

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

Supervised Classification Problems – Taxonomy of Dimensions and Notation for Problems Identification

Ireneusz Czarnowski [0000-0003-0867-3114]¹, **Senior Member, IEEE,**
and Piotr Jędrzejowicz [0000-0001-6104-1381]¹, **Member, IEEE**

¹Department of Information Systems, Gdynia Maritime University, Morska 83, 81-225 Gdynia, Poland

Corresponding author: Ireneusz Czarnowski (e-mail: i.czarnowski@umg.edu.pl).

“This work was supported by the Gdynia Maritime University.”

ABSTRACT The paper proposes a taxonomy for categorizing the main features of the supervised learning classification problems and a notation for the identification of the supervised learning classification problem categories. The proposed taxonomy has been based on the review and analysis of the recent literature. It allowed the construction of the landscape of decision problem factors influencing the supervised learning processes. To enable a concise and coherent identification of supervised classification problems we have suggested a notation enabling description and identification of various supervised learning classification problem types and their critical features. The notation consists of 5 fields representing, in a sequence, a structure and properties of decision classes, structural model and properties of attributes, features of the data source, and the performance measure used for constructing and evaluating a classifier. The proposed notation is open and could be extended in the case of need new developments within the machine learning theory.

INDEX TERMS Machine learning, supervised classification, classification problems, taxonomy of features, notation for problem description

I. INTRODUCTION

Classification is a term commonly used for describing the process of distinguishing and distributing kinds of "things" into different groups. Classification can be viewed as the assignment of elements to pre-defined classes [1] or as the act or process of dividing things into groups according to their type [14]. Classification re-mains one of the main topics of scientific research and is vital to practically all domains of human activities. As it has been recently observed "most of the classifications are still based on the evaluation of resemblances between objects that constitute the empirical data. This one is almost always computed by the means of some notion of distance and some algorithms of aggregation of classes" [58].

Classifications are produced using different reasoning schemes. For example, in statistics, a classification task (also called discrimination problem) requires a classification rule for assigning new data to one of the known classes. Such a rule is identified based on a set of data containing observations (or instances), whose category membership is known [3]. In the machine learning terminology [4],

classification is viewed as learning from examples (observations, instances), where a training set consisting of correctly identified observations is available and used to induce a model that describes and distinguishes data classes [11]. Such a model is called a classifier [5]. Classifiers are sometimes referred to as mathematical functions that map input data into a category. Such functions are also called the hypothesis.

Classification problems are encountered in many real-world activities where finding effective methods and tools requires an interdisciplinary effort [1], [2]. Classification methods and tools have been, and still are, under development in numerous research areas of computer science, such as, for example: image processing and analysis [6], [7], computer vision [8], signal recognition [9], decision support systems [10], knowledge discovery in databases and data mining [5], data science [14]. It would be impossible to list here all application areas where machine learning-based classifiers have proven to bring breakthrough advantages. Several examples of such areas include genetics and genomics [127], agriculture [128], molecular and materials

science [129], physical sciences [130], hydrology [131], chemistry [132] manufacturing [133], medical image analysis [134], just to mention a few.

In the current paper, we study classification problems that can be solved automatically. Besides, it is assumed that processes of classifier learning are based on the principle of supervised learning, where a classifier, often referred to as a classification model, is learned from a set of examples. Under this assumption classification is a two-stage process. The first stage involves learning from data to induce a classification model. The second one involves using the model to predict the class or category of instances with unknown class labels. Research effort undertaken by specialists in various areas of computer and data sciences has resulted in providing a wide range of classification models and approaches for constructing classifiers. These are based on different paradigms, with different scopes, different complexities, and varying degrees of performance. All of them are, however, based on a 2-dimensional conceptual data model consisting of the set of examples (instances, observations) and each example consisting of feature (attribute) values which are assumed to be in some unknown way related to the class or category an instance belongs to. Within the set of examples labels (categories) of instances are known.

The simplicity of such a conceptual model does not, unfortunately, mean that inducing classifiers assuring the required level of performance is an easy task. There is a multitude of factors that make the task complex and difficult or even impossible. Among such factors one should mention, for example, limited availability of examples, ambiguity, uncertainty, and distortions to feature values, different methods of representing and coding feature values, class imbalanced among instances available for inducing a classifier, presence of the concept drift, presence of outliers, complex feature values, and many others. One of the most difficult barriers encountered in developing classification methods is the sheer size of the data available for deriving a classifier. This barrier may apply to both data dimensions - the number of features and the number of instances. It is well known that data analysis including classification, is more complex in the so-called Big Data environment [12], [13].

The main contribution is the identification and ordering of factors influencing the construction and outcomes of classification models in supervised machine learning. Besides, we also propose a taxonomy for supervised classification problems categorization. In the available literature on machine learning, the authors haven't found any attempt to provide a wider scheme for categorization of the supervised learning problems based upon considering a variety of combinations of factors important from the point of view of constructive effective classification models. Our goal is to provide specialists and laymen with a simple classification scheme allowing them to identify what kind of classification problem they are facing. Consequently, they

can narrow their search for a suitable classification model considering at first methods that have proven successful in solving similar problems.

The proposed classification scheme is based on several characteristics of supervised classification problems referred to as their dimensions. Under the proposed scheme the type of the supervised classification problem can be identified considering the following factors:

- Characteristics and properties of the problem categories, (category view).
- Structure and properties of the dataset available for learning a classifier (attribute view).
- Sources of the dataset available for learning (data source view).
- Criterion or criteria associated with the supervised classification problem (performance criteria view).

Besides the classification scheme, we also propose the notation for the supervised classification problems. It has been inspired by the notation used in the theory of scheduling for the identification of scheduling problems. The notation was suggested in 1979 by R.L. Graham et al. [87] and is widely used ever since. The reason behind proposing the notation for denoting supervised classification problems in a short but coherent way based on their main properties is to enable efficient communication between specialists developing classification models.

Classification problems in real-life arise in a variety of settings. Researchers have studied hundreds of classification problems and it would be impossible to list all known variants in a single paper. We believe that offering a framework for ordering and grouping classification problems could be of value to both – the researchers looking for methods and tools, and the practitioners trying to find a method or tools for solving their particular classification problems. The proposed classification scheme and notation should be considered as a part of the meta-analysis of machine learning.

We would like also to stress that the proposed classification scheme does not depend on techniques, methods, and tools used for inducing classification models. It is also independent of techniques and methods used at the data pre-processing stage, for example for dealing with the missing data problem or outliers removal. There is, of course, a relation between a particular supervised classification problem type and methods or techniques which might be more effective for solving its cases as compared with other techniques. Identifying such relations could be a subject of analysis when constructing or selecting a learner.

The remainder of this paper is organized as follows. In the next section, the problem of classification is formally defined. Section III contains a review of the different problems and strategies of learning from data based on the relevant literature. Section IV provides a short overview of studies on complexity of classification problems. In Section V, we propose a notation for classification problems. Section VI

includes conclusions and discussion on some open research problems.

II. CLASSIFICATION PROBLEM FORMULATION

In the case of supervised learning, to induce a learner some examples are needed. The set of examples denoted U , is a non-empty, finite set and is called the universe. The example $x \in U$ is represented by a fixed set of attributes (features), $A = \{a_1, a_2, \dots, a_n\}$, where n is the number of attributes. Each attribute $a_{i:i=1,\dots,n}$ has a value $a_i(x) \in V_{a_i}$, where V_{a_i} is a set of all possible values for attribute a_i . V_{a_i} is also called an attribute domain. It is assumed that one particular attribute, say a_i , contains value representing the class label of the example [11].

The aim of learning from examples is to obtain a model (classifier, learner) able to reveal the value of the unknown class label by identifying the dependence between attribute values (our independent variables) and the value of the class label (our dependent categorical variable) using the set of available instances.

The class labels belong to a finite set of predefined decision categories (classes) $C = \{c_l = 1, \dots, k\}$, where k is a number of these categories. Hence, the U set for the classification task can be defined as:

$$U = \{[x_{ij}, d(x_j)]: i = 1, \dots, n; j = 1, \dots, N\}. \quad (1)$$

Based on the above, a single example is represented by $[x_{ij}, d(x_j)]$, the set U consists of N such vectors and $d(x)$ represents the value of the class label for the example x , where $\forall x \in U d(x) \in C$.

In [11] the process of learning from examples was called a "concept learning", i.e. "search through a predefined space of potential hypotheses for the hypothesis that best fits the training examples". When instances are represented by the n -dimensional input data (space) and each $[x_{1j}, \dots, x_{nj}]_{j=1}^N \in \mathcal{R}^n$, then the mission of a classifier is to map instances to the discrete class set, i.e. $h: \mathcal{R}^n \rightarrow C$.

The machine learning algorithm, called a learner, produces a classifier $h \in H$. The classifier is induced from the set U . H is called as the hypothesis space and consist of all possible hypotheses that can be drawn during the learning process. Thus, given a dataset U , a set of hypotheses H , a performance criterion or criteria F , the learning algorithm L outputs a hypothesis $h \in H$ using the learning algorithm L optimizing F . Thus, learning from examples is to generate L which will be able to determine the best possible $h \in H$ with respect to the adopted performance measure or measures F .

In case the classifier output is evaluated using a single criterion with the performance measure expressed as $f \in F$, the learning from examples may be formulated as a process maximizing the performance measure concerning the hypothesis h , such that:

$$h = \arg \max_{h \in H} f(h). \quad (2)$$

The role of a classifier predicts the category of an instance, where the category is unknown.

Definition: A classifier h is a function assigning examples from D to a predefined set of categories as shown in the equation (3):

$$h: U \rightarrow \{\emptyset, C_1, C_2, \dots, C_k\}, \quad (3)$$

where $C_{l:l=1,\dots,k}$ denotes subset of data set D , such that examples are labelled by a class $c_{l:l=1,\dots,k} \in C$, under the following conditions:

$$1. \bigcup_{l=1}^k C_l = D \quad (4)$$

$$2. \forall C_j \subset D, C_l \subset D, : C_j \cap C_l = \emptyset, j, l = [1, k], j \neq l. \quad (5)$$

In view of the above, the classification task is to assign of the class, $c \in C$, to the instance $x \in D$.

The process of classification involves several steps. A natural sequence of these steps can be seen as follows:

- Sample collection,
- Selection of instances and attributes for learning,
- Carrying other pre-processing actions (cleaning, removing outliers, etc.),
- Producing the training set,
- Induction of the classifier using instances from the training set.
- Using the induced classifier to predict classes of instances with unknown class labels.

III. PROPERTIES OF THE CLASSIFICATION PROBLEMS

The first known classifier based on the supervised learning paradigm was proposed back in 1967 by Cover and Hart [52]. Since then numerous dedicated and universal classification algorithms have been proposed and published (see, for example, Table I).

It is now widely recognized that the selection of methods and tools for constructing classifiers should be preceded by an analysis and assessment of properties of the classification problem at hand. Such an analysis should be carried out considering the following dimensions, shown also in Fig. 1:

- Structure, cardinality, scales, and relations category - features of classes (categories).
- Structural model of data and data characteristics.
- Data source features.
- Classifier performance criterion (or criteria).

TABLE I
SELECTED CLASSIFIER ALGORITHMS

Algorithm	Authors
A ^q - quasi-optimal covering algorithm	R. Michalski [15]
C 4.5 - decision tree	R. Quinlan [16]
CN2 - rule induction	P. Clark at al. [17]
kNN and Instance-based learning algorithms	W. David at al [18]
SVM - Support Vector Machine	V. N. Vapnik [19]
Bayes classifier	L. Devroye et al. [20]
Bagging method	L. Bbeiman [21]
MODLEM	J. Stefanowski [22]
Random forest	L. Breiman [23]

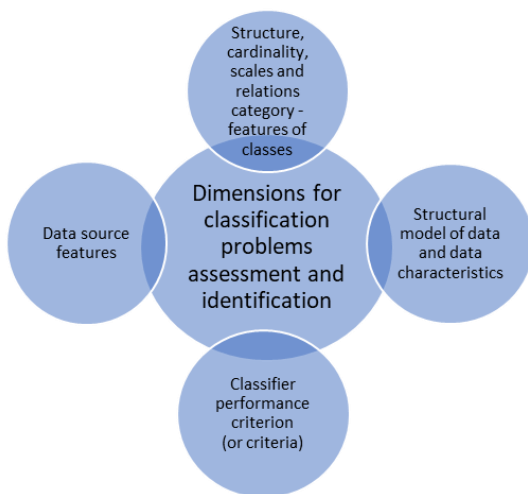


FIGURE 1. Dimensions for classification problems assessment.

A. THE NUMBER, PROPERTIES, AND STRUCTURE OF CLASSES (CATEGORY VIEW)

It is assumed that in the case of supervised learning the classification problem requires deciding to which class an instance belongs. In supervised learning, unlike the semi-supervised, and unsupervised ones, it is assumed that the number of classes (categories) and their possible labels are known at the outset. In the case of the two classes, the problem is called the *binary classification* [37]. In many cases, the binary classification problem has served as a basis for introducing new classification methods as, for example, in the case of the support vector machine (SVM) approach [38].

In the case of binary classification, the cardinality of the set of decision categories is equal to 2 (i.e. $|C| = 2$), and classifying is carried out into one of the two known classes. However, in many practical applications of machine learning techniques, the number of classes is greater than two, i.e. $|C| > 2$ (see for example [40] [41] [42]). In such a case, a *multiclass classification problem* is considered and instances belong to one of three or more classes [39]. It should be noted the binary classification is a special case of the multiclass classification. On the other hand, the multiclass classification can be seen as a natural extension of the binary classification problem.

In a special case where $|C| = 1$, one deals with the *one-class classification* or *unary classification problem*. Unary classification problem requires identification of instances belonging to one particular class only. Such class is arbitrarily referred to as the positive or target one, and it is assumed that the positive class is well characterized by instances. Instances that do not belong to the target class are assumed to belong to the negative class, however, they do not form a statistically representative sample of the negative concept [26], [27], [31]. In the discussed case the aim of the classifier is either to identify only one class amongst all the others possible or to identify positive instances when the negative class examples are either not available, not

adequately sampled, or ill-defined. The one-class classification is, however, encountered in many real-life situations like, for example, detection of outliers [28], novelties [29], faults [34], spam [33], and abnormal behaviors [36]. Some automatic diagnosis [35], document classification [32], and concept learning [30] problems can be also modelled as the *one-class classification problem*.

So far we have considered classification problems where each instance belongs to a single class and is associated with a single class label. Such problems can be solved using methods of *single-label learning* for training the classification model [25]. However in numerous situations classification problems are multi-labelled where one instance can be naturally associated with multiple, non-exclusive labels. Examples include document, gene, and image categorization. In *multi-label learning*, the aim is to learn model mapping instances to the powerset of the decision categories set C . Both, the single and the multi-label classifications are based on a fixed set of labels. Compared with the single-label classification, which predicts only one label for each instance, the multi-label classification is more complicated. Each instance are different and the number of labels per instance is not fixed. A review of the multi-label classification tools and algorithms can be found in [25], [43], and [45]. A special category of multi-label classification problems encountered in mining data streams is discussed in section III.D.

In the literature, the term multi-label classification is often used interchangeably with the term *multi-dimensional classification*. In fact, the multi-dimensional classification can be viewed as a generalization of the multi-label classification where each data instance is associated with multiple class variables. The goal of multi-dimensional classification is to assign each data instance to multiple classes. In the multi-dimensional classification, class labels are allowed to have more than a single value [48]. A real-life cases of the multi-dimensional approach including bio-informatics and multi-fault diagnosis are discussed in [47] and [48].

Another recently investigated classification problem category is the *multi-output classification*. The task of the multi-output classification is to simultaneously predict multiple outputs for a single input. In such case the output values belong to a diversified data types, such as, for example, binary, nominal, ordinal, and real-valued variables data. The problem of multi-output learning has attracted so far the interest of researchers from many areas including, for example, speech recognition, language processing, motion tracking, computer vision, document processing, and ranking in information retrieval [46]. The multi-output classification is also known as the *multitask classification* or *multi-output-multi-class classification*. Multi-label classification can be seen as a special case of the multi-output classification problem.

All the above-discussed classification problems assumed that to induce a classifier there is available a training set consisting of instances, each represented by a single attributes

vector and where each such vector has an associated class, classes, or labels. In many areas, the above assumption does not hold and the classification problems belong to the category of the *multiple instance classification* [50]. In the multiple instance classification, the aim is to learn a classifier based on a training set of bags. Each bag contains multiple attribute instances and has an associated class. However, the labels of the individual instances within a bag are unknown. According to [51] “bags may also contain instances that are not necessarily relevant, do not convey any information about bag class, or are better related to other classes of bags”. A set of bags can be used as a training set, and the multi-instance classification aims to predict the class of unlabelled bags. Examples of the multiple-instance classification problems can be found in medicine, chemistry, image recognition, etc. More information on multi instance learning can be found in [51].

Usually, it is assumed by default that label values can be measured using interval or ratio scales. Such an assumption does not always hold. It appeared that in case label values are defined using nominal and ordinal scales, the resulting supervised classification problems may require different approaches and techniques.

Ordinal classification is the special case of multiple class problems. Hence, the ordinal classification problems can be solved using standard approaches as in the case of other multiple class problems. While applying multiple class classification methods for ordinal data sometimes works, the outcome can be unsatisfactory since classes are treated equally without considering their interconnections and relative superiority [106]. Ordinal classification problems are further discussed, among others, in [107], [112], [108], [110]. The special case of the ordinal classification is monotonic classification [109]. In the ordinal classification, the different labels show an ordering relation, related to the specific nature of the target variable. If additionally, a set of monotonicity constraints has been imposed on the relationship between independent and dependent variables, then the problem is known as monotonic classification.

Other special cases are *multiple criteria ordinal classification problems*. An ordinal classification problem with multiple criteria consists of the assignment of objects to a finite number of ordered classes. Objects are characterized by attributes with ordered value sets and monotonicity constraints assuring that a higher value of an object on an attribute, with other values being fixed, should not decrease its class assignment. The problem was studied, by several authors, among them [114], [113], [118], [119].

In *nominal classification*, categories are mutually exclusive. According to Warrens [135] nominal classification can be further divided into two types. The distinction depends upon the presence of the category “absence”. When there is no ‘absence’ category, a classification can be described as having several unordered categories of “presence” characterizing

possible cases. Such type of nominal classification is referred to as regular as opposed to a dichotomous-nominal classification. History of developments in nominal classification can be found in [111].

Classification problems where there is some structure (hierarchical or not) among the classes form a wide category of structured classification problems [117]. According to the above authors, *hierarchical classification* can be seen as a particular type of structured classification problem, where the output of the classification algorithm is defined over a class taxonomy. Class taxonomy can be defined as a tree-structured regular concept hierarchy defined over a partially ordered set $(C, <)$, where C is a finite set that enumerates all class concepts in the application domain, and the relation $<$ represents the “IS-A” relationship [121].

According to [120], in many real-world classification problems, one or more classes can be divided into subclasses or grouped into superclasses, and instances can belong to more than one class simultaneously at the same hierarchical level. In this case, the classes follow a hierarchical structure, usually a tree or a directed acyclic graph. These problems are known in the literature of machine learning as *hierarchical multiple label classification problems*. Evaluation measures for hierarchical classification were discussed in [116]. An approach for hierarchical multilabel classification was proposed in [115].

The category view of the supervised classification problems is shown in Fig. 2. Category view focuses on cardinality, structure, and properties of categories (labels) encountered in different classification problems.

B. DATA VIEW

A natural structure of data used to induce learners is the relational database model or decision table model, where data are kept in tables [62]. Each row in the table represents an instance (example) with a unique class label. The columns of the table hold attribute values, and each instance usually has a value for each attribute. The model is appropriate for binary learning and multi-class learning, and data organized in a table-like structure are called well-structured. The well-structured data are also used in the case of the multiple instance learning, although the class labels are not necessarily provided for all instances from bags belonging to the training set [51]. The one-class classification is also based on the discussed structure even though the only positive class instances are guaranteed and the negative class instances can be absent, unlabelled, or not properly defined.

A more complex data structure is required for multi-output learning. In this case, the outputs can be of various types and structures. Different output structures typical for multi-output learning, including independent binary vectors, independent real-valued vectors, rankings, sequences, graphs, trees, links, images, text, audio, and time series, are discussed in [55].

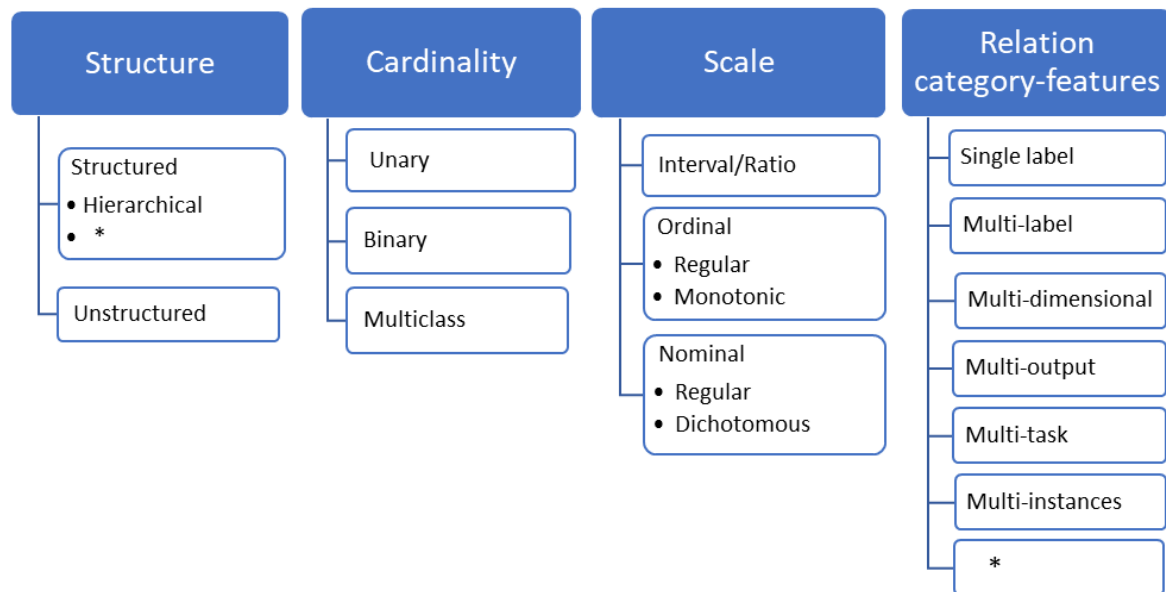


FIGURE 2. Supervised classification problems – category view (“*” denotes remaining options).

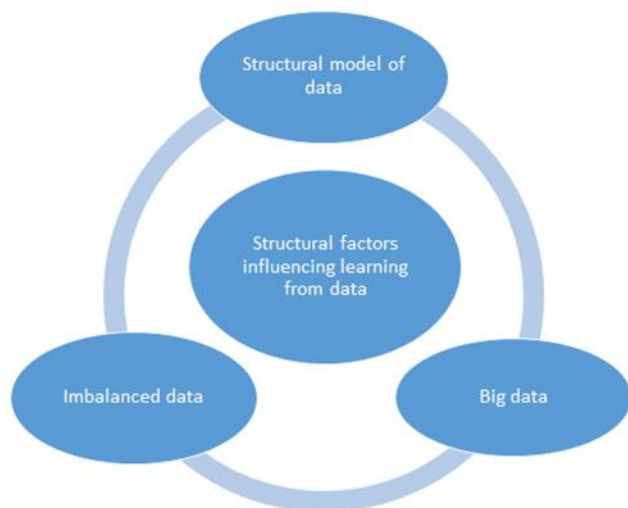


FIGURE 3. Data view – a general scheme.

Unstructured data either does not have a pre-defined data model or is organized loosely. Unstructured data has an internal structure, but it is not predefined through data models. It might be human-generated, or machine-generated in a textual or a non-textual format. Unstructured data can be defined as all the data that is not structured. Unstructured data is mostly qualitative. Examples of such data include all kinds of text and audio data, e-mails, web pages, business documents, FAQ's, multimedia content, spatial data, molecular structures, chemical structures, and others [56], [88], [89]. Solving classification problems based on unstructured data is referred to as *learning from unstructured data* [88], [89].

Structured data is most often quantitative data. When data has well-structured the process of solving classification problems is less demanding from the implementation and computational side [88]. However, learning from structured data such as sequences, trees or graphs is less trivial than learning from data organized as decision tables. Challenges in learning from structured data arise in the so-called hybrid domains, where, for example, continuous and discrete structures are mixed. Dealing with hybrid structures and structures representing social networks is discussed in [90]. In [91] it was observed that even learning from the structured data is nontrivial. The main reasons behind this finding include ignorance of structural information on input and output domains and the occurrence of high-dimensional structured data containing huge numbers of features and labels. Current methods for learning from structured data are also limited in handling large, isolated substructures [92].

Apart from structured and unstructured data, some specialists use also the term *semi-structured data*. Semi-structured data has characteristics of both structured and unstructured data. These data are not structured using a relational database model but they have elements of semantic markups that enforce hierarchies assuring that some structure is kept [57]. An example of *semi-structured data* is also discussed in [93], where conditions for *learning from semi-structured data* are presented. An approach for *learning from semi-structured data* was suggested in [94], where a genetic programming algorithm for extraction of the multiple tree-structured patterns from tree-structured data was proposed.

From the point of view of practical application, important examples of structured data include time-series data, multi-

view data. When data is represented by multiple, distinct feature sets one deals with the multi-view data and multi-view learning. An excellent survey of the multi-view learning approaches and algorithms can be found in [124]. Another important and challenging problem in data mining is *time-series classification*. In time-series classification a training dataset is a collection of pairs $[X_i, Y_i]$, where X_i could either be a univariate or multivariate time series with Y_i as its corresponding label vector. Reviews related to time-series classification can be found in [126] and [125].

The knowledge of the structural data model can determine the learning process with respect to the decision on the algorithm or tool which should be used or what learning

strategy should be applied to produce a strong and a high generalized system, highly competent to deal with different types of data, from numbers to textual format, from well-defined structures to undefined ones. Hence, we consider structural properties of data as an important dimension of classification problems influencing learning processes. On the other hand, we assume that quality of data is not a dimension of classification problems. Cleansing data, removing outliers, dealing with missing data, reducing data and other imputation efforts, remain an important tasks at the pre-processing stage.

To sum up this subsection, a general scheme for data view is shown in Fig. 3, while a graphic representation of the structural data model is proposed in Fig.4.

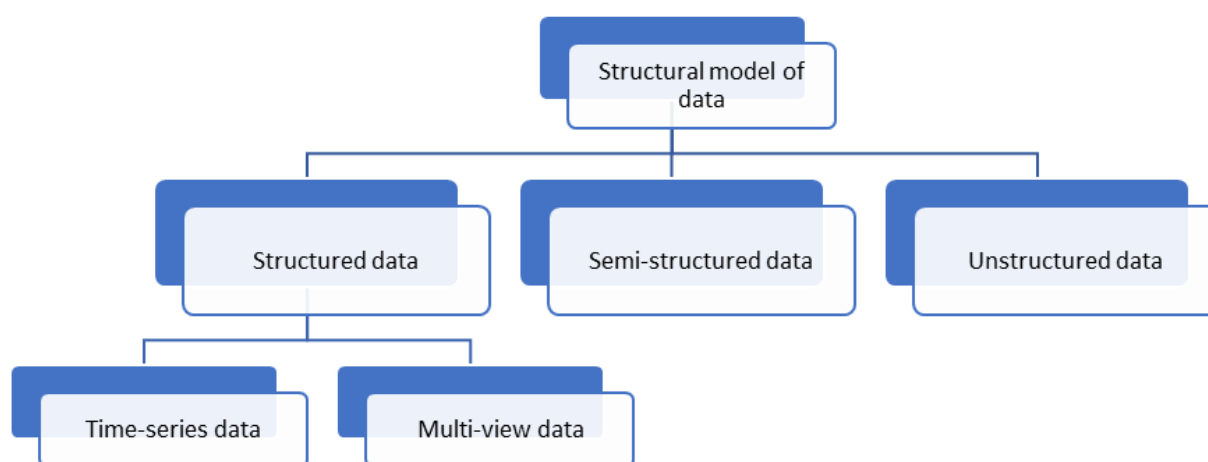


FIGURE 4. Graphic representation of the structural data model.

C. ATTRIBUTE VIEW

The traditional machine learning paradigm is based on the processing of examples as multidimensional vectors of attributes. Each attribute has a domain determined by the attribute type. The domain of each attribute may be either symbolic or numerical. The majority of the machine learning algorithms deal with the following types of attributes [61]:

- Numerical (continuous) attributes - they take real or integer values and can have an infinite number of states.
- Nominal attributes (also called categorical) - the values are determined on a predefined set of possible values.
- Ordinal attributes - they are numeric or nominal, and contain values that have a meaning in terms of ranking or order.
- Discrete attributes – they have a finite or countably infinite numerical or categorical value. In the case of the domain consisting of two possible values for this type of attribute, the attribute type is referred to as the binary.
- Complex attributes – they reach beyond a simple attribute-value pair and can be represented by a more complex structure like, for example, graphs [76].

Attributes belong to the qualitative, quantitative, or complex types. Attribute view covering the instance (samples)

attributes in the supervised machine learning is presented in Fig. 5.

The type of attributes characterizing the set of instances (examples) may influence the choice of the machine learning technique for solving a classification problem. Hence, in the literature the different techniques of learning from examples have been discussed including, for example, learning from numeric data [63], learning with symbolic attributes [65], learning problem with categorical data [64], learning from spatial data [66], learning from ordinal data [69], learning from collective data (bags of words or items) [67], learning from discrete data [67], and learning from multidimensional data [67], [68].

D. DATA SOURCE VIEW

The basic approach for solving classification problems using machine learning techniques assumes that data remain unchanged during the learning process, so means that are static. Such kind of learning is called *batch learning* [76] and learners are trained using a single batch of data. Batch learning ignores all new data and focuses entirely on previously learned concepts [70]. Batch learning relies on the assumption that data coming from the data source has a stationary distribution.

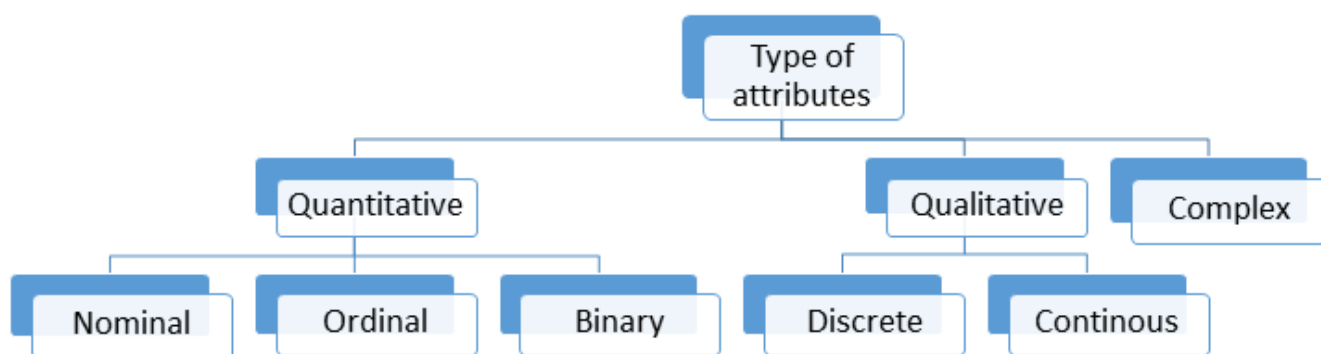


FIGURE 5. Attribute view - types of the instance attribute in the supervised machine learning.

In numerous real-life applications, batch learning is impossible or impractical. More and more often the size of the available datasets outpace the capability of computational hardware to analyze them. One method to deal with the problem is applying so called incremental algorithms that sequentially process chunks or packages of data one by one, combining the results from each chunk. Data chunks can be formed by the user to overcome problems with computational resources or may come sequentially in a natural way from the data source. Learning from the current chunk and modifying the model after the prediction results have been revealed to be ready for the next chunk, is called *incremental learning*, and the data source producing a sequence of chunks is called the *incremental data source*.

A special case of incremental learning is online learning. Online learning is needed to deal with an endless stream of received data like, for example, sensor data, currency rates, stock market indexes, or video streams. In online learning, the class label for the instance is predicted immediately when this instance incoming and the true class label is revealed afterward. In the next step, the incoming instance is incorporated into the training data dataset. In such an environment, learning is categorized as *online learning* [76] and the data source as the *online data source or data stream*.

If the distribution of data from the data source is not constant the domain is said to have a non-stationary distribution. In this case, changes of the underlying data distribution known as the concept drift may occur [73], [74], [77].

Among challenges facing online learning, there is the scalability of the learning process which should be ready to learn from thousands of training examples. Almost at the same time, there is a need to take classification decisions considering new data flowing into the system [80]. Another challenge is coping with the eventual concept drift and dynamic character of the observed data source [71], [72], [75], which requires timely and accurate drift detection mechanisms. It should be noted that distribution changes

may occur not only in the feature space but also in class space, or simultaneously, in both of these spaces [70]. Difficulties should be also expected when the data stream used to induce a learner is class imbalanced.

Many real-world problems involve data which are *multi-label data streams* [49], [95]. The problem of multi-label classification is characterized by unique properties as compared with other types of classification problems. A special feature of the multi-label data streams is that the set of labels is not fixed at the outset and may change during the learning. Besides, more than one label may be assigned to incoming items during the classification process. Examples of multi-label data streams include data from different sensor applications, traffic management, web exploration, manufacturing processes, as well as from the social media networks, where a photo posted on Facebook or Twitter might be labelled continuously and differently by users [49]. Another example is the categorization of the incoming mails, where each email may be relevant to a thematic label, as well as to a label concerning confidentiality. On the other hand, such labels may be correlated. Such labels are called orthogonal [95]. A review of the multi-label data streams learning algorithms can be found in [54], [95].

Data used to induce learners may be stored in one central repository. Another possibility is physical and geographical distribution of the data using, for example, cloud computing technologies, so means that the *distributed data sources* are considered [76].

Data stored in multiple separated sites may be homogenous. In such case each site consists from instances defined on the same set of attribute. When we have various sets of attributes is different in these separated sites, then these stored data are heterogeneous. Although some attributes among the sites can be common. Of course these separated data sets also may to have distinct structure (format differences, semantic differences, etc.) [78], and could have been exposed to horizontal and vertical data fragmentation [79].

E. PERFORMANCE CRITERIA VIEW

As it has been shown in Section II, classification problems belong to a wide class of optimization problems. A performance criterion (performance measure) or a set of criteria, cannot be considered as a feature of the particular optimization problem since the choice of the criterion is at hands of the user who carries out the optimization process. Besides, it is usual that a problem can be solved to optimum using different performance measures. The above observations also hold for the classification problems. Nevertheless, there exist a set of criteria that is commonly used when solving classification problems using machine learning techniques. Among measures belonging to this set, one can list classification accuracy, classification error, classification cost, sensitivity, specificity, the area under the curve, F1 score, precision, recall, and many others [86].

There are, however, classification problems where a narrow range of possible performance measures is justified. For example, in the case of the imbalanced data classification problems, a meaningful set of criteria include, among others, Geometric Mean (G), Area Under the Curve (AUC), Balanced Accuracy ($BACC$), and Mathews Correlation Coefficient (MCC). Another example of the supervised classification problems where specialized performance measures are better suited than the standard ones is ordinal classification. Cardoso and Sousa [122] discuss the problem and propose a specialized criterion for measuring the performance of ordinal classification named Ordinal Classification Index (OCI).

The majority of studies on learning from data are focused on a single-objective optimization, where the aim is to optimize a single performance measure selected by the user [82]. The problem of data classification can be also formulated and solved as the multi-objective optimization case. Solving multi-objective classification problems using the machine learning techniques and the supervised learning paradigm has been studied, for example, in [81], [83], [84], [85], [86].

In subsections III.A to III.E we have discussed various factors and properties of the supervised classification problems that may influence the learning process and could be decisive in selecting an effective learning technique or algorithm. In Fig. 6, the landscape of factors characterizing decision problems and influencing the supervised learning processes is shown.

IV. COMPLEXITY ISSUES

The complexity of the classification problems can be studied in several aspects. One of them takes into account the properties of the classifier induced from the available training set. If such a classifier, that is the function h assigning instances to a predefined set of decision classes, is linear and, at the same time, its predictions assure the required performance level, then the complexity of the classification problem can be considered as a low one. In such a case, finding a linear combination of features that characterizes or separates two or more classes of objects is not a difficult task. If, however, a linear discriminant

function cannot be found or it does not assure the required performance level then one has to look for a non-linear function and the problem becomes more complex [97].

Though there are so far only a few formal results reported in the literature on the complexity of the machine learning classification problems, some interesting ideas for the two-class problems were suggested by Zhao and Wu in [97]. According to [97]: “if a two-class problem is not K -degree linear separable, then we refer to it as a K -degree linear non-separable two class problem”, where K is the number of hyperplanes needed to discriminate between each pair of instances from different classes. Zhao and Wu in [97] further state that: “a two-class problem has K -degree classification complexity if it is K -degree linear separable but not $(K-1)$ degree linear separable”. The proposed concept of the classification complexity can be used to design a multi-layer perceptron with the minimum required number of layers.

The classification complexity can be also seen as depending on both the feature space and the data size. Big sets of data may cause an excessive demand for computational resources. On the other hand, a small training set can appear deceptively simple, however, when the cardinality of the set of attributes of such training set is high the classification problem may not be easy to solve satisfactorily [96].

Early studies on effects of dimensionality, sample size, and structure of classification algorithm on misclassification have concentrated on measures like use of probability distance measure bounds, entropy measures, interclass distance measures, scatter matrices, information-theory-based approaches, boundary methods, feature space partitioning methods [44], [105].

Singh in [96] emphasized that the classification problem complexity should be studied considering decision boundaries. He proposed two measures of classification complexity based on feature space partitioning: purity and neighborhood separability and compared them with probabilistic distance measures and several other nonparametric estimates of classification complexity.

In [98] the complexity of a discrimination problem has been also discussed taking account of the data structure and the number of data. The authors show that an incomplete or sparse sample (relatively small data set) adds a level of complexity, on the other words it means that when the sample is too small the problem may appear only deceptively simple. The small data effects are also considered in Vapnik's VC-dimension theory [99].

The classification complexity can be also evaluated using the computational learning theory (CoLT). It allows estimating potentials of learning algorithms for function approximation and generalization [100]. The CoLT theory is also related to the probably approximately correct (PAC) learning. PAC learning provides a way to quantify the computational difficulty of a machine learning task [101]. The theory is concerned with binary classification, but it remains valid for cases with more classes [102].

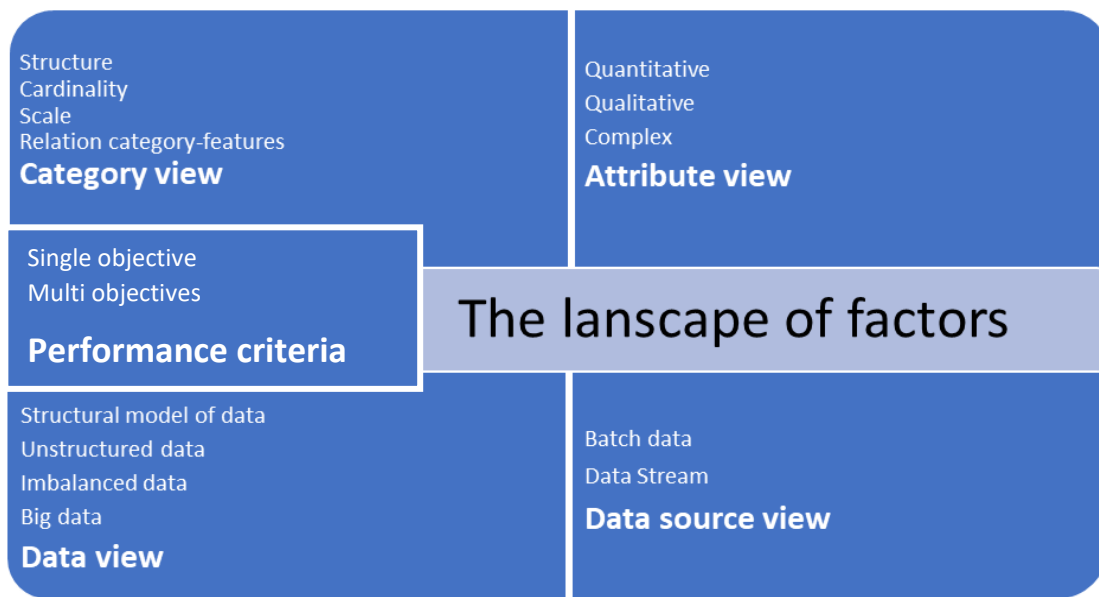


FIGURE 6. The landscape of decision problem factors influencing the supervised learning processes.

The problem of classification complexity is still considered from several different perspectives. Among them, there are these based on estimating the classification problem complexity using different measures. These measures focus on estimating the shape and size of the decision boundary (like, for example in [103]), for binary as well as for multiclass classification problems respectively (see, for example, [103] and [104]). In [103] different complexity measures have been divided into the following categories:

- Feature-based measures – used for characterizing how the available features are informative for class separation.
- Linearity measures – used for deciding whether the classes can be linearly separated.
- Neighborhood measures – used for describing the presence and density of same or different classes in local neighborhoods.
- Network measures – used for identifying structural information in the dataset.
- Dimensionality measures - for evaluating data sparsity.
- Class imbalance measures.

Complexity measures may support various supervised machine learning tasks including data preprocessing, design of machine learning algorithms, and choice of the classifier, adequately to features the available data [103].

V. NOTATION FOR IDENTIFICATION OF THE SUPERVISED LEARNING CLASSIFICATION PROBLEMS

Apart from the classification problem complexity, a rational choice of preprocessing tasks and, later on, the machine learning technique or algorithm, requires identification of all relevant features of the problem at hand. To make this task easier we suggest using a special notation. The idea is inspired

by the notation introduced and used in the field of operations research for scheduling problems [87].

A. COMPONENTS OF THE PROPOSED NOTATION

To identify a supervised classification problem we propose to use the following 4-tuple of fields:

$$\alpha | \beta | \gamma | \delta ,$$

where each field is a comma-separated string of symbols.

The first field, denoted α , represents the category view. It consists of 4 subfields describing structure, cardinality, scale, and the relation category – features. Symbols within a field are separated by a colon. Unknown or undefined values are replaced by *. The first subfield denotes the type of structures and may include one of the following symbols:

- S_{he} – hierarchical structure of categories.
- S^* – user-defined structure of categories.
- U_{ns} – unstructured categories.

The second subfield denotes the cardinality of categories and may include one of the following symbols:

- U_{nr} – unary classification problem.
- B_{in} – binary classification problem.
- M_{lc} – multiple class classification problems.

The third subfield denotes the scale of categories and may include one of the following symbols:

- I/R – interval/ratio scale.
- O_{rr} – ordinal regular scale.
- O_{rm} – ordinal monotonic scale.
- N_{or} – nominal regular scale.
- N_{od} – nominal dichotomous scale.

The fourth subfield denotes relation category - features

and may include one of the following symbols:

- S_{il} – single label problem.
- M_{ml} – multi-label problem.
- M_{di} – multi-dimensional problem.
- M_{ou} – multi-output problem.
- M_{tk} – multi-task problem.
- M_{in} – multi-instance problem.
- M^* – user-defined relation category – features.

Example: Notation $S_{he}:M_{lc}:I/R:S_{il}|*|*|*$ refers to a single label, multiple class problem with the hierarchical structure of categories that can be measured using interval/ratio scale.

The second field, denoted β , represents the data view. This field contains three subfields. The first subfield describes a structure model of data. The second described a data distribution and the third data size. The first subfield may include the following symbols:

- S_{ts} – time series structured data.
- S_{mv} – multi-view structured data
- S_{ot} – another type of structured data
- S_{sd} – semi-structured data.
- U_{nd} – unstructured data.

The second subfield may include the following symbols:

- R_{eg} – regular distribution of data.
- I_{mb} – imbalanced distribution of data.

The third field may include the following symbols:

- R_{si} – regular data size.
- B_{ig} – big data size.

Example: Notation $S_{he}:M_{lc}:I/R:S_{il}|U_{nd}:R_{eg}:R_{si}|*|*|*$ refers to a single label, multiple class problem with the hierarchical structure of categories that can be measured using interval/ratio scale. Besides, data are unstructured, with regular distribution and regular size.

The third field of the proposed notation - γ , represents the attribute type and contains two subfields. The first subfield describes data type and may consist of the following symbols:

- N_{oa} – nominal attributes.
- O_{ra} – ordinal attributes.
- C_{on} – continuous attributes.
- D_{is} – discrete attributes.
- U_{mu} – multidimensional attributes.
- S_{ym} – symbolic attributes.
- S_{pt} – spatial attributes.
- M_{ix} – a mixture of types or unknown attribute types.

The second subfield describes the features of the data source. It may contain one of the following symbols:

- C_{en} – centralised data repository or central repository.
- D_{dis} – distributed data repository
- H_{om} – homogenous data source.
- H_{et} – heterogenous data source.

- H_{or} – horizontal fragmentation.
- V_{er} – vertical fragmentation.
- I_{nc} – incremental data source. Data arrive in chunks.
- S_{tat} – static data source. Data are available in batches.
- S_{ds} – stationary data stream (stationary dynamic, stationary online data source). Data arrive one by one.
- N_{ds} – nonstationary data stream (nonstationary dynamic, nonstationary online data source, data with a concept drift). Data arrive one by one.
- D_{un} – data stream of the unknown character (dynamic, online data source of the unknown character). Data arrive one by one.

Example: Notation $S_{he}:M_{lc}:I/R:S_{il}|U_{nd}:R_{eg}:R_{si}|C_{on}:H_{om}|*$ refers to a single label, multiple class problem with the hierarchical structure of categories that can be measured using interval/ratio scale. Besides, data are unstructured, with regular distribution and regular size. In addition, data are continuous and come from a homogenous data source.

The fourth field δ represents the performance criterion and may contain one of the following symbols:

- S_{in} – single objective performance criterion.
- M_{op} – multiple objective performance criteria.

Example: Notation $S_{he}:M_{lc}:I/R:S_{il}|U_{nd}:R_{eg}:R_{si}|C_{on}:H_{om}|S_{in}$ refers to a single label, multiple class problem with the hierarchical structure of categories that can be measured using interval/ratio scale. Besides, data are unstructured, with regular distribution and regular size. In addition, data are continuous and come from a homogenous data source. The problem is to optimize a single objective performance criterion.

B. EXAMPLE CASES

To illustrate how the proposed notation can be used to describe the classification problem, several examples are discussed in this subsection.

Example 1. To implement the machine learning system for credit card fraud detection the following arbitrary assumptions have been made:

- The system is expected to decide whether a credit card transaction is fraudulent or not.
- The system will be used by more than one organization.
- Attributes of transactions are nominal, continuous, discrete, and symbolic.
- On the whole, there will be many more non-fraudulent transactions than fraudulent ones.
- Data sources are distributed.
- Transactional data from a stationary data stream.

For the considered case the following notation can be used:

$$S_{he} : B_{in} : I/R : S_{il} / S_{ts} : I_{im} / M_{ix} : D_{dis} : H_{om} : S_{ds} / S_{in}$$

Example 2. In [95] the problem of multi-label stream classification problem was considered. To deal with it a

Multiple Windows (MW) approach with a word bag model and a single performance criterion was proposed. The authors transformed the multi-label problem into multiple binary problems and solved each problem independently. The approach was validated using three large real-world multi-label datasets as shown in Table II.

TABLE II
DATASETS USED IN THE EXPERIMENT REPORTED IN [95]

Dataset	Number of instances	Number of attributes	Number of class labels	Type of attributes	Number of distinct label sets
<i>tmc2007</i>	28596	500	22	binary	1172
<i>imdb</i>	120919	1001	28	binary	4503
<i>rcv1v2</i>	804414	500	103	numeric	13922

Classification problems solved in [95] can be denoted, using the proposed notation, as follows:

- *tmc2007*:
 $S_{he} : M_{lc} : M_{ml} : I/R : S_{mv} : I_{im} : D_{is} : C_{en} : H_{om} : S_{tat} : S_{in}$
- *imdb*:
 $S_{he} : M_{lc} : M_{ml} : I/R : S_{mv} : I_{im} : D_{is} : C_{en} : H_{om} : S_{tat} : S_{in}$
- *rcv1v2*:
 $S_{he} : M_{lc} : M_{ml} : I/R : S_{mv} : I_{im} : C_{on} : C_{en} : H_{om} : S_{tat} : S_{in}$

Example 3. In [93], the collective intelligence system called RealTravel is discussed. The system has been designed to work in an environment where:

- Data are distributed.
- Data are represented by different types, i.e. they are text, numbers, photos, etc., i.e. are semi-structured, mixed, and multidimensional.
- The system generates multi-class recommendations.
- Evaluation of the recommendation quality is carried-out based on a multi-objective approach.

For the above case the notation can be as follows:

$$S_{he} : M_{lc} : U_{ud} : M_{ix} : U_{mu} : D_{dis} : M_{op}$$

VI. THE RESEARCH EFFORT

To evaluate the research effort spent on the development of models and tools designed for solving various types of classification problems we show in Fig. 6 the number of publications and their h-indexes as provided by Web of Science and Scopus. From Fig. 7 it appears that binary classification and multi-class classification problems have been studied most intensively among all classification problems.

VII. CONCLUSION

Two main contributions of the paper include:

- Offering a review of the current research effort in the field of supervised learning covering various types of classification problems tackled in the relevant literature.
- Proposing an original taxonomy for categorizing main dimensions of the supervised learning classification problems ordered by a category view, data view, attribute view, data source view, and performance criteria view.
- Proposing a simple notation for identification of the supervised learning classification problem categories.

The proposed taxonomy is based on the analysis of factors relevant for constructing and solving the supervised learning classification problems. The analysis of the machine learning publications has enabled compiling the landscape of decision problem factors influencing the supervised learning processes. The proposed notation offers a concise and coherent way to describe various supervised learning classification problem types and their critical features.

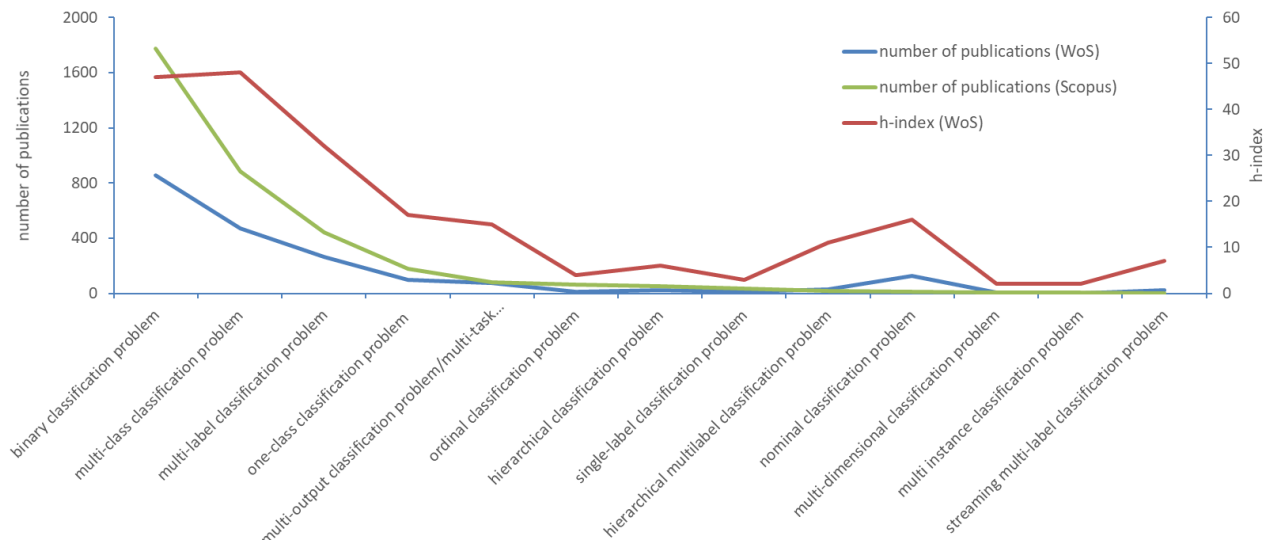


FIGURE 7. The number of publications and h-index for papers dedicated to different kinds of classification problems as of October 2022.

The ultimate goal of both - the proposed taxonomy and the notation, is to provide those interested in supervised learning with a simple way to identify main factors that have to be considered when looking for a method and a tool for solving the particular supervised classification problem. The proposed notation is open and can be further extended taking into account new methods and techniques. It could be also a starting point for constructing a decision support system or recommender able to help a layman in the machine learning field to select the proper method or tool for solving his problems. Constructing such a system will be the focus of future research.

REFERENCES

- [1] D. Morgan, G. Ceder, "Data Mining in Materials Development," in: Handbook of Materials Modeling: Methods, S. Yip Ed., Springer Netherlands, Dordrecht, 2005, pp. 395–421, DOI: 10.1007/978-1-4020-3286-8_19.
- [2] A. Tharwat, "Classification assessment methods," *Applied Computing and Informatics*, vol. 17, no. 1, pp. 168–192, 2021. DOI:10.1016/j.aci.2018.08.003.
- [3] R. A. Fisher, "The use of multiple measurements in taxonomic problems". *Annals of Eugenics*, vol. 7, no. 2, pp. 179–188, 1936. DOI:10.1111/j.1469-1809.1936.tb02137.x.
- [4] E. Alpaydin, *Introduction to Machine Learning*. MIT Press. 2010, pp. 5-9.
- [5] J. Han, and M. Kamber, *Data Mining. Concepts and Techniques*. Academic Press. San Diego. 2001, pp. 327-303.
- [6] R.C. Gonzalez, and R.E. Woods, *Digital Image Preprocessing*. Addison-Wesley, Reading, 1992, pp. 883-887.
- [7] M. Roach, J. Mason, N. Evans, L-Q. Xu, and F. Stentiford, "Recent trends in video analysis: A taxonomy of video classification problems," in *Proceedings of the International Conference on Internet and Multimedia Systems and Applications, IASTED*, Cambridge, Massachusetts, USA, 2002, pp. 360-106.
- [8] J.R. Parker, *Algorithms for Image Preprocessing and Computer Vision*. Wiley, 1996, pp. 285-320.
- [9] G. Zhang, H. Rong, L. Hu, and W. Jin, "Application of radial basis probability neural network to signal recognition," in *Proceedings of Fifth World Congress on Intelligent Control and Automation*, 2004, vol. 3, no. 15-19 June 2004, pp. 1950–1953.
- [10] S. Greco, B. Matarazzo, and R. Slowinski, "The use of rough sets and fuzzy sets in MCDM," in *Advances in Multiple Criteria Decision Making*, vol. 14, T.Gal, T.Stewart, T.Hanne Eds., Kluwer, 1999, pp. 14.1-14.59.
- [11] T. Mitchell, *Machine learning*. McGraw-Hill, New York, 1997, pp. 20-51.
- [12] M. H. ur Rehman, C.S. Liew, A. Abbas, P.P. Jayaraman, T.Y. Wah, and S. U. Khan, "Big Data reduction methods: A survey," *Data Science Engineering*, vol. 1, pp. 265-284, 2016.
- [13] N. García-Pedrajas, and A. de Haro-García, "Scaling up data mining algorithms: review and taxonomy," *Progress in Artificial Intelligence*, vol. 1, no. 1, pp. 71–87, 2012.
- [14] "Cambridge Dictionary," Dec. 12, 2020 [Online] Available: <https://dictionary.cambridge.org/dictionary/english/classification>.
- [15] R. S. Michalski, "On the quasi-minimal solution of the general covering problem," in *Proceedings of the V international symposium on information processing (FCIP 69) (Switching Circuits)*, vol. A3. Yugoslavia, Bled, 1969, pp. 125–128.
- [16] R. Quinlan, "Induction of decision trees," *Machine Learning*, vol. 1, 1986, pp. 81–106.
- [17] P. Clark and R. Boswell, "Rule induction with CN2: Some recent improvements," in *Proceedings of the 5th European Working Session on Learning (EWSL-91)*, 1991, pp. 151–163.
- [18] K. David, W. Aha, D. Kibler, and Marc K. Albert, "Instance-based learning algorithms," *Machine learning*, vol. 6, no. 1, pp. 37–66, 1991.
- [19] V. N. Vapnik, B. E. Boser, and I. Guyon, "A training algorithm for optimal margin classifiers," in *COLT 92 Proceedings of the fifth annual workshop on Computational learning*, Pittsburgh, PA, USA, vol. 5, ACM New York, 1992, , pp. 144-152.
- [20] L. Devroye, L. Györfi and G. Lugosi, *A probabilistic theory of pattern recognition*, Springer, 1996, pp. 9-20.
- [21] L. Berman, "Bagging Predictors," *Machine Learning*, vol. 24, 1996, pp. 123-140.
- [22] J. Stefanowski, "The rough set based rule induction technique for classification problems," w: *Proceedings of 6th European Conference on Intelligent Techniques and Soft Computing*, Aachen, EUFIT 98, 1998, pp. 109-113.
- [23] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, 2001, pp. 5–32. DOI: 10.1023/A:1010933404324.
- [24] Y. Show-Jane, and L. Yue-Shi, "Cluster-based under-sampling approaches for imbalanced data distributions," *Expert Systems with Applications*, vol. 36, pp. 5718–5727, 2009.
- [25] R. Wang, S. Kwong, X. Wang, and J. Yuheng, "Active k-label sets ensemble for multi-label classification," *Pattern Recognition*, vol. 109, 107583, 2021.
- [26] D. Tax, "One-class classification: Concept-learning in the absence of counter-examples," Ph.D. dissertation, The Netherlands: University of Delft, 2001.
- [27] M. Moya, M. Koch, and L. Hostetler, "One-class classifier networks for target recognition applications," in *Proceedings World Congress on Neural Networks*, 1993, Portland, OR. International Neural Network Society, INNS, pp. 797–801, 1993.
- [28] G. Ritter, and M. Gallegos, "Outliers in statistical pat-tern recognition and an application to automatic chromosome classification," *Pattern Recognition Letters*, vol. 18, pp. 525–539, 1997.
- [29] C. Bishop, "Novelty detection and neural network validation," *IEE Proceedings on Vision, Image and Signal Processing. Special Issue on Applications of Neural Networks*, vol. 141, no. 4, pp.217–222, 1994.
- [30] N. Japkowicz, "Concept-Learning in the absence of counter-examples: An autoassociation-based approach to classification," Ph.D. dissertation, New Brunswick Rutgers, The State University of New Jersey, 1999.
- [31] S.S. Khan, and M. Madden, "One-Class classification: Taxonomy of study and review of techniques," *The Knowledge Engineering Review*, vol. 29, no. 3, pp. 345–374, 2014.
- [32] X. Li, and B. Liu, "Learning to classify texts using positive and unlabeled data," in *18th International Joint Conf. on Artificial Intelligence (IJCAI-03)*, pp. 87–594, 2003.
- [33] I. Ali, S. Saad, and S. Ahmed, "Using one-class SVM with spam classification," *Iraqi Journal of Science*, vol. 57, no.1B, pp: 501-506, 2016.
- [34] B. Das, D.J. Cook, N.C. Krishnan, and M. Schmitter-Edgecombe, "One-class classification-based real-time activity error detection in smart homes," *IEEE Journal of Selected Topics in Signal Processing*, vol. 10, no. 5, pp. 914 – 923, 2016.
- [35] L. Mena, and J. a. Gonzalez, "Symbolic one-class learning from imbalanced datasets: application in medical diagnosis," *International Journal on Artificial Intelligence Tools*, vol. 18, no. 2, pp.273-309, 2009.
- [36] A. Vashighzaker, A. Sharma, and A. Dehzangi, "A novel one-class classification approach to accurately predict disease-gene association in acute myeloid leukemia cancer," *PLOS ONE*, vol. 14, no. 12, e0226115, 2019. DOI:10.1371/journal.pone.0226115.
- [37] R. O. Duda, P.E. Hart, and D.G. Stork, "Pattern Classification," John Wiley & Sons Ltd. 2001, pp. 157-158.
- [38] C. Cortes, and V. Vapnik, "Support vector networks," *Machine Learning*, vol. 20, pp. 273-297, 1995.
- [39] T. G. Dietterich, G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of Artificial Intelligence Research*, vol. 2, pp.263-286, 1995.
- [40] L. Zhao, S. Ying , Z. Yushi , Z. Cheng, and Z. Yi., "Face recognition based on multi-class SVM," 2009 Chinese Control and Decision Conference, Guilin, 2009, pp. 5871-5873.
- [41] Vi Nguyen Thanh Le, B. Apopei, and K. Alameh, "Effective plant discrimination based on the combination of local binary pattern operators and multiclass support vector machine methods," *Information Processing in Agriculture*, vol. 6, no. 1, pp. 116-131, 2019.
- [42] M. Koklu, I.A. Ozkan, „Multiclass classification of dry beans using computer vision and machine learning techniques," *Computers and Electronics in Agriculture*, vol. 174, no. 105507, 2020.
- [43] Z. Yan, W. Liu, S. Wen, and Y. Yang, "Multi-label image classification by feature attention network," *IEEE Access*, vol. 7, pp. 98005 – 98013, 2019.
- [44] S. Raudys and V. Pikelis, "On dimensionality, sample size, classification error, and complexity of classification algorithm, in pattern recognition," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 2, no. 3, pp. 243-252, 1980.

- [45] J. Liu, Y. Liu, Y. Wang, V. Prinet, S. Xiang, and C. Pan, "Decoupled representation learning for skeleton-based gesture recognition," 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, pp. 5750-5759.
- [46] D. Xu, Y. Shi, I. W. Tsang, Y. Ong, C. Gong and X. Shen, "Survey on multi-output Learning," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 7, pp. 2409-2429, 2020, DOI: 10.1109/TNNLS.2019.2945133.
- [47] B.-B. Jia, and M.-L. Zhang, "Multi-dimensional classification via kNN feature augmentation," Pattern Recognition, vol. 106, pp. 107423, 2020. DOI:10.1016/j.patcog.2020.107423.
- [48] J. Read, C. Bielza, and P.L. Naga, "Multi-dimensional classification with super-classes," IEEE Transaction on Knowledge Data Engineering, vol. 26, no. 7, pp. 1720-1733, 2014.
- [49] J. Read, A. Bifet, G. Holmes, and B. Pfahringer, "Streaming multi-label classification," in JMLR: Workshop and Conference Proceedings, vol. 17, pp. 19-2, 2011.
- [50] X. Xu, "Statistical learning in multiple instance problems," Master's thesis. University of Waikato, 2003.
- [51] J. Amores, "Multiple instance classification: Review, taxonomy and comparative study," Artificial Intelligence, vol. 201, pp. 81-105, 2013.
- [52] T. M. Cover, and P.E. Hart, "Nearest neighbor pattern classification," IEEE Transaction on Information Theory, vol. 13, pp. 21-7, 1967.
- [53] Z. Wang, L. Liu, and D. Tao, "Deep streaming label learning," in Proceedings of the 37th International Conference on Machine Learning, PMLR 119, Vienna, Austria, 2020, pp. 9963-9972.
- [54] X. Zheng, P. Li, Z. Chu, and X. Hu, "A Survey on multi-label data stream classification," IEEE Access, vol. 8, pp. 1249-1275, 2020.
- [55] D. Xu, Y. Shi, I. Tsang, Y. Ong, C. Gong, and X. Shen, "Survey on multi-output learning," IEEE Transactions on Neural Networks and Learning Systems, vol. 31, no. 7, pp. 2409-2429, 2020.
- [56] N. Mukund, S. Thakur, S. Abraham, A. K. Aniyar, S. Mitra, N. S. Philip, K. Vaghmare, and D. P. Acharya, "An information retrieval and recommendation system for astronomical observatories," The Astrophysical Journal Supplement Series, vol. 235, no. 1, pp. 1-9, 2018.
- [57] Y. Lin, Z. Jun, M. Hongyan, Z. Zhongwei, and F. Zhanfang, "A method of extracting the semi-structured data implication rules," Procedia Computer Science, vol. 131, pp. 706-716, 2018.
- [58] D. Parochia, "Classification," The Internet Encyclopedia of Philosophy, ISSN 2161-0002, [Online] Available: <https://www.icp.utm.edu/>
- [59] G. M. Weiss, F. Provost, "The effect of class distribution on classifier learning: An empirical study," Technical Report ML-TR-44, Department of Computer Science, Rutgers University, 2001.
- [60] H. He, and E.A. Garcia, "Learning from Imbalanced Data," IEEE Transactions on Knowledge and Data Engineering, vol. 21, no. 9, pp. 1263-1284, 2009.
- [61] I.H. Witten, and E. Frank, Data Mining: Practical Machine Learning Tools and Techniques. 2nd ed., Morgan Kaufman, San Francisco, California, USA, 2005, pp. 39-60.
- [62] J. W. Grzymala-Busse, and M. Hu, "A comparison of several approaches to missing attribute values in data mining," in RSCTC 2000, W. Ziarko and Y. Yao, Eds., LNAI 2005, Springer-Verlag Berlin Heidelberg, 2001, pp. 378-385.
- [63] Kubat, M.: Tree Structures of Linear Threshold Units for the Classification of Numeric Examples. Cybernetics and Systems, vol. 26, no. 5, 1995, pp. 521-533.
- [64] A. Piatti, "Learning under prior ignorance," Ph.D. dissertation, University of Logano, 2006.
- [65] S. Cost, and S. Salzberg, "A weighted nearest neighbor algorithm for learning with symbolic features," Machine Learning, vol. 10, pp. 57-78, 1993.
- [66] S. Trevisani, M. Cavalli, J. Golay, et al., "Learning from spatial data: unveiling the geo-environment through quantitative approaches," Environmental Earth Sciences, vol. 78, pp. 127, 2019.
- [67] L. Xiong, "On Learning from collective data," Ph.D. dissertation, Carnegie Mellon University, 2013.
- [68] H. Julkunen, "Predictive modelling of anticancer efficacy of drug combinations using factorization machines," M.Sc. Thesis, Alto University, Finland, 2019.
- [69] N. Qomariyah, and D. Kazakov, "Learning from ordinal data with inductive logic programming in description logic," Proceedings of the 27th International Conference on Inductive Logic Programming, in Lachiche, Nicolas and Vrain, Christel, (eds.), 2017, pp. 38-50.
- [70] T.R. Hoens, R. Polikar, N.V. Chawla, "Learning from streaming data with concept drift and imbalance: an overview," Progress in Artificial Intelligence, vol. 1, no. 1, pp. 89-101, 2012.
- [71] D. Caragea, A. Silvescu, and V. Honavar, "Agents that learn from distributed dynamic data sources," in: ECML 2000/Agents 2000 Workshop on Learning Agents, Barcelona, Spain, 2000, pp. 51-61.
- [72] Z. Sahel, A. Bouchachia, B. Gabrys, and P. Rogers, "Adaptive mechanisms for classification problems with drifting data," in KES 2007, B. Apolloni, et al., Eds., LNAI 4693, Springer-Verlag, 2007, pp. 419-426.
- [73] G. Widmer, and M. Kubat, "Learning in the presence of concept drift and hidden contexts," Machine Learning, vol. 23, no. 1, pp. 69-101, 1996.
- [74] L. Xu, A. Krzyzak, and C.Y. Suen, "Methods of combining multiple classifiers and their application to handwriting recognition," IEEE Transaction on Systems, Man and Cybernetics, vol. 22, pp. 418-435, 1992.
- [75] X. Zhu, P. Zhang, X. Lin, and Y. Shi, "Active learning from data streams," in Seventh IEEE International Conference on Data Mining, 2007, pp. 757-762.
- [76] D. Brzezinski, and J. Stefanowski, "Stream classification," in Encyclopedia of Machine Learning and Data Mining, C. Sammut, and G. I. Webb, Eds., Springer, Boston, MA, 2017, https://doi.org/10.1007/978-1-4899-7687-1_908
- [77] G. Widmer, and M. Kubat, "Learning in the presence of concept drift and hidden contexts," Machine Learning, vol. 23, pp. 69-101, 1996.
- [78] S. Zhang, X. Wu, C. Zhang, "Multi-database mining," IEEE Computational Intelligence Bulletin, vol. 2, no. 1, pp. 5-13, 2003.
- [79] H. Kargupta, B.H. Park, D. Hersherberger, and E. Johnson, "Collective data mining: A new perspective toward distributed data analysis," in Advances in Distributed Data Mining, H. Kargupta, P. Chan, Eds., AAAI/MIT Press, 1999, pp. 133-184.
- [80] M. Pagels, "Architecting a scalable real time learning system," in Fourkind Website, 2018, Accessed on: 30.10.2020, Available: <https://medium.com/value-stream-design/architecting-a-scalable-real-time-learning-system-95623d27dd15>
- [81] R. Pan, Z. Zhang, Y. Fan, J. Cao, K. Lu, and T. Yang, "Multi-objective optimization method for learning thresholds in a decision-theoretic rough set model," International Journal of Approximate Reasoning, vol. 71, pp. 34-49, 2016.
- [82] X. Jia, W. Liao, Z. Tang, and L. Shang, "Minimum cost attribute reduction in decision-theoretic rough set models," Information Sciences, vol. 219, pp. 151-167, 2013.
- [83] Y. Tian, C. Lu, X. Zhang, K.C. Tan, and Y. Jin, "Solving large-scale multiobjective optimization problems with sparse optimal solutions via unsupervised neural networks," IEEE Transactions on Cybernetics, vol. 51, no. 6, pp. 3115-3128, 2020, DOI:10.1109/tyb.2020.2979930.
- [84] R. Liu, F. Wang, M. He, and L. Jiao, "An adjustable fuzzy classification algorithm using an improved multi-objective genetic strategy based on decomposition for imbalance dataset," Knowledge Information Systems, vol. 61, pp. 1583-1605, 2019. <https://doi.org/10.1007/s10115-019-01342-5>
- [85] E. Bernado-Mansilla, X. Llorca, and I. Traus, "Multi-objective learning classifier systems," in Multi-Objective Machine Learning, Jin Y., Ed., Studies in Computational Intelligence, vol. 16, pp. 261-290. Springer-Verlag, Berlin / Heidelberg, 2006.
- [86] N. Lavesson, P. Davidsson, "Evaluating learning algorithms and classifiers," International Journal of Intelligent Information and Database Systems, vol. 1, no. 1, 37-52, 2007.
- [87] R. L. Graham, E.L. Lawler, J.K. Lenstra, A.H.G. R. Kan, "Optimization and approximation in deterministic sequencing and scheduling: A survey," in Proceedings of the Advanced Research Institute on Discrete Optimization and Systems Applications of the Systems Science Panel of NATO and of the Discrete Optimization Symposium. Elsevier, vol. 5, 1979, pp. 287-326.
- [88] N. Trask, R. G. Patel, B. J. Gross, and P. J. Atzberger, "GMLS-Nets: A framework for learning from unstructured data," Mathematics, Computer Science, Physics, preprint ArXiv (abs/1909.05371), 2019.
- [89] K. Hanaoka, "Deep neural networks for multicomponent molecular systems," ACS Omega, vol. 5, pp. 21042-21053, 2020, DOI:10.1021/acsomega.0c02599.
- [90] F. Orsin, "Learning from structured data with kernels, neural networks and logic," Ph.D. dissertation, KU Leuven, 2017.
- [91] H. Fei, "Learning from structured data with high dimensional structured input and output domain," PH.D. dissertation, University of Kansas, 2012.
- [92] T. Karunaratne, H. Boström, "Learning from structured data by finger printing," Publications of the Finnish Artificial Intelligence Society, 2006, p. 120-126.
- [93] T. Gruber, "Collective knowledge systems: Where the social web meets the semantic web," Journal of Web Semantics, vol. 6, no. 1, pp. 4-13, 2008.
- [94] M. Nagamine, T. Miyahara, T. Kuboyama, H. Ueda, and K. Takahashi, "Evolution of multiple tree structured patterns from tree-Structured data using clustering," in Proceedings of the 21st Australasian Joint Conference on

- Artificial Intelligence (AI 2008), W. Wobcke and M. Zhang (Eds.), Auckland, LNAI 5360, Springer-Verlag Berlin Heidelberg 2008, pp. 500–511, 2008.
- [95] E.S. Xioufis, M. Spiliopoulou, G. Tsoumakas, and I. P. Vlahavas, “Dealing with concept drift and class imbalance in multi-label stream classification,” in *Proceedings of the 22nd International Joint Conference on Artificial Intelligence IJCAI 2011*, T. Walsh, Ed., 2011, p. 1583-1588.
- [96] S. Singh, “Multiresolution estimates of classification complexity,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 12, pp. 1534-1539, 2003.
- [97] M. Zhao, and Y. Wu, “Classification complexity and its estimation algorithm for two-class classification problem,” in *Proceedings of the International Joint Conference on Neural Networks (IJCNN'99)*, Washington, DC, USA, 1999, vol.3, pp. 1631-1634, DOI:10.1109/IJCNN.1999.832616.
- [98] T.K. Ho, and M. Basu, “Measuring the complexity of classification problems,” in *Proceedings of the 15th International Conference on Pattern Recognition*, Barcelona, Spain, 2000, pp. 43-47.
- [99] V. Vapnik, *Statistical Learning Theory*, John Wiley & Sons, 1998.
- [100] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer Verlag, New York, 1995, ch. 7, pp. 167-175.
- [101] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. Warmuth, “Learnability and the Vapnik-Chervonenkis dimension,” *Journal of the ACM*, 36(4), pp. 929-965, 1989.
- [102] M. Karpinski, and A. McIntyre, “Polynomial bounds for VC dimension of sigmoidal and general Pfaffian neural networks,” *Journal of Computer and System Science*, vol. 54, pp. 169-176, 1997.
- [103] A.C. Lorena, L. P.F. Garcia, J. Lehmann, M.C.P. Souto, and T.K. Ho, “How complex is your classification problem? A survey on measuring classification complexity,” *arXiv:1808.03591*, 2019.
- [104] T.K. Ho, and M. Basu, “Complexity measures of supervised classification problems,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp.289–300, 2002.
- [105] S.Y. Sohn, “Meta-analysis of classification algorithms for pattern recognition,” *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 1137-1144, Nov. 1999
- [106] X. Qiao, “Noncrossing ordinal classification,” *Statistics and Its Interface*, vol. 10, pp. 187–198, 2017.
- [107] J. Cardoso, and da Costa, J. “Learning to classify ordinal data: the data replication method,” *Journal of Machine Learning Research*, vol. 8, 1393–1429, 2007.
- [108] X. Qiao, “Learning ordinal data,” *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 7, pp. 341–346, 2015.
- [109] P.A. Gutiérrez, and S. García, “Current prospects on ordinal and monotonic classification,” *Progress in Artificial Intelligence*, vol. 5, pp. 171–179, 2016. DOI 10.1007/s13748-016-0088-y
- [110] J. Błaszczyński, R. Slowinski, and J. Stefanowski, “Ordinal classification with monotonicity constraints by variable consistency bagging,” in *Proceedings of 7th International Conference on Rough Sets and Current Trends in Computing, RSCTC 2010*, Warsaw, Poland, 2010, pp. 392–401.
- [111] M. Kilariski, “Nominal classification: a history of its study from the classical period to the present,” Amsterdam & Philadelphia: John Benjamins. Pp. xiv, 405, 2013, pp. 7-58.
- [112] W. Kotłowski, K. Dembczyński, S. Greco, and R. Slowinski, “Stochastic dominance-based rough set model for ordinal classification,” *Information Sciences*, 178(21), pp. 4019-4037, 2008.
- [113] E. Fernandez, J.R. Figueira, J. Navarro, and B. Roy, “ELECTRE TRI-nB: A new multiple criteria ordinal classification method,” *European Journal Of Operational Research*, vol. 263, no. 1, pp. 214-224, 2017.
- [114] C. Zopounidis, and M. Doumpos, “Multicriteria classification and sorting methods: A literature review,” *European Journal Of Operational Research*, vol. 138, no. 2, pp. 229-246, 2002.
- [115] R. Cerri, ACPLF de Carvalho, A.A. Freitas, “Adapting non-hierarchical multilabel classification methods for hierarchical multilabel classification,” *Intelligent Data Analysis*, vol. 15, no. 6, pp. 861-887, 2011.
- [116] A. Kosmopoulos, I. Partalas, E. Gaussier, G. Paliouras, and I. Androutsopoulos, “Evaluation measures for hierarchical classification: a unified view and novel approaches,” *Data Mining And Knowledge Discovery*, vol. 29, no. 3, pp. 820-865, 2015.
- [117] C.N. Silla, and A.A. Freitas, “A survey of hierarchical classification across different application domains,” *Data Mining And Knowledge Discovery*, vol. 22, no. 1-2, pp. 31-72, 2011.
- [118] S. Greco, M. Kadzinski, and R. Slowinski, “Selection of a representative value function in robust multiple criteria sorting,” *Computers & Operations Research*, vol. 38, no. 11, pp. 1620–1637, 2011.
- [119] M. Kadzinski, S. Greco, R. Slowinski, “Selection of a representative value function in robust multiple criteria ranking and choice,” *European Journal of Operational Research*, vol. 217, no. 3, pp. 541–553, 2012.
- [120] R. Cerri, G. Pappa, ACPLF de Carvalho, and A.A. Freitas, “An extensive evaluation of decision tree-based hierarchical multi label classification methods and performance measures,” *Computational Intelligence*, vol. 31, no. 1, pp. 1-45, 2015.
- [121] F. Wu, J. Zhang, and V. Honavar, “Learning classifiers using hierarchically structured class taxonomies,” in *Proceedings of the Symposium on Abstraction, Reformulation, and Approximation*, vol 3607. Springer, Berlin, 2005, pp 313–320.
- [122] J.S. Cardoso, and R. Sousa, “Measuring the performance of ordinal classification,” *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 25, no. 8, pp. 1173-1195, 2011.
- [123] T.K. Ho, and M. Basu, “Complexity measures of supervised classification problems,” *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 24, no. 3, pp. 289-300, 2002.
- [124] S. Sun, “A survey of multi-view machine learning,” *Neural Computation & Applications*, vol. 23, pp.2031-2038, 2013.
- [125] A. Bagnall, J. Lines, A. Bostrom, J. Large, and E. Keogh, “The great time series classification bake off: a review and experimental evaluation of recent algorithmic advances,” *Data Mining and Knowledge Discovery* vol. 31, no. 3, pp. 606–660, 2017.
- [126] H. I. Fawaz, G. Forestier, J. Weber, L. Idoumghar, and P.A. Muller, “Deep learning for time series classification: a review,” *Data Mining and Knowledge Discovery* 33, pp. 917–963, 2019. DOI:10.1007/s10618-019-00619-1.
- [127] M.W. Libbrecht, and W.S. Noble, “Machine learning applications in genetics and genomics,” *Nature Review Genetics*, vol. 16, no. 6, pp. 321-332, 2015.
- [128] K.G. Liakos, P. Busato, D. Moshou, S. Pearson, and D. Bochtis, “Machine Learning in Agriculture: A Review,” *Sensors*. Vol. 18, 2674, 2018, DOI:10.3390/s18082674.
- [129] K.T. Butler, D.W. Davies, H. Cartwright, O. Isayev, and A. Walsh, “Machine learning for molecular and materials science,” *Nature*, vol. 559, pp. 547–555, 2018. DOI:10.1038/s41586-018-0337-2.
- [130] G. Carleo, I. Cirac, K. Cranmer, L. Daudet, M. Schuld, N. Tishby, L. Vogt-Maranto, and L. Zdeborová, “Machine learning and the physical sciences,” *Reviews of Modern Physics*, 91, 045002, 2019, DOI: 10.1103/RevModPhys.91.045002
- [131] S. N. Raghavendra, and P. C. Deka, “Support vector machine applications in the field of hydrology: A review,” *Applied Soft Computing*, vol. 19, pp. 372-386, 2014.
- [132] A.C. Mater, and M. Coote, “Deep Learning in Chemistry,” *Journal Of Chemical Information And Modeling*, vol. 59, no. 6, pp. 2545-2559, 2019.
- [133] A. Dogan, and D. Birant, “Machine learning and data mining in manufacturing,” *Expert Systems with Applications*, vol. 166, 114060, 2021.
- [134] D.G. Shen, G.R. Wu, and H.I. Suk, “Deep Learning in Medical Image Analysis,” *Annual Review of Biomedical Engineering*, vol. 19, no. 1, pp. 221-248, 2017.
- [135] M.J. Warrens, “Kappa coefficients for dichotomous-nominal classifications,” *Advances in Data Analysis and Classification*, vol. 15, pp. 193–208 (2021). DOI:10.1007/s11634-020-00394-8



IRENEUSZ CZARNOWSKI received the PhD degree in the field of computer science from the Poznan University of Technology, Poland. A postdoctoral degree, also in the field of computer science, he received from the Wrocław University of Science and Technology, Poland. He is currently a professor of computer science in the Department of Information Systems, Gdynia Maritime University, Poland. His research interests include artificial intelligence, multi-agent systems, decision support systems, machine learning, artificial neural networks, data

mining, data reduction, instance selection, data preprocessing based on population-based algorithms. He is an author and co-author more than 100 papers in international journals and conference proceedings. Between 2016 and 2020 he was the Vice Rector, Research and International Cooperation of the Gdynia Maritime University. He is associated with IEEE since 2016, currently is the Chair of the Polish branch of IEEE System, Man and Cybernetics.



PIOTR JĘDRZEJOWICZ is a professor of information systems and head of the Information Systems Department, Gdynia Maritime University. His research interests include artificial intelligence, machine learning, agent technology, combinatorial optimization and decision support systems. During his career Prof. Jędrzejowicz has been a visiting professor in Germany, Great Britain, China, Sweden, and a research fellow at the School of Computer Science, McGill University, Montreal. He has been leading

several research projects and serving as a general chair, program chair and program committee member in numerous international scientific conferences. He has been invited to edit several special numbers of the international scientific journals.