Boosting Effect for Classifier Based on Simple Granules of Knowledge

Boosting Effect for Classifier Based on Simple Granules of Knowledge

Piotr Artiemjew

University of Warmia and Mazury in Olsztyn, Faculty of Mathematics and Computer Science, Sloneczna 54, 10-710 Olsztyn, Poland, e-mail: artem@matman.uwm.edu.pl

Corresponding author: artem@matman.uwm.edu.pl

The CSG classifier (Classifier based on Simple Granules of Knowledge) is the rough set method based on the usage of rough mereology indiscernibility classes in the classification process. The classified objects are covered by r-indiscernible objects, which are involved in voting for decision. This simple method designed for symbolic data works surprisingly effective, what was proven in our previous works. The classifier among others turned out to be resistant for damages because it can absorb missing values. Seeing the effectiveness of boosting scheme with the rough set methodology, we were motivated to conduct a series of experiments to check the behaviour of our CSG method in the Ensemble model. In this article, we have checked Arcing, Bootstrap Committee (The Pure Bagging) and Ada-Boost with the Monte Carlo split. For experimentations, we have chosen the selected data from the UCI Repository. The results show the high level of stabilisation and further potentials of boosting effect for our classifier.

KEYWORDS: rough sets, rough inclusions, granules of knowledge, classification of symbolic data, CSG classifier, Bootstrap Ensemble, Ada-Boost, Arcing.

1. Introduction

In the recent works, we have developed a classifier based on simple granules of knowledge (CSG) [1, 2, 14]. This method is in the group of rough set classifiers based on the granular structures developed by Polkowski [15]. Granular classification became really effective for significant reduction of the training decision system size.

The ensemble scheme of classification (Boosting), in the case of rough set methods is really effective (see [7-10], [16]), therefore, we decided to examine the applicability of the selected methods on our classifiers.

The Boosting consists of sequential production of classifiers where each classifier is dependent on the previous one and focuses on the previous one's error

[11, 17, 22]. Incorrectly classified objects are chosen more frequent to training sets. The Ensemble of Bootstraps [23] forms a group of classifiers where each classifier is dependent on the previous one, however, there is no influence on the classification accuracy.

In this work, we have conducted three types of experiments, where the first one consists of the stabilisation of classification based on the Committee of Bootstraps [19]. The second is based on Arcing [5, 18], and the third is Ada-Boost with the Monte Carlo split [11, 17, 22].

The rest of the article is organised as follows. In Sections 1.1-1.3, we provide the theoretical introduction to the CSG classifier. In Section 2, we give a brief description of chosen Boosting methods. In Section 3, we report the results of experiments with their interpretation. Finally, in Section 4, we present the conclusion and outline of future works. The following sub-sections also explain the theoretical fundamentals.

1.1. Theory in the Nutshell

An information system [12] is a pair $I = (U, A)$, where U is the universe of objects (samples), and A is a set of conditional attributes. Decision system is a triple $DS = (U, A, d)$, where $d \notin A$. The basic form of granulation in decision and information systems consists in partitioning U into classes of the indiscernibility relation $IND(A)$ defined as $IND(A) = \{(u, v) : a(u) = a(v), \forall a \in A\}$. Each class $[u]_A = \{v \in U: a(u) = a(v), \forall\ a \in A\}$ is interpreted as an elementary granule and unions of elementary granules are granules of knowledge. Another approach to granulation proposed in [15] consists in using rough inclusions, cf. [15].

A rough inclusion [15] is a relation $\mu \subseteq U \times U \times [0,1]$, which can be regarded as graded similarity relation extending the indiscernibility relation by relaxing restrictions on attribute values. We let $IND(u, v) = \{a \in A : a(u) = a(v)\}$.

In our approach, we use rough inclusion proposed in [15] to classify test objects. Test object u is classified by granules, which are formed from the training set as follows:

$$g_{r_{gran}}(u) = \{v \in U : \frac{|IND(u,v)|}{|A|} \geq r_{gran}\},$$

where $r_{gran} \in [0,1]$ is the granulation radius.

The samples inside the granule transfer decision to our test object. If tie occurs, it is resolved by a random choice. This type of classification is one of the simplest among those studied by the author cf. [3, 4, 13]. The results of research for optimal parameters for this method are available in [2]. In that work, this approach is continued and the main purpose is to find a threshold of the optimal parameter stability for the random damage to the decision system. Additionally, the method for experimental detecting of the optimal radius value for a given data set, by means of multiple CV-5 and subsequent confirmation by means of Leave One Out, was proposed [2].

As soon as the optimal value is found for the test data, it can be used for classification of incoming objects without any need for full granulation procedure.

We will now describe the background information regarding how the classifiers are considered in terms of rough set theory.

1.2. Classification by Simple Granules of Knowledge - Theoretical Background

The rough inclusion is defined as $\mu_\pi(x, y, r)$, where x, y are individual objects and $r \in [0,1]$. This inclusion satisfies the following requirements in relation to a given part relation π on a set U of individual objects, see [15]:

1. $\mu_\pi(x, y, 1) \Leftrightarrow ing_\pi(x, y)$;
2. $\mu_\pi(x, y, 1) \Rightarrow [\mu_\pi(z, x, r) \Rightarrow \mu_\pi(z, y, r)]$;
3. $\mu_\pi(x, y, r) \wedge s < r \Rightarrow \mu_\pi(x, y, s)$.

Those requirements seem to be intuitively clear. 1. demands the predicate μ_π to be an extension to the relation ing_π of the underlying system of Mereology; 2. does express monotonicity of μ_π, and 3. assures the reading "to degree at least r". We use here only one rough inclusion, albeit a fundamental one, viz., see [15] for its derivation:

$$\mu_L(u, v, r) \Leftrightarrow \frac{|IND(u,v)|}{|A|} \geq r.$$

A granule $g_\mu(u, r)$ about u in U of the radius r, relative to μ, is defined by letting

$$g_\mu(u, r) \text{ is } ClsF(u, r),$$

where the property $F(u,r)$ is satisfied with an object v if and only if $\mu(v,u,r)$ holds, and Cls is the class operator, see, e.g., [15]. Practically, in case of μ_L, the granule $g(u,r)$ collects all v from U such that $|IND(v,u)| \geq r \times A$.

1.3. The Procedure of Classification via CSG Algorithm

1 For a given training decision system (U_{trn}, A, d) and the test system (U_{tst}, A, d), The granular radius

$$r_{gran} \in \{\frac{0}{card\{A\}}, \frac{1}{card\{A\}}, ..., \frac{card\{A\}}{card\{A\}}\},$$

where $card$ means the number of attributes in the set A.

2 For a classified test object $u \in U_{tst}$, the classification granule $g_{r_{gran}}(u)$ is found in the training set U_{trn} as follows:

$$g_{r_{gran}}(u) = \{v \in U_{trn} : \frac{|IND(u,v)|}{|A|} \geq r_{gran}\},$$

where $r_{gran} \in [0,1]$.

4 The most numerous decision class of granule $g_{r_{gran}}(u)$ transfers decision to our test object. If tie occurs, it is resolved randomly.

2. Boosting Methods

In this section, we present a short description of methods used in our work to boost CSG classifier. First of all, the description of Pure Bagging is presented.

2.1. Bootstrap Ensembles (Pure Bagging)

It is the random committee of bootstraps [23], a method in which the original decision system (the basic knowledge) is split into (TRN) training data set and ($TSTvalid$) validation test data set. For the TRN system, and fixed number of iterations, we form new training systems ($NewTRN$) by random choice with returning of $card\{TRN\}$ objects. In all iterations, we classify the $TRNvalid$ system in two ways: the first of them is based on the actual $NewTRN$ system and the second one is based on the committee of all performed classifications. In the committee, majority voting is performed and the ties are resolved randomly.

2.2 Boosting Based on Arcing

The method is similar to the previous one, but this time we use Bootstraps in the Ensemble model. The TRN is split into two data sets $NewTRN$ and $NewTST$ (see [5] and [18]). The split is based on Bootstraps and the $NewTRN$ system is formed by importance weights assigned to training objects. The objects are chosen for the $NewTRN$ with a fixed probability determined by such weights. The weights are initially equal, but after the first classification of the $NewTST$ system based on $NewTRN$, the weights are modified in such a way that well-classified objects have the weights lowered. The weights before their application require normalization. The method presented above is called Arcing. In each iteration of learning, the newly formed $NewTRN$ system classifies the $TSTvalid$ system in this single iteration separately. In addition, it takes part in the classification of TST-valid as one of the weak classifiers in the committee of classifiers for all iterations performed until the given time. In Arcing, the factor for weights modification is equal to

1-Accuracy/Accuracy.

2.3. Ada-Boost with Monte Carlo Split

It is a similar method of classification to the one previously described, but here a different method for $NewTRN$ and $NewTST$ forming has been used (see [11, 17, 22]). The TRN data set is split according to the fixed ratio. The objects of $NewTRN$ are chosen according to significance weights. The good split ratio is about 0.6, because it is close to the approximate size of the distinguishable objects in the bootstraps. The rest of the algorithm works in a similar way to the previously described.

Results of experiments are presented in the next section.

3. The Results of the Experiments

In this section, we check the boosting effect on our CSG classifier. Experiments with the use of multiple Cross Validation ($CV-5$) [6] method and the exemplary data from UCI Repository [21] (the list of data is displayed in Table 1) have been conducted. We have developed tools for experimentations in C++. In the

Table 1

The list of examined data sets from UCI Repository

Name	Attr type	attr_no	obj_no	class_size
Australian-credit	categorical, integer, real	15	690	2
Heart disease	categorical, real	14	270	2
Nursery	categorical	9	12960	5

above experiments, a complete spectrum of granulation radii for classification was used. In this article, selected, representative results for three methods of boosting are shown. An exemplary information of the full spectrum of radii and the average size of classification granules for Australian credit data set is shown in Tables 2 to 4.

The results of this work are shown in Figures 1 to 13. The results of Arcing for Australian Credit Approval data set and the selected radii from 0.5 to 0.642857 are given in Figures 1 to 3. The average size of classification granules is from 155.6 to 1.9. As it can be seen, the classification accuracy for the committee is really stable starting from the 20th iteration and increases until the radius is too high and the granule is too small. For the smaller radii, the classification granules contain the noise and the classification is not precise. When lowering the size of granules, the

classification accuracy increases, and eventually, for the radius 0.571429 with the average size of granules equal to 8.4, the best precision of classification is obtained. For the higher radii, the granules are too small and the information is faded which results in the accuracy diminishing. The analogous regularity was observed for the rest of the investigated data sets. For example, the best results for the Heart Disease and Nursery data sets are shown in Figures 4 and 5, respectively. The accuracy of classification is stabilised in the Ensemble scheme significantly – the classifier is highly boosted. In Figures 6 to 9, the result of Pure Bagging (the committee of Bootstraps) is presented. In Figures from 6 to 8, the result for an Australian credit is shown, and in Figure 9, the result for Heart Disease is given. As it can be seen, the best results produced by our method are fully comparable with those obtained by the previous method, howev-

Table 2

The average size of classification granules in Bagging model (Arcing) for Australian Credit data set

Radius	Average size
0	172
0.0714286	172
0.142857	168.5
0.214286	155.6
0.285714	129.08
0.357143	109
0.428571	52
0.5	24.4
0.57	8.4
0.64	1.9

Table 3

The average size of classification granules in Pure Bagging model

Radius	Average size
0	345
0.0714286	344.5
0.142857	338.3
0.214286	312.6
0.285714	258.5
0.357143	182.2
0.428571	106.4
0.5	48.3
0.57	15.5
0.64	3.4
0.71	0.5

Table 4

The average size of classification granules in Ada-Boost model

Radius	Average size
0	172
0.0714286	171.8
0.142857	168.6
0.214286	155.6
0.285714	129.2
0.357143	91.7
0.428571	53.5
0.5	23.1
0.57	7.8
0.64	1.8
0.71	0.25

er, the stability of classification for the raising radii is rather poor. This method works satisfactory for the stabilisation of classifier, however, it is difficult to anticipate the regularity. Finally, from Figures 10 to 13, the exemplary result for Ada Boost algorithm with the Monte Carlo split is presented. In Figures 10 to 12, the result for the Australian Credit is given. In addition, in Figure 13, the result for Heart Disease is presented. The Boosting effect for Ada Boost is really stable. The regularity is similar to Arcing. The results show that the CSG classifier can be Boosted in a really effective way by using Ensemble scheme.

Seeing the results, Arcing seems to be the best method for the CSG classifier. Despite the fact that the Pure Bagging in the best cases is fully comparable with Arcing, the Arcing is more regular. Somebody can ask about the optimality of parameters for CSG classifier and the way of proper parameters selection. The optimal parameters for CSG classifier can be obtained by double granulation of data sets, without any classification process, which was shown in [14].

Figure 1

Exemplary result for Bagging based on Arcing, Australian Credit dataset, Mean size of classification granules is 24.4, the radius = 0.5
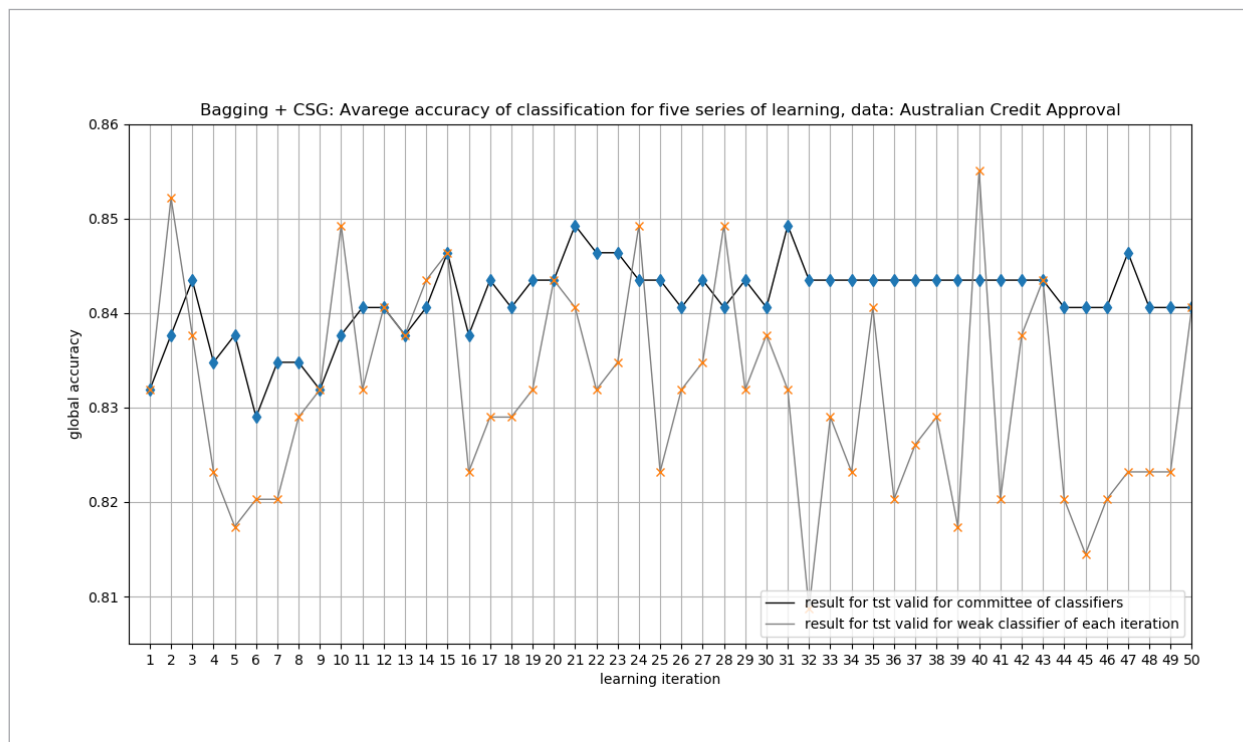
Figure 2

Exemplary result for Bagging based on Arcing, Australian Credit dataset, Mean size of classification granules is 8.4, the radius = 0.571429
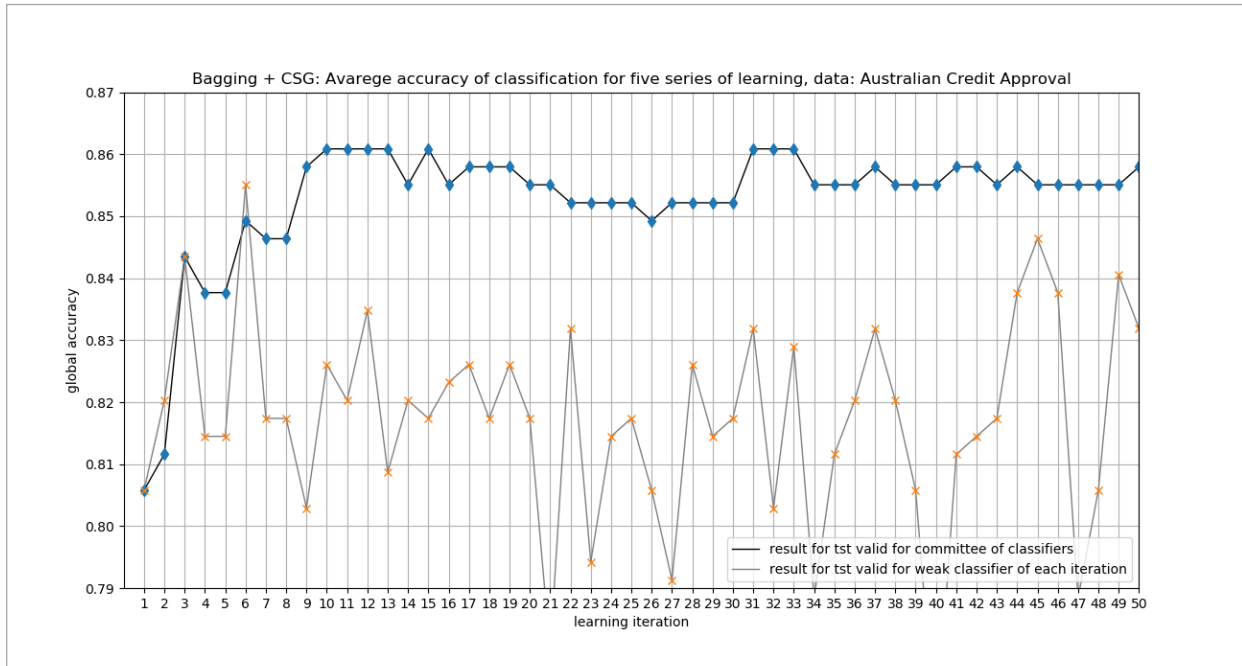

Figure 3

Exemplary result for Bagging based on Arcing, Australian Credit dataset, Mean size of classification granules is 1.9, the radius = 0.642857
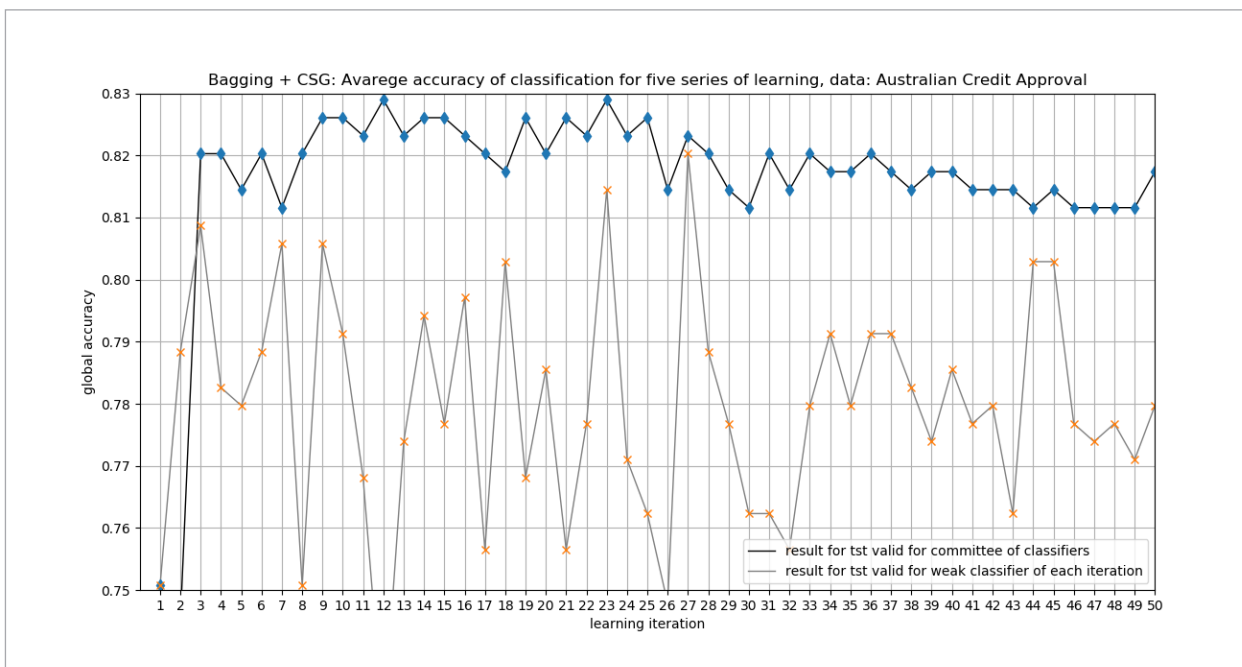
Figure 4

Exemplary result for Bagging based on Arcing, Heart Disease dataset, Mean size of classification granules is 44.1 , the radius = 0.307692
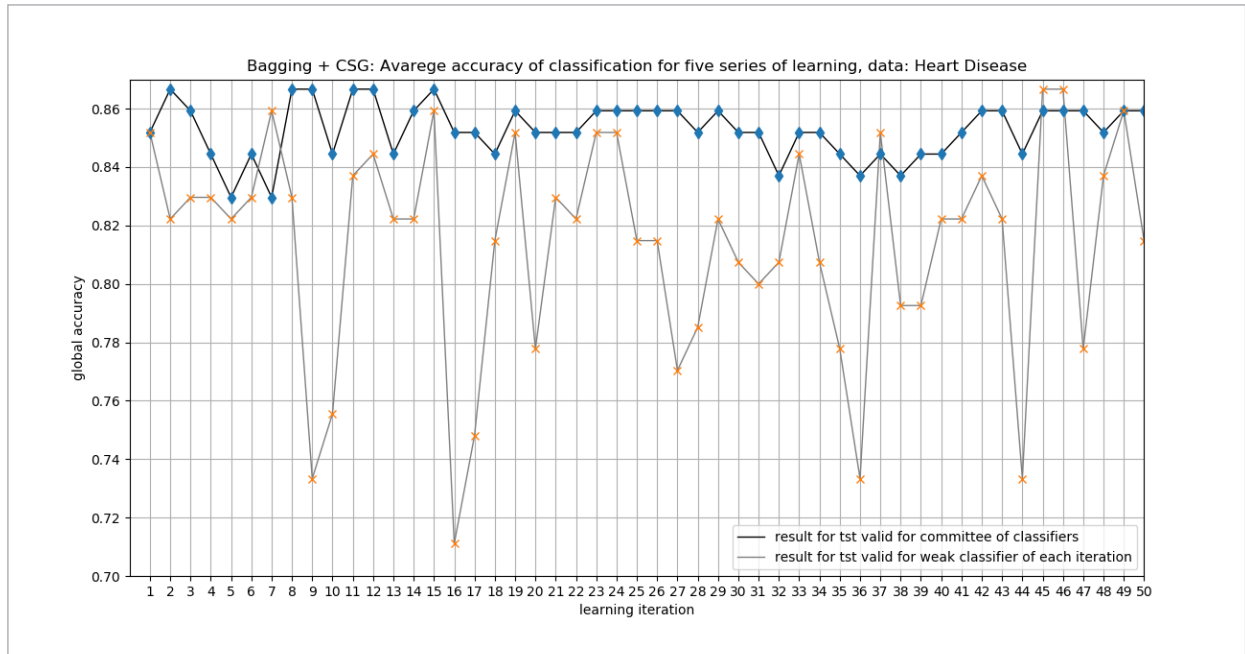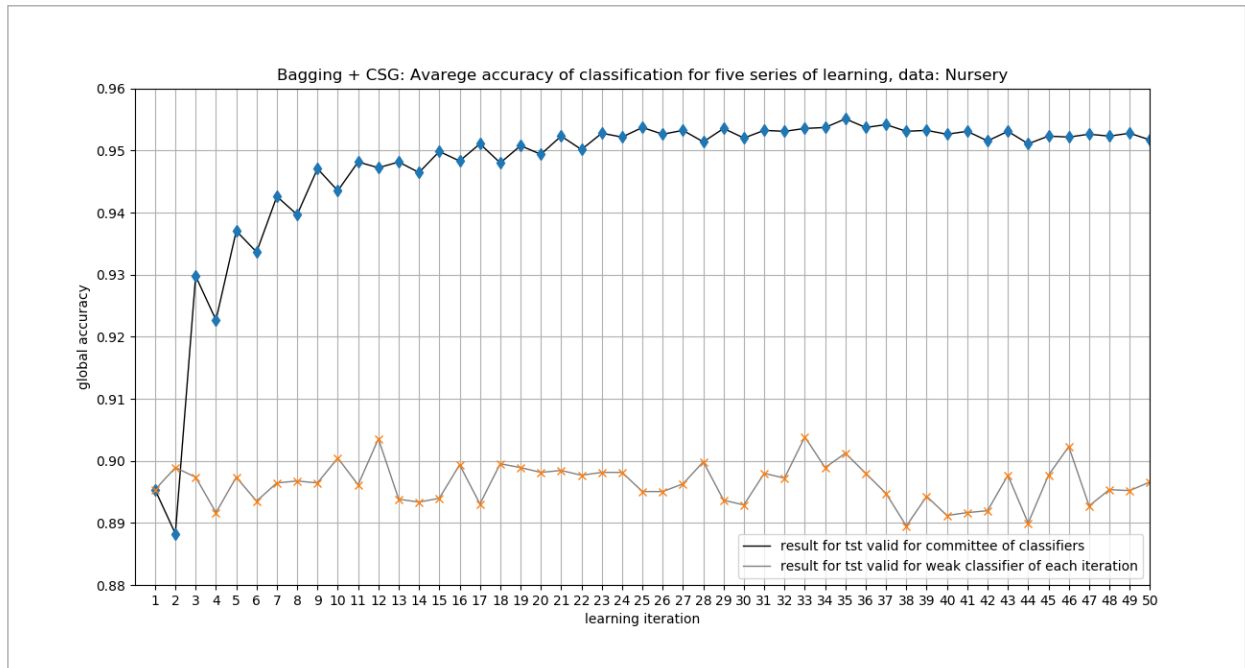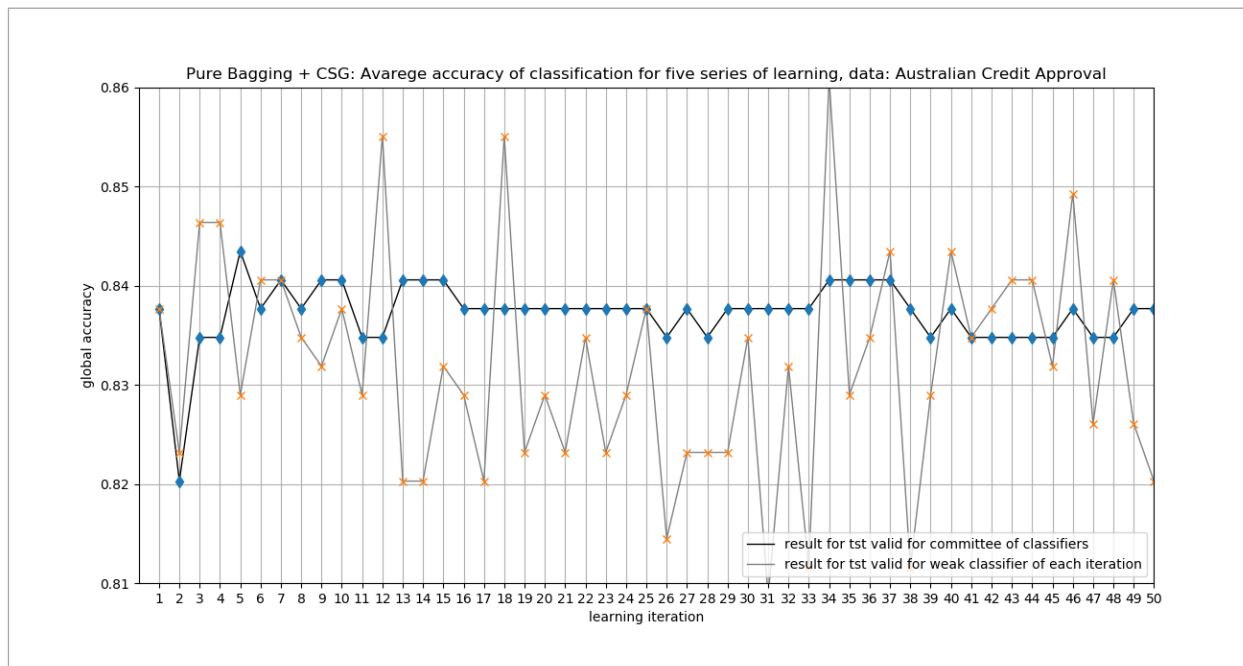

Figure 5

Exemplary result for Bagging based on Arcing, Nursery dataset, Mean size of classification granules is 4.7, the radius = 0.875

Figure 6

Exemplary result for Pure Bagging, Australian Credit dataset, Mean size of classification granules is 48.3, the radius = 0.5
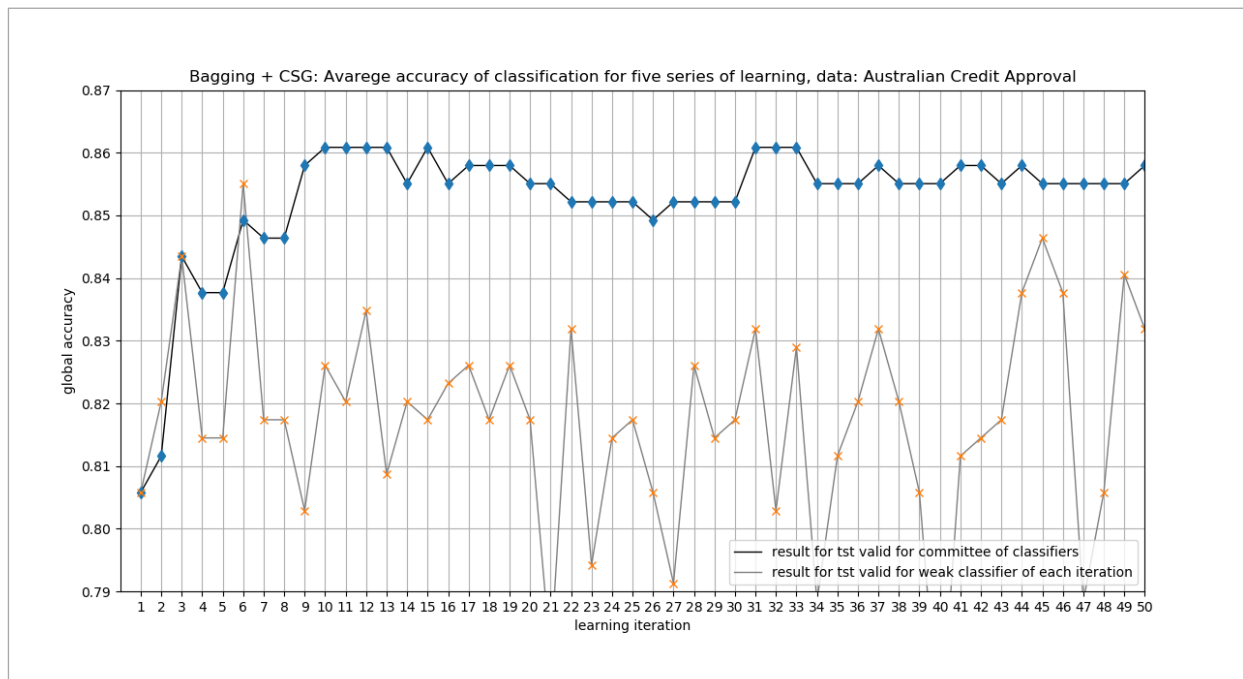

Figure 7

Exemplary result for Pure Bagging, Australian Credit dataset, Mean size of classification granules is 15.5, the radius = 0.571429

Figure 8

Exemplary result for Pure Bagging, Australian Credit dataset, Mean size of classification granules is 3.4, the radius = 0.642857
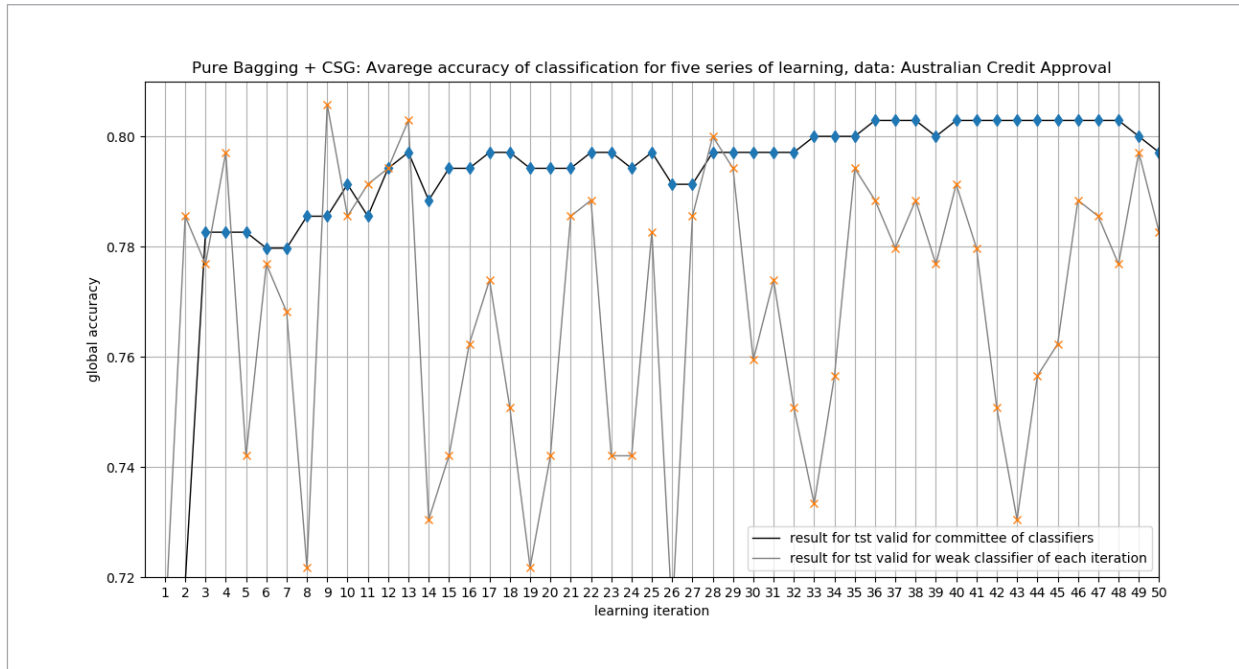

Figure 9

Exemplary result for Pure Bagging, Heart disease dataset, Mean size of classification granules is 106.5, the radius = 0.428571
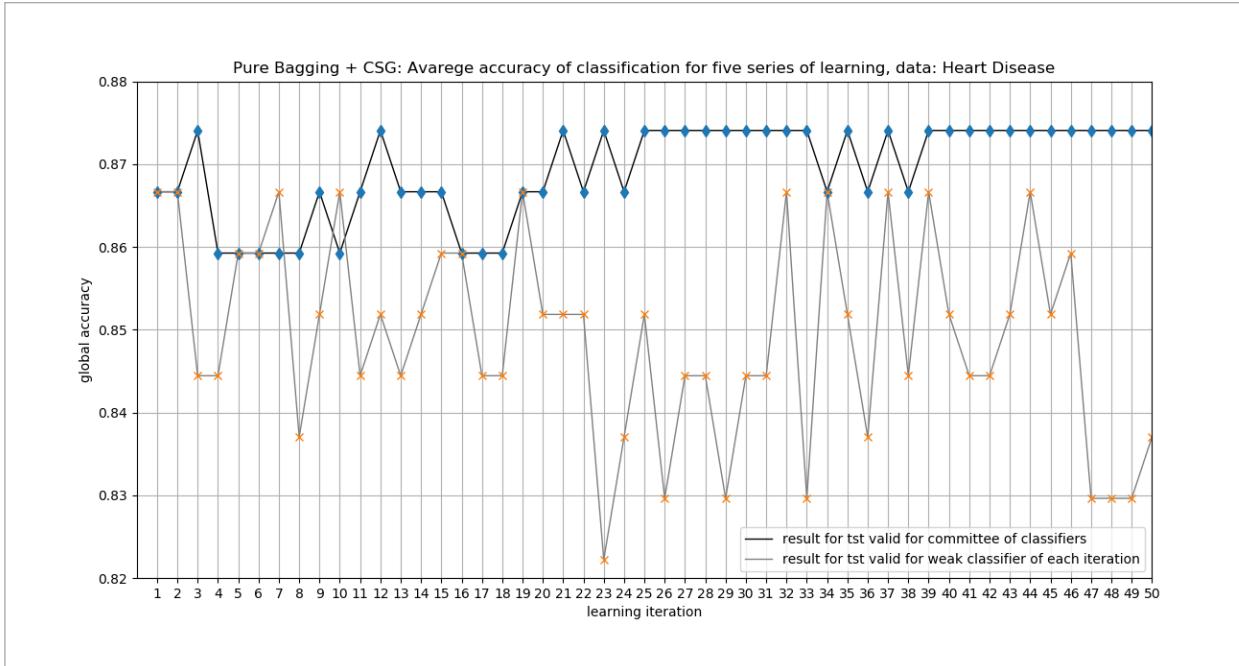
Figure 10

Exemplary result for Ada-Boost, Australian Credit dataset, Mean size of classification granules is 23.1, the radius = 0.5
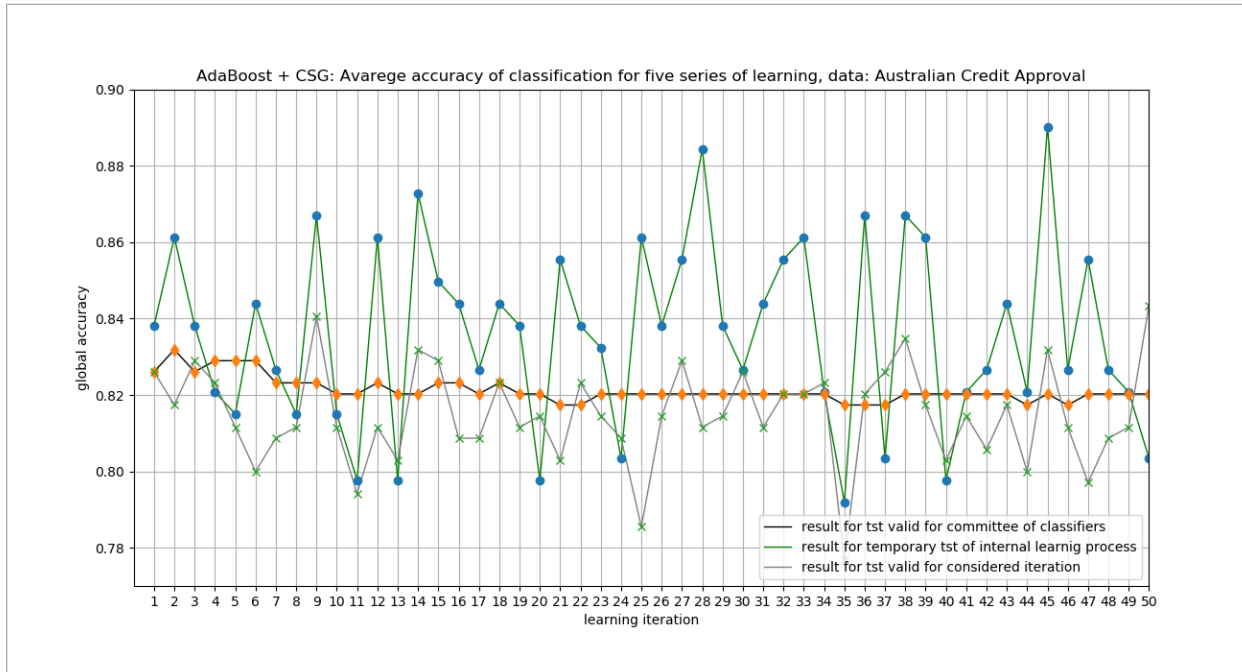

Figure 11

Exemplary result for Ada-Boost, Australian Credit dataset, Mean size of classification granules is 7.8, the radius = 0.571429
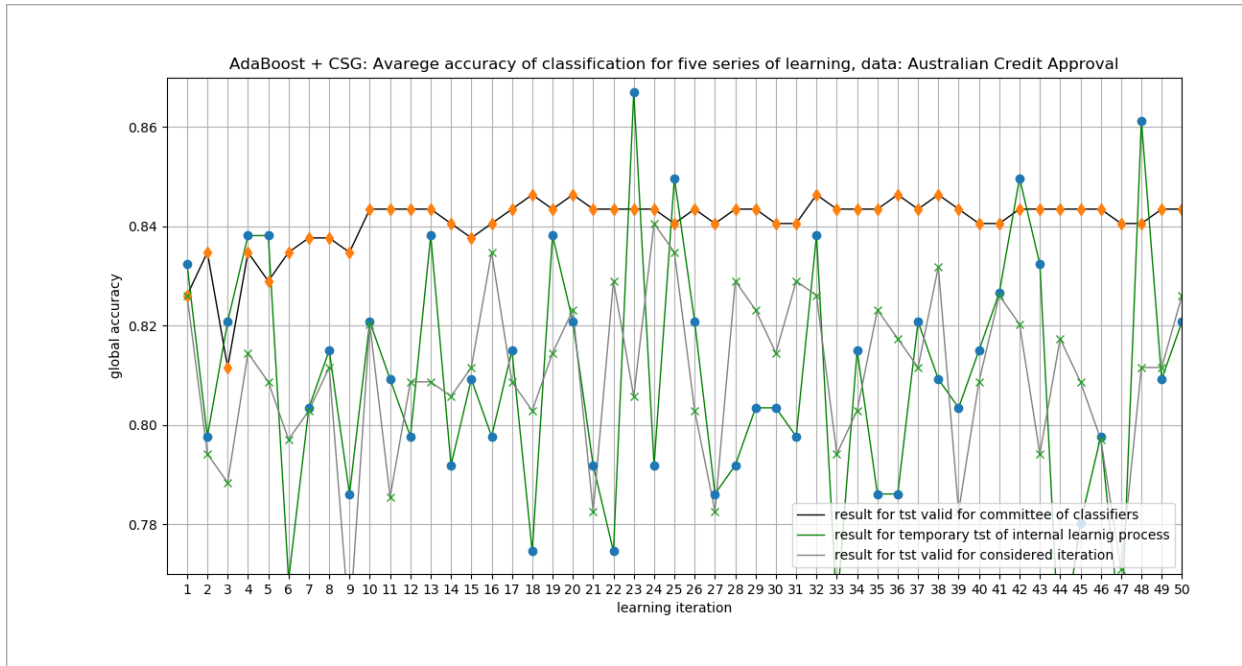
Figure 12

Exemplary result for Ada-Boost, Australian Credit dataset, Mean size of classification granules is 1.8, the radius = 0.642857
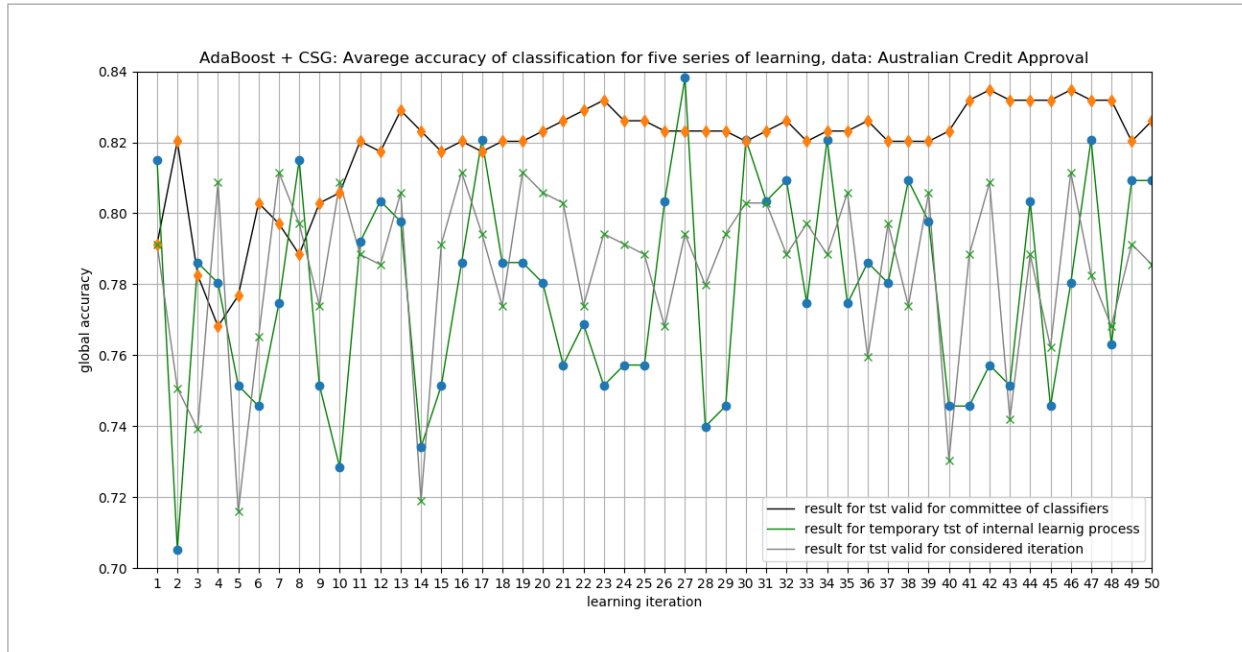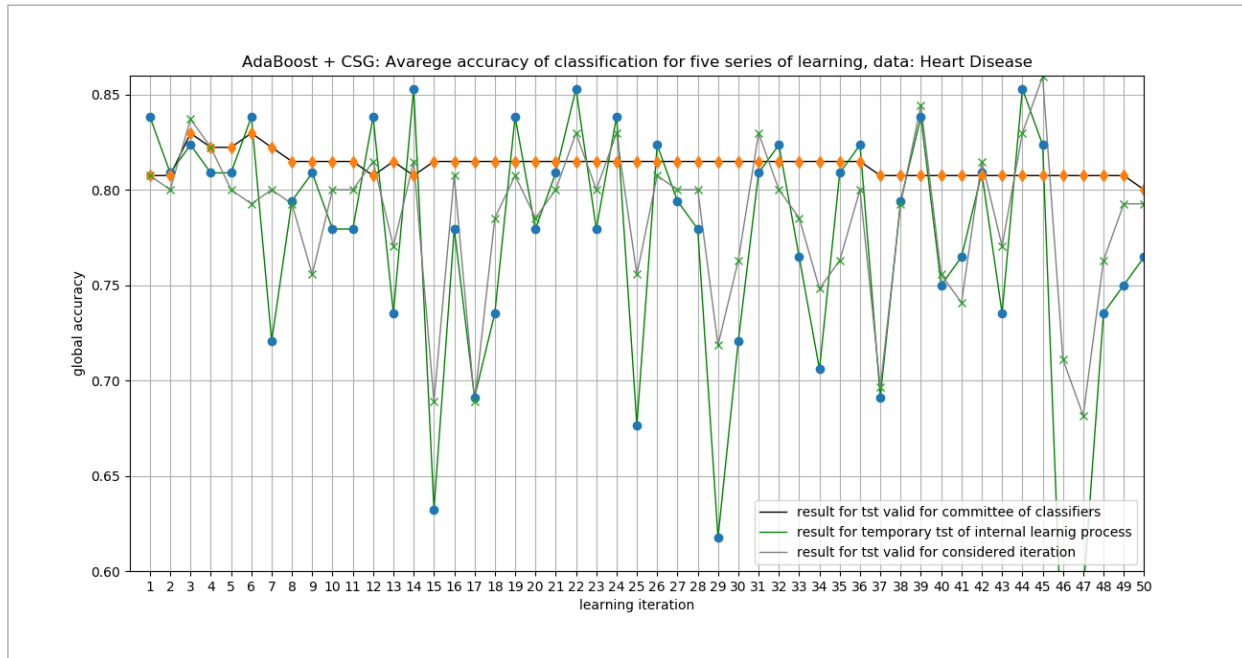

Figure 13

Exemplary result for Ada-Boost, Heart Disease dataset, Mean size of classification granules is 44.3, the radius = 0.307692

4. Conclusion

In this work, the boosting effect on the classifier based on simple granules of knowledge (CSG classifier) have been investigated. The Ensemble scheme turned out to be really effective and the classifier was improved in a significant way. Despite the fact that Pure Bagging works in terms of accuracy very well in many cases, the Arcing and Ada Boost are the most stable for the CSG classifier. The Arcing seems to be slightly better than Ada Boost. As we anticipated, in the classification (boosting) process, accuracy depends on the size of the classification granules. Commencing with the bigger granules, the classifier is disturbed by noise in the classification process. On the other hand, using too small granules, the classification process loses some information from the data. In both cases, the accuracy can be lowered. The best way is to use the optimal radii of classification with the proper size of granules [14]. In conclusion, the CSG can be effectively boosted what was shown in experiments presented in this work.

In future works, it is planned to perform more experiments to verify the boosting effect on the CSG classifier dedicated to the numerical data with the use of descriptors indiscernibility ratio.

References

1. Artiemjew, P. Stability of Optimal Parameters for Classifier Based on Simple Granules of Knowledge. Technical Sciences, UWM Publisher, Olsztyn, 2011, 14(1), 57-69.

2. Artiemjew, P. In Search of Optimal Parameters for Classifier Based on Simple Granules of Knowledge. III International Interdisciplinary Technical Conference of Young Scientists (InterTech 2010), Poznan University Press, 2010, 3, 138-142.

3. Artiemjew, P. On Strategies of Knowledge Granulation and Applications to Decision Systems, PhD Dissertation, Polish Japanese Institute of Information Technology, L. Polkowski, Supervisor, 2009, Warsaw.

4. Artiemjew, P. Natural Versus Granular Computing. Classifiers from Granular Structures. Proceedings of 6th International Conference on Rough Sets and Current Trends in Computing RSCTC'08. Akron, Ohio, USA, Springer Berlin/ Heidelberg, 2008, 5306, 150-159.

5. Breiman, L. Arcing Classifier (with Discussion and a Rejoinder by the Author). The Annals of Statistics, 801849, 26(3). Retrieved 18 January 2015, https://projecteuclid.org/euclid.aos/1024691079

6. Devroye, L., Gyorfi, L., Lugosi, G., A Probabilistic Theory of Pattern Recognition. Springer-Verlag, New York, 1996. https://doi.org/10.1007/978-1-4612-0711-5

7. Hu, X. Construction of an Ensemble of Classifiers Based on Rough Sets Theory and Database Operations. Proceedings of the IEEE International Conference on Data Mining (ICDM 2001), 2001.

8. Hu, X. Ensembles of Classifiers Based on Rough Sets Theory and Set-Oriented Database Operations. Presented at the 2006 IEEE International Conference on Granular Computing, Atlanta, GA, 2006.

9. Murthy, C., A., Saha, S., Pal, S., K. Rough Set Based Ensemble Classifier. Rough Sets, Fuzzy Sets, Data Mining and Granular Computing Lecture Notes in Computer Science Volume 6743, 2011, 27.

10. Nowicki, R., K., Nowak, B., Woźniak, M. Application of Rough Sets in k Nearest Neighbours Algorithm for Classification of Incomplete Samples. Advances in Intelligent Systems and Computing, Springer Publishing International, Switzerland, 2016, 416, 243-257.

11. Ohno-Machado, L. Cross-Validation and Bootstrap Ensembles, Bagging, Boosting. Harvard-MIT Division of Health Sciences and Technology, 2005, http://ocw.mit.edu/courses/health-sciences-and-technology/hst-951j-medical-decision-support-fall-2005/lecture-notes/hst951 6.pdf HST.951J: Medical Decision Support.

12. Pawlak, Z. Rough Sets. International Journal on Computer and Information Sciences, 1982, 11, 341-356. https://doi.org/10.1007/BF01001956

13. Polkowski, L., Artiemjew, P. On Granular Rough Computing: Factoring Classifiers Through Granular Struc-

tures. In Kryszkiewicz, M., Peters, J. F., Rybiński, H., Skowron, A. (Eds.), RSEISP 2007, LNAI, 4585, Springer, Berlin, 2007, 280-290.

14. Polkowski, L., Artiemjew, P. Granular Computing in Decision Approximation: An Application of Rough Mereology. In Series: Intelligent Systems Reference Library, Springer, 2015. https://doi.org/10.1007/978-3-319-12880-1

15. Polkowski, L. A Unified Approach to Granulation of Knowledge and Granular Computing Based on Rough Mereology: A Survey. In Pedrycz, W., Skowron, A., Kreinovich, V. (Eds.), Handbook of Granular Computing, John Wiley & Sons, New York, 375-401, 2008. https://doi.org/10.1002/9780470724163.ch16

16. Saha, S., Murthy, C., A., Pal, S., K. Rough Set Based Ensemble Classifier for Web Page Classification. Fundamenta Informaticae, 2007, 76(1-2), 171-187.

17. Schapire, R., E. The Boosting Approach to Machine Learning: An Overview. MSRI (Mathematical Sciences Research Institute) Workshop on Nonlinear Estimation and Classification, 2003. https://doi.org/10.1007/978-0-387-21579-2_9

18. Schapire, R., E., Freund, Y. A Short Introduction to Boosting. Journal of Japanese Society for Artificial Intelligence, 14, 771-780.

19. Steinhaus, H. Remarques sur le partage pragmatique. Annales de la SocitPolonaise de Mathmatique, 1946, 19, 230-231.

20. Shi, L., Weng, M., Ma, X., Xi, L. Rough Set Based Decision Tree Ensemble Algorithm for Text Classification. Journal of Computational Information Systems, 2010, 6(1), 89-95.

21. UCI Repository; http//www.ics.uci.edu/ mlearn/databases.

22. Zhou, Z.-H. Boosting 25 years. CCL 2014 Keynote.

23. Zhou, Z.-H. Ensemble Methods: Foundations and Algorithms. Chapman and Hall/CRC, 2012, ISBN 978-1439830031.