ROBUSTBENCH: A STANDARDIZED ADVERSARIAL ROBUSTNESS BENCHMARK

Francesco Croce* University of Tübingen
Maksym Andriushchenko* EPFL
Vikash Sehwag* Princeton University
Edoardo Debenedetti EPFL
Nicolas Flammarion EPFL
Mung Chiang Purdue University
Prateek Mittal Princeton University
Matthias Hein University of Tübingen

ABSTRACT

Evaluation of adversarial robustness is often error-prone leading to overestimation of the true robustness of models. Our goal is to establish a *standardized benchmark* of adversarial robustness, which as accurately as possible reflects the robustness of the considered models within a reasonable computational budget. This requires to impose some restrictions on the admitted models to rule out defenses that only make gradient-based attacks ineffective without improving actual robustness. We evaluate robustness of models for our benchmark with AutoAttack, an ensemble of white- and black-box attacks which was recently shown to improve almost all robustness evaluations compared to the original publications. Our leaderboard aims at reflecting the current state of the art in the ℓ_∞- and ℓ_2-threat models and on common image corruptions, with possible extensions in the future. Additionally, we open-source a library that provides unified access to state-of-the-art robust models to facilitate their downstream applications. Finally, we analyze general trends in ℓ_p-robustness and its impact on other tasks such as robustness to various distribution shifts and out-of-distribution detection.

1 INTRODUCTION

Since the finding that state-of-the-art deep learning models are vulnerable to small input perturbations called *adversarial examples* (Szegedy et al., 2013), achieving adversarially robust models has become one of the most studied topics in the machine learning community: more than 3000 papers on this topic have appeared, but it is often unclear which defenses are effective. Adaptive attacks, specifically designed to test a particular defense (Athalye et al., 2018; Carlini et al., 2019; Tramèr et al., 2020), have shown that several seemingly effective defenses fail to be robust, but Tramèr et al. (2020) observe that even some published defenses which have tried to perform adaptive evaluations can still be broken by new adaptive attacks. We observe repeating patterns in defenses that prevent standard attacks from succeeding without improving robustness. This motivates us to impose restrictions on the defenses we consider in our proposed benchmark, `RobustBench`, which aims at *standardized* adversarial robustness evaluation. We start from benchmarking robustness with respect to the ℓ_∞- and ℓ_2-threat models since they are the most studied settings in the literature. We use the recent AutoAttack (Croce & Hein, 2020b) as our current standard evaluation method, which is an ensemble of diverse parameter-free attacks (white- and black-box) that has shown reliable performance over a large set of models that satisfy our restrictions. Moreover, we also accept evaluations based on adaptive attacks whenever they improve our evaluation. Additionally, we collect models robust against common image corruptions (Hendrycks & Dietterich, 2019) as these represent another type of perturbations which should not modify the decision of a classifier although they are not produced in an adversarial way.

We make the following contributions: 1) **Leaderboard** (`https://robustbench.github.io/`), a website with the leaderboards based on *more than 30* recent papers where it is possible to

track the progress and the current state of the art in adversarial robustness, 2) **Model Zoo** (`https://github.com/RobustBench/robustbench`), a collection of the most robust models that are easy to use for any downstream application and as a testbed for new attacks, 3) **Analysis** of how robust models perform on other tasks, like common corruptions or detection of out-of-distribution inputs. We believe that our standardized benchmark and accompanied collection of models will accelerate progress on multiple fronts in the area of adversarial robustness.

2 Background and related work

Adversarial perturbations. Let $\boldsymbol{x} \in \mathbb{R}^d$ be an input point and $y \in \{1, \dots, C\}$ be its correct label. For a classifier $f : \mathbb{R}^d \to \mathbb{R}^C$, we define a *successful adversarial perturbation* with respect to the perturbation set $\Delta \subseteq \mathbb{R}^d$ as a vector $\boldsymbol{\delta} \in \mathbb{R}^d$ such that $\arg\max_{c \in \{1,\dots,C\}} f(\boldsymbol{x} + \boldsymbol{\delta})_c \neq y$ and $\boldsymbol{\delta} \in \Delta$, where typically the perturbation set Δ is chosen such that *all* points in $x + \delta$ have y as their true label. Then *robust accuracy* is the fraction of datapoints on which f predicts the correct class for all possible perturbations from Δ. Since computing the exact robust accuracy is in general intractable (Katz et al., 2017), an *upper bound* on it is obtained via some *adversarial attacks*. We here focus on the white-box setting, i.e. the model f is assumed to be fully known to the attacker. We focus on ℓ_p-perturbations, i.e. $\Delta_p = \{\boldsymbol{\delta} \in \mathbb{R}^d, \|\delta\|_p \leq \varepsilon\}$, since those are the most well-studied perturbations, particularly for $p \in \{\infty, 2\}$ (Szegedy et al., 2013; Goodfellow et al., 2015; Madry et al., 2018), and use thresholds ε established in the literature. Also, despite the simplicity of the ℓ_p-perturbation model, it has numerous interesting applications that go beyond security considerations (see App. A).

Common corruptions. Unlike adversarial perturbations, common corruptions (Hendrycks & Dietterich, 2019) try to mimic modifications of the input images which can occur naturally: they are not imperceptible and evaluation on them is done in the average case fashion, i.e. there is no attacker who aims at changing the classifier's decision. In this case, the robustness of a model is evaluated as classification accuracy on the corrupted images, averaged over types and severities of corruptions.

Related libraries and benchmarks. Many libraries focus primarily on implementations of popular adversarial attacks: FoolBox (Rauber et al., 2017), Cleverhans (Papernot et al., 2018), AdverTorch (Ding et al., 2019), AdvBox (Goodman et al., 2020), ART (Nicolae et al., 2018), SecML (Melis et al., 2019). Some of them also provide implementations of several basic defenses, but they do not include up-to-date state-of-the-art models. Moreover, many benchmarks aiming at tracking progress of adversarial defenses have appeared: the challenges (Kurakin et al., 2018; Brendel et al., 2018) hosted at NeurIPS 2017 and 2018, DEEPSEC (Ling et al., 2019), the work of Dong et al. (2020), RobustML (`https://www.robust-ml.org/`). However, these prior works present some weaknesses which we aim to overcome (see a detailed discussion in App. B).

3 Description of RobustBench

`RobustBench` presents a few different features compared to the existing benchmarks: (1) a baseline worst-case evaluation with an ensemble of *strong*, *standardized* attacks optionally extended by adaptive ones, (2) clearly defined threat models, (3) inclusion of the most recent improvements such as (Gowal et al., 2020), (4) Model Zoo providing convenient access to models robust with respect to different threat models. Next, we describe the main components of `RobustBench`.

3.1 Leaderboard

Restrictions. We argue that benchmarking adversarial robustness in a standardized way requires some restrictions on the considered models to prevent submissions of defenses that cause some standard attacks to fail without actually improving robustness. Specifically, we consider only classifiers that 1) have in general *non-zero gradients* with respect to the inputs, 2) have a *fully deterministic forward pass*, 3) do not have an *optimization loop* in the forward pass. All these cases would require specific techniques and adaptive evaluations, which can be hardly standardized. Moreover, we are not aware of existing defenses solely based on such techniques which would achieve competitive robustness (see App. C.1 for further discussion). Some of these restrictions were also discussed by Brown et al. (2018) (App. E therein) for the warm-up phase of their challenge.

Figure 1: Visualization of the robustness and accuracy of 46 CIFAR-10 models from the `RobustBench` ℓ_∞-leaderboard. Robustness is evaluated using ℓ_∞-perturbations with $\varepsilon_\infty = 8/255$.

Initial setup. We set up leaderboards for the ℓ_∞ (46 models), ℓ_2 (9 models), and common corruption (7 models) threat models. For the first two, we use fixed budgets of $\epsilon_\infty = 8/255$ and $\epsilon_2 = 0.5$ on CIFAR-10 (Krizhevsky & Hinton, 2009) respectively. For common corruptions, we evaluate the accuracy on CIFAR-10-C (Hendrycks & Dietterich, 2019). We highlight the models using data beyond the training set since it gives a clear advantage for both clean and robust accuracy.

Evaluation of defenses. Currently, we perform the standardized evaluation of the reported defenses using AutoAttack (Croce & Hein, 2020b), which is an ensemble of four adversarial attacks. We choose it since it includes both black-box and white-box attacks, does not require hyperparameter tuning (in particular, the step size), and consistently improves the results reported in the original papers for almost all the models as shown in Fig. 1 (middle). If in the future some new standardized and parameter-free attack is shown to consistently outperform AutoAttack on a wide set of models given a similar computational cost, we will adopt it as a standard evaluation. For reproducibility, we perform the standardized evaluation independently of the authors of the submitted models.

Adding new defenses and adaptive evaluations. The leaderboard is useful if it reflects the latest advances in the field, so it needs to be kept up-to-date. We require new entries to 1) satisfy the three restrictions, 2) be accompanied by a paper (e.g. an arXiv preprint) describing the technique used to achieve the reported results, and 3) make checkpoints available. We also allow *temporarily* adding entries without providing checkpoints given that the evaluation is done with AutoAttack, but mark such evaluations as *unverified* and reserve the right to remove them later on if the corresponding checkpoints are not provided. To achieve the most accurate approximation of the true robustness we accept the submission of adaptive evaluations to complement the standardized one.

Adding new threat models. Our intention is to add leaderboards for other threat models which are becoming widely accepted in the community, like sparse perturbations, e.g. bounded by ℓ_0, ℓ_1-norm or adversarial patches (Brown et al., 2017; Croce & Hein, 2019; Modas et al., 2019; Croce et al., 2020), multiple ℓ_p-norms perturbations (Tramèr & Boneh, 2019; Maini et al., 2020), adversarially optimized common corruptions (Kang et al., 2019a;b). The long term goal, and the direction towards which many recent works are moving, is achieving *general* robustness (Brown et al., 2018).

3.2 Model Zoo

We collect checkpoints of many networks from the leaderboard in a single repository and, with the permission of the authors, make them easily accessible (see examples in App. C.2), as it is often time-consuming and not straightforward to integrate models from different papers in the same framework due to small variations in the architectures, custom input normalizations, etc. Currently, we include only models implemented in PyTorch (Paszke et al., 2017): 20 models trained for ℓ_∞-robustness, 8 for ℓ_2-robustness, 2 for common corruptions, and a standardly trained one as a baseline.

A testbed for new attacks. An important use case of the Model Zoo is to simplify comparisons between different adversarial attacks on a wide range of models. First, the current leaderboard can already serve as a strong baseline for new attacks. Second, new attacks are often evaluated on the publicly available models from Madry et al. (2018) and Zhang et al. (2019b), with the risk of overfitting to these classifiers, which can be prevented if many models are used.

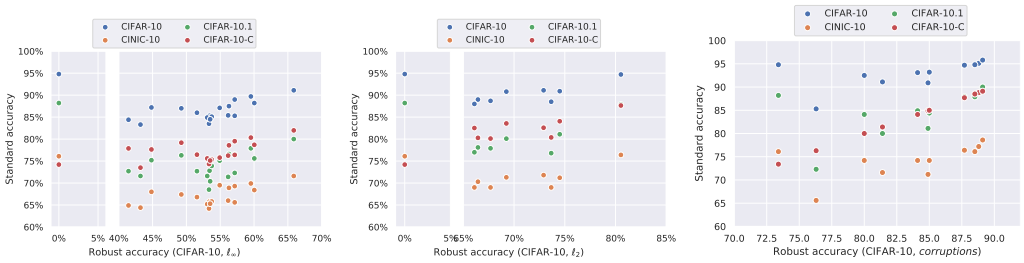


Figure 2: Standard accuracy of classifiers trained against ℓ_∞ (left), ℓ_2 (middle), and common corruption (right) threat model respectively, from our Model Zoo on various distribution shifts.

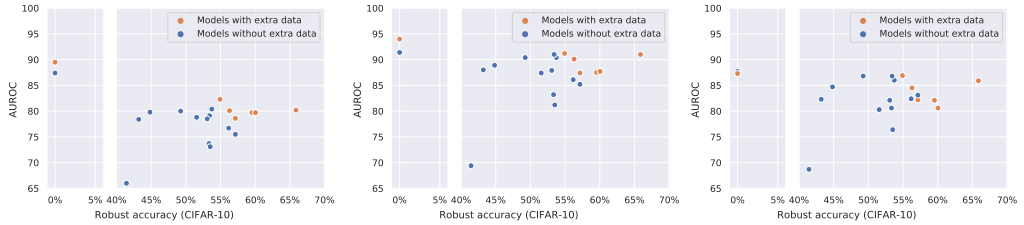


Figure 3: Visualization of the quality of OOD detection (higher AUROC is better) for the ℓ_∞-robust models on three OOD datasets: CIFAR-100 (left), SVHN (middle), Describable Textures (right).

4 Analysis

With unified access to the Model Zoo, one can easily compute various performance metrics. In Fig. 1 we plot some statistics for the ℓ_∞-robust classifiers: we observe that 1) for multiple *published* defenses the reported robust accuracy is highly overestimated, 2) using extra data can alleviate the robustness-accuracy trade-off as suggested in previous work (Raghunathan et al., 2020), 3) the most robust classifiers rely on adversarial training (Madry et al., 2018) or TRADES (Zhang et al., 2019b).

Performance across various distribution shifts. We test the performance of the ℓ_∞-robust classifiers from the Model Zoo on different distribution shifts, ranging from common image corruptions (CIFAR-10-C), dataset resampling bias (CIFAR-10.1, Recht et al. (2019)), and image source shift (CINIC-10, Darlow et al. (2018)). We plot the results in Fig. 2 which in particular shows that ℓ_p-robust models have lower standard accuracy on distribution shifts than a standard model except for CIFAR-10-C where there is a clear improvement. Moreover, ℓ_p adversarial robustness generalizes across different distribution shifts (see App. D).

Out-of-distribution (OOD) detection. Ideally, a classifier should exhibit uncertainty in its predictions on OOD inputs. Hendrycks & Gimpel (2017) extract this uncertainty information using some threshold on the predicted confidence, assuming that OOD inputs receive low confidence from the model. Song et al. (2020) demonstrated that ℓ_∞ adversarial training can degrade the ability to recognize OOD inputs, measured as area under the ROC curve (AUROC), which is confirmed by our evaluation in Fig. 3 on three OOD datasets, CIFAR-100 (Krizhevsky & Hinton, 2009), SVHN (Netzer et al., 2011), Describable Textures Dataset (Cimpoi et al., 2014). We notice that, unlike for ℓ_p-robustness, using extra data for training is not beneficial (see App. D for more results).

5 Outlook

We expect that a clearly defined, up-to-date leaderboard of the state-of-the-art robust models, easily accessible via a Model Zoo, will help to discover new insights and improve the current algorithms. We plan to expand our benchmark to other datasets such as CIFAR-100 and ImageNet.

REFERENCES

Motasem Alfarra, Juan C. Perez, Adel Bibi, Ali Thabet, Pablo Arbelaez, and Bernard Ghanem. Clustr: Clustering training for robustness. arXiv, 2020.

Maksym Andriushchenko, Francesco Croce, Nicolas Flammarion, and Matthias Hein. Square attack: a query-efficient black-box adversarial attack via random search. In ECCV, 2020.

Anish Athalye, Nicholas Carlini, and David Wagner. Obfuscated gradients give a false sense of security: Circumventing defenses to adversarial examples. In ICML, 2018.

Matan Atzmon, Niv Haim, Lior Yariv, Ofer Israelov, Haggai Maron, and Yaron Lipman. Controlling neural level sets. NeurIPS, 2019.

Maximilian Augustin, Alexander Meinke, and Matthias Hein. Adversarial robustness on in- and out-distribution improves explainability. ECCV, 2020.

Alexey Bochkovskiy, Chien-Yao Wang, and Hong-Yuan Mark Liao. Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934, 2020.

Wieland Brendel, Jonas Rauber, Alexey Kurakin, Nicolas Papernot, Behar Veliqi, Marcel Salathé, Sharada P Mohanty, and Matthias Bethge. Adversarial vision challenge. In NeurIPS Competition Track, 2018.

Tom B Brown, Dandelion Mané, Aurko Roy, Martín Abadi, and Justin Gilmer. Adversarial patch. In NeurIPS 2017 Workshop on Machine Learning and Computer Security, 2017.

Tom B Brown, Nicholas Carlini, Chiyuan Zhang, Catherine Olsson, Paul Christiano, and Ian Goodfellow. Unrestricted adversarial examples. arXiv preprint arXiv:1809.08352, 2018.

Jacob Buckman, Aurko Roy, Colin Raffel, and Ian Goodfellow. Thermometer encoding: One hot way to resist adversarial examples. In ICLR, 2018.

Nicholas Carlini. A critique of the deepsec platform for security analysis of deep learning models. arXiv preprint arXiv:1905.07112, 2019.

Nicholas Carlini, Anish Athalye, Nicolas Papernot, Wieland Brendel, Jonas Rauber, Dimitris Tsipras, Ian Goodfellow, Aleksander Madry, and Alexey Kurakin. On evaluating adversarial robustness. arXiv preprint arXiv:1902.06705, 2019.

Yair Carmon, Aditi Raghunathan, Ludwig Schmidt, Percy Liang, and John C. Duchi. Unlabeled data improves adversarial robustness. NeurIPS, 2019.

Alvin Chan, Yi Tay, Yew Soon Ong, and Jie Fu. Jacobian adversarially regularized networks for robustness. ICLR, 2020.

Jinghui Chen, Yu Cheng, Zhe Gan, Quanquan Gu, and Jingjing Liu. Efficient robust training via backward smoothing. arXiv, 2020a.

Tianlong Chen, Sijia Liu, Shiyu Chang, Yu Cheng, Lisa Amini, and Zhangyang Wang. Adversarial robustness: From self-supervised pre-training to fine-tuning. In CVPR, 2020b.

Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In CVPR, 2014.

Jeremy M Cohen, Elan Rosenfeld, and J Zico Kolter. Certified adversarial robustness via randomized smoothing. In ICML, 2019.

Francesco Croce and Matthias Hein. Sparse and imperceivable adversarial attacks. In ICCV, 2019.

Francesco Croce and Matthias Hein. Minimally distorted adversarial examples with a fast adaptive boundary attack. In ICML, 2020a.

Francesco Croce and Matthias Hein. Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks. In ICML, 2020b.

Francesco Croce, Maksym Andriushchenko, Naman D Singh, Nicolas Flammarion, and Matthias Hein. Sparse-rs: a versatile framework for query-efficient sparse black-box adversarial attacks. In ECCV Workshop on Adversarial Robustness in the Real World, 2020.

Jiequan Cui, Shu Liu, Liwei Wang, and Jiaya Jia. Learnable boundary guided adversarial training. arXiv, 2020.

Luke N Darlow, Elliot J Crowley, Antreas Antoniou, and Amos J Storkey. Cinic-10 is not imagenet or cifar-10. arXiv preprint arXiv:1810.03505, 2018.

Gavin Weiguang Ding, Luyu Wang, and Xiaomeng Jin. AdverTorch v0.1: An adversarial robustness toolbox based on pytorch. arXiv preprint arXiv:1902.07623, 2019.

Gavin Weiguang Ding, Yash Sharma, Kry Yik Chau Lui, and Ruitong Huang. Mma training: Direct input space margin maximization through adversarial training. In ICLR, 2020.

Yinpeng Dong, Qi-An Fu, Xiao Yang, Tianyu Pang, Hang Su, Zihao Xiao, and Jun Zhu. Benchmarking adversarial robustness on image classification. In CVPR, 2020.

Logan Engstrom, Andrew Ilyas, and Anish Athalye. Evaluating and understanding the robustness of adversarial logit pairing. NeurIPS 2018 Workshop on Security in Machine Learning, 2018.

Logan Engstrom, Andrew Ilyas, Hadi Salman, Shibani Santurkar, and Dimitris Tsipras. Robustness (python library), 2019a. URL `https://github.com/MadryLab/robustness`.

Logan Engstrom, Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Brandon Tran, and Aleksander Madry. Adversarial robustness as a prior for learned representations. arXiv preprint arXiv:1906.00945, 2019b.

Logan Engstrom, Brandon Tran, Dimitris Tsipras, Ludwig Schmidt, and Aleksander Madry. Exploring the landscape of spatial robustness. In ICML, 2019c.

Alhussein Fawzi and Pascal Frossard. Manitest: Are classifiers really invariant? In BMVC, 2015.

Justin Gilmer, Ryan P Adams, Ian Goodfellow, David Andersen, and George E Dahl. Motivating the rules of the game for adversarial example research. arXiv preprint arXiv:1807.06732, 2018.

Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and harnessing adversarial examples. In ICLR, 2015.

Dou Goodman, Hao Xin, Wang Yang, Wu Yuesheng, Xiong Junfeng, and Zhang Huan. Advbox: a toolbox to generate adversarial examples that fool neural networks. arXiv preprint arXiv:2001.05574, 2020.

Sven Gowal, Krishnamurthy (Dj) Dvijotham, Robert Stanforth, Rudy Bunel, Chongli Qin, Jonathan Uesato, Relja Arandjelovic, Timothy Mann, and Pushmeet Kohli. Scalable verified training for provably robust image classification. In ICCV, 2019a.

Sven Gowal, Jonathan Uesato, Chongli Qin, Po-Sen Huang, Timothy Mann, and Pushmeet Kohli. An alternative surrogate loss for pgd-based adversarial testing. arXiv preprint arXiv:1910.09338, 2019b.

Sven Gowal, Chongli Qin, Jonathan Uesato, Timothy Mann, and Pushmeet Kohli. Uncovering the limits of adversarial training against norm-bounded adversarial examples. arXiv, 2020.

Chuan Guo, Mayank Rana, Moustapha Cisse, and Laurens van der Maaten. Countering adversarial images using input transformations. In ICLR, 2018.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In ICLR, 2019.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In ICLR, 2017.

Dan Hendrycks, Kimin Lee, and Mantas Mazeika. Using pre-training can improve model robustness and uncertainty. In ICML, 2019.

Dan Hendrycks, Norman Mu, Ekin D. Cubuk, Barret Zoph, Justin Gilmer, and Balaji Lakshminarayanan. Augmix: A simple data processing method to improve robustness and uncertainty. ICLR, 2020.

J Edward Hu, Adith Swaminathan, Hadi Salman, and Greg Yang. Improved image wasserstein attacks and defenses. ICLR Workshop: Towards Trustworthy ML: Rethinking Security and Privacy for ML, 2020.

Lang Huang, Chao Zhang, and Hongyang Zhang. Self-adaptive training: beyond empirical risk minimization. NeurIPS, 2020.

Yunseok Jang, Tianchen Zhao, Seunghoon Hong, and Honglak Lee. Adversarial defense via learning to generate diverse attacks. ICCV, 2019.

Charles Jin and Martin Rinard. Manifold regularization for adversarial robustness. arXiv, 2020.

Daniel Kang, Yi Sun, Tom Brown, Dan Hendrycks, and Jacob Steinhardt. Transfer of adversarial robustness between perturbation types. arXiv preprint arXiv:1905.01034, 2019a.

Daniel Kang, Yi Sun, Dan Hendrycks, Tom Brown, and Jacob Steinhardt. Testing robustness against unforeseen adversaries. arXiv preprint arXiv:1908.08016, 2019b.

Guy Katz, Clark Barrett, David L Dill, Kyle Julian, and Mykel J Kochenderfer. Reluplex: an efficient smt solver for verifying deep neural networks. In ICCAV, 2017.

Simran Kaur, Jeremy Cohen, and Zachary C Lipton. Are perceptually-aligned gradients a general property of robust classifiers? In NeurIPS Workshop: Science Meets Engineering of Deep Learning, 2019.

Jungeum Kim and Xiao Wang. Sensible adversarial learning. OpenReview, 2019.

Klim Kireev, Maksym Andriushchenko, and Nicolas Flammarion. On the effectiveness of adversarial training against common corruptions. arXiv, 2021.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. Technical Report, 2009.

Alexey Kurakin, Ian Goodfellow, Samy Bengio, Yinpeng Dong, Fangzhou Liao, Ming Liang, Tianyu Pang, Jun Zhu, Xiaolin Hu, Cihang Xie, et al. Adversarial attacks and defences competition. In NeurIPS Competition Track, 2018.

Cassidy Laidlaw and Soheil Feizi. Functional adversarial attacks. In NeurIPS, 2019.

Cassidy Laidlaw, Sahil Singla, and Soheil Feizi. Perceptual adversarial robustness: Defense against unseen threat models. arXiv preprint arXiv:2006.12655, 2020.

Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana. Certified robustness to adversarial examples with differential privacy. In 2019 IEEE S&P, 2019.

Yingzhen Li, John Bradshaw, and Yash Sharma. Are generative classifiers more robust to adversarial attacks? In ICML, 2019.

Xiang Ling, Shouling Ji, Jiaxu Zou, Jiannan Wang, Chunming Wu, Bo Li, and Ting Wang. Deepsec: A uniform platform for security analysis of deep learning model. In IEEE S&P, 2019.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In ICLR, 2018.

Pratyush Maini, Eric Wong, and J Zico Kolter. Adversarial robustness against the union of multiple perturbation models. In ICML, 2020.

Chengzhi Mao, Ziyuan Zhong, Junfeng Yang, Carl Vondrick, and Baishakhi Ray. Metric learning for adversarial robustness. NeurIPS, 2019.

Marco Melis, Ambra Demontis, Maura Pintor, Angelo Sotgiu, and Battista Biggio. secml: A python library for secure and explainable machine learning. arXiv preprint arXiv:1912.10013, 2019.

Apostolos Modas, Seyed-Mohsen Moosavi-Dezfooli, and Pascal Frossard. Sparsefool: a few pixels make a big difference. In CVPR, 2019.

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Jonathan Uesato, and Pascal Frossard. Robustness via curvature regularization, and vice versa. CVPR, 2019.

Marius Mosbach, Maksym Andriushchenko, Thomas Trost, Matthias Hein, and Dietrich Klakow. Logit pairing methods can fool gradient-based attacks. In NeurIPS 2018 Workshop on Security in Machine Learning, 2018.

Aamir Mustafa, Salman Khan, Munawar Hayat, Roland Goecke, Jianbing Shen, and Ling Shao. Adversarial defense by restricting the hidden space of deep neural networks. ICCV, 2019.

Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. Technical Report, 2011.

Maria-Irina Nicolae, Mathieu Sinn, Minh Ngoc Tran, Beat Buesser, Ambrish Rawat, Martin Wistuba, Valentina Zantedeschi, Nathalie Baracaldo, Bryant Chen, Heiko Ludwig, Ian Molloy, and Ben Edwards. Adversarial robustness toolbox v1.2.0. arXiv preprint arXiv:1807.01069, 2018.

Tianyu Pang, Kun Xu, Yinpeng Dong, Chao Du, Ning Chen, and Jun Zhu. Rethinking softmax cross-entropy loss for adversarial robustness. ICLR, 2020a.

Tianyu Pang, Xiao Yang, Yinpeng Dong, Kun Xu, Hang Su, and Jun Zhu. Boosting adversarial training with hypersphere embedding. NeurIPS, 2020b.

Tianyu Pang, Xiao Yang, Yinpeng Dong, Hang Su, and Jun Zhu. Bag of tricks for adversarial training. ICLR, 2021.

Nicolas Papernot, Fartash Faghri, Nicholas Carlini, Ian Goodfellow, Reuben Feinman, Alexey Kurakin, Cihang Xie, Yash Sharma, Tom Brown, Aurko Roy, Alexander Matyasko, Vahid Behzadan, Karen Hambardzumyan, Zhishuai Zhang, Yi-Lin Juang, Zhi Li, Ryan Sheatsley, Abhibhav Garg, Jonathan Uesato, Willi Gierke, Yinpeng Dong, David Berthelot, Paul Hendricks, Jonas Rauber, and Rujun Long. Technical report on the cleverhans v2.1.0 adversarial examples library. arXiv preprint arXiv:1610.00768, 2018.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. Technical Report, 2017.

Chongli Qin, James Martens, Sven Gowal, Dilip Krishnan, Krishnamurthy Dvijotham, Alhussein Fawzi, Soham De, Robert Stanforth, and Pushmeet Kohli. Adversarial robustness through local linearization. NeurIPS, 2019.

Aditi Raghunathan, Sang Michael Xie, Fanny Yang, John Duchi, and Percy Liang. Understanding and mitigating the tradeoff between robustness and accuracy. In ICML, 2020.

Jonas Rauber, Wieland Brendel, and Matthias Bethge. Foolbox: A python toolbox to benchmark the robustness of machine learning models. In ICML Reliable Machine Learning in the Wild Workshop, 2017.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In ICML, 2019.

Leslie Rice, Eric Wong, and J Zico Kolter. Overfitting in adversarially robust deep learning. In ICML, 2020.

Jérôme Rony, Luiz G. Hafemann, Luiz S. Oliveira, Ismail Ben Ayed, Robert Sabourin, and Eric Granger. Decoupling direction and norm for efficient gradient-based l2 adversarial attacks and defenses. CVPR, 2019.

Parsa Saadatpanah, Ali Shafahi, and Tom Goldstein. Adversarial attacks on copyright detection systems. In ICML, 2020.

Hadi Salman, Andrew Ilyas, Logan Engstrom, Ashish Kapoor, and Aleksander Madry. Do adversarially robust imagenet models transfer better? NeurIPS, 2020.

Pouya Samangouei, Maya Kabkab, and Rama Chellappa. Defense-GAN: Protecting classifiers against adversarial attacks using generative models. In ICLR, 2018.

Vikash Sehwag, Shiqi Wang, Prateek Mittal, and Suman Jana. Hydra: Pruning adversarially robust neural networks. NeurIPS, 2020.

Ali Shafahi, Mahyar Najibi, Amin Ghiasi, Zheng Xu, John Dickerson, Christoph Studer, Larry S. Davis, Gavin Taylor, and Tom Goldstein. Adversarial training for free! NeurIPS, 2019.

Mayank Singh, Abhishek Sinha, Nupur Kumari, Harshitha Machiraju, Balaji Krishnamurthy, and Vineeth N Balasubramanian. Harnessing the vulnerability of latent layers in adversarially trained models. IJCAI, 2019.

Chawin Sitawarin, Supriyo Chakraborty, and David Wagner. Improving adversarial robustness through progressive hardening. arXiv, 2020.

Liwei Song, Vikash Sehwag, Arjun Nitin Bhagoji, and Prateek Mittal. A critical evaluation of open-world machine learning. arXiv preprint arXiv:2007.04391, 2020.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Dumitru Erhan Joan Bruna, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. In ICLR, 2013.

Rohan Taori, Achal Dave, Vaishaal Shankar, Nicholas Carlini, Benjamin Recht, and Ludwig Schmidt. Measuring robustness to natural distribution shifts in image classification. arXiv preprint arXiv:2007.00644, 2020.

Vincent Tjeng, Kai Xiao, and Russ Tedrake. Evaluating robustness of neural networks with mixed integer programming. In ICLR, 2019.

Florian Tramèr and Dan Boneh. Adversarial training and robustness for multiple perturbations. In NeurIPS, 2019.

Florian Tramèr, Pascal Dupré, Gili Rusak, Giancarlo Pellegrino, and Dan Boneh. Adversarial: Perceptual ad blocking meets adversarial machine learning. In ACM SIGSAC CCS, 2019.

Florian Tramèr, Nicholas Carlini, Wieland Brendel, and Aleksander Madry. On adaptive attacks to adversarial example defenses. In NeurIPS, 2020.

Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. In ICLR, 2019.

Jonathan Uesato, Jean-Baptiste Alayrac, Po-Sen Huang, Robert Stanforth, Alhussein Fawzi, and Pushmeet Kohli. Are labels required for improving adversarial robustness? NeurIPS, 2019.

Francisco Utrera, Evan Kravitz, N Benjamin Erichson, Rajiv Khanna, and Michael W Mahoney. Adversarially-trained deep nets transfer better. arXiv preprint arXiv:2007.05869, 2020.

Jianyu Wang and Haichao Zhang. Bilateral adversarial training: Towards fast training of more robust models against adversarial attacks. ICCV, 2019.

Yisen Wang, Difan Zou, Jinfeng Yi, James Bailey, Xingjun Ma, and Quanquan Gu. Improving adversarial robustness requires revisiting misclassified examples. ICLR, 2020.

Tsui-Wei Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Duane Boning, Inderjit S. Dhillon, and Luca Daniel. Towards fast computation of certified robustness for relu networks. In ICML, 2018.

Eric Wong and J Zico Kolter. Learning perturbation sets for robust machine learning. arXiv preprint arXiv:2007.08450, 2020.

Eric Wong and Zico Kolter. Provable defenses against adversarial examples via the convex outer adversarial polytope. ICML, 2018.

Eric Wong, Frank R Schmidt, and J Zico Kolter. Wasserstein adversarial examples via projected sinkhorn iterations. In ICML, 2019.

Eric Wong, Leslie Rice, and J. Zico Kolter. Fast is better than free: Revisiting adversarial training. ICLR, 2020.

Boxi Wu, Jinghui Chen, Deng Cai, Xiaofei He, and Quanquan Gu. Do wider neural networks really help adversarial robustness? arXiv, 2020a.

Dongxian Wu, Shu tao Xia, and Yisen Wang. Adversarial weight perturbation helps robust generalization. NeurIPS, 2020b.

Chang Xiao, Peilin Zhong, and Changxi Zheng. Enhancing adversarial defense by k-winners-take-all. ICLR, 2020.

Cihang Xie, Mingxing Tan, Boqing Gong, Jiang Wang, Alan L Yuille, and Quoc V Le. Adversarial examples improve image recognition. In CVPR, 2020.

Dinghuai Zhang, Tianyuan Zhang, Yiping Lu, Zhanxing Zhu, and Bin Dong. You only propagate once: Accelerating adversarial training via maximal principle. NeurIPS, 2019a.

Haichao Zhang and Jianyu Wang. Defense against adversarial attacks using feature scattering-based adversarial training. NeurIPS, 2019.

Haichao Zhang and Wei Xu. Adversarial interpolation training: A simple approach for improving model robustness. OpenReview, 2019.

Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Eric P. Xing, Laurent El Ghaoui, and Michael I. Jordan. Theoretically principled trade-off between robustness and accuracy. In ICML, 2019b.

Jingfeng Zhang, Xilie Xu, Bo Han, Gang Niu, Lizhen Cui, Masashi Sugiyama, and Mohan Kankanhalli. Attacks which do not kill training make adversarial learning stronger. ICML, 2020.

Jiachen Zhong, Xuanqing Liu, and Cho-Jui Hsieh. Improving the speed and quality of gan by adversarial training. arXiv preprint arXiv:2008.03364, 2020.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. Freelb: Enhanced adversarial training for natural language understanding. In ICLR, 2019.

A BACKGROUND ON ℓ_p-ROBUSTNESS

As mentioned in Sec. 2, computing the exact robust accuracy is in general intractable and NP-hard even for single-layer neural networks (Katz et al., 2017; Weng et al., 2018). Upper bounds are given by adversarial attacks which are mostly based on optimizing some differentiable loss (e.g., cross entropy) using local search algorithms like projected gradient descent (PGD) in order to find a successful adversarial perturbation. The tightness of the upper bound depends on the effectiveness of the attack: unsuitable techniques or suboptimal parameters (in particular, the step size and the number of iterations) can make the models appear more robust than they actually are (Engstrom et al., 2018; Mosbach et al., 2018), especially in the presence of phenomena like gradient obfuscation (Athalye et al., 2018). At the same time, certified methods such as Wong & Kolter (2018) and Gowal et al. (2019a) instead provide *lower bounds* on robust accuracy but often underestimate robustness significantly, in particular if the certification was not part of the training process. Thus, we do not consider lower bounds in our benchmark, and focus only on upper bounds which are typically much tighter (Tjeng et al., 2019).

We note that robustness towards small ℓ_p-bounded perturbations is a necessary but not sufficient notion of robustness which has been criticized in the literature (Gilmer et al., 2018). It is an active area of research to develop threat models which are more aligned with the human perception such as spatial perturbations (Fawzi & Frossard, 2015; Engstrom et al., 2019c), Wasserstein-bounded perturbations (Wong et al., 2019; Hu et al., 2020), perturbations of the image colors (Laidlaw & Feizi, 2019) or ℓ_p-perturbations in the latent space of a neural network (Laidlaw et al., 2020; Wong & Kolter, 2020). However, despite the simplicity of the ℓ_p-perturbation model, it has numerous interesting applications that go beyond security considerations (Tramèr et al., 2019; Saadatpanah et al., 2020) and span transfer learning (Salman et al., 2020; Utrera et al., 2020), interpretability (Tsipras et al., 2019; Kaur et al., 2019; Engstrom et al., 2019b), generalization (Xie et al., 2020; Zhu et al., 2019; Bochkovskiy et al., 2020), robustness to unseen perturbations (Kang et al., 2019a; Xie et al., 2020; Laidlaw et al., 2020), stabilization of GAN training (Zhong et al., 2020). Thus, improvements in ℓ_p-robustness have the potential to improve many of these downstream applications.

Therefore, since ℓ_p-bounded perturbations are widely studied, well-defined, and have many reasons of interest, they constitute the threat models on which we decided to focus on initially, in addition to common image corruptions which cover a complementary aspect of robustness.

B RELATED WORK

Alongside with the libraries for adversarial attacks (see Sec. 2), many benchmarks aiming at tracking progress of adversarial defenses have appeared, with different forms. The two challenges (Kurakin et al., 2018; Brendel et al., 2018) hosted at NeurIPS 2017 and 2018 aimed at finding the most robust models for specific attacks, but they had a predefined deadline, so they could capture the best defenses only at the time of the competition. Ling et al. (2019) proposed DEEPSEC, a benchmark that tests many combinations of attacks and defenses, but suffers from a few shortcomings as suggested by Carlini (2019), in particular: (1) reporting average-case performance over multiple attacks instead of worst-case performance, (2) evaluating robustness in threat models different from the one used for training, (3) using excessively large perturbations. Recently, Dong et al. (2020) have provided an evaluation of a few defenses (in particular, 3 for ℓ_∞- and 2 for ℓ_2-norm on CIFAR-10) against multiple commonly used attacks. However, they did not include some of the best performing defenses (Hendrycks et al., 2019; Carmon et al., 2019) and attacks (Gowal et al., 2019b; Croce & Hein, 2020a), and in a few cases, their evaluation suggests robustness higher than what was reported in the original papers. Moreover, they do not impose any restrictions on the models they accept to the benchmark. RobustML (`https://www.robust-ml.org/`) aims at collecting robustness claims for defenses together with external evaluations. Their format does not assume running any baseline attack, so it relies entirely on evaluations submitted by the community. However, external evaluations are not submitted often enough, and thus even though RobustML has been a valuable contribution to the community, now it does not provide a comprehensive overview of the recent state of the art in adversarial robustness.

Furthermore, it has become common practice to test new attacks wrt ℓ_∞ on the publicly available models from Madry et al. (2018) and Zhang et al. (2019b), since those represent widely accepted de-

fenses which have stood many thorough evaluations. However, having only two models per dataset (MNIST and CIFAR-10) does not constitute a sufficiently large testbed, and, because of the repetitive evaluations, some attacks may already overfit to those defenses. For example, currently the difference in robust accuracy between the first and second-best attacks in the CIFAR-10 leaderboard of Madry et al. (2018) is only 0.03%, and between the second and third is 0.04%.

Finally, for common corruptions, there exists a benchmark at `https://github.com/hendrycks/robustness` which, however, only considers ImageNet and limits the classifiers to use only the ResNet-50 architecture. Thus, we think it is useful to track the progress also in this task for other datasets, and we start with CIFAR-10.

C Details of RobustBench

C.1 Leaderboard

Restrictions. As mentioned in Sec. 3 we consider only classifiers $f : \mathbb{R}^d \to \mathbb{R}^C$ that

- have in general *non-zero gradients* with respect to the inputs. Models with zero gradients, e.g., that rely on quantization of inputs (Buckman et al., 2018; Guo et al., 2018), make gradient-based methods ineffective thus requiring zeroth-order attacks, which do not perform as well as gradient-based attacks. Alternatively, specific adaptive evaluations, e.g. with Backward Pass Differentiable Approximation (Athalye et al., 2018), can be used which, however, can hardly be standardized. Moreover, we are not aware of existing defenses solely based on having zero gradients for large parts of the input space which would achieve competitive robustness.

- have a *fully deterministic forward pass*. To evaluate defenses with stochastic components, it is a common practice to combine standard gradient-based attacks with Expectation over Transformations (Athalye et al., 2018). While often effective, it might be not sufficient, as shown by Tramèr et al. (2020). Moreover, the classification decision of randomized models may vary over different runs for the same input, hence even the definition of robust accuracy differs from that of deterministic networks. We also note that randomization *can* be useful for improving robustness and deriving robustness certificates (Lecuyer et al., 2019; Cohen et al., 2019), but it also introduces variance in the gradient estimators (both white- and black-box) which can make attacks much less effective.

- do not have an *optimization loop* in the forward pass. This makes backpropagation through the classifier very difficult or extremely expensive. Usually, such defenses (Samangouei et al., 2018; Li et al., 2019) need to be evaluated adaptively with attacks considering jointly the loss of the inner loop and the standard classification task.

As described, the evaluation of classifiers not fulfilling these requirements can hardly be standardized. Moreover, we are not aware of models falling in one of these categories and not relying on standard methods such as adversarial training which are shown to achieve competitive robustness.

Evaluation of defenses. As described in Sec. 3, we perform the standardized evaluation of the ℓ_p-robustness of the reported defenses using AutoAttack (Croce & Hein, 2020b), an ensemble of four attacks (see appendix), a variation of PGD attack with automatically adjusted step sizes, with (1) the cross entropy loss and (2) the difference of logits ratio loss, which is a rescaling-invariant margin-based loss function, (3) the targeted version FAB attack (Croce & Hein, 2020a), which minimizes the ℓ_p-norm of the perturbations, and (4) the black-box Square Attack (Andriushchenko et al., 2020). Moreover, we keep open the option of submitting new evaluations of adversarial robustness by adaptive attacks. The goal is to achieve the most accurate approximation of the true robustness that can complement the standardized evaluation in some exceptional cases. Thus, we will report in the leaderboard both the results of the standardized attack and the best adaptive evaluation if it outperforms the standard one.

Figure 4: Visualization of the robustness and accuracy of 9 CIFAR-10 models from the `RobustBench` ℓ_2-leaderboard. Robustness is evaluated using ℓ_2-perturbations with $\varepsilon_2 = 0.5$.

C.2 Model Zoo

As mentioned in Sec. 3.2, we gather the checkpoints of many networks from the leaderboard. Below we illustrate how a model can be automatically downloaded and loaded via its identifier, the dataset on which it has been trained, and threat model to which it is robust within two lines of code:

```
from robustbench.utils import load_model
model = load_model(model_name='Carmon2019Unlabeled',
                   dataset='cifar10', threat_model='Linf')
```

For the moment, the only available dataset is `'cifar10'` and the available threat models are `'Linf'` (for ℓ_∞ robustness), `'L2'` (for ℓ_2 robustness), and `'corruptions'` (for common corruption robustness).

The library also enables to benchmark the robustness of some models even if they are not included in the Model Zoo, on a given dataset and threat model. This can be achieved with the `benchmark` function, in the following way:

```
from robustbench.eval import benchmark
model = SomePyTorchModel()
clean_acc, robust_acc = benchmark(model,
                                  dataset='cifar10',
                                  threat_model='Linf')
```

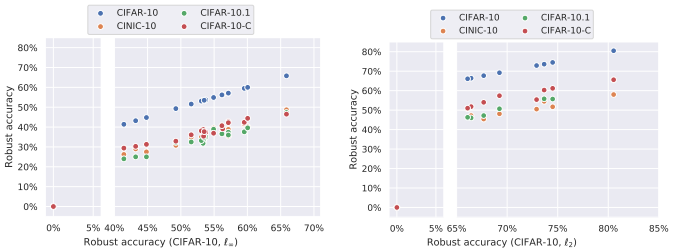
The function returns the accuracy on the clean dataset, and the robust accuracy in the specified threat model. The `benchmark` function supports the same datasets and threat models as described above.

D Additional analysis

We here extend the analysis of Sec. 4. As done above for the ℓ_∞-robust models, we show some statistics about the defenses for ℓ_2-robustness in Fig. 4. The most robust model exploits a very large network and additional data for training.

Performance across various distribution shifts. In Fig. 5 we show the robust accuracy of ℓ_∞- and ℓ_2-robust models from the Model Zoo on multiple distribution shifts of CIFAR-10. We report robust accuracy using AutoAttack and the same threat model as used for training on the standard CIFAR-10. We observe a trend that is very close to linear, and thus we conclude that ℓ_p robustness successfully transfers even under different distribution shifts. Finally, we note that concurrently with our work, Taori et al. (2020) also study the robustness to different distribution shifts of many models trained on ImageNet, including some ℓ_p-robust models. Our conclusions qualitatively agree with theirs, and we hope that our collected set of models will help to provide a more complete picture.

Out-of-distribution detection. Fig. 6 complements Fig. 3 and shows the ability of ℓ_2-robust models trained on CIFAR-10 to distinguish inputs from other datasets (CIFAR-100, SVHN, Describable Textures). We find that ℓ_2 robust models have in general comparable OOD detection performance to standardly trained models, while the model of Augustin et al. (2020) achieves even better perfor-



(a) `Linf` (b) `L2`

Figure 5: Robust accuracy of the robust classifiers, trained against ℓ_∞ and ℓ_2 threat model, respectively, from our Model Zoo on various distribution shifts. The data points with 0% robust accuracy correspond to a standardly trained model.

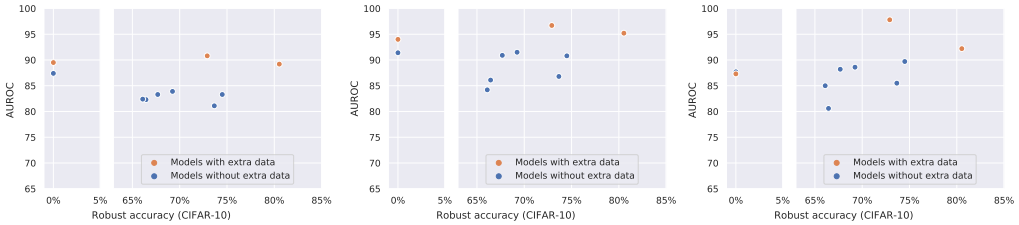


Figure 6: Visualization of the quality of OOD detection (higher AUROC is better) for the ℓ_2-robust models on three different OOD datasets: CIFAR-100 (left), SVHN (middle), Describable Textures (right).

mance since their approach explicitly optimizes both robust accuracy and worst-case OOD detection performance.

E LEADERBOARDS

We here report the details of all the models included in the various leaderboards, for the ℓ_∞-, ℓ_2-threat models and common corruptions. In particular, we show for each model the clean accuracy, robust accuracy (either on adversarial attacks or corrupted images), whether additional data is used for training, the architecture used, and the venue at which it appeared.

Table 1: Leaderboard for the ℓ_∞-threat model.

	model	clean	AA	extra	architecture	venue
1	Gowal et al. (2020)	91.10	65.87	Y	WideResNet-70-16	arXiv, Oct 2020
2	Gowal et al. (2020)	89.48	62.76	Y	WideResNet-28-10	arXiv, Oct 2020
3	Wu et al. (2020a)	87.67	60.65	Y	WideResNet-34-15	arXiv, Oct 2020
4	Wu et al. (2020b)	88.25	60.04	Y	WideResNet-28-10	NeurIPS 2020
5	Carmon et al. (2019)	89.69	59.53	Y	WideResNet-28-10	NeurIPS 2019
6	Gowal et al. (2020)	85.29	57.14	N	WideResNet-70-16	arXiv, Oct 2020
7	Sehwag et al. (2020)	88.98	57.14	Y	WideResNet-28-10	NeurIPS 2020
8	Gowal et al. (2020)	85.64	56.82	N	WideResNet-34-20	arXiv, Oct 2020
9	Wang et al. (2020)	87.50	56.29	Y	WideResNet-28-10	ICLR 2020
10	Wu et al. (2020b)	85.36	56.17	N	WideResNet-34-10	NeurIPS 2020
11	Uesato et al. (2019)	86.46	56.03	Y	WideResNet-28-10	NeurIPS 2019
12	Hendrycks et al. (2019)	87.11	54.92	Y	WideResNet-28-10	ICML 2019
13	Pang et al. (2021)	86.43	54.39	N	WideResNet-34-20	ICLR 2021
14	Pang et al. (2020b)	85.14	53.74	N	WideResNet-34-20	NeurIPS 2020
15	Cui et al. (2020)	88.70	53.57	N	WideResNet-34-20	arXiv, Nov 2020
16	Zhang et al. (2020)	84.52	53.51	N	WideResNet-34-10	ICML 2020
17	Rice et al. (2020)	85.34	53.42	N	WideResNet-34-20	ICML 2020
18	Huang et al. (2020)	83.48	53.34	N	WideResNet-34-10	NeurIPS 2020
19	Zhang et al. (2019b)	84.92	53.08	N	WideResNet-34-10	ICML 2019
20	Cui et al. (2020)	88.22	52.86	N	WideResNet-34-10	arXiv, Nov 2020
21	Qin et al. (2019)	86.28	52.84	N	WideResNet-40-8	NeurIPS 2019
22	Chen et al. (2020b)	86.04	51.56	N	ResNet-50	CVPR 2020
23	Chen et al. (2020a)	85.32	51.12	N	WideResNet-34-10	arXiv, Oct 2020
24	Sitawarin et al. (2020)	86.84	50.72	N	WideResNet-34-10	arXiv, Mar 2020
25	Engstrom et al. (2019a)	87.03	49.25	N	ResNet-50	GitHub, Oct 2019
26	Singh et al. (2019)	87.80	49.12	N	WideResNet-34-10	IJCAI 2019
27	Mao et al. (2019)	86.21	47.41	N	WideResNet-34-10	NeurIPS 2019
28	Zhang et al. (2019a)	87.20	44.83	N	WideResNet-34-10	NeurIPS 2019
29	Madry et al. (2018)	87.14	44.04	N	WideResNet-34-10	ICLR 2018
30	Pang et al. (2020a)	80.89	43.48	N	ResNet-32	ICLR 2020
31	Wong et al. (2020)	83.34	43.21	N	ResNet-18	ICLR 2020
32	Shafahi et al. (2019)	86.11	41.47	N	WideResNet-34-10	NeurIPS 2019
33	Ding et al. (2020)	84.36	41.44	N	WideResNet-28-4	ICLR 2020
34	Atzmon et al. (2019)	81.30	40.22	N	ResNet-18	NeurIPS 2019
35	Moosavi-Dezfooli et al. (2019)	83.11	38.50	N	ResNet-18	CVPR 2019
36	Zhang & Wang (2019)	89.98	36.64	N	WideResNet-28-10	NeurIPS 2019
37	Zhang & Xu (2019)	90.25	36.45	N	WideResNet-28-10	OpenReview, Sep 2019
38	Jang et al. (2019)	78.91	34.95	N	ResNet-20	ICCV 2019
39	Kim & Wang (2019)	91.51	34.22	N	WideResNet-34-10	OpenReview, Sep 2019
40	Wang & Zhang (2019)	92.80	29.35	N	WideResNet-28-10	ICCV 2019
41	Xiao et al. (2020)	79.28	18.50	N	DenseNet-121	ICLR 2020
42	Jin & Rinard (2020)	90.84	1.35	N	ResNet-18	arXiv, Mar 2020
43	Mustafa et al. (2019)	89.16	0.28	N	ResNet-110	ICCV 2019
44	Chan et al. (2020)	93.79	0.26	N	WideResNet-34-10	ICLR 2020
45	Alfarra et al. (2020)	91.03	0.00	N	WideResNet-28-10	arXiv, Jun 2020
46	Standard	94.78	0.0	N	WideResNet-28-10	N/A

Table 2: Leaderboard for the ℓ_2-threat model.

	model	clean	AA	extra	architecture	venue
1	Gowal et al. (2020)	94.74	80.53	Y	WideResNet-70-16	arXiv, Oct 2020
2	Gowal et al. (2020)	90.90	74.50	N	WideResNet-70-16	arXiv, Oct 2020
3	Wu et al. (2020b)	88.51	73.66	N	WideResNet-34-10	NeurIPS 2020
4	Augustin et al. (2020)	91.08	72.91	Y	ResNet-50	ECCV 2020
5	Engstrom et al. (2019a)	90.83	69.24	N	ResNet-50	GitHub, Sep 2019
6	Rice et al. (2020)	88.67	67.68	N	ResNet-18	ICML 2020
7	Rony et al. (2019)	89.05	66.44	N	WideResNet-28-10	CVPR 2019
8	Ding et al. (2020)	88.02	66.09	N	WideResNet-28-4	ICLR 2020
9	Standard	94.78	0.0	N	WideResNet-28-10	N/A

Table 3: Leaderboard for common corruptions.

	model	clean	corr.	extra	architecture	venue
1	Hendrycks et al. (2020)	95.83	89.09	N	ResNeXt-29-32x4d	ICLR 2020
2	Hendrycks et al. (2020)	95.08	88.82	N	WideResNet-40-2	ICLR 2020
3	Kireev et al. (2021)	94.77	88.53	N	ResNet-18	arXiv, Mar 2021
4	Gowal et al. (2020)	94.74	87.68	Y	WideResNet-70-16	arXiv, Oct 2020
5	Kireev et al. (2021)	93.24	85.04	N	ResNet-18	arXiv, Mar 2021
6	Gowal et al. (2020)	90.90	84.90	N	WideResNet-70-16	arXiv, Oct 2020
7	Kireev et al. (2021)	93.10	84.10	N	ResNet-18	arXiv, Mar 2021
8	Gowal et al. (2020)	91.10	81.84	Y	WideResNet-70-16	arXiv, Oct 2020
9	Kireev et al. (2021)	92.46	80.46	N	ResNet-18	arXiv, Mar 2021
10	Gowal et al. (2020)	85.29	76.37	N	WideResNet-70-16	arXiv, Oct 2020
11	Standard	94.78	73.46	N	WideResNet-28-10	N/A