Procedia Computer Science

Volume 51, 2015, Pages 1565–1573

ICCS 2015 International Conference On Computational Science

Wagging for Combining Weighted One-Class Support Vector Machines

Bartosz Krawczyk[1] and Michał Woźniak[1]

Department of Systems and Computer Networks,
Wrocław University of Technology, Wrocław, Poland.
{bartosz.krawczyk,michal.wozniak}@pwr.edu.pl

Abstract

Most of machine learning problems assume, that we have at our disposal objects originating from two or more classes. By learning from a representative training set a classifier is able to estimate proper decision boundaries. However, in many real-life problems obtaining objects from some of the classes is difficult, or even impossible. In such cases, we are dealing with one-class classification, or learning in the absence of counterexamples. Such recognition systems must display a high robustness to new, unseen objects that may belong to an unknown class. That is why ensemble learning has become an attractive perspective in this field. In our work, we propose a novel one-class ensemble classifier, based on wagging. A weighted version of boosting is used, and the output weights for each object are used directly in the process of training Weighted One-Class Support Vector Machines. This introduces a diversity into the pool of one-class classifiers and extends the competence of formed ensemble. Experimental analysis, carried out on a number of benchmarks and backed-up with statistical analysis proves that the proposed method can outperform state-of-the-art ensembles dedicated to one-class classification.

Keywords: ensemble learning, multiple classifier systems, one-class classification, bagging, wagging.

1 Introduction

Most of the existing machine learning algorithms assume, that the analyzed recognition task can be labeled as a binary or multi-class recognition problem. At the same time it is assumed, that we have a representative sample of each of considered classes in order to train an efficient classifier. However, in many real-life problems it is difficult, or even impossible to gather some types of examples due to various limitations. However, such challenging problems still require analytical and decision support systems to be introduced, like in a nuclear power plant monitoring [9]. For such cases one needs to apply learning in the absence of counterexamples.

One-class classification (OCC) works under an assumption, that during the classifier training stage objects coming from only a single distribution are at disposal [11]. Therefore, it creates a

Selection and peer-review under responsibility of the Scientific Programme Committee of ICCS 2015

doi:10.1016/j.procs.2015.05.351

description of target class data that will allow to characterize its unique properties. However, during the exploitation of the classifier we may observe new, incoming objects. They can belong to either the target concept, or to some unknown distributions. The latter instances are known as outliers and must be detected by one-class classifiers. OCC can be viewed as a specific type of a binary classification, with main difference lying i the training procedure utilizing objects from only a single class. This approach finds applications in scenarios, where we can gather positive examples with ease, but gathering a representative collection of counterexamples is costly, unethical, time-consuming or simply impossible [8, 10]. In OCC we are looking for methods that can display good robustness to outliers, without sacrificing their generalization abilities (preventing overfitting on the target class) [20]. Recently, the possibility of applying ensemble methods for OCC is gaining increasing attention [14].

Most of the classifier committees for OCC are based on existing methods originally designed for multi-class problems, such as boosting, bagging or random forest [5, 6, 21]. However, they do not take into consideration the specific nature of OCC problems which results in their varied performance. There are some works done on introducing pruning to OCC ensembles [4, 12] and dedicated diversity measures (as the standard ones tend to fail in this task) [13]. This all falls into a problem on how to create an efficient ensemble for OCC: how to create a pool of accurate and mutually complementary classifier, how to select the most valuable members to the committee, and how to combine their individual outputs. This steps are in general identical to forming multi-class ensembles [22], however multi-class methods cannot be used directly as one has access only to a single class during training. That is why there still is a need for introducing novel efficient ensemble algorithms dedicated to the nature of one-class problems.

In this paper, we propose a novel methodology for constructing efficient ensembles of one-class classifiers, based on random sub-sampling of the training set. Standard methods used so far apply bagging scheme. We propose to modify this by applying a weighted bagging (wagging) approach. With this we do not randomly select examples to each bag, but instead draw weights assigned to each observation according to a given probability distribution. We propose to use these weights to directly train a Weighted One-Class Support Vector Machine. Each wagging iteration constructs new weighted one-class classifier, while weights assigned to each object in a given iteration are utilized in the classifier's training process in order to assign a degree of importance to each training sample. With this, we can easily create a hybridization between wagging and weighted one-class learning. Such an approach improves the diversity of the ensemble, as different weights assigned to objects lead to different decision boundaries computed by classifiers. We compare our proposed approach to state-of-the-art one-class ensembles over a number of benchmark datasets. Furthermore, we apply different statistical tests of significance to prove the high quality of proposed one-class wagging ensembles.

2 Weighted One-Class Support Vector Machine

One-Class Support Vector Machine (OCSVM) [16] is considered among the most popular and most efficient one-class classifiers. It computes a closed boundary in a form of a hypersphere enclosing all the objects from ω_T. Object belongs to the target class, if it falls within this hypersphere. Otherwise it belongs to outliers.

OCSVM's hypersphere can be sufficiently described by two parameters': center a and a radius R. To have a low acceptance of the possible outliers the volume of this d-dimensional hypersphere, which is proportional to R^d, should be minimized to encompass all of the target class objects without any additional unoccupied decision space. The minimization of R^d implies minimization with respect to R^2. We can formulate the minimization functional as follows:

$$\Theta(a, R) = R^2, \tag{1}$$

with constraint:

$$\forall_{i\in\{1,\ldots,N\}} \quad \|x_i - a\|^2 \leq R^2, \tag{2}$$

where x_i are objects belonging to the target class, and N stands for the number of training objects. Additionally, as in a standard SVM, one may introduce slack variables ξ_i. They allow for some object to lie outside of the hypersphere and can, to some degree, filter out internal noise from the training set.

This idea can be further augmented, creating a Weighted One-Class Support Vector Machine (WOCSVM) [3]. Here, we introduce weights w_i that associate an importance measure to each of the training objects. This forces slack variables ξ_i, to be additionally controlled by w_i. If with object x_i there is associated a small weight w_i then the corresponding slack variable ξ_i indicates a small penalty. In effect, the corresponding slack variable will be larger, allowing x_i to lie further from the center a of the hypersphere. This reduces an impact of x_i on the shape of a decision boundary of WOCSVM.

To apply this, we need to modify the minimization functional:

$$\Theta(a, R) = R^2 + C\sum_{i=1}^{N} w_i\xi_i, \tag{3}$$

with the modified constraints that almost all objects are within the hypersphere:

$$\forall_{i\in\{1,\ldots,N\}} \quad \|x_i - a\|^2 \leq R^2 + \xi_i, \tag{4}$$

where $\xi_i \geq 0$, $0 \leq w_i \leq 1$. C denotes a parameter that controls the optimization process - the larger C, the less outliers are allowed with the increase of the volume of the hypersphere.

For establishing weights we may use techniques dedicated to a weighted multi-class support vector machines [6]. In this paper, we propose to use a method based on distance from the center of the hypersphere:

$$w_i = \frac{|x_i - x_{mean}|}{R + \delta}, \tag{5}$$

where $\delta > 0$ is prevents the case of $w_i = 0$. The value of x_{mean} is computed with the usage of all available learning samples.

3 Forming Ensembles of One-Class Classifiers with Wagging

The main problem in ensemble creation procedure is how to ensure the quality of the individual members of the committee. In order for the multiple classifier system to work, one needs to ensure that classifiers display high individual accuracy, while being mutually complementary to each other. Adding similar classifiers will only increase computational complexity of the ensemble, without extending it's area of competence. On the other hand combining highly diverse, but incompetent classifiers will lead to a poor ensemble. Therefore, one needs to take into consideration both of these factors [22].

In one-class classification two main approaches were used: Bagging [17] and Random Subspace [4]. This can be easily explained by a straightforward adaptation of these ensemble techniques to learning on the absence of counterexamples - they do not require class labels to work. Boosting-based methods are much more difficult to being adapted to OCC, due t being highly prone to overfitting while working only on a single class [15].

However, none of these methods can directly benefit from recently introduced weighted one-class classifiers [3]. These learners are much more efficient and robust to internal noise than standard one-class methods, but require dedicated ensemble forming algorithms to being combined in an efficient way [14]. This lead us to proposing a novel ensemble of weighted one-class classifiers, based on wagging.

Wagging [2] is a variant of Bagging method. It is also known as Weighted Bagging. Here each base classifier is trained on the entire training set, but each objects is stochastically assigned a weight.

Bagging can be considered as Wagging with weights drawn from the Poisson distribution, as each instance is represented in the bag a discrete number of times. On the other hand, Wagging often uses an exponential distribution to drawn weights from. This is because the exponential distribution is the real-value counterpart of the Poisson distribution.

Wagging offers an interesting way to modify the level of influence of each sample on the classifier's training process by differentiating weights. However, Wagging cannot be directly applied in most of the OCC methods, as they consider each object from the target class to be equally important during the training step. We propose to combine Wagging with WOCSVM classifier, and apply the weights established from wagging directly into the WOCSVM training phase (see Eq. 3).

The pseudo-code for proposed Wagging ensemble for one-class classification is presented in Alg. 1.

Algorithm 1 Wagging for forming ensembles of Weighted One-Class Support Vector Machines.

Require: WOCSVM training procedure,
number of iterations I ,
training set $\mathcal{TS}$,
weighting distribution d

1: $i \leftarrow 1$
2: **repeat**
3: $S_i \leftarrow S$ with random weights drawn from d
4: Train i-th WOCSVM on S_i according to weights assigned to each object
5: $i \leftarrow i + 1$
6: **until** $i > I$
7: Combine outputs of I trained WOCSVMs according to selected fusion method

To combine one-class classifiers, we require the knowledge about the values of support functions of each individual classifier from the pool. But WOCSVM work on the basis of distance between the new sample and its decision boundary. Therefore, to conduct the combination step we propose to use a heuristic mapping:

$$F(x, \omega_T) = \frac{1}{c_1} exp(-d(x|\omega_T)/c_2), \tag{6}$$

where $F(x, \omega_T)$ is the value of support function for a given observation x and target class ω_T. $d(x|\omega_T)$ stands for a distance metric, c_1 is the normalization constant and c_2 is the scale parameter. Parameters c_1 and c_2 should be fitted to the target class distribution.

Then we propose to use mean vote aggregation [19]:

$$F_{mv}(x, \omega_T) = \frac{1}{L} \sum_{k=1}^{L} I(F_k(x, \omega_T) \geq \theta_k), \tag{7}$$

where $F_k(x, \omega_T)$ stands for the discriminant function value returned by the kth individual classifier for a given observation x and class ω_T. $I(\cdot)$ is the *indicator function* and θ_k is a classification threshold.

The main advantages of the proposed approach are a significant reduction of the training complexity (weights are given directly by Wagging, instead of calculating them individually, e.g, according to Eq. 5) and increase of the diversity of ensemble members. Additionally, WOCSVM training scheme outputs a locally competent classifier, resulting in a committee of diverse and accurate base learners.

4 Experimental investigations

The aims of this experiment was to evaluate the effectiveness of the proposed Wagging approach for combining WOCSVMs and compare it with popular single-model and committee approaches for one-class classification.

4.1 Datasets

As there are no benchmarks dedicated to one-class classification, we have chosen 10 binary datasets - 9 from the UCI Repository and an additional one, originating from chemoinformatics domain and describing the process of discovering pharmaceutically useful isoforms of CYP 2C19 molecule. The data set is available for download at [18].

The objects from the minor class were used as the target concept, while objects from the major class as outliers.

Details of the chosen data sets are given in Table 1.

Table 1: Details of datasets used in the experimental investigation. Numbers in parentheses indicates the number of objects in the minor class in case of binary problems.

No.	Name	Objects	Features	Classes
1	Breast-cancer	286 (85)	9	2
2	Breast-Wisconsin	699 (241)	9	2
3	Colic	368 (191)	22	2
4	Diabetes	768 (268)	8	2
5	Heart-statlog	270 (120)	13	2
6	Hepatitis	155 (32)	19	2
7	Ionosphere	351(124)	34	2
8	Sonar	208 (97)	60	2
9	Voting records	435 (168)	16	2
10	CYP2C19 isoform	837 (181)	242	2

4.2 Set-up

For the experiment a Weighted One-Class Support Vector Machine with a RBF kernel is used as a base classifier. The pool of classifiers were homogeneous, i.e. consisted of classifiers of the same type.

Wagging committees consist of 10 base classifiers.

To put the obtained results into a context, we compare our method with a single WOCSVM and its bagged and boosted version (each consisting of 10 classifiers in the pool).

Classification threshold θ_k is set to 0.8 for combining one-class classifiers.

In order to present a detailed comparison among a group of machine learning algorithms, one must use statistical tests to prove, that the reported differences among classifiers are significant [7]. We use both pairwise and multiple comparison tests. Pairwise tests give as an outlook on the specific performance of methods for a given data set, while multiple comparison allows us to gain a global perspective on the performance of the algorithms over all benchmarks. With this, we get a full statistical information about the quality of the examined classifiers.

- For a pairwise comparison, we use a 5x2 combined CV F-test [1]. It repeats five-time two fold cross-validation so that in each of the folds the size of the training and testing sets is equal. This test is conducted by comparison of all versus all.
- For assessing the ranks of classifiers over all examined benchmarks, we use a Friedman ranking test [7]. It checks, if the assigned ranks are significantly different from assigning to each classifier an average rank.
- We use the Shaffer post-hoc test to find out which of the tested methods are distinctive among an n x n comparison. The post-hoc procedure is based on a specific value of the significance level α. Additionally, the obtained p-values should be examined in order to check how different given two algorithms are.

We fix the significance level $\alpha = 0.05$ for all comparisons.

4.3 Results

The results are presented in Table 2. $SINGLE$ stands for a single WOCSVM model, $BAGG$ stands for a bagged WOCSVM, $BOOST$ for a boosted WOCSVM, and $WAGG$ for the proposed method. Small numbers under each method stands for the indexes of models from which the considered one is statistically better. The last row presents ranks according to the Friedman test.

Results of the Shaffer post-hoc test between the proposed and reference methods are depicted in Table 3

4.4 Results Discussion

From the experimental results we may see, that Wagging-based ensemble proves highly competitive to other methods. In 7 cases out of 10 the obtained accuracy was highest and the differences were statistically significant. In one case (Breast-cancer dataset) Wagging achieved the highest accuracy, but the difference between it and other methods was statistically insignificant.

Only in two cases Bagging and Boosting-based ensembles of WOCSVM outperformed Wagging. This proves, that Wagging is a worthwhile choice for combining weighted one-class classifiers, as changing weights in each classifier has a more positive influence on the quality of

Table 2: Results of the experimental results with the respect to the accuracy [%] and statistical significance. Small numbers under each method stands for the indexes of models from which the considered one is statistically better.

No.	SINGLE[1]	BAGG[2]	BOOST[3]	WAGG[4]
1.	57.86 –	58.56 –	60.94 1,2	61.12 1,2
2.	87.21 –	89.52 1	89.87 1	91.45 *ALL*
3.	69.90 –	75.37 1,3	73.95 1	76.92 *ALL*
4.	58.45 –	59.21 –	59.12 –	60.88 *ALL*
5.	83.12 –	86.90 1,4	86.73 1,4	85.39 1
6.	58.23 −2	58.02 –	59.12 –	62.29 *ALL*
7.	73.52 –	79.41 1	81.04 1,2	81.80 1,2
8.	85.23 –	90.01 1,4	89.34 1,4	92.19 *ALL*
9.	87.45 –	89.32 1,4	89.71 1,4	88.05 1
10.	73.90 –	76.04 1,4	77.56 1,2,4	81.28 *ALL*
Rank	4.00	2.20	2.00	1.60

Table 3: Shaffer test for comparison between the proposed and reference methods. Symbol '=' stands for classifiers without significant differences, '+' for situation in which the method on the left is superior and '-' vice versa.

hypothesis	p-value
WAGG vs SINGLE	+ (0.0028)
WAGG vs BAGG	+ (0.0235)
WAGG vs BOOST	+ (0.0307)

the committee than using permutation of objects or simple boosting over a single class. This increases the diversity and extends the competence of the ensemble, making it more robust to potential outliers.

Interestingly Wagging performs at best for small datasets (such as Hepatitis or Sonar), significantly outperforming standard Bagging. This can be explained by a limited availability of samples for training. In small datasets, excluding a part of objects from the training procedure can significantly damper the quality of classifier (as it cannot capture the decision space properties properly). At the same time it is easy to create similar bags of objects over a small dataset. Wagging lifts those limitations, as it uses a full training set. Additionally, it enforces diversity by manipulating weights assigned to objects, not objects themselves.

5 Conclusions and Future Works

In this paper, we have presented a novel approach for creating efficient ensembles for one-class classification purposes. We applied Wagging, a weighted variation of bagging for creating data subsets for base learners. Wagging assigns a random weight to each sample drawn from a given distribution, instead of drawing samples as in standard Bagging. This changes the operation mode of the ensemble - instead of sub-sampling, it changes the importance of each sample in the training subset while using all of the available examples.

We proposed to directly utilize weights for training ensembles of weighted one-class classifiers. WOCSVM requires weights assigned to each object in order to establish their level of influence over the process of decision boundary estimation. Weights drawn from Wagging were inputed into the training phase of WOCSVM. By this we had reduced the training complexity and increased diversity of ensemble members.

Experimental analysis, backed-up with a thorough statistical analysis had proven the usefulness of our method.

In future, we plan to analyze the influence of the size of Wagging committees on their accuracy, propose a diversity measure dedicated specifically to weighted one-class classifiers, and to add a pruning step into our Wagging-based one-class ensemble.

Acknowledgments

This work was partially supported by The Polish National Science Centre under the grant PRELUDIUM number DEC-2013/09/N/ST6/03504 and by EC under FP7, Coordination and Support Action, Grant Agreement Number 316097, ENGINE European Research Centre of Network Intelligence for Innovation Enhancement (http:// engine.pwr.wroc.pl/).

References

[1] Ethem Alpaydin. Combined 5 x 2 cv f test for comparing supervised classification learning algorithms. *Neural Computation*, 11(8):1885–1892, 1999.

[2] E. Bauer and R. Kohavi. An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine Learning*, 36(1-2):105–139, 1999.

[3] M. Bicego and M. A. T. Figueiredo. Soft clustering using weighted one-class support vector machines. *Pattern Recognition*, 42(1):27–32, 2009.

[4] V. Cheplygina and D. M. J. Tax. Pruned random subspace method for one-class classifiers. In *Multiple Classifier Systems*, volume 6713 LNCS of *Lecture Notes in Computer Science*, pages 96–105, 2011.

[5] B. Cyganek. Image segmentation with a hybrid ensemble of one-class support vector machines. volume 6076 LNAI of *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, pages 254–261. 2010.

[6] B. Cyganek. One-class support vector ensembles for image segmentation and classification. *Journal of Mathematical Imaging and Vision*, 42(2-3):103–117, 2012.

[7] J. Demsar. Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, 7:1–30, 2006.

[8] A. B. Gardner, A. M. Krieger, G. Vachtsevanos, and B. Litt. One-class novelty detection for seizure analysis from intracranial eeg. *Journal of Machine Learning Research*, 7:1025–1044, 2006.

[9] K. Jackowski and J. Platos. Application of adass ensemble approach for prediction of power plant generator tension. In Jos Gaviria de la Puerta, Ivn Garca Ferreira, Pablo Garcia Bringas, Fanny

Klett, Ajith Abraham, Andr C.P.L.F. de Carvalho, lvaro Herrero, Bruno Baruque, Hctor Quintin, and Emilio Corchado, editors, *International Joint Conference SOCO14-CISIS14-ICEUTE14*, volume 299 of *Advances in Intelligent Systems and Computing*, pages 207–216. Springer International Publishing, 2014.

[10] H. Jiang, G. Liu, X. Xiao, C. Mei, Y. Ding, and S. Yu. Monitoring of solid-state fermentation of wheat straw in a pilot scale using ft-nir spectroscopy and support vector data description. *Microchemical Journal*, 102, 2012.

[11] M. W. Koch, M. M. Moya, L. D. Hostetler, and R. J. Fogler. Cueing, feature discovery, and one-class learning for synthetic aperture radar automatic target recognition. *Neural Networks*, 8(7-8):1081–1102, 1995.

[12] B. Krawczyk. One-class classifier ensemble pruning and weighting with firefly algorithm. *Neurocomputing*, 150:490–500, 2015.

[13] B. Krawczyk and M. Woźniak. Diversity measures for one-class classifier ensembles. *Neurocomputing*, 126:36–44, 2014.

[14] B. Krawczyk, M. Woźniak, and B. Cyganek. Clustering-based ensembles for one-class classification. *Inf. Sci.*, 264:182–195, 2014.

[15] G. Ratsch, S. Mika, B. Scholkopf, and K. . Muller. Constructing boosting algorithms from svms: An application to one-class classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1184–1199, 2002.

[16] B. Schölkopf and A.J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. Adaptive computation and machine learning. MIT Press, 2002.

[17] S. Seguí, L. Igual, and J. Vitrià. Bagged one-class classifiers in the presence of outliers. *IJPRAI*, 27(5), 2013.

[18] SIAM. *Proceedings of the Eleventh SIAM International Conference on Data Mining, SDM 2011, April 28-30, 2011, Mesa, Arizona, USA*. SIAM Omnipress, 2011. http://tunedit.org/challenge/QSAR.

[19] David M. J. Tax and Robert P. W. Duin. Combining one-class classifiers. In *Proceedings of the Second International Workshop on Multiple Classifier Systems*, MCS '01, pages 299–308, London, UK, 2001. Springer-Verlag.

[20] D.M.J. Tax and Robert P. W. Duin. Characterizing one-class datasets. In *Proceedings of the Sixteenth Annual Symposium of the Pattern Recognition Association of South Africa*, pages 21–26, 2005.

[21] T. Wilk and M. Woźniak. Soft computing methods applied to combination of one-class classifiers. *Neurocomput.*, 75:185–193, January 2012.

[22] M. Woźniak, M. Grana, and E. Corchado. A survey of multiple classifier systems as hybrid systems. *Information Fusion*, 16(1):3–17, 2014.