Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 192 (2021) 3560–3569

Procedia Computer Science

www.elsevier.com/locate/procedia

25th International Conference on Knowledge-Based and Intelligent Information & Engineering Systems

Stop Criterion in Building Decision Trees with Bagging Method for Dispersed Data

Małgorzata Przybyła-Kasperek*, Samuel Aning

University of Silesia in Katowice, Institute of Computer Science, Będzińska 39, 41-200 Sosnowiec, Poland

Abstract

This article discusses issues related to decision making based on applying decision trees and bagging methods on dispersed knowledge. In dispersed knowledge, local decision tables possess data independently in fragments. In this study, sub-tables are further generated with bagging method for each local table, based on which the decision trees are built. These decision trees classify the test object, and a probability vector is defined over the decision classes for each local table. For each vector, decision classes with the maximum value of the coordinates are selected and final joint decisions for all local tables are made by majority voting. Quality of decision making has been observed to increase when bagging method as an ensemble method is combined with decision trees on independent dispersed data. An important criterion in building a decision tree is to know when to stop growing the tree (stop splitting). That is, at what minimum number of objects on a working node do we stop building the tree to ensure the best decision results. The contribution of the paper is to observe the influence a stop criterion (expressed in the number of objects in the node) for decision trees used in conjunction with bagging method on independent data sources. It can be concluded that in dispersed data set, the stop split criteria does not influence the classification quality much. The statistical significance of the difference in the mean classification error values was confirmed only for a very high stop criterion (0.1× number of objects in training set) and for a very low stop criterion (equal to two). There is no significant statistical difference in the classification quality obtained for the stop criterion values: 4, 6, 8 and 10. An interesting remark is that for some dispersed data sets, in the case of smaller number of local tables and larger number of bootstrap samples, better quality of classification is obtained for a small number of objects in the stop criterion (mostly for two objects). Only, at a significant increase in the minimum number of objects at which growth of trees is stopped is quality of classification affected. However, the gain in reducing the complexity for trees that we get when using the larger values of stop criterion is significant.

Keywords: Ensemble of classifiers; Dispersed data; Stop criterion; Bagging method; Classification trees, Independent data sources

* Corresponding author. Tel.: +48-32-368-97-56 ; fax: +48-32-368-97-60.
E-mail address: malgorzata.przybyla-kasperek@us.edu.pl; samuel.aning@us.edu.pl

10.1016/j.procs.2021.09.129

1. Introduction

More recently, problems occurring in machine learning involves developing machine learning algorithms that tackle specialized data types and needs. For instance, the development of federated learning technique enables global machine learning algorithm to be built using data across multiple local severs without sharing data among local servers or central servers. In such doing effective model can be build using large data whiles solving the challenge of data privacy and security. Fog learning [1, 5] further advances on federated learning where local data is saved on large-scale networks of heterogeneous devices. In this article the focus is on dispersed data type. In terms of data used, in federal learning and fog learning, local data (called local tables) do not share any common objects whereas in dispersed data local tables may share some common objects. Dispersed data is mostly common in the fields of medicine, enterprise management, business, among others. Dispersed data are identified by decision tables called local tables [12, 13]. Each of these tables is defined and created by a separate unit. The definitions of local tables do not have to be consistent because the units are independent. Local tables may have different or common conditional attributes. They may also have different objects; some may be shared.

However, it is important that all these local tables relate to one field or discipline and therefore they have a common decision attribute. We can find examples of dispersed data in many different domains. For example, the medical field, medical data are collected by many different units. Many hospitals, clinics have units dealing with the same subject, e.g., oncology. Each of the units may have different diagnostic methods/attributes based on which it makes decisions about a disease. However, some methods/attributes may be common in among units. A patient may be diagnosed by all units under consideration. Some patients could be diagnosed simultaneously by several units. The existence of an identifier between such dispersed set could be difficult issue to ensure, in fact, often impossible. More precise algorithm for such dispersed data are been developed to tackle this classification issue using popular prescribed classification.

Decision tree techniques have been widely used to build classification models because models closely resemble human reasoning and are easy to understand [7]. Decision trees are also easy to prepare with less complexity compared to other classification algorithms. A decision tree is built by recursively splitting a node into two or more sub nodes with the intension of creating homogeneity with respect to target variable in each resultant sub-node. There are several proposed algorithms for building decision trees. The best-known algorithm are ID3, C4.5, CART and CHAID [7]. A very important concern in building decision trees has to with deciding when to stop the growth of the tree. In [4], Elposito et al. mentions that there are two major methods to make such decision; namely, prepruning and postpruning. Preprunning involves prospectively deciding when to stop the growth while building the tree whereas postpruning involves retrospectively reducing the size of a fully expanded tree, by pruning some branches. Since the prepruning approach is less computationally complex, it would probably be a better solution for dispersed data. In many articles, the advantages of prepruning have been confirmed based on data from various domains [2, 17].

In [10], the authors applies two prepruning methods; imposing a threshold on a measure (minimum number of objects) to stop tree growth and chi-squared pruning to some classifiable data. It is shown that in the use minimum number of objects pruning method, when minimum number of objects in the leaves is increased, the size of the tree increases however the accuracy of classification is generally unaffected.

Though decision trees are a well acceptable choice for classification, however they are known to be unstable [8]. Thus, a small variation in the training set may result different trees and different prediction. Primarily, independent way of collecting data as it is for all dispersed data helps to improve the stability of classification. Thus, building decision trees and making decisions from independently dispersed data would improve the stability of the classification. The second and most common option is the use of ensemble methods.

In [11], two new approaches on applying decision trees to dispersed data are observed: the approach with decision trees directly generated based on local tables and the approach with the bagging method and decision trees. It was found that the bagging method gives more unambiguous results than the method based on the direct generation of decision trees based on local tables. The objective of this work is to observe further, how imposing a threshold on the minimum number of objects in the node where the tree growth stops for decision trees in conjunction with bagging method on independent data sources affects the quality of classification. It was statistically confirmed that there was no significant difference between the mean classification error for most of the tested stop criteria values (4, 6, 8, 10).

Only for very large and very small values were significant statistical differences noted. While the trees constructed using the higher values of stop criterion are much less complex.

The paper is organized as follows. Section 2 is on growing decision trees, Section 3 is on description of the proposed approach, section 4 addresses the data sets that are used. Section 5 presents the conducted experiments and discussion on obtained results. Section 6 is on conclusions and future research plans

2. Building decision trees

Let $D = (U, A, d)$ be a decision table, where U is the universe, a set of objects, A is a set of conditional attributes and for each $a \in A$ we have $a : U \to V_a$, V_a is a set of values of attribute a, $d \notin A$ is a decision attribute, $d : U \to V_d$. A decision tree is a finite directed tree with the root in which each terminal node (leaf) is labeled with a value of decision, each nonterminal node (such nodes will be called working nodes) is labeled with an attribute from the set of conditional attributes $A = \{a_1, ..., a_n\}$ [9]. Let γ be a decision tree. For any object $x \in U$, the tree works in the following way: if the root of γ is a terminal node labeled with decision class j then j is the result of the tree γ work on the object x. Let the root of γ be a working node labeled with an attribute a_i. Then we compute the value $a_i(x)$ and pass along the edge labeled with $a_i(x)$, in a recursive manner until some decision class j is obtained. We will say that γ solves the considered problem if for any $x \in U$ the result of γ work coincides with the decision class for which x belongs.

CART algorithm proposed by the team Breiman, Friedman, Stone and Olshen [7] and ID3 proposed by Quinlan are among the first decision tree algorithms. In both algorithms, a set of training objects is considered and the optimal division of this set with regard to the value on the conditional attribute is determined. The choice of the best attribute for the current node is a typical example of a greedy algorithm. This procedure is repeated until the obtained set of objects in a node is clean (i.e. objects belong to one decision class) or when there is no possible division to define (i.e. all objects have the same values on conditional attributes) or another stop condition is satisfied. While Gini Index or Twoing criterion is used in the CART algorithm, the Entropy measure is used in the ID3 and C4.5 algorithm in order to determine the optimal division. In the CART algorithm [3] the Gini index used is defined as follows. Let X be the set of training observations belonging to n classes and let p_j denote the relative frequency of occurrence of the j class in the set X. The Gini index for the X set is expressed as $Gini(X) = 1 - \sum_{j=1}^{n}(p_j)^2$, where $p_j = \frac{|C_{j,X}|}{|X|}$, $|X|$ is the size of the training set X and $|C_{j,X}|$ is the number of objects from the j-th decision class. Given the splits of values $\{X_1, X_2\}$, that is defined based on the attribute a, Gini index of a is calculated as follows $Gini_a(X) = \frac{|X_1|}{|X|}Gini(X_1) + \frac{|X_2|}{|X|}Gini(X_2)$. The CART algorithm selects a partition with the minimum Gini index. Twoing criterion can also be used in the CART algorithm. In this measure maximum homogeneity is less important than equal division of the tree. Twoing algorithm is slower than the Gini algorithm but it builds trees that are more balanced.

In the ID3 algorithm [6, 15] the entropy used is defined as follows. Let X be a set of objects that has k possible values on selected attribute, and each of these values occurs with a probability of $p_i, i = 1, \ldots, k$. Entropy expresses the average number of bits that is needed to send a string of symbols representing the observed values in the set X and is given by the formula $E(X) = -\sum_{i=1}^{k} p_i \log_2(p_i)$. The attribute with the smallest entropy is used to split the training data set. Entropy is calculated for the remaining attribute and the procedure repeats itself in an iterative manner.

On growing decision trees, the tree complexity is measured by one of the following metrics: the total number of nodes, total number of leaves, tree depth and number of attributes used [16]. Imposing a threshold on any of these measures where the tree stop growing can be an effective prepruning procedure. A greater number of nodes in a tree implies that objects would be finely divided among these nodes creating a complex tree model. As such minimum number of nodes can be imposed to control tree growth. Imposing a threshold on number of leaves is positively correlated to imposing threshold number of nodes on a tree. Also imposing minimum number of attributes and minimum depth of tree as a metric to control tree complexity are both synonymous because as depth of the tree increases the more attributes are been set up at each node to be used as basis to divide objects.

The number of objects in each leaf is used as a threshold to control tree growth in this paper. Decision trees in each case is built with Python scikit-learn (which uses the CART algorithm) by imposing the minimum number of objects (2, 4, 6, 8 and $0.1 \times number\ of\ objects\ in\ training\ data$) in working node to observe how they effect classification quality.

3. Decision tree classification used on dispersed data

Earlier study [12, 13, 14] on classification in dispersed data mainly used the k-nearest neighbors classifier to generate decisions based on dispersed data. In these articles, the authors further put a lot of effort in determining the coalitions of classifiers and generating aggregated knowledge mainly using Pawlak's conflict model. In combining predicted results, several methods are proposed and applied on dispersed data in these research works.

In this article, decision trees and bagging method are applied to dispersed data. To define dispersed data, we assume that Ag is a set of classifiers and an identifier $ag \in Ag$ is a classifier that is built based on a decision table D_{ag}. A set of decision tables $D_{ag} := (U_{ag}, A_{ag}, d)$, from one discipline is available, where U_{ag} is the universe; A_{ag} is a set of conditional attributes; d is a decision attribute. Lets call the decision tables local tables. All local tables should be related to one discipline (thus they have a set of common decision attributes) whereas conditional attributes and objects in the local tables have no restrictions. We can find examples of dispersed data in many different domains such as banking (data on lending, profitability), biology, chemistry, physics (different research units dealing with the same topic) and many others.

In [11], where decision trees are first used to classify dispersed data, two approaches were considered. First, the approach of building one decision tree based on each local decision table and in second approach, one decision tree is built for each bag of training data (obtained by bootstrap method, selected with replacement). In the first approach each decision tree votes for one decision value while in the case of second approach, the collective results are aggregated into a vector over the decisions. The final decision is made by majority voting. Thanks to this, only if more than half of the classifiers are wrong, we get the wrong decision. For ensemble methods to actually improve the quality of the classification, it is very important that the base classifiers have a classification accuracy greater than the random assignment of decision classes and they should be diversified. Decision trees exhibit such properties and that is why such models work very well here.

In this paper, same experiments are performed on two different versions of the data set - dispersed and non-dispersed data set. However, in the dispersed data two levels of dispersion; first from the dispersed nature of the data sets and then from the creation of bootstraps samples occurs. As such two levels of aggregation is used. In both versions of the data sets, first bootstrap samples (bags) of the decision table (local tables in dispersed set and one global table in non-dispersed set) are created. Decision trees are built for each bag using the CART algorithm where Gini index is used as a measure to split data set in each working node. To classify a test object in the non-dispersed data set, the decision most often observed in the classification of each of the bags is simply used. In the version of dispersed data, for each local table, a vector that aggregates the results of all classification from decision trees generated based on bags of the table is created. Each vector coordinate corresponds to one decision class. In this way we obtain the set of vectors over decision classes with cardinality equal to the number of local decision tables. Final decisions are made by the majority voting. Here, classes that received the maximum number of votes defined by each local table based on all vectors are considered. This method of aggregation may generate ties.

4. Data description

Data used in this experiment are collected from UC Irvine Machine Learning Repository in a non-dispersed form. Three data sets are collected, namely; Vehicle Silhouettes, Lymphography and Soyabean (Large). For the Soybean set, there are both a training set and a test set available in the repository. For the Vehicle Silhouettes and the Lymphography data all objects are available in one table. These sets were randomly divided into a training set 70% of objects and a test set 30% of objects. All data sets are multidimensional (from 18 to 35 conditional attributes) and have several decision classes (from 4 to 19 decision classes). Both of these properties are important in the context of the analysed approaches as data sets. The characteristics of the data sets are given in Table 1.

Each of the training data sets are made dispersed for the experiments by creating a set of local decision tables in such a way to reflect the real dispersed data sets. Sets of objects and sets of attributes of these sets should be different but not necessarily disjoint. The following dispersion in terms of the number of local decision tables: 3, 5, 7, 9 and 11 are settled on. Sets of conditional attributes in local tables were adopted so that they were different (different elements) but had some common attributes. The numbers of conditional attributes in individual local tables were varied. In the case of dispersion that contains a smaller number of local tables, there were more attributes in the tables (from 6 for

Table 1. Data set characteristics

Data set	# The training set	# The test set	# Conditional attributes	# Decision classes
Vehicle Silhouettes	592	254	18	4
Soybean	307	376	35	19
Lymphography	104	44	18	4

the Vehicle Silhouettes and the Lymphography data sets up to several or dozens of attributes for the Soybean data set). In the case of dispersion into a larger number of local tables, there were fewer attributes in the tables (from 3 to 6 attributes). Generally the lesser number of local tables, the greater the number of attributes in the tables. All local tables contain the full set of objects but no identifiers were stored in the tables as such the identification of objects between the tables is impossible. The large number of decision classes in the analyzed data sets is important because in the experiments we allow that ties occur. For instance, when the system generates a set of two decision classes out of nineteen possible classes, the result can still be very useful. In the Soybean data set there is a need for data cleaning to deal with some missing values occurring in almost all conditional attributes (except one). The number of cases with missing values ranges from 1 to 41 depending on the attribute. First objects with more than 50% of conditional attributes with missing values are removed. Next, the dominant in relation to decision classes method is applied - the dominant values are determined separately for each decision class and each attribute. Thus, for each attribute the following procedure is performed: the dominant value is determined for objects from one decision class. Then all objects with missing values belonging to this class are filled with this determined value.

4.1. Results of experiments

In all the experiments, decision trees were built using the Python language. The code was created that randomly generates a predetermined number of bootstrap samples based on each local table. Then trees for bootstrap samples were generated using the function sklearn.tree.DecisionTreeClassifier. When we classify a test object based on the set of decision trees generated for one local table, a vector over the decision classes is generated. The final decisions are made by using majority voting over the vectors. The emphasis of the experiments is to vary parameter, min samples split, in the Python sklearn.tree.DecisionTreeClassifier function which represents the least number of objects for which new split is defined in the decision tree and observe how it affects classification quality.

Both dispersed data set and non dispersed versions of the data are used in the experiment. When non-dispersed data was used, bootstrap samples were generated randomly based on the full table. Then for each bootstrap sample a tree was built. The test object is classified to the decision class indicated by the highest number of decision trees.

In the bagging method different numbers of base classifiers were analysed: 50, 40, 30, 20 and 10. The quality of classification is measured by the fraction of the number of misclassified objects by the total number of objects in the test set.

Different stop criterion values were analyzed: 2, 4, 6, 8, 10, $0.1\#U$, where $\#U$ is the number of objects in the training set. For the Lymphography data set the value $0.1\#U$ was not tested as it is equal to 10 ($\#U = 104$). The results for the dispersed data are presented in Table 2, and the values of the classification error obtained for the non-dispersed version are shown in Table 3. Each result presented is the averages of evaluations performed five times due to the indeterminism of generating bootstrap samples. Columns of experiments results are labeled 10, 20, 30, 40, 50 according to the number of bags of bootstrap samples that were used. In Tables 2 and 3, for each column and for each minimum sample split, the best classification (minimum classification error e) is colored in blue. In the case where there is a tie among two minimum sample split used, we consider the best results to be in favor of the case which has the average number of generated decisions sets close to 1. It can be seen that there is no clear and obvious relation between the number of bootstrap samples and the optimal number of objects in the stop criterion. The same can be said about the correlation between the number of local decision tables (dispersion thickness) and the optimal number of objects in the stop criterion. In other words, different numbers of the stop criterion were recorded as optimal for different dispersed or non-dispersed data sets. Thus, a hypothesis arises that with different values of the stop criterion we obtain comparable values of the classification error.

Table 2. Results of classification error e for respective minimum samples split when bagging method, decision trees and the two-step process of aggregation results are applied for dispersed data set

No. of local tables	Min smp split	Data sets: Vehicle					Lymphography					Soyabean				
		No. of bootstrap samples in bagging method					No. of bootstrap samples in bagging method					No. of bootstrap samples in bagging method				
		10	20	30	40	50	10	20	30	40	50	10	20	30	40	50
3	2	0.226	0.220	0.225	0.219	0.227	0.214	0.205	0.205	0.209	0.205	0.110	0.110	0.116	0.117	0.114
	4	0.227	0.235	0.231	0.229	0.239	0.218	0.223	0.237	0.232	0.227	0.109	0.109	0.112	0.116	0.115
	6	0.229	0.233	0.242	0.232	0.236	0.223	0.209	0.218	0.236	0.218	0.115	0.108	0.116	0.115	0.114
	8	0.237	0.234	0.231	0.235	0.235	0.209	0.214	0.209	0.236	0.218	0.104	0.107	0.118	0.110	0.111
	10	0.226	0.240	0.246	0.231	0.230	0.205	0.200	0.205	0.223	0.209	0.105	0.113	0.103	0.110	0.116
	0.1#U	0.263	0.265	0.261	0.258	0.255						0.156	0.160	0.153	0.160	0.161
5	2	0.214	0.203	0.216	0.204	0.205	0.232	0.250	0.245	0.245	0.241	0.183	0.178	0.175	0.177	0.177
	4	0.222	0.218	0.212	0.213	0.208	0.241	0.268	0.245	0.255	0.264	0.168	0.182	0.177	0.174	0.169
	6	0.227	0.229	0.213	0.223	0.221	0.241	0.250	0.250	0.255	0.259	0.190	0.169	0.186	0.175	0.172
	8	0.231	0.222	0.223	0.217	0.223	0.264	0.264	0.264	0.264	0.277	0.177	0.158	0.176	0.171	0.178
	10	0.236	0.217	0.234	0.223	0.219	0.259	0.245	0.264	0.264	0.264	0.181	0.182	0.169	0.170	0.171
	0.1#U	0.259	0.246	0.235	0.239	0.231						0.242	0.246	0.240	0.232	0.214
7	2	0.251	0.254	0.255	0.258	0.255	0.323	0.295	0.318	0.332	0.323	0.207	0.206	0.208	0.208	0.215
	4	0.266	0.264	0.258	0.252	0.250	0.304	0.346	0.323	0.318	0.327	0.213	0.215	0.216	0.216	0.215
	6	0.262	0.256	0.254	0.251	0.254	0.323	0.341	0.341	0.336	0.331	0.222	0.217	0.217	0.217	0.215
	8	0.250	0.250	0.258	0.256	0.243	0.354	0.332	0.350	0.318	0.336	0.223	0.226	0.225	0.218	0.224
	10	0.271	0.255	0.263	0.259	0.255	0.318	0.327	0.332	0.336	0.309	0.233	0.229	0.230	0.226	0.231
	0.1#U	0.287	0.279	0.276	0.286	0.279						0.284	0.279	0.284	0.287	0.278
9	2	0.277	0.286	0.288	0.270	0.277	0.354	0.359	0.355	0.359	0.350	0.267	0.270	0.273	0.267	0.263
	4	0.283	0.282	0.283	0.283	0.281	0.327	0.341	0.350	0.355	0.341	0.250	0.261	0.263	0.271	0.255
	6	0.289	0.271	0.288	0.284	0.276	0.355	0.368	0.346	0.355	0.345	0.258	0.257	0.254	0.266	0.269
	8	0.276	0.283	0.284	0.288	0.279	0.355	0.350	0.350	0.346	0.341	0.270	0.271	0.267	0.263	0.268
	10	0.285	0.288	0.295	0.289	0.292	0.350	0.350	0.341	0.346	0.350	0.272	0.269	0.273	0.284	0.268
	0.1#U	0.295	0.304	0.309	0.303	0.302						0.336	0.322	0.329	0.325	0.324
11	2	0.280	0.288	0.291	0.286	0.294	0.373	0.386	0.368	0.364	0.386	0.349	0.350	0.345	0.353	0.347
	4	0.288	0.293	0.291	0.289	0.289	0.409	0.364	0.377	0.377	0.359	0.340	0.341	0.347	0.340	0.340
	6	0.278	0.286	0.293	0.290	0.285	0.350	0.355	0.359	0.382	0.373	0.348	0.338	0.345	0.340	0.343
	8	0.285	0.285	0.283	0.290	0.286	0.359	0.368	0.366	0.368	0.364	0.344	0.345	0.342	0.347	0.351
	10	0.278	0.283	0.286	0.297	0.283	0.355	0.373	0.373	0.364	0.382	0.349	0.346	0.352	0.350	0.351
	0.1#U	0.309	0.310	0.309	0.313	0.309						0.393	0.403	0.395	0.392	0.391

Table 3. Results of classification error e for respective minimum samples split when bagging method and decision trees are applied to non-dispersed version of data set

Min smp split	Data sets														
	Vehicle					Lymphography					Soyabean				
	No. of bootstrap samples in bagging					No. of bootstrap samples in bagging					No. of bootstrap samples in bagging				
	10	20	30	40	50	10	20	30	40	50	10	20	30	40	50
2	0.224	0.235	0.226	0.232	0.224	0.136	0.100	0.136	0.132	0.109	0.078	0.082	0.083	0.084	0.081
4	0.219	0.231	0.238	0.237	0.231	0.141	0.136	0.123	0.150	0.118	0.081	0.084	0.073	0.079	0.082
6	0.223	0.224	0.221	0.242	0.234	0.164	0.141	0.159	0.132	0.150	0.074	0.074	0.080	0.078	0.083
8	0.231	0.227	0.237	0.223	0.238	0.150	0.159	0.141	0.127	0.150	0.086	0.078	0.080	0.079	0.077
10	0.238	0.238	0.240	0.231	0.242	0.146	0.155	0.177	0.155	0.182	0.087	0.079	0.082	0.077	0.081
0.1#U	0.240	0.239	0.251	0.245	0.252						0.153	0.148	0.136	0.140	0.132

Table 4. Number of nodes (**n**) and decision tree height (**h**) for respective minimum samples split when bagging method and decision trees are applied to dispersed and non-dispersed version of Vehicle data set

Min smp split	No. of local tables for dispersed data set										Non-dispersed data set	
	3		5		7		9		11			
	n	**h**	**n**	**h**	**n**	**h**	**n**	**h**	**n**	**h**	**n**	**h**
2	211.453	14.747	263.392	16.016	308.503	16.717	340.547	16.596	352.32	16.535	148.880	13.100
4	188.733	14.120	234.424	15.372	265.017	15.991	303.142	16.144	310.88	16.131	127.120	12.940
6	169.213	13.747	203.072	15.020	224.743	15.423	257.093	15.733	263.545	15.622	125.640	12.780
8	153.827	13.273	179.264	14.320	195.554	14.991	217.413	15.142	222.109	15.078	117.400	12.180
10	140.107	13.113	159.968	13.996	170.400	14.574	184.956	14.607	189.418	14.556	110.040	11.540
0.1#U	39.733	8.227	39.904	8.348	40.309	8.603	41.427	8.749	40.575	8.804	42.240	7.940

In Table 4, 5 and 6, results of average number of nodes (denoted as **n**) and height (denoted as **h**) of decision trees from the experiment with 50 bags in the bagging method are presented for both dispersed and non-dispersed versions of the data set. For other versions of the bagging method, the results are similar, we do not include them due to the space limitation. A negative correlation between the stop criterion value and the complexity of the decision trees (expressed by the number of nodes and the height) is observed. Also decision trees built based on non-dispersed data set has higher complexity compared to those built based on dispersed data set (except for the Vehicle data set) as the complexity of the model decreases with increasing dispersion of the data sets. This statement is not satisfied for the Vehicle data set due to the multi-valued numeric variables in this data set.

4.2. Discussion of results

The main conclusion drawn from the results is that the stopping criterion does not have a significant impact on the classification error, unless a very small or very large values are used. Statistical analysis was performed to confirm these observations. The results were divided into six groups: Group 1 – results for the stop criterion value equal to 2; Group 2 – results for the stop criterion value equal to 4; Group 3 – results for the stop criterion value equal to 6; Group 4 – results for the stop criterion value equal to 8; Group 5 – results for the stop criterion value equal to 10 and Group 6 – results for the stop criterion value equal to 0.1#U. A set of 75 observations with six dependent variables (one for each group) was obtained (results from Table 2). The Friedman's test was performed at first. When all variables were selected (results for stop criterions: 2, 4, 6, 8, 10 and 0.1#U) then the test confirmed that differences among the classification error in these six groups are significant, with a level of p = 0.00001. But if the last group was omitted (results for stop criterion 0.1#U) then the p-value was equal to 0.076. In case where two groups were omitted (the first and the last which are stop criterions 2 and 0.1#U) then the p-value was equal to 0.093. These results

Table 5. Number of nodes (**n**) and decision tree height (**h**) for respective minimum samples split when bagging method and decision trees are applied to dispersed and non-dispersed version of Soyabean data set

Min smp split	No. of local tables for dispersed data set										Non-dispersed data set	
	3		5		7		9		11			
	n	**h**	**n**	**h**	**n**	**h**	**n**	**h**	**n**	**h**	**n**	**h**
2	81.467	10.340	54.456	7.720	61.371	6.834	56.076	6.424	46.295	5.835	82.240	12.420
4	75.200	10.200	49.816	7.532	56.869	6.789	53.658	6.404	45.233	5.789	73.920	12.000
6	68.120	9.800	45.784	7.468	50.589	6.691	49.396	6.322	42.844	5.807	70.160	11.360
8	61.067	9.373	41.840	7.092	45.566	6.571	45.418	6.269	40.087	5.749	64.480	10.940
10	56.267	9.280	38.552	6.968	41.074	6.469	41.213	6.231	37.244	5.707	60.360	10.440
0.1#U	32.547	8.28	22.872	5.816	22.097	5.52	22.418	5.442	21.171	5.138	37.520	8.100

Table 6. Number of nodes (**n**) and decision tree height (**h**) for respective minimum samples split when bagging method and decision trees are applied to dispersed and non-dispersed version of Lymphography data set

Min smp split	No. of local tables for dispersed data set										Non-dispersed data set	
	3		5		7		9		11			
	n	**h**	**n**	**h**	**n**	**h**	**n**	**h**	**n**	**h**	**n**	**h**
2	42.080	8.027	34.624	6.456	25.703	5.383	18.533	4.344	13.782	3.449	31.600	6.720
4	36.560	7.460	33.152	6.472	25.731	5.391	18.293	4.447	13.542	3.445	27.240	6.340
6	32.907	7.073	30.304	6.360	23.617	5.371	18.018	4.398	12.967	3.407	25.200	5.960
8	29.160	6.880	28.072	6.272	22.126	5.189	16.978	4.362	12.458	3.413	22.680	5.460
10	25.507	6.293	25.824	6.144	20.834	5.123	16.502	4.307	11.825	3.315	20.840	5.440

Table 7. p-values for the Wilcoxon each pair test

	Group 1 – stop 2	Group 2 – stop 4	Group 2 – stop 6	Group 2 – stop 8	Group 2 – stop 10	Group 2 – stop 0.1#U
Group 1 – stop 2	–	–	0.022	0.045	0.003	0.000001
Group 6 – stop 0.1#U	0.000001	0.000001	0.000001	0.000001	0.000001	–

should already be recognized that the differences are not statistically significant. Then, in order to determine the pairs of groups between which statistically significant differences occur, the Wilcoxon each pair test for dependent groups were performed. The test showed that there is significant difference only between:

- Group 1 (stop criterion 2) and almost all other groups (except for Group 2 – stop criterion 4),
- Group 6 (stop criterion 0.1#U) and all other groups,
- Group 2 and Group 5 (stop criterions 4 and 10) with p=0.022.

The significance level at which the importance of differences were confirmed are given in Table 7. This means that the hypothesis that the mean errors between all other pairs of groups (not mentioned above) are the same cannot be rejected. Thus, in general, changes in the stop criterion among values 4, 6, 8 and 10 does not significantly impact the quality of classification.

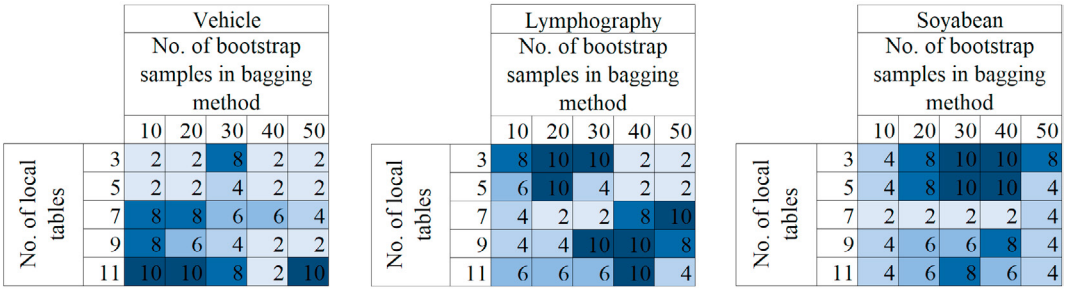

Fig. 1. Graphical representation of the optimal number of objects as a stop criterion for the dispersed data

In order to perform a deeper analysis of the relations between various parameters of data sets and the optimal stop criterion Figure 1 was created. The values of the stop criterion for different dispersed data sets are marked in color, the darker the color on the graph the higher the stop criterion value.

It can be observed that there is no dependency between various parameters of data sets and the optimal stop criterion that describe all data sets, but some correlations may be noted for the data sets separately. For the Vehicle data set, less dispersion results in lower optimal stop criterion values (the first two rows are brighter for the Vehicle data set). Also, greater number of bootstrap samples causes lower optimal stop criterion values (the last two columns are brighter for the Vehicle data set). For the Lymphography data set when both conditions are met (less dispersion and greater number of bootstrap samples) then we have lower values of the optimal stop criterion (upper right corner is brighter for the Lymphography data set). For the Soyabean data set generally higher values of the optimal stop criterion have been noted as optimal. Only for the dispersed set with 7 local decision tables lower values were observed. It can also be concluded that for almost all cases when the minimum sample split is highly increased (that is to $0.1\#U$), the classification quality is reduced.

In non-dispersed version of the Lymphography data sets, best classification results are observed mostly for minimum samples split of 2. For the Vehicle data set increasing the bootstrap samples increased the optimal value of the stop criterion, but for fifty bootstrap samples the optimal stop criterion is 2. For the Soyabean data set larger values of the stop criterion appear to be optimal.

The general conclusions that can be drawn based on the obtained results are as follows. There are data sets such as the Soyabean, for which greater values of the stop criterion always provide better results regardless of other parameters, such as the thickness of dispersion or the number of bootstrap samples. Of course, greater values of the stop criterion results in building decision trees that have better generalization properties and an improvement in the classification quality can be obtained in this case. However, for some dispersed data sets, in the case of smaller number of local tables and/or larger number of bootstrap samples, better quality of classification is obtained for a small number of objects in the stop criterion.

The main focus of the article was to evaluate how the quality of classification depends on the stop condition when we use the bagging method for dispersed data. And in general, it was shown that changing the value of the stop criterion in decision trees does not affect the quality of classification for dispersed data much. However, based on the presented results, we can also compare the classification quality in terms of different versions of dispersion. It can be seen that when we use decision trees and the bagging method, the greater the dispersion of data, the worse the classification quality. The best results were obtained for data gathered in one table. The only exception is the Vehicle data set with five local decision tables. However, we must bear in mind that we can not treat dispersed data in the same way as data collected in one decision table. Classification based on dispersed data is much more difficult due to the inability to combine data into a single table, due to the need to protect data and treat data sets as isolated islands. Such issues are very often considered in federated learning or fog learning. We should also noticed that, in these experiments, the increased number of tables does not increase the data/information in the tables. The increased number of tables only impact on the increased dispersion, while no additional data/information is available in the increased number of local decision tables.

It is planned to consider the CART algorithm with Twoing criterion that allows for balancing the tree in future work. This approach may be especially interesting when determining the stopping of the tree structure. It is also

planned to use other methods of generating an ensemble of classifiers. For example, in random forest bootstrapped samples are used, but in addition, at each node of the decision tree, the attribute space is randomly constrained. This can be important in determining the stopping level of decision tree construction.

5. Conclusions

In this paper the bagging method with decision trees are applied to independent dispersed data. Classification results are observed for experiments on three data sets that were dispersed in five versions with varying minimum number of objects at which building of decision trees is stopped call stop split criteria. The number objects for which stopping criteria was observed on are 2, 4, 6, 8, 10 and finally $0.1 \times number\ of\ objects\ in\ training\ data$. Synonymously observations are made for the non-dispersed version of the data sets. It can be concluded that in dispersed data set, the stop split criteria does not influence the classification quality much. It was statistically proven that the stop criterion 4, 6, 8 and 10 does not significantly impact the quality of classification. There are data sets (with numerous decision classes) for which greater values of the stop criterion always provide better results regardless of other parameters, such as the number of local tables or the number of bootstrap samples. For some dispersed data sets, in the case of smaller number of local tables and/or larger number of bootstrap samples, better quality of classification is obtained for a small number of objects in the stop criterion. Also, it is generally understandable that in both data sets, when stopping criteria is very high ($0.1 \times number\ of\ objects\ in\ training\ data$), quality of classification is reduced. This is because decision trees are not built to desirable length thus poor classification quality. On the other hand, the application of the pre-pruning approach in the form of a stop criterion defined by the number of objects in a node during tree construction significantly reduces the complexity of the tree, which is extremely important for dispersed data.

References

[1] Atlam, H. F., Walters, R. J., Wills, G. B.: Fog computing and the internet of things: A review. Big data and cognitive computing, 2(2), 10, (2018)

[2] Begon, J. M., Joly, A., Geurts, P.: Globally induced forest: A prepruning compression scheme. In International Conference on Machine Learning, 420–428, PMLR, (2017)

[3] Breiman, L., Friedman, J., Stone, C. J., Olshen, R. A.: Classification and regression trees. CRC press, (1984)

[4] Esposito, F., Malerba, D., Semeraro, G.: A Comparative Analysis of Methods for Pruning Decision Trees, Pattern Analysis and Machine Intelligence (1997)

[5] Hosseinalipour, S., Brinton, C. G., Aggarwal, V., Dai, H., Chiang, M.: From Federated to Fog Learning: Distributed Machine Learning over Heterogeneous Wireless Networks, IEEE Communications Magazine, 58, 12, 41–47, (2020)

[6] Hssina, B., Merbouha, A., Ezzikouri, H., Erritali, M.: A comparative study of decision tree ID3 and C4. 5. International Journal of Advanced Computer Science and Applications, 4(2), 13-19, (2014)

[7] Kotsiantis, S. B.: Decision trees: a recent overview. Artif. Intell. Rev., 39(4), 261–283 (2013)

[8] Mark, L., Maimon, O., Minkov, E.: Improving Stability of Decision Trees, International Journal of Pattern Recognition and Artificial Intelligence, Vol. 16, No. 02, pp. 145-159 (2002)

[9] Moskov, M., Zielosko, B.: Combinatorial Machine Learning, Studies in Computational Intelligence,Volume 360

[10] Patel, N., Upadhyay, S.: Study of Various Decision Tree Pruning Methods with their Empirical Comparison in WEKA, International Journal of Computer Applications Vol. 60 (2012)

[11] Przybyła-Kasperek, M., Aning, S.: Bagging and single decision tree approaches to dispersed data, ICCS 2021, In press

[12] Przybyła-Kasperek, M.: Generalized objects in the system with dispersed knowledge. Expert Syst. Appl., 162, 113773 (2020)

[13] Przybyła-Kasperek, M.: Coalitions' Weights in a Dispersed System with Pawlak Conflict Model. Group Decis. Negot., 1–43 (2020)

[14] Przybyła-Kasperek, M., Wakulicz-Deja, A.: Dispersed decision-making system with fusion methods from the rank level and the measurement level – A comparative study. Inf. Syst., 69, 124–154 (2017)

[15] Quinlan, J. R.: C4.5: programs for machine learning. Elsevier, (2014)

[16] Rokach, L., Miamon, O.: Data mining with decision trees, Series in Machine Perception and Artificial Intelligence. World Scientific Vol. 81, 2nd Edition, (2014)

[17] Sim, D. Y. Y., Teh, C. S., Ismail, A. I.: Improved boosting algorithms by pre-pruning and associative rule mining on decision trees for predicting obstructive sleep apnea. Advanced Science Letters, 23(11), 11593–11598, (2017)