Robustness May Be at Odds with Fairness: An Empirical Study on Class-wise Accuracy

Philipp Benz[*] PBENZ@KAIST.AC.KR
Chaoning Zhang[*] CHAONINGZHANG1990@GMAIL.COM
Adil Karjauv MIKOLEZ@GMAIL.COM
In So Kweon ISKWEON77@KAIST.AC.KR
Korea Advanced Institute of Science and Technology (KAIST)

Abstract

Convolutional neural networks (CNNs) have made significant advancement, however, they are widely known to be vulnerable to adversarial attacks. Adversarial training is the most widely used technique for improving adversarial robustness to strong white-box attacks. Prior works have been evaluating and improving the model average robustness without class-wise evaluation. The average evaluation alone might provide a false sense of robustness. For example, the attacker can focus on attacking the vulnerable class, which can be dangerous, especially, when the vulnerable class is a critical one, such as "human" in autonomous driving. We propose an empirical study on the class-wise accuracy and robustness of adversarially trained models. We find that there exists inter-class discrepancy for accuracy and robustness even when the training dataset has an equal number of samples for each class. For example, in CIFAR10, "cat" is much more vulnerable than other classes. Moreover, this inter-class discrepancy also exists for normally trained models, while adversarial training tends to further increase the discrepancy. Our work aims to investigate the following questions: (a) is the phenomenon of inter-class discrepancy universal regardless of datasets, model architectures and optimization hyper-parameters? (b) If so, what can be possible explanations for the inter-class discrepancy? (c) Can the techniques proposed in the long tail classification be readily extended to adversarial training for addressing the inter-class discrepancy?

Keywords: Fair Machine Learning, Class-wise Robustness, Adversarial training

1. Introduction

Convolutional neural networks (CNNs) (LeCun et al., 2015) have achieved enormous success in a wide range of vision applications (Zhang et al., 2019a; Kim et al., 2019, 2020; Pan et al., 2020; Zhang et al., 2020d, 2021a; Kim et al., 2021; Zhang et al., 2020c). However, they are still widely known to be vulnerable to adversarial examples (Szegedy et al., 2013; Goodfellow et al., 2015). Numerous endeavors have been attempted to improve adversarial robustness of deep classifiers, and adversarial training, to our best knowledge, is the only one that has not been broken by strong white-box attacks (Goodfellow et al., 2015; Madry et al., 2018; Carlini and Wagner, 2017). Prior works mainly report the model accuracy and robustness averaging on samples from all classes without class-wise evaluation. This average performance alone might be misleading for giving a wrong sense of robustness.

[*] Equal contribution

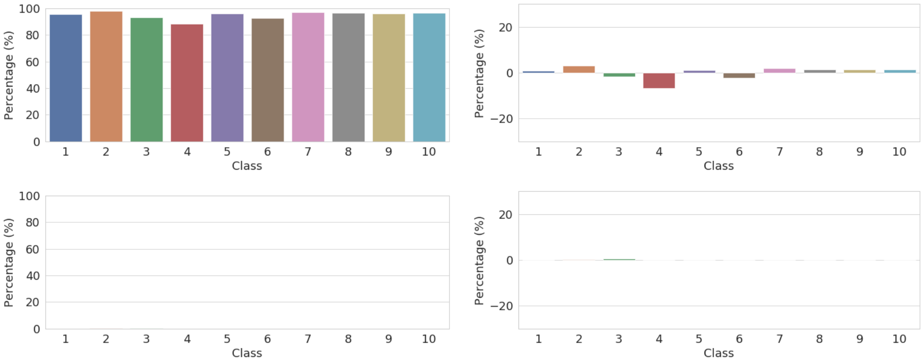

Figure 1: Inter-class discrepancy for standard model. First row: accuracy w/o (left) and w/ (right) mean subtracted. Second row: robustness w/o (left) and w/ (right) mean subtracted.

For example, in autonomous driving, a well-performing model with high accuracy and/or robustness averaging on all classes is particularly dangerous if a certain important class, such as *human*, is vulnerable.

Recognizing its practical relevance, we perform an empirical study to evaluate the class-wise accuracy and robustness of adversarially trained models. Preliminary investigation is conducted for CIFAR10 on ResNet18 (He et al., 2016) with both standard training (see Figure 1) and adversarial training (see Figure 2). There are several intriguing observations. First, there is a non-trivial inter-class discrepancy, even though the long-tail issue does not exist, *i.e.* each class is balanced with the same number of training samples. Second, a similar trend can be observed for the adversarially trained model, more notably, the inter-class discrepancy is more significant than that of a standard model. Third, the imbalance is the most significant for the adversarially trained model. Overall, it suggests that there exists inter-class discrepancy under balanced training dataset and adversarial training makes the inter-class discrepancy more serious. Beyond the already widely recognized importance on the model robustness, we argue that fairness in terms of decreasing the inter-class discrepancy also deserves the attention from the machine learning community.

Our empirical analysis will address the following questions regarding class-wise accuracy and robustness:

- Is the phenomenon of inter-class discrepancy universal in other setups?
- What are possible explanations for this inter-class discrepancy in accuracy and robustness?
- Can the techniques proposed in the long-tail setup be readily extended to adversarial training for addressing the inter-class discrepancy?

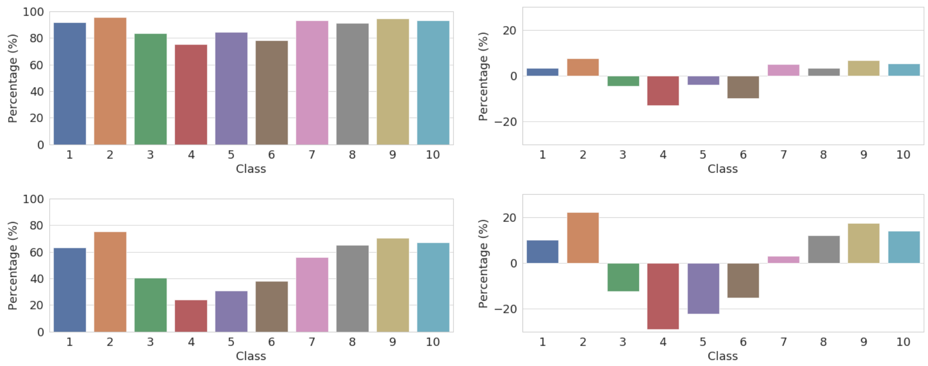

Figure 2: Inter-class discrepancy for adversarially trained model. First row: accuracy w/o (left) and w/ (right) mean subtracted. Second row: robustness w/o (left) and w/ (right) mean subtracted.

2. Related work

2.1. Adversarial examples

CNNs are widely known to be vulnerable to adversarial examples (Szegedy et al., 2013; Goodfellow et al., 2015; Ilyas et al., 2019; Benz et al., 2020c, 2021a, 2020a), which has inspired numerous investigations on both image-dependent attack (Szegedy et al., 2013; Goodfellow et al., 2015; Madry et al., 2018) and universal attack (Moosavi-Dezfooli et al., 2017; Zhang et al., 2020a,b; Benz et al., 2020b; Zhang et al., 2021b,c; Benz et al., 2021b) and defense (Qiu et al., 2019; Yuan et al., 2019). Most of the defense techniques have been broken, and currently, adversarial training (Goodfellow et al., 2015; Madry et al., 2018) is the most widely adopted one, empirically proven effective. In the past few years, numerous adversarial training methods (Zhang et al., 2019b; Wang and Zhang, 2019; Shafahi et al., 2019; Zhang et al., 2019c; Wong et al., 2020) have been proposed for improving either effectiveness or efficiency. Despite different motivations and implementations, they all fall into a min-max optimization problem (Madry et al., 2018), *i.e.* adversarial attack solving the loss maximization problem to generate adversarial examples and network training minimizing the loss to update the network weights. Adversarial training leads to some interesting findings. For example, adversarial training leads to higher robustness while at the cost of accuracy, inspiring to bridge the gap for the trade-off between robustness and accuracy (Zhang et al., 2019c). Adversarial training leads to a model with more robust features (Ilyas et al., 2019), consequently reducing information while improving transferability (Terzi et al., 2020). Adversarially trained models have also been found to be more fit for down-stream transfer learning (Utrera et al., 2020). Complementary to their findings, we empirically find that adversarial training increases inter-class discrepancy for accuracy and robustness.

2.2. Inter-class discrepancy in long tail recognition

The inter-class discrepancy problem in long-tail recognition is a fundamental issue in machine learning. In the real world setting, the long-tail problem exists when data is inherently imbalanced (Kendall et al., 1946), which undermines the performance of algorithms that do not take this problem into account (Buda et al., 2018; Japkowicz and Stephen, 2002). Due to its practical relevance, there is a body of work devoted to tackling this problem (He and Garcia, 2009). The core issue in the long tail recognition lies in low accuracy for the rare classes, and various techniques have been developed to improve the accuracy of those rare classes. Our results suggest that there exists an inter-class discrepancy for accuracy and robustness for a balanced dataset, especially, adversarial training leads to a more significant inter-class discrepancy. Conceptually, the vulnerable class in the adversarial training is similar to the rare class in the long tail recognition, since the accuracy of them is low and needs to be improved. Straightforwardly, the techniques proposed to solve the long tail recognition might also help mitigate the inter-class discrepancy for the accuracy and/or robustness of adversarially trained models. There are two common techniques in long-tail recognition for handling a class-imbalanced dataset. The first one is to modify the dataset to reduce the imbalance. One can either collect more data samples for the deficient classes (Chawla et al., 2002; He et al., 2008) or remove samples from the abundant classes to increase balance (Drummond et al., 2003). Alternatively, the sampling strategy can be designed to increase the sampling frequency for the rare classes during training. The second technique is called cost-sensitive learning, which modifies misclassification costs to account for the imbalance in the number of samples (Tang et al., 2008; Elkan, 2001; Huang et al., 2016). A recent method (Cui et al., 2019) introduces the weighting factors for each class to re-balance the loss function. These weights are inversely proportional to the effective number of samples for every class. (Cao et al., 2019) proposes a new re-balancing optimization procedure with a new loss function to encourage larger classification margins for deficient classes.

3. Methodology and experimental protocol

3.1. Is the phenomenon of inter-class discrepancy universal in other setups?

To test how universal the inter-class discrepancy for accuracy and robustness is, we mainly take into account three factors, *i.e.* datasets, model architectures and optimization methods. For dataset, we plan to test it on various benchmark datasets, including MNIST (LeCun et al., 1998), SVHN (Netzer et al., 2011), CIFAR10 (Krizhevsky et al., 2009), CIFAR100 (Krizhevsky et al., 2009), TinyImageNet, ImageNet (Deng et al., 2009). For models, we plan to evaluate on the most seminal models ranging from networks stacking a few convolutional layers to very deep networks, specifically including LeNet (LeCun et al., 1998), VGG family (VGG16 and VGG19) (Simonyan and Zisserman, 2015), ResNet family (ResNet18 and ResNet50) (He et al., 2016), DenseNet family (DenseNet121 and DenseNet169) (Huang et al., 2017), Inception family (GoogleNet, Inception-v3) (Szegedy et al., 2015). For optimization factors, we mainly consider optimizer, and learning rate schedule and weight decay.

To check whether the same phenomenon can be observed on the above datasets, we will adopt ResNet18 to train a standard and adversarially trained model for each dataset and evaluate the class-wise accuracy and robustness. For the ImageNet dataset, to avoid expensive computation, we will use the online available pre-trained ResNet50, for both standard and adversarially trained ones. For various models, we will test them only on CIFAR10 dataset for avoiding redundancy. For the optimization factors, we will test them with CIFAR10 on ResNet18, but with different optimizers, such as SGD and ADAM, different learning rate schedule (stepwise decrease and cyclic), and different weight decay factors.

Additionally, it would be interesting to see whether the vulnerable class changes during the training. Thus, we will also report the trend of class-wise accuracy and robustness during the whole training stage.

3.2. What might be possible explanations for the inter-class discrepancy?

At this stage, we assume that the results from the above investigation would support the following conclusion: The phenomenon of inter-class discrepancy should exist universally for a wide range of datasets on various model architectures with different optimization hyper-parameters. In the balanced dataset setup, each class has the same number of training samples, which increases the chance that the model architectures and/or optimization strategies might influence this phenomenon. We aim to analyze the following results:

- Is the same trend of inter-class discrepancy observed for different networks trained on the same dataset, *i.e.* a robust/vulnerable class on *model A* is also robust/vulnerable on *model B*? If so, we can conclude that the inter-class discrepancy has little to do with model architectures. Otherwise, model architecture can be one factor that causes the phenomenon of the inter-class discrepancy.
- Similarly, we can check the trend regarding different optimization factors. If the same trend is observed for models trained with different optimization, such as standard training vs. adversarial training, or SGD vs. ADAM, we can conclude that this inter-class discrepancy has little to do with optimization factors. Otherwise, optimization factors can be one factor influencing the inter-class discrepancy.

Another important factor is the semantic features in a dataset. We can conduct experiments on CIFAR100, which has super-classes, under which multiple sub-classes share similar semantic features. We can visualize a matrix of ground-truth classes and predicted classes. For example, for samples from ground-truth *class x*, we can count the number of predicted classes and the majority of the samples are likely to be predicted as *x*. If *class x* is semantically similar to *class y*, we expect that there would be a non-trivial amount of samples misclassified to be *class y*. Vice versa, there would also be a nontrivial amount of samples from ground-truth *class y* misclassified to be *class x*. Cross-class feature similarity, represented by the cosine similarity between the output logit vector of two different ground-truth labels, might be a good metric to measure the similarity between classes. Specifically, we measure the average logit vector for samples from each ground-truth label and then perform cross-class feature cosine similarity. If a label has high cross-class feature similarity with other labels, it means the model perceives it to be close to other labels, which might lead to the samples from this label being vulnerable to misclassification.

Feature perspective. Recently, adversarial robustness has been attributed to non-robust features in the dataset (Ilyas et al., 2019). It has been shown in (Ilyas et al., 2019) that both robust features and non-robust features are useful for classification. It would be interesting to distinguish whether robust or/and non-robust features lead to the inter-class discrepancy. Following (Ilyas et al., 2019), we will construct a dataset that has non-robust features, and train a new model on them and perform class-wise accuracy evaluation. As a control study, we will also construct a robust dataset and repeat the above procedure.

3.3. Can the techniques from long tail be extended to adversarial training?

Since our setup has a balanced number of samples for each class, we mainly evaluate the cost-sensitive learning strategy, by giving higher weight on the vulnerable class(es). Since we do not know the performance yet, it is challenging to provide more concrete procedures. We take the following two scenarios into account. First, we assume each class is equally important for the model and the target is to decrease the inter-class discrepancy while minimizing the decrease of the overall average accuracy and/or robustness. Second, we assume that a certain class is critical and the target is to increase the accuracy and/or robustness for that class while minimizing the decrease of accuracy/robustness for other classes. Additionally, we consider including a regularizer term to decrease the inter-class cosine similarity. For adversarial training, we will experiment with a targeted attack by choosing the vulnerable or important class as the target class, which intuitively might make that class more robust. Depending on the performance, we will tailor the above strategies accordingly.

3.4. Additional explorations for robustness against natural corruptions

It has been shown in (Gilmer et al., 2019) that corruptions such as noise corruptions, fog or contrast influence standard and adversarially trained models differently. It would be insightful to see whether they might influence the class-wise performance differently. Moreover, comparing and analyzing the behavior between adversarial perturbation and natural corruptions can provide insight into the phenomenon of the inter-class discrepancy.

4. Experimental Results

The content above is the same as that in the original proposal except for some fixed minor language issues. The content below presents the experimental results that follow the protocol in the proposal. To improve readability, we structure the experimental results in the same order as the proposal.

4.1. The Universality of the Class-wise Accuracy Phenomenon

Dataset and Model Architectures. We report the class-wise phenomenon for different model architectures trained on different datasets. Since we are mainly interested in the inter-class discrepancy, we report the accuracy deviations from the average accuracy for each class as well as the cosine similarities between the distributions of the different models trained on a given dataset. Figure 3 shows the results on the digit classification datasets MNIST and SVHN. For different model architectures, a similar trend regarding the accuracy deviation can be observed. For MNIST, the LeNet architecture shows overall higher deviations than the ConvNet architecture. On SVHN, apart from a few classes, most architectures exhibit similar tendencies. This can be further observed from the cosine similarity matrices. The cosine similarity between the class-wise distribution between ConvNet and LeNet is 0.64 on MNIST. The average cosine similarity on SVHN is close to 0.84. Figure 4 and Figure 5 show the results on CIFAR and ImageNet image classification datasets, respectively. For the image classification datasets, it is more apparent that different model architectures show very similar trends for different classes. For example in Figure 4 for CIFAR10 and CIFAR100 all cosine similarity values are above 0.93 and 0.83, respectively. In Figure 5 for the ImageNet dataset with 1000 classes the cosine similarity of the class-wise accuracy distributions is always higher than 0.88. Wide-ResNet and ResNext exhibit the highest similarity with a value of 0.95.

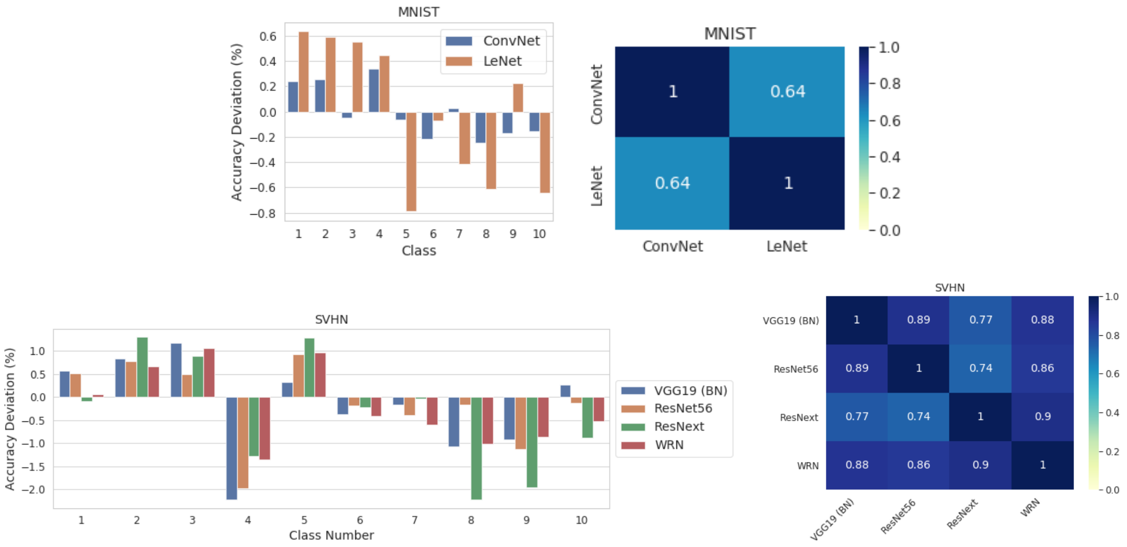

Figure 3: Class-wise deviation from the average accuracy (left) and cosine similarity of the class-wise accuracies of different model architectures (right) on digit classification datasets MNIST (top) and SVHN (bottom).

Influence of Optimization Factors. Here, we investigate the influence of gradient optimizer, learning rate schedule, and weight decay. The baseline was trained with stochastic gradient descent (SGD) and a weight decay of $1e^{-4}$. As an alternative to SGD, we test a model trained with the ADAM (Kingma and Ba, 2015) optimizer. Instead of a weight decay of $1e^{-4}$, we train an additional model with a weight decay of $1e^{-5}$. To test the influence of the learning rate schedule, we experiment with the cyclic learning rate (Smith, 2017). The accuracy deviation and cosine similarity plot are shown in Figure 6. It can be observed that none of the investigated optimization techniques has a significant influence on the class-wise accuracies, which is indicated by the cosine similarities being consistently higher than 0.97.

Evolution of Class-wise Accuracy Distribution During Training. Figure 7 shows the evolution of the class-wise accuracies during training for the task of digit classification and Figure 8 for image classification. It can be observed that in the early stages of training, the class-wise accuracies are very noisy and fluctuate significantly. In the latter stage of training with a smaller learning rate, there still exists the inter-class discrepancy

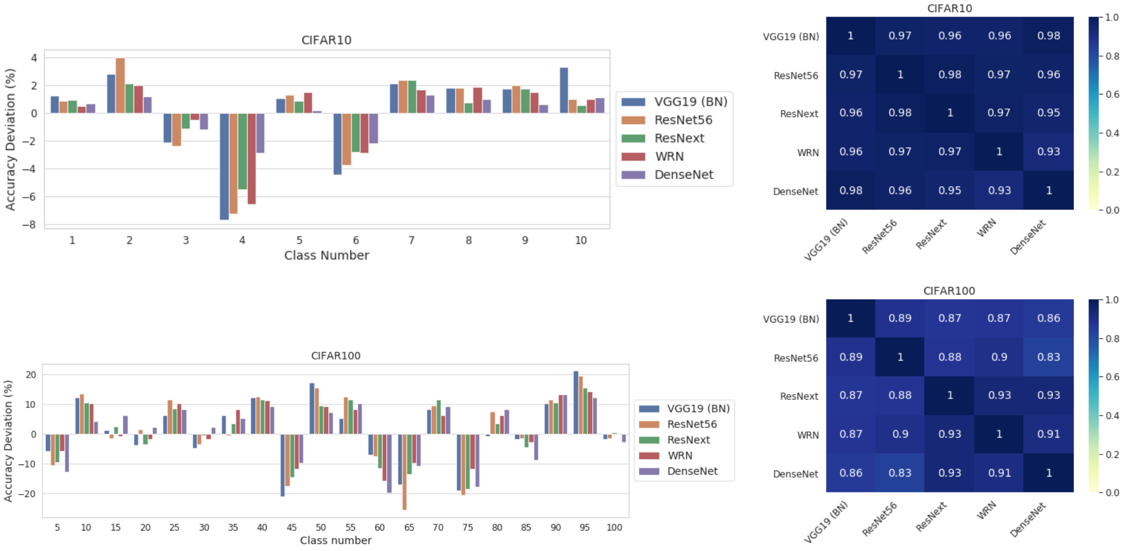

Figure 4: Class-wise deviation from the average accuracy (left) and cosine similarity of the class-wise accuracies of different model architectures (right) for different network architectures trained on CIFAR10 (top) and CIFAR100 (bottom).

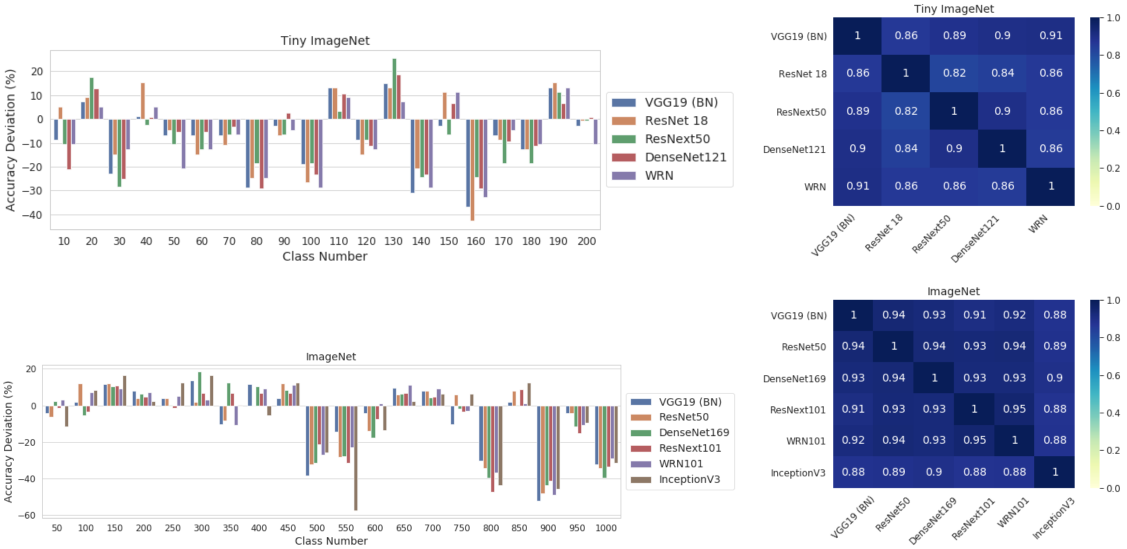

Figure 5: Class-wise deviation from the average accuracy (left) and cosine similarity of the class-wise accuracies of different model architectures (right) for different network architectures trained on Tiny-ImageNet (top) and ImageNet (bottom).

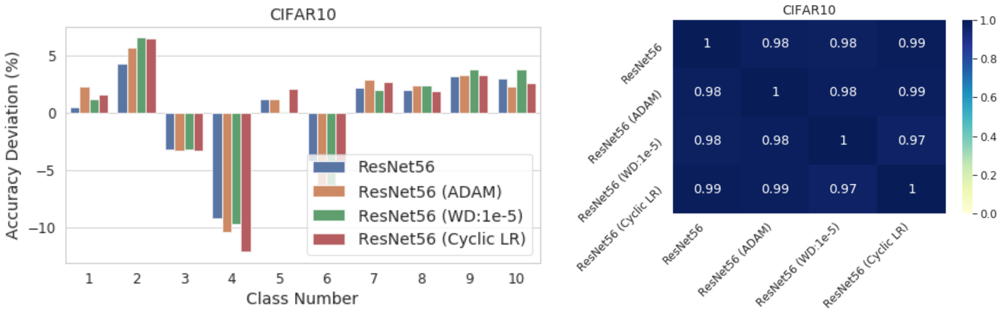

Figure 6: Influence of optimization factors on the class-wise accuracy distribution. The optimizer, weight decay, and learning rate schedule are investigated.

which, however, becomes more stable. For example, for MNIST, class 10 stays being the most vulnerable in the later epochs of training.

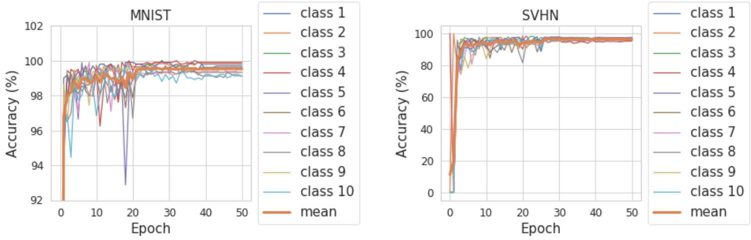

Figure 7: Class-wise accuracies during model training for digit classification.

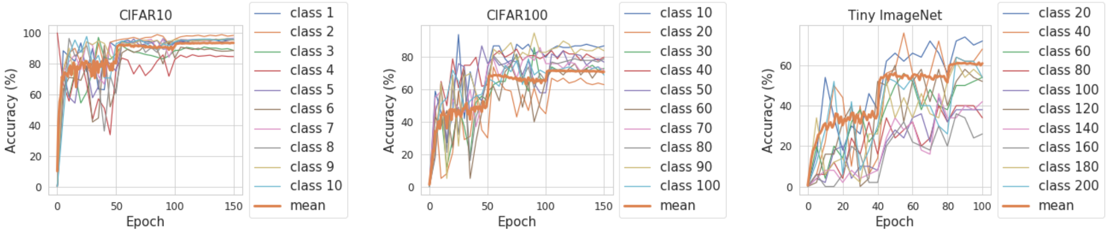

Figure 8: Class-wise accuracies during model training for image classification.

Adversarially Trained Models. We further investigate the class-wise accuracies and robustness of adversarially trained models. Figure 9 shows the comparison of a standard trained ResNet50 and adversarially trained ResNet50 as well as their cosine similarities. We observe that even between a normally trained model and an adversarially trained model a similar distribution can be observed. However, adversarially trained models usually exhibit higher class-wise discrepancies than models with standard training, as already indicated in our preliminary results in Figure 1 and Figure 2. The results suggest that robustness might be at odds with class-wise fairness. The class-wise accuracy deviations of adversarially trained models are shown in Figure 10 for 5 different adversarially trained models which are trained with PGD (Madry et al., 2018) bounded in the l_2-ball with an allowed perturbation magnitude of $\epsilon = 3.0$ (Salman et al., 2020). From the cosine similarity plot in Figure 11, it can be observed that all investigated adversarially trained models exhibit a high similarity, above 0.92, which is overall higher than that observed for the standard trained models in Figure 5. The results indicate that the similarity between robust models is higher than that of non-robust models, suggesting that robust models seem to learn similar robust features.

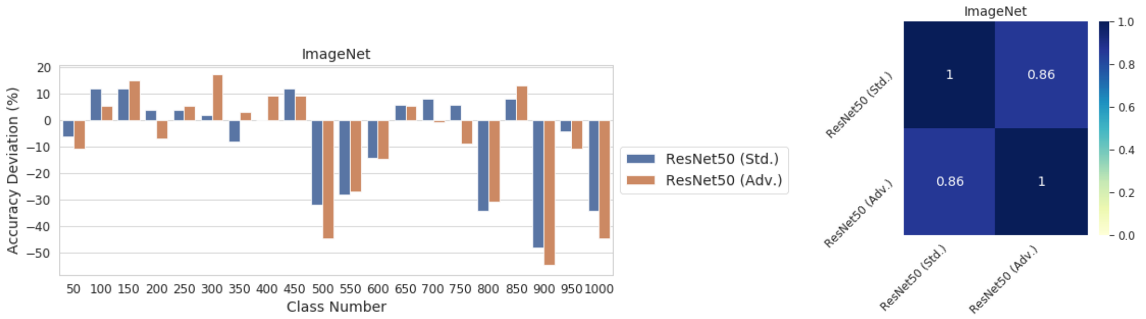

Figure 9: Class-wise deviation from the average accuracy (left) and cosine similarity (right) of a standard trained and adversarially trained ResNet50 architecture trained on ImageNet.

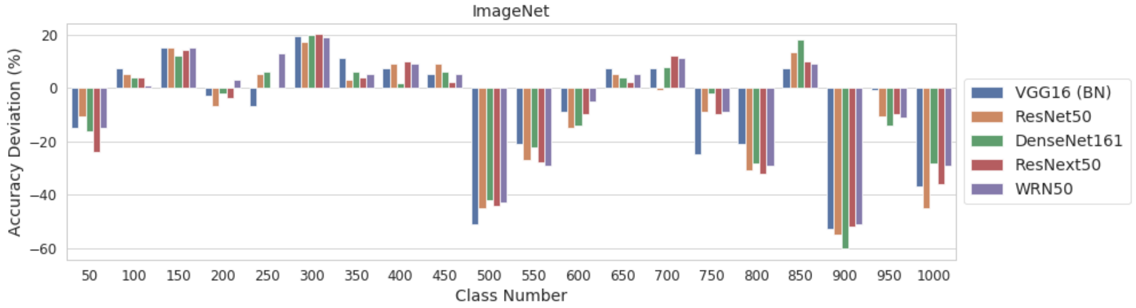

Figure 10: Class-wise deviation from the average accuracy of adversarially trained architectures trained on ImageNet.

4.2. Possible Explanations of the Class-wise Accuracy Discrepancy

Our above results demonstrate that the class-wise discrepancy phenomenon is universal in various setups regardless of the dataset classification type, model architectures, optimization factors, etc. Moreover, we find that on the same dataset, the cosine similarity between different models is always very high, indicating the class-wise discrepancy lies in the feature distribution of the dataset instead of model architectures or optimization factors. Thus,

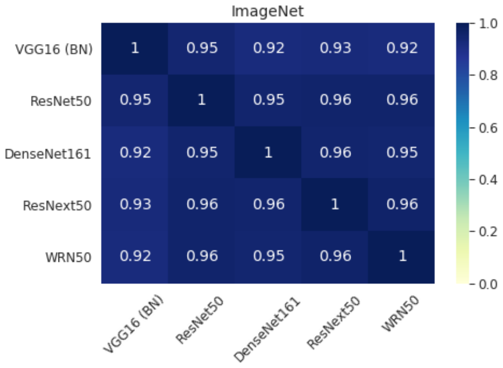

Figure 11: Cosine similarity between various architectures trained on ImageNet.

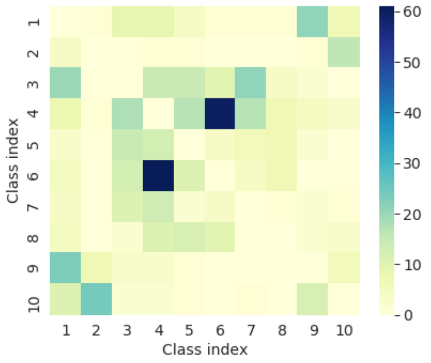

Figure 12: Confusion matrix of CIFAR10 on ResNet56.

we conjecture that the reason for low accuracy or robustness for a certain class can be attributed to the fact that another class having very similar features. For example, in CIFAR10, the class ”cat” and ”dog” have the lowest accuracies because they share more semantically similar features than other classes. To illustrate this, we plot a confusion matrix in Figure 12 where class 4 and 6 represent ”cat” and ”dog”, respectively. In the shown confusion matrix, the numbers indicate how many samples from the row class index are classified as the column class index. It is worth highlighting that the matrix shows a highly symmetric pattern, demonstrating the bidirectional confusion (or misclassification) between two classes often has a similar probability.

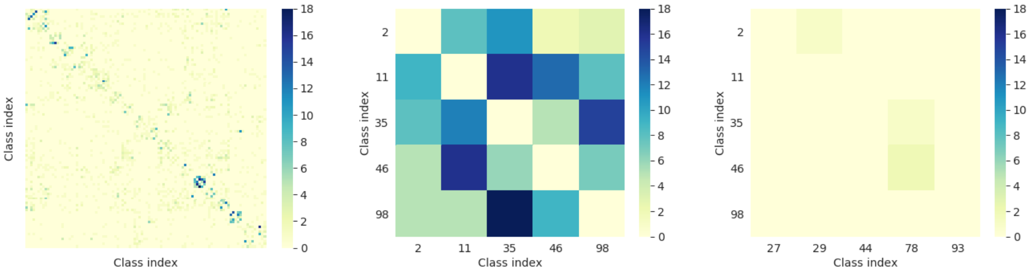

Figure 13: Confusion matrices for a ResNet56 trained on ImageNet for the entire dataset (left), and its super-classes ordered version (right).

A Study of the Class-wise Accuracy for Super-classes. With the assumption that confusion is more likely to happen between classes with similar semantic features, such as different dog breeds, we perform a study on CIFAR100 consisting of 20 super-classes, each having 5 sub-classes with semantically similar features. Figure 13 (left) shows that most confusions are located close to the diagonal, which already indicates that most confusions happen among the super-class. Note that the diagonal values were zeroed out, for better visualization. Further, since the CIFAR-100 dataset is not semantically sorted, we sort the class indices according to the CIFAR-100 super-classes. To further investigate this, we zoom in into the "people" super-class, for which the confusion matrix is shown in Figure 13 (center) as well as the confusion from the "people" super-class to the "reptiles" super-class in Figure 13 (right). It can be observed that the confusions are significantly higher among classes belonging to the super-class than towards a different super-class, where nearly no confusions can be observed.

Investigation from the Feature Perspective. Inspired by (Ilyas et al., 2019; Benz et al., 2020a), we generate datasets containing mainly non-robust features (NRFs) and robust features (RFs) from a standard model and an adversarially trained model, respectively. Figure 14 presents a ResNet18 trained NRFs and RFs (Madry et al., 2018). The results suggest that the class-wise discrepancy is observed on both, suggesting the robustness of the dataset features does not have a significant influence on the class-wise accuracy distribution.

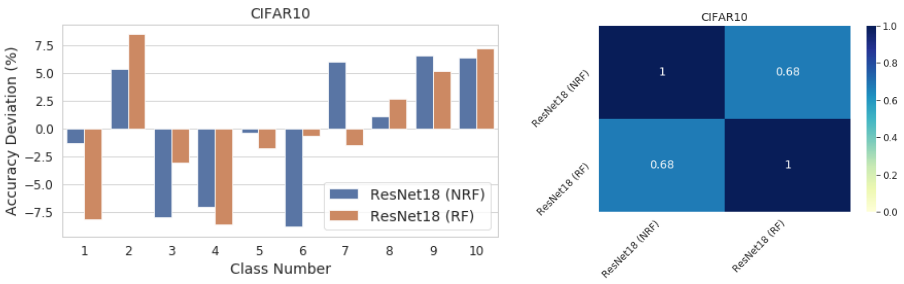

Figure 14: Class-wise accuracy deviation (left) and cosine similarity (right) for a ResNet18 trained on non-robust features (NRFs) and robust features (RFs).

4.3. Attempts to Mitigate the Class-wise Imbalance

Towards Fair Standard and Robust Training. We attempted to use the values of the confusion matrix as a regularizer to decrease the overall class-wise accuracy imbalance. Unfortunately, this approach did not lead to promising results, either significantly decreasing the overall accuracy or having only a limited effect on the class-wise imbalance. Here, we propose a simple approach to achieve a more balanced class-wise accuracy distribution. The method can be briefly summarized as a class-wise weighting of the cross-entropy loss. Hereby, the weighting adapts according to the class-wise accuracy deviations from the mean with a moving average. Specifically, given a weighting vector $\gamma \in \mathbb{R}^C$, whereby C indicates the number of classes and γ_c indicates the weighting specific for class c. In the beginning, γ is initialized to be all ones. After each epoch, the class-wise accuracies $\tau \in \mathbb{R}^C$ and the overall accuracy $\zeta \in \mathbb{R}$ are calculated. Based on the class-wise accuracy deviations from the mean, *i.e.* $(\tau - \zeta)$, the weighting values are updated with a value of α, which we set to 0.05. In short, the weighting value for a class is increased, if the class-wise accuracy is below the mean and the weighting value is decreased, if the class-wise accuracy is above the mean.

Mathematically, this can be expressed as:

$$\gamma_c = \gamma_c + \begin{cases} \alpha & \text{if } (\tau_c - \zeta) \leq 0, \\ -\alpha & \text{if } (\tau_c - \zeta) > 0. \end{cases} \tag{1}$$

In Figure 15, we present the results on ResNet56 for our fair weighting approach. For both standard training and adversarial training, the class-wise accuracy deviations are noticeably smaller than for the normal training without weighting. Specifically for normal standard training, the worst accuracy deviation is around -9% which is decreased to less than -3% for our fair weighting strategy. The improved fairness is at a small cost of overall accuracy, dropping from 92.9% to 91.1%. Similar observations can be made for adversarial training where the worst accuracy deviation decreases from around -30% to around -10%, at the cost of overall clean accuracy dropping from 78.9% to 75%. Moreover, we are interested in whether the class-wise robustness under adversarial attack also gets more imbalanced, which is confirmed by the results in Figure 15 (right). For example, the worst robustness deviation decreases from more than -30% to around -15% with overall robust accuracy dropping from 48.8% to 40.2%. It is worth mentioning that the robustness of most vulnerable class 4 increases by around 7% with our fair training strategy.

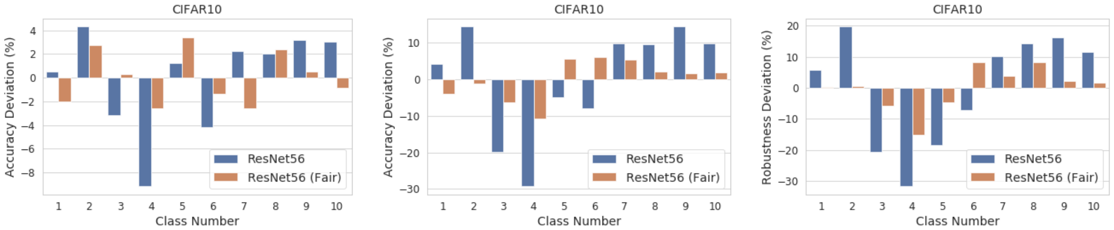

Figure 15: Comparison of the fair training method compared to normal model training for standard training (left) and adversarial training (center). For the adversarially trained models, the performance is also shown under adversarial attack (right).

Increasing the Class-wise Accuracy of an Important Class. In certain scenarios a specific class can be more important than others. For example in the case of autonomous driving the class "pedestrian" might be more important to detect than other classes. Hence, for such a class a higher class-wise accuracy would be required. We test if the weighting strategy can also be used to increase only the accuracy of an important class. Therefore we do not update the weight, but keep them fixed, with pre-defined values. Previously, we observed that for CIFAR-10 the class with index 4 usually exhibits the highest negative accuracy deviation from the mean accuracy. Hence, increasing class number 4 is the most challenging scenario, which we adopt for the following experiments. We set all values of γ to 1, except for γ_4, which we set to a higher value. Table 1 shows the influence of different γ_4 on the class-wise accuracy deviations and the cosine similarity. With increasing values of γ_4, an increase in the class-wise accuracy of 4 can be observed. However, the average accuracies also decrease with increasing γ_4 values.

Class-wise weight for class 4 (γ_4)	1	10	20	50	100
Class-wise accuracy for class 4 (τ_4)	83.7	84.1	84.6	93.1	97.0
Overall accuracy (ζ)	92.9	90.8	89.1	80.2	60.9

Table 1: Class-wise and overall accuracy result for increasing class with index 4 for different values of γ_4.

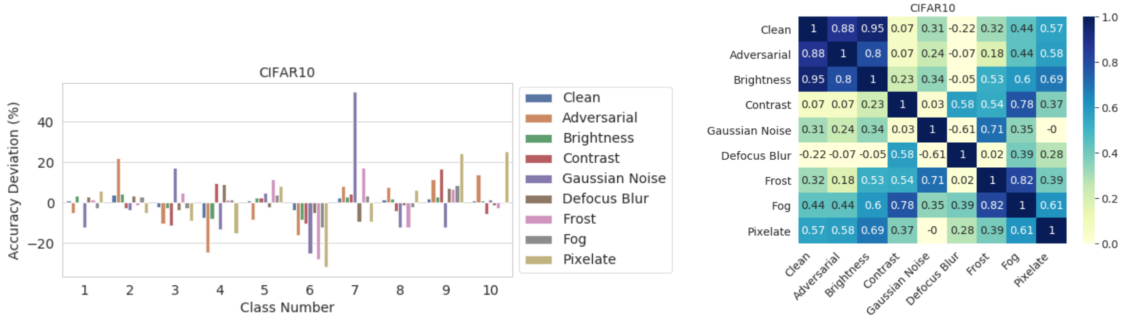

Figure 16: Class-wise accuracy deviation (left) and cosine similarity (right) for evaluating the influence of natural corruptions.

4.4. Additional Explorations for Robustness Against Natural Corruptions

Hendrycks and Dietterich (2019) demonstrates that DNNs can be vulnerable to corruptions. The authors (Hendrycks and Dietterich, 2019) provided corrupted CIFAR10 datasets, namely CIFAR10-C, on which we evaluate the class-wise accuracy deviations. We present the results on CIFAR10 with ResNet56 for different corruptions in Figure 16. Under corruptions, it is expected that the model accuracy significantly decreases. Specifically, the mean accuracies are 93.7%, 59.5%, 91.9%, 75.0%, 41.6%, 79.8%, 74.2% and 86.5% for without (clean), adversarial examples, brightness, contrast, Gaussian noise, defocus blur, frost, fog and pixelate corruption, respectively. It is worth mentioning that under adversarial perturbation, the most vulnerable class is still class 4, *i.e.* cat, while it becomes class 6 for most natural corruptions. For example, with the natural frost and pixelate corruptions, class 6 is significantly more vulnerable than other classes. Another interesting phenomenon is that the accuracy of class 7 with Gaussian noise is strikingly higher than other classes and a significant amount of images from other classes are misclassified as class 7.

5. Conclusion

Overall, this work performs an empirical study on the class-wise accuracy and robustness. We find that the phenomenon of inter-class discrepancy occurs in a wide range of setups, regardless of the dataset, model architecture, optimizer, learning schedule, etc. Adversarial training seems to make this inter-class discrepancy more serious, suggesting robustness might be at odds with inter-class fairness. The inter-class discrepancy can be, at least partially attributed to the fact that some classes share more similar features to other classes, causing confusion between those classes of similar features. Additionally, we find that the feature robustness, *i.e.* robust features or non-robust features, does not have much to do with this discrepancy phenomenon. Finally, we attempt to mitigate this phenomenon by adaptively adjusting the class-wise loss weight. The side effect is that it decreases the overall accuracy or robustness, further suggesting inter-class fairness might be not easily achieved without the cost of overall performance. Increasing the accuracy of a certain important class also leads to an overall accuracy decrease.

References

Philipp Benz, Chaoning Zhang, Tooba Imtiaz, and In-So Kweon. Data from model: Extracting data from non-robust and robust models. *CVPR workshop on Adversarial Machine Learning in Computer Vision*, 2020a.

Philipp Benz, Chaoning Zhang, Tooba Imtiaz, and In So Kweon. Double targeted universal adversarial perturbations. In *ACCV*, 2020b.

Philipp Benz, Chaoning Zhang, and In So Kweon. Batch normalization increases adversarial vulnerability: Disentangling usefulness and robustness of model features. *arXiv preprint arXiv:2010.03316*, 2020c.

Philipp Benz, Chaoning Zhang, Adil Karjauv, and In So Kweon. Revisiting batch normalization for improving corruption robustness. *WACV*, 2021a.

Philipp Benz, Chaoning Zhang, Adil Karjauv, and In So Kweon. Universal adversarial training with class-wise perturbations. *ICME*, 2021b.

Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 2018.

Kaidi Cao, Colin Wei, Adrien Gaidon, Nikos Arechiga, and Tengyu Ma. Learning imbalanced datasets with label-distribution-aware margin loss. In *NeurIPS*, 2019.

Nicholas Carlini and David Wagner. Towards evaluating the robustness of neural networks. In *Symposium on Security and Privacy (SP)*, 2017.

Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 2002.

Yin Cui, Menglin Jia, Tsung-Yi Lin, Yang Song, and Serge Belongie. Class-balanced loss based on effective number of samples. In *CVPR*, 2019.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *CVPR*, 2009.

Chris Drummond, Robert C Holte, et al. Class imbalance, and cost sensitivity: why under-sampling beats over-sampling. In *Workshop on learning from imbalanced datasets II*, 2003.

Charles Elkan. The foundations of cost-sensitive learning. In *IJCAI*, 2001.

Justin Gilmer, Nicolas Ford, Nicholas Carlini, and Ekin Cubuk. Adversarial examples are a natural consequence of test error in noise. In *ICML*, 2019.

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

Haibo He and Edwardo A Garcia. Learning from imbalanced data. *Transactions on knowledge and data engineering*, 2009.

Haibo He, Yang Bai, Edwardo A Garcia, and Shutao Li. Adasyn: Adaptive synthetic sampling approach for imbalanced learning. In *International joint conference on neural networks*, 2008.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

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

Chen Huang, Yining Li, Chen Change Loy, and Xiaoou Tang. Learning deep representation for imbalanced classification. In *CVPR*, 2016.

Gao Huang, Zhuang Liu, Laurens Van Der Maaten, and Kilian Q. Weinberger. Densely connected convolutional networks. In *CVPR*, 2017.

Andrew Ilyas, Shibani Santurkar, Dimitris Tsipras, Logan Engstrom, Brandon Tran, and Aleksander Madry. Adversarial examples are not bugs, they are features. In *NeurIPS*, 2019.

Nathalie Japkowicz and Shaju Stephen. The class imbalance problem: A systematic study. *Intelligent data analysis*, 2002.

Maurice George Kendall et al. The advanced theory of statistics. *The advanced theory of statistics.*, (2nd Ed), 1946.

Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Recurrent temporal aggregation framework for deep video inpainting. *TPAMI*, 2019.

Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Video panoptic segmentation. In *CVPR*, 2020.

Myungchul Kim, Sanghyun Woo, Dahun Kim, and In So Kweon. The devil is in the boundary: Exploiting boundary representation for basis-based instance segmentation. *WACV*, 2021.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Alex Krizhevsky et al. Learning multiple layers of features from tiny images. 2009.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 2015.

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

Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, Omar Fawzi, and Pascal Frossard. Universal adversarial perturbations. In *CVPR*, 2017.

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

Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon. Unsupervised intra-domain adaptation for semantic segmentation through self-supervision. In *CVPR*, 2020.

Shilin Qiu, Qihe Liu, Shijie Zhou, and Chunjiang Wu. Review of artificial intelligence adversarial attack and defense technologies. *Applied Sciences*, 2019.

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

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

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015.

Leslie N Smith. Cyclical learning rates for training neural networks. In *WACV*, 2017.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015.

Yuchun Tang, Yan-Qing Zhang, Nitesh V Chawla, and Sven Krasser. Svms modeling for highly imbalanced classification. *Transactions on Systems, Man, and Cybernetics*, 2008.

Matteo Terzi, Alessandro Achille, Marco Maggipinto, and Gian Antonio Susto. Adversarial training reduces information and improves transferability. *arXiv preprint arXiv:2007.11259*, 2020.

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. In *ICCV*, 2019.

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

Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. Adversarial examples: Attacks and defenses for deep learning. *IEEE Transactions on Neural Networks and Learning Systems*, 2019.

Chaoning Zhang, Francois Rameau, Seokju Lee, Junsik Kim, Philipp Benz, Dawit Mureja Argaw, Jean-Charles Bazin, and In So Kweon. Revisiting residual networks with nonlinear shortcuts. In *BMVC*, 2019a.

Chaoning Zhang, Philipp Benz, Tooba Imtiaz, and In-So Kweon. Cd-uap: Class discriminative universal adversarial perturbation. In *AAAI*, 2020a.

Chaoning Zhang, Philipp Benz, Tooba Imtiaz, and In-So Kweon. Understanding adversarial examples from the mutual influence of images and perturbations. In *CVPR*, 2020b.

Chaoning Zhang, Philipp Benz, Adil Karjauv, Geng Sun, and In Kweon. Udh: Universal deep hiding for steganography, watermarking, and light field messaging. *NeurIPS*, 2020c.

Chaoning Zhang, Francois Rameau, Junsik Kim, Dawit Mureja Argaw, Jean-Charles Bazin, and In So Kweon. Deepptz: Deep self-calibration for ptz cameras. In *WACV*, 2020d.

Chaoning Zhang, Philipp Benz, Dawit Mureja Argaw, Seokju Lee, Junsik Kim, Francois Rameau, Jean-Charles Bazin, and In So Kweon. Resnet or densenet? introducing dense shortcuts to resnet. In *WACV*, 2021a.

Chaoning Zhang, Philipp Benz, Adil Karjauv, and In So Kweon. Universal adversarial perturbations through the lens of deep steganography: Towards a fourier perspective. *AAAI*, 2021b.

Chaoning Zhang, Philipp Benz, Chenguo Lin, Adil Karjauv, Jing Wu, and In So Kweon. A survey on universal adversarial attack. *IJCAI*, 2021c.

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

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 *Proceedings of Machine Learning Research*, 2019c.