An Interpretable Measure of Dataset Complexity for Imbalanced Classification Problems

Jonatan Møller Nuutinen Gøttcke * Colin Bellinger † Paula Branco ‡ Arthur Zimek §

October 2, 2022

Abstract

The class imbalance problem is associated with harmful classification bias and presents itself in a wide variety of important applications of supervised machine learning. Measures have been developed to determine the imbalance complexity of datasets with imbalanced classes. The most common such measure is the Imbalance Ratio (IR). It is, however, widely accepted that the complexity of a classification task is the combined result of class imbalance and other factors, such as class overlap. Thus, in order to accurately assess the complexity of a problem, the data complexity measures ought to account for more than the simple IR. In this paper we demonstrate that IR has a weak correlation with classifier performance in terms of macro averaged recall, gmean score, and precision. Other more complete measures such as the adapted N1 and N3 measures use neighborhood information to assess overlap. These measures show a strong negative correlation with classifier performance, but their reported values are hard to interpret. This motivates a new measure which estimates overlap complexity and returns a value with a clear interpretation. Here we propose such a measure based on the number of minority instances entangled in a Tomek Link. The proposed measure is evaluated on a large selection of synthetic and real datasets and is found to be as good as or better than the best competitors in terms of its negative correlation with respect to mean classifier performance.

1 Introduction

In the field of supervised learning one of the most persistent issues is imbalanced data distributions. This is commonly referred to as the class imbalance problem. Imbalanced data distributions often lead to a bias towards predicting the majority class or classes, which in most cases is very undesired. Describing a datasets complexity accurately, in terms of some classification problem gives us some interesting possibilities. Before building a model on a dataset, it is important to have knowledge about the dataset, to know how to approach it, and which potential pitfalls are present. Therefore we must have a way of describing the severity of mechanisms which make classification tasks complex. When a new method is proposed, which claims to handle such a complexity, this measure can be used to determine which datasets are relevant in the benchmark. In everyday classification problems it is also relevant to have a tool which can assist us in checking the type of complexity that is hindering classifier performance. In many papers on the imbalance problem, where a larger experimental framework is made the method of determining whether class imbalance is present is done by reporting the Imbalance Ratio (IR). It can therefore be seen as the de facto standard when it comes to measuring the severity of the class imbalance problem. The Imbalance Ratio is defined in Equation 1.1 and gives an easy understanding of the level of imbalance. It is the ratio between the number of points in the largest majority class and the smallest minority class. $\mathcal{D}$ is the domain containing all instances within the dataset, and C_{maj} and C_{min} is respectively the majority and minority class subsets of the domain.

$$\text{IR}(\mathcal{D}) = \frac{|C_{maj}|}{|C_{min}|} \tag{1.1}$$

Several works have explored statistical measures to find an alternative to the class imbalance ratio such as the Imbalance Degree [7], or LRID [12], which both do a deeper analysis of the class distributions. A measure called the Adjusted Imbalance Ratio [11] scale the imbalance ratio given the number of discriminative features. Common for these measures is that they do not use any information about the overlap between data from imbalanced class distributions, and even in the most highly imbalanced case two classes can be separated by a straight line, if there is no overlap. In Luengo et al. [5] they study how 12 different dataset complexity measures are affected by class imbalance preprocessing. They conclude that IR is insufficient to predict if one of the two classifiers is going to perform

Goettcke: Insert citations

Bellinger: Consider linking the need for good imbalance complexity metrics to meta-learning and auto-ml

Bellinger: TODO There are papers that theoretically and empirically compare the

*University of Southern Denmark
†National Research Council of Canada
‡University of Ottawa
§University of Southern Denmark

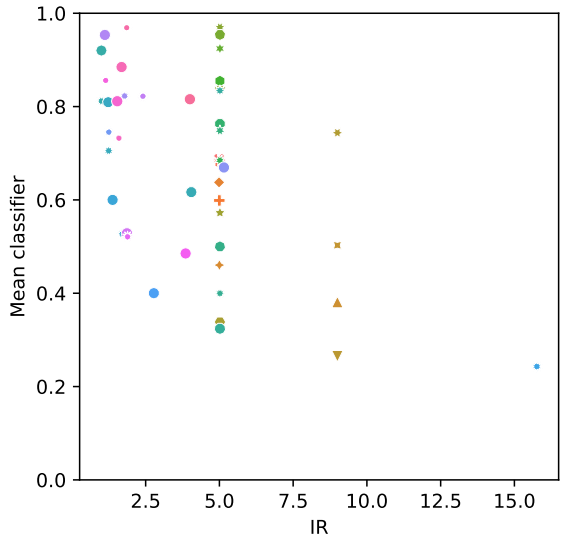


Figure 1: The correlation between the imbalance ratio and a mean classifiers performance in terms of macro averaged recall.

well or not. In this paper we arrive at the same conclusion. The correlation we find between IR and classifier performance can be seen in Figure 1. Since the most commonly reported measure for determining class imbalance complexity i.e. IR does not imply complexity in the classification task, it is necessary to explore this field in depth, and find alternatives. A considerable effort in recent years has been put into broadening our understanding of the origin of complexity in class imbalanced scenarios. Recently the presence of the imbalance ratio between classes as well as their overlap has been studied in several works [2, 8, 9]. Even though the new measures are much more correlated with a degradation in classifier performance none of them have succeeded in replacing IR as the de facto standard measure of imbalance complexity. That might be because IR has a simple interpretation. In Section 2 we introduce the measures, we consider the nearest neighbors of the proposed measure, in this research field. In Section 4 we introduce the experimental framework used to evaluate the measures, and present our findings. In Section 5 we discuss several findings derived from this research project, and observations about the state of and direction the field. Finally in 6 we conclude the paper. The contributions in this paper is threefold. We define a new measure to the field of complexity measures for imbalanced datasets with as good performance as the former best performers, while being much easier to interpret. We make a series of synthetic datasets available, which have been described in multiple papers [9, 6], as well as new datasets, which tests the local imbalance problem. We supply efficient Python implementations of all measures described in this paper, that unlike the existing implementations operate in the popular Scikit-Learn style.

Goettcke: Start adding all the citations.

2 Related Work

Most of the performant dataset complexity measures in the imbalanced classification problem space is based on nearest neighbors. In Barella et al. [2] *Assessing the data complexity of imbalanced datasets* they adapt a selection of existing data complexity measures, to account for class imbalance. Through their very thorough experimental framework, they find some extremely robust measures where possibly the two most noticeable measures are their adaptation of N1 and N3.

Goettcke: It feels like there's missing something about measures not using nearest neighbors like the L and F measures

$$\text{N1}_{c_1}(T) = \frac{1}{n_{c_1}} \sum_{i=1}^{n_{c_1}} \text{I}((x_i^{c_1}, x_j) \in \text{MST} \wedge c_1 \neq y_j) \tag{2.2}$$

The adapted *N1* measure as described in 2.2 builds a minimum spanning tree (MST) over the dataset, and returns the fraction of edges connecting vertices from class c_1 to vertices of class y_j where $y_j \neq c_1$. The identity function I in the equation is 1 if the inner expression is true.

$$\text{N3}_{c_1}(T) = \frac{\sum_{i=1}^{n} \text{I}(\text{NN}(x_i^{c_1}) \neq c_1)}{n_{c_1}} \tag{2.3}$$

The adapted N3 measure returns the leave one out error rate of the first nearest neighbor classifier for class c_1. Here $\text{NN}(x_i)$ returns the label of the nearest neighbor of instance x_i.

For both the adapted N1 and N3 which altered from the original versions to be class specific measures, the aggregate measure for the entire dataset is found by taking the mean of each measure.

Mercier et al. [6] presents two measures dubbed Degree IR (degIR) and Overlap degree (degOver) which measures the complexity as a function of local imbalance and overlap. The overlap is determined by the 5NN set. If the 5-nearest neighbors of an instance belong to the same class as the instance, then the instance is not in a region of overlap, otherwise it is.

The number of points from the minority class, that are in an overlapping region is defined as n_{min_over} and the number of points from the majority class, that are in an overlapping region is defined as n_{maj_over}. Then the overlap degree is defined as shown in Equation 2.4

$$\text{degOver}(\mathcal{D}) = \frac{n_{min\_over} + n_{maj\_over}}{n} \tag{2.4}$$

The degree IR after normalization is defined as 2.5

$$\text{degIR}(\mathcal{D}) = 1 - \frac{n_{min\_over}}{\frac{n}{2}} = 1 - \frac{2n_{min\_over}}{n} \tag{2.5}$$

Goettcke

Currently I did not include the degIR measure in the results, maybe we should include more results in the tables, and just put the plots for the best ones

3 Tomek link complexity measure

Goettcke

Is ImbCol package here anywhere?

Here we define a measure based on the To[10] commonly used in undersampling. Tome pairs of mutual nearest neighbors of opposite described in Equation 3.6 where x_i and x_j ar belonging to class y_i and y_j. The point arbitrary point in the dataset. C_{min} is minority class instances.

$$\begin{gathered} d(x_i, x_j) < d(x_i, x_k) \\ d(x_i, x_j) < d(x_j, x_k) \\ y_i = C_{min} \vee\ y_j = C_{min} \\ y_i \neq y_j \end{gathered} \tag{3.6}$$

Given the definition of a Tomek Link in Equation 3.6 the measure described in Equation 3.8 calculates the proportion of minority instances entangled in a Tomek Link.

Goettcke

Citation

$$\text{TLS}(\mathcal{D}) = \sum_{i \in C_{min}} \sum_{j \in \mathcal{D} \setminus C_{min}} TL(x_i, x_j) \tag{3.7}$$

$$\text{TLCM}(\mathcal{D}) = \frac{\text{TLS}(\mathcal{D})}{|C_{min}|} \tag{3.8}$$

Goettcke

Do we talk about dimensionality

In Equation 3.7 the number of Tomek links in the dataset is defined as TLS. Here $\mathcal{D}$ is the set of all instances in the domain and TL is the identity function that returns 1, if x_i and x_j form a Tomek link according to definition 3.6. The cardinality $|C_{min}|$ defines the number of points within the set of minority instances. The measure proposed in this paper is defined as the Tomek Link Complexity Measure (TLCM) and is defined in Equation 3.8. Tomek Links on the decision boundary adds to the complexity e.g. if we have two uniform linearly separable distributions parallel to each other as in Figure 3e, then a measure based on the first few nearest neighbors of the minority points will almost always have the majority class in the decision set. Tomek Links ensures this is not the case, since the two instances have to be mutual nearest neighbors, which improves this nearest neighbor based approach ability at catching separability. The TLCM measures can be summarized as the percentage of instances from the minority, entangled in a Tomek Link.

Goettcke

Mention how to compute for multiclass

Goettcke

Is D this first introduced here?

Goettcke

Maybe insert an illustration of what a Tomek

4 Experimental Evaluation

4.1 Datasets A subset of the synthetic datasets used in this paper are used in both Mercier et al. [6] and Santos et al. [9]. These datasets were rebuild using the adapted dataset generator from [9] and the dataset generator from Goettcke [4]. In addition we added the *local imbalance* dataset as well as the *uniform only boundary* dataset. The datasets *02a* and *02b* Figure 2a and 2b test the algorithms ability to predict groups within a noisy background distribution. Notice however that there are no noisy points within the subcluster distributions. The *03subcl5* dataset Figure 2c tests 5 sub-clusters within a noisy background distribution in the same way as *02a* and *02b*, but the density of the clusters becomes smaller with the top one being the most dense, and the bottom one being the least dense. The *clover* dataset Figure 2d increases the size of the decision boundary, as well as testing the classifiers ability to handle a non Gaussian distribution. The *paw3* dataset shown in Figure 2e tests the classifiers ability to predict borderline minority instances from a multi-modal minority distribution. The *paw3 border dense center* is used in other papers, but it is the authors belief that this tests the same problem as dataset *02a*. The Gaussian overlap datasets tests a basic type of overlap, where two Gaussians are placed close enough to each other, to have some instances falling into the span of the neighbour distribution. Here it is only shown for the case, where the means are respectively 1 and 4 standard deviations apart as shown in Figure 2g and 2h, but are also evaluated with 2 and 3 standard deviations. Overlap comes in many forms as shown in [9] and one of the forms, that were not present in the existing dataset lineup was a local imbalance problem, where only a part of the dataset is in an overlapping region. This was created by making one uniform majority distribution, and two connected uniform minority distributions. The two uniform minority distributions have different densities to change the local imbalance degree. The overall imbalance ratio between minority and majority will however not be altered by a change in the local imbalance degree, as fewer points in the overlapping distribution, just results in more points in the non-overlapping distribution. In addition to the synthetic datasets a set of commonly used datasets from the Keel repository was chosen. These datasets were chosen because of their popularity in the class imbalance literature but also based on three constraints, the number of instances should be smaller than ten thousand, they are binary problems, and the imbalance ratio is greater than 1.1.

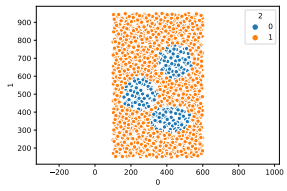
Goettcke

Since we have all of these datasets we should look at the performance in detail, and locate some of the datasets on the performacne plots

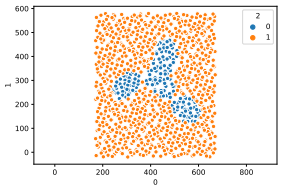
4.2 Evaluation Setup In the experiments for both the synthetic datasets and the real datasets a group of

Bellinger

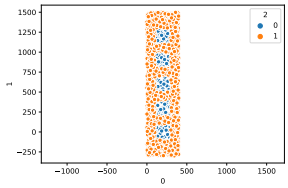
Ensure that the plots have axis-labels reason-



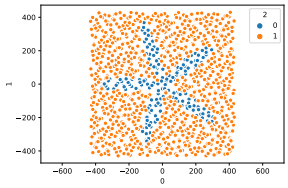
(a) 02a



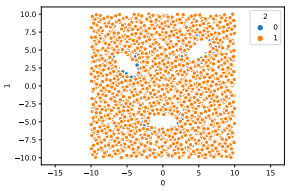
(b) 02b



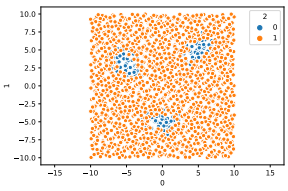
(c) 03subcl5



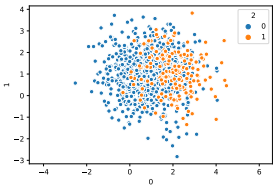
(d) 04clover5



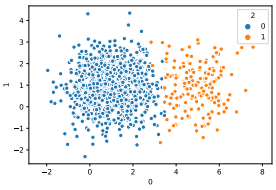
(e) paw3 only border



(f) paw3 border dense center

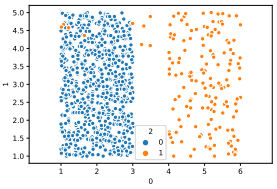


(g) Gaussian overlap 1 standard deviations apart.

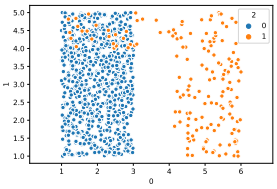


(h) Gaussian overlap 4 standard deviations apart.

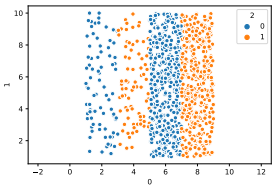
Figure 2: A subset of the synthetic datasets used in the experimental evaluation.



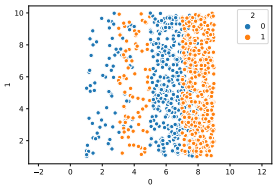
(a) Local imbalance degree with an IR of 5, and 5 percent of the minority data in the local overlap region.



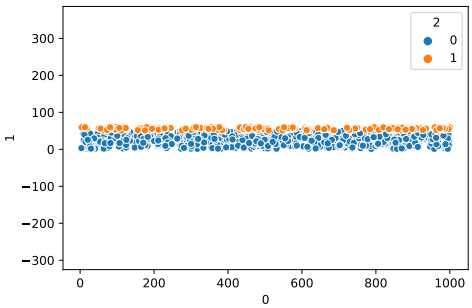
(b) Local imbalance degree with an IR of 5, and 20 percent of the minority data in the local overlap region.



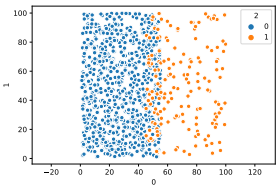
(c) multi modal no overlap



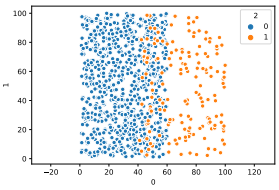
(d) multi modal with overlap



(e) Uniform only boundary



(f) 10% overlap.



(g) 20% overlap.

Figure 3: A series synthetic datasets highlighting different overlap scenarios, and dataset complexities. The imbalance ratio in all of the illustrated datasets was IR = 5.

Algorithm	Parameter	Space
kNN	k	$1, 3, 5, \ldots, 31$
MLP	learning rate	$0.1, 0.2, 0.3, \ldots, 1$
	neurons in hl.	$a-3, a-2, \ldots, a+3$
RF	trees	$100, 200, 300, \ldots, 1000$
	variables	$\frac{\sqrt{m}}{2}, \sqrt{m}, 2\sqrt{m}$
SVM	kernel	linear, radial, sigmoidal
	gamma	$2^{-10}, 2^{-9}, 2^{-8}, \ldots, 2^{10}$
	degree	$1, 2, 3, 4, 5$
DT	pt.	$0.1, 0.2, \ldots, 0.5$
	spl.	$1, 2, \ldots, 10$
NB	None	None

Table 1: Classifiers used in the experimental setup, and their respective parameter spaces. Hidden layers is abbreviated as hl. Decision trees is abbreviated DT. The pruning threshold as pt., and samples per leaf as spl.

classifiers were used to determine the correlation between the degradation in classifier performance, and the growth in complexity measure. The chosen supervised learning algorithms were Support Vector Machines (SVM), Random Forest (RF), Multilayer Perceptron (MLP), k-Nearest Neighbors kNN, decision trees (DT) and Naive Bayes (NB). All of the implementations came from the Scikit-Learn library for Python. To ensure decent performance for the classifiers a thorough search through a parameter space was done to find a close to optimal solution. The parameter space for each classifier can be found in Table 1. The parameter spaces were parameter optimized using the hyperopt package by Bergstra et al. [3], with up to 100 evaluations. The hyperopt package, by default, uses a tree-structured Parzen estimator to find close to optimal solutions in parameter spaces. In the parameter space for the Random Forest, m is the number of attributes in the dataset. In the parameter space for the Multilayer Perceptron $a = \frac{m+2}{2}$. These values were taken from the parameter spaces used in Barella et al. [2]. The function evaluated in the optimization step was 2 times 5 fold cross validation where the parameter optimized was the macro averaged recall. The result of the cross-validation run with the best performing hyper parameters was saved as the performance for that classifier, on that particular dataset. The detailed results as well as the parameters can be found in the papers repository. The mean classifier referred to in Section 4.3 is the mean performance the classifiers listed in 4 given the best parameters found during the parameter optimization. In this experimental framework the Euclidean distance was used for both the classifiers and the complexity measures. The adapted N1 and N3 complexity measures are available in the ImbCol package [1] for the R programming language. These were tested but found to be slow and unnecessarily memory intense. Furthermore they do not support the Euclidean distance, therefore it was deemed necessary to reimplement these measures. The new re-implementations were evaluated to ensure identical results to their R counterparts.

Measure	Cor. coef	p-value
IR	-0.497220	3.772179e-04
N1	-0.773819	1.794738e-10
N3	*-0.844482*	*8.708496e-14*
degIR	-0.421630	3.160467e-03
degOver	-0.374425	9.517588e-03
TLCM	**-0.852228**	**3.005538e-14**

Table 2: Pearson correlation analysis for correlation between macro averaged recall and the complexity measure

Measure	Cor. coef	p-value
IR	-0.279258	5.730783e-02
N1	*-0.829817*	*5.618854e-13*
N3	**-0.850518**	**3.820528e-14**
degIR	-0.126509	3.967933e-01
degOver	-0.552228	5.715403e-05
TLCM	-0.812841	3.945542e-12

Table 3: Pearson correlation analysis for correlation between gmean score and the complexity measure

4.3 Results

In Table 2,3 and 4 we have computed the Pearson correlation coefficient between the mean classifiers

Measure	Cor. coef	p-value
IR	-0.469837	8.621288e-04
N1	-0.748928	1.410436e-09
N3	**-0.781128**	**9.319564e-11**
degIR	-0.373966	9.612870e-03
degOver	-0.384524	7.615881e-03
TLCM	*-0.779805*	*1.051292e-10*

Table 4: Pearson correlation analysis for correlation between precision and the complexity measure

Goettcke
Point back to the synthetic datasets in the discussion.

Goettcke
Can be slightly hard to determine which dataset is what. Maybe add the mean-clf

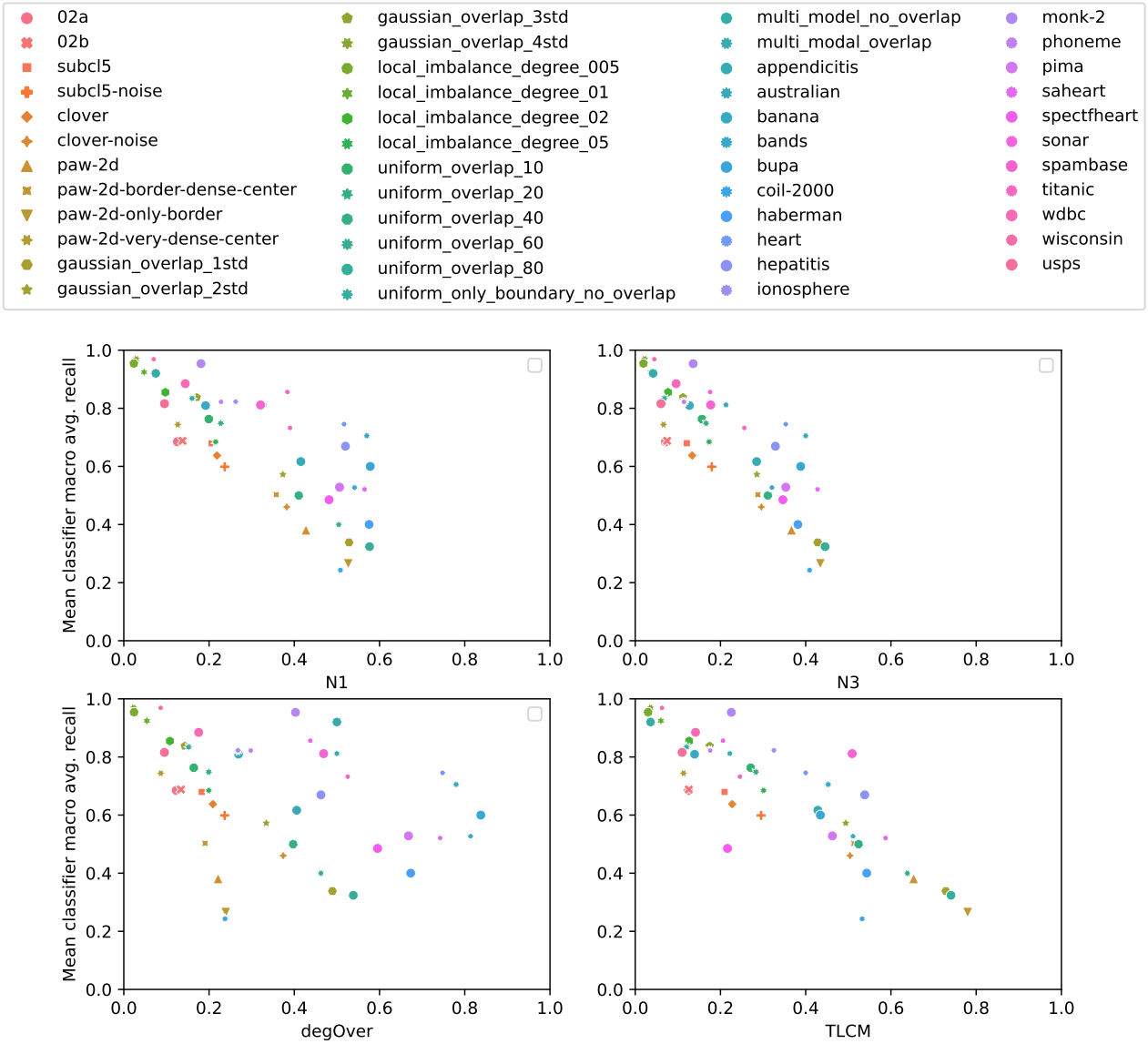


Figure 4: The correlation between a mean classifiers performance in terms of macro averaged recall, and the *TLCM* measure.

performance and the Imbalance ratio (IR), the N1 and N3 measure, degIR, degOver and finally the proposed TLCM measure. The corresponding p-value is also presented. Pearson's correlation coefficient measures the linear correlation between the two sets of data, and is normalized to be between -1 and 1. In the Tables we observe negative values approaching -1 which means that there is a linear correlation with a negative sign. In this case, this is the desired behaviour as we want to see a a degradation in classifier performance, as the complexity measure increases. Notice however, that a value of -1 does not mean that the relationship is one to one, i.e. a fall in classifier performance does not yield the same increase in complexity measure. The p-value expresses the probability, that we would have achieved this correlation coefficient if there was no linear correlation, and hence a lower value is better. The row with the best correlation coefficient is highlighted in **bold** and the second best is *emphasized*. As stated in the Section 1 the de facto standard complexity measure for imbalanced datasets is IR. As shown in the introduction, a visual inspection of the imbalance ratios linear correlation with the mean classifiers' performance Figure 1 is not very clear. This is confirmed for all three quality measures in the correlation tables. In terms of macro averaged recall 2, the TLCM measure performs best, with N1 as a close follow up. In terms of gmean scores N3 performs best, N1 second best and TLCM is the third best performer. In terms of precision N3 is number one TLCM number two and N1 is the number three. . Between these three measures the linear correlations are extremely close if we compare to the distances to the other measures in the lineup. The value they return is however not so similar which we can see in Figure 4.

Goettcke: Could be interesting to add the statistics on the distance to the cm=1-qm line we are compared to the others

5 Discussion

Because of the general linear relation between the performance measures typically used to evaluate class imbalanced problems and the TLCM measure, it would be interesting to determine, what causes the few datasets that are far out of this tendency, to exert such behavior. Given the robustness and performance of these complexity measures a new study should be done, to evaluate whether papers on methods handling the imbalance problem are actually evaluated on complex imbalance problems. A visualization to these imbalance problem handling techniques, could be to show where on the complexity plot a dataset is with respect to a certain benchmark classifier, and then show where it is after the imbalance handling technique has been applied.

Goettcke: We also need to show, that we are the most interpretable for all the other measures. Could be

6 Conclusion

Since the TLCM measure is the first proposed complexity measure which has a clear linear relationship with respect to mean classifier performance of the form $TLCM = 1 - (QM)$ where the quality measure can be both macro averaged recall and gmean score it fits well as a substitute to the imbalance ratio, because a score is as easy to interpret, yet holds much more information.

7 Contributions

- **We have made faster n1 and n3 implementations that works in the standard scikit-learn workflow style in Python. That also does not use the Gower distance, like the ImbCol measures.**
- **New TLCM measure**
- **We add a collection of datasets, and add the local imbalance datasets to the collection, so future researchers can think more about new problems than recreating old problems**

References

[1] V. H. Barella, L. P. F. Garcia, M. C. P. de Souto, A. C. Lorena, and A. C. P. L. F. de Carvalho, *Data complexity measures for imbalanced classification tasks*, in IJCNN, IEEE, 2018, pp. 1–8.

[2] ———, *Assessing the data complexity of imbalanced datasets*, Inf. Sci., 553 (2021), pp. 83–109.

[3] J. Bergstra, D. Yamins, and D. D. Cox, *Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures*, in ICML (1), vol. 28 of JMLR Workshop and Conference Proceedings, JMLR.org, 2013, pp. 115–123.

[4] J. M. N. Goettcke, *DS-Pipe. Available at* `https://git.imada.sdu.dk/goettcke/DS_Pipe`, 2022.

[5] J. Luengo, A. Fernández, S. García, and F. Herrera, *Addressing data complexity for imbalanced data sets: analysis of smote-based oversampling and evolutionary undersampling*, Soft Computing, 15 (2011), pp. 1909–1936.

[6] M. Mercier, M. S. Santos, P. H. Abreu, C. Soares, J. P. Soares, and J. A. M. Santos, *Analysing the footprint of classifiers in overlapped and imbalanced contexts*, in IDA, vol. 11191 of Lecture Notes in Computer Science, Springer, 2018, pp. 200–212.

[7] J. Ortigosa-Hernández, I. Inza, and J. A. Lozano, *Measuring the class-imbalance extent of multi-class problems*, Pattern Recognit. Lett., 98 (2017), pp. 32–38.

[8] V. Raj, S. Magg, and S. Wermter, *Towards effective classification of imbalanced data with convolutional neural networks*, in IAPR Workshop on Artificial Neural Networks in Pattern Recognition, Springer, 2016, pp. 150–162.

[9] M. S. Santos, P. H. Abreu, N. Japkowicz, A. Fernández, C. Soares, S. Wilk, and J. Santos, *On the joint-effect of class imbalance and overlap: a critical review*, Artificial Intelligence Review, (2022), pp. 1–69.

[10] I. Tomek, *Two Modifications of CNN*, IEEE Transactions on Systems, Man, and Cybernetics, 7(2) (1976), pp. 679–772.

[11] R. Zhu, Y. Guo, and J. Xue, *Adjusting the imbalance ratio by the dimensionality of imbalanced data*, Pattern Recognit. Lett., 133 (2020), pp. 217–223.

[12] R. Zhu, Z. Wang, Z. Ma, G. Wang, and J. Xue, *LRID: A new metric of multi-class imbalance degree based on likelihood-ratio test*, Pattern Recognit. Lett., 116 (2018), pp. 36–42.