Proceedings of the Federated Conference on
Computer Science and Information Systems pp. 89–92

ISBN 978-83-60810-51-4

Experiments on distance measures for combining one-class classifiers

Bartosz Krawczyk*
Department of Systems and Computer Networks
Wroclaw University of Technology
Wybrzeze Wyspianskiego 27, Wroclaw 50-370, Poland
Email: bartosz.krawczyk@pwr.wroc.pl

Michał Woźniak
Department of Systems and Computer Networks
Wroclaw University of Technology
Wybrzeze Wyspianskiego 27, Wroclaw 50-370, Poland
Email: michal.wozniak@pwr.wroc.pl

***Abstract*—The paper investigates the influence of different types of distance measures on the performance of a multiple classifier system consisting of one-class classifiers. This specific type of machine learning approach uses examples only from a single class to derive a decision boundary - hence its is often referred to as learning in the absence of counterexamples. Combining several one-class classifiers is a promising research direction, as it often results in a more precise classification than when using just a single model. Most one-class classifiers base their decision on a distance from an object to the decision boundary, canonically expressed in the Euclidean measure. When combining such predictors it is necessary to map the distance into probability, therefore the measure used has a crucial impact on the classifier fusion. This paper proposes alternative distance measures for one-class classification, which are evaluated through experimental investigations.**

***Index Terms*—one-class classification, multiple classifier systems, distance measures, one-class support vector machine, combined classifier.**

I. Introduction

ONE-CLASS classification (OCC) is known as learning in the absence of counterexamples, as primary object of OCC is to train a classifier using only patterns drawn from the target class distribution. Its main goal is to detect anomaly or a state other than the one for the target class [18]. It is assumed that only information of the target class is available. Therefore no information about the potential nature of outlier objects are needed to derive the decision boundary. Schema of OCC problem is given in Fig. 1.

This is a very useful approach in may real-life applications e.g. machine diagnosis, when we know what is the proper response from the device, but number of ways in which such a device may malfunction are numerous. Therefore instead of costly and time consuming generation of negative examples one may use OCC to get the classification boundary only on the basis of available samples. Various terms have been used in the literature to refer to one-class learning approaches. The term single-class classification originates from [11], but also outlier detection [7] or novelty detection [2] are used to name this field of study. Another view on the OCC is that it seeks to distinguish one specific class from the more broad set of classes (e.g., selecting apples from fruits, gearbox error from machine faults or fraud from set of transactions). The target class is considered as a positive one, while all other are considered as negative ones.

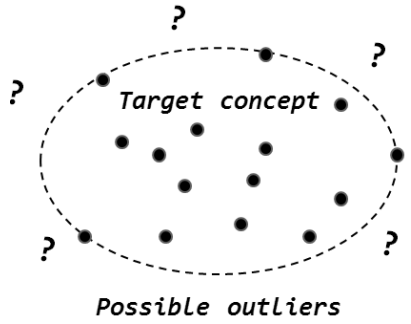

Fig. 1. Example of one-class classification with known target class used for the boundary creation. One-class classification assumes that in the exploratory phase of classification new, unknown objects not belonging to the target class may appear.

Usually for a given problem we may have a pool of several classifiers at our disposal. Canonical machine learning methods concentrate on selecting the single best classifier from the pool and delegating him to the problem solving task. This approach seems very reasonable and is rooted in a normal human behavior - when having a problem we tend to search for the most competent expert in a given area, not paying attention to lesser renown specialists. Yet when referencing to only a single classifier we discard the fact that other models from the pool may also offer a valuable contribution to the considered problem. That is why a combined approach was proposed, utilizing decisions of more than one classifier. Such methods, known as Multiple Classifier Systems (MCS) are considered as one of the most promising research directions in current field of machine learning and pattern recognition [10].

Using MCS in OCC is an approach that still awaits proper attention. So far several approaches were proposed, based on a simple bagging [13], boosting [21] or random subspaces [4]. Most of the works in this topics were application-oriented, e.g. on image retrieval [20] or on monitoring the information network [14]. Therefore there is a lack of works devoted to the theoretical advances in combination of one-class classifiers. Recently authors have proposed the diversity measures for selecting one-class predictors to the ensemble [12].

978-83-60810-51-4/$25.00 © 2012 IEEE

Several methods for combining one-class classifiers were proposed in [17]—they will be described in more details in Sec. II. One should notice that most one-class predictors base their decision on the distance from an object to the decision boundary. Therefore to perform the classifier fusion one must map the distance into a probability. Canonically used in OCC was Euclidean distance. This paper proposes to use different measures of distance and investigates their influence on the overall accuracy of the combined classifier.

The paper is organized in the following way: next section gives an overview of fusion methods for one-class ensembles, section three presents used distance measures, while following section is devoted to experimental assessment of the quality of proposed measures. Last section concludes the paper.

II. Combining one-class classifiers

One-class boundary methods (such as One-class Support Vector Machine) are based on computing the distance between the object x and target class ω_T . To apply fusion methods we require the probability (or classification support) of object x for a given class. Therefore to conduct the fusion a heuristic mapping must be made. This paper uses a following solution:

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

which models a Gaussian distribution around the classifier, where $d(x|\omega_T)$ is 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.

After performing such a mapping one may use proposed fusion functions. This paper applies all five of them, with the assumption that the pool consists of R one-class classifiers:

1) **Mean vote**, which combines binary output labels of one-class classifiers. It is expressed by:

$$y_{mv}(x) = \frac{1}{R} \sum_k I(P_k(x|\omega_T) \geq \theta_k), \tag{2}$$

where $I(\cdot)$ is the *indicator function* and θ_k is a classification threshold. When a threshold equal to 0.5 is applied this rule transforms into a majority vote for binary problems.

2) **Mean weighted vote** which introduces the weighting of base classifiers by f_{T_k}, where f_{T_k} is the fraction of target class objects accepted by k-th classifier:

$$\begin{aligned} y_{mwv}(x) = \tfrac{1}{R} \textstyle\sum_k f_{T_k} I(P_k(x|\omega_T) \geq \theta_k) + \\ +(1 - f_{T_k}) I(P_k(x|\omega_T) \leq \theta_k), \end{aligned} \tag{3}$$

which is a smoothed version of the mean vote method.

3) **Product of the weighted votes**, for which let's first define:

$$prod = \prod_k f_{T_k} I(P_k(x|\omega_T) \geq \theta_k), \tag{4}$$

and then the fusion method:

$$y_{pwv}(x) = \frac{prod}{prod + \prod_k (1 - f_{T_k}) I(P_k(x|\omega_T) \leq \theta_k)}, \tag{5}$$

4) **Mean of the estimated probabilities** which is expressed by:

$$y_{mp}(x) = \frac{1}{R} \sum_k (P_k(x|\omega_T). \tag{6}$$

5) **Product combination of the estimated probabilities**, which is expressed by:

$$y_{pc}(x) = \frac{\prod_k P_k(x|\omega_T)}{\prod_k P_k(x|\omega_T) + \prod_k \theta_k}. \tag{7}$$

This fusion method assumes that the outlier object distribution is independent of x and thus uniform in the area around the target concept.

III. Distance measures

As one may see from the previous section, used distance measure has a major impact on the fusion of one-class classifiers. Canonically for this problem the Euclidean squared distance was used [17]. Here we would like to investigate the performance of other popular distance measures and their influence on the classifier fusion step.

There exists a significant number of distance measures derived from many various fields such as mathematics, physics, information theory, computer science and econometrics. Good review of existing distance measures is presented in [3]. Many machine learning algorithms, such as minimal distance classifiers [5], rely on the distance metric for the input data patterns. In recent years, many studies have demonstrated, both empirically and theoretically, that a proper metric can significantly improve the performance in classification, clustering and retrieval tasks [9].

A distance metric, $d(m, n)$ is a function for calculating a distance between two objects, m and n. To consider given function as a distance metric it has to fulfill following three properties [19]:

1) It is always greater than or equal to zero.
2) The distance from object to itself is always equal to zero.
3) It obeys the triangle inequality, i.e for three points m, n and o, $d(m, n) + d(n, o) \geq d(m, o)$ for any choice of n.

This paper investigates the performance of five popular distance metrics, assuming that each object is described by i features:

1) **Euclidean squared distance**:

$$d_E(m, n) = \sqrt{\sum_i (m_i - n_i)^2} \tag{8}$$

2) **Canberr distance**:

$$d_C(m, n) = \sum_i \left| \frac{m_i - n_i}{m_i + n_i} \right|. \tag{9}$$

3) **Czebyszew distance**:

$$d_{CZ}(m,n) = \max_i(|m_i - n_i|). \quad (10)$$

4) **Manhattan distance**:

$$d_M(m,n) = \sum_i(|m_i - n_i|). \quad (11)$$

5) **Pearson correlation distance**: in which one must first compute the correlation parameter r:

$$r = \frac{1}{i}\sum_i \left(\frac{m_i - \bar{m}}{\sigma_m}\right)\left(\frac{n_i - \bar{n}}{\sigma_n}\right) \quad (12)$$

where $\bar{m}, \bar{n}$ are average values and σ_m, σ_n are standard deviations. Using this parameter, one may define a distance metric as:

$$d_P(m,n) = 1 - r, \quad (13)$$

where $d_P(m,n)$ takes values from [0,2].

IV. Experimental investigations

The aim of the experiments was to investigate if using different distance measures have an impact on the quality of one-class classifier fusion.

A. Datasets

For experimental evaluation of our proposition we have selected five binary datasets from the UCI Repository [6]. Details of used datasets are given in Tab. I. It is worth noticing

TABLE I
DETAILS OF DATASETS USED IN THE EXPERIMENTAL INVESTIGATION. NUMBERS IN PARENTHESES INDICATES THE NUMBER OF OBJECTS IN THE MINORITY CLASS.

No.	Name	Objects	Features	Classes
1	Breast-cancer	286 (85)	9	2
2	Diabetes	768 (268)	8	2
3	Heart-statlog	270 (120)	13	2
4	Ionosphere	351(124)	34	2
5	Voting records	435 (168)	16	2

that so far there have been no benchmarks for one-class datasets and only solution on how to asses the performance of new methods for this field was to treat one-class problems as a decomposition of multi-class benchmarks, as in our previous work [12]. This paper uses a simple, unified and effective scheme for transformation of multi-class sets into canonical one-class problems.

Let's assume that a dataset consist of objects drawn form class set $\mathcal{M}$ = {1, 2, ..., M}. Class $m \in \mathcal{M}$ is chosen to become target class ω_T. All other objects from $\widehat{\mathcal{M}}$ = {1, 2, ..., M} $\setminus$ $\{m\}$ become outliers objects with labels ω_O. The pool of R individual classifiers $\Psi^{(1)}, \Psi^{(2)}, ..., \Psi^{(R)}$ is then trained with normal procedure (such as cross-validation) on the objects from class ω_T while objects from ω_O are used for the testing phase. One should notice that from a single M-class problem this procedure may derive M separate one-class datasets.

In this work as a target class we have chosen the minority class, while the remaining class was used as outliers.

B. Set-up

As a base classifier a One-class Support Vector Machine (OCSVM) [16] with a polynomial kernel was selected. The pool of classifiers were constructed using a random subspace method [8] and consisted of five models. The focus of this paper is not on the problem of selecting classifiers to the committee, therefore always an entire pool was used.

The combined 5x2 cv F test [1] was carried out to asses the statistical significance of the obtained results.

All experiments were carried out in the R language [15].

C. Results and discussion

The results of the experiment are given in Tab. II – VI. Small numbers under classification accuracy indicates from which methods the given one was statistically better.

TABLE II
ACCURACY OF THE PROPOSED METHODS OVER THE BREAST-CANCER DATASET.

Fusion method	d_E^1	d_C^2	d_{CZ}^3	d_M^4	d_P^5
mv	84.32	**86.00**	85.40	80.86	**86.00**
	4	1,4	4	–	1,4
mwv	86.00	86.43	85.03	82.25	**87.30**
	4	4	4	–	1,3,4
pwv	83.25	83.25	82.15	79.13	**85.05**
	4	4	4	–	3,4
mp	83.25	**84.15**	84.15	80.60	83.90
	4	4	4	–	4
pc	85.36	85.36	83.25	83.00	**86.90**
	3,4	3,4	–	–	3,4

TABLE III
ACCURACY OF THE PROPOSED METHODS OVER THE DIABETES DATASET.

Fusion method	d_E^1	d_C^2	d_{CZ}^3	d_M^4	d_P^5
mv	79.05	**80.12**	79.78	78.15	**80.12**
	–	–	–	–	–
mwv	**81.05**	80.25	80.25	78.15	80.25
	4	4	4	–	4
pwv	77.60	75.10	76.00	74.20	**79.10**
	2,3,4	–	4	–	2,3,4
mp	78.53	77.90	**80.20**	77.90	79.15
	2,4	–	1,2,4	–	2,4
pc	**80.11**	78.10	77.56	77.56	79.21
	2,3,4	–	–	–	3,4

TABLE IV
ACCURACY OF THE PROPOSED METHODS OVER THE HEART-STATLOG DATASET.

Fusion method	d_E^1	d_C^2	d_{CZ}^3	d_M^4	d_P^5
mv	**84.00**	81.23	82.90	77.56	83.00
	2,3,4	4	2,4	–	2,4
mwv	**82.90**	80.11	79.46	78.34	80.95
	2,3,4,5	4	–	–	3,4
pwv	**85.05**	84.60	84.60	81.00	**85.05**
	3,4	4	4	–	4
mp	**82.90**	80.11	79.46	77.50	81.90
	2,3,4,5	4	–	–	3,4
pc	**84.00**	83.10	82.45	82.00	83.52
	3,4	4	–	–	4

From the experimental results one may clearly see that there is not a single distance measure outperforming other for every

TABLE V
ACCURACY OF THE PROPOSED METHODS OVER THE IONOSPHERE DATASET.

Fusion method	d^1_E	d^2_C	d^3_{CZ}	d^4_M	d^5_P
mv	71.87	72.03	**73.76**	72.00	**73.76**
	–	–	–	–	–
mwv	73.45	74.00	**75.33**	74.65	75.00
	–	–	1,2	1	1,2
pwv	70.75	71.20	70.98	70.75	**74.05**
	–	–	–	–	1,2,3,4
mp	69.53	**72.20**	**72.20**	70.00	70.65
	–	1,4	1,4	–	1
pc	72.45	72.45	**73.50**	73.02	73.35
	–	–	1,2	–	–

TABLE VI
ACCURACY OF THE PROPOSED METHODS OVER THE VOTING RECORDS DATASET.

Fusion method	d^1_E	d^2_C	d^3_{CZ}	d^4_M	d^5_P
mv	**85.11**	84.04	83.78	81.90	**85.11**
	3,4	4	4	–	3,4
mwv	**87.23**	85.93	84.37	83.90	86.11
	2,3,4,5	3,4	–	–	3,4
pwv	**83.80**	83.55	**83.80**	82.90	83.41
	–	–	–	–	–
mp	**85.79**	83.20	85.15	84.00	85.15
	2	–	2	–	2
pc	**84.25**	82.00	81.36	81.36	83.68
	2,3,4	–	–	–	2,3,4

instance tested. Canonical Euclidean distance proved itself best in all fusion methods for two datasets - Heart-statlog and Voting records. Yet for other three tested benchmarks different distance measures improved performance of the ensemble. Most frequently the best results were returned by Czebyszew and Pearson correlation distance measures. One should notice that in many cases the differences were small, but statistically significant. Manhattan distance returned worst results, with the exception of the Ionosphere dataset, where for all fuser setting it has outperformed the Euclidean distance.

Those results prove that investigating different measures than Euclidean distance for one-class classifiers is a promising research direction.

V. CONCLUSIONS AND FUTURE WORKS

This paper addressed the issue of distance metrics used in one-class classification. Boundary methods, such as used one-class Support Vector Machine rely their decision on the distance from an examined object to the decision surface. While combining several one-class classifiers one must additionally map the distance into the probability. Therefore the problem of measuring the distance has a strong influence on the creation of combined one-class classifiers.We have examined five different distance measures for five different classifier fusion blocks.

Experimental investigation, backed up with a statistical test of significance showed that in some cases switching to a different distance metric may lead to an improvement of overall accuracy of the ensemble. This proves that researching the distance metric problem for combining one-class classifiers is a worthwhile research direction.

Our future works will include incorporating more sophisticated distance measures e.g. based on entropy, investigating the relation between the feature space of datasets and performance of metrics on them and introducing metric learning system for one-class classification, which may automatically select the most promising distance metric for a given dataset.

Acknowledgements This work is supported by the Polish National Science Centre under a grant for the period 2011-2014.

REFERENCES

[1] Alpaydin, E. (1999). Combined 5 x 2 cv f test for comparing supervised classification learning algorithms. Neural Computation, 11(8):1885-1892.
[2] Bishop, C. M. (1994). Novelty detection and neural network validation. IEEE Proceedings: Vision, Image and Signal Processing, 141(4), 217–222.
[3] Cha, S.H. (2007). Comprehensive study on distance/similarity measures between probability density functions. International Journal of Mathematical Modeling and Methods in Applied Sciences, 1(4),300–307.
[4] Cheplygina, V., and Tax, D. M. J. (2011). Pruned random subspace method for one-class classifiers. Lecture Notes in Computer Science, Volume 6713 LNCS, 96–105.
[5] Domeniconi, C., Peng, J., and Gunopulos, D. (2002). Locally adaptive metric nearest-neighbor classification. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(9), 1281-1285.
[6] Frank, A. and Asuncion, A. (2010). UCI machine learning repository,http://archive.ics.uci.edu/ml.
[7] Hodge, V. J., and Austin, J. (2004). A survey of outlier detection methodologies. Artificial Intelligence Review, 22(2), 85–126.
[8] Ho, T. K. (1998). The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(8):832–844.
[9] Hu, Q., Yu, D., and Xie, Z. (2008). Neighborhood classifiers. Expert Systems with Applications, 34(2), 866-876.
[10] Jain, A. K. Duin, R. P. W., and Mao, J. (2000). Statistical pattern recognition: A review. IEEE Transactions on Pattern Analysis and Machine Intelligence, 22(1), 4–37.
[11] Koch, M. W., Moya, M. M., Hostetler, L. D., and Fogler, R. J. (1995). Cueing, feature discovery, and one-class learning for synthetic aperture radar automatic target recognition. Neural Networks, 8(7-8), 1081–1102.
[12] Krawczyk, B., and Woźniak, M. (2012). Combining diverse one-class classifiers.Volume 7209 LNAI, Issue PART 2, 590–601.
[13] Li, C., and Zhang, Y. (2008). Bagging one-class decision trees. Proceedings of 5th International Conference on Fuzzy Systems and Knowledge Discovery, FSKD 2008,2, 420–423.
[14] Mazhelis, O., ad Puuronen, S. (2004). Combining one-class classifiers for mobile-user substitution detection. Paper presented at the ICEIS 2004 - Proceedings of the Sixth International Conference on Enterprise Information Systems, 130-137.
[15] R Development Core Team. (2008). R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria, ISBN 3-900051-07-0.
[16] Scholkopf, B. and Smola, A. J. (2002). Learning with kernels: support vector machines, regularization, optimization, and beyond. Adaptive computation and machine learning. MIT Press.
[17] Tax, D.M.J. and Duin, R. P. W. (2001). Combining one-class classifiers. In Proceedings of the Second International Workshop on Multiple Classifier Systems, MCS '01, 299–308.
[18] Tax, D.M.J. and Duin, R. P. W. (2005). Characterizing one-class datasets. In Proceedings of the Sixteenth Annual Symposium of the Pattern Recognition Association of South Africa, pages 21–26.
[19] Wilson, D. R., and Martinez, T. R. (1997). Improved heterogeneous distance functions. Journal of Artificial Intelligence Research, 6, 1–34.
[20] Wu, R. -., and Chung, W. -. (2009). Ensemble one-class support vector machines for content-based image retrieval. Expert Systems with Applications, 36(3), 4451-4459.
[21] Yeh, C. -., Lee, Z. -., and Lee, S. -. (2009). Boosting one-class support vector machines for multi-class classification. Applied Artificial Intelligence, 23(4), 297-315.