



Data Augmentation Can Improve Robustness

A Paper Presentation *for* Security and Privacy in Machine Learning

Aryan Ahadinia / Ramtin Moslemi

Department of Computer Engineering, Sharif University of Technology

May 14, 2024

A Paper of 35th Conference on Neural Information Processing Systems

Data Augmentation Can Improve Robustness

**Sylvestre-Alvise Rebuffi*, Sven Gowal*, Dan Calian,
Florian Stimberg, Olivia Wiles and Timothy Mann**
DeepMind, London
`{sylvestre,sgowal}@deepmind.com`

Today's Agenda

- 1 Introduction
- 2 Preliminaries
- 3 Data Augmentation Can Improve Robustness
- 4 Results and Conclusion

Abstract

- Adversarial training suffers from **robust over-fitting**.
- This paper focuses on reducing robust over-fitting by using **data augmentation**.
- Contrary to previous findings, when combined with model weight averaging, data augmentation can significantly boost robust accuracy.

Adversarial Examples

- Addition of imperceptible deviations to the input, called adversarial perturbations, can cause neural networks to make incorrect predictions with high confidence.
- The art of crafting increasingly sophisticated adversarial examples has received a lot of attention.
- Goodfellow et al. proposed the FGSM which generates adversarial examples with a single normalized gradient step.
- It was followed by R+FGSM, which adds a randomization step,
- and the BIM, which takes multiple smaller gradient steps.

Adversarial Training

- Adversarial training as proposed by Madry et al. is so effective that it is the de facto standard for training adversarially robust neural networks.
- The adversarial training procedure feeds adversarially perturbed examples back into the training data by formulating a saddle point problem to find model parameters θ that minimize the adversarial risk:

$$\arg \min_{\theta} \mathbb{E}_{(\mathbf{x}, y) \sim \mathcal{D}} \left[\max_{\delta \in \mathbb{S}} l(f(\mathbf{x} + \delta; \theta), y) \right]$$





- To solve the inner optimization problem, we can use PGD, to replace the non-differentiable 0-1 loss l with the cross-entropy loss l_{ce} and compute an adversarial perturbation $\hat{\delta} = \delta^{(K)}$ in K gradient ascent steps of size α as:

$$\delta^{(k+1)} \leftarrow \text{proj}_{\mathbb{S}} \left(\delta^{(k)} + \alpha \text{sign} \left(\nabla_{\delta^{(k)}} l_{\text{ce}}(f(\mathbf{x} + \delta^{(k)}; \theta), y) \right) \right)$$

- It has been augmented in different ways – with changes in the attack procedure (e.g., by incorporating momentum), loss function (e.g., logit pairing) or model architecture (e.g., feature denoising).
- Adversarial training suffers from a phenomenon known as **robust overfitting**.

Data Augmentation

- Data augmentation improves the generalization of standard (non-robust) training.
- For image classification tasks, random flips, rotations and crops are commonly used.
- There are more sophisticated techniques:
 - *Cutout* which produces random occlusions
 - *CutMix* which replaces parts of an image with another
 - *MixUp* which linearly interpolates between two images

	ResNet-50	Mixup	Cutout	CutMix
Image				
Label	Dog 1.0	Dog 0.5 Cat 0.5	Dog 1.0	Dog 0.6 Cat 0.4
ImageNet Cls (%)	76.3 (+0.0)	77.4 (+1.1)	77.1 (+0.8)	78.4 (+2.1)
ImageNet Loc (%)	46.3 (+0.0)	45.8 (-0.5)	46.7 (+0.4)	47.3 (+1.0)
Pascal VOC Det (mAP)	75.6 (+0.0)	73.9 (-1.7)	75.1 (-0.5)	76.7 (+1.1)

- Surprisingly they remain ineffective when training adversarially robust networks.
- In this work, we revisit these common augmentation techniques.

Preliminaries

Overfitting in adversarially robust deep learning

- One of the surprising characteristics of deep learning is the relative lack of overfitting.
- Models can be trained to zero training error without detrimental effects on generalization.
- In adversarial training, excessive training will continue to decrease the robust training loss, while increasing the robust test loss (Rice et al., 2020).

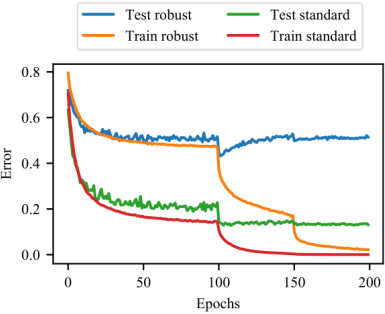


Figure: The learning curves for a robustly trained model replicating the experiment done by Madry et al. (2017) on CIFAR-10. The curves demonstrate “robust overfitting”; shortly after the first learning rate decay the model momentarily attains 43.2% robust error, and is actually more robust than the model at the end of training, which only attains 51.4% robust test error against a 10-step PGD adversary for ℓ_∞ radius of $\epsilon = 8/255$. The learning rate is decayed at 100 and 150 epochs.

Overcoming Robust Overfitting

- Rice et al. propose to use early stopping as the main contingency against robust overfitting, and demonstrate that it also allows to train models that are more robust than those trained with other regularization techniques (e.g. data augmentation or increased ℓ_2 -regularization).
- Other regularization techniques could reduce the impact of overfitting at the cost of producing models that are over-regularized and lack overall robustness and accuracy.
- There is one notable exception which is the addition of **external data**.

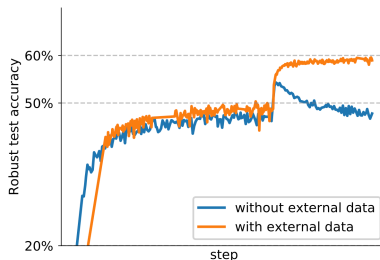


Figure: Adversarial training with and without additional data from 80M-TI

Averaging Weights Leads to Wider Optima and Better Generalization

- Deep neural networks are typically trained by optimizing a loss function with an SGD variant, in conjunction with a decaying learning rate, until convergence.
- Simple averaging of multiple points along the trajectory of SGD, with a cyclical or constant learning rate, leads to better generalization than conventional training.
- Stochastic Weight Averaging (SWA) procedure finds much flatter solutions than SGD.
- In short, SWA is extremely easy to implement, improves generalization, and has almost no computational overhead.
- Model weight averaging (WA) can be implemented using an exponential moving average θ' of the model parameters θ with a decay rate τ :

$$\theta' \leftarrow \tau \cdot \theta' + (1 - \tau) \cdot \theta$$

- WA can significantly improve robustness on a wide range of models and datasets.

Model Weight Averaging

- WA leads to a **flatter adversarial loss landscape** and a smaller robust generalization gap.
- WA reduces sensitivity to early stopping.
- However WA is still prone to robust overfitting since the exponential moving average “forgets” older model parameters as training goes on.
- We observe that, after the change of learning rate, the averaged weights are increasingly affected by overfitting, thus resulting in worse robust accuracy for the averaged model.

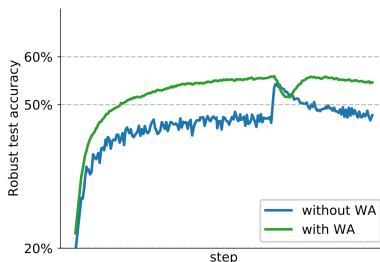
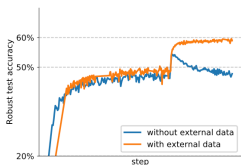
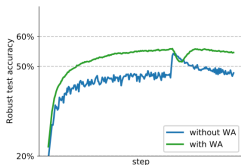


Figure: Effect of WA without external data

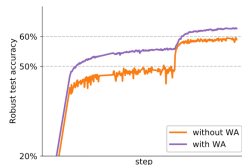
Effects of External Data and Weighted Averaging



(a) Adversarial training with and without additional data from 80M-TI (without WA)



(b) Effect of WA without external data



(c) Effect of WA with external data

Figure: We compare the robust accuracy against $\epsilon_\infty = 8/255$ on CIFAR-10 of an adversarially trained Wide ResNet (WRN)-28-10. Panel (a) shows the impact of using additional external data from 80M-TI and illustrates robust overfitting. Panel (b) shows the benefit of model weight averaging (WA) despite robust overfitting. Panel (c) shows that WA remains effective and useful even when robust overfitting disappears. The graphs show the evolution of the robust accuracy as training progresses (against PGD^{40}). The jump two-thirds through training is due to a drop in learning rate.

Data Augmentation Can Improve Robustness

Contributions

- Demonstrate data augmentation techniques such as *Cutout*, *CutMix* and *MixUp* can improve robustness when paired with WA.
- To the contrary of Gowal et al., Rice et al., Wu et al. we are able to use any of these three aforementioned techniques to obtain new **state-of-the-art robust accuracies**.

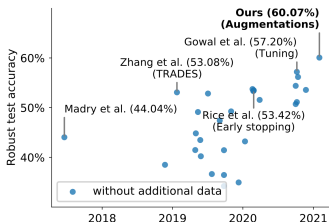


Figure: Robust accuracy of various models submitted to RobustBench against AUTOATTACK on CIFAR-10 with ℓ_∞ perturbations of size 8/255 displayed in publication order. Our method builds on Gowal et al. and explores how augmented data can be used to improve robust accuracy by +2.87% without using any additional external data.

- Show that approach generalizes across architectures, datasets and threat models.
- Investigate the trade-off between robust overfitting and underfitting.
- Provide empirical evidence that WA exploits data augmentation by ensembling snapshots.

Hypothesis

- As WA results in flatter, wider solutions compared to the steep decrease in robust accuracy observed for SGD, it is natural to ask ourselves whether WA remains useful in cases that do not exhibit robust overfitting.
- We notice that the robust performance in this setting is not only preserved but even boosted when using WA.
- Hence, we formulate the hypothesis that:
model weight averaging helps robustness to a greater extent when robust accuracy between model iterations can be maintained.
- This hypothesis is also motivated by the observation that WA acts as a temporal ensemble – akin to Fast Geometric Ensembling by Garipov et al. who show that efficient ensembling can be obtained by aggregating multiple checkpoint parameters at different training times.

Limiting Robust Overfitting Without External Data

- Rice et al. show that combining data augmentation methods such as *Cutout* or *MixUp* with early stopping does not improve robustness upon early stopping alone.
- While, these methods do not improve upon the “best” robust accuracy, they reduce the extent of robust overfitting, thus resulting in a slower decrease in robust accuracy compared to classical adversarial training (which uses random crops and weight decay).

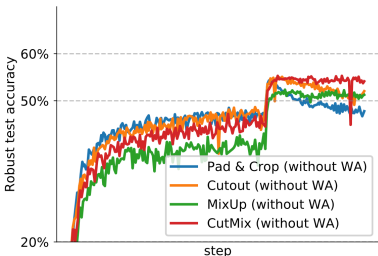


Figure: Accuracy against $\epsilon_{\infty} = 8/255$ on CIFAR-10 without using model WA for different data augmentation schemes. The model is a WRN-28-10 and the panel shows the evolution of the robust accuracy as training progresses (against PGD^{40}). The jump in robust accuracy two-thirds through training is due to a drop in learning rate.

Testing the Hypothesis

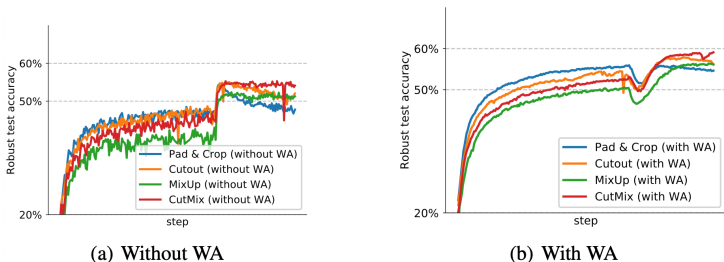


Figure: Accuracy against $\epsilon_{\infty} = 8/255$ on CIFAR-10 without using model weight averaging (WA) for different data augmentation schemes. The model is a WRN-28-10 and both panels show the evolution of the robust accuracy as training progresses (against PGD^{40}). The jump in robust accuracy two-thirds through training is due to a drop in learning rate. The accuracy drop just after the change of learning rate stems from averaging very different weights.

Experimental Setup

- **Architecture.** We use WRNs as our backbone network. Most of the experiments are conducted on a WRN-28-10 model which has a depth of 28, a width multiplier of 10 and contains 36M parameters. To evaluate the effect of data augmentations on wider and deeper networks, we also run several experiments using WRN-70-16, which contains 267M parameters.
- **Outer minimization.** We use TRADES optimized using SGD with Nesterov momentum and a global weight decay of 5×10^{-4} .
- **Inner minimization.** Adversarial examples are obtained by maximizing the Kullback-Leibler divergence between the predictions made on clean inputs and those made on adversarial inputs.
- **Evaluation.** We train two (and only two) models for each hyperparameter setting, perform early stopping for each model on a separate validation set using *PGD*⁴⁰. Finally, we report the robust test accuracy against a mixture of AUTOATTACK and MULTITARGETED, which is denoted by AA+MT.

Results and Conclusion

Comparing Data Augmentations

- WA is the most beneficial when robust overfitting is reduced.
- Spatial composition techniques which outperform blending techniques.

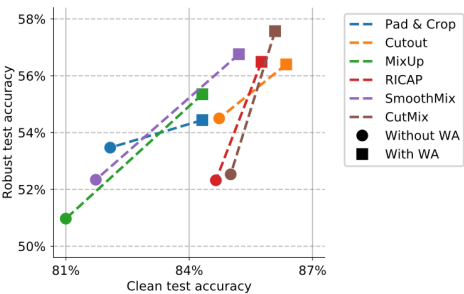


Figure: Clean (without adversarial attacks) accuracy and robust accuracy (against AA+MT) for a WRN-28-10 trained against $\epsilon_{\infty} = 8/255$ on CIFAR-10 for different data augmentation techniques. The lines from circles to squares represent the performance change obtained when using WA.

Blending Techniques

- *MixUp* samples the image mixing weight with a beta distribution $\text{Beta}(\alpha, \alpha)$
 - tends to either produce images that are far from the original data distribution (when α is large)
 - or too close to the original samples (when α is small)
- Increasing α can lead to robust underfitting while an α too close to 0 would lead to robust overfitting.

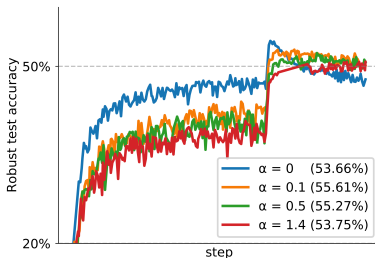


Figure: The graph shows the robust test accuracy against PGD^{40} with $\epsilon_\infty = 8/255$ on CIFAR-10 without using WA as we vary the mixing rate α of *MixUp*. We report in the legend the robust accuracy (against AA+MT) after applying weight averaging to the corresponding runs.

Spatial Composition Techniques

- *Cutout* and *CutMix* are most beneficial when using large window lengths.
- Low-level features tend to be destroyed by *MixUp*,
- whereas composition techniques locally maintain these low-level features.
- Hence, we hypothesize augmentations designed for robustness need to preserve low-level features.

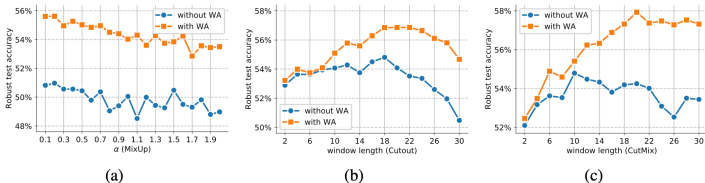


Figure: Robust test accuracy against AA+MT with $\epsilon_\infty = 8/255$ on CIFAR-10 as we vary (a) the mixing rate α of MixUp, (b) the window length when using Cutout and (c) the window length when using CutMix. The model is a WRN-28-10 and we compare the settings without and with WA.

Generalizing to other Architectures

SETUP	PAD & CROP		CUTMIX	
	CLEAN	ROBUST	CLEAN	ROBUST
VARYING THE ARCHITECTURE				
ResNet-18	83.12%	50.52%	80.57%	52.28%
ResNet-34	84.68%	52.52%	83.35%	54.80%
WRN-28-10	84.32%	54.44%	86.09%	57.50%
WRN-34-10	84.89%	55.13%	86.18%	58.09%
WRN-34-20	85.80%	55.69%	87.80%	59.25%
WRN-70-16	86.02%	57.17%	87.25%	60.07%

Figure: Robust test accuracy (against AA+MT) against $\epsilon_{\infty} = 8/255$ on CIFAR-10 for different architectures. In all cases, we use weight averaging and we compare *Pad & Crop* and *CutMix*.

Generalizing to other Threat Models

SETUP	ℓ_∞		ℓ_2	
	CLEAN	ROBUST	CLEAN	ROBUST
WRN-28-10				
Gowal et al. [20] (trained by us)	84.32%	54.44%	88.60%	72.56%
Ours (CutMix)	86.22%	57.50%	91.35%	76.12%
WRN-70-16				
Gowal et al. [20] (trained by us)	85.29%	57.14%	90.90%	74.50%
Ours (CutMix)	87.25%	60.07%	92.43%	76.66%

Figure: Clean (with and without adversarial attacks) accuracy and robust accuracy (against AA+MT) on CIFAR-10 as we both test against $\epsilon_\infty = 8/255$ and $\epsilon_2 = 128/255$.

Generalizing to other Datasets

MODEL	CLEAN	AA+MT	AA
CIFAR-100			
Cui et al. [14] (WRN-34-10)	60.64%	–	29.33%
WRN-28-10 (retrained)	59.05%	28.75%	–
WRN-28-10 (CutMix)	62.97%	30.50%	29.80%
Gowal et al. [20] (WRN-70-16)	60.86%	30.67%	30.03%
WRN-70-16 (retrained)	59.65%	30.62%	–
WRN-70-16 (CutMix)	65.76%	33.24%	32.43%
SVHN			
WRN-28-10 (retrained)	92.87%	56.83%	–
WRN-28-10 (CutMix)	94.52%	57.32%	–
TINYIMAGENET			
WRN-28-10 (retrained)	53.27%	21.83%	–
WRN-28-10 (CutMix)	53.69%	23.83%	–

Figure: Clean and robust accuracy (AA+MT and AutoAttack for select models) on CIFAR-100, SVHN and TINY-IMAGENET against $\epsilon_\infty = 8/255$ obtained by different models (with WA). The 'retrained' indication means that the models have been retrained according to Gowal et al.'s methodology.

Model Ensembling by Weight Averaging

Ensembling improves robustness by exploiting the diversity of equally performing models

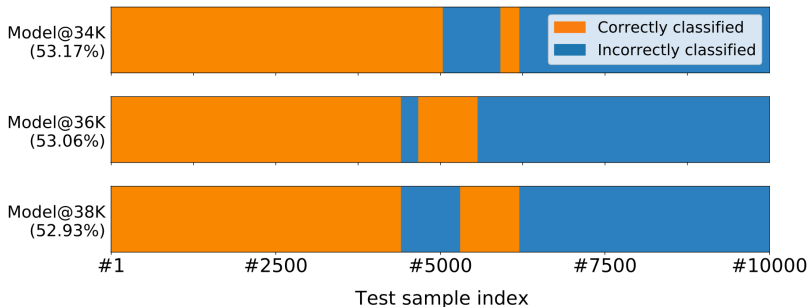


Figure: The bar plots show the outcome of each individual robust prediction for different snapshots of a same training run of a WRN-28-10 against $\epsilon_{\infty} = 8/255$ on CIFAR-10 without model weight averaging. The test sample indices have been re-ordered such as to show contiguous blocks. The plots show a significant variation in individual robust predictions across different snapshots while the total robust accuracy (i.e. the number in parenthesis) remains stable.

Limits of Exploiting Diversity

The diversity between model iterations can only compensate up to a certain point for the decrease in robust performance due to robust overfitting.

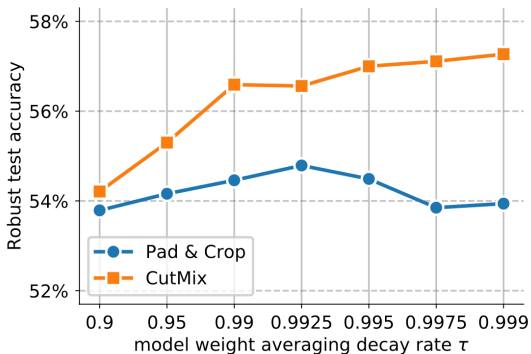


Figure: Robust test accuracy against AA+MT with $\epsilon_\infty = 8/255$ on CIFAR-10 as we vary the decay rate of the model weight averaging. The model is a WRN-28-10, which is trained either with *CutMix* or *Pad & Crop*.

Conclusion

- Contrary to previous works, which have tried data augmentation techniques to train adversarially robust models without success, we demonstrate that combining data augmentations with model weight averaging can significantly improve robustness.
- We also provide insights on why weight averaging works better with data augmentations which reduce robust overfitting.
- We show in fact that model snapshots of a same run have the same total robust accuracy but they greatly differ at the individual prediction level, thus allowing a performance boost when ensembling these snapshots.

Future Works

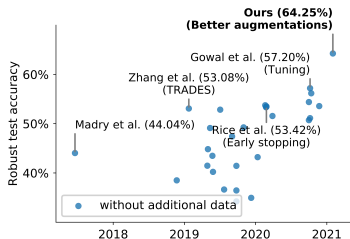


Figure: Fixing Data Augmentation to Improve Adversarial Robustness

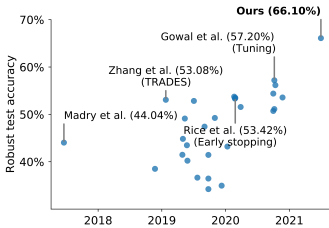








Figure: Improving Robustness using Generated Data

References

-  S. Rebuffi, S. Goyal, D. A. Calian, F. Stimberg, O. Wiles, and T. A. Mann, “Data augmentation can improve robustness,” *CoRR*, vol. abs/2111.05328, 2021.
-  P. Izmailov, D. Podoprikin, T. Garipov, D. P. Vetrov, and A. G. Wilson, “Averaging weights leads to wider optima and better generalization,” *CoRR*, vol. abs/1803.05407, 2018.
-  L. Rice, E. Wong, and J. Z. Kolter, “Overfitting in adversarially robust deep learning,” *CoRR*, vol. abs/2002.11569, 2020.
-  S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, “Cutmix: Regularization strategy to train strong classifiers with localizable features,” *CoRR*, vol. abs/1905.04899, 2019.
-  S. Rebuffi, S. Goyal, D. A. Calian, F. Stimberg, O. Wiles, and T. A. Mann, “Fixing data augmentation to improve adversarial robustness,” *CoRR*, vol. abs/2103.01946, 2021.
-  S. Goyal, S. Rebuffi, O. Wiles, F. Stimberg, D. A. Calian, and T. A. Mann, “Improving robustness using generated data,” *CoRR*, vol. abs/2110.09468, 2021.

Thank You