Good Classification Measures and How to Find Them

Anonymous Author(s)
Affiliation
Address
email

Abstract

Several validation indices can be used for evaluating classification results: accuracy, F-measure, and many others. Can we say that some of them are better than others, or, ideally, choose one index that is best in all situations? To answer this question, we conduct a systematic analysis of classification evaluation indices. We focus on binary and multiclass problems, formally define a list of desirable properties, and theoretically analyze which indices satisfy which properties. We also prove an impossibility theorem: some desirable properties cannot be simultaneously satisfied. Finally, we propose a new family of indices that generalizes known indices with good properties and satisfies all desirable properties except one. We believe that our systematic approach gives an important tool to practitioners for adequately evaluating classification results.

1 Introduction

Classification is a classic machine learning task that is used in countless applications. To evaluate classification results, one has to compare *reference* and *candidate* classifications of a given set of elements. For this, *validation* indices are used, and there are many well-known ones like accuracy, F-measure, and so on [7]. The fact that different indices behave differently is known throughout the literature [1, 7, 15]. For instance, accuracy is known to be biased towards the majority class. Thus, different indices may give different evaluation results, and it is important to choose a *good* index. While there are attempts to compare evaluation indices and describe their properties [2, 4, 7, 9, 15], the problem still lacks a systematic approach, and our paper aims at filling this gap.[1] Our research is particularly motivated by a recent paper [6] providing a systematic analysis of evaluation indices for the *clustering* task. We transfer many of the proposed properties to the classification problem and additionally extend the research by adding more properties, new indices, and novel theoretical results.

To motivate our study, we first demonstrate through a series of experiments that different measures can be inconsistent in various situations. Also, some pairs of indices agree with each other more often than others. As shown by our theoretical analysis, these consistent indices have more desirable properties.

To provide a systematic comparison of validation indices, we formally define a list of properties that are desirable across various classification tasks. The proposed properties can be applied both to binary and multiclass problems. Some properties are intuitive and straightforward, like symmetry, while others are more tricky. A particularly important property is called *constant baseline*. It requires an index to not be biased towards particular predicted class sizes. For each index and each property, we formally prove or disprove that the property is satisfied. We believe that this analysis is essential for better understanding the differences between the indices.

Then, we analyze relations between different properties in the binary case and prove an impossibility theorem: we show that it is impossible for an index to be linearly transformable to a metric and

[1] We describe related research in more detail in Supplemental A.

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

simultaneously have the constant baseline property. This means that at least one of these properties has to be discarded. If we relax the set of properties by discarding the distance requirement, we get a set of properties that can be simultaneously satisfied. In fact, we propose a family of indices called Generalized Means (GM), satisfying all the properties and generalizing the well-known Matthew Correlation Coefficient (CC). In addition to MCC, this class contains a so-called Sokal&Sneath index that has not been used for classification evaluation but has good properties and is sometimes used for comparing clustering algorithms. If we instead discard the constant baseline (but keep its approximation), then the arccosine of CC is an index satisfying all the properties.

We hope that our research will motivate further studies analyzing the properties of validation indices for classification and other problems since there are still plenty of questions to be answered.

2 Validation indices for classification

In this section, we define indices that are commonly used for evaluating classification results. Classification problems can be divided into binary, multiclass, and multilabel. In this paper, we focus on *binary* and *multiclass* and leave multilabel for future research. There are several types of validation indices: *threshold* indices assume that each element is deterministically assigned to a class (or to several classes in the multilabel case) in the candidate classification (e.g., accuracy); *probability* indices assume that the predicted assignments are soft and compare the predicted probabilities with the actual outcome (e.g., the cross-entropy loss); *ranking* indices take into account the relative order of the predicted soft values, i.e., quantify whether the elements belonging to a class have higher predicted probabilities compared to other elements (e.g., area under the ROC curve or average precision). Due to space constraints, our research is focused on threshold indices. However, the discussed concepts can be extended to probability-based indices.

Now we introduce notation needed to formally define binary and multiclass threshold indices.[2] Let $n > 0$ be the number of elements in the dataset. Let $m \geq 2$ denote the number of classes. We assume that there is a *reference classification* of the elements into m classes. We also assume that we are given a *candidate classification* of the same elements. Let $\mathcal{C}$ be the confusion matrix: each matrix element c_{ij} denotes the number of elements belonging to the class i in the reference classification and classified as j in the candidate one. For binary classification, c_{11} is *true positive* (TP), c_{00} is *true negative* (TN), c_{10} is *false negative* (FN), and c_{01} is *false positive* (FP). We use the notation $a_i = \sum_{j=1}^{m} c_{ij}, b_i = \sum_{j=1}^{m} c_{ji}$ for the sizes of i-th class in the reference and candidate partitions, respectively, and $P_{ij}^{i} = \frac{c_{ij}}{a_i+b_i}, P_{ij}^{j} = \frac{c_{ij}}{a_j+b_j}$. Finally, we denote classification indices by $I(\mathcal{C})$ or $I(A, B)$, where A and B are reference and candidate classifications, and write $I(c_{11}, c_{10}, c_{01}, c_{00})$ for binary ones.

Table 1 lists several widely used classification indices. The most well-known is *Accuracy* which is the fraction of correctly classified elements. Accuracy is known to be biased towards the majority class, so it is not the best choice for unbalanced problems. To overcome this, *Balanced Accuracy* re-weights the terms to treat all classes equally. *Cohen's Kappa* uses a different approach to overcome this bias: it corrects the number of correctly classified samples by the expected value obtained by a random classifier [3]. *Matthews Correlation Coefficient* is the Pearson correlation coefficient between reference and candidate classifications for binary classification [5]. For the multiclass case, covariance is computed for each class, and the obtained values are averaged before computing the correlation coefficient. Finally, *Confusion Entropy* computes the entropy of the misclassification distribution for each class and averages the obtained values, see Table 1 and [16] for more details.[3]

Some indices are exclusively defined for binary classification. In this case, the classes are often referred to as 'positive' and 'negative'. The *Jaccard* index measures the fraction of correctly detected positive examples among all positive ones (both in reference and candidate classifications). *F-measure* is the (possibly weighted) harmonic mean of Recall (c_{11}/a_1) and Precision (c_{11}/b_1). For indices that do not have a natural multiclass variant, there are several universal extensions obtained via *averaging* the results obtained for m one-vs-all binary classifications [8]. For each such one-vs-all classification, a particular class i is considered positive while all other classes are grouped to a negative class.

[2] For convenience, notation used in the paper are listed in Table 1 of the supplemental material.

[3] There can be cases when a class is not present in the candidate partition. Then, some indices may contain division by zero. Such singularities can be filled in straightforwardly, as we discuss in Supplemental B.

Table 1: Commonly used (above the line) and novel (below the line) validation indices

Index	Binary	Multiclass
F-measure (F_β)	$\frac{(1+\beta^2)\cdot c_{11}}{(1+\beta^2)\cdot c_{11}+\beta^2\cdot c_{10}+c_{01}}$	micro / macro / weighted
Jaccard (J)	$\frac{c_{11}}{c_{11}+c_{10}+c_{01}}$	micro / macro / weighted
Matthew Coefficient (CC)	$\frac{c_{11}c_{00}-c_{01}c_{10}}{\sqrt{b_1\cdot a_1\cdot b_0\cdot a_0}}$	$\frac{n\sum_{i=1}^m c_{ii}-\sum_{i=1}^m b_i a_i}{\sqrt{\left(n^2-\sum_{i=1}^m b_i^2\right)\left(n^2-\sum_{i=1}^m a_i^2\right)}}$
Accuracy (Acc)	$\frac{\sum_{i=1}^m c_{ii}}{n}$	
Balanced Accuracy (BA)	$\frac{1}{m}\sum_{i=1}^m \frac{c_{ii}}{a_i}$	
Cohen's Kappa (κ)	$\frac{\sum_{i=1}^m c_{ii}-\frac{1}{n}\sum_{i=1}^m a_i b_i}{n-\frac{1}{n}\sum_{i=1}^m a_i b_i}$	
Confusion Entropy (CE)	$-\sum_{j=1}^m \frac{a_j+b_j}{2n}\sum_{i:i\neq j}\left(P_{ji}^j \log_{2m-2} P_{ji}^j + P_{ij}^j \log_{2m-2} P_{ij}^j\right)$	
Sokal&Sneath (S&S)	$\frac{1}{2m}\sum_{i=1}^m\left(\frac{c_{ii}}{a_i}+\frac{c_{ii}}{b_i}\right)$	
Fowlkes&Mallows (FM)	$\frac{c_{11}}{\sqrt{a_1 b_1}}$	micro / macro / weighted
Generalized Means (GM)	$\frac{n\,c_{11}-a_1 b_1}{\sqrt[r]{\frac{1}{2}\left(a_1^r a_0^r+b_1^r b_0^r\right)}}$	micro / macro / weighted
Correlation Distance (CD)	$\arccos(\mathrm{CC})$	micro / macro / weighted

Micro averaging sums up all binary confusion matrices corresponding to m one-vs-all classifications. Formally, it sets true positive as $\sum_{i=1}^m c_{ii}$, false negative and false positive as $n-\sum_{i=1}^m c_{ii}$, and true negative as $(m-2)n+\sum_{i=1}^m c_{ii}$. Then, a given binary index is applied to the obtained matrix.

Macro averaging computes the metric values for m binary classification sub-problems and then averages the results: $\frac{1}{m}\sum_{i=1}^m I(c_{ii}, a_i-c_{ii}, b_i-c_{ii}, n-a_i-b_i+c_{ii})$, where $I(\cdot)$ is a given binary classification index. Note that macro averaging gives equal weights to all one-vs-all binary classifications.

In contrast, *weighted averaging* weights one-vs-all binary classifications according to the sizes of the corresponding classes: $\frac{1}{n}\sum_{i=1}^m a_i\cdot I(c_{ii}, a_i-c_{ii}, b_i-c_{ii}, n-a_i-b_i+c_{ii})$.

As clearly seen from the discussion above, there are many options for a classification validation index. The next section demonstrates that the indices may give contradictory results, affecting both research outcomes and real services. In this paper, we propose a formal approach that allows for a better understanding of the differences between the indices and for making an informed decision among them for a particular application.

3 Motivational experiments

In this section, we describe several experiments that demonstrate the importance of the problem considered in this paper. Throughout the section, we also refer to novel indices from Table 1, while we discuss and motivate them in Section 5. In a nutshell, we analyze *inconsistency* of different indices. For binary classification, we consider all indices from Table 1. For the F-measure, we take $\beta=1$, for Generalized Means, we consider $r=1$. Let us also note that S&S and CC are also instances of GM with $r=-1$ and $r\to 0$, respectively (see Section 5). Let us also note that Jaccard is a monotone transformation of F_1, and CD is a monotone transformation of CC. Thus, we omit CD and Jaccard from all inconsistency tables. The code for our experiments supplements the submission.

Inconsistency on synthetic data First, we construct simple synthetic examples showing the inconsistency of all pairs of binary classification indices. Following [6], we say that two indices I_1 and I_2 are inconsistent on a triplet of classifications (A, B_1, B_2) if $I_1(A,B_1)>I_1(A,B_2)$ but $I_2(A,B_1)<I_2(A,B_2)$.

We take $n\in\{2,3,\ldots,10\}$ and went through all the possible triplets (A, B_1, B_2) of binary classifications on n elements (we additionally require that all classifications contain both classes). For

each triplet, we check which pairs of indices are inconsistent. We say that a pair of indices is indistinguishable for a given n if it is consistent on all triplets.

Table 2 lists all indices that are indistinguishable for a given n. Importantly, the most consistent indices are CC, S&S, and GM; and these are indices having the best properties according to our analysis in Sections 4 and 5. This supports our intuition that "good" indices agree with each other better than those having fewer desired properties.

For $n = 10$, we were able to find five triplets (A, B_1, B_2) allowing us to distinguish all pairs of indices. Due to space constraints, we move these examples to Supplemental E together with the inconsistency analysis for synthetic multiclass data.

Table 2: Indistinguishable indices for given n

n	indices
2	all
3-4	[Acc, BA, κ, F_1, FM, GM, CC, S&S]
5-6	[BA, κ, GM, CC, S&S] [F_1, FM]
7	[GM, CC, S&S]
8-9	[CC, S&S]
10	—

Inconsistency on UCI datasets In our second set of experiments, we measure how often inconsistency of indices occurs in practice when algorithms are compared on real datasets. For binary classification, we consider 8 UCI datasets, see Supplemental E for a detailed description and references. To these datasets, we apply 10 algorithms for binary classification available in the scikit-learn library [13]: kNN Classifier, Linear SVM, RBF SVM, Gaussian Process, Decision Tree, Random Forest, MLP, AdaBoost, Gaussian Naive Bayes, and Quadratic Discriminant Analysis. Each algorithm is used with default parameters. This gives us $8 \cdot \binom{10}{2} = 360$ triplets (A, B_1, B_2), where A is the reference classification and B_1, B_2 are provided by two different algorithms.

For each pair of indices, Table 3 shows the percentage of triplets for which this pair is inconsistent. Note that all pairs of indices are inconsistent in some cases. Again, the most consistent indices are CC and S&S, which both have good properties. Also, these two indices agree well with GM and κ. This additionally supports our claim that good indices often agree with each other. Surprisingly, the most inconsistent pair is BA and F_1, while both are widely accepted in the ML community. This inconsistency can be caused by the bias of F_1. Additional details on this experiment, together with the extended table, can be found in Supplemental E. There, we also include a similar experiment on UCI multiclass datasets. We omit this experiment from the main text since we discuss the multiclass scenario further in this section for two practical tasks: image and sentiment classification.

Table 3: Inconsistency of indices on UCI binary tasks, %

	Acc	BA	κ	F_1	GM	CC	S&S
Acc	—	11.4	7.5	2.8	6.9	7.5	10.8
BA		—	2.2	11.9	1.7	2.8	3.3
κ			—	8.9	1.1	1.1	1.7
F_1				—	7.8	8.3	11.4
GM					—	1.1	1.7
CC						—	0.56
S&S							—

Image classification Now, we move to a popular task of image classification. We consider the classic ImageNet dataset [12] and the top-10 that considered to be state-of-the-art at the moment of submission.[4] We check whether the leaderboard based on accuracy is consistent with the leaderboard based on other measures. For this, we apply the models to the test set, compute the confusion matrices, and compare all measures defined in Table 1.

Notably, the ImageNet dataset is balanced. This makes all measures more similar to each other. For instance, Accuracy and BA are equal in this scenario. Also, the *constant baseline* property defined in the next section is needed only for unbalanced datasets. Thus, indices are *more consistent* on balanced data. Nevertheless, we notice that the ranking can be inconsistent starting from the algorithm ranked fifth on the leaderboard.

The (partial) results are shown in Table 4. Here we compare EfficientNet-B7 NoisyStudent [17] and Swin-B Transformer (patch size 4x4, window size 12x12, image size 384^2) [10]. One can see that the measures inconsistently rank the algorithms: Confusion Entropy, Jaccard, and S&S disagree with

[4] `https://github.com/rwightman/pytorch-image-models/blob/master/results/results-imagenet.csv` (May 8, 2021).

Table 4: Inconsistent results on ImageNet. For all indices without default multiclass formula, we use the macro averaging.

	Acc	GM	κ	CE	F_1	J	CC	CC_{macro}	S&S
Efficientnet	**86.46**	**86.28**	**86.44**	6.59	**86.30**	77.52	**86.44**	86.419	86.57
Swin	86.43	86.26	86.42	**6.49**	86.27	**77.53**	86.42	**86.423**	**86.61**

Table 5: Ranking algorithms according to different indices on SST-1: from 1 (best) to 7 (worst)

	Acc	BA	invBA	κ	CE	F_1	FM	GM	J	CC	CC_{macro}	S&S
Flair+ELMo	1	1	2	1	1	1	1	1	1	1	1	1
Flair+BERT	2	4	1	2	2	5	5	5	5	2	2	2
Svm	3	3	4	3	5	3	3	3	3	3	4	4
Logistic	4	5	3	5	3	4	4	4	4	5	5	3
FastText	5	2	6	4	6	2	2	2	2	4	3	5
VADER	6	6	7	6	7	6	6	6	6	6	6	7
TextBlob	7	7	5	7	4	7	7	7	7	7	7	6

accuracy. Moreover, different averagings may often disagree with each other, as we see with macro and default (from Table 1) averagings of CC. Full results can be found in Supplemental E.

Sentiment analysis In the previous experiment, we noticed that despite several disagreements, the measures usually rank the algorithms similarly. We assume that this is due to the balance of the test set: all classes have equal sizes. However, the test set in practical applications is rarely balanced. Thus, we also consider an unbalanced classification task. In this experiment, we take the 5-class Stanford Sentiment Treebank (SST-5) dataset [14]. We compare the following algorithms: TextBlob, VADER, Logistic Regression, SVM, FastText, Flair+ELMo, Flair+BERT [11]. Table 5 shows that different metrics rank the algorithms differently. Among the indices shown in the table, the only consistent rankings are the one provided by κ and CC (default) and the second given by F_1, FM, GM, J. Note that the latter one significantly disagrees with the ranking by accuracy.

Section E of the supplemental material contains many additional experimental results. It also includes an experiment within a weather forecasting system, where the task is the binary classification of the presence of precipitation. In this experiment, we show that the choice of a measure affects the decisions made in production systems.

4 Properties of validation indices

This section discusses properties of validation indices that can be useful across various applications and formally checks which indices satisfy which properties. In this regard, we follow an approach proposed in [6] for comparing validation indices for *clustering* tasks.

First, we observe that some theoretical results from [6] can be directly applied to *binary* classification indices. Indeed, a popular subclass of clustering validation indices are *pair-counting* indices. Such indices are defined in terms of the following values: the number N_{11} of element-pairs belonging to the same cluster in both partitions, the number N_{00} of pairs belonging to different clusters in both partitions, the number N_{10} of pairs belonging to the same cluster in the reference partition but to different clusters in the candidate partition, and the number N_{01} of pairs belonging to different clusters in the reference partition but to the same cluster in the candidate partition. We note that the values N_{ij} essentially define a confusion matrix for binary classification on *element pairs*. Thus, replacing N_{ij} in pair-counting clustering indices by c_{ij}, we obtain a *binary* classification index. We refer to Supplemental B for the correspondence of some widely-used classification and clustering indices. In particular, Accuracy is equivalent to Rand, while Cohen's Kappa corresponds to Adjusted Rand Index. This equivalence allows us to transfer results from [6] to the context of binary classification. However, an important contribution of our work is to extend the properties and analysis to the multiclass case.

Similarly to [6], we note that all the discussed properties are invariant under linear transformations and interchanging reference and candidate classifications. Hence, we may check the properties for

Table 6: Properties of validation indices and averagings

Index	Max	Min	CSym	Sym	Dist	Mon	SMon	CB	ACB
F_1	✓	✗	✗	✓	✗	✓	✗	✗	✗
J	✓	✗	✗	✓	✓	✓	✗	✗	✗
CC	✓	✓/✗	✓	✓	✗	✓	✓/✗	✓	✓
Acc	✓	✓	✓	✓	✓	✓	✓	✗	✗
BA	✓	✓	✓	✗	✗	✓	✓	✓	✓
κ	✓	✗	✓	✓	✗	✓	✗	✓	✓
CE	✓	✗	✓	✓	✗	✗	✗	✗	✗
S&S	✓	✓	✓	✓	✗	✓	✓	✓	✓
FM	✓	✗	✗	✓	✗	✓	✗	✗	✗
GM	✓	✓	✓	✓	✗	✓	✓	✓	✓
CD	✓	✓	✓	✓	✓	✓	✓	✗	✓
Preserving properties by various averaging types									
Micro	✓	✗	✓	✓	✓	✓	✗	✗	✗
Macro	✓	✗	✓	✓	✓	✓	✗	✓	✓
Weighted	✓	✗	✓	✗	✗	✓	✗	✓	✓

at most one representative of each equivalence class and restrict to indices for which higher values indicate higher similarity between classifications.

Table 6 summarizes our findings: for each index, we mathematically prove or disprove each desirable property. Note that there are cases when a property is satisfied only for the binary case, they are marked as ✓/✗. In addition to individual indices, we also analyze the properties for micro, macro, and weighted multiclass averagings: for each averaging, we analyze whether it preserves a given property, assuming that binary classification index satisfies it. All the proofs can be found in Supplement E.

4.1 Minimal and maximal agreement

These properties make the upper and lower range of an index interpretable. The *maximal agreement* property requires the index to have an upper bound that is only achieved when the compared classifications are identical.

Definition 1. *We say that an index I satisfies* maximal agreement *if there exists a constant $c_{\max}$ such that for all $\mathcal{C}$, $I(\mathcal{C}) \leq c_{\max}$ holds, with equality iff $\mathcal{C}$ is diagonal.*

Also, for a given reference classification, there are several "worst" candidates, i.e., classifications that are wrong everywhere. This leads to the following property.

Definition 2. *We say that an index I satisfies* minimal agreement *if there exists a constant $c_{\min}$ such that for all $\mathcal{C}$, $I(\mathcal{C}) \geq c_{\min}$ holds, with equality iff the diagonal of $\mathcal{C}$ is zero, i.e. $c_{ii} = 0$ for all i.*

These properties allow for an easy and intuitive interpretation of the index values. While all of the indices in Table 6 do satisfy maximal agreement, there are popular indices such as Recall (c_{11}/a_1) and Precision (c_{11}/b_1) that do not satisfy this property as the maximum can also be achieved when the compared classifications are not identical. For minimal agreement, there are many indices that violate it. For example, Cohen's Kappa is obtained from Accuracy by subtracting the expected value of Accuracy and normalizing the result. As a result of the particular normalization that is used, it has minimal value $-\left(\sum_{i=1}^{m} a_i b_i\right) / \left(n^2 - \sum_{i=1}^{m} a_i b_i\right)$, which is clearly not constant.

If a binary index satisfies maximum agreement, then its multiclass variant obtained through micro, macro, or weighted averaging also satisfies maximum agreement as each of the one-vs-all binary classifications agrees maximally. This does not hold for minimal agreement: though each of the one-vs-all binary classifications will have zero true positives, the number of true negatives may still be positive.

4.2 Symmetry

Definition 3. *We say that an index I is* symmetric *if $I(\mathcal{C}) = I(\mathcal{C}^T)$ holds for all $\mathcal{C}$.*

In other words, we require symmetry with respect to permuting candidate and reference classifications. This property is often desirable since similarity is usually understood as a symmetric concept. However, in some specific applications, there may be reasons to treat the reference and predicted partitions differently. In such situations, some asymmetric index may be more desirable. An example of an asymmetric index is Balanced Accuracy. Let us also introduce *class-symmetry*, i.e., invariance to permuting the classes.

Definition 4. *We say that an index I is* class-symmetric *if, for any permutation π of the classes $\{1, \ldots, m\}$, and any confusion matrix $\mathcal{C}$, $I(\mathcal{C}) = I(\tilde{\mathcal{C}})$ holds, where $\tilde{\mathcal{C}}$ is given by $\tilde{c}_{ij} = c_{\pi(i),\pi(j)}$.*

Note that known multiclass indices are all class-symmetric, while in binary classification tasks, there can be asymmetry between 'positive' and 'negative' classes. Examples of well-known class-asymmetric binary classification indices are Jaccard and F_1.

4.3 Distance

In some applications, it is desirable to have a distance interpretation of an index: whenever a classification A is similar to B, while B is similar to C, it should intuitively hold that A is also somewhat similar to C. In particular, in many applications, the true classification is unknown, and the reference classification can be assumed to be an approximation of the truth. Then, we would want the similarity between predicted and reference classifications to be not too different from the similarity between predicted and true classifications. This would be guaranteed if the index is a distance.

Definition 5. *An index I is a* distance *if it can be linearly transformed to a metric distance.*

A function $d(A, B)$ is a metric distance if it is symmetric, nonnegative, equal to zero only when $A = B$, and satisfies the triangle inequality $d(A, C) \leq d(A, B) + d(B, C)$. Note that the first requirement is equivalent to symmetry as defined in Definition 3 while the second and third imply maximal agreement as defined in Definition 1. Furthermore, note that if d is a distance, then $c \cdot d$ is also a distance for any $c > 0$. Therefore, we can conclude that I is a distance if and only if I satisfies symmetry and maximal agreement while $c_{\max} - I(A, B)$ satisfies the triangle inequality.

While most of the indices cannot be linearly transformed to a distance, there are a few indices that actually do satisfy this property. For example, the Jaccard index can be transformed to the Jaccard distance $1 - \mathrm{J}(A, B)$. Similarly, Accuracy can be transformed to a distance by $1 - \mathrm{Acc}(A, B)$.

4.4 Monotonicity

Monotonicity is one of the most important properties of a similarity index: intuitively, changing one classification such that it becomes more similar to the other ought to result in an increase of the similarity score. Then, in order to formalize monotonicity, we need to determine what kind of changes make the classifications A and B more similar to each other. The simplest option would take one element on which A and B disagree, and resolve this disagreement.

Definition 6. *An index I is* monotone *if $I(\mathcal{C}) < I(\tilde{\mathcal{C}})$ for any confusion matrices $\mathcal{C}$ and $\tilde{\mathcal{C}}$ where $\tilde{\mathcal{C}}$ is obtained from $\mathcal{C}$ by decrementing an off-diagonal entry c_{ab} and incrementing either c_{aa} or c_{bb}.*

The definition of strong improvements defines a partial ordering over confusion matrices with the same total number of elements. However, we can relax the latter restriction and obtain the following, stronger notion of monotonicity that defines a partial ordering across different numbers of elements.

Definition 7. *An index I is* strongly monotone *if $I(\mathcal{C}) < I(\tilde{\mathcal{C}})$ holds for any confusion matrix $\mathcal{C}$ that has at least one positive off-diagonal entry and every $\tilde{\mathcal{C}}$ that has at least one positive diagonal entry and is obtained from $\mathcal{C}$ by either increasing a diagonal entry or decreasing an off-diagonal entry.*

Note that we have to require that $\mathcal{C}$ is not a diagonal matrix, since otherwise it would contradict the maximal agreement property as $I(\mathcal{C}) = c_{\max} \geq I(\tilde{\mathcal{C}})$ holds when $\mathcal{C}$ is diagonal. Similarly, we must require $\tilde{\mathcal{C}}$ to not be zero on the whole diagonal in order to not contradict minimal agreement.

While almost all indices (except for CE) in Table 6 satisfy monotonicity as defined in Definition 6, many fail to satisfy strong monotonicity. In particular, some widely used indices such as F_1, Jaccard and Cohen's Kappa do not satisfy this intuitive property.

4.5 Constant baseline

The constant baseline is perhaps the most important non-trivial property. On the one hand, it ensures that an index is not biased towards candidates with particular class sizes $b_1, \ldots, b_m$. On the other hand, it also ensures some interpretability for 'mediocre' predictions.

Intuitively, if a predicted classification is drawn at random and independently of the reference classification, we would expect it to have a low similarity with the reference classification. Then, if another prediction has a similarly low score, we can say that it is roughly as bad as a random guess. However, this is only possible when such random classifications achieve similar scores, independent from their class sizes. To formalize this, let $U(b_1, \ldots, b_m)$ denote the uniform distribution over classifications with the class sizes $b_1, \ldots, b_m$. We say that the class sizes $b_1, \ldots, b_m$ are *unary* if $b_i = n$ for some $i \in \{1, \ldots, m\}$. That is, if all elements get classified to the same class so that $U(b_1, \ldots, b_m)$ is a constant distribution.

Definition 8. *We say that an index I has a* constant baseline *if there exists $c_{base}(m)$ that does not depend on n but may depend on m, such that for any reference classification A and non-unary class sizes $b_1, \ldots, b_m$, it holds that $\mathbb{E}_{B \sim U(b_1, \ldots, b_m)}[I(A, B)] = c_{base}(m)$.*

Note that we need to require the class sizes to be non-unary: if these class sizes are unary, we will have contradictions with maximal and minimal agreement when the class sizes of A are also unary. Many popular indices such as F_1, Accuracy and Jaccard do not have a constant baseline. Furthermore, most of the indices that do have a constant baseline, were deliberately designed to have one. For example, Cohen's Kappa was obtained from Accuracy by correcting it for chance.

All of the indices that satisfy constant baseline turn out to be linear functions in terms of c_{ii} for fixed class sizes $a_1, \ldots, a_m, b_1, \ldots, b_m$. For such indices, linearity of the expectation can be utilized to easily compute the baseline by substituting the expected values $\mathbb{E}_{B \sim U(b_1, \ldots, b_m)}[c_{ii}] = \frac{a_i b_i}{n}$. While our definition of the constant baseline does allow for a baseline $c_{\text{base}}(m)$ that depends on the number of classes m, many indices such as the Matthew Coefficient and Cohen's Kappa have a baseline that is constant w.r.t. m. We also propose the following relaxation of the constant baseline property.

Definition 9. *An index I is said to have an* approximate constant baseline *if there exists a function $c_{base}(m)$ that does not depend on n but may depend on m such that for any sets of class sizes $a_1, \ldots, a_m$ and $b_1, \ldots, b_m$, $I(\bar{C}) = c_{base}(m)$, where $\bar{c}_{ij} = \frac{a_i b_j}{n}$.*

The advantage of this relaxation is that it allows us to non-linearly transform indices while still maintaining an approximate constant baseline. Take for example Matthews Coefficient: it cannot be linearly transformed to a distance while the transformations $\text{CD}(A, B) = \arccos(\text{CC}(A, B))$ and $\sqrt{2(1 - \text{CC}(A, B))}$ do yield distances. Because Matthews Coefficient has a constant baseline, these non-linear transformations have an approximate constant baseline.

5 An impossibility theorem for classification

In this section, we focus on binary classification and deeper analyze the relations between the properties discussed above. Unfortunately, it turns out that the properties formulated in the previous section cannot all be simultaneously satisfied:

Theorem 1. *There is no classification index that simultaneously satisfies the monotonicity, distance, and constant baseline properties.*

Proof. Let A be a classification with a single positive and $n - 1$ negatives. Let B_1 be a random classification with a single positive and let B_2 be a random classification with two positives. The constant baseline requires $\mathbb{E}[I(A, B_1)] = \mathbb{E}[I(A, B_2)]$, which gives

$$\frac{1}{n} c_{\max} + \frac{n-1}{n} I(0, 1, 1, n-2) = \frac{2}{n} I(1, 0, 1, n-2) + \frac{n-2}{n} I(0, 1, 2, n-3),$$

which we rewrite to

$$2I(1, 0, 1, n-2) - c_{\max} = (n-1) I(0, 1, 1, n-2) - (n-2) I(0, 1, 2, n-3). \tag{1}$$

Now, we consider a classification C with a single positive that does not coincide with the positive of A and a classification B that has two positives which are the positives of A and C. The triangle inequality tells us that

$$c_{\max} - I(0,1,1,n-2) \leq 2c_{\max} - I(1,1,0,n-2) - I(1,0,1,n-2) = 2(c_{\max} - I(1,1,0,n-2)),$$

where the last step follows from symmetry (implied by distance). This is rewritten to

$$2I(1,1,0,n-2) - c_{\max} \leq I(0,1,1,n-2). \tag{2}$$

Combining (1) and (2), we obtain

$$(n-1)I(0,1,1,n-2) - (n-2)I(0,1,2,n-3) \leq I(0,1,1,n-2).$$

We rewrite this to $I(0,1,1,n-2) \leq I(0,1,2,n-3)$, which clearly contradicts monotonicity. □

Thus, we have to discard one of these properties. Obviously, discarding monotonicity would be highly undesirable, since higher index values would then not necessarily indicate higher similarity. For this reason, we analyze what happens if we discard either *distance* or *constant baseline*. Due to space constraints, we only state the results. For more detail, we refer to the Supplemental D.

Discarding distance In the supplemental material, we mathematically describe the class of indices that satisfy all properties of Table 6 except for distance. This derivation does rely on assuming some additional smoothness conditions. These smoothness conditions are, however, satisfied by all indices discussed in this paper. The class of indices satisfying all properties is quite wide and contains many unelegant indices. An interesting subclass of this wider class is *Generalized Means* (see Table 1): for any $r \in \mathbb{R}$, the index GM_r satisfies all properties except for being a distance. The name 'Generalized Means' refers to the fact that the index is normalized by the generalized mean of a_0a_1 and b_0b_1. When setting $r = -1$ (corresponding to the harmonic mean), the resulting index is linearly equivalent to S&S via $\text{S\&S}(A,B) = (1 + \text{GM}_{-1}(A,B))/2$. When taking the limit $r \to 0$ (corresponding to the geometric mean), we retrieve Matthew's Correlation. Thus, for binary classification tasks where a distance interpretation is not required, our analysis suggests using an index from this Generalized Means class (such as CC or S&S). In this class, CC has the additional advantage that it can non-linearly be transformed to a distance by taking its arccosine, while S&S has the advantage that all properties are also satisfied for multiclass classification.

Discarding (exact) constant baseline Note that Theorem 1 only proves an impossibility for the *exact* constant baseline, but not the *approximate* constant baseline. In particular, the transformations $\text{CD}(A,B) = \arccos(\text{CC}(A,B))$ and $\text{CD}'(A,B) = \sqrt{2(1-\text{CC}(A,B))}$ have an approximate constant baseline while still satisfying all other properties. In the Supplemental D, we argue that CD approximates a constant baseline with one order of precision better than CD′. Essentially, this is a consequence of the fact that the transformation $\arccos(\text{CC})$ is a symmetric function around the constant baseline $\text{CC} = 0$ while $\sqrt{2(1-\text{CC})}$ is not. In more detail, we show that the leading error term of CD′ is of the order $\mathbb{E}[\text{CC}(A,B)^2]$ while the leading error term for CD is of the order $\mathbb{E}[\text{CC}(A,B)^3]$. Currently, we are not aware of other distance indices for which the constant baseline is approximated up to the same order of precision. We thus argue that for binary classification tasks where a distance interpretation is desirable, the Correlation Distance is the most suitable index.

6 Future work

In this paper, we propose a systematic approach to choosing a suitable validation index for classification tasks. However, there are still many open questions and promising directions for future research. First of all, we would like to see whether one could construct a set of desirable properties that can be used as axioms to uniquely define one good index (or a parametrized group of indices). Second, it is an open problem whether Generalized Mean indices in general (or S&S in particular) can be converted to a distance via a continuous transformation. Finally, our work does not cover ranking and probability-based indices. Thus, we leave aside such widely used indices as Logloss and AUC. Formalizing and analyzing their properties is an important direction for future research.

References

[1] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann. The balanced accuracy and its posterior distribution. In *2010 20th international conference on pattern recognition*, pages 3121–3124. IEEE, 2010.

[2] D. Chicco and G. Jurman. The advantages of the matthews correlation coefficient (mcc) over f1 score and accuracy in binary classification evaluation. *BMC genomics*, 21(1):1–13, 2020.

[3] J. Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46, 1960.

[4] C. Cortes and M. Mohri. Auc optimization vs. error rate minimization. *Advances in neural information processing systems*, 16(16):313–320, 2004.

[5] J. Gorodkin. Comparing two k-category assignments by a k-category correlation coefficient. *Computational biology and chemistry*, 28(5-6):367–374, 2004.

[6] M. Gösgens, L. Prokhorenkova, and A. Tikhonov. Systematic analysis of cluster similarity indices: How to validate validation measures. *International Conference on Machine Learning (ICML)*, 2021.

[7] M. Hossin and M. Sulaiman. A review on evaluation metrics for data classification evaluations. *International Journal of Data Mining & Knowledge Management Process*, 5(2):1, 2015.

[8] O. Koyejo, N. Natarajan, P. Ravikumar, and I. S. Dhillon. Consistent multilabel classification. In *NIPS*, volume 29, pages 3321–3329, 2015.

[9] C. X. Ling, J. Huang, H. Zhang, et al. Auc: a statistically consistent and more discriminating measure than accuracy. In *Ijcai*, volume 3, pages 519–524, 2003.

[10] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv preprint arXiv:2103.14030*, 2021.

[11] P. Rao. Fine grained sentiment classification. `https://github.com/prrao87/fine-grained-sentiment`, 2021.

[12] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[13] Scikit-learn. Clustering algorithms. `https://scikit-learn.org/stable/supervised_learning.html`, 2021.

[14] R. Socher, A. Perelygin, J. Wu, J. Chuang, C. D. Manning, A. Y. Ng, and C. Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642, 2013.

[15] M. Sokolova and G. Lapalme. A systematic analysis of performance measures for classification tasks. *Information processing & management*, 45(4):427–437, 2009.

[16] J.-M. Wei, X.-J. Yuan, Q.-H. Hu, and S.-Q. Wang. A novel measure for evaluating classifiers. *Expert Systems with Applications*, 37(5):3799–3809, 2010.

[17] Q. Xie, M.-T. Luong, E. Hovy, and Q. V. Le. Self-training with noisy student improves imagenet classification. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10687–10698, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 6.
   (c) Did you discuss any potential negative societal impacts of your work? [No] Our work is theoretical and its aim is to make research and experiments less biased.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Sections 4 and 5.
   (b) Did you include complete proofs of all theoretical results? [Yes] See the supplemental material.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See the supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 3 and the supplemental material.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [No] We do not have any heavy computations and do not measure computation time.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 3 and the supplemental material.
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]