OpenAUC: Towards AUC-Oriented Open-Set Recognition

Anonymous Author(s)
Affiliation
Address
email

Abstract

Traditional machine learning follows a close-set assumption that the training and test set share the same label space. While in many practical scenarios, it is inevitable that some test samples belong to unknown classes (open-set). To fix this issue, Open-Set Recognition (OSR), whose goal is to make correct predictions on both close-set samples and open-set samples, has attracted rising attention. In this direction, the vast majority of literature focuses on the pattern of open-set samples. However, how to evaluate model performance in this challenging task is still unsolved. In this paper, a systematic analysis reveals that most existing metrics are essentially inconsistent with the aforementioned goal of OSR: (1) For metrics extended from close-set classification, such as Open-Set F-score, Youden's index, and Normalized Accuracy, a poor open-set prediction can escape from a low performance score with a superior close-set prediction. (2) Novelty detection AUC, which measures the ranking performance between close-set and open-set samples, ignores the close-set performance. To fix these issues, we propose a novel metric named OpenAUC. Compared with existing metrics, OpenAUC enjoys a concise pairwise formulation that evaluates open-set performance and close-set performance in a coupling manner. On top of this, further analysis shows that OpenAUC is free from the inconsistency properties of existing metrics. Finally, an end-to-end learning method is proposed to minimize the OpenAUC risk, and the experimental results on six popular benchmark datasets speak to its effectiveness.

1 Introduction

Traditional classification algorithms have achieved tremendous success under the close-set assumption that all the test classes are known during the training period. However, in many practical scenarios, it is inevitable that some test samples belong to none of the known classes. In this case, a close-set model will classify all the novel samples as known classes, inducing a significant performance degeneration. To fix this issue, Open-Set Recognition (OSR) has attracted rising attention in recent years [1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14], where the model is required to not only (1) correctly classify the close-set samples but also (2) discriminate the open-set samples from the close-set ones. In this complicated setting, how to evaluate model performance becomes a challenging problem. Existing work has proposed several metrics, which fall into two categories:

The first direction extends traditional classification metrics to the open-set scenario. To this end, one first extends the close-set confusion matrix with unknown classes, where a threshold decides whether the input sample belongs to the unknown classes. On top of this, **open-set F-score** [2, 15, 9, 16, 11, 12] summarizes the True Positive (TP), False Positive (FP), and False Negative (FN) performance of known classes. **Youden's index** [17] takes the sum of the True Positive Rate (TPR) and True Negative Rate (TNR) performance of known classes as the performance measure. Besides,

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Normalized Accuracy [15] summarizes the close-set accuracy and open-set accuracy via a convex combination. Although it is intuitive to extend close-set metrics, we point out that these metrics are essentially inconsistent with the goal of OSR. Specifically, open-set F-score and Youden's index evaluate open-set performance only via the FP and FN performance of each known class. As a result, these metrics encourage classifying close-set samples as unknown classes to decrease the FN of known classes. Moreover, our analysis shows that Normalized Accuracy encourages selecting the threshold classifying more open-set samples as known classes. In extreme cases, even a close-set model (*i.e.*, all the open-set samples are classified as known classes) can obtain a high performance on these metrics.

The second category regards OSR as a novelty detection problem [18, 19] with multiple known classes. Based on such observation, the Area Under ROC Curve (AUC) [20], which measures the ranking performance between known classes and unknown classes, has become a popular metric [3, 4, 5, 6, 8, 10]. Compared with classification-based metrics, AUC is insensitive to the selection of threshold since it summarizes the True Positive Rate (TPR) performance for all possible thresholds. However, the limitation of AUC is also obvious: the close-set performance is ignored. A natural remedy is to adopt the accuracy defined on known classes as the metric for close-set performance [3]. However, what we expect is a model that can make correct predictions on close-set and open-set simultaneously. This decoupling strategy will induce a challenging multi-objective optimization problem and is also unfavorable to comparing the overall performances of different models.

In view of this, a natural question arises:

Whether there exists a numeric metric that is consistent with the goal of OSR?

To answer this question, we propose a novel metric named **OpenAUC**. Specifically, the proposed metric enjoys a concise pairwise formulation, where each pair consists of a close-set sample and an open-set sample. For each pair, only if the close-set sample is classified as the correct known class, OpenAUC will check whether the open-set sample is ranked higher than the close-set one. In this sense, OpenAUC evaluates the close-set performance and the open-set performance in a coupling manner, which is consistent with the goal of OSR. What's more, benefiting from the ranking operator, OpenAUC overcomes the sensitivity of the threshold, and further analysis shows that maximizing OpenAUC will guarantee a better open-set performance under a mild assumption on the threshold. Considering these advantages, we further establish an end-to-end learning method to maximize OpenAUC. Finally, extensive experiments conducted on six benchmark datasets validate the proposed metric and learning method. To sum up, the contribution of this paper is three-fold:

- We make a detailed analysis of existing metrics for OSR. The theoretical results show that existing metrics, including classification-based ones and AUC, are essentially inconsistent with the goal of OSR due to their own limitations.
- A novel metric, named OpenAUC, is proposed. Benefiting from its concise formulation, further analysis shows that OpenAUC overcomes the aforementioned limitations and thus is free from the inconsistency properties.
- An end-to-end learning method is proposed to optimize OpenAUC, and the empirical results on six benchmark datasets validate its effectiveness.

2 Preliminary

Problem definition. In open-set recognition, the training samples $\{\boldsymbol{z}_i = (\boldsymbol{x}_i, y_i)\}_{i=1}^n$ are drawn from a product space $\mathcal{Z}_k = \mathcal{X} \times \mathcal{Y}_k$, where $\mathcal{X}$ is the input space and $\mathcal{Y}_k = \{1, \cdots, C\}$ is the label space of known classes. During the test period, some samples might belong to none of the known classes. For the sake of simplicity, all these samples can be allocated to one super unknown class. In other words, the open-set samples are drawn from a product space $\mathcal{Z}_u = \mathcal{X} \times \mathcal{Y}_u$, where $\mathcal{Y}_u = \{C+1\}$ is the label space of unknown classes. To make predictions on known classes, OSR requires a classifier $h = g_1 \circ f$, where $f : \mathcal{X} \to \mathbb{R}^C$ is the close-set score function and $g_1 : \mathbb{R}^C \to \mathcal{Y}_k$ is the close-set decision function. Meanwhile, a rejector $R = g_2 \circ r$ is required to judge whether an input sample comes from open-set, where $r : \mathcal{X} \to \mathbb{R}$ is the open-set score function and $g_2 : \mathbb{R} \to \{0, 1\}$ is the open-set decision function. To be specific, an input $\boldsymbol{x}$ will be classified as an open-set sample if $r(\boldsymbol{x})$ is greater than a given threshold $t \in \mathbb{R}$. Then, the goal of OSR can be summarized as follows:

Table 1: The consistency analysis of existing metrics for OSR.

Metric	G1 (close)	G2 (open)	G3 (threshold-free)	numeric
Open-set F-score [15]	✓	×	×	✓
Youden's index [17]	✓	×	×	✓
Normalized Accuracy [15]	✓	✓	×	✓
AUC [3]	×	✓	✓	✓
The OSCR curve [4]	✓	✓	✓	×
OpenAUC (Ours)	✓	✓	✓	✓

- **(G1)** For close-set samples, not only the classifier h should make correct predictions, but also the close-set score function r should output low open-set scores.
- **(G2)** For open-set samples, the close-set score function r should output high open-set scores.
- **(G3)** The metrics should be insensitive to the ratio of open-set samples since this ratio is unavailable during the training period.

Roadmap. Next, we first present a detailed analysis of the inconsistency properties in Sec.3, whose results are summarized in Tab.1. Based on these results, we propose a novel metric named OpenAUC in Sec.4.2. Then, in Sec.4.3, further analysis shows that OpenAUC is free from inconsistency properties due to its pairwise formulation. Considering its advantages, an end-to-end learning method is finally proposed for OpenAUC in Sec.4.4.

3 Existing metrics for Open-set Recognition

Existing metrics for OSR fall into two categories: the classification-based ones and the novelty-detection ones. The first category extends existing classification metrics to the open-set scenario, while the second one regards OSR as a generalized novelty detection problem. We will present a detailed analysis of these metrics in the rest of this section.

3.1 Open-set F-score and Youden's Index

To extend classification metrics, one should first extend the confusion matrix with the unknown class. Let $\mathtt{TP}_i, \mathtt{TN}_i, \mathtt{FP}_i, \mathtt{FN}_i$ denote the True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN) of the class $i \in \mathcal{Y}_k \cup \mathcal{Y}_u$ under the given threshold t, respectively. Note that we omit the classifier h and the rejector R since there exists no ambiguity.

Open-set F-score [15] is a representative classification-based metric for OSR. Compared with close-set F-score, this metric evaluates the open-set performance via $\mathtt{FP}_i$ and $\mathtt{FN}_i$, where $i \in \mathcal{Y}_k$. To be specific, this metric summarizes $\mathtt{TP}_i, \mathtt{FP}_i, \mathtt{FN}_i$ of known classes by the harmonic mean of Precision and Recall:

$$\mathtt{F\text{-}score} := 2 \times \frac{\mathtt{P}_k \times \mathtt{TPR}_k}{\mathtt{P}_k + \mathtt{TPR}_k}, \tag{1}$$

where

$$\mathtt{P}_k := \frac{1}{C}\sum_{i=1}^{C} \frac{\mathtt{TP}_i}{\mathtt{TP}_i + \mathtt{FP}_i}, \mathtt{TPR}_k := \frac{1}{C}\sum_{i=1}^{C} \frac{\mathtt{TP}_i}{\mathtt{TP}_i + \mathtt{FN}_i} \tag{2}$$

if we aggregate model performances in a macro manner, and

$$\mathtt{P}_k := \frac{\sum_{i=1}^{C} \mathtt{TP}_i}{\sum_{i=1}^{C} (\mathtt{TP}_i + \mathtt{FP}_i)}, \mathtt{TPR}_k := \frac{\sum_{i=1}^{C} \mathtt{TP}_i}{\sum_{i=1}^{C} (\mathtt{TP}_i + \mathtt{FN}_i)} \tag{3}$$

when model performances are summarized in a micro manner. Compared with open-set F-score, Youden's index considers $\mathtt{TN}_i$, where $i \in \mathcal{Y}_k$ [21, 17]:

$$\mathtt{J} := \mathtt{TPR}_k + \mathtt{TNR}_k - 1, \tag{4}$$

where $\mathtt{TNR}_k$ denotes the True Negative Rate (TNR) of known classes. However, we point out that these two metrics suffer from an inconsistency property, as illustrated in Proposition 1. Please refer to Appendix.A.1 for the proof.

Proposition 1 (Inconsistency Property I)**.** *Given a dataset $\mathcal{S}$ and a metric $\mathtt{M}$ that is invariant to $\mathtt{TP}_{C+1}$, $\mathtt{FN}_{C+1}$ and $\mathtt{FP}_{C+1}$, then for any (h, R) such that $\sum_{i=1}^{C} \mathtt{FP}_i(h, R) \geq \mathtt{TP}_{C+1}(h, R)$, there exists $(\tilde{h}, \tilde{R})$ such that $\mathtt{M}(\tilde{h}, \tilde{R}) = \mathtt{M}(h, R)$ but $\mathtt{TP}_{C+1}(\tilde{h}, \tilde{R}) = 0$.*

Remark 1. *If a metric $\mathtt{M}$ suffers from the inconsistency property I, then for any (h, R), we can construct $(\tilde{h}, \tilde{R})$ that performs as well as (h, R) on $\mathtt{M}$ but actually misclassifies all the open-set samples as known classes, which is inconsistent with the goal **(G2)**.*

Remark 2. *$\sum_{i=1}^{C} \mathtt{FP}_i(h, R) \geq \mathtt{TP}_{C+1}(h, R)$ is a mild condition. To be specific, when $\mathtt{TP}_{C+1}$ is $\mathcal{O}(C)$, it only requires that $\mathtt{FP}_i$ is $\mathcal{O}(1)$ for any $i \in \mathcal{Y}_k$. What's more, even if this condition does not hold, we still have $\mathtt{TP}_{C+1}(\tilde{h}, \tilde{R}) < \mathtt{TP}_{C+1}(h, R)$ as long as $\sum_{i=1}^{C} \mathtt{FP}_i(h, R) \neq 0$.*

Corollary 1. *Open-set F-score and Youden's index both suffer from the inconsistency property I.*

3.2 Normalized Accuracy

Normalized Accuracy (NAcc) [15] measures the convex combination of the close-set accuracy and the open-set accuracy:

$$\mathtt{NAcc} := \lambda_{na}\mathtt{AKS} + (1 - \lambda_{na})\,\mathtt{AUS}, \tag{5}$$

where $\lambda_{na} \in (0, 1)$ is the balance constant, and

$$\mathtt{AKS} := \frac{\sum_{i=1}^{C} [\mathtt{TP}_i + \mathtt{TN}_i]}{\sum_{i=1}^{C} [\mathtt{TP}_i + \mathtt{TN}_i + \mathtt{FP}_i + \mathtt{FN}_i]}, \mathtt{AUS} := \frac{\mathtt{TP}_{C+1}}{\mathtt{TP}_{C+1} + \mathtt{FP}_{C+1}} \tag{6}$$

are the Accuracy on Known and Unknown Samples (AKS, AUS), respectively. Since the close-set performance is explicitly involved, NAcc avoids the inconsistency property I. Ideally, if $\lambda_{na} = \mathbb{P}[y = C + 1]$, NAcc becomes exactly the close-set accuracy. However, it is generally hard to decide the balance constant λ_{na} since we generally have no idea about the ratio of open-set samples in the test set. Besides, it suffers from another type of inconsistency property, as shown in Proposition 2. Please refer to Appendix.A.2 for the proof.

Proposition 2 (Inconsistency Property II)**.** *Given a dataset $\mathcal{S}$, for any classifier-rejector pair (h, R) such that $\sum_{i=1}^{C} \mathtt{FN}_i(h, R) \geq \mathtt{TP}_{C+1}(h, R)$ and $\mathtt{TP}_{C+1}(h, R) > \mathtt{FP}_{C+1}(h, R)$, there exists $(\tilde{h}, \tilde{R})$ such that $\mathtt{NAcc}(\tilde{h}, \tilde{R}) > \mathtt{NAcc}(h, R)$ but $\mathtt{TP}_{C+1}(\tilde{h}, \tilde{R}) = 0$.*

Remark 3. *For any (h, R), we can construct $(\tilde{h}, \tilde{R})$ such that $\mathtt{NAcc}(\tilde{h}, \tilde{R}) > \mathtt{NAcc}(h, R)$ but actually misclassifies all the open-set samples to known classes. In other words, NAcc encourages selecting a threshold such that more open-set samples are classified as known classes, which is inconsistent with the goal **(G3)**.*

Remark 4. *Similar to the condition in Proposition 1, $\sum_{i=1}^{C} \mathtt{FN}_i(h, R) \geq \mathtt{TP}_{C+1}(h, R)$ is a mild condition. Meanwhile, $\mathtt{TP}_{C+1}(h, R) > \mathtt{FP}_{C+1}(h, R)$ is also mild since it is a basic requirement for open-set models.*

3.3 The Area Under the ROC Curve (AUC)

When regarding OSR as a novelty detection problem, we no longer need to extend the confusion matrix, and AUC [3] comes into play. Traditional AUC is a popular metric for the imbalanced binary classification problem since it is insensitive to the label distribution [22, 23]. To extend this metric to OSR, one has to allocate all the known classes to one super known class, which is the same as the label space of unknown classes. Let $\mathtt{TPR}_1, \mathtt{FPR}_1$ denote the TPR and FPR of the super known class, respectively. Then, AUC is defined as the area under the $\mathtt{TPR}_1$-$\mathtt{FPR}_1$ curve:

$$\mathtt{AUC} := \int_{-\infty}^{+\infty} \mathtt{TPR}_1(\mathtt{FPR}_1^{-1}(t))\, dt. \tag{7}$$

If we assume that there exist no ties in r, AUC will enjoy a much simpler formulation, which equals the probability that the close-set score function r ranks an open-set sample higher than a close-set one [22]:

$$\mathtt{AUC} = \mathop{\mathbb{E}}_{\substack{\boldsymbol{z}_k \sim D_k \\ \boldsymbol{z}_u \sim D_u}} \left[\mathbf{1}\left[r(\boldsymbol{x}_u) > r(\boldsymbol{x}_k)\right]\right], \tag{8}$$

where D_k, D_u are the sample distribution of known and unknown classes, respectively. Compared with classification-based metrics, AUC summarizes the model performance under different thresholds, and the pairwise formulation (8) requires no thresholds. However, its limitation is also obvious: **the close-set performance is ignored**, which is inconsistent with the goal **(G1)**. To fix this issue, one can adopt the close-set accuracy, denoted by $\texttt{Acc}_k$, as the metric for the close-set performance. In other words, this strategy divides OSR into two traditional tasks: multiclass classification and novelty detection. Although intuitive, it is also inconsistent with the goal of OSR since the open-set performance and the close-set performance are evaluated in a decoupling manner. On one hand, if r ranks some close-set samples higher than open-set ones, a correct prediction of h becomes meaningless. On the other hand, if the classifier h has misclassified some close-set samples, it also becomes unnecessary to calculate AUC on these samples. What's more, simply aggregating the two metrics will lead to another inconsistency property, whose proof is presented in Appendix.A.3:

Proposition 3 (Inconsistency Property III)**.** *Given a dataset $\mathcal{S}$, for any classifier and open-set score function (h, r) satisfying $\texttt{Acc}_k(h), \texttt{AUC}(r) \neq 1$, there exists $(\tilde{h}, \tilde{r})$ that performs worse on the OSR task but satisfies:*

$$agg(\texttt{Acc}_k(\tilde{h}), \texttt{AUC}(\tilde{r})) = agg(\texttt{Acc}_k(h), \texttt{AUC}(r)),$$

where $agg : \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ is the aggregation function.

Corollary 2. *The following metrics suffer from the inconsistency property III:*

- *The product of the close-set accuracy and AUC, i.e., $\texttt{Acc}_k \cdot \texttt{AUC}$;*
- *The summation of the close-set accuracy and AUC, i.e., $\texttt{Acc}_k + \texttt{AUC}$;*
- *The pointwise summation of the close-set accuracy and AUC:*

$$\texttt{Acc}_k \oplus \texttt{AUC} := \mathop{\mathbb{E}}_{\substack{\boldsymbol{z}_k \sim D_k \\ \boldsymbol{z}_u \sim D_u}} \left[\mathbf{1}\left[y_k = h(\boldsymbol{x}_k)\right] + \mathbf{1}\left[r(\boldsymbol{x}_u) > r(\boldsymbol{x}_k)\right] \right].$$

Remark 5. *Simply aggregating the close-set metric and the open-set metric is still inconsistent with the goal of OSR.* ***In view of this, we will present a novel metric in the next section****.*

4 OpenAUC

4.1 Open Set Classification Rate (OSCR)

Considering the limitation of AUC, [4] proposes the Open Set Classification Rate (OSCR) curve. In this curve, $f(\boldsymbol{x})_c$ and $r(\boldsymbol{x})$ are assumed to be proportional to $\mathbb{P}\left[y = c|\boldsymbol{x}\right]$ and $1/\max_{c \in \mathcal{Y}_k} f(\boldsymbol{x})_c$, respectively. Then, x-axis is defined as the FPR performance on open-set samples, while y-axis is the Correct Classification Rate (CCR) on close-set samples:

$$\texttt{CCR}(t) := \frac{1}{N_k} \sum_{i=1}^{N_k} \mathbf{1}\left[y_i = \arg\max_{c \in \mathcal{Y}_k} f(\boldsymbol{x}_i)_c\right] \cdot \mathbf{1}\left[f(\boldsymbol{x}_i)_{y_i} > t\right] \tag{9}$$

where N_k is the number of close-set samples, and $\mathbf{1}\left[\mathcal{A}\right]$ is the indicator function of an event $\mathcal{A}$. Compared with AUC, the OSCR curve considers the close-set performance via CCR. Meanwhile, as FPR varies from 0 to 1, this curve will present the CCR performance under different thresholds. Although the curve can illustrate model performance in an intuitive manner, a numeric metric is generally necessary for model comparison. To this end, [13] and [14] estimate the area under the $\texttt{CCR-FPR}_1$ curve on real-world datasets in a histogram manner. However, **there still exists no concise formulation to obtain a numeric metric, making it difficult to compare this metric with others**. Moreover, the assumptions on (f, r) also limits its application. For example, considering that $\mathcal{Z}_u = \neg\mathcal{Z}_1 \cap \neg\mathcal{Z}_2 \cap \cdots \cap \neg\mathcal{Z}_C$, [24, 13] propose to model potential unknown space via the prototypes of $\{\neg\mathcal{Z}_c\}_{c=1}^{C}$. Inspired by this, we adopt the following definition, where the aforementioned assumptions on (f, r) do not hold:

Definition. *Let $f(\boldsymbol{x})_c$ denote the probability that the input $\boldsymbol{x}$ does not belong to the class c, that is, $\forall c \in \mathcal{Y}_k, f(\boldsymbol{x})_c \propto \mathbb{P}\left[y \neq c|\boldsymbol{x}\right]$. In this case, $r(\boldsymbol{x})$ is defined as $\min_{k \in \mathcal{Y}_k} f(\boldsymbol{x})_k$ since given an open-set sample $\boldsymbol{x}$, a well-trained model tends to output a large $f(\boldsymbol{x})_k$ for any $k \in \mathcal{Y}_k$.*

4.2 The Definition of OpenAUC

To overcome the limitations of the OSCR curve, we first remove the assumptions on (h, r) and present a generalized formulation of open-set FPR and CCR, denoted by OFPR and OTPR, respectively:

$$\begin{aligned} \texttt{OFPR}(t) &:= \mathop{\mathbb{E}}_{\boldsymbol{z}\sim D_u} \left[\mathbf{1}\left[r(\boldsymbol{x}) \le t\right]\right], \\ \texttt{OTPR}(t) &:= \mathop{\mathbb{E}}_{\boldsymbol{z}\sim D_k} \left[\mathbf{1}\left[r(\boldsymbol{x}) \le t, y = h(\boldsymbol{x})\right]\right], \end{aligned} \tag{10}$$

In other words, OFPR represents the probability that r misclassifies a randomly-chosen open-set sample as known classes, and OTPR equals the probability that (h, r) makes a correct prediction on a randomly chosen close-set sample. Then, OpenAUC is defined as the area under the OFPR-OTPR curve:

$$\texttt{OpenAUC} := \int_{-\infty}^{+\infty} \texttt{OTPR}(\texttt{OFPR}^{-1}(t))\, dt. \tag{11}$$

However, this integral formulation is generally hard to calculate. To fix this issue, we present the following reformulation, whose proof is presented in Appendix.B.1.

Proposition 4. *Given (h, r) and a close-open-set pair $(\boldsymbol{z}_k, \boldsymbol{z}_u)$, OpenAUC equals the probability that h makes correct prediction on $\boldsymbol{z}_k$ and r ranks $\boldsymbol{z}_u$ higher than $\boldsymbol{z}_k$:*

$$\texttt{OpenAUC} = \mathop{\mathbb{E}}_{\substack{\boldsymbol{z}_k\sim D_k \\ \boldsymbol{z}_u\sim D_u}} \Big[\underbrace{\mathbf{1}\left[y_k = h(\boldsymbol{x}_k)\right]}_{(a)} \cdot \underbrace{\mathbf{1}\left[r(\boldsymbol{x}_u) > r(\boldsymbol{x}_k)\right]}_{(b)}\Big] \tag{12}$$

Remark 6. *First, for close-set samples, term (a) and term (b) require that h makes correct predictions and r outputs low open-set scores, which is consistent with **(G1)**. Second, term (b) requires r to rank open-set samples higher than the close ones, which is consistent with **(G2)** and **(G3)**. Last but not least, the product operator guarantees that OpenAUC evaluates the close-set performance and the open-set performance in a coupling manner, which is essentially different from the metrics mentioned in Proposition 3.*

So far, we have known that OpenAUC is consistent with the goal of OSR. One might wonder whether OpenAUC suffers from the aforementioned inconsistency property. Next, we will present a detailed analysis on this issue.

4.3 OpenAUC vs. Existing Metrics for OSR

Essentially, inconsistency property I is induced by the fact that exchanging the predictions of open-set samples and close-set samples will not lead to a larger error on these metrics. However, such an exchange at least breaks a sample pair that has been correctly ranked by r, and will not improve the close-set accuracy. Thus, we have the following proposition, whose details are presented in Appendix.B.2.

Proposition 5. *Given a close-open pair $((\boldsymbol{x}_1, C+1), (\boldsymbol{x}_2, y_2))$, where $y_2 \neq C+1$, for any (h, r) such that $R(\boldsymbol{x}_1) = 1, R(\boldsymbol{x}_2) = 0, h(\boldsymbol{x}_2) \neq y_2$, if $(\tilde{h}, \tilde{r})$ makes the same predictions as (h, r) expect that $\tilde{R}(\boldsymbol{x}_1) = 0, \tilde{h}(\boldsymbol{x}_1) = h(\boldsymbol{x}_2)$ and $\tilde{R}(\boldsymbol{x}_2) = 1$, we have $\texttt{OpenAUC}(\tilde{h}, \tilde{r}) < \texttt{OpenAUC}(h, r)$.*

Remark 7. *If we construct $(\tilde{h}, \tilde{r})$ in the way that leads to the inconsistency property I, $(\tilde{h}, \tilde{r})$ will perform inferior to (h, r) on OpenAUC, that is, OpenAUC is free from the inconsistency property I.*

Besides, the following proposition reveals that optimizing OpenAUC will guarantee a lower bound of $\texttt{TPR}_{C+1}$, which helps avoid the inconsistency property II. Please refer to Appendix.B.3 for the proof.

Proposition 6. *Given a dataset $\mathcal{S}$, for any (f, r) such that $\texttt{OpenAUC} = k$ and any threshold t_{C+1} such that $\texttt{FPR}_{C+1} = a \neq 0$, we have $\texttt{TPR}_{C+1} \ge 1 - (1-k)/a$.*

Remark 8. *It is clear that $\texttt{TPR}_{C+1} > 0$ when $a > 1-k$. As we optimize OpenAUC, that is, $k \to 1$, it will become easier to select a threshold t_{C+1} such that $a > 1-k$. In other words, OpenAUC is free from the inconsistency property II under mild condition.*

On the other hand, inconsistency property III is induced by the fact that $agg(\texttt{Acc}_k, \texttt{AUC})$ evaluates the close-set performance and the open-set performance essentially in a decoupling manner. As a comparison, only if (h, r) makes correct predictions on both close-set samples and open-set samples, it will be free from the punishment of OpenAUC. In light of this, we have the following proposition, whose details will be presented in Appendix.B.4.

Proposition 7. *Given two close-set samples $(\boldsymbol{x}_1, y_1)$ and $(\boldsymbol{x}_2, y_2)$ and an open-set sample $(\boldsymbol{x}_3, C+1)$, if $(\tilde{h}, \tilde{r})$ makes the same predictions as (h, r) expect that $h(\boldsymbol{x}_1) = \tilde{h}(\boldsymbol{x}_1) = y_1, h(\boldsymbol{x}_2), \tilde{h}(\boldsymbol{x}_2) \neq y_2, r(\boldsymbol{x}_2) > r(\boldsymbol{x}_3) > r(\boldsymbol{x}_1), r(\boldsymbol{x}_3) = \tilde{r}(\boldsymbol{x}_3)$ and $\tilde{r}(\boldsymbol{x}_2) = r(\boldsymbol{x}_1), \tilde{r}(\boldsymbol{x}_1) = r(\boldsymbol{x}_2)$, we have $\texttt{OpenAUC}(\tilde{h}, \tilde{r}) < \texttt{OpenAUC}(h, r)$.*

Remark 9. *If we construct $(\tilde{h}, \tilde{r})$ in the way that leads to the inconsistency property III, $(\tilde{h}, \tilde{r})$ will perform inferior to (h, r) on OpenAUC, that is, OpenAUC is free from the inconsistency property III.*

To sum up, OpenAUC is free from the aforementioned escape properties. In view of this, it has become an appealing problem to design learning methods that can optimize OpenAUC efficiently.

4.4 Learning Method for OpenAUC

Following the standard machine learning paradigm [25], we should first reformulate the metric optimization problem to a risk minimization problem. For traditional metrics such as the close-set accuracy, one can simply replace $\mathbf{1}[\mathcal{A}]$ with $\mathbf{1}[\neg\mathcal{A}]$. However, due to the product term, it is clear that

$$\mathcal{R}(f, r) := 1 - \texttt{OpenAUC} \neq \mathop{\mathbb{E}}_{\substack{\boldsymbol{z}_k \sim D_k \\ \boldsymbol{z}_u \sim D_u}} \left[\mathbf{1}[y_k \neq h(\boldsymbol{x}_k)] \cdot \mathbf{1}[r(\boldsymbol{x}_u) \leq r(\boldsymbol{x}_k)] \right]. \tag{13}$$

To fix this issue, we reformulate the product term to a simpler summation term, whose proof is presented in Appendix.B.5.

Proposition 8. *Optimizing OpenAUC is equivalent to the following risk minimization problem:*

$$\min_{f,r} \mathcal{R}(f, r) = \mathop{\mathbb{E}}_{\substack{\boldsymbol{z}_k \sim D_k \\ \boldsymbol{z}_u \sim D_u}} \Big[\underbrace{\mathbf{1}[y_k \neq h(\boldsymbol{x}_k)]}_{(a)} + \underbrace{\mathbf{1}[y_k = h(\boldsymbol{x}_k)]}_{(b)} \cdot \underbrace{\mathbf{1}[r(\boldsymbol{x}_u) \leq r(\boldsymbol{x}_k)]}_{(c)} \Big] \tag{14}$$

Remark 10. *If we view term (b) as a switch of term (c), then $\mathcal{R}(f, r)$ essentially is a conditional combination of the close-set error and AUC risk, that is, term (a) and term (c). In other words, we first calculate the close-set error, and only if f makes a correct prediction, we will calculate the open-set AUC risk. This intuition is consistent with the goal of OSR.*

On top of this observation, we have the following empirical minimization objective:

$$\hat{\mathcal{R}}_{L,\ell}(f, r) := \frac{1}{N_k} \sum_{i=1}^{N_k} L(h(\boldsymbol{x}_i), y_i) + \frac{1}{N_k N_u} \sum_{i=i}^{N_k} \sum_{j=1}^{N_u} \left[\mathbf{1}[y_i = h(\boldsymbol{x}_i)] \cdot \ell(r(\boldsymbol{x}_j) - r(\boldsymbol{x}_i)) \right], \tag{15}$$

where L is a common close-set classification loss function such as the cross-entropy loss; ℓ represents a continuous surrogate loss for AUC optimization [26], such as the square loss $\ell_{sq}(t) = (1-t)^2$; $\mathbf{1}[y_k = h(\boldsymbol{x}_k)]$ acts as the switch of ℓ and will not be updated in each epoch. Note that there exist no open-set samples in the training set. Inspired by [11], we adopt manifold mixup [27] to generate open-set samples. Specifically, the convex combinations of samples from different known classes are regarded as open-set samples:

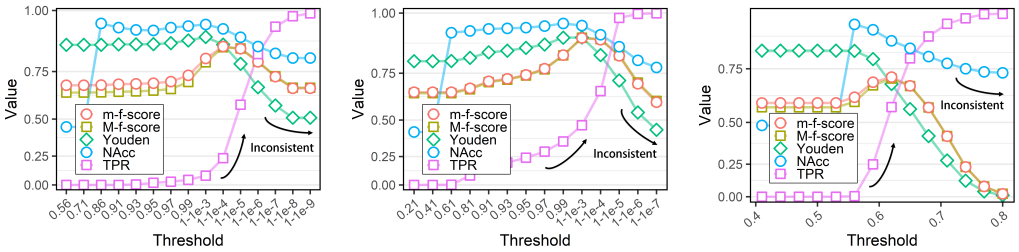
$$\boldsymbol{x}_u = \lambda_\beta f_{pre}(\boldsymbol{x}_i) + (1 - \lambda_\beta) f_{pre}(\boldsymbol{x}_j), y_i \neq y_j, \tag{16}$$

where $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are samples from different known classes; f is decomposed as $f_{post}(f_{pre}(\boldsymbol{x}))$; $\lambda_\beta \in (0, 1)$ is sampled from a Beta distribution $B(\alpha, \alpha)$. Then, the score of $\boldsymbol{x}_u$ is obtained by $f_{post}(\boldsymbol{x}_u)$. Compared with other generative models [3, 12, 16], manifold mixup enjoys a significant efficiency advantage. While a potential problem is that $\boldsymbol{x}_u$ might locate close to the manifolds of other known classes, *i.e.*, $\lambda_\beta f_{pre}(\boldsymbol{x}_i) + (1 - \lambda_\beta) f_{pre}(\boldsymbol{x}_j) \approx f_{pre}(\boldsymbol{x}_k), k \neq i, j$. To this end, we set an extra hyperparameter λ for the AUC risk, and the final objective becomes:

$$\hat{\mathcal{R}}_{L,\ell,\lambda}(f, r) := \frac{1}{N_k} \sum_{i=1}^{N_k} L(h(\boldsymbol{x}_i), y_i) + \frac{\lambda}{N_k N_u} \sum_{i=i}^{N_k} \sum_{j=1}^{N_u} \left[\mathbf{1}[y_i = h(\boldsymbol{x}_i)] \cdot \ell(r(\boldsymbol{x}_j) - r(\boldsymbol{x}_i)) \right]. \tag{17}$$

5 Experiments

In this section, extensive experiments are conducted to answer the following research questions: **(Q1)** Does the inconsistency property exist in empirical results? **(Q2)** Does optimizing OpenAUC avoid the inconsistency properties and thus help the model outperform the state-of-the-art methods? **(Q3)** How does the proposed method perform under different hyperparameter settings?



(a) CIFAR+10 (b) CIFAR+50 (c) TinyImageNet

Figure 1: The inconsistency property of classification-based metrics. We can find that all these metrics decrease rapidly as the TPR performance of unknown classes increases.

Table 2: The OpenAUC results on six benchmark datasets for OSR, where C+10, C+50 and CE+ represent **CIFAR+10**, **CIFAR+50** and Cross-Entropy+, respectively. The best and the runner-up method on each dataset are marked with **red** and **blue**, respectively.

Method	MNIST	SVHN	CIFAR10	C+10	C+50	TinyImageNet
Softmax	99.2±0.1	92.8±0.4	83.8±1.5	90.9±1.3	88.5±0.7	60.8±5.1
GCPL [28]	99.1±0.2	93.4±0.6	84.3±1.7	91.0±1.7	88.3±1.1	59.3±5.3
RPL [24]	99.4±0.1	93.6±0.5	85.2±1.4	91.8±1.2	89.6±0.9	53.2±4.6
ARPL [13]	99.4±0.1	94.0±0.6	86.6±1.4	93.5±0.8	91.6±0.4	62.3±3.3
ARPL+CS [13]	**99.5±0.1**	**94.3±0.3**	87.9±1.5	**94.7±0.7**	**92.9±0.3**	65.9±3.8
CE+ [14]	99.1±0.2	93.9±0.4	**88.1±1.7**	93.2±0.6	90.2±0.4	**74.3±3.9**
Acc+AUC	99.3±0.2	94.0±0.9	87.6±1.9	93.6±1.0	92.0±0.5	74.0±4.0
Ours	**99.4±0.1**	**95.0±0.4**	**89.2±1.9**	**95.2±0.7**	**93.6±0.3**	**75.9±4.1**

5.1 Protocol

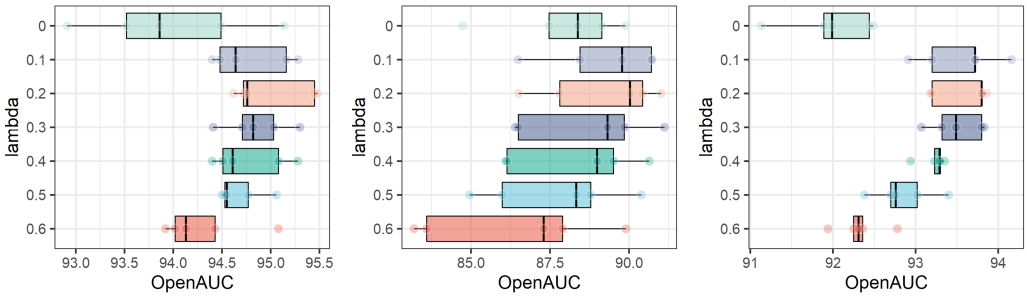
Following the protocol in [13], all experiments are conducted on the following datasets: (1) **MNIST**[1] [29], **SVHN**[2] [30] and **CIFAR10**[3] [31], where 4 classes are randomly sampled as the unknown classes; (2) **CIFAR+10** and **CIFAR+50**, where 4 classes are sampled from CIFAR10 as known classes, and N non-overlapping classes sampled from the CIFAR100 dataset[3] [31] act as the unknown classes; (3) **TinyImageNet**[4] [32], where 20 and 180 classes are randomly sampled to evaluate the close-set and open-set performance.

To answer the question **(Q1)**, we record the model perform of Cross-Entropy+ [14] on open-set F-score, Youden' index, Normalized Accuracy, AUC, close-set accuracy and TPR of the open-set class. To answer the question **(Q2)**, we compare the proposed method with the state-of-the-art methods, including Softmax, GCPL [28], RPL [24], ARPL [13], ARPL+CS [13], Cross-Entropy+ [14]. Besides, to validate the inconsistency property III, we also compare the objective that removes the term $\mathbf{1}\left[y_k = h(\boldsymbol{x}_k)\right]$ in Equation (17), which is denoted by Acc+AUC.

5.2 Implementation details

Infrastructure. All the experiments are carried out on an ubuntu server equipped with Intel(R) Xeon(R) Silver 4110 CPU and an Nvidia(R) TITAN RTX GPU. We implement the codes via `python` (v-3.8.11), and the main third-party packages include `pytorch` (v-1.9.0) [33], `numpy` (v-1.20.3) [34], `scikit-learn` (v-0.24.2) [35] and `torchvision` (v-0.10.1) [36].

[1] http://yann.lecun.com/exdb/mnist/. Licensed GPL3.
[2] http://ufldl.stanford.edu/housenumbers/. Licensed GPL3.
[3] https://www.cs.toronto.edu/~kriz/cifar.html. Licensed MIT.
[4] http://cs231n.stanford.edu/tiny-imagenet-200.zip. Licensed MIT.



(a) SVHN (b) CIFAR10 (c) CIFAR+50

Figure 2: The sensitivity analysis of the proposed learning method for OpenAUC.

Backbone and Optimization Method. We adopt the widely-used VGG32 model as the backbone [1, 2, 5, 13, 14]. Inspired by [14], we train the model with a batch size of 128 for 600 epochs except TinyImageNet, for which we use 64. Meanwhile, we adopt an initial learning rate of 0.1 for all datasets except TinyImageNet, for which we use 0.01. We train with a cosine annealed learning rate, restarting the learning rate to the initial value at epochs 200 and 400. Besides, the RandAugment and label smoothing strategy provided by [14] is utilized for all experiments.

Generation of Open-set Samples. As elaborated in Sec.4.4, we utilize manifold mixup to generate open-set samples. According to the empirical results in [11], we set $\alpha = 2$ as the default value. Meanwhile, the hyperparameter λ is searched in $\{0.1, 0.2, 0.3, 0.4, 0.5, 0.6\}$.

5.3 Results and Analysis

Metric comparison (Q1). We present the model performance on classification-based metrics under different thresholds in Fig.1. From these results, we can find that all the classification-based metrics decrease rapidly as the TPR performance of the unknown class increases. Moreover, on CIFAR+10 and TinyImageNet, the best NAcc performance corresponds to a zero TPR performance of the unknown class, which is consistent with our analysis in Proposition 2. In other words, all these metrics encourage us selecting a threshold that misclassifies more the open-set samples. All these results show that the inconsistency properties do exist in practical scenarios.

Performance Comparison (Q2). We compare the proposed method with competitors in Tab.2 and have the following observations: (1) Our method outperforms the competitors consistently on all the datasets, which validate the effectiveness of the proposed learning method. (2) Acc+AUC performs inferior to the proposed method, which is consistent with the inconsistency property III.

Sensitivity Analysis (Q3). We present the sensitivity in terms of the hyperparameter λ in Fig.2. From the results, we have the following observation: (1) Optimizing the OpenAUC risk consistently helps improve the model performance, which is consistent with our goal to design the OpenAUC objective. (2) When λ equals 0.1 or 0.2, the proposed method achieves the best performance. (3) As λ increases, there exists a decreasing trend in model performance, which might be induced by the noise in generated samples.

6 Conclusion and Further Work

This paper presents an extensive analysis of existing metrics for Open-Set Recognition. The theoretical results show that existing metrics are essentially inconsistent with the goal of OSR due to their own limitations. To fix this issue, a novel metric named OpenAUC is proposed. Compared with existing metrics, OpenAUC enjoys a concise pairwise formulation that evaluates the close-set performance and open-set performance in a coupling manner. A series of propositions show that OpenAUC is free from the inconsistency properties of existing metrics. Finally, an end-to-end learning method is proposed for OpenAUC, and the experimental results on six benchmark datasets validate the theoretical results and the effectiveness of the proposed method. In the further, we will consider the generalization bound of OpenAUC and propose more principled learning methods.

References

[1] Abhijit Bendale and Terrance E. Boult. Towards open set deep networks. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1563–1572, 2016.

[2] Zongyuan Ge, Sergey Demyanov, and Rahil Garnavi. Generative openmax for multi-class open set classification. In *British Machine Vision Conference*, 2017.

[3] Lawrence Neal, Matthew L. Olson, Xiaoli Z. Fern, Weng-Keen Wong, and Fuxin Li. Open set learning with counterfactual images. In *European Conference on Computer Vision*, pages 620–635, 2018.

[4] Akshay Raj Dhamija, Manuel Günther, and Terrance E. Boult. Reducing network agnostophobia. In *Advances in Neural Information Processing Systems*, pages 9175–9186, 2018.

[5] Poojan Oza and Vishal M. Patel. C2AE: class conditioned auto-encoder for open-set recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2307–2316, 2019.

[6] Pramuditha Perera and Vishal M. Patel. Deep transfer learning for multiple class novelty detection. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 11544–11552, 2019.

[7] Ryota Yoshihashi, Wen Shao, Rei Kawakami, Shaodi You, Makoto Iida, and Takeshi Naemura. Classification-reconstruction learning for open-set recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4016–4025, 2019.

[8] Pramuditha Perera, Vlad I. Morariu, Rajiv Jain, Varun Manjunatha, Curtis Wigington, Vicente Ordonez, and Vishal M. Patel. Generative-discriminative feature representations for open-set recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 11811–11820, 2020.

[9] Xin Sun, Zhenning Yang, Chi Zhang, Keck Voon Ling, and Guohao Peng. Conditional gaussian distribution learning for open set recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 13477–13486, 2020.

[10] Hongjie Zhang, Ang Li, Jie Guo, and Yanwen Guo. Hybrid models for open set recognition. In *European Conference on Computer Vision*, pages 102–117, 2020.

[11] Da-Wei Zhou, Han-Jia Ye, and De-Chuan Zhan. Learning placeholders for open-set recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 4401–4410, 2021.

[12] Zhongqi Yue, Tan Wang, Qianru Sun, Xian-Sheng Hua, and Hanwang Zhang. Counterfactual zero-shot and open-set visual recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 15404–15414, 2021.

[13] Guangyao Chen, Peixi Peng, Xiangqian Wang, and Yonghong Tian. Adversarial reciprocal points learning for open set recognition. *CoRR*, abs/2103.00953, 2021.

[14] Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need. *CoRR*, abs/2110.06207, 2021.

[15] Pedro Ribeiro Mendes Júnior, Roberto Medeiros de Souza, Rafael de Oliveira Werneck, Bernardo V. Stein, Daniel V. Pazinato, Waldir R. de Almeida, Otávio A. B. Penatti, Ricardo da Silva Torres, and Anderson Rocha. Nearest neighbors distance ratio open-set classifier. *Mach. Learn.*, 106:359–386, 2017.

[16] Shu Kong and Deva Ramanan. Opengan: Open-set recognition via open data generation. In *IEEE/CVF International Conference on Computer Vision*, pages 793–802, 2021.

[17] Matthew D. Scherreik and Brian D. Rigling. Open set recognition for automatic target classification with rejection. *IEEE Trans. Aerosp. Electron. Syst.*, 52:632–642, 2016.

[18] Marco A. F. Pimentel, David A. Clifton, Lei A. Clifton, and Lionel Tarassenko. A review of novelty detection. *Signal Process.*, 99:215–249, 2014.

[19] Mohammadreza Salehi, Hossein Mirzaei, Dan Hendrycks, Yixuan Li, Mohammad Hossein Rohban, and Mohammad Sabokrou. A unified survey on anomaly, novelty, open-set, and out-of-distribution detection: Solutions and future challenges. *CoRR*, abs/2110.14051, 2021.

[20] Tom Fawcett. An introduction to ROC analysis. *Pattern Recognit. Lett.*, 27:861–874, 2006.

[21] W. J. Youden. Index for rating diagnostic tests. *Cancer*, 3:32–35, 1950.

[22] Charles X. Ling, Jin Huang, and Harry Zhang. AUC: a statistically consistent and more discriminating measure than accuracy. In *International Joint Conference on Artificial Intelligence*, pages 519–526, 2003.

[23] David J. Hand. Measuring classifier performance: a coherent alternative to the area under the ROC curve. *Mach. Learn.*, 77:103–123, 2009.

[24] Guangyao Chen, Limeng Qiao, Yemin Shi, Peixi Peng, Jia Li, Tiejun Huang, Shiliang Pu, and Yonghong Tian. Learning open set network with discriminative reciprocal points. In *European Conference on Computer Vision*, pages 507–522, 2020.

[25] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of Machine Learning*. MIT Press, 2012.

[26] Wei Gao and Zhi-Hua Zhou. On the consistency of AUC pairwise optimization. In *International Joint Conference on Artificial Intelligence*, pages 939–945, 2015.

[27] Vikas Verma, Alex Lamb, Christopher Beckham, Amir Najafi, Ioannis Mitliagkas, David Lopez-Paz, and Yoshua Bengio. Manifold mixup: Better representations by interpolating hidden states. In *International Conference on Machine Learning*, pages 6438–6447, 2019.

[28] Hong-Ming Yang, Xu-Yao Zhang, Fei Yin, and Cheng-Lin Liu. Robust classification with convolutional prototype learning. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 3474–3482, 2018.

[29] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. IEEE*, 86:2278–2324, 1998.

[30] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. In *Neural Information Processing Systems Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[31] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[32] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *Int. J. Comput. Vis.*, 115:211–252, 2015.

[33] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Köpf, Edward Z. Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pages 8024–8035, 2019.

[34] Charles R. Harris, K. Jarrod Millman, Stéfan van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with numpy. *Nat.*, 585:357–362, 2020.

[35] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake Vanderplas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Edouard Duchesnay. Scikit-learn: Machine learning in python. *J. Mach. Learn. Res.*, 12:2825–2830, 2011.

[36] Sébastien Marcel and Yann Rodriguez. Torchvision the machine-vision package of torch. In *Proceedings of the 18th International Conference on Multimedia 2010, Firenze, Italy, October 25-29, 2010*, pages 1485–1488, 2010.

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 Sec.6.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] There are no potential negative societal impacts associated with our research.
   (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]
   (b) Did you include complete proofs of all theoretical results? [Yes] See the supplementary materials.
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]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] The experiments are conducted on five different splits of each dataset, and we report the standard divation in Tab.2.
   (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)? [Yes]
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]
   (b) Did you mention the license of the assets? [Yes]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (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]