Proposal for a Unified Methodology for Evaluating Supervised and Non-supervised Classification Algorithms

Salvador Godoy-Calderón [1], J. Fco. Martínez-Trinidad [2], and Manuel Lazo Cortés [3]

[1] Center for Computing Research, IPN, Mexico
sgodoyc@prodigy.net.mx
[2] Computer Science Department, INAOE, Puebla, Mexico
fmartine@inaoep.mx
[3] Pattern Recognition Group, ICIMAF, Havana, Cuba
mlazo@icmf.inf.cu

Abstract. There is presently no unified methodology that allows the evaluation of supervised and non-supervised classification algorithms. Supervised problems are evaluated through *Quality Functions* that require a previously known solution for the problem, while non-supervised problems are evaluated through several *Structural Indexes* that do not evaluate the classification algorithm by using the same pattern similarity criteria embedded in the classification algorithm. In both cases, a lot of useful information remains hidden or is not considered by the evaluation method, such as the quality of the supervision sample or the structural change generated by the classification algorithm on the sample. This paper proposes a unified methodology to evaluate classification problems of both kinds, that offers the possibility of making comparative evaluations and yields a larger amount of information to the evaluator about the quality of the initial sample, when it exists, and regarding the change produced by the classification algorithm.

1 Introduction

When one works in pattern recognition, whether in field applications or in research, it is a common need to evaluate the result of a classification algorithm [1-4]. On many occasions the objective of such evaluation is, either to study the behavior of the classification algorithm used, or to establish the appropriateness of applying such algorithm to the type of problem being evaluated. Classification problems may be shown in three different ways [5] known as s*upervised* problems, p*artially-supervised* problems and *non-supervised* problems. Unfortunately, nowadays there is no methodology that allows us to evaluate, under the same criteria, the action of an algorithm in any of the forms of a problem.

A classification problem is informally called supervised when there is previous knowledge (called supervision sample or learning information) on the classes or categories into which it is possible to classify the objects or patterns being studied.

A classification problem is considered non-supervised when such previous knowledge does not exist. In that case, the problem starts with a universe of patterns without structure that must be classified. Finally, the other form that a classification problem

J.F. Martínez-Trinidad et al. (Eds.): CIARP 2006, LNCS 4225, pp. 674 – 685, 2006.

can adopt is an intermediate state between supervised and non-supervised problems. A classification problem is considered partially-supervised when the previous knowledge regarding the nature of its solution is partial.

2 Traditional Evaluation Methods

In order to evaluate supervised problems, the classification algorithm is applied to a test sample and its result is compared with a previously known solution considered as valid [3,4]. This comparison is made by means of a *Quality Function* that generates a score, which is typically a real number, that synthesizes the evaluation of the problem and thus measures the performance of the classification algorithm.

In many cases, simple quality functions, such as the following, are applied: Let A be a supervised classification algorithm and let $\Phi(A)$ be the quality function that evaluates it and which expression is $\Phi_1(A) = x/(x+y+z)$, where x is the number of patterns correctly classified by the algorithm, y is the number of patterns incorrectly classified, z is the number of abstentions. Other times, much more detailed quality functions are applied, such as $\Phi_2(A) = \frac{1}{n}\left(\sum_{i=1}^{k}\sum_{i=}^{k}\alpha_{ij}E_{ij} + \sum_{s=1}^{k}\beta_s A_s\right)$, where: n is the total number of patterns in the control sample, k is the number of classes in the problem, α_{ij} is the amount of objects that belong to class i, mistakenly classified in class j, E_{ij} is the specific weighting of the mistake counted in α_{ij}, β_s is the amount of objects that belong to class i in which the algorithms refrained from classifying, and A_s is the specific weight of the error counted in β_s.

Of course, the decision regarding the quality function to be used in the evaluation of a specific problem depends largely on the conditions and semantics of the problem, so there is an infinite amount of possible quality functions. Regardless of how complex the selected quality function may be, the result of the evaluation is always expressed with only one number, which hides the details and the specific reasons for the assigned classification.

In the case of non-supervised problems, there is no explicit formula to evaluate the quality of the classification algorithm. However, opposite to what happens in the supervised case, the idea of measuring the quality of the resulting covering in terms of its structural conditions [6] is quite common. The structural aspects evaluated in a covering are several, but commonly aspects considered include the compacting of clusters, the separation between clusters, the max and min degree of membership of each cluster, etc. (See [7,8]).

Several indexes have been proposed to evaluate partitions and coverings. Three of the more widely used are the *Partition Coefficient* and the *Entropy index* proposed by Bezdek [9], and the *Xie-Beni index*[6]. Let us examine each one of them.

For a non-supervised classification problem, with n patterns and with k being the pre-determined number of classes to be formed, Bezdek defines the partition coefficient (PC) in [9] as $PC = \sum_{i=1}^{n}\sum_{j=1}^{k}\left(\mu_{ij}\right)\Big/n$, with μ_{ij} being the membership of pattern i to class j. Under the same assumptions, Bezdek also defines the partition

entropy (*PE*) as $PE = \sum_{i=1}^{n} \sum_{j=1}^{k} \left(\mu_{ij} \log \left(\mu_{ij} \right) \right) / n$. The main disadvantage of these indexes, as Bezdek himself states in [9] is that they evaluate each class by considering exclusively the degrees of membership assigned to the patterns and not their (geometric) structure or the structure of the whole covering. X. L. Xie and G. Beni proposed an index that evaluated two structural aspects: the *Compactation* and the *Separation* of the classes [6]. For them, an optimum partition is that which has a strong compacting and a noticeable separation between clusters. Therefore, they proposed the compacting measure as $\Psi = \sum_{i=1}^{k} \sum_{j=1}^{n} \mu_{ij} \left\| x_j - v_i \right\|^2$ with v_i being the centroid of each class. The second factor then represents the Euclidean norm of the difference between each object and the corresponding centroid in each class. The separation between classes, is calculated as $\Xi = n \left(\underbrace{\min}_{i \neq k} \left\| v_i - v_k \right\|^2 \right)$. Lastly, the Xie-Beni (*XB*) index is formed as the quotient of these two quantities, i.e., $XB = \Psi / \Xi$.

Like in the case of supervised problems, all of these structural indexes limit their evaluation to only one number which, in this case, represents the quality of the structuring in the solution covering generated by the classification algorithm.

Most authors do not even consider partially-supervised problems as a different category of problems [10]. These problems are treated as supervised in what regards the evaluation of the classification algorithms. Therefore, in the rest of this paper, no explicit reference will be made to partially-supervised problems and the same conditions of supervised problems will be assumed for them.

3 Advantages and Disadvantages of Traditional Evaluation Methods

The most evident advantage of evaluating supervised problems through quality functions is the flexibility of the latter. The researcher can build a quality function as thorough as the problem requires, and one that can encompass situations of very different kind, such as abstentions of the classifying algorithm or a different weighing for each type of error made in assigning memberships. In return for this, the way of evaluating supervised problems has some evident disadvantages. The first and most noticeable one is the need for having a previously known solution for the problem being evaluated, and its consideration as "the correct solution to the problem". This requirement makes it impossible to evaluate problems for which such a solution is not available, and even more: the consideration of such solution as the correct one may cause important biases in the evaluation of the algorithm. There are two main reasons for these biases in the evaluation: first, the quality of the supervision sample used for the evaluated classification algorithm. Second: the quality of the structure induced on the solution covering by the evaluated algorithm is not measured.

Not evaluating the quality of the supervision sample used for a supervised problem seriously limits the ability to judge the action of the classifying algorithm. It is not hard to imagine that a very well built sample (with the more representative patterns of

each class) may induce the generation of the same solution even by less precise algorithms, while a poorly built sample (with patterns not very representative of each class) may induce errors or abstentions in the algorithms based on the similarity of patterns. The criteria through which a solution can be selected and considered as correct, are not clear. Should the methodology include any type of measurement of the structure of the solution covering generated by the classification algorithm, the evaluation would not depend so much on the quality of the supervision sample. Nonetheless, the quality function is limited to comparing the membership to each of the classes assigned by the classifying algorithm to each pattern. Lastly, notice that most of the classification algorithms (both for supervised and for non-supervised problems) are based on measuring the similarity between two patterns. The criterion or set of criteria through which the similarity is measured is called *Pattern Analogy Function* and it is evident that in spite of the fact that this function is the most important element for the algorithm, it is in no way considered by the evaluation methodology for supervised problems. In summary, the following disadvantages may be noticed:

1. The quality of the supervision sample is not measured.
2. The structural quality of the solution covering is not measured.
3. The pattern analogy function is not involved in the evaluation.

The way to evaluate non-supervised problems has very different characteristics. The evaluation is made based on the quality of the structure of the solution instead of comparing with a previously known solution is by far the most evident advantage of this method. Unlike what happens with supervised problems, no elements, such as the magnitude of the membership assigned to each pattern or the number of abstentions in which the algorithm incurs are considered (although non-supervised classification algorithms almost never have the possibility of abstaining from classifying any pattern). In general, the elements considered to make the evaluation are precisely those which are not considered in supervised problems. These evaluation methods are radically different in both cases, but the diverse conditions of each type of problem do not allow the indiscriminate use of the respective methods. Nevertheless, in both cases the evaluation of the algorithm is reduced in its expression to only one number which generally hides more information than the one it gives, because it does not allow an analysis of the specific situation of a pattern or category. Therefore, the list of deficiencies of classical methods may be completed as follows:

4. The evaluation is synthesized in only one number which does not allow alternative interpretation.
5. The evaluation methods are not unified for all types of problems.

This leads us to ask the following question: Is it possible to devise an evaluation methodology that can overcome the deficiencies found in the present methods and produces unified criteria to evaluate classification algorithms applied to any type of problem?

4 The Main Definitions and Proposed Methodology

Before presenting the methodology proposed by the authors for the solution of the question mentioned above, we now introduce the three most important theoretical

concepts on which the design and methodology are based. These concepts are: the *Covering*, the *Classification Problem* and the *Classification Algorithm*. For a more detailed description see [10].

Let Ω be a known universe of objects under study and let $O \subseteq \Omega$.

Definition 1. A Covering of O is a tuple $(O, \Re, \delta, Q, \pi, C_c, f)$ where O, $\Re$ and Q (called structural sets) are respectively sets of objects, descriptive features for the objects and classes. Components δ and π (called structural relations) are, respectively, description and membership functional relations. The first one describes the objects of O in terms of the features in $\Re$ and the second one assigns to each o_i object a membership to each of the C_j classes. Last, C_c and f are respectively a set of comparison criteria and the pattern analogy function (see [10]).

According to the definition given above, the special types of coverings shown in the following table may be characterized.

Table 1. Types of Coverings.

Covering	Conditions
Total	Every object belongs to a class
Partial	There is an object that does not belong to any class
Blind	No object belongs to any class
Strict	All classes are not empty
Flexible	There is an empty class

Definition 2. A classification problem is a tuple of the form (Z_0, Θ) where Z_0 is an initial cover.

Following definition 2 a problem is supervised if and only if its initial covering is strict; partially supervised if and only if its initial covering is flexible and non-supervised if and only if its initial covering is blind.

Definition 3. A classification algorithm is an algorithm of the form $A(P) = Z_1$ such that, as a parameter, it receives a classification problem (in any of its forms) and delivers a total final covering which is the solution to this problem.

When the design of this evaluation methodology was made, two general objectives were established: 1) to generate a unified methodology for all types of classification problems, and 2) to keep the advantages of each classical method, but to overcome their disadvantages. According to the definitions of the previous section, the methodology designed to evaluate classification algorithms is based on the structural comparison between the initial covering of a problem and the final covering generated as a solution by the classifying algorithm. Such a comparison can always be made, even in cases in which one of the two compared coverings is a blind covering (as in the case of an initial covering in non-supervised problems). The comparison of all types of properties in the covering that involve the membership of patterns to classes and the similarity between them, in accordance with the analogy function between patterns is considered as a structural comparison.

The proposed evaluation methodology obviously starts with the application of the classifying algorithm to the problem being solved. From that moment on, the evaluation process is developed in the following three stages:

Stage 1 (Structural Analysis of the coverings) During this stage the initial and final coverings of the problem are analyzed separately, calculating for each of them the same set of structural properties. These properties are discussed in detail in a later section. The analysis takes place at three levels for each covering:

Level of the Objects: The structural properties are calculated for each object, making reference to each class in the covering.

Level of the Classes: The values corresponding to each of the structural properties in the patterns that form the support of each class are accumulated and averaged.

Level of the Covering: The indexes for the structural properties for the covering under study are calculated.

Stage 2 (Comparison between Coverings) The difference in the value of each one of the structural properties calculated for each covering during the previous stage is calculated. The calculated set of differences is called *Difference Tuple* and it is the score assigned to the classifying algorithm. This tuple expresses the structural change generated by the classifying algorithm in the initial covering of the problem.

Stage 3 (Interpretation of the Score) Once we have the partial results of each of the previous stages, particularly those corresponding to the three levels of structural analysis of the coverings, the researcher interprets the obtained score.

Unlike classical methods, the one proposed here refrains from reducing the evaluation process to only one final score that hides the details involved in the evaluation process. The partial results obtained in each stage are valuable sources of information for the researcher, where he can study particular situations regarding the problem being solved. Another distinctive characteristic of this methodology is the fact that it is useful independently of the quantity and selection of the structural properties calculated during the first stage. Sometimes the researcher may be interested in using a specific set of structural properties, according to the characteristics of the problem under study. For this reason, the methodology described above was introduced without any reference to the specific properties used in the analysis of coverings. In this sense, the set of structural properties that have been used and are described in the next section are shown for the sole purpose of clarifying all of the elements involved in the methodology. Nonetheless, the researcher is free to use the set of properties that he deems to be more adequate for his particular study.

5 Application Details

For the structural analysis stage, only four properties, considered as determining factors in the structure of a covering, are calculated:

The *Tipicity* (*T*) of an o_i object, with regard to a C_j class, understood as the degree to which the object is representative of such class and it is calculated as follows:

$$T(o_i, C_j) = \sum f(o_i, o_s)\pi(o_s, C_j) \Big/ \left|Sop(C_j)\right|$$

where $f(o_i,o_s)$ is the similarity between the objects (calculated by the pattern analogy function), $\pi(o_s, C_j)$ is the membership of the o_i object to the C_j class, $\left|Sop(C_j)\right|$ is the cardinality of support of the C_j class.

The *Contrast* (C) of an o_i object with regard to a C_j class, understood as the degree to which the object is representative of all of the other classes in the covering is defined as:

$$C(o_i, C_j) = \sum_{Cs \neq Cj} T(o_i, C_s) \Big/ k - 1$$

where $T(o_i,C_s)$ is the tipicity of the o_i object, in the C_j class, and k is the total number of classes in the covering.

The *Discrimination Error* (ε) of an o_i object with regard to a C_j class, understood as the degree of confusion of the object in the covering is defined as:

$$\varepsilon(o_i, C_j) = \sum_{Cs \neq Cj} \pi(o_i, C_{js})$$

where $\pi(o_i, C_{js})$ is the degree of membership of o_i to the intersection of the C_j and C_s classes.

The *Characterization Error* (γ) of an o_i object with regard to a C_j class, understood as the difference between the belonging of the object to the class and its Tipicity in this same class is defined as:

$$\gamma(o_i, C_j) = \left|\pi(o_i, C_j) - T(o_i, C_j)\right|$$

During the analysis at the level of the classes, each of these structural properties is averaged in the analyzed class. During the analysis at the level of the covering, the structural indexes corresponding to each property are calculated. In every case, the index is calculated as one minus the corresponding property averaged in the whole covering.

Striving to give this methodology the same flexibility shown by the quality functions in supervised problems, a special technique for the structural analysis of the coverings during the first stage was developed. This technique consists of adding to each covering an additional class which represents the complementing set for the rest of the classes in the covering and then calculating all the structural properties, also regarding this class. In the initial covering of a problem all of the patterns that are not classified will be considered to have maximum membership to the complementing class. This technique allows the proposed analysis to account for the abstentions incurred by the classification algorithm although, evidently, without achieving the same degree of flexibility achieved by the quality functions.

6 Experimental Results

In order to test the designed methodology, we used the famous *IrisData* set consisting of 150 Iris flowers, described by 4 features (length and width of petals and sepals, all

measures in centimeters) and grouped in 3 classes called *Iris Setosa*, *Iris Versicolor* and *Iris Virginica* (See [11]). This data set was used to evaluate two different algorithms: a (supervised) simple voting algorithm and the (non-supervised) Fuzzy C-Means algorithm. Two experiments were performed with the supervised algorithm, one of them with a very well built training sample and the other one with a badly built one. In the non-supervised case the algorithm was simply applied to the whole Iris-Data set to study the solution covering generated. In every experiment, the action of the classification algorithm was evaluated with the traditional methods as well as with the proposed methodology, and both evaluations were compared.

Table 2. Training samples for the supervised experiments

Well-built sample:

Iris Setosa					Iris Versicolor					Iris Virginica				
object No.	length & width Petals		length & width Sepals		object No.	length & width Petals		length & width Sepals		object No.	length & width Petals		length & width Sepals	
27	5	3.4	1.6	0.4	84	6.0	2.7	5.1	1.6	117	6.5	3.0	5.5	1.8
8	5	3.4	1.5	0.2	56	5.7	2.8	4.5	1.3	148	6.5	3.0	5.2	2.0
24	5.1	3.3	1.7	0.5	64	6.1	2.9	4.7	1.4	104	6.3	2.9	5.6	1.8
29	5.2	3.4	1.4	0.2	74	6.1	2.8	4.7	1.2	138	6.4	3.1	5.5	1.8
40	5.1	3.4	1.5	0.2	79	6.0	2.9	4.5	1.5	105	6.5	3.0	5.8	1.8

Badly-built sample:

Iris Setosa					Iris Versicolor					Iris Virginica				
object No.	length & width Petals		length & width Sepals		object No.	length & width Petals		length & width Sepals		object No.	length & width Petals		length & width Sepals	
16	5.7	4.4	1.5	0.4	94	5	2.3	3.3	1	107	4.9	2.5	4.5	1.7
42	4.5	2.3	1.3	0.3	58	4.9	2.4	3.3	1	110	7.2	3.6	6.1	2.5
15	5.8	4	1.2	0.2	51	7	3.2	4.7	1.4	114	5.7	2.5	5	2
82	5.5	2.4	3.7	1	22	5.1	3.7	1.5	0.4	25	4.8	3.4	1.9	0.2
121	6.9	3.2	5.7	2.3	113	6.8	3	5.5	2.1	80	5.7	2.6	3.5	1

In the supervised experiments, both training samples consisted of five objects representing each class. For the well-built case the five objects with more intra-class similarity were selected and for the badly-built case, each class was represented by the three less intra-similar objects and two more objects randomly selected from the other two classes. Both samples are shown in table 2.

Table 3. Traditional evaluation for the two supervised experiments

	$\Phi_1(A)$	$\Phi_2(A)$
Case 1 well-formed sample	0.793	0.854
Case 2 badly-formed sample	0.760	0.837

The two quality functions $\Phi_1(A)$ and $\Phi_2(A)$ introduced in section 2 were used for the traditional evaluation of these supervised experiments. When the rest of the objects not contained in the training sample were submitted for classification, the supervised algorithm produced, for each experiment, the results shown in table 3.

Notoriously, in both cases, the traditional evaluation showed very similar results for the well-built and the badly-built samples. This can only be explained due to the restrictive nature of evaluation methodology using quality functions. This traditional evaluation method measures only the degree of matching between the results obtained by the algorithm and the results contained in the previously known solution. All other aspects of the evaluation are not taken into account, including the quality of the test sample.

In contrast, the unified methodology proposed herein measures the structural quality of both the initial and final coverings on the problem. As stated in section 4, this unified methodology unfolds over three stages. During stage #1 both coverings (initial and final) are analyzed separately and the four structural properties introduced in section 5 are calculated for each one (tipicity, contrast, discrimination and characterization). Since the analysis of each covering takes place at three levels (objects, classes and covering), this stage produces very large tables of intermediate results. These tables are not included in this paper for space reasons. During stage #2, the analysis of both coverings is compared and the Difference Tuples depicted in table 4 are produced.

Table 4. Unified methodology results for the supervised experiments

Case 1 (well-built sample):

T	C	ε	γ
-0.213	-0.016	-0.299	-0.114

Case 2 (badly-built sample):

T	C	ε	γ
-0.038	-0.196	-0.411	+0.342

Finally, during stage #3 the difference tuples are interpreted. The interpretation of these results indicates that by establishing the quality of the initial sample given to the algorithm, the unified methodology manages to evaluate both cases in a notoriously different way. The above results show the structural change produced by the algorithm between the initial and final coverings of each experiment. Interpreting each index of the above table the following observations may be stated:

1. The Tipicity index (T) was reduced reduced much more in the well-built case than in the badly-built one. This means that in the first case the quality of the training sample was so high that the algorithm did not manage to group the rest of the objects with the same representativity in each class. By contrast, in the second case the quality of the sample was low enough that the algorithm kept almost the same covering quality while classifying the rest of the objects.
2. The Contrast index (C) had the expected opposite behavior of the tipicity index, meaning that classified objects kept to be equally representative to all classes in the first case but not in the second one.
3. The Discrimination index (ε) reduction in the second case nearly doubled that of the first case. This means that, after applying the classification algorithm in the second case, the resulting covering has more overlapping in its memberships than the first case. This is an expected result if one considers the quality of the respective samples and the behavior of the tipicity and contrast indexes.

4. The Characterization index (γ) shows the most dramatic change by reducing its value in the first case and growing considerably in the second case. This is also the most unexpected and significant result of this evaluation. In the first case the slight reduction is explained because of the high quality of the training sample. In the second case, the classification algorithm behaves very consistently with the low inter-class similarity of the objects contained in the sample, so by classifying the rest of the objects it reduces the average difference between the membership and the tipicity of all objects.

These arguments lead to the following conclusion: the supervised algorithm used in this experiments is highly sensitive to the quality of the initial sample, especially to the inter-class similarity of the objects. The proposed evaluation methodology allows the researcher to consider different structural aspects that the traditional evaluation methods hide when synthesizing to only one number. This evaluation takes into account both, the structure of the initial and final coverings in the problem, and the change induced by the classification algorithm separately.

In order to traditionally evaluate the non-supervised experiment, the *PE* and *XB* indexes were used yielding the results shown in table 5.

Table 5. Traditional evaluation for the non-supervised experiment

	PE	*XB*
non-supervised experiment	0.157	0.395

The low partition entropy is a good score for the classification algorithm and it means that the final covering has a very clear structure. Nevertheless the middle-range magnitude of the *XB* index indicates an unbalanced ratio between compactation and separation among classes. So these indexes are not very consistent with each other.

For this experiment, the proposed methodology overcomes the inconsistency of the structural indexes and their inability to evaluate by using the same pattern analogy function used by the classification algorithm. Again, the unified methodology unfolds over its three stages and once more the tables containing intermediate results for stage #1 are not shown. Stage #2 (comparison between coverings) yields the difference tuple shown in table 6.

Table 6. Unified methodology results for the non-supervised experiment

T	C	ε	γ
+0.421	+0.421	+0.883	+0.815

During stage #3 the interpretation of each index in the same way as it was done in the supervised experiments, leads to the following observations:

1. In contrast with the supervised experiments, the change in all indexes has a positive magnitude. This is to be expected since the initial covering was blind and so it had no structure.

2. The growth in tipicity and contrast is exactly the same. Also the growth in discrimination and characterization is very similar and notoriously high.
3. Tipicity and contrast are consistent with each other, and they mean that the classification algorithm produces an increase in the tipicity of all classes almost in 42%.
4. Discrimination and characterization are also consistent with each other and they indicate that the algorithm produces 80% more distinguishable objects with 80% more consistency between their tipicity and their membership in each class.
5. The pattern analogy function used by the algorithm is also used in the calculation of each of the four indexes.

In summary, this experiment shows that the consistency of interpretation among the four structural indexes is far superior to that of other structural indexes used in traditional evaluation methods, and that each calculation uses the same pattern analogy function employed by the classification algorithm. So, the methodology proposed herein showed that it fulfills its design objectives and at the same time, it gives more information and flexibility to the researcher.

7 Conclusions

Comparison between the initial and final coverings of a problem allow the evaluation of the behavior of the classifying algorithm independently from other circumstantial factors in the problem, such as the quality of the control sample in the case of supervised problems. Thanks to the definitions previously established, such comparison is a common element between supervised and non-supervised problems and unifies the evaluation methodology.

The specification of what is meant by structural properties allows us to include in the analysis of the coverings both, the basic elements considered by the quality functions (membership assigned to each pattern in each class), and those considered by most of the structural indexes with which non-supervised problems are evaluated. At the same time, the main disadvantages of classic methodologies are avoided. Notoriously, the discussed methodology neither requires a previously known solution to the problem, nor evaluates the algorithm by considering such solution as a reference point.

The flexibility of the discussed methodology may be seen in two main aspects: first, the possibility of changeing the set of structural properties to be used during the analysis of the coverings, and second, the possibility of accounting for the abstentions of the classifying algorithm by using the complementing class technique.

References

1. Fukunaga, K., Hayes, R.R. Estimation of classifier performance, IEEE Transactions on Pattern Analysis and Machine Intelligence, 11(10), 1087 – 1101, 1989.
2. Berikov V., Litvinenko A. The influence of prior knowledge on the expected performance of a classifier. Pattern Recognition Letters, 24, 15, 2537 – 2548, 2003.

3. Fawcett, T. An introduction to ROC analysis. Pattern Recognition Letters, 27, 8, 861-874, 2006.
4. Arbel, R., Rokach, L. Classifier evaluation under limited resources. Pattern Recognition Letters. 2006 Available online.
5. Martínez-Trinidad, J.F., Guzman-Arenas A. The logical combinatorial approach to Pattern Recognition an overview through selected works. Pattern Recognition, 34, 4, 1-11, 2001.
6. Xie, X.L., Beni, G. A validity measure for fuzzy clustering. IEEE Transactions on Pattern Analysis and Machine Intelligence, 13(8), 841-847 ,1991.
7. Lee-Kwang, H., Song, Y.S., Lee, K.M. Similarity measure between fuzzy sets and between elements. Fuzzy Sets and Systems 62, 291-293, 1994.
8. Dae-Won, K., Kwang, H.L., Doheon, L. On cluster validity index for estimation of the optimal number of fuzzy clusters. Pattern Recognition 37, 2009-2025, 2004.
9. Bezdek, J.C. Cluster validity with fuzzy sets. Journal of Cybernetics, 3, 58-73, 1974.
10. Godoy-Calderón, S., Lazo-Cortés, M., Martínez-Trinidad, J.F. A non-classical view of Coverings and its implications in the formalization of Pattern Recognition problems. WSEAS Transactions on Mathematics, 2, 1-2, 60-66, 2003.
11. Bezdek J. C., Keller M. J., Krishnapuram R. *Will the Real Iris Data Please Stand Up?*. IEEE Transactions on Fuzzy Systems, 7, 3, 368-369, 1999.