Deleting or Keeping Outliers for Classifier Training?

Antonio J. Tallón-Ballesteros and José C. Riquelme
Department of Languages and Computer Systems
University of Seville
Seville, Spain
e-mail: atallon@us.es

***Abstract*—This paper introduces two statistical outlier detection approaches by classes. Experiments on binary and multi-class classification problems reveal that the partial removal of outliers improves significantly one or two performance measures for C4.5 and 1-nearest neighbour classifiers. Also, a taxonomy of problems according to the amount of outliers is proposed.**

Keywords-outlier detection; classification; statistical outlier detection; partial removal;inter-quartile range;attribute noise

I. INTRODUCTION

Data mining is a growing area of computer science with important challenges. Among the major issues, we underline the mining methodology [1] covering topics like i) mining knowledge in multidimensional space, e.g. looking for interesting patterns or ii) handling uncertainty, noise, or incompleteness of data. Data often contain noise, errors or exceptions. Errors and noise may confuse the data mining process, leading to the derivation of erroneous patterns.

Data pre-processing or data preparation [2] comprises those techniques concerned with analyzing raw data such as data cleaning. The importance of data preparation is due to several aspects. Two of them are especially relevant: a) real-world data are impure (incomplete, noisy and inconsistent) which can disguise useful patterns. Noisy data are those containing errors or outliers; b) quality data yields high-quality patterns. In relation to this topic we can purify data: correcting errors or removing outliers.

Data cleaning, outlier detection and removal are examples of techniques that need to be addressed within the data mining process [1]. A good number of tasks or functionalities can be used to specify the knowledge to be found in data mining such as classification, regression, clustering and outlier detection. This paper focuses on noise handle by means of outlier detection in classification problems. In relation to noise, we specifically have considered outliers included in the original data set instead of adding artificial noise to the problem that is a very common practice in order to evaluate the robustness of a classifier at hand. Hereinafter, the terms outlier and noise are used in an interchangeably way.

The routines for data cleaning attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data. Although most mining methods have some procedures for dealing with noisy data, they are not always robust. Outlier detection, also known as anomaly detection, is the process of finding data objects with behaviours that are very different from expectation. Such objects are called outliers or anomalies.

This paper goals to evaluate the outlier effect in classification problems and to shed light on the controversial issue about the outlier removal. For this purpose, two statistical outlier detection methods are proposed.

II. RELATED WORK

Several definitions of outlier have been presented in the data mining literature. We have selected two of them: i) an outlier is defined as a data point which is very different from the rest of the data based on some measure [3] and ii) an outlier is a case that does not follow the same model as the rest of the data and appears as though it comes from a different probability distribution [4].

As such, an outlier does not only include erroneous data but also surprisingly correct data. The detection of outlier is a procedure that selects k samples that are considerably dissimilar, exceptional, or inconsistent with respect to the remaining data [5].

There are controversial approaches to outlier dropping. On one hand, it is claimed that if outliers are removed completely, information lost may happen [1]; we include some citations: a) “Eliminating instances which contain attribute noise is not a good idea, because many other attributes of the instance may still contain valuable information” [4]; b) in a medical domain, Klawikowski et al. [6] “found that the removal of outliers did not significantly change the overall error distribution, accuracy, …”. On the other hand, there are works that showed successful results with their deletion such as [7] and [8].

According to the output, outlier detection methods can be categorized into labelling and scoring techniques. Labelling methods partition the data into two non-overlapping sets (outliers and non-outliers) and scoring methods offer a ranking list by assigning to each datum a factor reflecting its degree of outlierness [9].

Outliers can be classified into three categories [1], namely global outliers, contextual (or conditional) outliers, and collective outliers. A data object is a global outlier if it deviates significantly from the rest of the data set. An object is a contextual outlier if it deviates significantly with respect to a specific context of the object. A subset of data objects forms a collective outlier if the objects as a whole deviate significantly from the entire data set.

Other criterion divides outlier detection methods based on assumptions about outliers versus the rest of the data. According to the assumptions made, we can categorize outlier detection methods into four types: statistical methods, proximity-based methods, clustering-based methods and classification-based methods. The three first kinds are described deeply in [1]. Statistical methods (also known as model-based methods) make assumptions of data normality. They assume that normal data objects are generated by a statistical model, and that data not following the model are outliers. Proximity-based methods assume that an object is an outlier if the nearest neighbours of the object are far away in feature space, that is, the proximity of the object to its neighbours significantly deviates from the proximity of most of the other objects to their neighbours in the same data set. Clustering-based methods assume that the normal data objects belong to large and dense clusters, whereas outliers belong to small or sparse clusters, or do not belong to any clusters. Classification-based methods are those that rely on one or more preliminary classifiers built as references for deciding which data instances are incorrectly classified and should be removed [10]; some papers falling into this category are [8] and [11].

This paper is in the line of the first category and this is the reason to add more details next. In general, statistical methods for outlier detection [1] can be divided into two major categories: parametric methods and non-parametric methods, according to how the models are specified and learned. A parametric method assumes that the normal data objects are generated by a parametric distribution. A nonparametric method does not assume an a priori statistical model. Instead, a nonparametric method tries to determine the model from the input data. The former includes representative procedures like those based on inter-quartile range or parametric distributions. The latter is covered by techniques like histograms or kernel density estimation.

III. PROPOSED METHODS

Often, outlier detection methods consider instances from different classes in the same data set. According to the definition, statistical methods assume that normal data objects are generated by a statistical model.

This paper proposes a statistical outlier detection method to determine the outliers based on inter-quartile range (IQR) by classes. The idea is to divide the training set in as many partitions as classes are in the problem. An instance is considered an outlier if at least the value of one attribute is an exceptional value with relation to the values of the remaining instances in the same class. The previous sentence leads us to introduce a new kind of outlier that we have named intra-class outlier. On one hand, it might be considered as a special type of contextual outlier, however the former is based in the instances of the same class and latter takes into account the context as a part of the problem that is affected by contextual and behavioural attributes. On the other hand, also an intra-class outlier may be considered a particular type of a global outlier where a restricted data set composed by the instances of the same class is included. In our opinion, the term intra-class outlier could result simpler and contains important ideas about the scope. Once we have collected the outliers for each class then we combine all of them in the outlier set. Getting the outliers by classes might be very useful for the experts in order to study them according with the current interest. The outright removal of outliers is an unclear question as the literature reported.

Mathematically, the IQR is the difference between the third (Q3) and the first quartiles (Q1). An outlier may be defined as a point that is: a) a number of times the standard deviations out of Q1 and Q3, or in other way b) a certain number of IQR times out of Q1 and Q3 (a typical value for this number is 1.5 in the case of box-plots [1]). We have restricted the condition to points that are three times the IQR out from Q1 and Q3. Thus, the current proposal will consider as outlier any instance I with a $\mathbf{x}_i$ value for the attribute $\mathbf{x}$ which fulfils the following expression:

$$outlier(I, \mathbf{x}_i) \; if \begin{cases} \mathbf{x}_i > \mathbf{x}_{Q3} + 3 * \mathbf{x}_{IQR} \\ \; or \\ \mathbf{x}_i < \mathbf{x}_{Q1} - 3 * \mathbf{x}_{IQR} \end{cases} \quad (1)$$

In order to complete a deeper study, we have proposed two approaches. The first one removes completely by classes all the outliers from the training set while the test set remains unchanged; we have given to it the name of OUTLIERoutF, which stands for Outlier out Fully. The second one, called OUTLIERoutP (it means Outlier out Partially), divides the outlier set containing the outlier from all classes, once the outlier detection method has been executed for each class, in two random stratified halves; alternatively, in each trial one half is added to the training set and the classification algorithm is evaluated with the model obtained with the extended training set -that is, the original training set plus the instances on a half of the outlier set- using the unseen instances of the test set. Figs. 1 and 2 depict the proposals, called OUTLIERoutF and OUTLIERoutP, respectively.

The idea of applying outlier detection by classes has been proposed in some previous works. For instance, Laurikkala et al. [12] identified outliers in an informal way via box-plots for two medical problems. They concluded that the benefit obtained by excluding outliers is data set dependent.

IV. TAXONOMY OF DATA SETS ACCORDING TO THE OUTLIER AMOUNT

Generally speaking, it is more likely non-outlier instances than outlier ones. Zhu and Wu [4] carried out experimentation on a good number of problems and due to most of the datasets do not contain noise they add it in an artificial way by manual mechanisms. It is a fact that the research in outlier detection is not new; however, the taxonomy of problems depending on the instances composition with regard to the quantity of outliers has never been addressed, to the best of our knowledge.

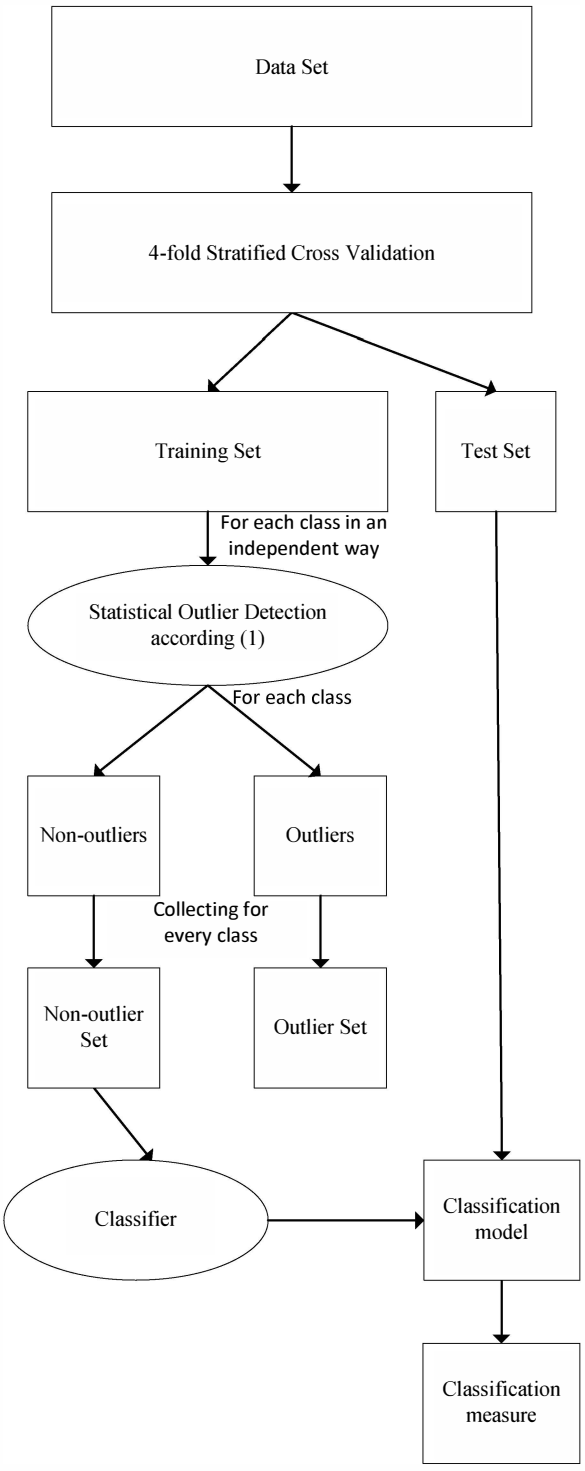

Figure 1. OUTLIERoutF framework.

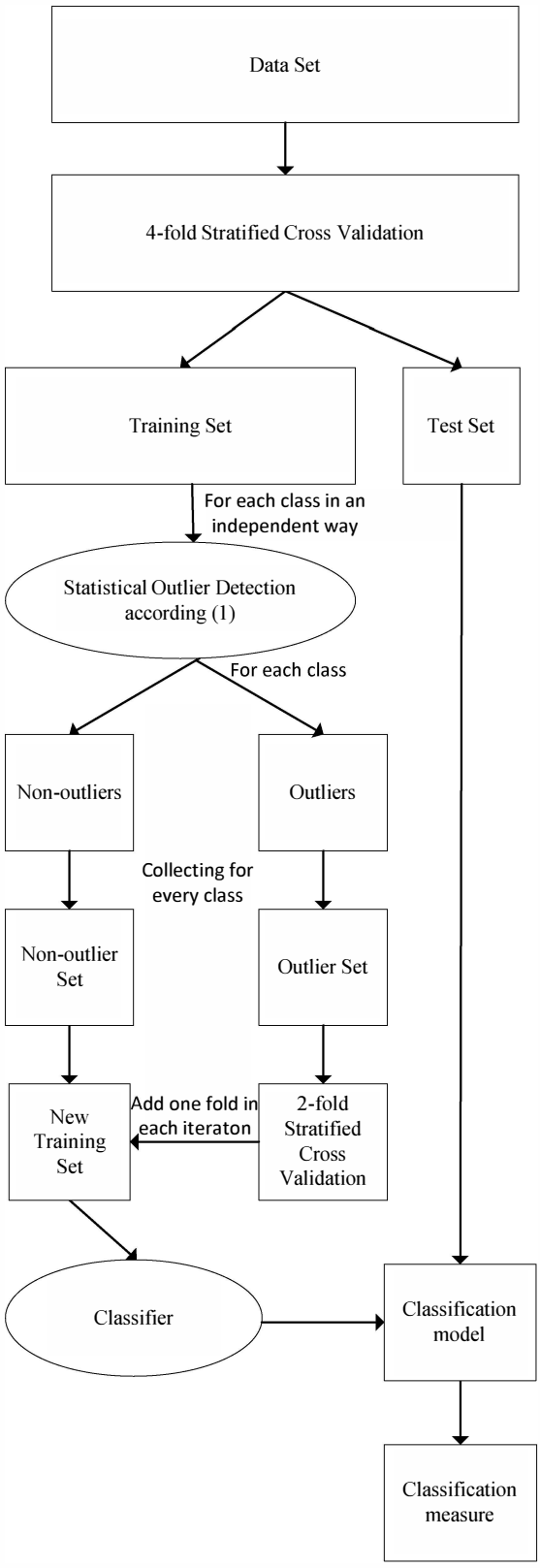

Figure 2. OUTLIERoutP framework.

This paper proposes to group the data sets into the following four categories according to the outlier percentage in the training set (OPTra):

- Level I. The outlier percentage is included in the]0-10[range.
- Level II. For problems with a percentage belonging to [10-20[.
- Level III. This category represents data sets with [20-30[% of outliers.
- Level IV. The last level is for problems containing outliers with proportions in [30-40[% of the instances.

The OPTra depends on the problem and the cross validation procedure; it should be averaged between the values of the different training sets. It justifies that the levels include a wide range. The information about the outlier amount may be very interesting to be specified in any general-purpose research on data mining because outliers can hinder the performance. The outlier percentage in the test set (OPTst) is unknown, but we should bear in mind that if an outlier happened in the scope of a particular problem with the current data, it may occur again in the future or at a certain time period. It is obvious, that data sets without outliers have no place in the previous taxonomy.

V. EXPERIMENTAL SETTING

The experimentation is carried out with ten binary and multi-class classification problems from the University of California at Irvine [13] with different outlier levels. The experimental design follows a 4-fold stratified cross validation. Table I summarizes the properties of the different data sets with special emphasis to the last column that represent the outlier level of the problem according to the taxonomy proposed in the previous section. Most of them belong to level II. All data sets have been minimally pre-processed, that is, the missing values have been replaced by the mode in the case of nominal variables or by the mean for continuous variables, taking into account the full data set. Authors would like to express that no outlier has been artificially added into the problems.

TABLE I. SUMMARY OF DATA SETS

Data set	Size	Features	Classes	Outlier level
CTG2	2126	23	2	II
CTG3	2126	23	3	II
german	1000	61	2	I
liver	345	6	2	I
magic	14265	11	2	I
olitos	120	26	4	II
pima	768	8	2	I
sleep	105908	13	5	II
tokyo	959	44	2	IV
water	527	38	3	II

Three well-known machine learning algorithms, namely C4.5 [14], 1-nearest neighbour (1-NN) [15] and SVM (Support Vector Machines) [16], have been tested with the original data sets, with the data sets after the application of OUTLIERoutF and once OUTLIERoutP has been carried out. We have used the implementations of the three aforementioned classifiers provided in WEKA tool [17] with the default parameters that are those recommended by the own authors of the algorithms when the corresponding code was released. The reported performance measures are the accuracy and the Roc obtained on the test set, whose values are averaged between all the executions in a single repetition since the classification algorithms are deterministic.

VI. RESULTS AND STATISTICAL ANALYSIS

Tables II, III and IV report the accuracy and Roc test results along with the Wilcoxon signed-ranks statistical tests for classifiers C4.5, 1NN and SVM. Each table represents three cases of results: a) with the original data set (baseline results for the pairwise comparison), b) with OUTLIERoutF and c) with OUTLIERoutP. Since there are ten datasets, the T value at $\alpha = 0.05$ should be less or equal than 8 (critical value) to reject the null hypothesis. The sign (*) means that differences are significant at this confidence level and the sign (º) expresses a lower confidence level (90%).

In relation to C4.5 classifier, there are significant differences for both measures in favour of OUTLIERoutP. For the 1NN classifier, the significant improvements are for Roc via OUTLIERoutP at $\alpha = 0.10$. With regards to SVM classifier, there are not significant differences in any case.

The best performance of OUTLIERoutP, in the mentioned situations, is supported by the fact that at least some instances with outliers should be trained in order to be able to generalize instances containing potential outliers.

From a descriptive point of view and bearing in mind the data sets in outlier level II -the level with the majority of problem instances-, OUTLIERoutP outperforms OUTLIERoutF in 4 or 5 out of 5 data sets, depending on the performance measure, with C4.5 classifier.

VII. CONCLUSIONS

Two statistical outlier detection methods by classes were proposed. Experiments on ten binary and multi-class classification problems shed light on that the outright removal of outliers did not reach significant improvements with no classifier. However, the partial drop of outliers overcame significantly the results with the original data sets for some classifiers (C4.5 and 1NN) according to one or both performance measures. Results showed that SVM is a robust classifier and the outlier effect is not a special problem. In addition, a taxonomy of problems according to the amount of outliers was introduced based on four levels, from I to IV. The classification performance of C4.5 improved in an outstanding way by the partial removal of outliers in the training set.

ACKNOWLEDGMENT

This work has been partially subsidized by TIN2007-68084-C02-02 and TIN2011-28956-C02-02 projects of the Spanish Inter-Ministerial Commission of Science and Technology (MICYT), FEDER funds and P11-TIC-7528 project of the "Junta de Andalucía" (Spain).

REFERENCES

[1] J. Han, M. Kamber, and J. Pei, Data Mining: Concepts and Techniques, 3rd ed., San Francisco: Morgan Kaufmann, 2011.

[2] S. Zhang, C. Zhang, and Q.Yang, "Data preparation for data mining", Applied Artificial Intelligence, vol. 17, no 5-6, 2003, pp. 375-381.

[3] C. C. Aggarwal and P. S. Yu, "Outlier detection for high dimensional data", Proc. of the 2001 ACM SIGMOD international conference on Management of data (ACM Sigmod 2001), ACM, June 2001, pp. 37-46, doi:10.1145/375663.375668.

[4] X. Zhu and X. Wu, "Class noise vs. attribute noise: a quantitative study of their impacts", Artificial Intelligence Review, vol. 22, no 3, November 2004, pp. 177-210, doi:10.1007/s10462-004-0751-8.

[5] S. Chen, W. Wang, and H. van Zuylen, "A comparison of outlier detection algorithms for ITS data", Expert Systems with Applications, vol. 37, no 2, March 2010, pp. 1169-1178, doi:10.1016/j.eswa.2009.06.008

[6] S. J. Klawikowski, C. Zeringue, L. S. Wootton, G. S. Ibbott, and S. Beddar, "Preliminary evaluation of the dosimetric accuracy of the in vivo plastic scintillation detector OARtrac system for prostate cancer treatments", Physics in medicine and biology, vol. 59, no 9, May 2014, pp. N27-N36, doi:10.1088/0031-9155/59/9/N27.

[7] K. Shin, A. Abraham, and S. Y. Han, "Improving kNN text categorization by removing outliers from training set", Proc. of the 7th international conference on Computational Linguistics and Intelligent Text Processing (CICLing'06), Springer Berlin Heidelberg, February 2006, pp. 563-566, doi:10.1007/11671299_58.

[8] G. H. John, "Robust Decision Trees: Removing Outliers from Databases", Proc. of the First International Conference on Knowledge Discovery and Data Mining (KDD 1995), AAAI Press, August 1995, pp. 174-179.

[9] A. Albanese, S. K. Pal, and A. Petrosino, "Rough Sets, Kernel Set, and Spatiotemporal Outlier Detection", IEEE Transactions on Knowledge and Data Engineering, vol. 26, no 1, January 2014, pp. 194-207, doi:10.1109/TKDE.2012.234.

[10] S. Fong, S. Deb, and S. Thampi, "Classifying Sonar Signals Using an Incremental Data Stream Mining Methodology with Conflict Analysis", Advances in Signal Processing and Intelligent Recognition Systems, Springer International Publishing, vol. 264, 2014, pp. 171-182, doi:10.1007/978-3-319-04960-1_15.

[11] M. R. Smith and T. Martinez, "Improving classification accuracy by identifying and removing instances that should be misclassified. Proc. of the 2011 International Joint Conference on Neural Networks (IJCNN 2011), IEEE, July 2011, pp. 2690-2697, doi:10.1109/IJCNN.2011.6033571.

[12] J. Laurikkala, M. Juhola, and E. Kentala, "Informal identification of outliers in medical data", Proc. of the Fifth International Workshop on Intelligent Data Analysis in Medicine and Pharmacology (IDAMAP 2000), at the 14th European Conference on Artificial Intelligence (ECAI 2000), August 2000, pp. 20-24.

[13] K. Bache and M. Lichman, "UCI machine learning repository. Irvine, CA: University of California", School of Information and Computer Science, 2013.

[14] J. R. Quinlan, "C4.5: programs for machine learning", San Francisco: Morgan Kaufmann, 1993.

[15] T. Cover and P. Hart, "Nearest neighbor pattern classification", IEEE Transactions on Information Theory, vol. 13, no 1, January 1967, pp. 21-27, doi:10.1109/TIT.1967.1053964.

[16] V. Vapnik, "The nature of statistical learning theory", New York: Springer, 2000.

[17] R. R. Bouckaert, E. Frank, M. A. Hall, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten, "WEKA---Experiences with a Java Open-Source Project", The Journal of Machine Learning Research, vol. 11, March 2010, pp. 2533-2541.

TABLE II. C4.5 CLASSIFIER: TEST RESULTS AND STATISTICAL TESTS

Data set	**Accuracy**						
	Original	***OUTLIERoutF***			***OUTLIERoutP***		
	Average	***Average***	***Difference***	***Ranking***	***Average***	***Difference***	***Ranking***
CTG2	87.40	88.81	1.41	7	89.49	2.09	9
CTG3	83.54	83.87	0.33	2	84.67	1.13	8
german	68.50	72.00	3.50	10	70.95	2.45	10
liver	65.80	66.66	0.86	5	66.38	0.58	7
magic	85.12	84.70	-0.42	3	85.22	0.10	3
olitos	65.00	64.17	-0.83	4	65.42	0.42	5
pima	74.86	73.43	-1.43	8	74.86	0.00	1.5
sleep	73.33	73.23	-0.10	1	73.33	0.00	1.5
tokyo	89.42	88.29	-2.61	9	90.62	-0.31	4
water	83.50	82.35	-1.15	6	84.07	0.57	6
			T=min{39,16}=16			T=min{49.5,5.5}=5.5 **(*)**	
	Roc						
	Original	***OUTLIERoutF***			***OUTLIERoutP***		
	Average	***Average***	***Difference***	***Ranking***	***Average***	***Difference***	***Ranking***
CTG2	0.8115	0.8508	0.0393	9	0.8276	0.0161	7
CTG3	0.8353	0.8543	0.0190	7	0.8566	0.0213	9
german	0.6010	0.6463	0.0453	10	0.6170	0.0160	6
liver	0.6490	0.6685	0.0195	8	0.6643	0.0153	5
magic	0.8678	0.8648	-0.0030	4	0.8691	0.0013	2
olitos	0.7580	0.7425	-0.0155	6	0.7480	-0.0100	4
pima	0.7385	0.7443	0.0058	5	0.7553	0.0168	8
sleep	0.8298	0.8283	-0.0015	2	0.8288	-0.0010	1
tokyo	0.8873	0.8480	-0.0002	1	0.9070	-0.0070	3
water	0.7618	0.7590	-0.0028	3	0.8035	0.0417	10
			T=min{24,31}=24			T=min{47,8}=8 **(*)**	

TABLE III. 1NN CLASSIFIER: TEST RESULTS AND STATISTICAL TESTS

Data set	Accuracy						
	Original	*OUTLIERoutF*			*OUTLIERoutP*		
	Average	*Average*	*Difference*	*Ranking*	*Average*	*Difference*	*Ranking*
CTG2	88.76	88.48	-0.28	5	88.81	0.05	1
CTG3	84.85	84.71	-0.14	3	84.95	0.10	2.5
german	70.90	70.70	-0.20	4	70.80	-0.10	2.5
liver	64.35	64.92	0.57	8	63.77	-0.58	9
magic	80.69	81.05	0.36	6	80.84	0.15	4
olitos	77.50	80.83	3.33	10	80.00	2.50	10
pima	70.70	70.83	0.13	2	70.96	0.26	7
sleep	69.77	70.19	0.42	7	69.99	0.22	6
tokyo	91.14	91.03	-0.11	1	91.35	0.21	5
water	83.11	84.83	1.72	9	83.68	0.57	8
			T=min{42,13}=13			T=min{43,12}=12	
	Roc						
	Original	*OUTLIERoutF*			*OUTLIERoutP*		
	Average	*Average*	*Difference*	*Ranking*	*Average*	*Difference*	*Ranking*
CTG2	0.8130	0.8103	-0.0027	4	0.8138	0.0008	3
CTG3	0.8005	0.7963	-0.0042	5	0.8015	0.0010	4
german	0.6435	0.6423	-0.0012	2	0.6429	-0.0006	1.5
liver	0.6235	0.6308	0.0073	7	0.6180	-0.0055	7
magic	0.7780	0.7825	0.0045	6	0.7798	0.0018	5
olitos	0.8385	0.8600	0.0215	10	0.8553	0.0168	10
pima	0.6763	0.6778	0.0015	3	0.6821	0.0058	8
sleep	0.7893	0.7903	0.0010	1	0.7899	0.0006	1.5
tokyo	0.9015	0.9145	0.0130	8	0.9084	0.0069	9
water	0.7340	0.7530	0.0190	9	0.7390	0.0050	6
			T=min{44,11}=11			T=min{46.5,8.5}=8.5 **(º)**	

TABLE IV. SVM CLASSIFIER: TEST RESULTS AND STATISTICAL TESTS

Data set	Accuracy						
	Original	*OUTLIERoutF*			*OUTLIERoutP*		
	Average	*Average*	*Difference*	*Ranking*	*Average*	*Difference*	*Ranking*
CTG2	92.53	92.48	-0.05	3	92.81	0.28	6.5
CTG3	83.40	83.58	0.18	4	83.35	-0.05	4
german	74.60	75.40	0.80	6	75.05	0.45	9
liver	58.26	59.70	1.44	8	58.26	0.00	1
magic	79.13	79.11	-0.02	1	79.14	0.01	2
olitos	87.50	85.00	-2.50	10	87.92	0.42	8
pima	76.56	77.21	0.65	5	76.36	-0.20	5
sleep	73.08	73.05	-0.03	2	73.05	-0.03	3
tokyo	91.87	90.20	-1.67	9	92.34	0.47	10
water	85.99	86.94	0.95	7	86.27	0.28	6.5
			T=min{40,15}=15			T=min{42.5,12.5}=12.5	
	Roc						
	Original	*OUTLIERoutF*			*OUTLIERoutP*		
	Average	*Average*	*Difference*	*Ranking*	*Average*	*Difference*	*Ranking*
CTG2	0.8998	0.8948	-0.0050	3	0.9024	0.0026	6.5
CTG3	0.8298	0.8390	0.0092	4	0.8288	-0.0010	4
german	0.6615	0.6755	0.0140	9	0.6680	0.0065	9
liver	0.5043	0.5243	0.0200	10	0.5044	0.0001	1
magic	0.7473	0.7473	0.0000	1	0.7475	0.0002	2
olitos	0.9423	0.9288	-0.0135	8	0.9449	0.0026	6.5
pima	0.7098	0.7198	0.0100	6	0.7091	-0.0007	3
sleep	0.8265	0.8220	-0.0045	2	0.8216	-0.0049	8
tokyo	0.9023	0.9118	0.0095	5	0.9139	0.0116	10
water	0.7805	0.7925	0.0120	7	0.7826	0.0021	5
			T=min{41.5,13.5}=13.5			T=min{40,15}=15	