Journal of Intelligent Learning Systems and Applications, 2025, 17(3), 126-132
https://www.scirp.org/journal/jilsa
ISSN Online: 2150-8410
ISSN Print: 2150-8402

Optimization of Data Structure in Classification and Clustering Problems

Vladimir N. Shats

Independent Researcher, St. Petersburg, Russia
Email: vlshats@hotmail.com

How to cite this paper: Shats, V.N. (2025) Optimization of Data Structure in Classification and Clustering Problems. *Journal of Intelligent Learning Systems and Applications*, **17**, 126-132.
https://doi.org/10.4236/jilsa.2025.173009

Received: April 28, 2025
Accepted: July 1, 2025
Published: July 4, 2025

Abstract

The paper is devoted to the optimization of data structure in classification and clustering problems by mapping the original data onto a set of ordered feature vectors. When ordering, the elements of each feature vector receive new numbers such that their values are arranged in non-decreasing order. For update structure, the main volume of computational operations is performed not on multidimensional quantities describing objects, but on one-dimensional ones, which are the values of objects individual features. Then, instead of a rather complex existing algorithm, the same simplest algorithm is repeatedly used. Transition from original to ordered data leads to a decrease in the entropy of data distribution, which allows us to reveal their properties. It was shown that the classes differ in the functions of feature values for ordered object numbers. The set of these functions displays the information contained in the training sample and allows one to calculate class of any object in the test sample by values of its features using the simplest total probability formula. The paper also discusses the issues of using ordered data matrix to solve problems of partitioning a set into clusters of objects that have common properties.

Keywords

Feature Vector Ordering, Functional Dependencies of Features and Classes, Objects Closeness Concept

1. Introduction

The paper proposes a new computational technology for solving of classification problem based on a new concept of object similarity, which is one of the fundamental concepts in machine learning, because it allows one to compare subsets of data in order to recognize objects of different classes [1]. Usually, the similarity of objects is assessed by the distance between them in metric space. Here, objects of

a finite set of the same class are considered to be close in the value of a certain feature if these values are close enough.

According to this concept, the center of computational procedures is not the object as an element of a multidimensional feature space, but the object feature value as an element of each feature vector. Therefore, the majority of calculations are performed for one-dimensional rather than multidimensional functions, which leads to a qualitative simplification of algorithm.

We considered several options for implementing this approach, which differed in the way of transforming structure of data matrix. Of greatest interest is the method that boils down to splitting the values of each feature of the objects in the combined sample (consisting of training and test samples) into the same number of intervals, which play the role of calculated parameters [2]. Lists of the TS objects of the same class falling within these intervals were considered as information granules [3]. Then, the frequency of any feature value in a certain class is equal to the frequency of corresponding granule, the frequency of an object in each class is equal to average frequency of all its features in each class, and the class of an object corresponds to maximum of these frequencies.

Let us note that in [4], it was shown that the above classification algorithm according to the mechanism for processing information received from the environment by receptors of various sensory systems of a mammal. The totality of this data is supplemented by previously obtained information and is generalized in the brain only at the last stage. Thus, the approach being considered is bio-inspired. In this paper, a new version of this approach was developed [5] based on the use of ordered data.

2. Ordering of Feature Vectors

2.1. Some Properties of Ordered Features

Let us consider the training sample (TS) of a classification problem. Let $G = \|x_{sk}\|_{M\times N}$ is quantitative data matrix the TS, $s = 1,\cdots,M$ are the numbers of objects and $X^k = \left(x_{1k},\cdots,x_{Mk}\right)^{\mathrm{T}}$ is the feature vector $k = 1,\cdots,N$. We will call the elements set of the vector X^k ordered if they were renumbered and received new numbers $(s)^k = (1)^k,(2)^k,\cdots,(M)^k$ such that the corresponding values of this vector form a non-decreasing sequence $x_{(1)^k} \le x_{(2)^k} \le \cdots \le x_{(M)^k}$. We will extend the term "ordered" to similar sets of quantities [6].

By definition, ordered elements of a vector are the nearest neighbors by feature value. Note that nearest neighbor methods [7] are widely used in solving classification problems. However, these methods consider the issues of objects features distribution in metric space, and not changes in the data structure as in the present article.

The numbering of elements of the ordered vector X^k has important peculiarities. Obviously, if the feature value $x_{s2k} > x_{s1k}$ of objects $s1$ and $s2$, then the ordered numbers of objects $(s2)^k > (s1)^k$, and this relationship is preserved for objects of the same class. Therefore, objects of a certain class can be identified

not only by their ordered numbers, but also by the sequence of these numbers $(s)_i^k = 1, 2, \cdots, l_i$. Here l_i is the length of class $i = 1, \cdots, C$, and the number $(s)_i^k = 1$ corresponds to the minimum value $(s1)^k$ for objects of this class.

Let us illustrate the peculiarities of numbering using the example of vector X^3 for the case $C = 2$:

$$X^3 = (0.23, 0.11, 0.73, 0.05, 0.42, 0.421, 0.065)^{\mathrm{T}}.$$

Here the set $\{(s^3)\} = \{4^3, 7^3, 2^3, 1^3, 5^3, 6^3, 3^3\}$ is the union of objects subsets $\{(s^3)\} = \{7^3, 5^3\}$ and $\{(s^3)\} = \{4^3, 2^3, 1^3, 6^3, 3^3\}$ for class $i = 1$ and $i = 2$, respectively. In ordinal scales these subsets have the form $\{(\tilde{s})_1^3\} = 1, 2$ and $(\tilde{s})_2^3 = 1, 2, 3, 4, 5$. Then the vectors $(0.065, 0.42)^{\mathrm{T}}$ and $(0.05, 0.11, 0.23, 0.421, 0.73)^{\mathrm{T}}$ will describe in these scales the classes objects features of $i = 1$ and $i = 2$, respectively.

Note that in the case $x_{s2k} = x_{s1k}$ we get an ambiguous relation $(s2)^k = (s1)^k \pm 1$. But this circumstance will not affect subsequent results, since both object numbers correspond to the same feature value.

As shown above, for any k the values $x_{(s)^k}$ form a non-decreasing sequence on the set of points $\{(\tilde{s})_i^k\}$. In other words, on the specified set there is defined a deterministic function $f((s)_i^k)$ such that $x_{(s)^k} = f((\tilde{s})_i^k)$. This function describes the relationships between classes and features of the TS.

However for objects of the same class, the values distribution of each feature on the set $\{s\}$ has many jumps that are close in magnitude to the range of the feature values. Therefore, for the original data there is no functional dependence between classes and features. For ordered values, this distribution will be quite smooth, since the nearest neighbors by the feature value are arranged on the set $\{\tilde{s}\}$. It can be considered that due to the ordering the complex chaotic relationship between classes and feature values for the same class objects is transformed into the deterministic function $f((\tilde{s})_i^k)$. This conclusion means that the reduction of the uncertainty level of information and, accordingly, information entropy of data contained in the TS is achieved by ordering the features values.

Updated data matrix by structuring will be a set of N ordered feature vectors. Compared to the original one, the new structure has an important advantage in relation to solving the classification problem, since functions $f((\tilde{s})_i^k)$ are defined on the set of its ordered features, which significantly simplify, as will be shown below, the algorithm for solving the problem. Note that these functions exist for any TS, since their derivation did not require the introduction of any assumptions or restrictions. Moreover, structuring is reduced to the simplest sorting of the values of individual TS features.

The new structure can be viewed as an independent version of the given data matrix. Apparently, for some databases and solution methods this version of the matrix may be preferable to the original one. Next, we will limit ourselves to solving the classification problem for the updated structure.

2.2. Classification of Structured Data

In the example above, the characteristics of the vector X^3 are specified for objects of individual classes of the TS, which can be visualized on a plane by constructing diagrams, the horizontal axis of which corresponds to the values $(\tilde{s})_i^3 = 1, 2, \cdots$ and the vertical axis to the values $x_{(\tilde{s})_i^3}$. To visualize the information contained in the TS, we consider similar scatter plots of the function vector $f\left((\tilde{s})_i^k\right)$ for some k.

Such diagrams are presented in the panels of Figure 1 for the Wine database for $k = 8$ (left) and $k = 2$ (right). Here the maximum value of $(\tilde{s})_i^k$ is equal to the maximum value of l_i, since each panel contains points for objects of classes $i = 1, 2, 3$. For clarity, the diagrams are constructed for normalized values $z_{(s)^k} \in (0,1)$, calculated using the formula $z_{(s)^k} = \dfrac{x_{(s)^k} - \left(x_{(s)^k}\right)_{\min}}{\left(x_{(s)^k}\right)_{\max} - \left(x_{(s)^k}\right)_{\min}}$, where the subscripts min and max correspond to the minimum and maximum of the feature values k. Further we will assume that all features are normalized.

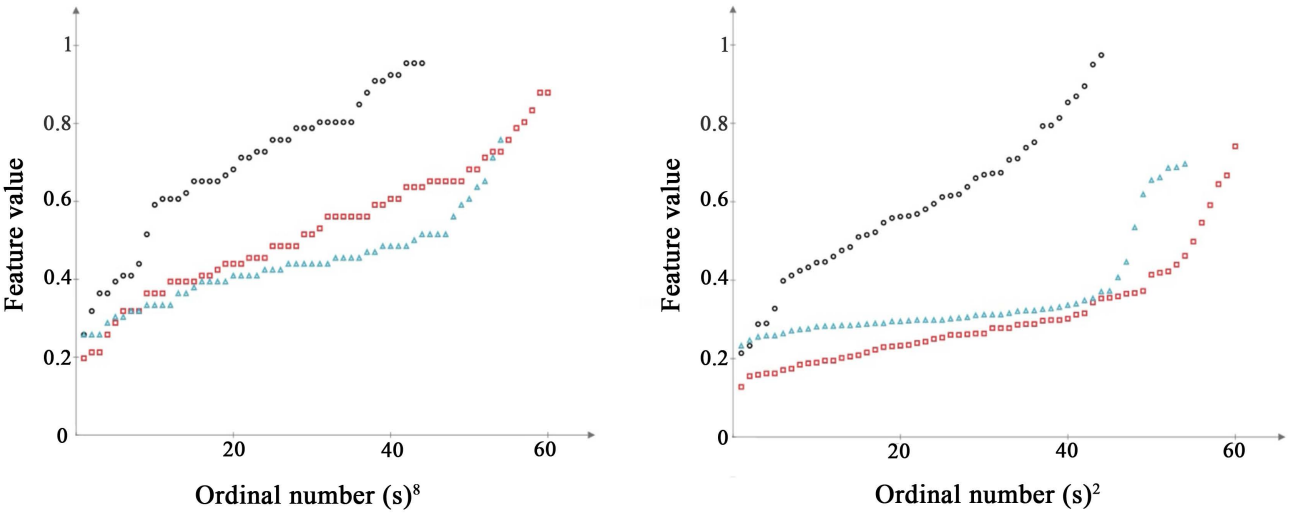

Figure 1. Graphs of functions $f\left((s)_i^k\right)$ for the Wine database for $k = 8$ (left) and $k = 2$ (right).

The diagrams display that the values of the corresponding feature of the same class objects are represented by their own chain of points, which, with some exceptions, are quite far from the points corresponding to other classes. These points are the closest neighbors in terms of the feature value. Therefore, the following classification algorithm based on the new concept of similarity is proposed.

Let $\left\{Z^k \mid k = 1, \cdots, N\right\}$ be the vectors set of objects normalized feature values in the test sample, $Z^k = \left\{z_{tk} \mid t = 1, \cdots, L\right\}$. For each object t of this set, we find the value $q(t,k)$ equal to the class i of the TS object, the feature value k of which $x_{(s)^k}$ is in the h-neighborhood of the value z_{tk}. Here h is the proximity parameter, which characterizes the acceptable level of accuracy in the problem under

consideration and satisfies the condition

$$\left| z_{tk} - f\left(\left(\tilde{s} \right)_i^k \right) \right| \le h \tag{1}$$

Then the average frequency of class i of object t over all features is equal to

$$\gamma(t,i) = \frac{1}{N} \sum_{k=1}^{N} q(t,k) \tag{2}$$

The maximum the frequency determines the object class t

$$i(t) = \arg\max_{1 \le i \le C} \gamma(t,i). \tag{3}$$

Calculations performed for the Iris and Wine [8] databases showed that the number of classification errors for the test sample for $0 \le h < 0.15$ ranges from 0 to 2.

Let us note that for the objects features of any class of the combined sample, inequality (1) is fulfilled randomly, in particular, due to the error of observations and data measurements. Considering that the discrete function $f\left(\left(\tilde{s} \right)_i^k \right)$ is monotone, we can quite simply reduce the influence of these errors and, accordingly, increase the accuracy of the solution to the problem if we transform it into a continuous one by using interpolation or approximation. However, issues of improving the proposed algorithm are beyond the scope of this article.

3. Properties of an Ordered Data Matrix

Consider the ordering effect for a data matrix. It is obvious that the arrangement of elements $x_{(s)^k}$ along the length of the TS depends on the distribution X^k values. At the same time, the sets of these elements must describe the given objects represented by rows of the data matrix G. Therefore, the ordering of the entire set of the TS data is carried out for each of the features separately.

Then the data matrix is mapped onto a set of N data matrices $G_k = \left\| x_{\left(s^k\right)} \right\|_{M*N}$.

Columns of matrices G_k and G, corresponding to the same features, will consist of the same elements and differ only in the order of arrangement of these elements. The rows of such matrices differ only in the order of their arrangement, since they represent feature values sets of individual objects.

Example of data matrices G, G_1, G_2 and G_3.

$$G = \begin{Vmatrix} 3 & 2.1 & 5 \\ 4 & 0.7 & 1 \\ 2 & 0.9 & 6 \end{Vmatrix}, \quad G_1 = \begin{Vmatrix} 2 & 0.9 & 6 \\ 3 & 2.1 & 5 \\ 4 & 0.7 & 1 \end{Vmatrix}, \quad G_2 = \begin{Vmatrix} 4 & 0.7 & 1 \\ 2 & 0.9 & 6 \\ 3 & 2.1 & 5 \end{Vmatrix}, \quad G_3 = \begin{Vmatrix} 4 & 0.7 & 1 \\ 3 & 2.1 & 5 \\ 2 & 0.9 & 6 \end{Vmatrix}.$$

Let class i of the TS object number s, equal to the row number of the data matrix G, be given by the dependence $i = g(s)$. We will consider the properties of objects subsets, called clusters i, whose features are described by row the (s) of the matrix G_k when $i = g\left((s)^k \right)$. Notice that clusters i and classes i have the same length. To assess the objects coincidence level of class i and clusters i, we find the average number of objects of the TS for all classes for which the

dependence is satisfied

$$g(s) = g\left((s)^k\right), k = 1, \cdots, N. \quad (4)$$

The number of such coincidences, divided by the set length, was called the coincidence index $\psi_k \in (0,1)$ of the feature k.

Index analysis was performed for 10 databases [9]. Calculations showed that for a third of the databases considered, the maximum index value exceeds 0.9, 0.7 or 0.5, for one of the databases it reaches 0.961, and for another $\psi_k \sim 0$ for all k. From the results obtained, it follows that classes i and clusters i split many objects into subsets, which partially (in many cases) or almost completely (in some cases) consists of the same objects.

The result obtained is unexpected, but to a certain extent corresponds to ideas regarding the role of order in nature and allows us to penetrate deeper into the essence of the concept class of set [10]. As you know, classes are subsets of objects have common properties that differ for different subsets. The division of a set into classes is performed by specialists in a certain field of knowledge based on the analysis of any common properties of objects, for example, those related to cost, health, quality of products or services. But, when calculating the index, we do not take into account the specifics of these properties.

This conclusion indicates that structuring by ordering features allows us to identify the relationship between classes and features in the clustering problem. A similar relationship in the form of a functional dependence was established in the previous section to solve the classification problem.

For each i, dependence (4) determines the objects numbers s of class i, as well as cluster i. Their feature description for each k is represented by row (s) of the matrix G_k. Thus, all objects of cluster i are nearest neighbors by feature k and, according to the similarity hypothesis, will have common properties by this feature. Since such a situation will occur for all k, then object (s) has certain properties that distinguish it from objects in other clusters. It can be assumed that these properties will be close to the properties of class i objects. Note that the wide range of index values ψ_k is partly caused by measurement errors and selection of features characterizing the properties of the class objects.

4. Conclusions

The paper develops a new concept for solving problems of classification and clustering, based on transforming the structure of the original data by ordering feature vectors. This concept is bio-inspired.

It has been established that the ordering of features leads to a decrease in the entropy of features distribution which allows us to detect functional dependencies of object classes on the features values. When they are used, the algorithm for solving the classification problem is qualitatively simplified.

The updated data structure can serve as the basis for a new type of neural net-

works, in which the functional dependencies obtained in the article are used to simplify and speed up training.

It is shown that by ordering the features, one can find a large number of options for partitioning the set into clusters that are close to the corresponding classes in the composition of objects.

Conflicts of Interest

The author declares no conflicts of interest regarding the publication of this paper.

References

[1] Luger, G.F. (2016) Artificial Intelligence: Structures and Strategies for Complex Problem Solving. 6th Edition, Addison-Wesley.

[2] Shats, V. (2022) Properties of the Ordered Feature Values as a Classifier Basis. *Cybernetics and Physics*, **11**, 25-29. https://doi.org/10.35470/2226-4116-2022-11-1-25-29

[3] Yao, J.T., Vasilakos, A.V. and Pedrycz, W. (2013) Granular Computing: Perspectives and Challenges. *IEEE Transactions on Cybernetics*, **43**, 1977-1989. https://doi.org/10.1109/tsmcc.2012.2236648

[4] Shats, V.N. (2017) The Classification of Objects Based on a Model of Perception. In: Kryzhanovsky, B., *et al.*, Eds., *Advances in Neural Computation*, *Machine Learning*, *and Cognitive Research*, *Studies in Computational Intelligence*, Springer International Publishing, 125-131. https://doi.org/10.1007/978-3-319-66604-4_19

[5] Shats, V.N. (2024) Feature Ordering as a Way to Reduce the Entropy of the Training Sample and the Basis of the Simplest Classification Algorithms. *Proceeding* 26*th International Conference Neuroinformatics*, Moskow, 24-26 October 2024, 164-173.

[6] David, H.A. and Nagaraja, H.N. (2003) Order Statistics. 3rd Edition, Wiley. https://doi.org/10.1002/0471722162

[7] Hastie, T., Tibshirani, R. and Friedman, R. (2009) The Elements of Statistical Learning: Data Mining, Inference, and Prediction. 2nd Edition, Springer, 764.

[8] Asuncion, A. and Newman, D. (2007) UCI Machine Learning Repository. Irvine University of California.

[9] Shats, V.N. (2023) Principle Splitting of Finite Set in Classification Problem. *Proceeding* 25*th International Conference Neuroinformatics*, Moskow, 23-27 October 2023, 262-270.

[10] Prigogine, I. and Stengers, I. (1984) Order out of Chaos: Men's New Dialogue with Nature. Flamingo Edition.