ECAI 2023
K. Gal et al. (Eds.)

doi:10.3233/FAIA230454

Adversarial Benchmark Evaluation Rectified by Controlling for Difficulty

Behzad Mehrbakhsh [a,b,c;*], **Fernando Martínez-Plumed**[a,b] **and José Hernández-Orallo**[a,b,c]

[a]UPV - Universitat Politècnica de València
[b]VRAIN - Valencian Research Institute for Artificial Intelligence
[c]ValgrAI - Valencian Graduate School and Research Network of Artificial Intelligence

Abstract. Adversarial benchmark construction, where harder instances challenge new generations of AI systems, is becoming the norm. While this approach may lead to better machine learning models —on average and for the new benchmark—, it is unclear how these models behave on the original distribution. Two opposing effects are intertwined here. On the one hand, the adversarial benchmark has a higher proportion of difficult instances, with lower expected performance. On the other hand, models trained on the adversarial benchmark may improve on these difficult instances (but may also neglect some easy ones). To disentangle these two effects we can control for difficulty, showing that we can recover the performance on the original distribution, provided the harder instances were obtained from this distribution in the first place. We show this difficulty-aware rectification works in practice, through a series of experiments with several benchmark construction schemas and the use of a populational difficulty metric. As a take-away message, instead of distributional averages we recommend using difficulty-conditioned characteristic curves when evaluating models built with adversarial benchmarks.

1 Introduction

Benchmarks are increasingly prevalent in AI, and in machine learning in particular [37, 18]. Benchmarks such as ImageNet [6], CIFAR10/100 [16], (Super)GLUE [35] or SQuAD [26] are becoming reference points to which all techniques are expected to be compared. This success has led to an acceleration in the development of new and more varied benchmarks [11]. When human-equivalent performance is reached for these benchmarks, they are often discontinued and replaced, or extended through the inclusion of more challenging problems, in a kind of 'challenge-solve-and-replace' evaluation dynamic [29], or a 'dataset-solve-and-patch' adversarial benchmark co-evolution [36]. For example, the ImageNet dataset has been regularly updated with more challenging images and categories, as AI systems have improved (e.g., ImageNetV2 [27]). Also, CIFAR10 is accompanied by the more challenging CIFAR100 [16]. The same happens in the field of NLP, where SQuAD1.1 has been replaced by SQuAD2.0 [26] or GLUE by SuperGLUE [35]. Overall, as AI systems have become more advanced and powerful, the benchmarks used to evaluate their performance have also become more challenging.

Instead of replacing the benchmark completely, adversarial benchmarking has gained popularity as a more gradual and systematic approach to improving machine learning models by continuously incorporating increasingly difficult or *adversarial* examples. This method also causes a temporary drop in performance which is seen as a challenged, and re-aligns expectations for systems that are intuitively less good than what the previous state of the benchmark indicated. Adversarial examples can be either artificial [9] or natural [10]. The former involve the generation of new hand-crafted adversarial examples which are designed to exploit the weaknesses of a model and are typically created using optimisation algorithms that find the minimum necessary perturbation that lead to incorrect predictions. The latter consist of real-world, unmodified, and naturally occurring examples (e.g., difficult examples or anomalies of unforeseen classes) that should lead to low-confidence predictions, causing the model performance to degrade significantly. In contrast to artificial adversarial examples, by including natural adversarial examples in benchmarks, researchers and developers can test the ability of AI systems to handle these types of variations and maintain their performance in the face of real-world challenges. However, the cyclical nature

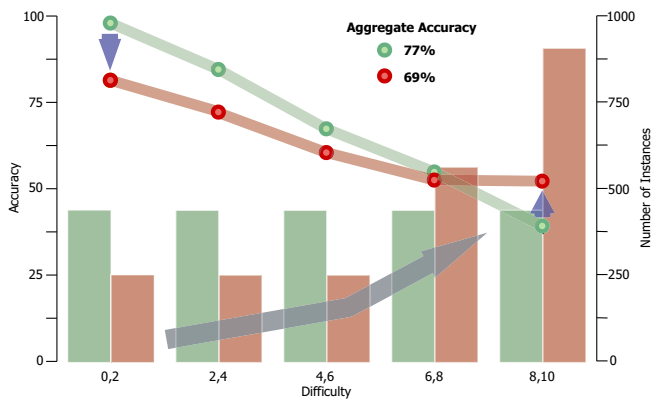

Figure 1: Performance of a CNN on CIFAR10 controlling for difficulty. Bottom: the original (green) and adversarial (red) difficulty distributions. Top: the characteristic curves where performance is binned by difficulty. The first effect we expect (grey arrow) is aggregate performance to decrease (as it does in this case) on the adversarial benchmark simply because there are more examples that are difficult. The second effect (blue arrows) is due to models being now better on the region of higher difficulty as more examples in this area are used for training, but sometimes worse on the easy instances. Aggregate accuracy cannot disentangle these two effects.

* Corresponding Author. Email: bmehrba@upv.es.

of constructing these adversarial benchmarks introduces many subtleties and potential contaminants that must be carefully considered to avoid evaluations based on the assumption that the data come from the original distribution.

Although the idea behind all of the above interventions is to keep the field of AI progressing and prevent stagnation, the effectiveness of this adversarial benchmarks can be compromised. The benchmark distribution is no longer a good representation of the problem, and the performance of the AI system on the benchmark (even if made larger and larger) is no longer a good proxy for performance on the original problem. This can occur in a variety of ways. For example, and focusing on natural adversarial examples, if the training data is populated with these challenging examples, it may end up being biased or not adequately covering the range of possible inputs. If this happens, the model may not generalise well to new data. On top of this, if these challenging examples have been recognised by being failures on some previous systems, this creates a loop of causal contamination, since the new benchmarks depend on previous systems and benchmarks, and so will the new systems created and evaluated in this context. By selecting or creating more difficult instances, the training data may become biased. We use the term 'adversarial benchmark contamination' for this specific kind of bias.

Figure 1 (bottom histograms) shows the difficulty density plot for two distributions: the original problem distribution (green) and the adversarial benchmark distribution (red). Aggregate accuracy (77% and 69% respectively) can be seen as a product of this probability and the results per difficulty (the curves on the top). In other words, we analyse the issue of adversarial benchmark contamination by decomposing the natural adversarial examples into bins based on their difficulty, and evaluating the model's performance on each bin separately. This can be done by using system characteristic curves (SCC) [22, 20, 23, 21], inspired by the concept of person characteristic curve previously developed in Item Response Theory (IRT) [8, 5]. A SCC plots the response probability (average accuracy) of a particular model as a function of instance difficulty. This makes it possible to compare models in a more insightful way. Actually, it can help disentangle different effects, such as the decrease in aggregate performance due to a higher proportion of difficult examples in the test set, but the increase of performance for more difficult examples because they had a higher proportion too in the training data. We will also show that if the adversarial examples are i.i.d. from the original distribution conditioned to difficulty, the characteristic curves allow us to recover the performance on the original distribution.

We will analyse all this theoretically, confirm it empirically, and show how different effects are disentangled in a few image classification scenarios. The main findings and contributions of this paper are:

- A theoretical analysis decomposing aggregate performance as a weighted sum of the partial performance per difficulty, showing how we can construct an adversarial benchmark whose original distribution is 'recoverable' just by weighting the difficult instances differently.
- An empirical analysis where we provide and implement different methodologies for the construction of adversarial benchmarks (using different computer vision benchmarks) and analyse (and evaluate) the impact of models trained with them on the original distribution.
- The capability of disentangling the negative effect of having a higher number of difficult instances from the positive effect of having trained the model on more difficult examples.
- The finding that there are cases with a performance loss for the easy examples, especially when the number of easy examples left in the adversarial benchmark is small.

All this comes with some take-away recommendations when using adversarial benchmarks at the end of paper.

2 Background

Progress in AI is undeniable, but how much of it remains when the conditions change is more debatable. Criticisms go from the well-known effects such as the Clever Hans phenomenon [28], appearing repeatedly in AI systems (e.g., [4]), to the outright denial of any progress in performance at all (e.g., [32, 10]). There are mainly three main ways of looking at the problem of changing conditions in evaluation. (1) A lack of generalisation to *out-of-distribution data* [1]. However this does not explain why systems do not generalise well when no change of distribution happens. (2) A possible explanation for this is simply blaming this to small benchmarks. Hence, a common practice is to make benchmarks broader, so covering a wider sample from the distribution. One option is to *replace the benchmark by a much larger one* [29, 36]. (3) But larger does not mean broader. As adding more unconditioned examples does not necessarily broaden the sample in the right direction, there have been platforms to extend benchmarks only in the direction where the model fails (because it does not generalise well). This is known as *adversarial testing or evaluation* [12, 13, 7]. In what follows, we cover these three main concepts-

2.1 Out of distribution testing

A machine learning system will have more robustness and practical value if it shows an ability to generalise, i.e., to produce relevant outputs for data beyond its training set. While most datasets are built following the independent and identically distributed (i.i.d.) assumption, in many cases this assumption is violated, and in many ML systems their generalisation capacity is simply related to the fact that the system has captured the idiosyncrasies of the dataset, including spurious correlations that manifest in both the training and test sets [31, 17]. Out-of-distribution (OOD) testing is increasingly popular for evaluating a machine learning system's ability to generalise beyond the biases of a training set. Focusing too much on the target leads to results that are unrealistic and unreliable, because in many cases research teams come up with methods, techniques and approaches that do perform well on the OOD dataset, but the model ends up relying on spurious cues and shortcuts to reach the desired target. In other words, the metrics of the benchmark are gamed in a way that defeats its original, well-intended purpose. Instead of making advances towards generalisation, the performance on specific datasets has been treated as a standalone objective [31, 17]. In this paper we want to clearly distinguish the loss of performance because the distribution really changes or because the examples are more difficult, but belonging to the same distribution.

2.2 Benchmarks replaced by other benchmarks

While it took 15 years before models reached superhuman performance on the MNIST dataset, for imageNet this period was reduced to 6 years, for SQuAD 1.1 two years, and for SQuAD 2.0 and GLUE only one year [15]. This phenomenon, which is referred to as a "challenge-solve-and-replace" [29] or a "dataset-solve-and-patch"

[36] dynamics, encourages the introduction of bigger datasets. The new benchmarks are usually richer, and training and evaluating the model in these new benchmarks leads to further generalisation. The problem arises when the same model is evaluated on the new distribution (represented by the new benchmark) and the original distribution (represented by the old benchmark). Because the metrics of performance are linked to the distribution in which they are calculated, the results of these new models do not hold when testing on the original distribution. This is one of the issues we analyse in this paper.

2.3 Adversarial testing

Replacing or extending datasets to make benchmarks more challenging can be performed in a more gradual and systematic way. This is actually what adversarial data collection (ADC) [14] and dynamic adversarial data collection (DADC) [34] represent, and they are gaining traction, especially in the NLP community. In the hope of building models that rely less on part of the distribution models are usually successful, ADC uses a human workforce to interact with the model in real-time trying to produce instances that elicit incorrect predictions. In practice, though, ADC and DADC do not always lead to robust models. Moreover, while such models usually perform better on other adversarial datasets, the results for a diverse collection of out-of-domain evaluation sets are not promising [33, 14]. These problems are beginning to be realised but we lack the tools to properly see some of the underlying effects. This is another issue that we address in this paper.

3 Controlling for Difficulty

Let us start with a theoretical analysis of a shift in difficulty where the conditional probability on difficulty is maintained. Let us consider a class of AI systems Π and a class of items M, representing problems or tasks to be solved. Given a subject $\boldsymbol{\pi} \in \Pi$ to be evaluated and an item $\boldsymbol{\mu} \in \mathrm{M}$, we want an evaluation function $\hat{R}(\boldsymbol{\pi}, \boldsymbol{\mu})$ estimating the result of $\boldsymbol{\pi}$ on $\boldsymbol{\mu}$. We want this value to be as close as possible to the actual expected result $R(\boldsymbol{\pi}, \boldsymbol{\mu})$. As there are infinitely many subjects and tasks, pre-evaluating all combinations is not feasible, and we need to rely on some features from which we could infer or extrapolate. These features can appear originally (e.g., the pixels in an image or the number of parameters of a neural network) or can be inferred from intrinsic (theoretical) analysis or extrinsic (empirical) evaluations.

One key latent feature that can be extracted is the *difficulty* of an item, denoted by $\hbar(\boldsymbol{\mu})$). Instance difficulty can be defined as a metric $\hbar$ that decreases with the expected performance R for a customary system. A good difficulty metric $\hbar$ would maximise the following expected probability:

$$\mathbb{E}_{\boldsymbol{\mu}_i, \boldsymbol{\mu}_j, \boldsymbol{\pi}}[\hbar(\boldsymbol{\mu}_i) < \hbar(\boldsymbol{\mu}_j) \Rightarrow R(\boldsymbol{\pi}, \boldsymbol{\mu}_i) > R(\boldsymbol{\pi}, \boldsymbol{\mu}_j)] \qquad (1)$$

with $\boldsymbol{\mu}_i, \boldsymbol{\mu}_j$ being items sampled from M and $\boldsymbol{\pi}$ sampled from Π, according to a reference distribution of instances and systems respectively. It is important to note that a difficulty predictor is not required; instead, any metric of individual instance difficulty can be used. If an intrinsic difficulty metric (e.g., image blur or clutter) is available, it can be used directly. In most cases, however, access to such a metric is limited and a population metric must be used. This is why the populational nature of Eq. 1 is convenient in many areas of AI, since various techniques are typically applied before selecting a system for deployment, and discarded suboptimal systems can be reused for the calculation of difficulty as the average error for each instance, or with more complex approaches such as *Instance Hardness* metrics [30] or Item Response Difficulty (IRT) [8, 5]. Finally, note that, for new benchmarks, this population difficulty is calculated using a set of models applied to the new dataset, but we must ensure that these models were built using techniques introduced before the benchmark evolved adversarially over time.

3.1 Additively-Aggregated Performance

Going back to the estimation of $\hat{R}(\boldsymbol{\pi}, \boldsymbol{\mu})$, the most common and easiest—yet unrealistic—way to estimate this is to assume one single performance estimator for all instances: $\hat{R}(\boldsymbol{\pi}, \cdot)$. Then, we calculate expected performance on the distribution as a weighted sum on the probability of each instance, as follows:

$$\hat{R}(\boldsymbol{\pi}, \boldsymbol{\mu}) \simeq \hat{R}(\boldsymbol{\pi}, M, p) \stackrel{\text{def}}{=} \sum_{\boldsymbol{\mu}' \in \mathrm{M}} p(\boldsymbol{\mu}') R(\boldsymbol{\pi}, \boldsymbol{\mu}') \qquad (2)$$

3.2 Performance Decomposition by Difficulty

If adversarial evaluation shifts the difficulty distribution, we should control for it. To do so, we can break down performance by difficulty:

$$\begin{aligned} \hat{R}(\boldsymbol{\pi}, M, p) &= \sum_{\boldsymbol{\mu} \in \mathrm{M}} p(\boldsymbol{\mu}) R(\boldsymbol{\pi}, \boldsymbol{\mu}) \\ &= \sum_{h} \sum_{\boldsymbol{\mu} \in \mathrm{M}, \hbar(\boldsymbol{\mu}) = h} p(\boldsymbol{\mu}|h) p(h) R(\boldsymbol{\pi}, \boldsymbol{\mu}) \\ &= \sum_{h} p(h) \sum_{\boldsymbol{\mu} \in \mathrm{M}} p(\boldsymbol{\mu}|h) R(\boldsymbol{\pi}, \boldsymbol{\mu}) \\ &\stackrel{\text{def}}{=} \sum_{h} p(h) R(\boldsymbol{\pi}, M, p|h) \end{aligned} \qquad (3)$$

The first step is by Bayes' rule, but as we are assuming $\hbar(\boldsymbol{\mu}) = h$, h is determined by $\boldsymbol{\mu}$ and hence the denominator $p(h|\boldsymbol{\mu}) = 1$. The second step is just by realising that $p(\boldsymbol{\mu}|h) = 0$ for any μ that has other difficulties.

The previous derivation shows that aggregate performance can be decomposed as a weighted sum of the partial performances per each difficulty, the newly defined $R(\boldsymbol{\pi}, M, p|h)$ in the final step. These are the points of the *system characteristic curve* (SCC). SCCs are inspired by the concept of person characteristic curve previously developed in IRT. The red and green curves in Fig. 1 are SCCs.

From the previous decomposition, we can see the following result follows:

Proposition 1. *Given two distributions p_1 and p_2 such that the instance conditional probability on difficulty is equal, i.e., $p_1(\boldsymbol{\mu}|h) = p_2(\boldsymbol{\mu}|h)$, then the characteristic curves are equal.*

This derives directly from Eq. 3 and the definition of $R(\boldsymbol{\pi}, M, p|h)$. The corollary of this is that given the full characteristic curve for p_1, under this same assumption of equal difficulty-conditional probabilities, we can calculate the actual $\hat{R}(\boldsymbol{\pi}, M, p_2)$ for p_2 and vice versa. This is a way of seeing aggregate performance as the area of a SCC with a weighted transformation of the x-axis. Note that the area under the SCC assumes $p(h)$ uniform, but this does not hold in general.

The direct application to our case is that if p_{Orig} is the original distribution and p_{Adv} is an adversarial distribution built in such a way that $p_{\text{Orig}}(\boldsymbol{\mu}|h) = p_{\text{Adv}}(\boldsymbol{\mu}|h)$, then we can calculate performance on the original distribution from the SCC of the adversarial distribution.

Based on Eq. 3, adversarial benchmark construction only needs to modify $p(h)$, by making difficult instances more frequent. If this is done by choosing higher difficulties and sampling from the original space M using $p(\boldsymbol{\mu})$, then the aggregate performance changes, but the agent characteristic curve does not. Accordingly, if we know how $p(h)$ has changed, we can always go back and forth from the characteristic curve and aggregate performance. This kind of modification of the distribution is 'recoverable'.

However, on many occasions, the difficult instances are not chosen from $p(\boldsymbol{\mu})$, but usually constructed or selected in a very particular way, clearly distorting $p(\boldsymbol{\mu}|h)$. If we cannot disentangle how $p(\boldsymbol{\mu}|h)$ has been modified precisely, then this kind of modification is not 'recoverable', and we will not be able to know how $\boldsymbol{\mu}$ will behave for the original distribution.

4 Empirical Analysis

We now conduct an empirical analysis to determine whether the decomposition is capable of disentangling several confounding effects that happen when evaluating models on adversarial datasets. In this sense, for an illustrative combination of benchmarks and models, we will carry out a confirmatory analysis to verify that we obtain the same SCCs for the original dataset and the adversarial dataset when keeping the same $p(\boldsymbol{\mu}|h)$, also evaluating whether we can extrapolate the aggregate result on the adversarial test data to the original distribution without measuring it directly. Furthermore, we will disentangle and interpret the different positive and negative effects regarding the individual increase in performance for harder instances, the overall decrease in performance caused by having more of the hard instances in the test set, or the individual decrease in performance for easy instances when the original distribution changes.

4.1 Methodology

In order to perform the analysis we are going to consider three methods that keep $p(\boldsymbol{\mu}|h)$ and one that deviates slightly. These methods can increase the number of difficult instances chosen from $p(\boldsymbol{\mu})$ [14] and of course remove easy instances provided this is performed randomly [25] [2].

In order to demonstrate the impact of different adversarial data collection (ADC) approaches on evaluation, we use three illustrative image recognition datasets (see Table 1). We put a special emphasis on this domain and classification problems for several reasons. Firstly the populational difficulty metric for these datasets is available from previous work [19]. Secondly, the number of instances in these datasets are big enough to construct the datasets we need as shown in Figure 2. Finally, these datasets are very common datasets for image classification tasks.

Dataset	#inst	#feat	Description	Difficulty
CIFAR10	60K	3072	32x32 colour images, 10 classes of objects	
Fashion-MNIST	70K	784	Zalando's 28x28 article images	
MNIST	70K	784	Database of 28x28 handwritten digits	

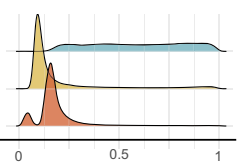

Table 1: Datasets categorised by name, number of instances, number of features, and difficulty distribution (1 − average error).

For the generation of the adversarial versions of the above datasets, we create four modified datasets following different methodologies to closely examine the effect of ADC on the evaluation results (see Fig. 2 for a summary). We refer to a random sample from the Original dataset as D_{Orig} and use it as a baseline dataset. We then create a Simple Adversarial dataset D_{SAdv}, which consists of two halves: one from the original distribution and the other half consisting of the hardest instances from the held-out part of the original distribution ($p(\boldsymbol{\mu})$). The third dataset, which we call Balanced Adversarial dataset (D_{BAdv}), is constructed in the same way as D_{SAdv}, but the hardest instances contain the same number of instances for each class. Finally, we introduce a Double Adversarial dataset (D_{DAdv}), which is created by first sampling the easiest instances from D_{Orig}, selecting a random set of the size equal to half the size of D_{Orig} and then adding the same number of the hardest instances from the held-out part of the original distribution. In all adversarial cases we sample the hard instances by adding them from $p(\boldsymbol{\mu})$. Hence, we can examine if controlling by difficulty holds in practice.

Figure 3 shows the distribution of instances per difficulty range for those adversarial datasets (D_{SAdv}, D_{BAdv}, D_{DAdv}) compared to the original one D_{Orig} (basically uniform). This information helps us understand by how much the number of instances increases for the higher difficulty ranges (in all cases: D_{SAdv}, D_{BAdv}, D_{DAdv}) and how much it decreases for the lower ones (only for D_{Dadv}).

For each adversarial dataset, we train two classical classification techniques: a convolutional neural network (CNN) and a simple neural network (NN), a fully-connected multi-layer perceptron. We wanted to train a more complex and powerful model and a simpler and lighter one to analyse situations where almost performance is saturated and other cases where performance is still far from ideal. The CNN we train for both MNIST and FashionMNIST datasets consists of two convolutional layers and two fully connected layers with ReLU activation function and pooling layers in between. The model has a total of 225,034 parameters. We train the models for 30 epochs . For CIFAR10 the CNN model that we train consist of six convolutional layers with max-pooling and normalisation layers in between, followed by two fully connected layers. The total number of parameters for this network is nearly 2.4 million. In this case the model is trained for 50 epochs. In case of simple neural network (NN), which is a Multi-layer Perceptron, the network consists of only two fully connected layers with total parameters of 3,985 for MNIST and FashionMNIST datasets and we trained them for 30 epochs. For CIFAR10, the model consists of six fully connected layers with a total number of 2.49 million parameters, trained for 100 epochs. The choice on number of epochs was guided by both literature precedents and our own experiments to ensure that the models achieved satisfactory performance without overfitting.

We evaluate all models on the corresponding test set as a random sample extracted from the modified dataset, with the sample size equating 10% of the training set. We also evaluate all three adversarially built models on the original dataset (D_{Orig}). For all the aforementioned models, we perform 20 cross-validation repetitions to obtain reliable estimates of model performance given the size of the datasets.. We choose categorical cross-entropy as loss function and we set batch size to 128.

For our experiments, we use a populational difficulty metric defined as the average error for each instance (provided in [19]). See further discussion of the difficulty of the datasets addressed in the appendix [24] (A.1). We scale the difficulty values between 0 and 10 to improve intelligibility. Results are shown using SCCs. For their generation, we divide the instances in 5 bins according to difficulty

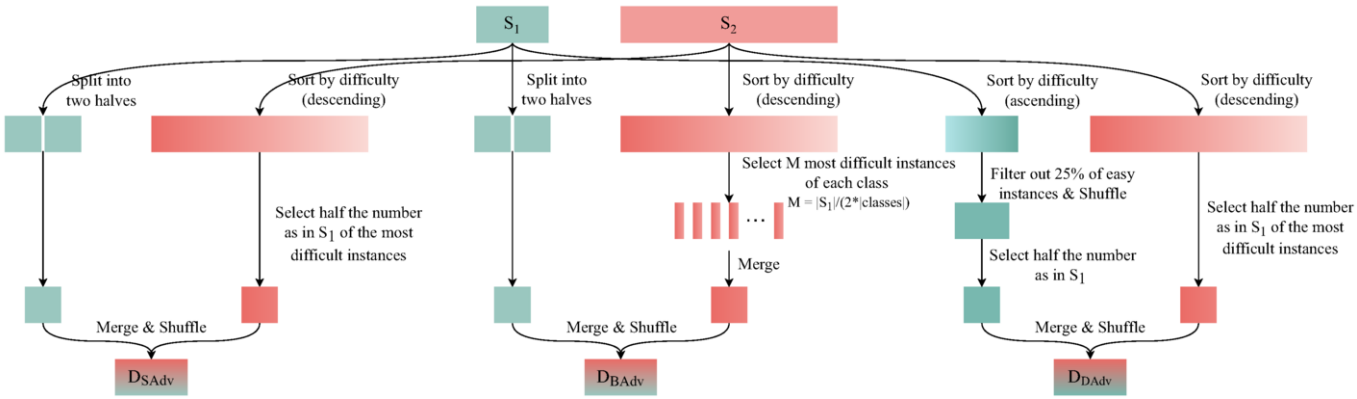

Figure 2: Dataset construction: The main dataset is divided into two non-overlapping sections, S_1 and S_2, which comprise 25% and 75% of the dataset, respectively. We use S_1 as D_{Orig}. The construction process for the Simple Adversarial dataset (D_{SAdv}) is illustrated on the left, Balanced Adversarial dataset (D_{BAdv}) in the centre, and Double Adversarial dataset (D_{DAdv}) on the right.

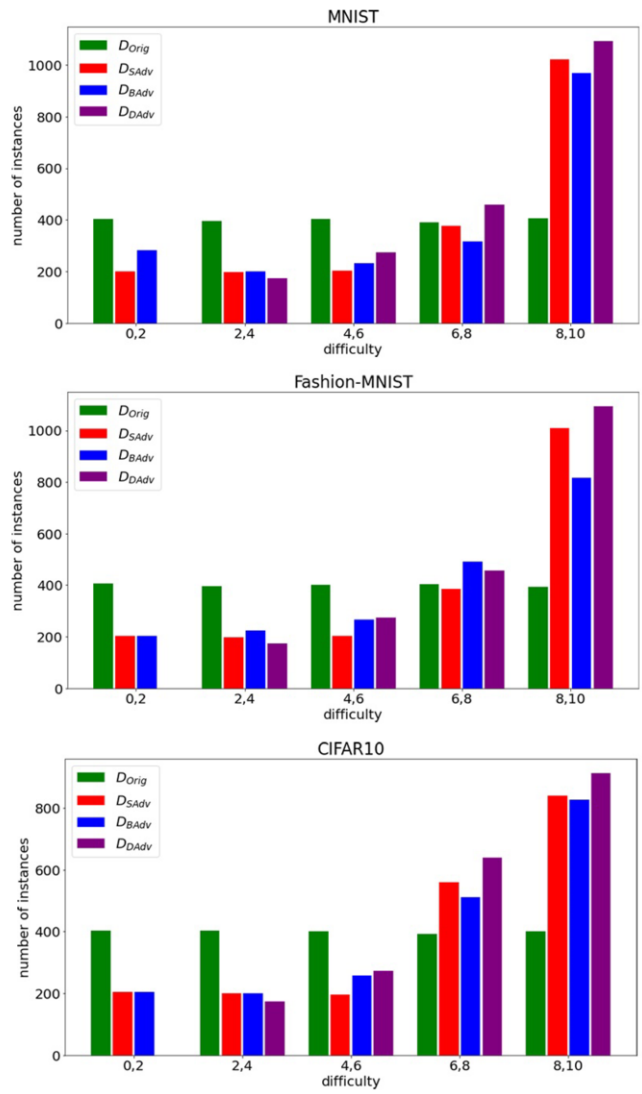

Figure 3: Distribution of instances per difficulty bin for the original (D_{Orig}) and all adversarial datasets (D_{SAdv}, D_{BAdv}, D_{DAdv})

range[1]. The first bin contains the instances with difficulty level between 0 to 2. This range for the second, third, forth and fifth bin is 2-4, 4-6, 6-8 and 8-10 respectively. For each bin, we plot the difficulty on the x-axis and we plot the average accuracy of the instance in the bins on the y-axis. See figure 8 in the appendix for some illustrative sample images at each difficulty level for each data set.

4.2 Results

If we look at the aggregate results, as shown in Table 2, we see that the accuracy for the adversarial datasets is worse than for the original dataset. This means that from the two effects in Fig. 1, the difficulty shift in grey dominates over the blue effect. A much more insightful view appears when we look at Fig. 4, showing different behaviours depending on the difficulty. In general, the performance of all adversarial models is below the performance of the original model (in green) for the first three/four bins, but higher for the last one/two bins.

We can see that, in general, the behaviours of the solid and dashed curves are very similar[2], with the only exception of the final point for M_{DAdv} in MNIST and most of the curves for M_{BAdv} for FashionMNIST. This second case might be caused by a high imbalance in those ranges of difficulty (see Fig. 3), which affects the evaluation in those ranges when the dataset is balanced. This means that when looking at the SCCs, if we keep the same $p(\boldsymbol{\mu}|h)$ for the generation of adversarial datasets, the use of the adversarial or original dataset for evaluation is irrelevant. The exception to the above are those cases where there is a class re-balance per difficulty, which happens for M_{BAdv}, and it's more noticeable when there is originally more imbalance.

In turn, we can as well estimate the aggregated result of testing the adversarial model for the original distribution without measuring it directly. Table 4 demonstrates that the performance of adversarial models on the original distribution can be accurately recovered by calculating it using the model's SCC and the difficulty distribution of

[1] The bias-variance decomposition is related to the uncertainty of the difficulty metric and helps to optimise the metric by identifying the optimal bin size to minimise overall error. Increasing the number of bins reduces bias but increases variance due to smaller bin sample sizes. In our case, using five bins proved suitable for balancing bias and variance while providing valuable insight into model performance on adversarial benchmarks.

[2] Note that, for M_{DAdv},D_{DAdv} curve (solid violet) there is a discontinuity for difficulty range {0,2}. This is due to the construction of the dataset D_{DAdv} for which we undersample the easiest instances from D_{Orig}, so there are no instances in this bin (see Fig. 3).

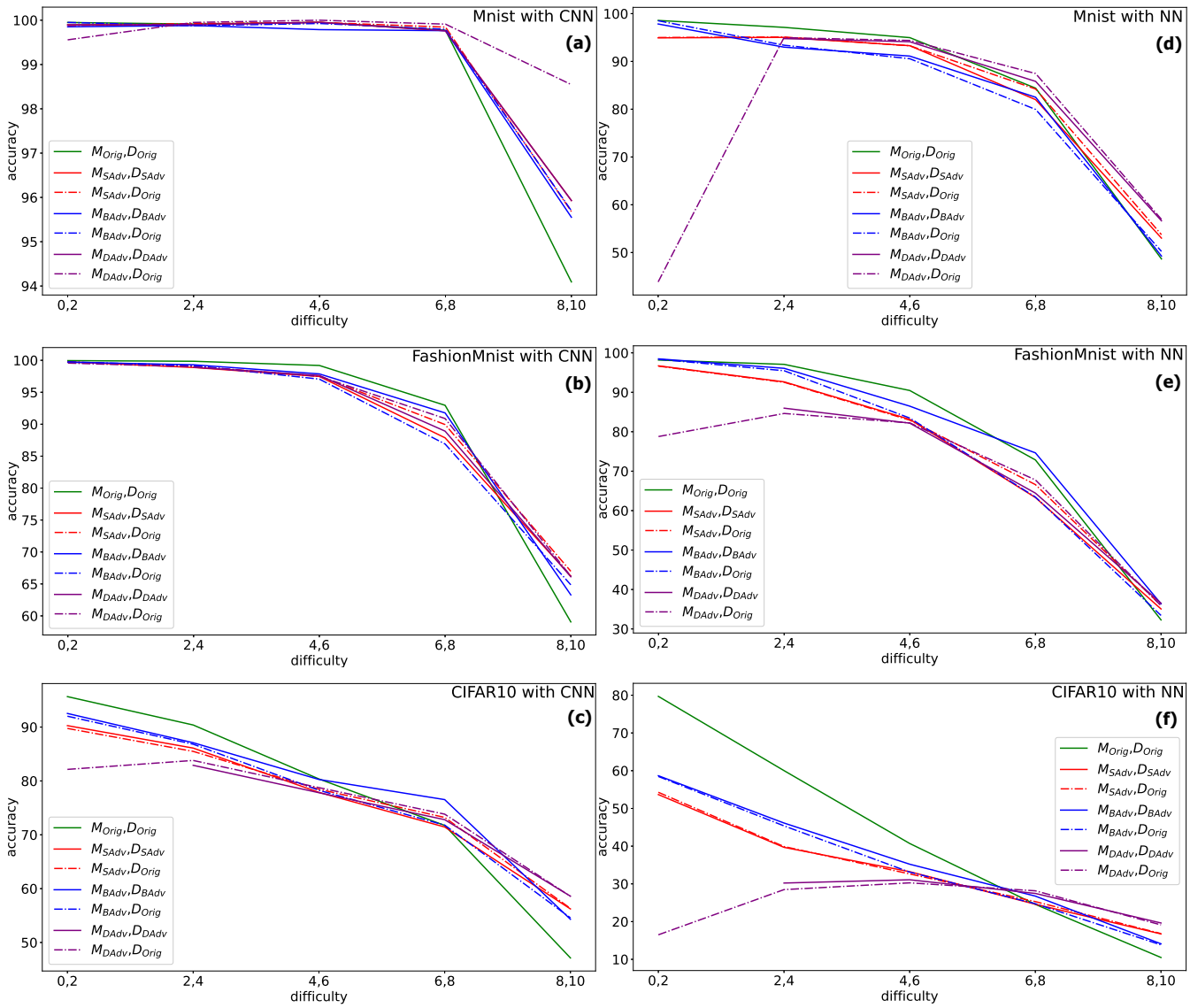

Figure 4: SCC of all models trained on datasets constructed from MNIST (a & d), FashionMNIST (b & e) and CIFAR10 (c & f) using CNN (top) and NN (bottom). The tuples $\langle M_a, D_b \rangle$ indicate that the model has been trained on dataset a and tested on dataset b (with $a, b \in \{\text{Orig}, \text{SAdv}, \text{BAdv}, \text{DAdv}\}$).

		$M_{\text{Orig}},D_{\text{Orig}}$	$M_{\text{SAdv}},D_{\text{SAdv}}$	$M_{\text{SAdv}},D_{\text{Orig}}$	$M_{\text{BAdv}},D_{\text{BAdv}}$	$M_{\text{BAdv}},D_{\text{Orig}}$	$M_{\text{DAdv}},D_{\text{DAdv}}$	$M_{\text{DAdv}},D_{\text{Orig}}$
MNIST	CNN	98.71 ± 0.30	97.85 ± 0.35	99.04 ± 0.21	97.75 ± 0.44	99.03 ± 0.17	97.72 ± 0.27	99.06 ± 0.25
	NN	84.65 ± 1.63	70.94 ± 2.22	84.19 ± 2.19	70.59 ± 2.21	82.45 ± 1.83	71.77 ± 1.83	75.53 ± 2.06
FashionMNIST	CNN	90.32 ± 0.65	80.21 ± 0.73	90.74 ± 0.67	82.61 ± 0.74	89.65 ± 0.44	78.50 ± 1.08	90.67 ± 0.70
	NN	78.34 ± 2.43	57.34 ± 3.10	75.08 ± 4.22	65.41 ± 1.22	74.97 ± 2.10	53.47 ± 3.09	69.90 ± 5.92
CIFAR10	CNN	77.11 ± 1.28	69.05 ± 1.24	76.69 ± 1.06	70.51 ± 1.59	76.75 ± 0.91	67.87 ± 1.46	75.43 ± 1.55
	NN	43.24 ± 0.77	26.60 ± 1.47	33.84 ± 1.75	27.81 ± 1.05	35.18 ± 1.08	24.62 ± 0.79	24.49 ± 1.24

Table 2: Aggregate performance and standard deviation for models trained on datasets constructed from MNIST, FashionMNIST and CIFAR10 using CNN and NN. Notation as in Figure 4.

the original dataset, if we compare these values with the D_{Orig} dataset in Table 2. Again, with the exception of those adversarial sampling methods that modify the class balance (see the difference for the NN in FashionMNIST between the actual one, 74.97 and the estimated one, 78.75.

In addition, for all tasks, models and samplings in Fig. 4 we see the decrease in performance on the last bins. This is sufficiently important to obscure that most curves get better than the green one in that

		M_{Orig},D_{Orig}	M_{SAdv},D_{SAdv}	M_{SAdv},D_{Orig}	M_{BAdv},D_{BAdv}	M_{BAdv},D_{Orig}	M_{DAdv},D_{DAdv}	M_{DAdv},D_{Orig}
MNIST	CNN	99.16	99.38	99.37	99.28	99.35	99.38	99.72
	NN	87.53	86.07	86.74	85.03	84.57	81.23	81.80
FashionMNIST	CNN	92.87	91.80	92.46	92.60	91.36	92.06	92.56
	NN	81.40	76.24	77.09	81.15	77.06	72.56	72.99
CIFAR10	CNN	78.47	77.16	77.55	79.32	77.54	75.97	76.68
	NN	42.59	33.17	33.34	36.12	34.84	26.70	26.18

Table 3: Area under the curve (SCC) for models trained on datasets constructed from MNIST, FashionMNIST and CIFAR10 using CNN and NN. Notation as in Figure 4.

		M_{SAdv}, $D_{\text{SAdv}\to\text{Orig}}$	M_{BAdv}, $D_{\text{BAdv}\to\text{Orig}}$	M_{DAdv}, $D_{\text{DAdv}\to\text{Orig}}$
MNIST	CNN	99.07	98.95	99.07
	NN	83.56	82.64	74.88
FashionMNIST	CNN	90.13	90.49	90.32
	NN	74.32	78.75	69.68
CIFAR10	CNN	76.42	78.17	74.87
	NN	33.58	36.15	24.94

Table 4: Recovered performance of adversarial models on original distribution: Having the characteristic curve of a model in hand, we can calculate the performance of the (adversarial) model on the original distribution without actually testing it.

area. In particular, M_{SAdv} and M_{DAdv} are especially good at the end, but as we can see in Table 2, not sufficiently to compensate for the loss in accuracy in these bins on aggregate accuracy, not even in area under the SCC. While the full insight can be obtained when looking at the histogram of difficulties (see Fig. 3), we can clearly disentangle the positive effect of the increase in performance for harder instances from the negative effect of having a high proportion of hard instances in the test.

Finally, we see some other effects. For instance, a significant decline for M_{DAdv} in the first bin is observed in almost all the plots. This is because D_{DAdv} is created by first undersampling the easiest instances from D_{Orig}, which results in less contribution of these instances for the loss and a worse fit for them (see Figures 7 (d) in the Appendix). This is very pronounced in some cases, especially those with NN in Fig. 4. But this is a more general phenomenon, happening when the curves detach from 100% performance. For instance, FashionMNIST is dominated by the original model (green line) in medium difficulties, and the same happens for MNIST with NN. In total, the areas of the curves (see Table 3) are seldom better than the original model, and shifting towards the right would only make this worse (see appendix A.4 for further analysis of model performance). Beyond any generic pattern, the SCCs allow us to precisely analyse where there are gains in performance and where there are losses.

5 Discussion

The analysis of experimental results with aggregate metrics makes sense as far as the reference distribution is representative of the problem we want to solve. For instance, if we want to evaluate a self-driving car on the distribution of journeys that happened last year in a particular country, the average of some metric of success is an estimate of the expected value for that metric. This is informative and can lead to decisions about what technology is better than the rest. However, this is no longer valid if the distribution is changing. In many cases we cannot anticipate how this distribution is going to change, so this is still a hopeful bet for evaluation.

However, adversarial testing changes the distribution in a very specific, systematic way, not because the target problem is changing, but because we want to make systems better. Following with the previous example, if the distribution of journeys has not changed, we should not change the distribution to test our systems. Doing that on the adversarial dataset for testing is *wrong if we still evaluate by unweighted aggregate metrics*. What we have shown in this paper is that aggregating is *right if controlled by what is shifting the distribution, which is difficulty*. This leads to the SCCs, which is a very convenient way of doing this and extrapolating back to the original distribution, or even to other distributions where the difficulties change. As far as the dataset modification (including making it adversarial) does not distort $p(\boldsymbol{\mu}|h)$, we can do all these extrapolations.

We hence recommend SCCs to summarise results whenever there is an adversarial situation. If only one number is preferred, at least we should choose the area under the SCC and not accuracy, although the curve provides more valuable information, particularly regarding trends. Actually, we see that some adversarial methods are slightly better on the original distribution in Table 2 (CNN for MNIST and FashionMNIST) but this is even less clear for the areas under the curves (Table 3). We might also consider creating multiple plots for each difficulty bin (e.g., in the case of two-point summary metrics, such as refinement and calibration, we would have two curves instead of one). Although this approach might be less visually appealing than a single curve for a one-point metric, it might provide more detailed insights. Furthermore, this approach could be extended to conditional density estimators by plotting the errors for the estimated mean and variance for each example, with the data binned by level of difficulty.

This work has made some assumptions and presents some limitations. First, difficulty is estimated from a battery of techniques, and we should be well aware about the nature of the x-axis. Second, the scale and binning of this same x-axis may lead to different conclusions if we look at the area, so it is better to look at the trend. Third, we kept the size of the datasets fixed to keep similar training conditions (the focus of this paper was not to see what adversarial training setting is best but to determine good adversarial testing settings).

The new methodology and insights lead to several possibilities for future work beyond the natural extension of these results to other domains and types of tasks. We would like to explore the effect of these benchmark contaminations when adversarial training and testing are performed iteratively. Also, we would like to explore other ways of introducing the adversarial examples where the distribution is not preserved, but the changes are traceable and invertible, so that the expected performance on the original dataset can be recovered as well.

Acknowledgments

We thank the anonymous reviewers for their comments. This work was funded by valgrAI, the Norwegian Research Council grant 329745 Machine Teaching for Explainable AI, the Future of Life Institute, FLI, under grant RFP2-152, the EU (FEDER) and Spanish grant RTI2018-094403-B-C32 funded by MCIN/AEI/10.13039/501100011033 and by CIPROM/2022/6 funded by Generalitat Valenciana, EU's Horizon 2020 research and innovation programme under grant agreement No. 952215 (TAILOR), US DARPA HR00112120007 (RECoG-AI) and Spanish grant PID2021-122830OB-C42 (SFERA) funded by MCIN/AEI/10.13039/501100011033 and "ERDF A way of making Europe" In compliance with the recommendations of the Science paper about reporting of evaluation results in AI [3], we include all the results at the instance level (https://github.com/Behzadmeh/AdverserialDatasets).

References

[1] Martin Arjovsky, *Out of distribution generalization in machine learning*, Ph.D. dissertation, New York University, 2020.

[2] Eli Bronstein, Sirish Srinivasan, Supratik Paul, Aman Sinha, Matthew O'Kelly, Payam Nikdel, and Shimon Whiteson, 'Embedding synthetic off-policy experience for autonomous driving via zero-shot curricula', *arXiv preprint arXiv:2212.01375*, (2022).

[3] Ryan Burnell, Wout Schellaert, John Burden, Tomer D Ullman, Fernando Martinez-Plumed, Joshua B Tenenbaum, Danaja Rutar, Lucy G Cheke, Jascha Sohl-Dickstein, Melanie Mitchell, Douwe Kiela, Murray Shanahan, Ellen M. Voorhees, Anthony G. Cohn, Joel Z Leibo, and Jose Hernandez-Orallo, 'Rethink reporting of evaluation results in AI', *Science*, **380**(6641), 136–138, (2023).

[4] Brandon Carter, Siddhartha Jain, Jonas W Mueller, and David Gifford, 'Overinterpretation reveals image classification model pathologies', *Advances in Neural Information Processing Systems*, **34**, 15395–15407, (2021).

[5] Rafael Jaime De Ayala, *Theory and practice of item response theory*, Guilford Publications, 2009.

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei, 'Imagenet: A large-scale hierarchical image database', in *2009 IEEE CVPR*, pp. 248–255. IEEE, (2009).

[7] Desmond Elliott, 'Adversarial evaluation of multimodal machine translation', in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2974–2978, (2018).

[8] S. E. Embretson and S. P. Reise, *Item response theory for psychologists*, L. Erlbaum, 2000.

[9] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy, 'Explaining and harnessing adversarial examples', *arXiv:1412.6572*, (2014).

[10] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song, 'Natural adversarial examples', in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15262–15271, (2021).

[11] José Hernández-Orallo, Marco Baroni, Jordi Bieger, Nader Chmait, David L Dowe, Katja Hofmann, Fernando Martínez-Plumed, Claes Strannegård, and Kristinn R Thórisson, 'A new AI evaluation cosmos: Ready to play the game?', *AI Magazine*, **38**(3), (2017).

[12] Robin Jia and Percy Liang, 'Adversarial examples for evaluating reading comprehension systems', *arXiv preprint arXiv:1707.07328*, (2017).

[13] Anjuli Kannan and Oriol Vinyals, 'Adversarial evaluation of dialogue models', *arXiv preprint arXiv:1701.08198*, (2017).

[14] Divyansh Kaushik, Douwe Kiela, Zachary C Lipton, and Wen-tau Yih, 'On the efficacy of adversarial data collection for question answering: Results from a large-scale randomized study', *arXiv preprint arXiv:2106.00872*, (2021).

[15] Douwe Kiela, Max Bartolo, Yixin Nie, Divyansh Kaushik, Atticus Geiger, Zhengxuan Wu, Bertie Vidgen, Grusha Prasad, Amanpreet Singh, Pratik Ringshia, et al., 'Dynabench: Rethinking benchmarking in nlp', *arXiv preprint arXiv:2104.14337*, (2021).

[16] Alex Krizhevsky. Learning multiple layers of features from tiny images. https://www.cs.toronto.edu/~kriz/cifar.html, 2009.

[17] Kyungmin Lee, Hunmin Yang, and Se-Yoon Oh, 'Adversarial training on joint energy based model for robust classification and out-of-distribution detection', in *2020 20th International Conference on Control, Automation and Systems (ICCAS)*, pp. 17–21. IEEE, (2020).

[18] Fernando Martínez-Plumed, Pablo Barredo, Sean O Heigeartaigh, and José Hernández-Orallo, 'Research community dynamics behind popular ai benchmarks', *Nature Machine Intelligence*, **3**(7), 581–589, (2021).

[19] Fernando Martínez-Plumed, David Castellano, Carlos Monserrat-Aranda, and José Hernández-Orallo, 'When AI difficulty is easy: The explanatory power of predicting IRT difficulty', in *Proceedings of the AAAI Conf. Artificial Intelligence*, volume 36, pp. 7719–7727, (2022).

[20] Fernando Martínez-Plumed and José Hernández-Orallo. AI results for the Atari 2600 games: difficulty and discrimination using IRT. Evaluating General-Purpose Artificial Intelligence, August 20, 2017, 2nd Intl. Workshop held in conjunction with IJCAI, Melbourne, 2017.

[21] Fernando Martínez-Plumed and José Hernández-Orallo, 'Dual indicators to analyse AI benchmarks: Difficulty, discrimination, ability and generality', *IEEE Transactions on Games*, **12**(2), 121–131, (2020).

[22] Fernando Martínez-Plumed, Ricardo B. C. Prudêncio, Adolfo Martínez-Usó, and José Hernández-Orallo, 'Making sense of item response theory in machine learning', in *ECAI 2016 - 22nd European Conference on Artificial Intelligence, Best Paper Award*, pp. 1140–1148, (2016).

[23] Fernando Martínez-Plumed, Ricardo BC Prudêncio, Adolfo Martínez-Usó, and José Hernández-Orallo, 'Item response theory in AI: Analysing machine learning classifiers at the instance level', *Artificial Intelligence*, **271**, 18–42, (2019).

[24] Behzad Mehrbakhsh, Fernando Martínez-Plumed, and José Hernández-Orallo. Further details on examining adversarial evaluation: Role of difficulty. http://hdl.handle.net/10251/181335, 2023.

[25] Jason Phang, Angelica Chen, William Huang, and Samuel R Bowman, 'Adversarially constructed evaluation sets are more challenging, but may not be fair', *arXiv preprint arXiv:2111.08181*, (2021).

[26] Pranav Rajpurkar, Robin Jia, Percy Liang, and ., 'Know what you don't know: Unanswerable questions for squad', *arXiv preprint arXiv:1806.03822*, (2018).

[27] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar, 'Do imagenet classifiers generalize to imagenet?', in *ICML*, pp. 5389–5400. PMLR, (2019).

[28] Laasya Samhita and Hans J Gross, 'The "clever hans phenomenon" revisited', *Communicative & integrative biology*, **6**(6), e27122, (2013).

[29] David Schlangen, 'Language tasks and language games: On methodology in current natural language processing research', *arXiv preprint arXiv:1908.10747*, (2019).

[30] Michael R Smith, Tony Martinez, and Christophe Giraud-Carrier, 'An instance level analysis of data complexity', *Machine learning*, **95**(2), 225–256, (2014).

[31] Damien Teney, Ehsan Abbasnejad, Kushal Kafle, Robik Shrestha, Christopher Kanan, and Anton Van Den Hengel, 'On the value of out-of-distribution testing: An example of goodhart's law', *Advances in Neural Information Processing Systems*, **33**, 407–417, (2020).

[32] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Andrew Ilyas, and Aleksander Madry, 'From imagenet to image classification: Contextualizing progress on benchmarks', in *International Conference on Machine Learning*, pp. 9625–9635. PMLR, (2020).

[33] Eric Wallace, Adina Williams, Robin Jia, and Douwe Kiela, 'Analyzing dynamic adversarial training data in the limit', *arXiv preprint arXiv:2110.08514*, (2021).

[34] Eric Wallace, Adina Williams, Robin Jia, and Douwe Kiela, 'Analyzing dynamic adversarial training data in the limit', in *Findings of the Association for Computational Linguistics: ACL 2022*, pp. 202–217, Dublin, Ireland, (May 2022). Association for Computational Linguistics.

[35] Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman, 'Superglue: A stickier benchmark for general-purpose language understanding systems', *arXiv preprint arXiv:1905.00537*, (2019).

[36] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi, 'Hellaswag: Can a machine really finish your sentence?', *arXiv preprint arXiv:1905.07830*, (2019).

[37] Daniel Zhang, Saurabh Mishra, Erik Brynjolfsson, John Etchemendy, Deep Ganguli, Barbara Grosz, Terah Lyons, James Manyika, Juan Carlos Niebles, Michael Sellitto, et al., 'The ai index 2021 annual report', *arXiv preprint arXiv:2103.06312*, (2021).