ASSISTANT Inductive Learning System"*, M.Sc. Thesis, E.Kardelj University, Faculty for Electrical and Computer Engineering, Ljubljana, 1985 (in Slovene).

[6] Michie D., Al Attar A.: "Use of sequential Bayes with class probability trees", to appear in: J.E.Hayes – Michie, D.Michie, E.Tyugu (eds.), *Machine Intelligence 12*, Oxford University Press.

[7] Mladenić, D.: *"Magnus Assistant: a system for machine learning"*, B.Sc. Thesis, E.Kardelj University, Faculty for Electrical Engineering and Computer Science, Ljubljana, 1990 (in Slovene).

[8] Quinlan, J.R.: "Induction of Decision Trees", *Machine Learning 1*, Kluwer Academic Publishers, Boston, 1986.