Voting Classifier Using Discretisation in Aggregating Decisions

Urszula Stańczyk
Silesian University of Technology
Department of Computer Graphics Vision and Digital Systems
Gliwice, Poland *urszula.stanczyk@polsl.pl*

Beata Zielosko
University of Silesia in Katowice
Faculty of Science and Technology/Institute of Computer Science
Sosnowiec, Poland *beata.zielosko@us.edu.pl*

Grzegorz Baron
Silesian University of Technology
Department of Computer Graphics Vision and Digital Systems
Gliwice, Poland *grzegorz.baron@polsl.pl*

Abstract

In popular approaches to classification by aggregating decisions, there are two main trends. One path leads to the construction of a classifier ensemble, where a group of diversified inducers vote on a label to be assigned a sample. The second direction is to obtain a decision based on dispersed data, through some form of information fusion. The paper proposes a new mode of operation for a voting classifier, where one and the same inducer can reach a final decision relaying on labels assigned through partially dispersed data, but also different forms of the same data, resulting from discretisation. The experiments were carried out on several datasets, classifiers, and algorithms for aggregating decisions. They resulted in observation of cases and scenarios for improved predictions, showing the merits of the presented research methodology.

Keywords: Discretisation, Aggregated Decision, Voting Classifier, Distributed Data

1. Introduction

Efficient development of information systems requires making the best of available data to arrive at the most informed decisions, crucial especially for business applications. Nowadays, the input data are often not only distributed but also include continuous and categorical attributes. Representation of attributes can reflect on their relevance for a task [9]. Transformations of features, such as discretisation, can affect importance of variables and performance [8]. If supervised discretisation is conducted, it can evaluate attributes, finding some of them as not helping distinction of classes. Such attributes can be treated as irrelevant in a discrete domain. On the other hand, this characterising property of discretisation can be used to support a voting classifier [3].

An inducer collecting information from multiple sources can rely on suggestions from a group of classifiers. Each base estimator gives its opinion and this ensemble votes on the final decision [6]. The data on which individual components of a verdict operate can be dispersed and diversified. Varied data formats enable to capture more information, as specific modes of handling attributes and their domains can be better adapted to learning from continuous or discrete values. A classifier capable of dealing with both numeric and categorical features can report different performance for the same data, depending on the representation.

In the paper, a new voting classifier was proposed, based on dispersed data. The input features are divided into two groups with the help of a supervised discretisation algorithm [13]. The variables, which after transformation receive only one bin, are separated from those for which multiple intervals are found. All attributes are considered in the continuous or discrete domain. To achieve that, the support of unsupervised discretisation methods is needed. To reach a decision, the inducer refers to both categories of features, and performs simple majority voting in one of defined formulas. The research methodology takes into account the issues of discretisation, but also the nature of the data itself is considered—that is why various voting methods are proposed. Algorithms for aggregating decisions are with voting in a single level or two levels, and preference of the information discovered in continuous or discrete domain.

The effectiveness of the new voting classifier was validated through experiments involving several datasets and three popular inducers, the Bayesian Network, J48, and the k-Nearest Neighbours. Analysis of the results allowed to observe conditions where the proposed mode of operation was advantageous and showed the merits of the research framework.

The paper is organised as follows. Section 2 presents the fundamental notions of classifiers aggregating decisions and the description of the novel mode of operation of a voting inducer. Section 3 provides an explanation of the experiments. Section 4 contains comments on the results obtained. Section 5 concludes the paper and indicates future research paths.

2. Classifiers aggregating decisions

To classify samples in the simplest case, a learner can rely on its own mode of operation and approach to knowledge discovery. In more complex scenarios, data could be diversified or distributed and multiple methods can be involved in exploration, with some voting strategies employed. The section presents popular approaches and a description of a new voting classifier.

2.1. Standard approaches

A classifier ensemble is a machine learning approach that combines predictions from multiple models. The basic idea is to obtain the suggestions of multiple classifiers on the original data and combine them to form the final decision, possibly using some voting schemes involving simple majority (hard voting) or weighted votes (soft voting) [4]. The main goal of such processing is to reduce the misclassification of weak inducers and obtain a strong predictor [1], decrease the risk of over-fitting, and increase the stability of predictions by aggregating the results of multiple models. Popular types of ensemble methods include boosting, bagging, and stacking [6].

In stacking, different models are trained independently and their outputs become inputs to a “meta-model” that makes the final prediction. In Bootstrap Aggregating (known as bagging), multiple models are trained on subsets of data, sampled with replacement [3]. Boosting improves model accuracy by sequential training of base models. Each base model is trained in a way that emphasises the examples misclassified by previous models. Therefore, more weight is given to misclassified samples so that subsequent models focus on these difficult cases.

2.2. Proposed mode of operation for a new voting classifier

Two considerations provided motivation for the operation of a voting classifier: i) discretisation can result in attribute reduction, and ii) depending on the properties of a particular inducer, discretisation can have a strong impact on the observed performance. The new voting classifier works on partially distributed data and on the same data represented in various forms, as follows.

Let a denote the available attributes. a_m gives variables that are assigned multiple bins in supervised discretisation, and a_1 are features with a single categorical representation found. Let R_a denote attributes with continuous domains and C_a with discrete domains. Then R_{a_m} means all features with multiple intervals found by supervised discretisation, but represented in the continuous domain, while C_{a_1} reflects some discrete representation for single bin variables.

A classifier aggregates the final decision for a sample based on several voting scenarios. In each, the attributes are distributed, and a_m and a_1 groups still represented in continuous input domain or discretised. There are three main components with single and equal votes, so a tie is not possible. The simple majority results in the assigned label:

- $R_{a_m}C_{a_m}R_{a_1}$—aggregation on a single level, a_m part taken both in continuous and discrete form, a_1 only in the continuous domain. The a_m part has two votes over the one of a_1, so the role of the latter is visible only when R_{a_m} and C_{a_m} lead to different decisions;
- $R_{a_m}C_{a_m}C_{a_1}$—aggregation on a single level, a_m part taken both in continuous and discrete form, a_1 part in a discrete domain obtained after transformation with unsupervised discretisation algorithm, therefore there are several variants possible;
- $R_{a_m}C_{a_m}(C_{a_1})$—aggregation on two levels, a_m part taken both in continuous and discrete form, a_1 part in discrete domains obtained after transformation with unsupervised discretisation algorithm. From all discrete variants possible for a_1 part, through simple majority voting a decision is obtained first and then this is provided as one of the three main components of the decision at the second level.

Depending on the number and type of discretisation methods used, apart from indicating multi- and single-interval attributes, the specific approach could also be indicated. Unsupervised algorithms work based on the input parameter, typically providing the number of bins to be constructed for the transformed variables, which needs to be given to avoid any ambiguities.

3. Experimental setup

The new approach to aggregating decisions for a voting classifier needed experimental validation. The section details the input data explored, the methods for attribute transformations employed, and the algorithms used for data mining.

3.1. Input datasets

To provide a wider scope for observations and minimise the number of influential factors that could bias investigations, in the research three pairs of datasets were constructed. The first two pairs were chosen from the datasets available in the UCI Machine Learning Repository [10]. Avila1 and Avila2 are based on samples selected from the Avila dataset [5]. It is dedicated to the task of associating specific patterns of writing with copyists of the Bible produced in the XIIth century. Wave1 and Wave2 refer to the Waveform Database Generator (version 1) [4]. Different waveforms can be generated as a combination of two or three base waves. The third pair of datasets, Style1 and Style2, are dedicated to a task of authorship attribution from the stylometric analysis of texts [12]. Authors of texts are recognised based on the linguistic characteristics of their writing styles, visible regardless of a particular topic.

For stylometric datasets, the performance evaluation by standard cross-validation fails to deliver reliable predictions due to stratification caused by sub-concepts [2]. Instead, averaging accuracy for multiple test sets is used. Therefore, for all datasets the same evaluation strategy was adopted and each contained three sets: one learning set (with 200 samples) and two test sets (90 and 80 samples). All sets were prepared for a binary classification problem with balanced classes. Classes were considered to be of the same importance, with the same costs of misclassification. The input attributes available were continuous. There were no missing values.

3.2. Transformation of features

The input datasets were independently discretised by one supervised and one unsupervised approach [8]. The Fayyad and Irani algorithm [7] (denoted dsF) is a supervised method with the main objective of selecting cut points for continuous attributes in such a way as to minimise entropy in each interval, which ensures that the data in each bin are as homogeneous as possible.

The algorithm starts working by assigning all attribute values to one interval and then dividing this interval into smaller ones until further division does not result in a sufficient increase in information gain. Equal frequency binning (denoted duf) is an unsupervised discretisation method. It divides the attribute values into intervals so that each bin contains approximately the same number of observations. The number of bins is the input parameter.

For the proposed methodology, an important consideration was the existence of variables for which a supervised discretisation algorithm assigned a single categorical representation in a discrete domain. These characteristics of the datasets used in the investigations are shown in Table 1. It can be observed that the number of 1 bin features was close to 50%. Without additional transformations, these variables would be treated as irrelevant and their informative content in the continuous domain would be lost.

Table 1. Characteristics of attributes in datasets found by discretisation with Fayyad & Irani method.

	Dataset					
	Avila1	Avila2	Wave1	Wave2	Style1	Style2
Number of attributes: Total/1 bin	10 / 4	10 / 5	21 / 8	21 / 9	12 / 6	12 / 7

3.3. Classifiers employed in investigations

There are many state-of-the-art inducers that can be used in data mining tasks. In the research, three selected classifiers were employed: the Bayesian Network (BNet), J48, and the k-Nearest Neighbours (k-NN). All are implemented on the popular WEKA platform [16].

The BNet classifier uses the Bayes' theorem to determine the probability of predicting the class to which a given object belongs [9]. This probabilistic model is represented as a directed graph. The nodes correspond to the attributes, and the edges indicate the conditional dependencies between the variables. The result of the classifier's operation is the assignment of a new observation to the class for which the highest probability has been determined.

J48 is an implementation of the C4.5 algorithm [11]. It uses a tree representation. Each internal node corresponds to an attribute, and each terminal node (leaf) corresponds to a class label. The selection of attributes that form the nodes is based on the information gain measure.

k-NN belongs to the group of lazy learning algorithms because, during the classification process, the algorithm does not explicitly learn a model. The main idea is based on calculating the distance (similarity) between the objects studied and determining the k nearest neighbours who will participate in the process of determining the class label for the tested object.

As a measure of inducer performance, the classification accuracy was selected [15]. It reports the percentage of correctly labelled samples, regardless of class. The accuracy was averaged over the results obtained for the two test sets present in each dataset.

4. Obtained results

The results of the experiments were mainly analysed with respect to the specific algorithm for aggregating decisions of a voting classifier. The preferences of the inducers shown for either numeric or categorical attributes, reflected in the reported performance, were also studied.

4.1. Reference points

Firstly, some reference points were established. They included the performance for the input data in the continuous domain and when it was discretised with the supervised Fayyad and Irani algorithm. The results given in Table 2 show that for all classifiers, the change of representation of the input data in some cases (but not all) gave better predictions. For the Avila datasets, all inducers suffered in consequence of discretisation. The efficiency of BNet recognition was improved for the two Waveform datasets, while J48 and k-NN worked better for the Wave1 and Style2 datasets. The influence of discretisation showed the impact of data irregularities, existing in independently transformed sets and variable domains [14].

Table 2. Performance [%] of inducers operating on datasets in continuous domain (R_a), and in the discrete domain obtained by the supervised Fayyad & Irani algorithm (C_a).

	Dataset											
	Avila1	Avila2	Wave1	Wave2	Style1	Style2	Avila1	Avila2	Wave1	Wave2	Style1	Style2
Inducer	Continuous domain (R_a)						Discrete dsF domain (C_a)					
BNet	87.71	78.82	87.78	83.13	91.60	77.43	80.00	63.33	90.00	84.79	50.00	67.22
J48	86.94	82.99	83.19	87.22	89.79	75.63	83.75	63.33	89.03	76.18	87.78	80.76
k-NN	82.36	70.21	85.63	86.46	86.18	77.92	80.00	66.25	85.83	72.57	62.22	83.54

The power of the classifiers operating on data discretised by equal frequency binning (duf) was another reference point. The number of bins to be constructed ranged from 2 to 10. It resulted in nine discrete data variants, as displayed in Fig. 1. The categories in the X axis correspond to the inducers and the series specify the number of bins. Unsupervised discretisation for many conditions proved to be more advantageous than supervised transformations. For both the Avila datasets, Wave2 and Style1, all learners returned enhanced accuracy. The J48 classifier achieved a lower ratio of correct predictions for Wave1 and Style2 in all duf domains. The k-NN improved for Wave1 but returned degraded performance for Style2, while the BNet did the opposite. For further comparisons with voting classifiers, only the maximal or minimal classification accuracy detected among all discrete variants was referred to.

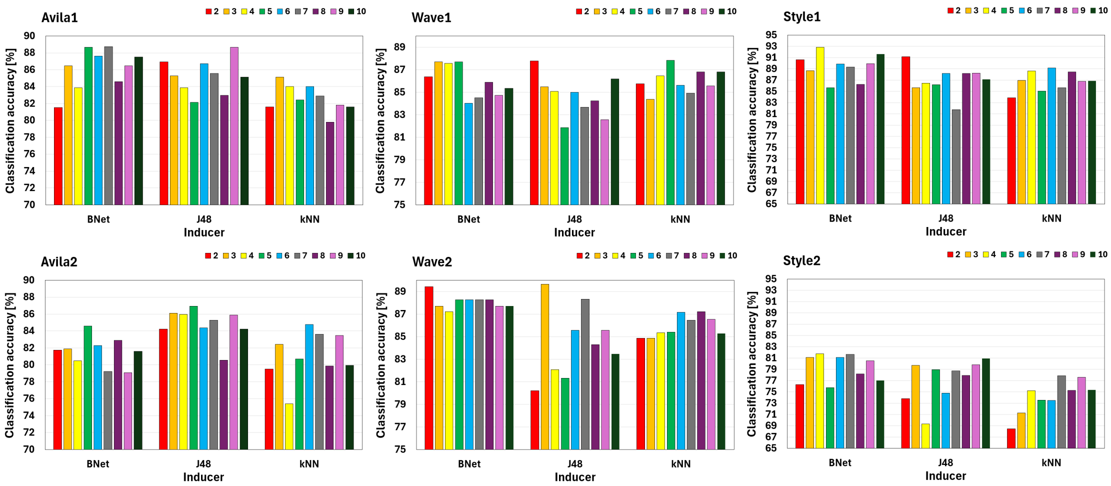

Fig. 1. Performance [%] of inducers operating on all datasets transformed with unsupervised equal frequency binning of all attributes (C_a).

Another point of reference was the accuracy for a subset of attributes, obtained by rejecting these variables for which single bins were defined in the Fayyad and Irani method, which is shown in Table 3. The feature reduction turned out to be mostly advantageous for the J48 classifier, as only for the Style2 dataset it fared worse than when operating on the entire set of attributes. For the k-NN for three datasets (Avila2, Wave1, Style1), the improvement was noted, and for the other three the decreased predictions were returned. The BNet reported exactly the same results as when working on all available features.

All of these results (four groups) were treated as reference points. Once they were established, the next part of the experiments was dedicated to testing the proposed mode of operation of the voting classifiers with the defined scenarios.

Table 3. Performance [%] of inducers operating on datasets in continuous domain reduced to these attributes that after supervised discretisation were represented by multiple intervals (R_{a_m}).

	Dataset					
Inducer	Avila1	Avila2	Wave1	Wave2	Style1	Style2
BNet	87.71	78.82	87.78	83.13	91.60	77.43
J48	87.01	86.74	85.35	87.78	92.22	74.38
k-NN	81.18	74.65	85.97	82.71	91.67	76.11

4.2. Classification by voting

The voting classifiers aggregated decisions based on data in various forms. Depending on a formula, the attributes found as multi-bin by the Fayyad and Irani algorithm (a_m), were used in their continuous (R_{a_m}) or categorical representation (C_{a_m}). The attributes for which supervised discretisation constructed only single bins (a_1) could be processed in the continuous domain, but operation on their non-trivial discrete form required unsupervised discretisation.

Table 4 includes the results for voting classifiers using multi-bin variables in both numerical and discrete form, and 1 bin variables in the continuous domain. This algorithm can be treated as a classifier working mainly in the continuous domain, but with distributed data, and with suggestion from the part of multi-bin features reinforced by their discrete form.

Table 4. Performance [%] of inducers voting based on dispersed data, by continuous and discrete form for multi-interval attributes and single bin variables in continuous domain ($R_{a_m} C_{a_m} R_{a_1}$)

	Dataset					
Inducer	Avila1	Avila2	Wave1	Wave2	Style1	Style2
BNet	88.82	65.14	90.76	84.24	91.60	66.11
J48	87.50	78.75	88.40	80.63	91.53	79.24
k-NN	87.85	76.94	88.75	79.79	80.90	83.40

For this voting algorithm, the BNet classifier outperformed itself with respect to all reference points for the Avila1 and Wave2 datasets. On the other hand, for the Style2, the performance was below those previously observed. For Style1 the predictions were at the same level as established for working in the original continuous domain for all variables or with the reduced set, while it was improved over the dsF domain but degraded with respect to the duf domains.

For J48 and k-NN inducers, the advantage of voting was visible compared to regular supervised discretisation for all datasets but Style2. It turned out to be disadvantageous for the J48 when compared to all features represented in some duf domain for the Avila and Wave datasets pairs. Operation on all variables in the continuous domain brought better results than the voting scenario studied for J48 only for Avila2 and Wave2, and for the reduced attribute set for Avila2, Wave2, and Style1. For the k-NN this voting was undeniably beneficial when operating on the Wave1 dataset, and for Wave2 and Style1 the improvement was only over data in the dsF domain. For Avila2, only duf discretisation of all features returned better classification, while for Avila1 it happened for duf but also for a_m attributes in continuous form. In the case of the Style2 dataset, this voting caused outperforming the reference points in the continuous domain for all and reduced variables, and worse results than C_a in the dsF and duf domains.

The second voting scenario also involved one level of aggregating decisions. It differed from the first in relying on unsupervised discretisation of a_1 attributes. As before for the reference points, equal frequency binning was employed, ranging the number of bins constructed from 2 to 10. The results are shown in Fig. 2, where it can be noted that for the Bayesian Network the performance was independent on the number of intervals formed for variables, and it was the same as for the first voting formula. For the other two inducers, some variations were visible.

This voting formula led to the J48 classifier returning better performance than the first scenario for all six datasets for some conditions, depending on the number of constructed intervals.

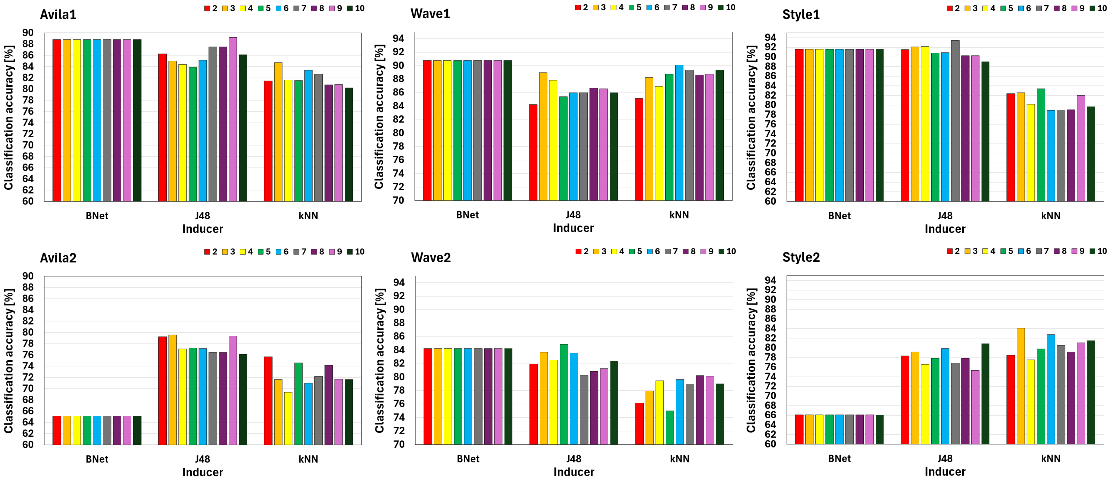

Fig. 2. Performance [%] of inducers voting based on dispersed data, by continuous and discrete form for multi-interval attributes and single bin variables transformed by unsupervised equal frequency binning with varying the number of constructed bins ($R_{a_m}C_{a_m}C_{a_1}$).

For Avila1, the predictions were higher than all reference points apart from the case when all features were transformed with the duf method, and for Avila2 it was higher only with respect to working on the dsF domain, but below other reference points. However, only the representation in the dsF domain was more advantageous for the Wave1 dataset, and the only less advantageous for Wave2. For Style1 this voting led to the best accuracy presented so far, while for Style2 only the duf representation of all attributes measured as high.

For the k-NN classifier, the second voting algorithm was more efficient than the first for the Wave and Style datasets. Yet for Avila1 and Avila2, the accuracy was lower than only one reference point: of exploring data subjected to unsupervised discretisation of all features. For Wave2, the predictions were better only when compared to working in the dsF domain, and the same statement was valid for Style1. For Style2, the accuracy reported for this type of voting was the best from those presented to this point.

The results of the third scenario of voting are included in Table 5. The main difference from the first two algorithms lies in the two levels of aggregating decisions. The first, internal level, leads to the decision agreed upon among all discrete variants of data resulting from transformations of a_1 variables with unsupervised equal frequency binning. There were nine such data variants, and their independent processing returned nine suggestions for a classification verdict, from which a simple majority caused selection that was passed on to the second voting level.

Table 5. Performance [%] of inducers voting based on dispersed data, by continuous and discrete form for multi-interval attributes and single bin variables transformed by unsupervised equal frequency binning, with two levels of aggregating decisions ($R_{a_m}C_{a_m}(C_{a_1})$).

Inducer	**Dataset**					
	Avila1	Avila2	Wave1	Wave2	Style1	Style2
BNet	88.82	65.14	90.76	84.24	91.60	66.11
J48	87.43	79.38	85.97	81.32	90.28	76.04
kNN	82.08	72.22	89.38	79.03	80.07	79.24

For this voting algorithm, BNet reported the same level of predictions as for the two presented above. For J48 and k-NN, the results were worse than for the second scenario discussed, for all datasets. However, the relations with reference points were very similar or the same. The second level of aggregation of decisions by consulting all duf variants of attributes a_1 did not improve the best cases detected when these data versions were explored independently.

4.3. Discussion of results

The best cases obtained in the experiments are included in Table 6. As a measure of quality, the maximal classification accuracy was taken, and conditions are provided when it was detected. If it took place for a variant of unsupervised discretisation, the number of defined bins is listed.

Table 6. The best performance [%] of inducers with conditions where it occurred.

Inducer	Dataset					
	Avila1	Avila2	Wave1	Wave2	Style1	Style2
BNet	88.88	84.58	90.76	89.44	92.85	81.74
	All voting scenarios	C_a-duf5	All voting scenarios	C_a-duf7	C_a-duf4	C_a-duf4
J48	89.17	86.94	89.03	89.65	93.40	80.90
	$R_{a_m}C_{a_m}C_{a_1}$-duf9	C_a-duf5	C_a-dsF	C_a-duf3	$R_{a_m}C_{a_m}C_{a_1}$-duf7	$R_{a_m}C_{a_m}C_{a_1}$-duf10
k-NN	87.85	84.79	90.07	87.22	91.67	84.10
	$R_{a_m}C_{a_m}R_{a_1}$	C_a-duf6	$R_{a_m}C_{a_m}C_{a_1}$-duf6	C_a-duf8	R_{a_m}	$R_{a_m}C_{a_m}C_{a_1}$-duf3

For the BNet and Avila1 and Wave1, the highest accuracy was detected in the voting scenarios, the same for all. For the remaining datasets, the best performance was observed in the regular operation of the classifier for a variant of data received from unsupervised discretisation. The J48 reported the best predictions for the $R_{a_m}C_{a_m}C_{a_1}$ formula in the case of Avila1 and both Style datasets. For the Wave datasets and Avila2, the highest accuracy occurred when the attributes were discretised either by supervised or unsupervised approach. For the k-NN, two voting scenarios improved performance over reference points, $R_{a_m}C_{a_m}R_{a_1}$ and $R_{a_m}C_{a_m}C_{a_1}$, the former for Avila1 and the latter for Wave 1 and Style2. The most advantageous voting in this case was not only the one referring to duf variants of a_1 attributes, but also relying on their numerical representation. For Style1 the reduced numerical attributes caused the highest accuracy, whereas for Avila2 and Wave2 unsupervised discretisation turned out to be the most beneficial.

The experimental results show some conditions where the new voting scenarios caused decreased powers of the inducers employed, as well as the cases of improved predictions. These observations confirmed that the proposed framework is worth a deeper study and validated the new operation mode of a voting classifier, based on dispersed data and transformations of the input space. It can be adapted to any inducer, but the application is limited to continuous input data with a noticeable number of 1 bin attributes obtained from supervised discretisation.

5. Conclusions

The paper presents research works dedicated to a new mode of operation for a voting classifier, focused on various forms of the input data resulting from discretisation. Changes in the representation of attribute domains can affect patterns hidden in the data and therefore also the knowledge that can be discovered in data exploration. The classifier operates on dispersed data, obtained with the help of the evaluating property of a supervised discretisation algorithm.

To verify the usefulness of the proposed classifier, experiments were carried out on several datasets, learners, and algorithms to aggregate the decisions. The final decision was reached by combining suggestions based on groups of attributes, for which multiple or single intervals were constructed, with reference to either the original continuous domain or discrete domains. Voting was performed in one level or two, and preference was given to different representations.

The experimental results allowed to observe scenarios where the proposed approach brought some improvement and showed its advantages. This work is an introduction to further data-related research, in order to propose a voting method that will account for the specificity of data and allow for proposing a more universal solution in the context of distributed data, which is important for development of efficient information systems that need to reach informed decisions.

In future work, the proposed voting approach will be tested for other estimators with different mathematical foundations. In addition, a range of discretisation algorithms will be extended.

Acknowledgements

The research works presented were carried out in the Department of Computer Graphics, Vision and Digital Systems (RAU-6, 2024), at the Silesian University of Technology, Gliwice, Poland, and at the Institute of Computer Science, University of Silesia in Katowice, Sosnowiec, Poland.

References

1. Baron, G.: Analysis of multiple classifiers performance for discretized data in authorship attribution. In: Czarnowski, I., et al. (eds.) Proceedings of the KES-IDT 2017, Smart Innovation, Systems and Technologies, vol. 72, pp. 33–42. Springer (2018)
2. Baron, G., Stańczyk, U.: Standard vs. non-standard cross-validation: evaluation of performance in a space with structured distribution of datapoints. In: Wątróbski, J., et al. (eds.) Proceedings of the KES-2021, Procedia Computer Science, vol. 192, pp. 1245–1254. Elsevier (2021)
3. Bauer, E., Kohavi, R.: An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. Machine Learning **36**(1), 105–139 (1999)
4. Breiman, L., Friedman, J., Olshen, R., Stone, C.: Classification and Regression Trees. Wadsworth International Group, Belmont, California. (1984)
5. De Stefano, C., Maniaci, M., Fontanella, F., Scotto di Freca, A.: Reliable writer identification in medieval manuscripts through page layout features: The "Avila" Bible case. Engineering Applications of Artificial Intelligence **72**, 99–110 (2018)
6. Dietterich, T.G.: Ensemble methods in machine learning. In: Multiple Classifier Systems. pp. 1–15. Springer Berlin Heidelberg, Berlin, Heidelberg (2000)
7. Fayyad, U., Irani, K.: Multi-interval discretization of continuous valued attributes for classification learning. In: Proceedings of the 13th International Joint Conference on Artificial Intelligence. vol. 2, pp. 1022–1027. Morgan Kaufmann Publishers (1993)
8. Garcia, S., Luengo, J., Saez, J., Lopez, V., Herrera, F.: A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning. IEEE Transactions on Knowledge and Data Engineering **25**(4), 734–750 (2013)
9. Han, J., Kamber, M., Pei, J.: Data Mining: Concepts and Techniques. Morgan Kaufmann (2011)
10. Kelly, M., Longjohn, R., Nottingham, K.: The UCI Machine Learning Repository. https://archive.ics.uci.edu
11. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA (1993)
12. Rybicki, J., Eder, M., Hoover, D.: Computational stylistics and text analysis. In: Crompton, C., Lane, R., Siemens, R. (eds.) Doing Digital Humanities: Practice, Training, Research, pp. 123–144. Routledge, 1 edn. (2016)
13. Stańczyk, U.: Evaluating importance for numbers of bins in discretised learning and test sets. In: Czarnowski, I., et al. (eds.) Proceedings of the 9th KES-IDT International Conference 2017 – Part II, Smart Innovation, Systems and Technologies, vol. 72, pp. 159–169. Springer (2018)
14. Stańczyk, U., Zielosko, B.: Data irregularities in discretisation of test sets used for evaluation of classification systems: A case study on authorship attribution. Bulletin of the Polish Academy of Sciences: Technical Sciences **69**(4), 1–12 (2021)
15. Stąpor, K., Ksieniewicz, P., García, S., Woźniak, M.: How to design the fair experimental classifier evaluation. Applied Soft Computing **104**, 107219 (2021)
16. Witten, I., Frank, E., Hall, M.: Data Mining. Practical Machine Learning Tools and Techniques. Morgan Kaufmann, 3rd edn. (2011)