Continuous Attributes for FCA-based Machine Learning[⋆]

Dmitry V. Vinogradov[1,2][0000−0001−5761−4706]

[1] FRC Computer Science and Control RAS, Moscow 119333, Russia
vinogradov.d.w@gmail.com
[2] Russian State University for Humanities, Moscow 125993, Russia
http://isdwiki.rsuh.ru

Abstract. In this paper we extend previously developed approach to FCA-based machine learning with discrete attributes to the case with objects described by continuous attributes. We combine the logistic regression with an entropy-based separation of attribute values, which is similar to Quinlan's approach to dealing with continuous attributes. We apply Cox-Snell and McFadden significance criteria to logistic regression. Finally, we present the results of applying the new version of FCA-based learning system to the analysis of Wine Quality dataset from UCI Machine Learning Repository.

Keywords: FCA · Machine Learning · continuous attributes · entropy.

Introduction

In [8] the author extended some FCA ideas [1] by developing a probabilistic approach to Machine Learning (ML) based on similarity operation. FCA provides a very efficient representation for training objects by means of bitsets (fixed length strings of bits) with bit-wise multiplication as a similarity between them. The previous version of the ML program (called 'VKF system' in honor to Prof. V.K. Finn) was applicable to objects described by discrete attributes only. However, a variety of interesting data needs both discrete and continuous attributes for their representation. The first step to include continuous cases to VKF system is an analogue of J.R. Quinlan's approach to similar problem for C4.5 decision tree algorithm [5]. He splits the whole domain of a continuous attribute into several intervals to reach minimal mean entropy.

To obtain bitset representation from such division we introduce indicator variables and combine their values. The main result asserts that bit-wise multiplication corresponds to a convex hull of intervals of values under similarity, which was studied in [2] and [4] in terms of interval pattern structures within, an FCA-based approach to analysis of data with continuous attributes. A more important problem is to discover complex combinations of continuous attributes.

[⋆] partially supported by RFBR grant 18-29-03063mk.

Since our main goal is to discover a classifier, we apply Bayes Machine Learning ideas to generate such complex attributes through well-known logistic regression.

The key question is to detect significance of essential relationships (interactions) between pairs of attributes. Hence, we apply well-known Cox-Snell and McFadden criteria to discover such interactions.

The structure of the paper is as follows. In Section 1 we recall general definitions and some facts from FCA. Section 2 covers main algorithms of VKF-method. Section 3 describes new results. Subsection 3.1 reproduces Quinlan's technique to separate continuous feature domain into several intervals. It also introduces a representation of the occurrence of an attribute value in some interval by bitset. Subsection 3.2 introduces a logistic regression approach to discovering relationships between continuous features.

1 Formal Concept Analysis (FCA)

A **(finite) context** is a triple (G, M, I) where G and M are finite sets and $I \subseteq G \times M$. The elements of G and M are called **objects** and **attributes**, respectively. As usual, we write gIm instead of $\langle g, m\rangle \in I$ to denote that object g has attribute m.

For $A \subseteq G$ and $B \subseteq M$, define

$$A' = \{m \in M | \forall g \in A(gIm)\}, \tag{1}$$

$$B' = \{g \in G | \forall m \in B(gIm)\}; \tag{2}$$

so A' is the set of attributes common to all the objects in A and B' is the set of objects possessing all the attributes in B. The maps $(\cdot)' : A \mapsto A'$ and $(\cdot)' : B \mapsto B'$ are called **derivation operators** (**polars**) of the context (G, M, I).

If we fix attribute subsets $\{g_1\}' \subset M$ and $\{g_2\}' \subset M$ for objects $g_1 \in G$ and $g_2 \in G$, respectively, with corresponding bitsets, then the derivation operator on a pair of objects corresponds to bit-wise multiplication, since $\{g_1, g_2\}' = \{g_1\}' \cap \{g_2\}'$. More generally, the polars correspond to the iteration of bit-wise multiplication (in arbitrary order) of corresponding bitset-represented objects and attributes, respectively. The last remark is important, since bit-wise multiplication is a basic operation of modern CPU and GPGPU. The aim of the article is to invent a bitset representation of continuous features in such a way that bit-wise multiplication of resulting bitsets has clear meaning with respect to original values!

A **concept** of the context (G, M, I) is defined to be a pair (A, B), where $A \subseteq G$, $B \subseteq M$, $A' = B$, and $B' = A$. The first component A of the concept (A, B) is called the **extent** of the concept, and the second component B is called its **intent**. The set of all concepts of the context (G, M, I) is denoted by $L(G, M, I)$.

Definition 1. *For $(A, B) \in L(G, M, I)$, $g \in G$, and $m \in M$ define*

$$CbO((A, B), g) = ((A \cup \{g\})'', B \cap \{g\}'), \tag{3}$$

$$CbO((A, B), m) = (A \cap \{m\}', (B \cup \{m\})''). \tag{4}$$

We call these operations CbO because the first one is used in well-known Close-by-One (CbO) Algorithm [3] for generating all concepts from $L(G,M,I)$.

Lemma 1. *Let (G,M,I) be a context, $(A,B) \in L(G,M,I)$, $g \in G$, and $m \in M$. Then*

$$CbO((A,B),g) = (A,B) \vee (\{g\}'', \{g\}'), \tag{5}$$

$$CbO((A,B),m) = (A,B) \wedge (\{m\}', \{m\}''). \tag{6}$$

This lemma proves the correctness of definition 1 of operations CbO. Most important property of these operations is represented in the following

Lemma 2. *Let (G,M,I) be a context, $(A_1,B_1),(A_2,B_2) \in L(G,M,I)$, $g \in G$, and $m \in M$. Then*

$$(A_1,B_1) \le (A_2,B_2) \Rightarrow CbO((A_1,B_1),g) \le CbO((A_2,B_2),g), \tag{7}$$

$$(A_1,B_1) \le (A_2,B_2) \Rightarrow CbO((A_1,B_1),m) \le CbO((A_2,B_2),m). \tag{8}$$

2 FCA-based Machine Learning

We deal with supervised Machine Learning. Hence we have training examples together with the target values on them. All examples are described by binary attributes from M, i.e. they can be given by bitsets of fixed length. Usually, a subset of examples is used as a **test sample** G^τ for checking the quality of training. The training examples are divided into positive G^+ and negative G^- subsets according to the value of the target attribute. The elements of G^+ and G_- make the **training sample**, elements of G^- are called **counter-examples** (**obstacles**). Formal context (G^+,M,I) is the main data set.

The well-known example $(S,S,\neq)$ of context for Boolean algebra demonstrates difficulties of brute force approach. For Boolean algebra of all subsets of n elements the context uses n^2 bits, and all the concepts need $n \cdot 2^n$ bits. For $n = 32$ the first number is 1 Kb (or 128 bytes) and the second one is 16 Gigabytes! The time complexity is exponential too.

Hence we need to replace computation of the whole lattice of all concepts by randomized algorithms to generate a random subset of the lattice. The author introduced and investigated mathematical properties of several algorithms of this kind, the best of which are variants of coupling Markov chains.

Now we represent the classical version of coupling Markov chain that is a core of probabilistic approach to machine learning based on FCA (VKF-method).

```
Data: context $(G^+,M,I)$, external function $CbO( \, , \, )$
Result: random concept $(A,B) \in L(G^+,M,I)$
$X := G \sqcup M$; $(A,B) := (M',M)$; $(C,D) = (G,G')$;
while $((A \neq C) \vee (B \neq D))$ do
    select random element $x \in X$;
    $(A,B) := CbO((A,B),x)$;
    $(C,D) := CbO((C,D),x)$;
end
```

Algorithm 1: Coupling Markov chain

The ordering of two concepts $(A, B) \leq (C, D)$ at any intermediate step of the while loop of Algorithm 1 is defined by Lemma 2.

For Boolean lattice (contranomial) context the author [8] computed the mean length of trajectory of Algorithm 1 as $n \sum_{j=1}^{n} \frac{1}{n}$ and proved strong concentration of length of arbitrary trajectory about its mean. For $n = 32$ the mean is ≤ 130, hence every trajectory generates about 260 (since two concepts is a state of the coupling Markov chain) subsets. Hence, only a small fraction of concepts occurs during computation of a moderate size subset of the Boolean algebra. We have in mind that there are 4,294,967,296 elements of Boolean algebra on 32 attributes.

Machine Learning procedure has two steps: induction and prediction. At the first step the system generate hypotheses about causes of the target property from training sample. At the prediction step the system applies the hypotheses to predict the target value for test examples.

The induction step of FCA-based learning applies the Coupling Markov chain Algorithm 1 to generate a random formal concept $(A, B) \in L(G^+, M, I)$. The program saves the concept (A, B) if there is no obstacle (counter-example) $o \in G^-$ such that $B \subseteq o'$.

```
Data: number $N$ of concepts to generate
Result: random sample $S$ of formal concepts without obstacles
$G^+$ := (+)-examples, $M$ := attributes; $I \subseteq G^+ \times M$ is a formal context
 for (+)-examples;
$G^-$ := (-)-examples; $S := \emptyset$; $i := 0$;
while ($i < N$) do
    Generate concept $(A, B)$ by Algorithm 1;
    $hasObstacle$ := false;
    for ($o \in G^-$) do
        if ($B \subseteq o'$) then
            $hasObstacle$ := true;
        end
    end
    if ($hasObstacle$ = false) then
        $S := S \cup \{(A, B)\}$;
        $i := i + 1$;
    end
end
```

Algorithm 2: Inductive generalization

Condition $(B \subseteq o')$ of Algorithm 2 means the inclusion of intent B of concept (A, B) into the intent of counter-example o.

If a concept "avoids" all such obstacles it is added to the result set of all the concepts without obstacles.

We replace a time-consuming deterministic algorithm (for instance, "Close-by-One" [3]) for generation of all concepts by the probabilistic one to randomly generate the prescribed number of concepts.

The goal of Markov chain approach is to select a random sample of formal concepts without computation of the (possibly exponential size) set $L(G, M, I)$ of all the concepts.

Finally, machine learning program predicts the target class of test examples and compares the results of prediction with the original target value.

```
Data: random sample $S$ of concepts, list $G^\tau$ of test objects
Result: prediction of target class of $G^\tau$ elements
for ($o \in G^\tau$) do
    $PredictPositively(o) :=$ false;
    for ($(A, B) \in S^+$) do
        if ($B \subseteq o'$) then
            $PredictPositively(o) :=$ true;
        end
    end
end
```

Algorithm 3: Prediction of target class by analogy

The author proved [8] the following theorem to estimate parameter N from Algorithm 2.

Test object o is an ε-**important** if probability of all concepts (A, B) with $B \subseteq \{o\}'$ exceeds ε.

Theorem 1. *For $n = |M|$ and for any $\varepsilon > 0$ and $1 > \delta > 0$ random sample S of concepts of cardinality*

$$N \geq \frac{2 \cdot (n+1) - 2 \cdot \log_2 \delta}{\varepsilon} \tag{9}$$

with probability $> 1 - \delta$ has property that every ε-important object o contains some concept $(A, B) \in S$ such that $B \subseteq \{o\}'$.

This theorem is an analogue of the famous results of V. Vapnik and A. Chervonenkis [7] from Computational Learning Theory (here $n + 1$ corresponds to $\log_2 d$, where d is a VC-dimension).

From the practical point of view this theorem asserts the sufficiency of polynomial number of random concepts as causes of the target property to minimize 1-type error (wrong prediction of positive test examples) with respect to prediction by analogy (Algorithm 3).

3 Continuous attributes

3.1 Entropy approach

Let $G = G^+ \cup G^-$ be a disjoint union of training examples G^+ and counter-examples G^-. Interval $[a, b) \subseteq \mathbb{R}$ of values of continuous attribute $V : G \to \mathbb{R}$ generates three subsets

$$G^+[a, b) = \{g \in G^+ : a \leq V(g) < b\},$$

$$G^-[a,b) = \{g \in G^- : a \leq V(g) < b\},$$

$$G[a,b) = \{g \in G : a \leq V(g) < b\}$$

.

Definition 2. ***Entropy*** *of interval* $[a,b) \subseteq \mathbb{R}$ *of values of continuous attribute* $V : G \to \mathbb{R}$ *is*

$$\text{ent}[a,b) = -\frac{|G^+[a,b)|}{|G[a,b)|} \cdot \log_2\left(\frac{|G^+[a,b)|}{|G[a,b)|}\right) - \frac{|G^-[a,b)|}{|G[a,b)|} \cdot \log_2\left(\frac{|G^-[a,b)|}{|G[a,b)|}\right) \quad (10)$$

Mean information *for partition* $a < r < b$ *of interval* $[a,b) \subseteq \mathbb{R}$ *of values of continuous attribute* $V : G \to \mathbb{R}$ *is*

$$\inf[a,r,b) = \frac{|E[a,r)|}{|E[a,b)|} \cdot \text{ent}[a,r) + \frac{|E[r,b)|}{|E[a,b)|} \cdot \text{ent}[r,b). \quad (11)$$

Threshold *is a value* $V = r$ *with minimal mean information.*

For continuous attribute $V : G \to \mathbb{R}$ denote $a = \min V$ by v_0 and let v_{l+1} be an arbitrary number greater then $b = \max V$. Thresholds $\{v_1 < \ldots < v_l\}$ are computed sequentially by splitting the largest entropy subinterval.

These constructions were introduced by J.R. Quinlan for C4.5, the well-known system for learning Decision Trees [5].

Definition 3. *For each* $1 \leq i \leq l$ *indicator (Boolean) variables corresponds to*

$$\delta_i^V(g) = 1 \Leftrightarrow V(g) \geq v_i \quad (12)$$

$$\sigma_i^V(g) = 1 \Leftrightarrow V(g) < v_i \quad (13)$$

Then string $\delta_1^V(g) \ldots \delta_l^V(g) \sigma_1^V(g) \ldots \sigma_l^V(g)$ *is a* ***bitset-representation*** *of continuous attribute* V *on element* $g \in G$.

Lemma 3. *Let* $\delta_1^{(1)} \ldots \delta_l^{(1)} \sigma_1^{(1)} \ldots \sigma_l^{(1)}$ *represent* $v_i \leq V(A_1) < v_j$ *and* $\delta_1^{(2)} \ldots \delta_l^{(2)} \sigma_1^{(2)} \ldots \sigma_l^{(2)}$ *represent* $v_n \leq V(A_2) < v_m$. *Then*

$$(\delta_1^{(1)} \& \delta_1^{(2)}) \ldots (\delta_l^{(1)} \& \delta_l^{(2)})(\sigma_1^{(1)} \& \sigma_1^{(2)}) \ldots (\sigma_l^{(1)} \& \sigma_l^{(2)})$$

corresponds to $\min\{v_i, v_n\} \leq V((A_1 \cup A_2)'') < \max\{v_j, v_m\}$.

In other words, Lemma 3 asserts that the result of bit-wise multiplication of bitset representations is a convex hull of its arguments' intervals.

The proof follows immediately from definition 3.

Similar bitset presentation for continuous features was mentioned earlier in [4] for interval pattern structures. However this work uses a priori given subdivision of a feature domain into disjoint subintervals.

3.2 Logistic regression between attributes

A **classifier** is a map $c : \mathbb{R}^d \to \{0, 1\}$, where $\mathbb{R}^d$ is a domain of objects to classify (described by d attributes) and $\{0, 1\}$ are *class marks.*

Probability distribution of $\langle \boldsymbol{X}, K\rangle \in \mathbb{R}^d \times \{0, 1\}$ can be decomposed as

$$p_{\boldsymbol{X},K}(\boldsymbol{x}, k) = p_{\boldsymbol{X}}(\boldsymbol{x}) \cdot p_{K|\boldsymbol{X}}(k \mid \boldsymbol{x}),$$

where $p_{\boldsymbol{X}}(\boldsymbol{x})$ is a marginal distribution of objects and $p_{K|\boldsymbol{X}}(k \mid \boldsymbol{x})$ is a conditional distribution of marks on given object, i.e. for every $\boldsymbol{x} \in \mathbb{R}^d$ the following $p_{K|\boldsymbol{X}}(k \mid \boldsymbol{x}) = \mathbb{P}\{K = k \mid \boldsymbol{X} = \boldsymbol{x}\}$ holds.

Error probability of classifier $c : \mathbb{R}^d \to \{0, 1\}$ is

$$R(c) = \mathbb{P}\left\{c(\boldsymbol{X}) \neq K\right\}. \qquad (14)$$

Bayes classifier $b : \mathbb{R}^d \to \{0, 1\}$ with respect to $p_{K|\boldsymbol{X}}(k \mid \boldsymbol{x})$ corresponds to

$$b(\boldsymbol{x}) = 1 \Leftrightarrow p_{K|\boldsymbol{X}}(1 \mid \boldsymbol{x}) > \frac{1}{2} > p_{K|\boldsymbol{X}}(0 \mid \boldsymbol{x}) \qquad (15)$$

We remind well-known

Theorem 2. *The Bayes classifier b has the minimal error probability:*

$$\forall c : \mathbb{R}^d \to \{0, 1\} \left[R(b) = \mathbb{P}\{b(\boldsymbol{X}) \neq K\} \leq R(c)\right]$$

Bayes Theorem implies

$$\begin{aligned} p_{K|\boldsymbol{X}}(1 \mid \boldsymbol{x}) &= \frac{p_{\boldsymbol{X}|K}(\boldsymbol{x} \mid 1) \cdot \mathbb{P}\{K = 1\}}{p_{\boldsymbol{X}|K}(\boldsymbol{x} \mid 1) \cdot \mathbb{P}\{K = 1\} + p_{\boldsymbol{X}|K}(\boldsymbol{x} \mid 0) \cdot \mathbb{P}\{K = 0\}} = \\ &= \frac{1}{1 + \frac{p_{\boldsymbol{X}|K}(\boldsymbol{x}|0)\cdot\mathbb{P}\{K=0\}}{p_{\boldsymbol{X}|K}(\boldsymbol{x}|1)\cdot\mathbb{P}\{K=1\}}} = \frac{1}{1 + \exp\{-a(\boldsymbol{x})\}} = \sigma(a(\boldsymbol{x})) \end{aligned}$$

where $a(\boldsymbol{x}) = \log \frac{p_{\boldsymbol{X}|K}(\boldsymbol{x}|1)\cdot\mathbb{P}\{K=1\}}{p_{\boldsymbol{X}|K}(\boldsymbol{x}|0)\cdot\mathbb{P}\{K=0\}}$ and $\sigma(y) = \frac{1}{1+\exp\{-y\}}$ is the well-known **logistic function**.

Equation (15) transforms to

$$b(\boldsymbol{x}) = 1 \Leftrightarrow a(\boldsymbol{x}) > 0 \qquad (16)$$

Let approximate unknown $a(\boldsymbol{x}) = \log \frac{p_{\boldsymbol{X}|K}(\boldsymbol{x}|1)\cdot\mathbb{P}\{K=1\}}{p_{\boldsymbol{X}|K}(\boldsymbol{x}|0)\cdot\mathbb{P}\{K=0\}}$ by linear combination $\boldsymbol{w}^T \cdot \varphi(\boldsymbol{x})$ of basis functions $\varphi_i : \mathbb{R}^d \to \mathbb{R}$ $(i = 1, \ldots, m)$ with respect to unknown weights $\boldsymbol{w} \in \mathbb{R}^m$.

For training sample $\langle \boldsymbol{x}_1, k_1\rangle, \ldots, \langle \boldsymbol{x}_n, k_n\rangle$ introduce $t_j = 2k_j - 1$. Then

$$\log\{p(t_1, \ldots, t_n \mid \boldsymbol{x}_1, \ldots, \boldsymbol{x}_n, \boldsymbol{w})\} = -\sum_{j=1}^{n} \log\left[1 + \exp\{-t_j \sum_{i=1}^{m} w_i \varphi_i(\boldsymbol{x}_j)\}\right].$$

Lemma 4. $\log\left[1 + \exp\{-t \cdot \sum_{i=1}^{m} w_i\varphi_i\}\right]$ *is a convex function of* $\boldsymbol{w}$.

Hence, the logarithm of likelihood

$$L(w_1,\ldots,w_m) = -\sum_{j=1}^{n} \log\left[1+\exp\{-t_j\sum_{i=1}^{m} w_i\varphi_i(\boldsymbol{x}_j)\}\right] \to \max \tag{17}$$

is concave.

Newton-Raphson method leads to iterative procedure

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t - (\nabla_{\boldsymbol{w}^T}\nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t))^{-1}\cdot\nabla_{\boldsymbol{w}} L(\boldsymbol{w}_t). \tag{18}$$

Use $s_j = \frac{1}{1+\exp\{t_j\cdot(w^T\cdot\Phi(x_j))\}}$ we obtain

$$\nabla L(\boldsymbol{w}) = -\Phi^T\mathrm{diag}(t_1,\ldots,t_n)\boldsymbol{s}, \nabla\nabla L(\boldsymbol{w}) = \Phi^T R\Phi,$$

where $R = \mathrm{diag}(s_1(1-s_1), s_2(1-s_2),\ldots,s_n(1-s_n))$ is diagonal matrix with elements $s_1(1-s_1), s_2(1-s_2),\ldots,s_n(1-s_n)$ and $\mathrm{diag}(t_1,\ldots,t_n)\boldsymbol{s}$ is vector with coordinates $t_1s_1, t_2s_2,\ldots,t_ns_n$.

$$\boldsymbol{w}_{t+1} = \boldsymbol{w}_t + \left(\Phi^T R\Phi\right)^{-1}\Phi^T\mathrm{diag}(t)\boldsymbol{s} = (\Phi^T R\Phi)^{-1}\Phi^T R\boldsymbol{z}, \tag{19}$$

where $\boldsymbol{z} = \Phi\boldsymbol{w}_t + R^{-1}\mathrm{diag}(t_1,\ldots,t_n)\boldsymbol{s}$ are iterative calculated weights.

As usual, the ridge regression helps to avoid ill-conditioned situation

$$\boldsymbol{w}_{t+1} = (\Phi^T R\Phi + \lambda\cdot I)^{-1}\cdot(\Phi^T R\boldsymbol{z}).$$

In the computer program 'VKF system' we use standard basis: constant 1 and attributes themselves.

At last, we need a criterion for significance of regression. For logistic regression two types of criteria were applied:

Criterion of Cox-Snell declares attribute V_k significant, if

$$R^2 = 1-\exp\{2(L(w_0,\ldots,w_{k-1}) - L(w_0,\ldots,w_{k-1},w_k))/n\} \geq \sigma. \tag{20}$$

McFadden criterion declares attribute V_k significant, if

$$1-\frac{L(w_0,\ldots,w_{k-1},w_k)}{L(w_0,\ldots,w_{k-1})} \geq \sigma. \tag{21}$$

Conclusion

We have extended the 'VKF system' approach to FCA-based machine learning on examples with both discrete and continuous attributes.

Experiments with Wine Quality Dataset [6] demonstrate a very good behavior of the proposed approach. For red wines with high scores (more than 7) all examples were classified correctly.

The pair-wise logistic regression is combined with single threshold computation. Lemma 3 gives a condition of non-triviality of similarity on values of continuous attribute: if the corresponding part of the resulting bitset is nonvoid, then the values $V(B')$ belong to a common interval.

When analyzing relationship between 'alcohol' and 'sulphates' for red wines we observe a phenomenon directly corresponding to the well-known

Lemma 5. *Disjunction* $x_{i_1} \vee \ldots \vee x_{i_k}$ *of Boolean variables holds, if and only if* $x_{i_1} + \ldots + x_{i_k} \geq \sigma$ *holds for any* $0 < \sigma < 1$.

Positive (but slightly different) weights correspond to different scaling of various attributes. So we have not only conjuction of attributes by also a disjunction. Similar case is a relationship between 'citric acid' and 'alcohol'.The situation with the pair ('pH', 'alcohol') is radically different. The alcohol's weight is positive, whereas pH's weight is negative. With the help of aforementioned lemma and standard logic we obtain the implication ('pH'⇒'alcohol').

Acknowledgements

The author would like to thank Prof. S.O. Kuznetsov for motivation and support. He is also grateful to his colleagues from Dorodnicyn Computing Center of Federal Research Center "Computer Science and Control" of Russian Academy of Science and Russian State University for Humanities for long-term support and useful discussions. Finally, the author is grateful to anonymous reviewers, whose comments helped to significantly improve the presentation.

References

1. Ganter, B., Wille, R.: Formal Concept Analysis. Springer, Berlin (1999)
2. Kaytoue, M., Kuznetsov, S.O., Napoli, A., Duplessis, S.: Mining gene expression data with pattern structures in formal concept analysis. Inf. Sci. 181(10): 1989-2001 (2011)
3. Kuznetsov, S.O.: A Fast Algorithm for Computing all Intersections of Objects in a Finite Semi-Lattice. Autom. Doc. Math. Linguist. 27 (5), 11-21 (1993)
4. Makhalova, M., Kuznetsov, S.O., Napoli, A.: Numerical Pattern Mining through Compression. 2019 Data Compress. Conf. Proc. IEEE. 112-121 (2019)
5. Quinlan, J.R.: C4.5 Programs for Machine Learning. Morgan Kaufmann, San Francisco (1993)
6. UCI Machine Learning Repository: Wine Quality Data Set, https://archive.ics.uci.edu/ml/datasets/Wine+Quality. Last accessed 20 June 2020
7. Vapnik, V., Chervonenkis, A.: On the Uniform Convergence of Relative Frequencies of Events to Their Probabilities. Theory Probab. Appl. 16 (2), 264–280 (2004)
8. Vinogradov, D.V.: Machine Learning Based on Similarity Operation. In: Kuznetsov S., Osipov G., Stefanuk V. (eds) Artificial Intelligence. RCAI 2018. Communications in Computer and Information Science. **934**, 46–59 (2018)
9. Vinogradov, D.V.: On Object Representation by Bit Strings for the VKF-Method. Autom. Doc. Math. Linguist. 52 (4), 113–116 (2018)