Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 96 (2016) 1418 – 1427

Procedia Computer Science

20th International Conference on Knowledge Based and Intelligent Information and Engineering Systems, KES2016, 5-7 September 2016, York, United Kingdom

On influence of representations of discretized data on performance of a decision system

Grzegorz Baron[a]*

[a] *Silesian University of Technology, Akademicka 16, Gliwice 44-100, Poland*

Abstract

When discretization is used for preprocessing datasets in a decision system different representations of data can be taken into consideration. Typical approach is to use data as it is returned by discretizer, namely as nominal values. But in specific cases such form of data cannot be utilized by next modules of the decision system. Then the possible solution is to convert nominal data again into a numerical form. The paper presents comparison of such approaches applied for different classifiers in stylometry domain.

Peer-review under responsibility of KES International

Keywords: numerical representation; nominal representation; discretization; Naive Bayes; Bayesian network; decision tree; C4.5; *k*-Nearest Neighbors;

1. Introduction

Discretization is a preprocessing approach to numerical input data. Two main cases exist for which application of data discretization should or even must be considered. In the first case, when next elements of a decision system (such as classifier) cannot operate on numerical data. Then use of discretization for data preparation is obligatory. In the second case such data preprocessing is facultative and it should be analyzed if any benefits can be obtained, for example improvement of classification performance [1].

It is obvious that outcome data of discretization module is nominal which means that each instance of an attribute has values coming from the finite set of enumerated data type (using computer programming nomenclature). Such processing often reduces amount of data and in some cases allows to extract knowledge contained in input data in more effective way. The question is if discretized data must be processed in subsequent modules of the decision system in this form or it can be converted back to the numerical form. The conversion can be performed for example by assigning respectively subsequent positive integers to elements of the set of discrete values (and in parallel updating discretized dataset). Motivation for this approach comes from the author's former research, where in some cases discretization

* Corresponding author. Tel.: +48-32-237-2009 ; fax: +48-32-237-2733.
E-mail address: grzegorz.baron@polsl.pl

Peer-review under responsibility of KES International
doi:10.1016/j.procs.2016.08.187

of learning and test datasets led to situations where the numbers of bins in both sets varied which caused problems during evaluation of some classifiers, providing discrete data. At the same time the classifiers evaluation could have been performed when numerical data was applied. To observe possible influence of such approaches on overall performance of a decision system the following classifiers were chosen: Naive Bayes, Bayesian network, *k*-Nearest Neighbors and decision tree C4.5. They are widely used in different areas, especially in authorship attribution domain which is considered in presented research.

Authorship attribution as a task from the stylometry domain deals with text analysis in order to determine an author of a given text. Besides recognition of texts' authors also author's characterization, similarity detection (i.e. plagiarism) are the research areas of stylometry. One of the most important issues is the selection of descriptors which allow to construct characteristic features sets containing properties being invariant of its author. Machine learning and statistics are the most popular techniques[2,3,4].

The paper presents analysis of influence of both aforementioned approaches to representation of discretized data on performance of selected classifiers for an authorship attribution task. Although research addressing discretization are very popular, to the best of the author's knowledge there are no publications directly addressing the problem discussed in this paper

The paper is organized as follows, Section 2 gives the theoretical background and an overview of methods employed in the research. Section 3 presents the experimental setup, datasets used and techniques employed. The test results and their discussion are given in Section 4. Section 5 provides conclusions.

Nomenclature

EWd	equal width discretization
EWOd	optimized equal width discretization
EFd	equal frequency discretization
FId	Fayyad & Irani discretization
Kd	Kononenko discretization

2. Theoretical background

The following subsections describe discretization algorithms including forms of discretized data representations, and classifiers used in the decision system.

2.1. Discretization

Discretization is defined as the process of conversion of continuous attributes into nominal ones but also as the process of merging of nominal attributes. In real world domains continuous attributes, which are typically represented in decision systems by numerical values, occur frequently. On the other hand application of some machine learning algorithms or methods in continuous domain can be impossible or associated with certain problems. In many cases discretization can improve learning process delivering results quicker or allowing to obtain simpler or more accurate decision system.

Discretization methods can be categorized as local or global, and unsupervised or supervised. The first division is related to the approach to defining ranges of bins during the discretization process. In the global method cut points are defined for whole attribute domains whereas for local discretization they are determined by different relations (even with other attributes), and are defined separately for distinctive parts of domain. The second categorization is related to the way of treating the instance's class attribution during the discretization process. Supervised methods utilize class information whereas unsupervised ones omit such information. Supervised methods are able to deliver better results which represent nature of input data in more accurate way comparing to the unsupervised ones[5,6].

As representatives of unsupervised discretization methods the equal width EWd and equal frequency EFd binning can be mentioned. The EWd method for each attribute determines the minimum and maximum values and calculates

cut points in order to obtain desired number of equal-sized discrete intervals. The algorithm available in WEKA[7] provides a modified version of EWd (denoted in the paper as EWOd) which optimizes a number of bins basing on the leave-one-out estimation of estimated entropy. Therefore the resulting number of bins and cut points allows to obtain the dataset which reflects better the nature of discretized data. The EFd algorithm sorts values of an attribute in ascending order, and after evaluation of the minimum and maximum attribute's values, calculates the cut points in such way that the same number of attribute's instances is placed in each bin[6]. There also exists an alternative version of EFd algorithm, where instead of requiring number of bins, the weight of instances per bin is set as the input parameter. If instances are not weighted such processing delivers the resulting number of bins containing assumed number of instances.

Apart from unsupervised discretization methods, two supervised algorithms were chosen. Both are based on the Minimum Description Length principle (MDL). The first one was introduced by Fayyad and Irani's[8] whereas the second one was developed by Kononenko[9].

2.2. Classifiers

In order to obtain a wide range of results which would allow to conduct discussion about influence of approaches to representation of discretized data on overall performance of a decision system, four different classifiers were selected: Naive Bayes, Bayesian network, decision tree – C4.5, and k-Nearest Neighbors – k-NN. The following points introduce them briefly.

2.2.1. Naive Bayes classifier

Naive Bayes classifier is considered as simple but in fact it is a very efficient tool applicable in a wide range of application domains. It is often used to establish reference levels for other classification research and discussions. Two versions of Naive Bayes classifier can be considered when analysis is focused on text classification tasks. The multivariate Naive Bayes classifier is useful for input data containing only binary variables providing information about presence of selected function words in analyzed texts. If datasets contain quantitative information about word's occurrences, the multinomial Naive Bayes classifier can be applied. Such approach is recommended for huge sizes of the vocabulary sets[10] and was utilized in the presented research.

Naive Bayes classifier is founded on Bayes' rule of conditional probability:

$$p(c_j \mid d) = \frac{p(d \mid c_j)p(c_j)}{p(d)}, \tag{1}$$

where:

- $p(c_j \mid d)$ – a'posteriori probability of instance d being in class c_j,
- $p(d \mid c_j)$ – probability of generating instance d given class c_j,
- $p(c_j)$ – a'priori probability of occurence of class c_j,
- $p(d)$ – probability of instance d occurring.

For series of instances the total value of conditional probability $p(d \mid c_j)$ can be calculated as a product of elementary probabilities for all instances d_i, as the following equation presents:

$$p(d \mid c_j) = p(d_1 \mid c_j)p(d_2 \mid c_j) \dots p(d_m \mid c_j). \tag{2}$$

The result of classification task $NBC(d_1, \dots, d_n)$ is determined by the MAP (maximum a'posteriori) decision rule:

$$NBC(d_1, \dots, d_n) = \underset{c}{\operatorname{argmax}} p(C = c) \prod_{i=1}^{n} p(D_i = d_i \mid C = c). \tag{3}$$

An important issue related to the Naive Bayes method is the assumption about distribution of analyzed data. In the presented research the values of numeric attributes were assumed to be normally distributed, so the Gaussian probability density function was utilized. For specific tasks other distributions can be considered as more suitable and delivering better results[11].

2.2.2. Bayesian networks

Bayesian networks[12,13] can be considered as part of a wider class of statistical models which are called graphical models. Their structure is founded on a network of nodes connected by edges. Each node is dedicated to one attribute, connections are directed and there are no cycles in such graph (so the directed acyclic graph is obtained). They represent probabilistic relationships between attributes, therefore the structure of Bayesian network allows to store joint probability of all attributes a_i, which is described by the following formula:

$$p[a_1, a_2, \ldots, a_m] = \prod_{i=1}^{m} p[a_i | a_{i-1}, \ldots, a_1]. \tag{4}$$

The advantage of this method is that to infer value of any attribute a_i only conditional probabilities of its parents are necessary. That allows to store joint probabilities in more compact form comparing to the straight forward approach, where all probabilities must be stored.

A fundamental issue in Bayesian networks domain is a learning which can be considered as two stage process: creating the network structure and determining necessary probabilities. The nature of a problem and the applied learning algorithm determine if one or both stages are performed automatically. During the presented research the K2 algorithm introduced by Cooper and Herskovits was utilized[14].

2.2.3. k-Nearest Neighbors classifier

The k-Nearest Neighbors algorithm relies on the idea of finding k nearest neighbors of an unknown feature vector v_x in the feature space V, given some defined distance measure. The training vectors $v_t \in V$, where $t = 1 \ldots T$, have their classes $c_t; c \in C$ attributed. The vector v_x is assigned to the class $c_x \in C$, which has most representatives in the calculated set of nearest neighbors. Different methods of specifying the nearest neighbors and distance measures can be utilized[12]. For the purpose of binary classification the number of neighbors is $k \neq 2$. Generally k should not be the multiple of the number of classes existing in the studied solution space.

2.2.4. Decision tree C4.5

Decision trees belong to the class of nonlinear classifiers which in multistage sequential process perform splitting of the feature space into disjunctive regions representing classes. The problem is how to construct a decision tree in order to obtain the best possible classifier, given the learning data. The process starts from selecting an attribute to be the root node of the tree and continues by creating branches for all possible values of the attribute (assuming for simplicity nominal nature of data). The process can be repeated recursively for each branch until some stop criterion is reached, i.e. all leaves of the tree contain the same classification values. Of course this simple description does not exhaust the existing problems. Starting from the elementary but fundamental questions, like how to select the sequence of attributes to obtain best fitting of the model to the real problem, through ways of dealing with numerical attributes and missing values, and finishing with pruning of the resulting decision tree to obtain the classifier with better generalization abilities. The decision tree algorithm selected for the presented research is C4.5 developed by Quinlan[15]. The algorithm implementation contained in WEKA[7] was utilized. It provides many options to control the base process, like use of MDL correction for numerical attributes, possibility to perform binary splitting, or additional parameters for pruning process.

3. Experimental setup

During experiments datasets were processed using the following steps:

1. preparation of input data,
2. discretization and optional conversion of outcome data to numerical representation,
3. classification using different classifiers applied for both forms of data,
4. classifiers evaluation.

3.1. Datasets

As the base for all experiments texts of some authors were chosen [16]. Several works of each author were studied. In the presented research input texts were preprocessed in order to create datasets containing sets of characteristic features which should be unique for a given author and differentiating among other authors.

In order to satisfy the aforementioned condition linguistic descriptors from lexical and syntactic groups were chosen, reflecting frequencies of usage for function words and punctuation marks, as follows [16]:

- lexical elements – but, and, not, in, with, on, at, of, this, as, that, what, from, by, for, to, if,
- syntactic elements – a fullstop, a comma, a question mark, an exclamation mark, a semicolon, a colon, a bracket, a hyphen.

It is assumed that lexical elements characterize literary styles of authors whereas the style of sentences building is described by syntactic features. Authors were grouped into 2-elements sets, given gender. Consequently, each dataset contains only two classes and performed classification was binary.

Test datasets were used for validation of classifiers. To obtain objective results training and test sets were prepared based on the separate suites of writers' works. As the result the separate training and testing datasets were obtained, with two balanced classes in each set with selected attributes.

3.2. Discretization

The following discretization methods were utilized during presented experiments: equal width (EWd), optimized equal width (EWOd), equal frequency (EFd), and supervised Fayyad & Irani (FId) and Kononenko (Kd) MDL. The last two methods do not require any parameters whereas unsupervised algorithms do. Based on the author's previous experiences the parameters for unsupervised algorithms were prepared as follows: for EWd and EWOd methods the parameter which represents the required number of bins ranged from 2 to 10 with step 1, and from 10 to 100 with step 10. For EFd the binning parameter was varied from 2 to 19. The upper bounds for all algorithms could be greater but former experiments proved that overall performance of decision systems degrades for higher values of parameters, so performing analysis in that range is futile.

After discretization each instance of a given attribute gets a discrete value belonging to the set obtained during the process. Exemplary result for the attribute "with" discretized using EWd (for the required number of bins set to 4) looks as follows:

```
@attribute with {'\'(-inf-0.0065955]\'','\'(0.0065955-0.00798]\'',
                 '\'(0.00798-0.0093645]\'','\'(0.0093645-inf)\''}
@attribute author {edith,jane}

@data
'\'(0.0093645-inf)\'',edith
'\'(0.00798-0.0093645]\'',edith
'\'(0.00798-0.0093645]\'',edith
'\'(0.0093645-inf)\'',edith
'\'(0.0065955-0.00798]\'',jane
'\'(0.0065955-0.00798]\'',jane
'\'(0.0093645-inf)\'',jane
'\'(0.00798-0.0093645]\'',jane
```

The format presented is typical for WEKA environment used for performing the experiments. First lines describe two attributes. The first one was discretized whereas the second contains class information and was kept unchanged. The way of presenting ranges of bins in a human-readable form allows to get information about cut points calculated during the discretization. In fact that is the set of literals separated by commas, representing subsequent elements of the discrete set, where each element is formed from values of lower and upper bound of the given bin. The "-inf" and "inf" represent minus and plus infinity which are used for formal description of the boundary bins. In other words,

elements of the set can be considered as the bin's descriptors. The "@data" field contains subsequent instances in discrete form. The first field of each instance contains discrete value of an attribute, the second one stores class value.

As it was introduced in Section 1, the aim of presented research was to analyze influence of conversion of discretized data into numerical form on overall performance of decision system. Based on the example presented above, the new numerical form of discretized data looks as follows:

```
@attribute with real
@attribute author {edith,jane}

@data
4,edith
3,edith
3,edith
4,edith
2,jane
2,jane
4,jane
3,jane
```

The attribute "with" is now declared in data header as real (numerical) value. The attribute values assigned to each instance were determined by using ordinal numbers representing position of a required bin descriptor in the set of all discrete bin values, given the attribute.

The last issue related to discretization is the approach to discretization of test datasets. As aforementioned, during the preparation of input datasets, the separate works of authors were used to obtain proper test dataset to be used for evaluation of decision systems. When input data is discretized, both learning and test data must be processed. The question arises, how to discretize test datasets. The simplest solution is to perform discretization applying the same parameters and methods like for learning data. But the possible artifacts can occur:

- bin ranges (cut points) in learning and test sets can differ,
- a number of bins in test dataset may be different than in the learning one.

The issue mentioned in the first point can raise doubts if evaluation performed in such situation is confident. The fact presented in the second point makes evaluation in discrete domain impossible. So it is necessary to find the way of test datasets discretization without these problems. The possible approach, which was applied in the presented research, contains few steps. Initially the learning and test sets were concatenated together, then discretization was performed for given parameters and resulting data was split to obtain teaching and test sets. The only problem is that test data, which should be totally independent from learning one, was processed in such way that an influence of teaching data on test one was possible. But it must be investigated separately and was not considered in the current research.

3.3. Classifiers

In order to deeply investigate the influence of representation of discretized data on performance of decision systems, four classifiers were selected: Naive Bayes, Bayesian network, *k*-NN, and decision tree C4.5. Each classifier was built using the given learning datasets and then evaluated applying test datasets. During the preliminary stage nondiscretized data was used to obtain reference results. Then series of experiments for different discretization methods, parameters, and for both ways of representation of discrete data were performed. Because input datasets were prepared and processed separately for male and female authors final outcomes were averaged to obtain results for further analysis. As the measure of classification efficiency the percent of correctly classified instances was selected. For supervised FId and Kd methods the single value for each classifier were obtained, given the nominal or numerical representation of discretized data. For parametrized EWd, EWOd and EFd for each case the series of data were obtained as the function of the given number of bins parameter. In case of *k*-NN classifier experiments were additionally iterated, and a given number of neighbors parameter *k* ranged from 1 to 180. As the result two-dimensional

matrices were created. To obtain one dimensional data comparable with results of other classifiers, the data series for k parameter containing maximum efficiency value were selected.

4. Results and discussion

The first few experiments were performed for data without discretization to obtain a reference point for discussion of further results. The values obtained are presented in Table 1 in the column "No discretization". The other columns present results describing efficiency of subsequent classifiers and ways of data representation for supervised FId and Kd discretizations. The preliminary observations allow to state that for almost all cases performance of classifiers is slightly better for discrete representation of data or at least it is equal. The only exception is k-NN classifier which performed better for numerical data, providing FId method. It is also worth of notice that performance of all classifiers was far better comparing to results obtained for non-discretized input data.

Table 1. Results of experiments obtained for all classifiers using numerical representation for non-discretized data, and nominal and numerical representation of discretized data for input sets processed using FId and Kd.

Classifier	Data representation	No discretzation	Fayyad & Irani	Kononenko
Naive Bayes	numerical	86.94	92.50	92.22
Naive Bayes	discrete		93.06	92.22
Bayesian Net	numerical	86.11	93.06	89.72
Bayesian Net	discrete		93.06	92.22
decision tree C4.5	numerical	75.28	84.44	84.44
decision tree C4.5	discrete		85.56	85.00
k-NN	numerical	90.00	93.06	90.83
k-NN	discrete		92.50	93.06

Figure 1 presents results obtained for parametrized EWd, EWOd, and EFd discretization methods. To clearly visualize data the boxplot diagrams were prepared, one for each method of discretization. Because of the character of outcome data such way of representation seems to be the most suitable for comparison purposes.

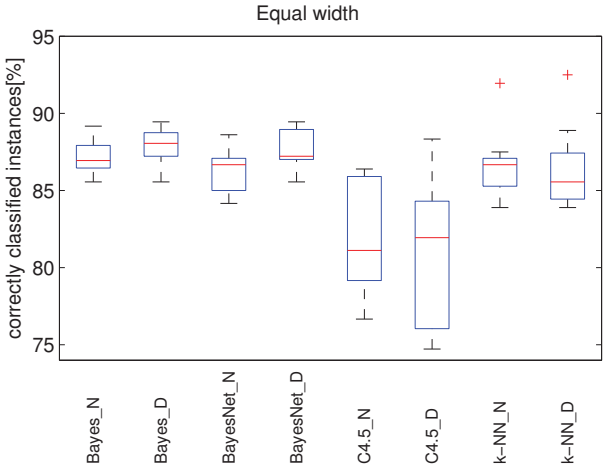
Preliminary observations of median levels of subsequent boxplots confirm dependencies described for supervised discretization methods. Results for Naive Bayes, Bayesian network, and C4.5 classifiers are better (or almost equal for some cases: EFd and Bayesian net; EWOd and C4.5) for nominal representation od discretized data. The k-NN classifier performs better for numerical data which is especially strongly visible for EFd. When analyzing information delivered by boxplots more globally the most unequivocal conclusions can be formulated for Naive Bayes and Bayesian network classifiers. Both present strong tendency to perform better using discrete representation of discretized datasets, especially for EWd and EWOd methods. For EFd discretization that statement is valid for Naive Bayes, whereas Bayesian network seems to behave comparably for both representations of data.

The behavior of C4.5 classifier is more ambiguous. Medians of boxplots present tendency to promote discrete representation of discretized data. On the other hand almost all boxplots for C4.5 are widest, comparing with other classifiers. Which means that classification performance varies quite strongly, given discretization method parameters, and possibly both data representations can be taken into consideration during decision system creation and adjusting.

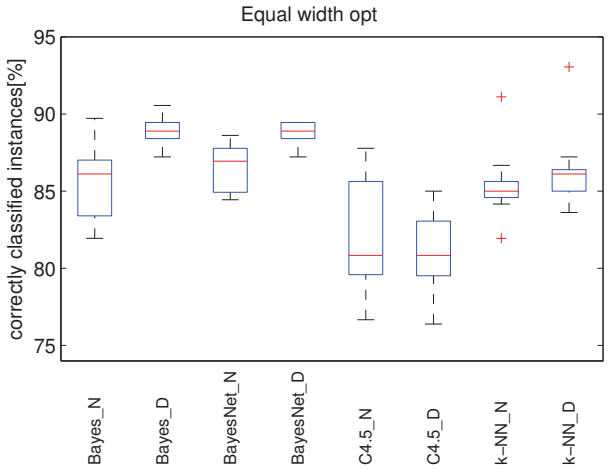
The results obtained for k-NN classifier also does not allow to formulate general conclusions. The most explicit outcomes were obtained for EFd, where dominance of numerical representation of discretized data is visible. Such trend is also proved (but not so strongly) by classifier performance measurements delivered for EWd and FId methods. But EWOd and Kd discretization algorithms behave inversely. There also exist outliers for EWd which present better classifier efficiency for discrete representation.

Concluding the presented observations, it can be stated that globally looking nominal representation of discretized data delivers better results for many of the presented classifiers and for almost all discretization methods. So if one does not want to go deeper into analysis of possible results, such representation would be suggested especially in combination with Naive Bayes and Bayesian network. Use of k-NN classifier can be also taken into consideration because in specific conditions it can perform well for discretized data delivered in nominal representation. The C4.5 classifier seems to be more universal and able to work properly for both representations. It is important to remember

a)



b)



c)

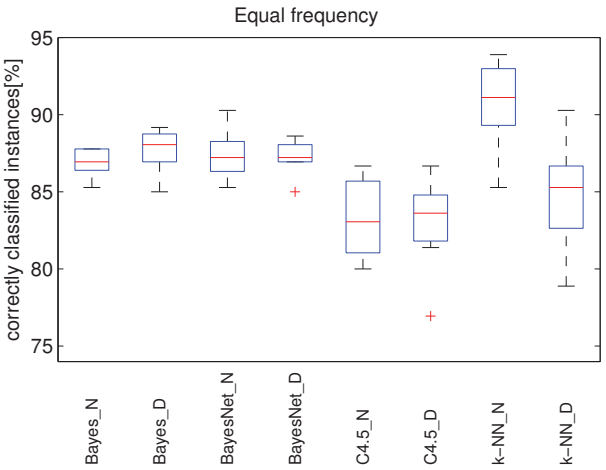


Fig. 1. Classifiers performance for: a) EWd, b) EWOd, c) EFd discretization. Suffix "_D" after classifier's names denotes results for nominal (discrete) representation of discretized data, suffix "_N" denotes results for numerical representation of data.

that presented results and conclusions are founded on the research performed in stylometry domain for specific data. They may not be still valid in other domains and their application should be preceded by deeper analysis.

5. Conclusions

The paper presents results of research on influence of representations of discretized data on performance of decision systems. Two representations of data were taken into consideration: nominal – in the form obtained as the output of the discretization process, and numerical – where operation of conversion of discretized data was additionally applied. Analysis was performed for selected discretization methods, such as: equal width, optimized equal width, equal frequency, Fayyad & Irani MDL, and Kononenko MDL. The following classifiers were utilized: Naive Bayes, Bayesian network, decision tree – C4.5, and *k*-Nearest Neighbors – *k*-NN. The research was performed in stylometry domain implementing authorship attribution tasks.

A global study of different combinations of discretization methods, approaches to representations of discretized data and types of classifiers leads to conclusion that nominal representation of discretized data is generally speaking more recommended. Only selected cases, like *k*-NN classifier applied for data discretized using equal frequency, equal width or Fayyad & Irani's MDL binning, delivered better results for numerical data. But, as aforementioned, some of them can perform well also for nominal data representation, providing carefully selected parameters of decision systems.

The decision tree C4.5 classifier was assessed as the most universal solution to be applied for sets containing data in numerical as well as nominal representation. Both types of representation delivered comparable results but there were strong relations between parameters applied to the discretization process and classification performance. The range of obtained quality results was relatively wide. Therefore C4.5 classifier can be applied for both data representations, but discretization method and its parameters should be thoroughly investigated for each specific task in order to obtain satisfactory effects.

Summarizing, the research proved better suitability of nominal representation of discretized data in the vast majority of analyzed examples. On the other hand, some cases showed usefulness of numerical representation. In some situations both approaches were comparable. So if numerical data representation must be used due to other conditions, it can be applied without hesitation, but the decision system must be adjusted more carefully. Furthermore, discretization should be considered as an element of the decision system which improves quality of data exploration, so its employment is recommended.

Acknowledgements

The research described was performed at the Silesian University of Technology, Gliwice, Poland, in the framework of the project BK/RAu2/2016. All experiments were performed using WEKA workbench[7].

References

1. Baron, G.. Influence of data discretization on efficiency of Bayesian Classifier for authorship attribution. *Procedia Computer Science* 2014; **35**(0):1112 – 1121. doi:http://dx.doi.org/10.1016/j.procs.2014.08.201.
2. Kotsiantis, S.B.. Supervised machine learning: A review of classification techniques. In: *Proceedings of the 2007 Conference on Emerging Artificial Intelligence Applications in Computer Engineering: Real Word AI Systems with Applications in eHealth, HCI, Information Retrieval and Pervasive Technologies*. Amsterdam, The Netherlands: IOS Press; 2007, p. 3–24.
3. Stańczyk, U.. Rule-based approach to computational stylistics. In: Bouvry, P., Kłopotek, M., Marciniak, M., Mykowiecka, A., Rybiński, H., editors. *Security and Intelligent Information Systems*; vol. 7053 of *LNCS (LNAI)*. Berlin Heidelberg: Springer Verlag; 2012, p. 168–179.
4. Stańczyk, U.. On performance of DRSA-ANN classifier. In: Romay, M., Corchado, E., Garcia-Sebastian, M., editors. *Hybrid Artificial Intelligence Systems. Part 1*; vol. 6679 of *Lecture Notes in Artificial Intelligence*. Springer-Verlag; 2011, p. 172–179.
5. Dougherty, J., Kohavi, R., Sahami, M.. Supervised and unsupervised discretization of continuous features. In: *Machine Learning: Proceedings of the 12th International Conference*. Morgan Kaufmann; 1995, p. 194–202.
6. Kotsiantis, S., Kanellopoulos, D.. Discretization techniques: A recent survey. *International Transactions on Computer Science and Engineering* 2006;**1**(32):47–58.
7. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.. The weka data mining software: an update. *SIGKDD Explorations* 2009;**11**(1):10–18.

8. Fayyad, U.M., Irani, K.B.. Multi-interval discretization of continuous-valued attributes for classification learning. In: *Proceedings of the 13th International Joint Conference on Artificial Intelligence (IJCAI)*. 1993, p. 1022–1029.
9. Kononenko, I.. On biases in estimating multi-valued attributes. In: *14th International Joint Conference on Articial Intelligence*. 1995, p. 1034–1040.
10. McCallum, A., Nigam, K.. A comparison of event models for Naive Bayes text classification. In: *AAAI-98 Workshop On Learning For Text Categorization*. AAAI Press; 1998, p. 41–48.
11. John, G., Langley, P.. Estimating continuous distributions in Bayesian classifiers. In: *Proceedings of the 11th Conference on Uncertainty in Artificial Intelligence*. Morgan Kaufmann; 1995, p. 338–345.
12. Theodoridis, S., Koutroumbas, K.. *Pattern Recognition, Fourth Edition*. Academic Press; 4th ed.; 2008.
13. Witten, I.H., Frank, E., Hall, M.A.. *Data Mining: Practical Machine Learning Tools and Techniques*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 3rd ed.; 2011.
14. Cooper, G.F., Herskovits, E.. A bayesian method for the induction of probabilistic networks from data. In: *MACHINE LEARNING*. 1992, p. 309–347.
15. Quinlan, J.R.. *C4.5: Programs for Machine Learning*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.; 1993.
16. Stańczyk, U.. Ranking of characteristic features in combined wrapper approaches to selection. *Neural Computing and Applications* 2015; **26**(2):329–344.