Contingency Space: A Semimetric Space for Classification Evaluation

Azim Ahmadzadeh, Dustin J. Kempton, Petrus C. Martens, and Rafal A. Angryk

Abstract—In Machine Learning, a supervised model's performance is measured using the evaluation metrics. In this study, we first present our motivation by revisiting the major limitations of these metrics, namely one-dimensionality, lack of context, lack of intuitiveness, uncomparability, binary restriction, and uncustomizability of metrics. In response, we propose Contingency Space, a bounded semimetric space that provides a generic representation for any performance evaluation metric. Then we showcase how this space addresses the limitations. In this space, each metric forms a surface using which we visually compare different evaluation metrics. Taking advantage of the fact that a metric's surface warps proportionally to the degree of which it is sensitive to the class-imbalance ratio of data, we introduce Imbalance Sensitivity that quantifies the skew-sensitivity. Since an arbitrary model is represented in this space by a single point, we introduce Learning Path for qualitative and quantitative analyses of the training process. Using the semimetric that contingency space is endowed with, we introduce Tau as a new cost sensitive and Imbalance Agnostic metric. Lastly, we show that contingency space addresses multi-class problems as well. Throughout this work, we define each concept through stipulated definitions and present every application with practical examples and visualizations.

Index Terms—Machine learning, model validation and analysis, knowledge representation formalisms and methods

1 Introduction

In order to evaluate the performance of a supervised model, we often analyze the confusion matrix (a.k.a. the truth table). For a categorical (binary or not) classification problem, this matrix shows the interrelation between two variables; the actual and the estimated class labels of the data, inspired by the contingency table [1]. For a k-class problem, this matrix contains k^2 values. A function that aggregates these values into one single value is called a supervised performance evaluation metric. In this document, we call them *single-value* metrics.

There are numerous performance evaluation metrics and they each quantify the success of models with respect to their unique objectives. Abundance of such metrics is an indication that there is no "one size fits all" metric. In 1884, an interesting conversation arose by an overly optimistic verification methodology of a tornado forecast model that claimed a 95% success rate [2]. This superficial success rate initiated a decade-long, focused discussion about the adequacy of different evaluation methods. This event is now known as the "Finley affair" [3] and it gave birth to many forecast metrics. Similar critical views have been expressed in other domains as well.

In spite of decades of outstanding research in this direction, the community continues to see shortcomings and room for improvement in the evaluation process. The 2016 National Artificial Intelligence R&D Strategic Plan, published by the United States government, Executive Office of the President, highlighted the importance of evaluation measures and methodologies for machine learning algorithms ([4], Strategy 6). It emphasized defining quantifiable measures "in order to characterize AI technologies, including but not limited to: accuracy, complexity, trust and competency, risk and uncertainty; explainability; unintended bias; comparison to human performance; and economic impact'' (page 33). This was reiterated in the 2019 update as well [5], in realization of the importance of trustworthiness, fairness, and bias of models. This strategic plan correctly identified and prioritized the need for more intuitive measures and transparent evaluation methodologies.

More broadly, there has been a number of fundamental concerns raised by many influential applied researchers regarding how the goals of AI/ML are set and being pursued. Chasing the competitions' leaderboards, proposing models which are not scalable to real-world problems, assessing models' performance against overly simplified benchmark datasets, "mathiness" of research at the cost of intuitiveness, and overexpectations and complacency in Machine Learning and most recently in Deep Learning, are some of these concerns [6], [7], [8], [9], [10], [11].

Although we do not claim to have a simple solution for all of these complicated challenges, we believe the limitations

- *Azim Ahmadzadeh, Dustin J. Kempton, and Rafal A. Angryk are with the Department of Computer Science, Georgia State University, Atlanta, GA 30303 USA. E-mail: {aahmadzadeh1, dkempton1, rangryk}@gsu.edu.*
- *Petrus C. Martens is with the Department of Physics and Astronomy, Georgia State University, Atlanta, GA 30303 USA. E-mail: martens@astro.gsu.edu.*

Manuscript received 20 June 2021; revised 4 Jan. 2022; accepted 9 Apr. 2022. Date of publication 13 Apr. 2022; date of current version 6 Jan. 2023.
This work was supported in part by two NASA under Grants NNH14ZDA001N and 80NSSC20K1352, and in part by two NSF under Grants AC1443061 and AC1931555. The AC1443061 Award was supported in part by funding from the Division of Advanced Cyber infrastructure within the Directorate for Computer and Information Science and Engineering, in part by the Division of Astronomical Sciences within the Directorate for Mathematical and Physical Sciences, and in part by the Division of Atmospheric and Geospace Sciences within the Directorate for Geosciences.
(Corresponding author: Azim Ahmadzadeh.)
Recommended for acceptance by L. Wang.
Digital Object Identifier no. 10.1109/TPAMI.2022.3167007

TABLE 1
A Reference Table of Basic Random Variables (R.V.) Commonly Used for Performance Evaluation of Binary, Classification Algorithms Notations

R.V.	Name	Description
p	condition positive	the number of real positive instances in the data
n	condition negative	the number of real negative instances in the data
p'	predicted condition positive	the number of predicted positive instances
n'	predicted condition negative	the number of predicted negative instances
tp	true positive	the number of positive instances classified as positive
tn	true negative	the number of negative instances classified as negative
fp	false positive	the number of negative instances classified as positive
fn	false negative	the number of positive instances classified as negative
tpr	true-positive rate; recall; sensitivity	the proportion of tp with respect to positive instances
tnr	true-negative rate; specificity; selectivity	the proportion of tn with respect to negative instances

we highlight in Section 2 contribute significantly to many of these concerns. Our proposed contingency space addresses those limitations in two layers: it provides an intuitive framework for a visual analysis of performance and its metrics, and more importantly, it gives birth to a number of concepts that allow new quantitative methods for performance analyses.

The organization of this paper is as follows. We highlight the limitations of single-value metrics in Section 2. In Section 3, we present the preliminary concepts through a number of stipulated definitions. The main idea, i.e., the definition of contingency space, is given in Section 4. This is followed by Section 5 in which a number of different applications of contingency space are discussed. After we have introduced the contingency space and its applications, we draw parallels between contingency space and ROC space in Section 6.1. We conclude this paper by laying out the future work.

2 Limitations of Single-Value Metrics

In this work we propose the contingency space to address the following concerns regarding the effectiveness of the single-value metrics. For unfamiliar variables the reader may consult Table 1.

One-Dimensional View. One of such concerns that is widely known to the community and intrinsic to the fact that these metrics are (by definition) summaries of the confusion matrix, is the one-dimensional view of the classification performance. An immediate cost of such a summarization is that these metrics may easily obscure the strengths and/or weaknesses of models, which might be visible from other points of view. Take recall (i.e., $\frac{tp}{p}$) as a simple example. This metric measures the probability of correct classification of positive instances while totally (although by design) disregarding models' performance on the negative class. Therefore, a model that correctly classifies all positive instances would be projected as 'perfect', even though it might misclassify many negative instances. This is why it is always coupled with precision (i.e., $\frac{tp}{p'}$), or instead, f_β score as the harmonic mean of precision and recall is used. Such remedies, although informative, recursively inherit the problem.

Another issue with the one-dimensional view of these metrics is that they map an infinite number of unique models to a constant value, and consequently estimate the discrepant performances of those models as equally good. Later in Section 5.1, we show how such families of models are distinguished in our proposed space. We go even further and show that many of these presumed-identical performances are not even relatively identical (see Def. 3.3).

Of course, if the utilized metric perfectly matches the objective of the problem, the one-dimensionality of the metric turns into a feature and will no longer be a limiting factor. But the "necessity and sufficiency conditions" must be investigated—an important step which is often considered as only complementary, uncommon, and at times redundant.

Lack of Context. Another major concern is a lack of context for the quantitative analysis of models' performance using these metrics. What a single-value evaluation metric returns as the quality of performance falls short of providing any context except the simple "the higher the better" interpretation. More accurately, for two arbitrary models, m_1 and m_2, and a given metric, μ, we say m_1 outperforms m_2 if and only if $\mu(m_1) > \mu(m_2)$. At least this is what the metric μ implies (see Def. 3.4). In the absence of any knowledge about the distribution of the utilized metrics, the degree of improvement with respect to a baseline model m_0 is then quantified with the difference $\mu(m_i) - \mu(m_0)$, implying a uniform distribution for μ—almost always a wrong assumption.

Lack of Intuitiveness. While performance evaluation metrics are well-defined statistical concepts, they lack intuitiveness, perhaps with the exception of the simplest ones such as tpr. An interesting example that illustrates the abstractness of these metrics is the f_β score [12]. We understand it as "the weighted harmonic mean of precision and recall". But this is not an intuitive measure given that the harmonic mean is not an intuitive concept and moreover, its inputs are functions themselves. It is an interesting observation that the f_1 score is far more popular than its generic form, the f_β score, despite the fact that in many real-world problems datasets are class imbalanced and the assumption of $\beta = 1$ completely disregards that. An experienced researcher can think of many other cases in their discipline, in which intuitiveness of metrics could have helped a team to choose a better measure and evaluate their models' performance more effectively.

Uncomparability. Although each of these metrics is a rather simple function of a few variables, they are uncomparable measures. Despite much effort put in correctly understanding the statistical meaning of such metrics, a perusal of the literature shows that there has been very little attention to the direct comparison of the metrics themselves. For instance, take two metrics from Table 2: the true skill

TABLE 2
Some of the Popular Classification Performance Metrics and Their Formulas Based on the Confusion Matrix

Notation	Metric	Definition
acc	accuracy	$\frac{tp+tn}{p+n}$
ba	balanced accuracy	$\frac{1}{2}(\frac{tp}{p}+\frac{tn}{n})$
gm	geometric mean	$(\frac{tp}{p}\cdot\frac{tn}{n})^{\frac{1}{2}}$
pre	precision	$\frac{tp}{p'}$
rec	recall (sensitivity)	$\frac{tp}{p}$
f_1	f_1 score	$2\cdot\frac{pre\cdot rec}{pre+rec}$
gss	Gilbert's skill score [16]	$\frac{tp-r}{tp+fp+fn-r}$
dss	Doolittle's skill score [17]	$\frac{(tp\cdot tn-fp\cdot fn)^2}{((tp+fp)\cdot(tp+fn)\cdot p\cdot n}$
tss	true skill statistic [13]	$\frac{tp}{p}-\frac{fp}{n}$
hss	Heidke skill score [18]	$\frac{2\cdot((tp\cdot tn)-(fn\cdot fp))}{p\cdot(fn+tn)+n\cdot(tp+fp)}$
j	Youden's J index [14]	$\frac{tp\cdot tn-fn\cdot fp}{(tp+fn)\cdot(fp+tn)}$
τ	Tau	defined in Def. 5.4.

statistic (tss) [13] and the Youden's j index (j) [14]. To draw any insightful parallels between the two measures one should spend considerable time to try and algebraically deduce (and only hope that there exists) a linear and simple-to-interpret relationship. Note that tss and j index, despite their very different formulas, are identical metrics. While it may take a few simple steps to verify this fact, only highly trained eyes may be able to infer this by only looking at the metrics' formulas. Not surprisingly, comparison of two arbitrary metrics rarely results in such a satisfying finding.

When researchers need to dig into the large pool of metrics and pick an appropriate one(s) for their performance evaluation, in the absence of intuitive methods for pairwise comparisons, they will either (1) rely on less appropriate measures—due to their popularity in their domain—or (2) reinvent the existing ones, which in turn only worsens the problem. In our previous example, tss (proposed for rare-event forecast models in the meteorology domain, in 1965) was simply a reinvention of the j index (introduced for a similar purpose but in the medical domain, in 1950). And as more and more domains of research are utilizing machine learning algorithms, the natural differences in the jargon and notations only make pursuit of the appropriate metrics more difficult. This has been pointed out before, (e.g., [3]), and it is still occurring in recent interdisciplinary research projects.

Binary Restriction. The evaluation process is more challenging for non-binary classification problems. Most performance evaluation metrics, however, are defined solely for handling the evaluation of binary problems. The common solution to this limitation is to aggregate the results by 'micro' and 'macro' averaging methods [15]. Averaging, as we know, is sensitive to outliers, and at the same time, obscures the important details of the per-class performance. A metric that captures the overall performance of a model on a multi-class classification problem, without relying on external aggregation, is a much needed, yet missing piece.

Uncustomizability. In many real-world problems the cost of the type I error (false positive or false alarm) is different from that of the type II error (false negative or miss). For instance, for a hurricane forecast model there is a higher tolerance for false alarms than a miss (of a hurricane) which can have devastating consequences. Whereas, in identifying suspicious banking activities the cost of a miss is more tolerable, as not every suspicious activity is necessarily fraudulent. But most of the performance evaluation metrics do not have built-in variables for the costs, and consequently, cannot be adjusted to different problems. Similarly, different datasets have different class-imbalance ratios. With the exception of the f_β score, popular performance evaluation metrics do not account for imbalance ratios. This lack of customizability is perhaps the main reason for the proliferation of evaluation metrics.

3 Basic Concepts

In the interest of ease and accuracy, throughout this paper we present a number of stipulated definitions for the prerequisite concepts, and also for a few novel ideas which are among the main contributions of this work. In doing so, to the extent possible, we try to avoid unnecessary overcomplications, by means of visualizations and examples. Below, we give the prerequisite definitions.

Definition 3.1 (Confusion Matrix). *Given a dataset of k classes as $\{a_1,\cdots,a_k\}$,* confusion matrix *is the tuple $cm=\langle a_{ij}\rangle_{ij}\in\mathbb{N}^{k^2}$ of k^2 random variables that together describe the performance of a supervised, k-class classification model. Each random variable a_{ij} keeps the total count of the class a_i's instances which are classified as the class a_j's instance. We denote confusion matrices by cm.*

Definition 3.2 (Binary Confusion Matrix). *The* binary confusion matrix *is a confusion matrix with $k=2$. For convenience, it is denoted by $cm=\langle tp, fn, tn, fp\rangle$, where tp, fn, tn, and fp are the total counts of true positives (tp), false negatives (fn), true negatives (tn), and false positives (fp), respectively.*

We shall often, provided it leads to no confusion, use the term *confusion matrix* for both the confusion matrix and an instance of it. Also, we may drop the term 'binary' if its meaning can be inferred from the context.

Definition 3.3 (Relatively Identical Confusion Matrices). *For a given class-imbalance ratio $r\in[1,+\infty)$, two or more binary confusion matrices are called* relatively identical, *if they are identical independent of the sample size, i.e., when simplified to the form $\langle tpr, 1-tpr, r\cdot tnr, r(1-tnr)\rangle$.*

Note that the above-mentioned simplified form is nothing but the normalized confusion matrix, i.e., $\langle\frac{tp}{p},\frac{fn}{p},\frac{tn}{n},\frac{fp}{n}\rangle$. Therefore, not surprisingly, relatively identical confusion matrices are considered equivalent by most of the performance evaluation metrics. That is, a metric returns the same value for all such confusion matrices. However, relatively identical confusion matrices only account for a subset of all confusion matrices which are considered equivalent by a metric. Later on in Sections 5.1 and 5.3, we see the impact of this difference.

Definition 3.4 (Performance Evaluation Metric). *Let $\mathcal{CM}$ be the set of all confusion matrices. A* performance evaluation

metric *is a function* $\mu : \mathcal{CM} \rightarrow \mathbb{R}$ *with the following implications: for all* $cm_1, cm_2 \in \mathcal{CM}$, *(1) if* $\mu(cm_1) < \mu(cm_2)$ *then* cm_2 *is ranked, by* μ, *higher in performance, and (2) if* $\mu(cm_1) = \mu(cm_2)$ *then* μ *does not prefer one over the other.*

Although performance evaluation metrics can have unbounded ranges, in this study we only consider those with bounded ranges. Moreover, to have the same range across all metrics, we use a linear transformation to unify all ranges to [0,1] and thus $\mu : \mathcal{C} \rightarrow [0, 1]$.

4 Contingency Space

Our goal is to set up a geometrical setting in which an arbitrary confusion matrix can be represented as a single point. Since binary confusion matrices are 4-dimensional tuples (see Def. 3.2), a 4-dimensional space is needed to be able to directly map confusion matrices to unique points in this space. However, knowing that a large subset of confusion matrices are relatively identical (see Def. 3.3), with a fixed class-imbalance ratio we can reduce the dimensionality of the needed space to two. That is, each confusion matrix is represented by its tpr and tnr. Moreover, we would like this space to be endowed with the concept of distance metric so that the spatial information in this setting allows comparison of any pairs of confusion matrices independent of any pre-defined performance evaluation metrics. All these requirements lead us to the concept of metric spaces [19]. Recall that a *metric space* is a set X that is endowed with a metric d, denoted by (X, d). And $d : X^2 \rightarrow \mathbb{R}$ is a *metric* if, and only if, for all $a, b, c \in X$: (1) $d(a, b) \geq 0$, (2) $d(a, b) = d(b, a)$, and (3) $d(a, b) \leq d(a, c) + d(c, b)$. A metric that does not necessarily hold the third condition (triangle inequality) is called a *semimetric*. Using these tools, we introduce the contingency space.

Definition 4.1 (Base Contingency Space). *Given a class-imbalance ratio* $r \in [1, +\infty)$, *a* base contingency space *is a bounded semimetric space,* $\mathcal{C}_b^r = ([0, 1]^2, d)$ *where* d *is a semimetric, and each point in the space,* $(x, y) \in [0, 1]^2$, *represents all relatively identical confusion matrices of the form* $\langle y, 1-y, r \cdot x, r(1-x) \rangle$.

Note that in Def. 4.1, we use semimetrics so that a larger family of performance evaluation metrics can be defined in $\mathcal{C}_b^r$. We will discuss the benefits of this decision in Section 5.4.

It can be directly inferred from the definition of the base contingency space that the *perfect model*'s performance where $tp = p$ and $tn = n$, is mapped to the upper-right corner of this bounded space, i.e., at $(1, 1)$, and the central point, $(0.5, 0.5)$, is reserved for a *random-guess* model's performance where $tp = \frac{p}{2}$ and $tn = \frac{n}{2}$. These points are the special cases of a more general concept that we define next.

Definition 4.2 (Model Point). *Each point in a base contingency space* $\mathcal{C}_b^r$ *represents the performance of a binary classification model under the class-imbalance ratio* r. *These points are called* Model points, *or simply* points, *and are denoted by* p.

As the reader may have already noticed, one can convert the base contingency space to the well-known Receiver Operating Characteristic (ROC) space [20], [21], by replacing tnr (specificity) with fpr (1−specificity). Although the difference may seem negligible, only the geometrical setting of the base contingency space allows expansion of this space such that it addresses the multi-class problems. We discuss this in more details in Section 5.5.

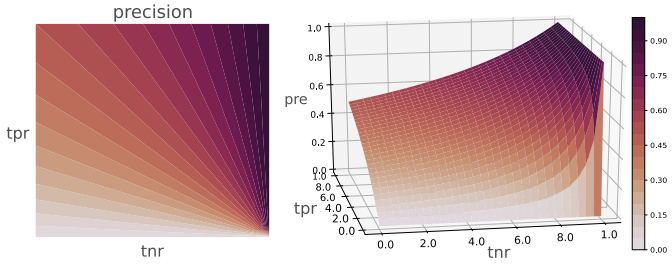


Fig. 1. The metric surface of precision ($\mathcal{S}^1_{\text{pre}}$) visualized in the contingency space. On the left, a contour plot is used to illustrate the surface, where the darker values represent higher precision and the contours are added to show the changes of the curvature. On the right, the surface's actual 3D view is shown to better illustrate the bivariate distribution function representing precision's values.

Now that we have the base contingency space, we can define the contingency space.

Definition 4.3 (Contingency Space). *A* contingency space *is a bounded semimetric space,* $([0, 1]^3, d)$, *expanded upon a base contingency space* $\mathcal{C}_b^r$. *The expansion is done by introducing the third dimension that represents the values returned by any performance evaluation metric. This space is denoted by* $\mathcal{C}^r$, *where* r *is the same imbalance ratio used in* $\mathcal{C}_b^r$.

The contingency space provides means for visualization of any binary performance evaluation metric. This can be done by generating unique surfaces for metrics, as defined in Def. 4.4.

Definition 4.4 (Metric Surface). *Suppose* μ *is a performance evaluation metric and* $\mathcal{C}^r$ *is a contingency space. A* metric surface *is a subspace of* $\mathcal{C}^r$ *on the set* $\{(x, y, \mu(p)); p = (x, y) \in \mathcal{C}_b^r\}$, *where* $\mathcal{C}_b^r$ *is the base contingency space of* $\mathcal{C}^r$. *A metric surface is denoted by* $\mathcal{S}_\mu^r$.

A metric surface, in fact, depicts the bivariate distribution function of all possible performances, with tpr and tnr as its independent random variables. As an example, the corresponding surface for precision in a contingency space is illustrated in Fig. 1. The one on the left is the contour plot of the precision's surface, and the one on the right is its actual $3D$ visualization with precision being represented on the z axis. Throughout this paper, to avoid obfuscated $3D$ plots, we visualize surfaces using their corresponding contour maps instead. The color tones in these plots represent the third dimension, i.e., the models' performance measured by a given metric. All such plots are generated using the Python plotting library, *matplotlib* v3.1.2 [22].

$3D$ surfaces often impose a heavy computational burden. This is, however, not the case for metric surfaces. To represent a metric surface, only a square matrix ($C_{l \times l}$) and an imbalance ratio r are needed. The number of rows/columns, l, of this matrix only determines the smoothness of surfaces for visualization, and visualization only. Therefore, l is a constant value, which results in the time complexity of calculating such a surface being $O(N^2)$ for the input size N. Moreover, each entry of this matrix, say c_{ij}, corresponds to a set of relatively identical confusion matrices where $tpr = \frac{i}{l}$

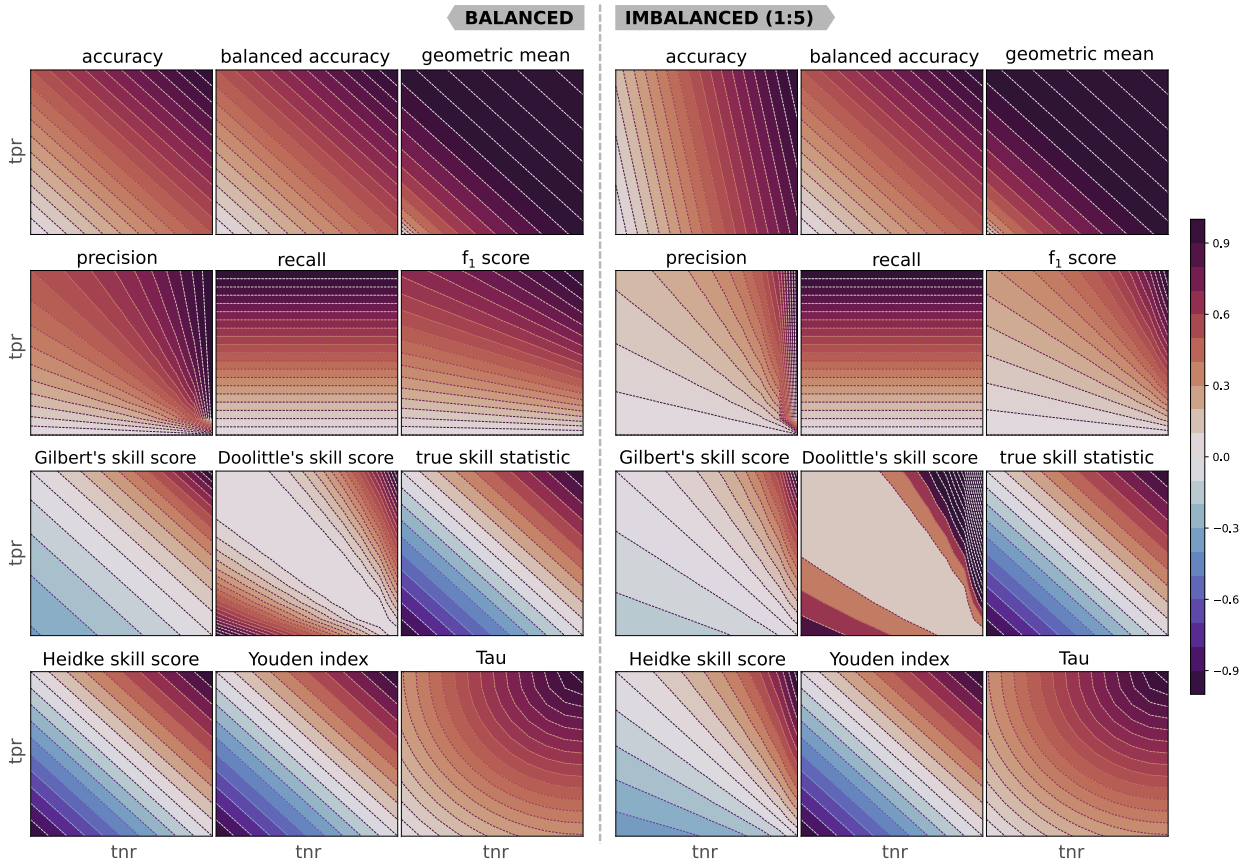


Fig. 2. The metric surfaces of 12 performance evaluation metrics (listed in Table 2) are visualized. The surfaces on the left are generated under the assumption of having a balanced dataset (therefore, in the contingency space $\mathcal{C}^1$), and for those on the right, an imbalance ratio of 1:5 is used (therefore, generated in $\mathcal{C}^5$). The juxtaposition of the surfaces on the two sides sheds light on the impact of class imbalance on metrics' behavior (as discussed in Section 5.1). The color scale on the right maps the interval $[-1, 1]$ to a spectrum of dark blue to dark red, respectively. The contours are drawn only to accent the curvatures of the surfaces and not to imply that the metrics form piecewise surfaces.

and $tnr = \frac{j \cdot r}{l}$. These confusion matrices are calculated independent of the actual values of the entries. In other words, the matrix itself, without its entries, represents the base contingency space. Therefore, the only variables needed for generating the points of a metric surface in the contingency space are the fixed number of entries of the matrix, i.e., l^2 quantities. This concludes a linear time complexity, $O(\varepsilon N)$ where $N = l^2$ and ε is the time needed for calculating the metric itself.

5 Applications of Contingency Space

So far we have laid the groundwork and introduced the contingency space and metric surfaces. Equipped with these tools we can approach the model evaluation challenge from a number of different angles. In this section, we present some of such applications and provide real-world use cases for each of them.

5.1 Analysis of Metrics

The contingency space is an intuitive concept because of its graphical representation. As the first application of this space, we use this graphical interface to analyze and compare performance evaluation metrics, and address the limitations we listed in Section 2. The metric surfaces of the 12 popular performance evaluation metrics listed in Table 2 are depicted in Fig. 2. The surfaces are generated under two assumptions; with balanced data (left) and imbalanced data with the ratio of 1:5, i.e., $r = 5$ (right). Below, we briefly discuss the insights these surfaces provide into the metrics.

- It quickly stands out that some of the plotted metrics reflect the changes in models' behavior in terms of only one class and obscure that for the other. For example, recall by definition disregards the fraction of incorrect classifications, and this is captured by the horizontal patterns on its surface.
- The curvatures of a metric's surface show what family of models are seen (by the metric) as identical in terms of their classification performance. For instance, on the surface of accuracy all model points lying diagonally along the same contours are considered 'equally good'. These families have been identified before as 'iso-performance lines' in [23] and later on in [24] on ROC space. To the best of our knowledge, however, they were never used to compare performance metrics themselves. This concept is a very important , and often overlooked, realization that it is our chosen metrics which equate some models, and this is far from the models' confusion matrices being identical or even relatively identical. To see the true similarities of models' performance one should compare their confusion matrices, or equivalently and more intuitively, their model points on the corresponding surfaces in

the contingency space. The lack of understanding of the bivariate distribution of the models, i.e., metrics' surfaces, is the primary cause of such oversimplifications.

- Curvatures of surfaces give us another interesting tool to differentiate some metrics from the others. Some metrics form surfaces with a constant curvature, such as accuracy, balanced accuracy, recall, true skill statistic or Youden's j index, and Tau. The curvature in other surfaces, however, vary at different points. Examples of such surfaces are precision, the f_1 score, Gilbert's skill score, and Doolittle's skill score. This is an important distinction and users should have a good justification for choosing such curvatures for the evaluation of their models. In rare-event forecast domains, for instance, it is often significantly more important to avoid a miss (failing to predict the occurring event) than a false alarm (a false prediction of an event). Two metrics that are popular in such a rare-event forecasting domain are Gilbert's skill score or Heidke skill score [25]. By comparing their corresponding surfaces as depicted in Fig. 2, with and without the class-imbalance assumption, it is evident that they both take into account the class imbalance of data inherited from the scarcity of rare events. Both of these metrics emphasize on the high tprs, of course, but more importantly, on the much higher tnrs. This unequal weighting favors models with a lower chance of a miss (i.e., more reliable on the *all-clear* state). This might be a fair justification for using such surfaces but perhaps not sufficient as the curvature differences yet seek further justification.
- For a metric to better align with a task's objective, its curvature may call for some adjustments. This is important because the cost of a miss or a false alarm changes from one problem to another and the listed performance evaluation metrics are not cost sensitive. Using these surfaces, incorporation of the costs in metrics' formula can be directly examined. Moreover, while looking at these surfaces immediately triggers an array of such questions (about the degrees of different curvatures; their usefulness and impact), the statistical reasoning which originally led to the metrics' definitions do not encourage such arguments.
- As pointed out earlier, some metrics are unbiased to the class imbalance of the data, such as balanced accuracy, geometric mean, recall, true skill statistic or Youden's j index, and Tau. The surfaces corresponding to such metrics remain unchanged in both scenarios, with the balanced and imbalanced data. Others, however, warp proportionally to the imbalance ratio. It is necessary to note that the imbalance ratio used in Fig. 2 (right) is only 1:5. In many real-world examples, the imbalance ratio is expected to be much higher. To put these numbers into context, a recently released benchmark dataset presents an unsurprising 1:95 imbalance ratio of positive to negative instances [26]. Such an extreme scarcity should raise questions about the effectiveness of the popular metrics in the relevant domains, such as true skill statistic and Heidke skill score [27]. Interestingly, true skill statistic is completely insensitive to the imbalance ratio, which makes it an appropriate metric for comparison of models with varying imbalance ratio. Heidke skill score, however, despite its usefulness becomes a progressively stricter metric as the imbalance ratio increases. This renders comparison of models' performance meaningless if the imbalance ratio is not fixed across models. This strictness is visualized by the red region that has significantly shrunk (pushed to the right of the contour map) as the imbalance ratio increased to 1:5.

Without the geometrical setting of contingency space, it may not be as intuitive to deduce such insights from the abstract definitions of the performance evaluation metrics. This simplicity in highlighting the differences and similarities, and raising novel critiques about metrics exhibits the visual power of the proposed space. That said, the visual strength is not the only application of contingency space. In the following sections we dive deeper in other ways this space can provide insight into model evaluation challenges.

5.2 Learning Path

The iterative learning process of an algorithm is often analyzed by monitoring either the loss function of the classifier or one or more performance evaluation metrics. But these two approaches do not necessarily account for the same objectives; the utilized loss function can help diagnose the optimization weaknesses such as overfitting or convergence, while the performance evaluation metrics measure the appropriateness of the trained model for a specific application. Given that model points in the contingency space provide context and multi-dimensional view of performance, tracking the learning process of a classification algorithm using such points can give us unique and intuitive insights into what happens during the training phase. Below, we define *learning path*; a path that an algorithm takes, in terms of its performance, as it learns the discriminating features.

Definition 5.1 (Learning Path). *Given a classification algorithm, let $\mathcal{C}^r$ denote a contingency space, and μ denote a performance evaluation metric. Suppose $(m_i)_{i=1}^n$ is a sequence of model points in the corresponding base, $\mathcal{C}_b^r$, where the i-th element is obtained by evaluating the algorithm at the end of the i-th epoch of an n-step train-and-validation process. We call this sequence a* learning path*, and it lies on the surface $\mathcal{S}_\mu^r$. This learning path is denoted by $\mathcal{L}_\mu((m_i)_{i=1}^n)$.*

Of course, the learning path of an arbitrary classification algorithm is not unique. Two trials of training of an algorithm, with a fixed setting and performed on the same dataset, can yield two different learning paths. This is because of the non-deterministic nature of many learning algorithms. Therefore, of interest are the patterns and statistics extracted from the paths and not the exact sequence of points. Analysis of such patterns opens up several interesting avenues. As a proof of concept, in the following we present one application of analyzing the learning path.

Our empirical analysis of the learning path leads us to propound a hypothesis that there is a correlation between the "complexity" of learning path and the "struggle" of

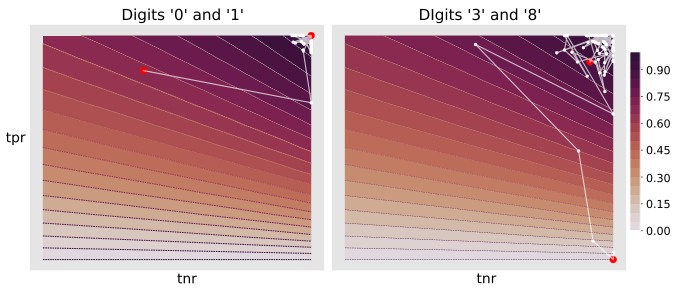


Fig. 3. Two learning paths of a convolutional neural network on subsets of the MNIST dataset are compared. On the left, the classifier is learning to distinguish between the digits '0' and '1' (problem A), while on the right, it does the same but on the digits '3' and '8' (problem B) which have more similar structures. The difference between the two learning paths can be used as a proxy to verify that problem A is an easier classification task for CNN than problem B.

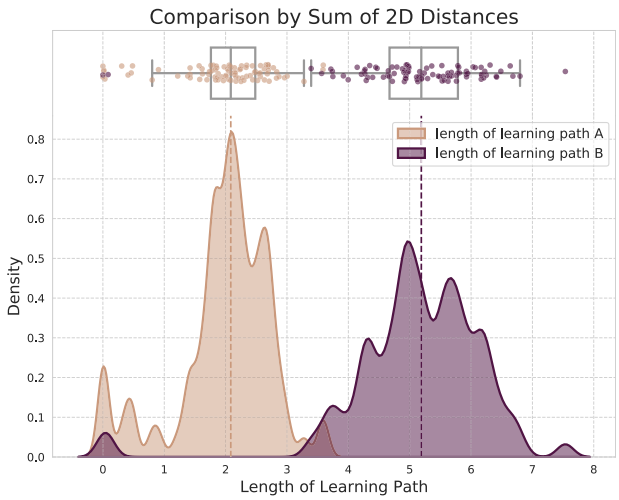


Fig. 4. Distributions of the length of learning paths obtained by training a CNN algorithm on MNIST hand-written digits, in two scenarios: (1) trained on digits '0's and '1's (left), and (2) trained on hand-written digits '3's and '8's. When repeating this experiment 100 times, the Kolmogorov-Smirnov test returns a very small p-value indicating that the two distributions are not similar.

algorithms in learning discriminating patterns. To test this hypothesis we design an experiment where two classification problems, one more difficult than the other, are compared using the learning paths of a classification algorithm. In order to obtain an evident distinction in the difficulty levels of the problems, we use one of the most known computer vision datasets, namely the MNIST dataset of hand-written digits [28], [29]. Although the dataset is now considered only as the "Hello World" of Pattern Recognition and Computer Vision[1], we believe it serves our purpose too well to be disregarded, as we explain in the following.

For this experiment, we use two subsets of the MNIST dataset, one made up of the digits '0' and '1', and the other, of the digits '3' and '8'. The hand-written digits of the former subset has more distinct patterns than the digits of the latter subset; hand-written '3's can be easily mistaken as '8's, and vice versa. Therefore, we expect that the learning process of a classification algorithm to be meaningfully different on these two problems, and be reflected in their learning paths. Let the letters A and B denote these two classification problems, '0' & '1' and '3' & '8', respectively.

Regarding the classification algorithm, we put together a vanilla Convolutional Neural Network (CNN) using the PyTorch framework [30]. For simplicity, our CNN has only 4 hidden layers, 2 max-pooling layers, and a softmax activation layer, with the pre-set hyper-parameters (learning rate of 0.01 and momentum of 0.9). We run a 100-step train-and-validation of this classifier on each of the two subsets of MNIST separately.

A pair of learning paths obtained from training our CNN for the classification problems A and B are depicted in Fig. 3. In this example, we use the metric surface of the f_1 score in the contingency space $\mathcal{C}^1$ to provide context. It is easy to see that the learning path corresponding to problem A is shorter than that of B. This visual observation hints at the validity of our hypothesis; problem A is easier for our vanilla CNN, i.e., CNN can more easily and quickly find some powerful discriminative features when dealing with problem A, compared to when it deals with problem B.

To verify whether the difference in the learning paths of the classifier is statistically meaningful we consider the length of each learning path as our statistic and compare its distributions corresponding to problems A and B. Our null hypothesis is that there is no significant difference between the two distributions. We repeat this experiment 100 times and use the non-parametric, Kolmogorov-Smirnov test [31], [32] to assess the null hypothesis. The low p-value of $1.68e-47$ allows us to confidently reject the null hypothesis, indicating that the two distributions are indeed different. This is more clearly depicted in Fig. 4. Note that the box plots (within $\pm 1.5IQR$) have no overlap.

One can also inspect the steps in a learning path. The learning path of CNN trained on problem B (the right plot in Fig. 3) starts from an all-negative model, and in only a few steps goes all the way to a model with a very high true-positive rate and a ~ 0.5 true-negative rate. Right after that, the model moves along a contour line to the right edge of the contingency space and achieves a 1.0 true-negative rate and ~ 0.5 true-positive rate. Although this move may appear as a significant change, the f_1 score's surface reveals the opposite; the corresponding confusion matrices are almost relatively identical (recall Def. 3.3). And lastly, the model is stopped at a sub-optimal performance (compared to its prior performance) at the end of the 100-th epoch. This is a clue that may hint towards an overfitting issue which seeks further investigation.

At the beginning of this subsection, we brought up the idea of monitoring the loss function. The learning process in Fig. 3 might bear some resemblance to the optimization process of a model's loss function, that is often visualized as a point moving over a surface formed by a loss function, towards the 'global' minimum. But it is important to note that in most cases, the loss functions are extremely high-dimensional and only low-dimensional projections of them are possible to be visualized. In such settings, tracking the learning process using the contingency space, as we proposed in this section, can play an important role in the analysis of the learning process. Of course, such an analysis does not provide any direct insight about the optimization of the loss function but it sheds light on how the

1. See the impressive success of several studies in classification of the digits of MNIST reported here: http://yann.lecun.com/exdb/mnist/

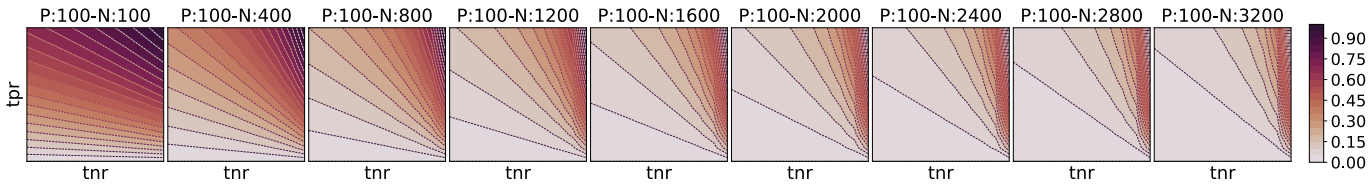


Fig. 5. Changes in the metric surface of the f_1 score as the imbalance ratio changes linearly from 1:1 (far left) to 1:32 (far right).

training of the model progresses in terms of the correct and incorrect classifications of instances. For an example of comparing different metrics using the cost/error space (limited to 2-D spaces) see [33].

5.3 Measuring Class-Imbalance Sensitivity

Performance metrics may or may not be sensitive to the class-imbalance ratio. Those that are sensitive give a more realistic picture of performance under different class-imbalance ratios. Others completely disregard the impact of class imbalance on evaluation. However, it is only meaningful to use the latter group for comparing models' performances in spite of the different imbalance ratios they are validated on. To deal with this duality in choosing an appropriate metric, to the best of our knowledge, no methodological approach has yet been proposed to measure the degree of this sensitivity (a.k.a. 'skew-sensitivity') to provide information for a better decision making process. Instead of a binary approach, a sensitive metric with a low sensitivity rate may still be acceptable for an evaluation task. Using the concepts introduced above we can build the tools we need to address this issue. Let us first define what exactly we mean when we say a metric is insensitive to the class imbalance.

Definition 5.2 (Imbalance Agnostic). *For an instance of a confusion matrix cm_0, let $\mathcal{CM}_{r_1}$ and $\mathcal{CM}_{r_2}$ be the sets of all confusion matrices which are relatively identical to cm_0, with the class-imbalance ratios r_1 and r_2 ($r_1 \neq r_2$), respectively. We say a metric μ is* imbalance agnostic, *if for any $cm_1 \in \mathcal{CM}_{r_1}$ and $cm_2 \in \mathcal{CM}_{r_2}$, $\mu(cm_1) = \mu(cm_2)$.*

Recalling Def. 4.4, a metric surface is simply a function that maps a confusion matrix (a model point in the base contingency space $\mathcal{C}_b^r$) to a point in the contingency space $\mathcal{C}^r$. Directly deduced from Def. 5.2, the surfaces corresponding to an imbalance-agnostic metric should remain unchanged for all imbalance ratios $r \in [1, +\infty)$. And if it does not, like several examples in Fig. 2, the metric is not imbalance agnostic. Using this geometrical setting, and the fact that metric surfaces warp proportionally to the imbalance ratio (e.g., see Fig. 5), we can quantify the imbalance sensitivity.

Definition 5.3 (Imbalance Sensitivity). *The sensitivity of a metric μ to the class-imbalance ratio r is measured by the volume confined between the two surfaces $\mathcal{S}_\mu^1$ and $\mathcal{S}_\mu^r$ ($r \neq 1$). This volume which is a function of the imbalance ratio is called* imbalance sensitivity *of μ to the imbalance ratio r, and is denoted by $\mathcal{IS}_\mu(r)$.*

This volume is illustrated in Fig. 6. Two surfaces corresponding to the f_1 score metric are shown with the imbalance ratios $r=1$ (surface on top) and $r=32$ (surface at bottom). The volume confined between them is the proxy used in Def. 5.3 for measuring the imbalance sensitivity of the f_1 score to the imbalance ratio 32, i.e., $\mathcal{IS}_{f_1}(32)$. In Fig. 7, several metrics' sensitivity is plotted against the imbalance ratio. Note that the upper bound for metrics' sensitivity is the volume of the contingency space, i.e., $\lim_{r\to+\infty} \mathcal{IS}_\mu(r) \leq 1$.

Note that the two surfaces in Def. 5.3, $\mathcal{S}_\mu^1$ and $\mathcal{S}_\mu^r$, may occasionally intersect. This does not cause any issues in calculating the confined volume as we use the Riemann Sum to measure it. This is explained below.

The volume in Def. 5.3 can be calculated in linear time. Recall that, as mentioned at the end of Section 4, a metric surface is represented by a fixed-size matrix, $C_{l\times l}$. Therefore, in practice, the volume confined between two such surfaces

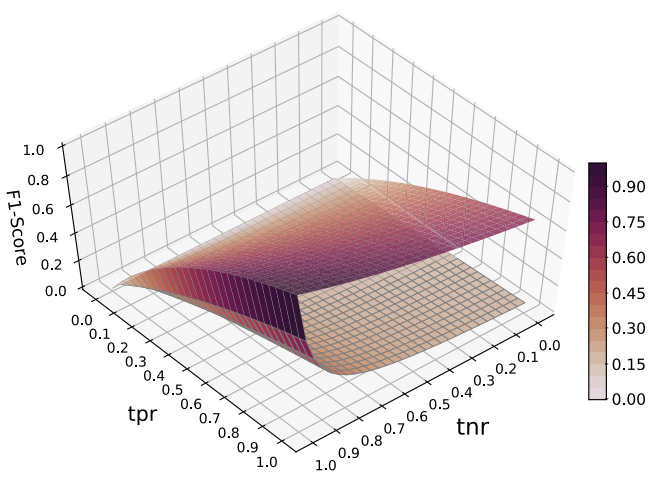


Fig. 6. The metric surfaces of f_1 score for two class imbalance ratios; 1:1 (the top surface) and 1:32 (the bottom surface). The volume confined between the two metric surfaces is used to define the imbalance sensitivity, $\mathcal{IS}_\mu(r)$, a proxy to quantify a metric's sensitivity to class imbalance.

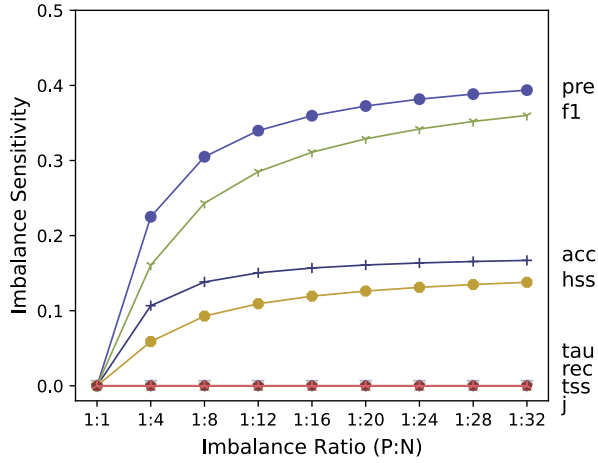


Fig. 7. The imbalance sensitivity $\mathcal{IS}_\mu(r)$ for 8 classification evaluation metrics (listed in Table 2) are compared as the positive-to-negative class-imbalance ratio changes linearly from 1:1 to 1:32. Metrics such as Tau (tau), recall (rec), true skill statistic (tss), and Youden J index (j) are imbalance agnostic while others are impacted logarithmically.

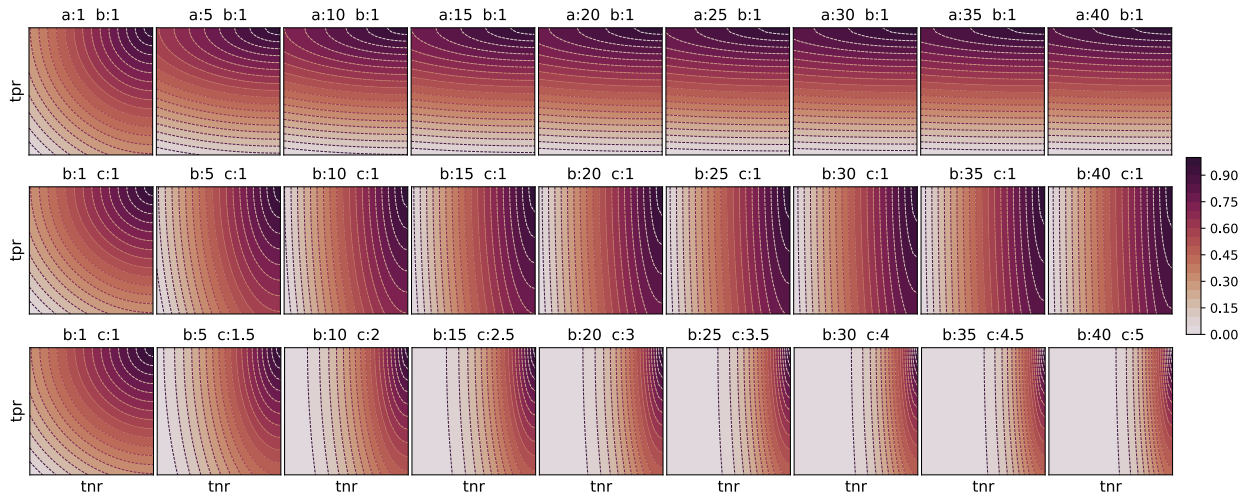


Fig. 8. Customization of weighted Tau through its 3 weight scalars. This prepares this metric for objectives of different tasks.

is nothing but the sum of absolute pairwise differences of their corresponding matrices. Let C^1 and C^r denote two matrices representing two metric surfaces $\mathcal{S}^1_\mu$ and $\mathcal{S}^r_\mu$. Also, let $c^1_{ij}(\mu)$ and $c^r_{ij}(\mu)$ denote the entries of the matrices. Then, this volume can be calculated by $\sum_{i=1}^{l}\sum_{j=1}^{l}|c^1_{ij}(\mu)-c^r_{ij}(\mu)|$. Thus, the computation time is $O(N)$ where $N=l\cdot l$, i.e., the total number of entries of a fixed-size matrix.

It is worth mentioning that this sum is known as the Riemann Sum [34] and gives an approximation of the area (volume) under a curve (surface). But the continuous surfaces depicted in this study are only for visualization purposes and the matrices are the actual mathematical objects representing metric surfaces. Therefore, the Riemann Sum in this case measures the exact sensitivity of metrics and not its approximation.

5.4 Engineering Custom Performance Metrics

So far, we have not used the semimetrics that contingency spaces are endowed with. But the primary reason for defining contingency spaces as metric spaces was to use these internally defined functions as performance evaluation metrics. This possibility opens new windows towards introducing task-specific metrics. In the following, we first define a performance evaluation metric using these functions and then show its major strength which is its customizability.

Recall the special model points in the base contingency space: the perfect and the random-guess model points. We can use the distance between an arbitrary model point and either of these two points as a proxy for models' performance; model points closer to the perfect model point are ranked higher in terms of performance. Such measures which quantify the performance relative to a baseline are often called skill scores [35]. Def. 5.4 introduces Tau using the perfect model point as the baseline.

Definition 5.4 (Tau). *Given a base contingency space $\mathcal{C}^r_b$ and the euclidean distance function d, the semimetric* Tau, $\tau(p)=1-\frac{d(p,p_{\text{perfect}})}{\sqrt{2}}$, *quantifies an arbitrary model's performance (p) by measuring the normalized, euclidean distance between its corresponding model point p and the perfect model point p_{perfect} in $\mathcal{C}^r_b$.*

Note that altering the perfect model point with the random-guess model point as the baseline in Def. 5.4 also provides valuable insight in some applications. The Heidke skill score (hss) is such a statistic; it measures models' performance in terms of their success relative to random guess [18]. But the challenge is that there might be more than one point that could represent random-guess models, and this depends on the metrics' definition. Whereas regardless of the metric of choice, there is only one perfect classification and that is represented by a single point, i.e., (1,1).

As is depicted in Fig. 2, like any other performance evaluation metric, Tau can also be assigned a unique surface in the contingency space. Unlike other metrics, however, this metric is defined directly inspired by the geometrical settings of the base contingency space; Tau measures the performance improvement a model needs for correctly classifying all instances. Note that in Def. 5.4, τ subtracts the normalized distance from 1 so that the metric is consistent with the common higher-the-better implication of Def. 3.4.

This geometrical intuition encourages us to investigate customizability of such metrics. To adjust Tau for problems with unequal classification costs for different classes, we can freely contort its corresponding surface, i.e., the distribution of models' performance. To this end, Def. 5.5 defines the weighted Tau.

Definition 5.5 (Weighted Tau). *Given a base contingency space $\mathcal{C}^r_b$, the semimetric* weighted Tau, $w\tau(p)=\nu-\frac{\nu}{\sqrt{2}}d_\omega(p,p_{\text{perfect}})$ *is the weighted version of Tau for an arbitrary model point p, where $\nu\in\mathbb{R}$ and $\omega=(\omega_x,\omega_y)\in\mathbb{R}^2$ are the weights, and $d_\omega=(\omega_x(1-tnr)^2+\omega_y(1-tpr)^2)^{\frac{1}{2}}$*

By adjusting the weights along one or two axes of the base contingency space, i.e., tuning ω_x and ω_y of $w\tau$, the spread and shape of model points' distribution can be adjusted and consequently, it can better fit the task-specific classification costs. Generic examples of such modifications are depicted in Fig. 8. In the first row, increasing ω_x results in lower kurtosis along x axis and higher kurtosis along y axis. Conversely, in the second row, ω_y is increased and the outcome is the opposite. In the third row, the effect of changing ν in combination with ω_y is shown. ν allows magnification of the impact.

The real practicality of weighted Tau manifests itself in evaluating the cost-effective learning algorithms. These algorithms are essential for problems in which the (estimated)

costs of classification errors are known and unequal among classes. The algorithms' cost functions are designed to take into account per-class costs. Most performance evaluation metrics, however, are not. In a binary case, knowing that the cost of a miss (fn) is k times the cost of a false alarm (fp), weighted Tau takes this into account by setting ω_x to k (see the second row of Fig. 8).

To give a practical example, consider the impact of solar storms on transpolar flights. Airlines constantly monitor strong solar storms and upon positive forecasts reroute their flights to keep passengers and the crew safe from dangerous radiations. One of the best known indicators used for forecasting of solar flares (that cause solar storms) is the changes in the magnetic flux (x) of the active regions of the Sun. The historical observations give us the likelihood of active regions to flare, or not, within a fixed time window in future. Having the two probability density functions of flaring (f_F) and non-flaring (f_N) active regions, one can define the total error term, E, of forecast by finding the optimal decision threshold (x_0) of the predictor, magnetic flux. More specifically, the error can be calculated as $E = \int_{-\infty}^{x_0} f_F\, dx + \int_{x_0}^{\infty} f_N\, dx = E_{fn} + E_{fp}$. The optimal threshold is thus the value of x for which $\frac{dE}{dx_0} = 0$. This, however, does not take into account the significant difference between the actual cost of a miss and a false alarm. Economically, the cost of rerouting a flight in the absence of any solar storm (c_{fp}) is significantly less than that of exposing hundreds of passengers and the crew to high degree of radiation (c_{fn}), which could be costs of lawsuits and/or payouts, not to mention the damage to the reputation of the airlines' corporate identity, and most importantly, the irreparable damage to passengers' health. Knowing the costs, the error term can be updated to $E_c = c_{fn} \cdot E_{fn} + c_{fp} \cdot E_{fp}$. These per-class costs terms can easily be incorporated in weighted Tau as well, and form the surface that precisely reflects the specific objectives of this task. The benefits of using weighted Tau, over E_c, are all those mentioned in Section 2.

5.5 Evaluation of Multi-Class Problems

In Section 4, we briefly touched on the advantage of using tnr as the x axis of the base contingency space, unlike ROC space where the x axis represents $1-tnr$. Here, we elaborate on how this change allows representation of multi-class model points in the base contingency space. This is due to the geometrical setting of the base contingency space that puts the perfect model point at (1,1), farthest from the origin and away from either of the axes, whereas in ROC space, it is located at (0,1), lying on the y axis. Consequently, when base contingency space is expanded to higher dimensions, the perfect model point keeps its unique location, i.e., farthest from the origin. Therefore, this point can still be used as the reference point (i.e., the baseline model) for measuring models' performance on multi-class problems. With this in mind, we can define the multi-class base contingency space and the multi-class Tau.

Definition 5.6 (Multi-Class Base Contingency Space). *Given k classes, suppose $r = (r_1, r_2, \cdots, r_k) \in [1, +\infty)^k$ is a tuple of class-imbalance ratios where $r_i = \frac{|c_i|}{|c_1|}$ in which $|c_i|$ is the sample size of class c_i. A* multi-class base contingency space *is a bounded semimetric space, $\mathcal{C}_b^r = ([0,1]^k, d)$ where d is a semimetric. Each point in this space, $(x_1, x_2, \cdots, x_k) \in [0,1]^k$, represents all relatively identical confusion matrices of the form $\langle x_1, 1-x_1, r_2 \cdot x_2, r_2(1-x_2), \cdots, r_k \cdot x_k, r_k(1-x_k)\rangle$.*

Having the multi-class base contingency space defined, we can now expand the definition of Tau and weighted Tau to multi-class performance evaluation metrics.

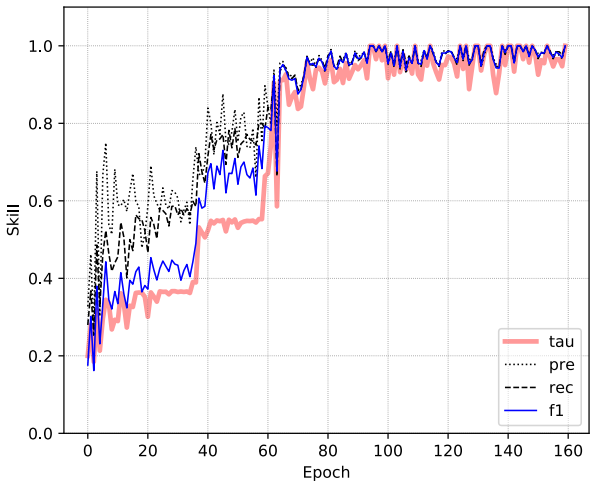
Definition 5.7 (Multi-Class Tau). *Given a k-class base contingency space $\mathcal{C}_b^r$, the semimetric* multi-class Tau, *$\tau(p) = 1 - \frac{1}{\sqrt{k}} d(p, p_{\text{perfect}})$, quantifies an arbitrary model's performance by measuring the normalized euclidean distance between its corresponding model point p and the perfect model point p_{perfect} in $\mathcal{C}_b^r$.*

Definition 5.8 (Weighted Multi-class Tau). *Given a k-class base contingency space $\mathcal{C}_b^r$, the semimetric* weighted multi-class Tau, *$w\tau(p) = \nu - \frac{\nu}{\sqrt{k}} d_\omega(p, p_{\text{perfect}})$ is the weighted version of multi-class Tau, where $\nu \in \mathbb{R}$ and $\omega = (\omega_1, \omega_2, \cdots, \omega_k) \in \mathbb{R}^k$ is the weight vector, and $d_\omega = (\sum_{i=1}^{b} \omega_i (1 - tc_i r)^2)^{\frac{1}{2}}$ in which $tc_i r$ is the tpr for the class c_i.*

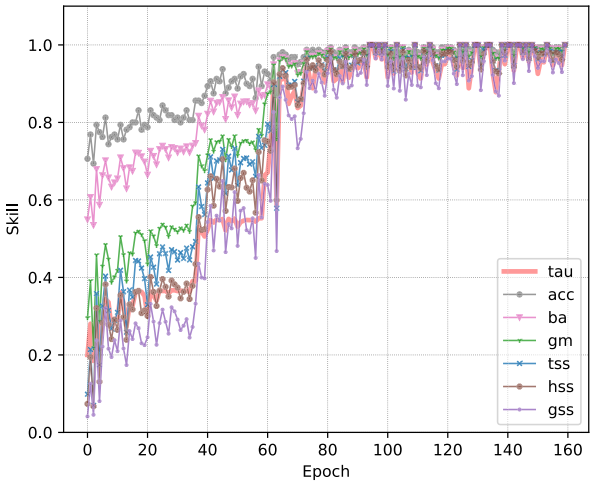
To give an example for multi-class Tau, we use our vanilla CNN introduced in Section 5.2 and track its performance during the training process on the MNIST dataset. This time, we train on 1000 instances of hand-written digits, equally distributed among 5 classes, the digits '1' through '5'. To put the results in context, we compare it with other previously-discussed metrics (see Table 2). Unlike Tau, none of those metrics have a built-in, multi-class evaluation capability. Therefore, we use macro averaging technique which is the most popular solution for making use of binary metrics for non-binary classification problems. In macro averaging, the performance is first measured for each class (as the positive class) against others (as the negative class) and then averaged across all classes. Our motivation for using this technique is rooted solely in its popularity, otherwise, we are aware of their limitations for multi-class evaluation [36].

The 5-class comparison of those metrics is depicted in Fig. 9. We broke down the results into two plots for a better visibility. In both plots, the thick, red line represents the Tau's performance. Interestingly, tracking of the model's performance by Tau for the second half of the epochs is quite similar to that by the other metrics; the general trends, as well as the small fluctuations, are very similar. For the first half of the epochs, however, Tau shows lower performance on average, with three distinct 'steps'. Such a stepwise pattern is not captured by all other metrics. Within these steps the model points seem to have moved along the contours (iso-performance lines) of Tau's surface, hence no real improvements; between the steps, on the other hand, the model points seem to have moved rather perpendicular to the contours of the Tau's surface, hence the significant improvements reported by Tau.

It is important to note that the MNIST dataset has 10 classes but we confined our experiment to 5. This is because the euclidean distance used in Tau does not perform well in high dimensional spaces [37]. While choosing the right distance metric for high dimensional spaces has always been a challenge, what matters is that Tau (and its variants) can be defined with any semimetric. One can decide on the most appropriate distance metric by carefully studying the specific



(a) Tau versus precision (pre), recall (rec), and f_1 score (f1).



(b) Tau versus accuracy (acc), balanced accuracy (ba), geometric mean (gm), true skill statistic (tss), Heidke skill score (hss), and Gilbert's skill score (gss).

Fig. 9. Comparison of Tau with the metrics listed in Table 2 on evaluation of CNN's learning process on a multi-class dataset.

characteristics of their data, e.g., the distribution of data points, presence of outliers, etc.

6 Discussion, Conclusion, and Future Work

6.1 Contingency Space Versus ROC Space

As we mentioned before, base contingency space is, to a large degree, similar to the ROC space, while they bear significant differences as well. In the following, we discuss the similarities and distinctions:

1) Base contingency and ROC spaces are topologically identical in their two-dimensional bases, but not in higher dimensions. ROC in its original setting cannot be extended to high dimensions (see [15], [38], [39], [40] for a few other approaches) while by altering *fpr* (x axis of ROC) with *tnr*, base contingency space allows this expansion. We discussed this important difference in Section 5.5 and provided an example to show its effectiveness.
2) The similarities between the two spaces are advantageous; the contingency space preserves all the important characteristics of the ROC space and their interpretations while it adds to its applications. All the extensively studied concepts such as analysis of the ROC curves, comparison of different methods for computing the AUC, iso-performance lines and their slopes, slope of the tangent lines on the curves, and all others are still completely valid in the base contingency space, with no change or some minor modifications. For example, the iso-performance lines in ROC space are horizontally mirrored in the base contingency space, therefore the angle α should be adjusted to $\pi - \alpha$.
3) Additionally, the base contingency space takes into account the class-imbalance ratio. This is an important realization that is entirely disregarded in the ROC space and made it susceptible to variance in the class-imbalance ratios between different experiments.
4) Although the perfect model in the two spaces may be mapped to different locations, τ and distance-from-(0,1) (used in ROC space) are topologically identical. They form identical surfaces (only mirrored) and both are class-imbalance agnostic.
5) Despite the similarities between the two concepts, the way contingency space is defined provides a more intuitive understanding of this space; the contingency space is a space in which each point represents a family of (relatively identical) confusion matrices. This degree of intuitiveness is not evident in the ROC space which is defined as a mapping of true-positive rate and false-positive rate.
6) The ROC space, to the best of our knowledge, was never used for analysis and comparison of different metrics. This addition makes the contingency space to be used as a framework for choosing appropriate metrics for different tasks.

6.2 Conclusion and Future Work

We reviewed six main limitations of the performance evaluation metrics used for evaluation of supervised models. They are one-dimensionality, lack of context, lack of intuitiveness, uncomparability, binary restriction, and uncustomizability of metrics. To remedy these limitations, we introduced, and mathematically defined, a number of new concepts based on a bounded semimetric space, called *contingency space*. Every point in contingency space can be decoded to a family of *relatively identical* confusion matrices and therefore, represents a model's performance. We showed that using this concept, a given metric can be visually analyzed as a surface in this space independent of the unique characteristics of the data used and the models trained. We named it a *metric surface*. We presented another application for contingency space by analyzing models' *learning path*s and the complexity of such paths. Using this idea, we tested a hypothesis that whether classification of the hand-written digits '0' and '1' is easier than that of the digits '3' and '8', due to the more similar patterns evident among the instances of the latter group. We further showed that metrics' sensitivity to class imbalance is proportional to the degree of which their corresponding surfaces warp as a function of the imbalance ratio. This let us introduce the concept of *imbalance sensitivity*, a criterion to qualitatively and quantitatively guide researchers in choosing the right metric for their specific problems. Defining the contingency space as

a semimetric opened the door for introducing new and customizable metrics which can be adjusted to the misclassification costs. In this direction, we introduced *Tau*, and *weighted Tau* as a cost-sensitive metric. Lastly, we showed that because of the unique geometrical setting of the *base contingency space*, custom metrics such as Tau can be easily extended for multi-class evaluation. Therefore, we introduced *multi-class base contingency space* and *multi-class Tau*.

Contingency space is an intuitive concept that we believe opens the door to several new avenues that we are interested in exploring. The following avenues are of our primary interest: We would like to further investigate knowledge extraction from the learning paths about the models; the algorithms, their cost functions, optimizers, and discriminative power, as well as signs of overfitting, and their robustness. Such analyses about data are also equally important. The classical classification complexity measures often focus on the characteristics of the data, such as the separability of classes, overlap of some statistical features, and uniformity and normality of manifolds [41]. A focus on models' learning path for understanding more about the data is a different angle and may shed light on this family of problems. Furthermore, in regard with the multi-class evaluation of models, we would like to experiment with other semimetrics that are more appropriate for high-dimensional spaces. As mentioned in Section 5.5, we limited our multi-class experiment to 5 classes because we were not satisfied with the results obtained by using euclidean distance for computing multi-class Tau on the 10 classes of the MNIST. While dealing with high-dimensional spaces has always been a challenge, it is critical to note that the distribution of points in the contingency space endowed with a semimetric such as Tau is independent of data and model. This makes the search for an appropriate metric easier.

References

[1] K. Pearson, *On the Theory of Contingency and its Relation to Association and Normal Correlation; On the General Theory of Skew Correlation and Non-Linear Regression*. Cambridge, U.K.: Cambridge Univ. Press, 1904.

[2] J. P. Finley, "Tornado predictions," *Amer. Meteorological J. A Monthly Rev. Meteorol. Allied Branches Study (1884–1896)*, vol. 1, no. 3, 1884, Art. no. 85.

[3] A. H. Murphy, "The finley affair: A signal event in the history of forecast verification," *Weather Forecasting*, vol. 11, no. 1, pp. 3–20, 1996.

[4] N. Science, T. C. U. Networking, I. T. Research, and D. Subcommittee, "The national artificial intelligence research and development strategic plan," United States, Executive Office of the President, Nov. 2016. [Online]. Available: https://www.hsdl.org/?view&did=796343

[5] N. Science, T. C. U. Networking, I. T. Research, and D. Subcommittee, "The National Artificial Intelligence Research and Development Strategic Plan: 2019 Update, A Report by the Select Committee on Artificial Intelligence of the National Sciences & Technology Council," United States, Executive Office of the President, Jun. 2019. [Online]. Available: https://www.hsdl.org/?view&did=831483

[6] P. R. Cohen and A. E. Howe, "How evaluation guides AI research: The message still counts more than the medium," *AI Mag.*, vol. 9, no. 4, pp. 35–43, 1988. [Online]. Available: http://www.aaai.org/ojs/index.php/aimagazine/article/view/952

[7] K. Wagstaff, "Machine learning that matters," in *Proc. 29th Int. Conf. Mach. Learn.*, 2012. [Online]. Available: http://icml.cc/2012/papers/298.pdf

[8] Z. C. Lipton and J. Steinhardt, "Troubling trends in machine learning scholarship," *ACM Queue*, vol. 17, no. 1, 2019, Art. no. 80.

[9] H. Kerner, "Too many AI researchers think real-world problems are not relevant," Aug. 2020. [Online]. Available: https://www.technologyreview.com/2020/08/18/1007196/ai-research-machine-learning-applications-problems-opinion/

[10] G. Marcus, "Deep learning: A critical appraisal," 2018, *arXiv: 1801.00631*.

[11] J. Dunietz, "The field of natural language processing is chasing the wrong goal," 2020. [Online]. Available: https://www.technologyreview.com/2020/07/31/1005876/natural-language-processing-evaluation-ai-opinion/

[12] N. Chinchor and B. Sundheim, "MUC-5 evaluation metrics," in *Proc. 5th Conf. Message Understanding*, 1993, pp. 69–78.

[13] A. Hanssen and W. Kuipers, "On the Relationship Between the Frequency of Rain and Various Meteorological Parameters." (With Reference to the Problem of Objective Forecasting.). De Bilt, Netherlands: Koninklijk Nederlands Meteorologisch Instituut, 1965.

[14] W. J. Youden, "Index for rating diagnostic tests," *Cancer*, vol. 3, no. 1, pp. 32–35, 1950.

[15] D. J. Hand and R. J. Till, "A simple generalisation of the area under the ROC curve for multiple class classification problems," *Mach. Learn.*, vol. 45, no. 2, pp. 171–186, 2001.

[16] G. K. Gilbert, "Finley's tornado predictions," *Amer. MeteoroLog. J. A Monthly Rev. Meteorol. Allied Branches Study (1884–1896)*, vol. 1, no. 5, 1884, Art. no. 166.

[17] M. Doolittle, "Association ratios," *Bull. Philos. Soc. Washington*, vol. 7, pp. 122–127, 1888.

[18] C. C. Balch, "Updated verification of the space weather prediction center's solar energetic particle prediction model," *Space Weather, Int. J. Res. Appl.*, vol. 6, no. 1, 2008.

[19] M. O'Searcoid, *Metric Spaces*. Berlin, Germany: Springer, 2006.

[20] J. A. Swets, "The relative operating characteristic in psychology: A technique for isolating effects of response bias finds wide use in the study of perception and cognition," *Science*, vol. 182, no. 4116, pp. 990–1000, 1973.

[21] W. Peterson, T. Birdsall, and W. Fox, "The theory of signal detectability," *Trans. IRE Professional Group Informat. Theory*, vol. 4, no. 4, pp. 171–212, 1954.

[22] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Comput. Sci. Eng.*, vol. 9, no. 3, pp. 90–95, 2007.

[23] R. Vilalta and D. Oblinger, "A quantification of distance bias between evaluation metrics in classification," in *Proc. 17th Int. Conf. Mach. Learn.*, 2000, pp. 1087–1094.

[24] F. J. Provost and T. Fawcett, "Robust classification for imprecise environments," *Mach. Learn.*, vol. 42, no. 3, pp. 203–231, 2001.

[25] D. S. Bloomfield *et al.*, "Toward reliable benchmarking of solar flare forecasting methods," *The AstroPhys. J.*, vol. 747, no. 2, Feb. 2012, Art. no. L41. [Online]. Available: https://iopscience.iop.org/article/10.1088/2041–8205/747/2/L41

[26] R. A. Angryk *et al.*, "Multivariate time series dataset for space weather data analytics," *Sci. Data*, vol. 7, pp. 1–13, 2020.

[27] A. Ahmadzadeh, B. Aydin, M. K. Georgoulis, D. J. Kempton, S. S. Mahajan, and R. A. Angryk, "How to train your flare prediction model: Revisiting robust sampling of rare events," *CoRR*, 2021. [Online]. Available: https://arxiv.org/abs/2103.07542

[28] Y. LeCun , "The MNIST database of handwritten digits," 1998. [Online]. Available: http://yann.lecun.com/exdb/mnist/

[29] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[30] A. Paszke *et al.*, "PyTorch: An imperative style, high-performance deep learning library," in *Proc. Adv. Neural Informat. Process. Syst.*, 2019, pp. 8024–8035. [Online]. Available: https://proceedings.neurips.cc/paper/2019/hash/bdbca288fee7f92f2bfa9f7012727740-Abstract.html

[31] A. Kolmogorov-Smirnov, A. Kolmogorov, and M. Kolmogorov, "Sulla determinazione empírica di uma legge di distribuzione," 1933.

[32] N. Smirnov, "Table for Estimating the Goodness of Fit of Empirical Distributions," *Ann. Math. Statist.*, vol. 19, no. 2, pp. 279–281, 1948.

[33] R. Caruana and A. Niculescu-Mizil , "Data mining in metric space: An empirical analysis of supervised learning performance criteria," in *Proc. 10th ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining*, 2004, pp. 69–78.

[34] D. Hughes-Hallett, A. M. Gleason, and W. G. McCallum, *Calculus: Single and multivariable*. Hoboken, NJ, USA: Wiley, 2020.

[35] I. T. Jolliffe and D. B. Stephenson, *Forecast Verification: A Practitioner's Guide in Atmospheric Science*. Hoboken, NJ, USA: Wiley, 2012.
[36] M. Hossin and M. Sulaiman, "A review on evaluation metrics for data classification evaluations," *Int. J. Data Mining Knowl. Manage. Process*, vol. 5, no. 2, 2015, Art. no. 1.
[37] C. C. Aggarwal, A. Hinneburg, and D. A. Keim, "On the surprising behavior of distance metrics in high dimensional spaces," in *Proc. 8th Int. Conf. Database Theory*, 2001, pp. 420–434.
[38] D. Mossman, "Three-way ROCs," *Med. Decis. Mak.*, vol. 19, pp. 78–89, 1999.
[39] A. Srinivasan, "Note on the location of optimal classifiers in N-dimensional ROC space," 1999.
[40] N. Lachiche and P. A. Flach, "Improving accuracy and cost of two-class and multi-class probabilistic classifiers using ROC curves," in *Proc. 20th Int. Conf.Mach. Learn.*, 2003, pp. 416–423. [Online]. Available: http://www.aaai.org/Library/ICML/2003/icml03–056.php
[41] T. K. Ho and M. Basu, "Complexity measures of supervised classification problems," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 3, pp. 289–300, Mar. 2002.

Azim Ahmadzadeh received the Associate's degree in mathematics from Guilan University, Iran (2008), the BS degree in computer science from the Warsaw University of Technology, Poland (2016), and the PhD degree in computer science from Georgia State University, U.S. (2021). He is currently a postdoctoral research associate with Data Mining Laboratory (dmlab.cs.gsu.edu), Georgia State University. His interdisciplinary research centers on machine learning for detection, segmentation, and forecast of scientific events; Heliophysics data-driven pattern extraction, evaluation of object detection algorithms, and prediction of occurrence of solar events with sever socio-economic impact.

Dustin J. Kempton received the BS degree in computer science from the University of South Dakota in 2013 and the PhD degree in computer science from Georgia State University in 2018. He is currently a research assistant professor with the Department of Computer Science at Georgia State University. His research interests include computer vision, machine learning, and surrogate modeling as applied to solar physics and space weather prediction.

Petrus C. Martens received the BS degree in astronomy in 1977, and the MS and PhD in theoretical astrophysics, from Utrecht University, the Netherlands, in 1979 and 1983, respectively. He is currently a full tenured professor with GSU. He has well more than 100 refereed publications in his name and has mentored several successful graduate students. He is a solar physicist with experience in theory, simulations, data processing and analysis, automated feature recognition, data analytics, science operations, and space instrumentation. He has had mission involvement with Yohkoh, SoHO, TRACE, and SDO. Together with Prof. Angryk, he leads the interdisciplinary solar-stellar informatics research cluster with GSU.

Rafal A. Angryk received the MA degree in business management from the University of Szczecin, in 1999, three years later the MS degree in computer science from the Technical University of Szczecin, and the MS and PhD degrees in computer science from Tulane University in 2004. He is currently employed as a professor with the Computer Science Department, Georgia State University (GSU). Before joining GSU in August 2013, he spent almost a decade as a faculty member with Montana State University (MSU). He is the founding director with the MSU/GSU Data Mining Laboratory (dmlab.cs.gsu.edu) and holds two affiliate professor appointments with (1) the Department of Physics and Astronomy, College of Arts and Sciences and (2) the Institute for Insight, J.M. Robinson College of Business due to the interdisciplinary research he is conducting on massive repositories of scientific data.