

Exploring the Relationships between Data Complexity and Classification Diversity in Ensembles

Nathan Formentin Garcia¹^a, Frederico Tiggeman¹^b, Eduardo N. Borges¹^c, Giancarlo Lucca¹^d,
Helida Santos¹^e and Graçaliz Dimuro^{1,2}^f

¹*Centro de Ciências Computacionais, Universidade Federal do Rio Grande, Av. Itália, km 8, 96203-900, Rio Grande, Brazil*

²*Departamento de Estadística, Informática y Matemáticas, Universidad Publica de Navarra, Pamplona, Spain*

Keywords: Machine Learning Ensembles, Complexity Measures, Diversity Measures.

Abstract: Several classification techniques have been proposed in the last years. Each approach is best suited for a particular classification problem, i.e., a classification algorithm may not effectively or efficiently recognize some patterns in complex data. Selecting the best-tuned solution may be prohibitive. Methods for combining classifiers have also been proposed aiming at improving the generalization ability and classification results. In this paper, we analyze geometrical features of the data class distribution and the diversity of the base classifiers to understand better the performance of an ensemble approach based on stacking. The experimental evaluation was conducted using 32 real datasets, twelve data complexity measures, five diversity measures, and five heterogeneous classification algorithms. The results show that stacked generalization outperforms the best individual base classifier when there is a combination of complex and imbalanced data with diverse predictions among weak learners.

1 INTRODUCTION

Machine learning (ML) is a field of study in the artificial intelligence area. Among others, ML is used to deal with classification problems. Under a supervised point of view, such problem consists of finding a model or a function that can identify patterns and describe different data classes. The goal of the classification is to label new examples by applying the learned model or function. This model is based on a set of features extracted from the available data.

There are several techniques proposed in the literature to tackle this problem. Support Vector Machines (SVM) (Steinwart and Christmann, 2008), Decision Trees (DT) (Quinlan, 1986), Artificial Neural Networks (ANN) (Haykin, 2007) and Fuzzy Rule-Based Classification Systems (FRBCS) (Ishibuchi et al., 2005) are examples of well known classifiers.

In recent years, some strategies have been pro-

posed to relate classification algorithms' performance with structural and geometric properties of the data. (Michie et al., 1994) proposed a method based on statistical data measures to predict the applicability of a classifier. These domains codify the characteristics of the problems that are suitable or not for the classifier.

Techniques that combine multiple classification algorithms, known as ensemble methods (Opitz and Maclin, 1999) have also been proposed. The objective was to improve the classification results, since it takes advantage of several classification schemes. Classifiers that implement different algorithms potentially provide additional information on the patterns to be classified. Based on this hypothesis, an ensemble approach called stacked generalization was proposed (Wolpert, 1992). This method consists of training a meta-classifier (strong learner) with the outputs of several diverse base classifiers (weak learners) (Ting and Witten, 1999; Dzeroski and Zenko, 2004). Each base classifier is trained from the same dataset, but using different algorithms.

Diversity can be defined as how much classifiers disagree when predicting class labels (Wang and Yao, 2009). Stacking models benefit greatly when there is diversity among the weak learners since others can balance a pattern wrongly detected by an algorithm.

^a <https://orcid.org/0000-0002-3149-9903>

^b <https://orcid.org/0000-0000-0000-0000>

^c <https://orcid.org/0000-0003-1595-7676>

^d <https://orcid.org/0000-0002-3776-0260>

^e <https://orcid.org/0000-0003-2994-2862>

^f <https://orcid.org/0000-0001-6986-9888>

For both majority voting ensembles and stacking, diversity as an indicator of an ensemble’s accuracy is inappropriate (Kuncheva and Whitaker, 2003) (Lanes et al., 2017a).

In this paper, we are interested in investigating the following research questions: what is the effect of data complexity on the ensemble’s quality? Can the relationships between data complexity and classifiers diversity influence the performance of the ensemble? Therefore, we aim to explore the relations between data complexity measures and diversity measures to understand better the performance of an ensemble approach based on stacking. Our experiments were performed over 32 datasets using 12 different DCMs and five distinct diversity measures. Moreover, we have constructed an ensemble composed of five individual heterogeneous classifiers.

2 PRELIMINARIES

In this section, we recall some concepts that are relevant to this paper. Precisely, we start presenting the data complexity measures that are used. After that, we define some diversity measures and finally the ensemble approach based on stacking.

2.1 Data Complexity Measures

(Ho and Basu, 2002) introduced the concept of Data Complexity Measures (DCMs) used to analyze the characteristics of the dataset, which are crucial in the classification accuracy. The authors have organized these measures into three sets according to their characteristics. In what follows, we present the used DCMs of the study.

2.1.1 Measures of Overlap of Individual Feature Values

This type of measure commonly analyzes the overlap regions considering a binary classification problem.

Volume of Overlap Region - F2. For each feature, the maximum and the minimum values for each class are found. $F2$ performs the ratio between the overlap regions and the range of values spanned by both classes.

Feature Efficiency - F3. This DCM was designed for high dimensional data because it describes how much each feature contributes to the classes’ separation. The efficiency of each feature is defined as the fraction of points possible to be used, i.e., without overlapping values for the analyzed

Table 1: The relationship matrix between the classifiers C_a and C_b , where each value $n_{ij}|i = 0 \vee i = 1, j = 0 \vee j = 1$ is the number of instances labeled correctly (1) or not (0).

	C_b correct	C_b incorrect
C_a correct	n_{11}	n_{10}
C_a incorrect	n_{01}	n_{00}
$N = n_{11} + n_{00} + n_{01} + n_{10}$		

classes. $F3$ returns the maximum efficiency for all features.

2.1.2 Measures of Separability of Classes

Linear Separability - L1, L2. These measures are also applied considering binary classification problems. They verify whether the dataset is linearly separable, fitting an objective function, where the returned value is used as $L1$. If $L1 = 0$, the problem is linearly separable. Also, $L2$ is the error rate of the linear classifier. This measure is strongly affected by outliers or overlapping points.

Mixture Identifiability - N1, N2, N3. In order to perform these measures, a minimum spanning tree is built connecting all data points. Using the Euclidean distance between each point and the nearest neighbor within or outside the class, $N2$ is defined as the ratio between the average distances to intraclass and interclass nearest neighbors.

2.2 Diversity Measures

Several diversity measures were proposed by (Kuncheva and Whitaker, 2003). These measures are not about how structurally different the classifiers are, but how much they disagree when predicting a class label. Considering two classifiers C_a and C_b predicting N instances, there are four possible scenarios for each instance: both predict correctly, both predict wrongly, C_a predicts correctly and C_b wrongly, and C_b predicts correctly and C_a wrongly. Table 1 provides a visual explanation of possible scenarios.

In the next subsections, we explain every measure used in this paper. Most of them are adaptations from other fields to machine learning classifier algorithms. We use the same notation of Table 1. Besides, L refers to the number of weak learners.

2.2.1 Q Statistics

This measure is inversely proportional to the diversity between classifiers. The multi-class oriented version

of Q Statistics is represented by Eq. (1). These equations return a value in the range $[-1, 1]$.

$$\bar{Q} = \frac{2}{L(L-1)} \sum_{i=1}^{L-1} \sum_{k=i+1}^L Q_{ik} \quad (1)$$

2.2.2 Disagreement

This measure proposed by (Skalak et al., 1996) is directly proportional to the diversity between the classifiers and varies in the range $[0, 1]$. Eq. (2) defines the average disagreement between all weak learners.

$$\bar{Dis} = \frac{2}{L(L-1)} \sum_{i=1}^{L-1} \sum_{k=i+1}^L Dis_{ik} \quad (2)$$

2.2.3 Kohavi-Wolpert Variance

This non-pairwise diversity measure was proposed by (Bauer and Kohavi, 1999) and is directly proportional to the diversity among L base classifiers. It is very similar to the average disagreement measure, differing by a coefficient. Its possible value range is $[0, 1/2]$.

$$KW = \left(\frac{L-1}{2L} \right) \bar{Dis} \quad (3)$$

2.3 Stacked Generalization

Stacked generalization consists in using the prediction of multiple classifiers as training set for the meta-classifier (Merz, 1999; Kotsiantis and Pintelas, 2004) with the objective of achieving a greater accuracy. This technique can be considered an excellent bias filter when there are disagreement between the base classifiers. As pointed out by (Breiman, 1996), stacking can have better accuracy through linear combinations of different predictors. Each classifier performs differently according to the dataset. Figure 1 is presented to provide a visual scheme that is easier to understand.

3 RELATED WORK

(Lucca et al., 2018) used data complexity measures to analyze the behavior of different aggregations and pre-aggregation functions when used to tackle classification problems with Fuzzy Rule-Based Classification Systems. The authors found that there is a directed relation between some measures and performance. However, in this study, only binary datasets were considered.

Stacking has been applied successfully in several different fields. To name a few, (Lee, 2017)

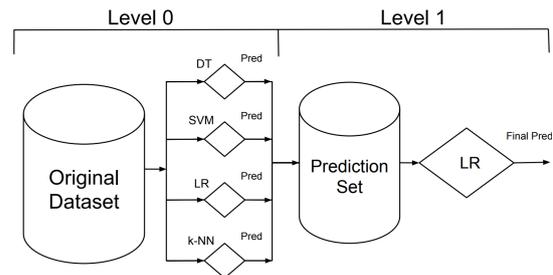


Figure 1: Stacking method exemplified. The first level (0) consists in training the weak learners with the original data. In the next level, we use the generated dataset (first level predictions) in order to train the meta-classifier. Then, the classification schema is ready to be tested and used. The used algorithms are presented in the ensemble construction subsection.

uses stacking to predict depression among the elderly. (Álvarez et al., 2016) uses stacked generalization to better recognize emotion in speech and (Li and Zou, 2017) to detect the subject's native language.

The connection between the diversity among weak learners and the stacking accuracy was investigated by (Lanes et al., 2017a; Lanes et al., 2017b). The authors measured diversity in several datasets, both real and synthetic ones. The results show that 3 out of 7 evaluated measures are related to the final classification accuracy. Although there are proven connections between diversity and accuracy in some particular cases (Shipp and Kuncheva, 2002; Dymitr and Bogdan, 2005; Kuncheva and Whitaker, 2003), the results raised some doubts about the usefulness of diversity measures in building sets of classifiers in real-life pattern recognition problems. However, other authors report success in using diversity to detect noise to generate more accurate classification systems (Muhammad and Jim, 2010; Makhtar et al., 2012; Whalen and Pandey, 2013; Faria et al., 2014).

4 METHODOLOGY

In this section, we present the methodology adopted in this study. We start by describing the datasets that are used. After that, we show how the DCMs are applied since some were proposed for binary classification problems. Then, we discuss the ensemble construction with the considered classifiers. Finally, the statistical test is presented as well as the configuration of the proposal.

4.1 Datasets

In order to provide a robust result, in this study we apply the DCMs and diversity measures over 32 het-

Table 2: Information about the datasets used in the experiment. The datasets are alphabetically ordered.

Id	Dataset	#I	#F	#C	Distribution
App	Appendicitis	106	7	2	0.8:0.2
Bal	Balance	625	4	3	0.46:0.46:0.8
Ban	Banana	5300	2	2	0.55:0.45
Bnd	Bands	365	19	2	0.63:0.37
Bup	Bupa	345	6	2	0.58:0.42
Clv	Cleveland	297	13	5	0.53:0.18:0.11:0.11:0.04
Con	Contraceptive	1473	9	3	0.42:0.34:0.22
Gla	Glass	214	9	6	0.35:0.32:0.13:0.07:0.06:0.04
Hab	Haberman	306	3	2	0.73:0.27
Hay	Hayes-Roth	160	4	3	0.4:0.4:0.2
Ion	Ionosphere	351	33	2	0.64:0.36
Iri	Iris	150	4	3	0.33:0.33:0.33
Led	Led7Digit	500	7	10	0.114:0.114:0.106:0.104:0.104: 0.102:0.098:0.094:0.090:0.074
Mag	Magic	19020	10	2	0.65:0.35
New	Newthyroid	215	5	3	0.7:0.16:0.14
Pag	Page-blocks	5472	10	5	0.9:0.06:0.02:0.015:0.005
Pen	Penbased	10992	16	10	0.104:0.104:0.103:0.103:0.103: 0.096:0.096:0.096:0.096:0.096
Pim	Pima	768	8	2	0.65:0.35
Pho	Phoneme	5404	5	2	0.7:0.3
Rin	Ring	7400	20	2	0.5:0.5
Sah	Saheart	4625	9	2	0.65:0.35
Sat	Satimage	6435	36	6	0.24:0.23:0.21:0.11:0.11:0.1
Seg	Segment	2310	19	7	0.143 for each
Shu	Shuttle	57999	9	5	0.78:0.15:0.05:0.002:0.0008: 0.002:0.000
Son	Sonar	208	60	2	0.53:0.47
Spe	Spectfheart	266	44	2	0.79:0.21
Tit	Titanic	2201	3	2	0.67:0.33
Two	Twonorm	7400	20	2	0.5:0.5
Veh	Vehicle	846	18	4	0.26:0.26:0.25:0.23
Win	Wine	178	13	3	0.4:0.33:0.27
Wis	Wisconsin	683	9	2	0.65:0.35
Yea	Yeast	1484	8	9	0.32:0.29:0.16:0.11:0.035: 0.03:0.023:0.01:0.0034

erogeneous datasets. They vary from 106 to 19,020 instances, from 2 to 60 features, and from 2 to 10 classes. These datasets are available in the UCI Machine Learning (Dua and Graff, 2017) and KEEL-dataset (Alcalá-Fdez et al., 2011) repositories.

Table 2 summarizes the characteristics of each dataset. We present an identifier (Id), the complete name (Dataset), the number of instances (#I), the number of features (#F), the number of classes (#C), and the class distribution (Distribution). We also highlight that, in this study, 16 datasets are multi-class and 16 are binary. For example, App dataset contains 106 instances of 7 features, distributed into two classes, where 80% are labeled c_1 and 20% c_2 .

4.2 Data Complexity Measures

For each dataset, we have computed all DCMs presented in Section 2.1. To calculate the average of the binary DCMs $F2$, $L1$ and $L2$, we use two different strategies: One vs. One (OvO) combines them for all pairs of data classes, while One vs. All (OvA) merges one measure per class, trained to distinguish the samples in a single class from the samples in all remaining classes. We have ignored invalid values returned by some measures (-1 , Infinity or $0/0$) where

Algorithm 1: *Stacking* algorithm used in this experiment.

```

Input : Original training samples  $s^j \in S$ 
Output: final predictions  $y_f^j$ 

1 begin
2   Select L weak learners ( $L_1, L_2, L_3 \dots L_L$ )
3   for  $i=1$  to L do
4     Train classifier  $C_i$  using  $L_i$  on
       cross-validated  $S$ ;
5      $y_i^j = C_i(S)$ ;
6      $p_i^j = pC_i(S)$ ;
7   end
8   Train strong learner estimator  $M$  using
       CV  $y^j$ ;
9   Evaluate the model  $eval1 = acc(M, y^j)$ 
10  Train strong learner estimator  $M$  using
       CV  $p^j$ ;
11  Evaluate the model  $eval2 = acc(M, p^j)$ 
12  if  $eval1 \geq eval2$  then
13     $y_f^j = M(y^j)$ 
14  else
15     $y_f^j = M(p^j)$ 
16  end
17 end
18 return  $y_f^j$ 

```

there was no variance between pairs of features. So, for these DCMs, we have reported the mean and the standard deviation in the experiments.

4.3 The Ensemble Construction

In this subsection, we describe the ensemble construction process, which is defined by Algorithm 1.

After selecting a set of learning algorithms (line 2), the weak learners will be trained over a considered dataset S (lines 3–7) using Cross-Validation (CV). In this step, for each classifier i and sample j , we generate two new datasets: one containing the classifiers' predictions (y_i^j) and another having their prediction's probability distributions (p_i^j). After that, having these datasets, the meta-classifier estimator M is trained and evaluated using accuracy (lines 8–11). The predictions of the best model, i.e. with higher accuracy, are returned by the Stacking algorithm (lines 12–18). The best weak learner accuracy and the strong learner accuracy are compared later, applying a statistical test explained in 4.4.

To improve the method's quality, we pick linear and non-linear classifiers. We present the considered classifiers discussing their main characteristics:

Naïve Bayes (NB). Proposed by (Zhang, 2005), The Naïve Bayes algorithm considers that the probability distribution is Gaussian. Based in Bayes Theorem (Bolstad and Curran, 2016), NB assumes features independence. This classifier fits a probabilistic model.

Decision Tree (DT). Decision Trees classifiers are one of the most popular algorithms in the field, mainly for its simplicity and explainability (Loh, 2011). A decision tree is composed of nodes (that represents features) and links that represent a decision. The decision aims to reduce the highest impurity possible. Impurity can be computed using indices based on Gini or Entropy.

Logistic Regression (LR). Proposed by (Nelder and Wedderburn, 1972), Logistic Regression is a powerful classifier that is mainly used to predict binary classes. It describes the relationship between a dependent variable and independent variables. The OvA approach is used to deal with multi-class problems. All data are regularized by default.

Support Vector Machines (SVM). The SVM analyzes the point distribution in space and tries to separate them in categories divided by the widest gap possible, as pointed by (Cortes and Vapnik, 1995). It is very versatile since it can be a linear and non-linear classifier, depending on the selected kernel.

k-Nearest Neighbours (K-NN). k-NN is a non-parametric algorithm proposed by (Altman, 1992) that can be used for classification and regression problems. This algorithm estimates how likely a sample is to be part of one group or another. It measures the distance between the sample and the neighbors and labels it to the nearest neighbor.

All diversity measures defined in 2.2 are computed. The classifier picked as the strong learner was the Logistic Regression because this is a well-known classifier that offers good performance for different data types.

To obtain the accuracy score, we consider a cross-validation method. For each partition, one will be selected at random to be the validation set. The remaining parts will be used as a training set. It is important to note that the folds were stratified since it reduces bias and variance compared to non-stratified models as pointed by (Kohavi, 1995).

4.4 Statistical Study

In order to give statistical support of the obtained results, we used a paired Student's t -test on the performance results. We guarantee that a sample only ap-

pears as part of the training or validation set with 2-fold cross-validation for a single performance estimation. As suggested by (Dietterich, 1998), this statistical test is recommended for situations in which time or computational resources are not a problem, mainly because it needs several performance evaluations to obtain the p -value.

The 5x2 cross-validated t -test was proposed by (Dietterich, 1998) to compare machine learning models. Considering two classifiers, C_a and C_b , we repeat five times a two-fold cross-validation. Both classifiers are then fitted and validated using these generated sets. Equation (4) shows how the t statistic is computed, where $p_1^{(1)}$ is the performance difference in the current iteration and s_i^2 is the variance of performance differences.

$$t = \frac{p_1^{(1)}}{\sqrt{1/5 \sum_{i=1}^5 s_i^2}} \quad (4)$$

Considering a t distribution with five degrees of freedom, we obtain the p -value. The null hypothesis means the models have equal performance. When the p -value is smaller than the significance level, we can say that there is a significant difference between C_a and C_b .

5 RESULTS AND DISCUSSION

In this section the obtained results are summarized and discussed. Table 3 summarizes the results for each dataset presented in the first column (Id). The second to the fourth columns are related to the diversity measures \bar{Q} , \overline{Dis} and KW . The fifth to eighth columns are related to the performance of the classifiers, where LO_{best} is the learning algorithm of the best base classifier, $LO_{best}Acc$ is the best base classifier accuracy, and $L1_{best}Acc$ is the accuracy reached by the ensemble. The values between parenthesis in these columns are the standard deviation. The p -value obtained by the statistical test is also presented. The remaining columns are related to the different DCMs using One vs. One (OvO) and One vs. All (OvA) approaches.

It is important to note that some data complexity and diversity measures were not presented, mainly because we could not see a relation between them and the ensemble's performance. The omitted DCMs are $\bar{F1}$, $\bar{L2}$, $\bar{L3}$, $N1$, $N3$, $N4$, $T1$, and $T2$. The diversity measures is double-fault and entropy. However, full reports are available in the provided results. The Iris dataset does not contain information about diversity measures because the weak learner's predictions were

Table 3: Obtained results considering the ensemble performance and different diversity and data complexity measures.

Id	Diversity Measure			Performance metrics				Data Complexity Measures						
	\bar{Q}	\bar{Dis}	KW	$L0_{best}$	$L0_{best}Acc$	$L1_{best}Acc$	p	$\bar{L1}_{OvO}$	$\bar{F2}_{OvO}$	$\bar{F3}_{OvO}$	$\bar{L1}_{OvA}$	$\bar{F2}_{OvA}$	$\bar{F3}_{OvA}$	N2
App	0.4672	0.0454	0.0113	k-NN	0.8660 (0.0362)	0.8500 (0.0000)	0.5623	0.4182	0.0446	0.2925	0.5266	0.0154	0.5733	0.2120
Bal	0.4491	0.0368	0.0122	SVM	0.9011 (0.0057)	0.8996 (0.0136)	0.6507	0.3687	1	0	1.4386	0.2169	0.1215	0.3318
Ban	0.2874	0.1311	0.0327	SVM	0.9023 (0.0027)	0.9017 (0.0045)	0.7151	0.8966	0.6257	0.0042	0.4483	1	0	0.0278
Bnd	0.0330	0.2739	0.0684	LR	0.6684 (0.0316)	0.6281 (0.0256)	0.8850	0.7855	0	0.0493	0.4182	0.0446	0.2925	0.6388
Bup	0.2591	0.1652	0.0413	LR	0.6707 (0.0235)	0.6152 (0.0246)	0.2681	1.0068	0.0732	0.0319	0.4855	1	0	0.6064
Clv	0.3534	0.0950	0.0380	LR	0.5818 (0.0479)	0.5713 (0.0363)	0.2216	0.5757	0.1602	0.0539	0.4523	0	0.1909	0.6308
Con	0.1735	0.1630	0.0540	LR	0.5116 (0.0125)	<u>0.5243</u> (0.0129)	0.0390	0.9025	0.7659	0.0679	0.4321	1	0	0.6770
Gla	0.0040	0.2069	0.0862	k-NN	0.6271 (0.0500)	0.6175 (0.0339)	0.8109	0.3304	0.0416	0.2897	53.2287	0	0.0185	0.2603
Hab	0.4027	0.0564	0.0141	NB	0.7464 (0.0281)	0.7057 (0.0209)	0.6383	0.4705	0.0017	0.0294	0.8313	0.0041	0.0097	0.7819
Hay	0.3746	0.0843	0.0281	DT	0.8012 (0.0452)	0.7766 (0.0504)	0.2414	0.5533	0.7177	0.0813	0.7174	0.0815	0.0059	0.6497
Ion	0.4494	0.0436	0.0109	SVM	0.9373 (0.0136)	0.9464 (0.0144)	0.1606	0.6645	0.5309	0.1909	0.5502	0.0004	0.8837	0.3244
Iri	-	0	0	k-NN	0.9653 (0.0199)	0.9467 (0.0163)	0.3577	0.4523	0	0.5733	0.5533	0.7177	0.0294	0.7948
Led	0.4598	0.0790	0.0355	LR	0.7360 (0.0179)	0.7030 (0.0145)	0.4819	0.2917	0.0179	0	0.6331	0	0.9944	0.1814
Mag	0.4121	0.0813	0.0203	SVM	0.8205 (0.0024)	<u>0.8385</u> (0.0042)	0.0111	0.2089	1	0.0059	0.6887	0.2516	0.0065	0.8397
New	0.4079	0.0348	0.0116	NB	0.9702 (0.0108)	0.9744 (0.0136)	0.4225	0.7174	0.0815	0.8837	8.5841	0.2105	0.3930	0.3239
Pag	0.4430	0.0421	0.0168	DT	0.9625 (0.0037)	0.9625 (0.0030)	0.6131	0.7013	0.0005	0.0185	0.6672	0.2708	0.1223	0.2693
Pen	0.3970	0.0543	0.0244	SVM	0.9922 (0.0011)	0.9921 (0.0010)	0.5367	60.5615	0.0004	0.3930	2.4677	0	0.5674	0.5343
Pho	0.3980	0.0760	0.0190	k-NN	0.8615 (0.0031)	<u>0.8704</u> (0.0072)	0.0213	4.2620	0.0458	0.1223	0.7515	0.2963	0.0813	0.4857
Pim	0.3980	0.0779	0.0194	LR	0.7625 (0.0126)	0.7616 (0.0207)	0.0593	0.6672	0.2708	0.0065	0.2932	0.1170	0.0539	0.9117
Rin	0.3297	0.0639	0.0159	NB	0.9796 (0.0016)	0.9802 (0.0015)	0.1132	0.6887	0.2516	0.0538	0.9081	0.3590	0.0498	0.8125
Sah	0.3316	0.1225	0.0306	LR	0.7060 (0.0164)	0.7198 (0.0167)	0.9113	0.7206	0	0.0498	0.7206	0	0.0538	0.8657
Sat	0.3813	0.0766	0.0319	k-NN	0.8979 (0.0028)	<u>0.9064</u> (0.0019)	0.0272	0.9081	0.3590	0.5674	0.8966	0.6257	0.0042	0.1513
Seg	0.4017	0.0658	0.0282	DT	0.9506 (0.0073)	<u>0.9643</u> (0.0082)	0.0101	6.2676	0	0.9944	1.0068	0.0732	0.0319	0.9271
Shut	0.0598	0.0749	0.0321	DT	0.9995 (0.0001)	0.9995 (0.0002)	1	0.3740	0	0.9990	0.6469	0	0.0481	0.7413
Son	0.2590	0.1214	0.0303	SVM	0.7375 (0.0480)	0.7723 (0.0512)	0.6063	1.8729	0	0.0481	0.4029	0.0005	0.7640	0.5747
Spe	0.2119	0.1685	0.0421	SVM	0.7939 (0.0186)	0.7953 (0.0254)	0.5443	0.6469	0	0.2959	3.6855	0	0.2959	0.8033
Tit	0.4838	0.0297	0.0074	DT	<u>0.7867</u> (0.0092)	0.7827 (0.0071)	0.0254	3.6855	0	0	0.6626	0.7633	0.0679	1.0166
Two	0.4313	0.0213	0.0053	NB	0.9781 (0.0009)	0.9786 (0.0021)	0.6070	0.4483	1	0.0097	0.2148	0.0056	0.0357	0.9515
Veh	0.1286	0.1805	0.0677	LR	0.7047 (0.0214)	0.7343 (0.0165)	0.2415	0.8313	0.0041	0.4574	2.4908	0.0009	0.4574	0.7178
Win	0.2151	0.0694	0.0231	NB	0.9752 (0.0149)	0.9563 (0.0204)	1	2.8113	0.0282	0.7640	4.6044	0.0016	0.9990	0.1287
Wis	0.4858	0.0131	0.0032	SVM	0.9677 (0.0049)	0.9707 (0.0073)	1	0.2915	0	0.1215	0.7855	0	0.0493	0.5275
Yea	0.1351	0.1781	0.0801	SVM	0.5855 (0.0210)	0.3243 (0.0006)	0.9504	0.3837	0.0037	0.0357	0.3414	0.0043	0.2897	0.6895

the same and, hence, there was no diversity between them.

When analyzing the results, it is possible to see a significant difference between the maximum and minimum value of the complexity and diversity measures, making it easier to collectively analyze their behavior. For instance, \bar{Q} Statistics maximum value calculated for our experiment was 0.4858 for the Titanic dataset, which is considered a very low diversity ensemble. For the Glass dataset, this measure was 0.004.

We can also see that our $L0_{best}$ selections are very diverse: k-Nearest Neighbors had greater accuracy in five cases, Support Vector Machines in nine, Logistic Regression in eight, Naive Bayes in five, and Decision Tree in five. This reinforces the idea proposed by Wolpert (Wolpert, 1996), that each classification algorithm has a particular performance depending on data. No particular scenario could be detected to predict which type of classifiers are better for complex or non-complex datasets.

After comparing the best weak learner to the strong learner for each dataset, we could see that the strong learner was better, statistically significant with a 5% confidence interval for five datasets, namely: Contraceptive, Satimage, and Segment (multi-class problems), Magic and Phoneme (binary problems). The ensemble was significantly worse only in the Ti-

tanic dataset. In Table 3, these datasets and p -values are highlighted in **bold** while the best accuracy is underlined. It is observable that there were no statistical differences for the remaining datasets.

When comparing the measures and our performance results, we could detect a relation between the DCMs $N2$, $F2$, $L1$, and the diversity measures \bar{Q} Statistics, Disagreement, and Kohavi-Wolpert Variance. The statistically significant better ensembles have the following characteristics: medium-high diversity ensembles applied in datasets with reasonable complexity.

Contraceptive, in which the ensemble performed statistically better, was measured as an above-average complexity dataset, and the diversity was also significantly above average. Moreover, for the Titanic dataset, the only case in which our ensemble was significantly worse, the complexity and diversity measures were one of the lowest.

It is also important to note that the datasets in which our ensemble performed better are considerably imbalanced or with a great number of class labels. Magic dataset, per example, is a binary problem with the class distribution of 65% for the majority class and 35% for the minority class. The pattern can be observed for other datasets as well: Phoneme, Satimage and Segment datasets present large class

imbalance or a high number of classes, combined with significant diversity and complexity.

6 CONCLUSION

In this study, we investigated the effect of data complexity on the ensemble's accuracy. Also, we analyzed the relationship between data complexity and the diversity among classifiers. We discovered that complexity and diversity measures alone are not good indicators of when to use an ensemble. We believe that there are several important factors other than those. We could see a specific pattern between data complexity, diversity measures and class imbalance, though.

For all the datasets in which the ensemble performed statistically better, there were a high diversity, complexity and class imbalance. In cases that only two of these factors are present, we could not see benefit of using ensembles.

The combination of complex and imbalanced datasets with diverse predictions among weak learners seems to be the cases in which stacking methods thrive. For instance, the Cleveland dataset presented a complexity very similar to the Contraceptive dataset, and both are imbalanced. The difference between them is that the weak learners presented low diversity in its predictions for Cleveland and one of the highest for Contraceptive. The same pattern can be observed in several other datasets, such as Pageblocks and Yeast. Titanic can be an example, as well: a low diversity ensemble applied over a simple dataset resulted in a worse classifier.

We could not justify the use of stacked generalization for simple datasets, in which base classifiers can already have almost perfect performance. If they have an almost perfect accuracy, the base classifiers do not differ and the ensemble does not benefit from the combination. Hence, the ensembles provide the same results or worse. We can also conclude that a higher complexity measure does not necessarily trigger diversity among weak learners.

After conducting this study, we can say that these three measures can help when deciding when to use an ensemble or not. For future studies, we intend to use a greater number of datasets with different settings (i.e. some of them complex and imbalanced, others complex and balanced) to understand the nuances between diversity, complexity and class imbalance. Also, we aim to find other measures that could help to explain when to use ensembles, since we firmly believe that, even though these three measures can be used when deciding to use or not ensembles, there could be some

others in which we could combine with them to better understand this problem's nature and provide a guideline to make the decision.

ACKNOWLEDGMENTS

This study was supported by CAPES Financial Code 001, PNPd/CAPES (464880/2019-00), CNPq (301618/2019-4), and FAPERGS (19/2551-0001279-9, 19/2551-0001660).

REFERENCES

- Alcalá-Fdez, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L., and Herrera, F. (2011). Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic & Soft Computing*, 17.
- Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. *The American Statistician*, 46(3):175–185.
- Álvarez, A., Sierra, B., Arruti, A., López-Gil, J.-M., and Garay-Vitoria, N. (2016). Classifier subset selection for the stacked generalization method applied to emotion recognition in speech. *Sensors*, 16(1):21.
- Bauer, E. and Kohavi, R. (1999). An empirical comparison of voting classification algorithms: Bagging, boosting, and variants. *Machine learning*, 36(1-2):105–139.
- Bolstad, W. M. and Curran, J. M. (2016). *Introduction to Bayesian statistics*. John Wiley & Sons, Hoboken, USA.
- Breiman, L. (1996). Stacked regressions. *Machine learning*, 24(1):49–64.
- Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine learning*, 20(3):273–297.
- Dietterich, T. G. (1998). Approximate statistical tests for comparing supervised classification learning algorithms. *Neural computation*, 10(7):1895–1923.
- Dua, D. and Graff, C. (2017). UCI machine learning repository.
- Dymitr, R. and Bogdan, G. (2005). Classifier selection for majority voting. *Information Fusion*, 6(1):63 – 81.
- Dzeroski, S. and Zenko, B. (2004). Is combining classifiers with stacking better than selecting the best one? *Mach. Learn.*, 54(3):255–273.
- Faria, F. A., dos Santos, J. A., Rocha, A., and da S. Torres, R. (2014). A framework for selection and fusion of pattern classifiers in multimedia recognition. *Pattern Recognition Letters*, 39:52 – 64. *Advances in Pattern Recognition and Computer Vision*.
- Haykin, S. (2007). *Neural Networks: A Comprehensive Foundation*. Prentice-Hall, Inc., Upper Saddle River, USA.

- Ho, T. K. and Basu, M. (2002). Complexity measures of supervised classification problems. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(3):289–300.
- Ishibuchi, H., Nakashima, T., and Nii, M. (2005). *Classification and Modeling with Linguistic Information Granules, Advanced Approaches to Linguistic Data Mining*. Advanced Information Processing. Springer, Berlin.
- Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *Proceedings of the 14th International Joint Conference on Artificial Intelligence - Volume 2, IJCAI'95*, page 1137–1143, San Francisco, CA, USA. Morgan Kaufmann PUBLISHERS Inc.
- Kotsiantis, S. and Pintelas, P. (2004). A hybrid decision support tool. In *Proceedings of 6th International Conference on Enterprise Information Systems*, pages 448–453, Porto, Portugal. Springer.
- Kuncheva, L. I. and Whitaker, C. J. (2003). Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Mach. Learn.*, 51(2):181—207.
- Lanes, M., Borges, E. N., and Galante, R. (2017a). The effects of classifiers diversity on the accuracy of stacking. In *SEKE*, pages 323–328, New York, USA. ACM Press.
- Lanes, M., Schiavo, P. F., Pereira Jr, S. F., Borges, E. N., and Galante, R. (2017b). An analysis of the impact of diversity on stacking supervised classifiers. In *ICEIS (I)*, pages 233–240, Setúbal, Portugal. ScitePress.
- Lee, E. S. (2017). Exploring the performance of stacking classifier to predict depression among the elderly. In *2017 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 13–20, Park City, UT, USA. IEEE.
- Li, W. and Zou, L. (2017). Classifier stacking for native language identification. In *Proceedings of the 12th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 390–397, Park City, Utah, USA. Association for Computational Linguistics.
- Loh, W.-Y. (2011). Classification and regression trees. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 1(1):14–23.
- Lucca, G., Sanz, J., Dimuro, G. P., Bedregal, B., and Bustince, H. (2018). Analyzing the behavior of aggregation and pre-aggregation functions in fuzzy rule-based classification systems with data complexity measures. In Kacprzyk, J., Szmidt, E., Zadrożny, S., Atanassov, K. T., and Krawczak, M., editors, *Advances in Fuzzy Logic and Technology 2017*, pages 443–455, Cham. Springer International Publishing.
- Makhtar, M., Yang, L., Neagu, D., and Ridley, M. (2012). Optimisation of classifier ensemble for predictive toxicology applications. In *14th International Conference on Computer Modelling and Simulation*, pages 236–241, Washington, USA. IEEE Computer Society.
- Merz, C. J. (1999). Using correspondence analysis to combine classifiers. *Machine Learning*, 36(1-2):33–58.
- Michie, D., Spiegelhalter, D. J., Taylor, C. C., and Campbell, J., editors (1994). *Machine Learning, Neural and Statistical Classification*. Ellis Horwood, Upper Saddle River, NJ, USA.
- Muhammad, A. T. and Jim, S. (2010). Creating diverse nearest-neighbour ensembles using simultaneous metaheuristic feature selection. *Pattern Recognition Letters*, 31(11):1470–1480.
- Nelder, J. A. and Wedderburn, R. W. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384.
- Opitz, D. and Maclin, R. (1999). Popular ensemble methods: An empirical study. *Journal of Artificial Intelligence Research*, 11:169–198.
- Quinlan, J. (1986). Induction of decision trees. *Machine Learning*, 1:81–106.
- Shipp, C. A. and Kuncheva, L. I. (2002). Relationships between combination methods and measures of diversity in combining classifiers. *Information Fusion*, 3(2):135 – 148.
- Skalak, D. B. et al. (1996). The sources of increased accuracy for two proposed boosting algorithms. In *Proc. American Association for Artificial Intelligence, AAAI-96, Integrating Multiple Learned Models Workshop*, volume 1129, page 1133, Menlo Park, CA, USA. Citeseer, AAAI Press.
- Steinwart, I. and Christmann, A. (2008). *Support Vector Machines*. Springer Publishing Company, Incorporated, New York, USA, 1st edition.
- Ting, K. M. and Witten, I. H. (1999). Issues in stacked generalization. *Journal of Artificial Intelligence Research*, 10:271–289.
- Wang, S. and Yao, X. (2009). Diversity analysis on imbalanced data sets by using ensemble models. In *2009 IEEE Symposium on Computational Intelligence and Data Mining*, pages 324–331, New York, USA. IEEE, IEEE.
- Whalen, S. and Pandey, G. (2013). A comparative analysis of ensemble classifiers: Case studies in genomics. In *2013 IEEE 13th International Conference on Data Mining*, pages 807–816, Washington, USA. IEEE Computer Society.
- Wolpert, D. H. (1992). Stacked generalization. *Neural networks*, 5(2):241–259.
- Wolpert, D. H. (1996). The lack of a priori distinctions between learning algorithms. *Neural computation*, 8(7):1341–1390.
- Zhang, H. (2005). Exploring conditions for the optimality of naive bayes. *International Journal of Pattern Recognition and Artificial Intelligence*, 19(02):183–198.