Influence of probability estimation parameters on stability of accuracy in prototype rules using heterogeneous distance functions.

prof. dr hab. Tadeusz Wieczorek
mgr inż. Marcin Blachnik
Division of Engineering Informatics, Department of Electrotechnology
Faculty of Materials Enginering and Metallurgy
The Silesian University of Technology
Krasińskiego 8, 40-019 Katowice
Poland
tadeusz.wieczorek@polsl.pl marcin.blachnik@polsl.pl

prof. dr hab. Włodzisław Duch
Department of Informatics
Nicholaus Copernicus University
Grudziądzka 5, 87-100 Toruń
Poland
School of Computer Engineering,
Nanyang Technological University
Singapore
www.phys.uni.totun.pl/~duch

Abstract –

Many different approaches to the problem of classification have been collected. An interesting way to understand data leads to prototype rules (P-rules). In this approach the aim is to find optimal position of prototypes to which we compare unknown vectors.

One of important problems in applications P-rules for real datasets are distance functions operating on different type of attributes like discrete, linear, symbolic, nominal. Solution for such problems are heterogeneous distance functions. This type of functions are usually based on probability distance measure like Value Difference Matrix (VDM), adopted for continues attributes by estimation of probability density function for continues values. The process of estimation requires selection of several parameters, which have important influence on overall classification accuracy.

Accuracy and this impact is investigated in the paper. Various heterogeneous distance function based on VDM measure are presented, among them some new heterogeneous distance functions based on different type of probability estimation. Practical experiments using the described methods and discussion of obtained results are presented.

I. INTRODUCTION

One of the most important aims in artificial intelligence field are classification problems and after so many years of researches this issue is still open. We have collected many different approaches to this aim. One of most

popular methods which try to solve classification problem are artificial neural networks, however their applications are limited, because we don't know how do they work and if there are any weaknesses of their solution we can not find them because they are "black boxes". This is, why we can't use them in some classes of problems like for example an autopilot in airplanes or in medical applications. Much more better algorithms in this field are systems basing on rules, however the question is how to generate a set of rules, which will be reliable, accurate and as small as possible, but not smaller so that we could understand them without losing accuracy [6]. The first idea are statistical methods like decision trees, which are generating rules operating on each attribute separately. The most popular examples are C4.5 [8] Quinlan algorithm, or SSV tree [7].

Another solution are Fuzzy Sets [4],[10] which can be used for rule construction. Another interesting way to understand data leads to prototype rules (P-rules) [5]. How experiment shows they allow to fulfill defined earlier criteria, generating small and easy to understand set of rules characterized by very good accuracy [2]. In this approach the aim is to optimize position of prototypes to which we compare unknown vectors using previously chosen distance function or similarity measure. One of the most frequently type of rules in P-rules are nearest neighbor rules, where we calculate distance between unknown case and all the prototypes and look for nearest prototype, saying that output class is the some as class of closest prototype.

The question is what type of measure shall we use? and of course the simplest answer is Euclidian distance function However in practical applications we find datasets, which have mixed attribute types, some are continues, some are discrete and some are symbolic or nominal, where Euclidian distance function does not work so well, moreover in case of symbolic features obtained result depend on the method of convertion into numeric values. This problem also pay a rule in fuzzy rules where we do not know how to generate so specific type of membership function.

Solution for such problems are heterogeneous distance functions which use different type of measure for different type of attributes joining them together. This type of functions usually basing on probability distance measure like Value Difference Matrix (VDM) [1], adopted for continues attributes by estimation of probability density function for continues values. The process of estimation requires selection of several parameters, which have important influence on overall classification accuracy and this impact is investigated in the paper.

In section II we present different heterogeneous distance function based on VDM measure. Section III presents some new heterogeneous distance functions based on different type of probability estimation. Practical experiment is presented in section IV and in section V we summarize obtained results and draw conclusions.

II. HETEROGENEOUS DISTANCE FUNCTIONS

In most similarity based systems like nearest neighbor, radial bases function networks [9] or self-organizing maps mostly Euclid's, or rather Minkovsky's distance function is used, or other modified functions like Mahalanobis distance function. Unfortunately this group of functions does not support symbolic and nominal features, which we can often find in real applications, although Value Difference Matrix (VDM) [9] gives very good results for symbolic attributes, but using it with continues attributes is impossible. Building an universal similarity system specially, when we are looking for prototype rules, we should consider both types of similarity functions, which are called heterogeneous distance function.

VDM distance measure is based on calculation the differences between posteriori probabilities, that is described by equation (1).

$$VDM(\mathbf{x},\mathbf{y}) = \sum_{a=1}^{m} vdm(x_a, y_a) \quad (1)$$

$$\mathrm{vdm}(x_a, y_a) = \sqrt{\sum_{i=1}^{n} \left(p(C_i \mid x_a) - p(C_i \mid y_a)\right)^2} \quad (2)$$

Where probabilities are worked out by the form (2).

$$p(C_i \mid x_a) = \frac{Nx_{ai}}{Nx_a} \quad (3)$$

Where **X** and **Y** are input vectors, N_a is number of instances in a training set that has got a value of x for the attribute a, N_{ai} is the some as N_a but for class i, n is number of classes and m is number of attributes.

In P-rules we are interested to operate on all types of features so the only solution for such situation are heterogeneous distance functions (HDF). One of the simplest way leading to HDF is combination of Euclid's and VDM matrix called Heterogeneous Value Difference Matrix (HVDM) [3]:

$$\mathrm{HVDM}(\mathbf{x},\mathbf{y}) = \sqrt{\sum_{a=1}^{m} \mathrm{d}_a^2(x_a, y_a)} \quad (4)$$

Where

$$\mathrm{d}_a(x,y) = \begin{cases} 1, & \text{x or y are unknown} \\ \mathrm{n\_vdm}_a(x,y) & \text{a is discrete or nominal} \\ \mathrm{n\_dif}_a(x,y) & \text{a is continues} \end{cases} \quad (5)$$

If we operate on nominal data, $\mathrm{d}_a(x,y)$ assumes form

N1 :

$$\mathrm{n\_vdm}_a(x,y) = \sum_{i=1}^{n} \left| \frac{Nx_{ai}}{Nx_a} - \frac{Ny_{ai}}{Ny_a} \right|$$

N2 :

$$\mathrm{n\_vdm}_a(x,y) = \sqrt{\sum_{i=1}^{n} \left| \frac{Nx_{ai}}{Nx_a} - \frac{Ny_{ai}}{Ny_a} \right|^2} \quad (6)$$

N3 :

$$\mathrm{n\_vdm}_a(x,y) = \sqrt{n \cdot \sum_{i=1}^{n} \left| \frac{Nx_{ai}}{Nx_a} - \frac{Ny_{ai}}{Ny_a} \right|^2}$$

and for continous data

$$n\_vdm_a(x,y) = \frac{|x-y|}{4\sigma_a} \quad (7)$$

where σ is the standard deviation for the attribute a.

Main problem using HVDM is normalization, because it is very difficult to receive a form of the distance matrix which can be compared to obtain correct and optimal results of joined distance value. In this situation three different forms of VDM distance with different normalization technique are used, and the decision which one should be chosen depend on a designer of the system The benefits of HVDM measure is the Euclid's distance (7) used for continues features, however it is normalized by standard deviation to reduce the influence of outliers.

Distance functions, where the problem of normalization does not occur are value difference matrix with posterior probabilities estimated for both discrete and continues features. However, in such case the estimation of probability density for continues features is a big problem. Martinez and Willson in [3] describe Discretized Value Difference Matrix (DVDM) and Interpolated Value Difference Matrix (IVDM).

DVDM is based on discretization process and for continous attributes a simple constant width discretization method is used (9).

DVDM is described by the equation:

$$\mathrm{DVDM}(\mathbf{x},\mathbf{y})^2 = \sum_{a=1}^{m} vdm_a\left(disc_a(x_a), disc_a(y_a)\right)^2 \quad (8)$$

Where *disc* is a discretization function defined as:

$$disc_a(x_a) = \begin{cases} \left\lfloor \dfrac{x - \min_a}{w_a} \right\rfloor + 1 & \text{if x is continous} \\ x & \text{if x is discrete} \end{cases} \quad (9)$$

min_a is the minimum of attribute a and w_a is a parameter describing number of ranges. However upper part of equation (9) can be swapped by a different form of discretization algorithm.

IVDM is very similar to DVDM, but to improve shape of posterior probability a simple linear interpolation was used. In this situation IVDM can be described:

$$IVDM(\mathbf{x},\mathbf{y}) = \sum_{a=1}^{m} indm_a(x_a, y_a) \quad (10)$$

$$ivdm_a(x_a, y_a) = \begin{cases} vdm_a(x_a, y_a) & \text{a is discrete} \\ \sum_{i=1}^{n} \left| p_{ai}(x) - p_{ai}(y) \right|^2 & \text{a is continous} \end{cases} \quad (11)$$

where

$$p_{ai}(x) = p_{aiu} + \frac{x - mid_{au}}{mid_{a,u+1} - mid_{a,u}} \cdot \left(p_{ai,u+1} - p_{aiu}\right) \quad (12)$$

Where p_{aiu} and $p_{ai,u+1}$ are posterior probabilities calculated in the middle of the discretized range u and $u+1$, $u=disc(x)$ and mid_{au} and $mid_{a,u+1}$ are middles of discretized ranges u and next $u+1$, for which actual x_a fulfill inequality.

III. NEW HETEROGENEOUS DISTANCE FUNCTIONS

Main problem in taking advantage of VDM distance measure to continues attributes is the way to obtain appropriate shape of posterior probabilities. For discrete or symbolic features it can be easily computed by frequencies with equation (3) but for continues attributes it does not work. Two simple techniques was presented in previous section but better algorithms used for determining posterior probabilities may lead to better overall results. All this new methods are based on equation (11) but with a different density analyze technique.

A. Gaussian value difference matrix

An interesting solution is Gaussian smoothness which is very popular in Bayesian neural networks. In this kind of algorithms posterior probability is calculated as (13)

$$p(C_i \mid x_a) = \left(\sum_{j=1}^{M_i} e^{-\left(\frac{x_{aj}}{\sigma}\right)^2} \right) \cdot norm \quad (13)$$

Where M_i is number of all vectors from the some class i, σ is width of Gauss function and *norm* is normalization factor calculated by the form (14)

$$norm = \frac{1}{\sum_{k=1}^{n} \left(\sum_{j=1}^{M_i} e^{-\left(\frac{x_{aj}}{\sigma}\right)^2} \right)} \quad (14)$$

B. Local Value Difference Matrix (LVDM)

Very simple and very fast technique estimating probability is the Local Value Difference Matrix (LVDM). This method is based on local calculation of data density surrounding interesting data point for which we are trying to determine probability. In this method probability is calculated by the equation (3), but value of N_{xai} is the number of points in class i of area limited to range

$\left[x_a - \frac{width_a}{2}, x_a + \frac{width_a}{2} \right]$, and N_{xa} is the some as N_{xai} but calculated for all classes. $Width_a$ is a parameter defining range of width for attribute a.

C. Parzen Value Difference Matrix (PVDM)

Another solution for density estimation is based on Parzen Window technique where rectangle window is moved by the step through whole range of attribute a and probability is calculated as a mean value of all window probabilities where x occurs (15).

$$p(C_i \mid x_a) = \frac{1}{Z} \sum_{z=b+1}^{b+Z} \frac{N_{iz}(x_a)}{N_z(x_a)} \quad (15)$$

Where Z is number of windows $Z = \frac{width_a}{step_a}$, b index of first window where x occurs, $N_{iz}(x_a)$ number of data points in z-th window which class is i, $N_z(x_a)$ the some as $N_{iz}(x_a)$ but for all classes, $width_a$ is window width for attribute a, and $step_a$ is size of window movement.

IV. EXPERIMENT AND RESULTS

Experiment was performed in two steps. At the first step we wanted to verify quality of probability estimation, and influence of estimation parameters. In this case two artificial datasets were generated. First one was two dimensional, three class problem where each class was generated with normal distribution, and a the other dataset, was also two dimensional three class problem but data points where generated with uniform distribution. In both datasets classes were overlapping.

In the second step we perform a classification task on real datasets chosen from UCI repository, to verify true abilities of classification and to verify results obtained in the first step. In this approach we selected datasets with different type of attributes: continues, discrete, symbolic and nominal.

All tasks were carried out with a self created SBPS system. SBPS is a similarity based rules generating system, which allows to define different type of distance function for different attributes, in the last step joining obtained results for each feature into one value. This system has build in different type of prototype selection and optimization algorithms which are used to reduce and improve obtained rules. Making results obtained in each task comparable for all of them we used simple Fuzzy C-means algorithm for prototype selection and LVQ algorithm for their optimization.

A. Artificial datasets

How it was previously mentioned, artificial datasets were created to verify quality of probability estimation and meaning of adjustment parameters into final classification results. For the first artificial dataset with normal distributed classes optimal border shape can be obtain with Euclidian distance function. These results determine a basis to judge and compare quality of probability estimation and classification for other functions. In this test only one prototype per class was selected and to reduce influence of randomness and verify generalization ten fold cross validation test was performed. Results presented in tab. 1 show balanced accuracy for each method.

		HVDM	GVDM			LVDM				IVDM		DVDM	
			sig 0.2	sig 0.5	sig 0.7	width 0	width 0	width 0	width 0	CW 10	CW 5	CW 10	CW 5
Dataset 1	Bal. Acc	96,830	95,670	96,500	96,170	95,000	95,330	95,500	95,330	95,17	94,33	96,5	90,5
Dataset 2	Bal. Acc	90,500	88,330	90,670	90,330	86,000	88,170	88,330	89,000	86,83	87,5	85,17	81,33
		PVDM											
		Step 0.1				Step 0.01				Step 0.05			
		W0.2	W0.4	W0.6	W0.7	W0.2	W0.4	W0.6	W0.7	W0.2	W0.4	W0.6	W0.7
Dataset 1	Bal. Acc	94,670	94,830	95,830	96,170	95,000	95,500	96,000	96,500	94,670	94,000	96,000	96,170
Dataset 2	Bal. Acc	86,330	88,170	90,000	90,000	86,670	87,170	88,330	89,000	86,500	87,000	88,830	88,670

B. Real datasets

Each of HDF have been also tested on real datasets to verify theoretical considerations. We have chosen a group of datasets with different types of attributes, from UCI repository: Flag, Glass, Iris, Lancet and Pima Indians. Because our aim was to obtain maximum balanced accuracy for all this distance measures we have used the algorithm for constructive rule generation to maximize classifier abilities.

The constructive algorithm used in our researches do not favor any distance function because it adds new prototype to class with lowest accuracy, maximizing overall balanced accuracy calculated as a mean value of individual accuracies. In all cases constructive algorithm was stopped after 10 iterations, so maximum we could get 10 prototypes per class.

Because of problem of normalization different distance functions, all continues features in all datasets were previously standardized and then normalized to the interval [0,1]. Obtained results – highest balanced accuracy for each combination of parameters - are presented in Table 2

V. RESULTS DISCUSSION AND CONCLUSIONS

Theorem "No free lunch" says that gold algorithm for data analyzing and optimization does not exist and obtain results have proofed it. However we can see that for artificial data, the GVDM distance function is better than other methods, moreover for second artificial data set obtained accuracy was higher then obtained with Euclidian distance. It was predictable that this algorithm should give very good results because for such data distribution with so high density this method generate smoothest shape of estimated probability, but selection of appropriate values is very significant.

As we can see on real datasets important problem with HVDM is adjustment of parameters values. Choosing correct value is now much more important and selection of the best method is not so easy, even impossible. Marked as bold highest accuracies appear in different methods for each dataset, but what is interesting now

GVDM distance do not work so well, sometimes leading to spread results. Obtained results unfortunately do not lead us to any strict conclusion about what type of distance shall we use or which values are the best. If some values of estimation parameters are wrongly chosen it may appears as very jagged contour of probability, then we say about overfitting, or it may lead to lose an important information about data, what is also undesirable.

	flag	**glass**	**iris**	**lancet**	**pima**
HVDM	Bal. Acc	Bal. Acc	Bal. Acc	Bal. Acc	Bal. Acc
	18,958	37,772	96,000	90,228	**73,740**
GVDM					
sig 0.2	23,229	48,948	96,000	89,994	71,815
sig 0.5	**30,208**	55,367	96,667	89,777	71,401
sig 0.7	28,438	46,865	96,667	89,777	71,386
mean	27,292	50,394	96,444	89,849	71,534
std	3,628	4,431	0,385	0,126	0,244
LVDM					
width 0.2	25,625	47,778	96,000	90,103	72,886
width 0.4	27,708	44,147	96,667	89,994	72,049
width 0.6	26,563	48,978	95,333	89,994	71,490
width 0.7	26,875	42,054	94,000	89,777	71,676
mean	26,693	45,739	95,500	89,967	72,025
std	0,861	3,202	1,139	0,137	0,619
PVDM					
W0.2 St0.1	30,104	39,722	96,667	90,103	71,613
W0.4 St0.1	26,563	42,639	96,667	89,994	71,504
W0.6 St0.1	24,375	49,702	95,333	89,777	70,531
W0.7 St0.1	27,396	49,206	96,667	89,876	71,034
W0.2 St0.01	29,479	46,359	96,000	90,005	71,820
W0.4 St0.01	25,625	45,694	96,000	89,994	71,468
W0.6 St0.01	24,375	**58,046**	96,667	89,777	71,234
W0.7 St0.01	27,083	48,075	96,667	89,777	71,041
W0.2 St0.05	28,542	46,319	96,000	90,103	71,386
W0.4 St0.05	26,250	44,345	96,000	89,994	71,482
W0.6 St0.05	24,375	56,141	96,000	89,777	70,970
W0.7St0.05	27,813	56,379	96,667	89,777	71,555
mean	26,832	48,552	96,278	89,913	71,303
std	1,953	5,717	0,446	0,133	0,355
IVDM					
CW 10	26,563	46,984	96,000	90,225	70,818
CW 5	26,042	48,651	96,667	90,117	72,375
mean	26,302	47,817	96,333	90,171	71,597
std	0,368	1,179	0,471	0,077	1,101
DVDM					
CW 10	26,979	43,810	**97,333**	90,325	71,081
CW 5	27,083	50,635	94,667	**90,330**	70,142
mean	27,031	47,222	96,000	90,327	70,612
std	0,074	4,826	1,886	0,003	0,664

Calculations of some datasets show that even simplest DVDM measure may give good results. This situation occurs when a gap between different classes is very small, so any advanced techniques usually lead to increase number of faults, especially it is important in datasets with low number of training vectors.

Interesting extension of described here methods may be replacement VDM matrix with different probability distance matrix like minimum risk matrix (MRM) or Short and Fukunga marix (SFM) [7]. Also other smoothness techniques should be analyzed and compared together, a specially different more advanced and supervised discretization algorithms should lead to increase accurancy. This group of methods will be analyzed in the next step of our work and we hope that obtain results will be also interesting.

VI. REFERENCES

[1] E. Blanzieri, F. Ricci "Probability Based Metrics for Nearest Neighbor Classification and Case Based Reasoning", Proceedings of the third International Conference on Case-Based Reasoning, Munich, August 1999.

[2] W. Duch M. Blachnik "Fuzzy rule-based system derived from similarity to prototypes", Neural Information Processing, Lecture Notes in Computer Science vol. 3316, Springer, 2004, pp 912-917

[3] D. Randall Wilson, T R. Martinez "Improved Heterogeneous Distance Function", Jurnal of Artificial Inteligence Research 6, 1997, pp. 1-34

[4] A. Piegat "Modelowanie I sterowanie rozmyte" AOW Exit, Warszawa 2003

[5] W. Duch, K. Grudziński "Prototype based rules - a new way to understand the data" IJCNN 2001, Washington D.C. USA

[6] W. Duch, R Setiono, J Żurada "Computational intelligence methods for rule-based data understanding" Proceedings of the IEEE, Vol 92/5, 2004

[7] K. Grąbczewski, W. Duch "The separability of split value criterion" 5'th Conference Neural Network and Soft Computing, Zakopane 2000

[8] M. Kłopotek "Inteligentne wyszuiwarki internetowe" AOW Exit, Warszawa 2001

[9] N. Janowski „Ontogeniczne sieci neuronowe, o sieciach zmieniających swoją strukturę" AOW EXIT, Warszawa 2003

[10] A. Łacha „Rozmyty świat zbiorów, liczb, relacji, faktów reguł i decyzji" AOW Exit, Warszawa 2001