Backpropagating Smoothly Improves Transferability of Adversarial Examples

Chaoning Zhang*
chaoningzhang1990@gmail.com

Philipp Benz*
pbenz@kaist.ac.kr

Gyusang Cho*
gyusang.cho@kaist.ac.kr

Adil Karjauv
mikolez@gmail.com

Soomin Ham
smham@kaist.ac.kr

Chan-Hyun Youn
chyoun@kaist.ac.kr

In So Kweon
iskweon77@kaist.ac.kr

Korea Advanced Institute of Science and Technology (KAIST)

Abstract

Probably the most popular yet controversial explanation for adversarial examples is the hypothesis on the linear nature of modern DNNs. Initially supported by the FGSM-attack this has been challenged by prior works from various perspectives. Further aligning with the linearity hypothesis, a recent work shows that backpropagating linearly (LinBP) improves the transferability of adversarial examples. One widely recognized issue of the commonly used ReLU activation function is that its derivative is non-continuous. We conjecture that the reason LinBP improves the transferability is mainly due to a continuous approximation for the ReLU in the backward pass. In other words, backpropagating continuously might be sufficient for improving transferability. To this end, we propose ConBP that adopts a smooth yet non-linear gradient approximation. Our ConBP consistently achieves equivalent or superior performance than the recently proposed LinBP, suggesting the core source of improved transferability lies in the approximation derivative being smooth, regardless of being linear or not. Our work highlights that any new evidence for either supporting or refuting the linearity hypothesis deserves a closer look. As a byproduct, our investigation also results in a new variant backpropagation method for improving the transferability of adversarial examples.

1. Introduction

Deep neural networks (DNNs) have been widely known to be vulnerable to adversarial examples [8, 22]. One intriguing phenomenon of adversarial examples is their transferable property, *i.e*. adversarial examples generated on a certain model can transfer well to another unseen black-model [5, 6, 27]. Numerous works have attempted to explain the existence of adversarial examples as well as their transferability [8, 11, 13, 24]. One of the most famous yet controversial explanations is the linear nature hypothesis of modern DNNs [8]. The linear hypothesis has been mainly supported by the success of the widely used fast gradient sign method (FGSM) [8]. However, follow-up works have refuted this linear hypothesis from various perspectives [21, 24].

One recent work [9] revisits this hypothesis and demonstrates that backpropagating linearly (LinBP), can non-trivially improve the transferability of the adversarial examples. The authors ascribe the improved transferability of LinBP to the linear nature of modern DNNs, thus constituting another strong empirical evidence for supporting the controversial linearity hypothesis. Since this controversial hypothesis is highly relevant for understanding the adversarial examples, we argue that any new evidence that either supports or refutes it deserves a closer look for not misleading the community in the wrong direction of exploring adversarial examples. To this end, our work revisits this recent evidence supported by their LinBP. LinBP disentangles the ReLU in the forward and backward pass. Specifically, they calculate the ReLU as normal in the forward pass but treats it as an identity mapping with a constant derivative in the backward pass. Note that the derivatives of ReLU are non-continuous at zero, and this discontinuity might result in unstable gradients near zero and consequently decreases the gradient quality and transferability. Instead, the identity mapping has a continuous derivative, which motivates a conjecture that the continuous property of the approximation gradient might be the cause for the improved transferability of LinBP. The core issue this work addresses is what property of the adopted approximation derivative boosts the transferability of adversarial examples. LinBP [9] attributes it to the linear property, while we conjecture the primary cause might be the continuous property, regardless of being linear or not.

To this end, we investigate whether backpropagating smoothly but non-linearly improves transferability. Analo-

[*] Equal contribution

gous to LinBP, we term our proposed approach ConBP. We find that ConBP consistently achieves comparable or superior transferability compared with the existing LinBP. Our findings empirically collaborate our conjecture that the continuous property of the approximation derivative is one key factor improving transferability. In other words, the success of LinBP for improving transferability might not constitute strong evidence for supporting the famous yet controversial linear hypothesis. Our investigation highlights the necessity of providing a closer look at any new evidence for supporting or refuting the linearity hypothesis. As a by-product of our investigation, our investigation also leads to a new backpropagation variant for improving transferability.

2. Background and Related work

White-box attack. Suppose we are given a classifier $f(x) : \mathcal{X} \rightarrow \mathcal{Y}$ which predicts the label of a sample. The classifier $f(x)$ is pretrained on a dataset $\mathcal{D}$ of samples (x, y), $x \in \mathbb{R}^d$ and their corresponding ground-truth label $y \in \{1, \ldots, k\}$. Assuming the classifier is well trained, and usually $f(x) = y$ and adversarial attack seeks a small perturbation δ that fools the classifier, *i.e.* $f(x^{adv}) \neq y$, where $x^{adv} = x + \delta$. In the targeted setting, the new prediction needs to be a predefined target class y^t, *i.e.* $f(x^{adv}) = y^t$. The de facto standard approaches maximize the prediction loss *e.g.* cross-entropy loss $\mathcal{L}$ with an additional constraint on the perturbation l_p norm as:

$$\arg\max_{\delta} \mathcal{L}(f(x+\delta), y), \quad \text{s.t. } ||\delta||_\infty \leq \epsilon, \tag{1}$$

Inspired by their linear hypothesis, Goodfellow *et al*. proposed FGSM for simply calculating the perturbation as $\epsilon \cdot sign(\nabla_x \mathcal{L}(x_t^{adv}, y))$. Despite its efficiency, FGSM often suffers from a low attack success rate especially when the ϵ is small. I-FGSM [14] and PGD [16] are iterative variants of FGSM that result in a stronger attack. Note that the core difference between I-FGSM and PGD lies in whether to initialize the initial values with random perturbations. PGD with the random initialization can be stronger given multiple trials for the white-box attack. Another variant of white-box attack attempts to minimize the perturbation magnitude and two famous representative ones are DeepFool [17] and CW attack [2].

Black-box attack. One intriguing property of adversarial examples is that they are transferable, which facilitates the transfer-based black-box attack. Another query-based variant of black-box attack also exists and typically requires numerous queries to the black-box model [1, 3, 4, 7, 18, 20, 28]. Along this direction, transferability has also been exploited for reducing the number of queries. As the core of most black-box methods, adversarial transferability has attracted significant attention since [8] attributes it to the linear nature of DNNs. Compared with FGSM, I-FGSM constitutes a stronger white-box attack but at the cost of lower transferability. Ensembling multiple source models has been found in [15] to boost transferability. Processing the inputs or the gradients with input diversity [27], gradient momentum [5], smoothing kernel [6] has been found beneficial for transferability enhancement. Optimizing the loss on the feature level is found in [10, 12, 29] to boost the transferability. It has been shown in [9, 25] that disentangling some components in the forward and backward pass is beneficial for improving the performance. Specifically, [25] shows that using more gradients from the skip connections rather than the residual modules in the backward pass results in more transferable adversarial examples, and [9] shows that LinBP, *i.e.* discarding some non-linear functions in the backward pass, improves transferability. Our work is mainly inspired by LinBP but differs in replacing the linear derivatives with nonlinear but smooth approximations. A similar approach has also been investigated in a concurrent work for adversarial training [26], while ours focuses on its influence on transferability.

3. Motivation

Revisiting the linear hypothesis. Since the discovery of adversarial examples, numerous works have attempted explanation from various perspectives. The probably most famous yet controversial one is the linear nature hypothesis first introduced in [8]. This hypothesis is in contrast to the earlier prevailing belief that the non-linearity of DNNs is the cause. This linearity hypothesis is also partially supported by [13] that studied adversarial examples in the context of dense associate memory models. However, this hypothesis has also been challenged by multiple follow-up works. For example, Tanay and Griffin [24] provided an alternative boundary tilting perspective on the cause while claiming the linearity hypothesis is “unconvincing”. This is also collaborated by the finding that a shallow and thus more linear classifier is just as vulnerable to the adversarial examples as their much deeper counterpart [23]. A relatively more recent work [21] discovered a strong correlation between robustness and empirical linearity of a network, motivating the authors to “reject” the linearity hypothesis. A similar finding has also been reported in [19]. Despite these counter-evidences, one recent work still builds their explanation on the linearity hypothesis and proposed LinBP for improving the transferability. Specifically, LinBP discards the non-linear functions in the backward pass for calculating the gradient, thus is considered as linear backpropagation, *i.e.* LinBP. This intriguing phenomenon constitutes new non-trivial evidence for supporting the long-existing controversial hypothesis. Since it is highly relevant to explain the adversarial examples, any new evidence that either supports or refutes the linearity hypothesis deserves a closer look. To this end, our work revisits why LinBP im-

Table 1. Success rates of transfer-based attacks on *ImageNet* using I-FGSM with ℓ_∞ constraint under the untargeted setting. The source model is a VGG19 and the symbol * indicates that the victim model is the same as the source model. Average is obtained from models different from the source.

Dataset	Method	#layers	VGG* (2016)	ResNet (2015)	Inception v3 (2016)	DenseNet (2017)	MobileNet v2 (2018)	PNASNet (2018)	SENet (2018)	**Average**
ImageNet	I-FGSM	N.A.	**100.00%**	40.00%	28.00%	38.00%	54.00%	28.00%	38.00%	37.66%
	LinBP+I-FGSM	1	**100.00%**	45.00%	25.00%	40.00%	56.00%	30.00%	35.00%	38.5%
		3	**100.00%**	56.00%	33.00%	50.00%	65.00%	31.00%	45.00%	**46.67%**
		5	**100.00%**	49.00%	31.00%	44.00%	57.00%	33.00%	39.00%	42.16%
		7	**93.00%**	51.00%	34.00%	52.00%	56.00%	30.00%	35.00%	43.00%
		9	**92.00%**	24.00%	24.00%	23.00%	40.00%	13.00%	18.00%	23.68%
		11	**77.00%**	23.00%	22.00%	30.00%	38.00%	15.00%	17.00%	24.68%
		13	**79.00%**	25.00%	20.00%	15.00%	29.00%	5.00%	13.00%	17.83%
		15	**66.00%**	18.00%	18.00%	16.00%	32.00%	8.00%	13.00%	17.5%
	ConBP+I-FGSM	1	**100.00%**	41.00%	25.00%	42.00%	53.00%	29.00%	32.00%	37.00%
		3	**100.00%**	46.00%	30.00%	47.00%	59.00%	34.00%	34.00%	41.64%
		5	**100.00%**	48.00%	33.00%	48.00%	58.00%	35.00%	36.00%	43.0%
		7	**100.00%**	50.00%	31.00%	52.00%	63.00%	38.00%	41.00%	45.83%
		9	**100.00%**	48.00%	31.00%	52.00%	63.00%	40.00%	44.00%	46.33%
		11	**100.00%**	49.00%	28.00%	54.00%	63.00%	38.00%	45.00%	46.14%
		13	**100.00%**	56.00%	34.00%	56.00%	67.00%	41.00%	45.00%	49.83%
		15	**100.00%**	60.00%	37.00%	63.00%	73.00%	46.00%	49.00%	**54.64%**

Table 2. Success rates of transfer-based attacks on *ImageNet* using I-FGSM with ℓ_∞ constraint under the untargeted setting. Note that renormalization is **Not** applied. The source model is a ResNet-50 and the symbol * indicates that the victim model is the same as the source model. Average is obtained from models different from the source.

Dataset	Method	ϵ	ResNet* (2016)	Inception v3 (2016)	DenseNet (2017)	MobileNet v2 (2018)	PNASNet (2018)	SENet (2018)	Average
ImageNet	I-FGSM	0.1	**100.00%**	48.00%	73.00%	68.00%	43.00%	54.00%	57.2%
		0.05	**100.00%**	27.00%	54.00%	54.00%	25.00%	31.00%	38.2%
		0.03	**100.00%**	21.00%	47.00%	46.00%	21.00%	25.00%	32.00%
	LinBP+I-FGSM(†)	0.1	**100.00%**	60.00%	82.00%	83.00%	62.00%	77.00%	72.8%
		0.05	**98.00%**	37.00%	59.00%	59.00%	32.00%	42.00%	45.8%
		0.03	**94.00%**	26.00%	43.00%	45.00%	16.00%	25.00%	31.00%
	ConBP + I-FGSM	0.1	**100.00%**	85.00%	96.00%	97.00%	87.00%	93.00%	**91.6%**
		0.05	**100.00%**	62.00%	89.00%	84.00%	65.00%	76.00%	**75.2%**
		0.03	**100.00%**	47.00%	76.00%	73.00%	48.00%	54.00%	**59.6%**

proves transferability. One widely known issue of ReLU in the backpropagation is its non-continuous derivative at zero (See Figure 1). By discarding the ReLU in the backward propagation, LinBP in essence alleviates this discontinuity issue with a linear thus continuous derivative. This motivates a conjecture that the improved transferability of LinBP lies in the continuous property of the approximation derivative.

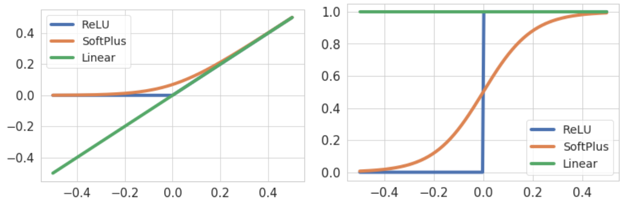

Figure 1. Activation functions (left) and their derivatives (right).

ConBP: Backpropagating Smoothly. Note that the linear property is one special form of the continuous property. Thus, to investigate whether the continuous derivative property is sufficient for improving transferability, we adopt a continuous yet non-linear gradient approximation for the ReLU activation. Since it backpropagates continuously, analogous to LinBP we term it ConBP. Specifically, ConBP adopts the ReLU function as normal in the forward pass but uses the continuous approximation of a certain non-linear activation function, *e.g.* softplus function. In essence, both LinBP and ConBP disentangle the ReLU in the forward and backward pass and their only difference lies in that the adopted gradient approximation in ConBP is continuous but not necessarily linear. An astute reader can quickly notice that LinBP can be seen as a special case of our ConBP, however, ConBP generally adopts a non-linear gradient approximation unless specified. If ConBP performs inferiorly against LinBP or does not improve the transferability at all, the improved transferability of LinBP should be, at least partially, attributed to the adopted gradient being *linear* instead of just being smooth, *i.e.* supporting the above mentioned linear nature hypothesis. Otherwise, it should be ascribed to the adopted alternative gradient being continuous, regardless of being linear or not.

Table 3. Success rates of transfer-based attacks on *ImageNet* using I-FGSM with ℓ_∞ constraint under the untargeted setting. Note that renormalization is applied. The source model is a ResNet-50 and the symbol * indicates that the victim model is the same as the source model. Average is obtained from models different from the source.

Dataset	Method	ϵ	ResNet* (2016)	Inception v3 (2016)	DenseNet (2017)	MobileNet v2 (2018)	PNASNet (2018)	SENet (2018)	Average
ImageNet	I-FGSM	0.1	**100.00%**	48.00%	73.00%	68.00%	43.00%	54.00%	57.2%
		0.05	**100.00%**	27.00%	54.00%	54.00%	25.00%	31.00%	38.2%
		0.03	**100.00%**	21.00%	47.00%	46.00%	21.00%	25.00%	32.00%
	LinBP+I-FGSM(†)	0.1	**100.00%**	88.00%	98.00%	97.00%	90.00%	94.00%	93.4%
		0.05	**100.00%**	60.00%	84.00%	89.00%	61.00%	73.00%	73.4%
		0.03	**100.00%**	49.00%	67.00%	73.00%	43.00%	50.00%	56.4%
	ConBP + I-FGSM	0.1	**100.00%**	91.00%	99.00%	99.00%	97.00%	98.00%	**96.8%**
		0.05	**100.00%**	69.00%	94.00%	89.00%	78.00%	84.00%	**82.4%**
		0.03	**100.00%**	49.00%	82.00%	77.00%	58.00%	66.00%	**66.4%**

Table 4. Success rates of *combined methods on ImageNet*. The source model is a ResNet-50 and the symbol * indicates that the victim model is the same as the source model. Average is obtained from models different from the source.

Dataset	Method	ϵ	ResNet* (2016)	Inception v3 (2016)	DenseNet (2017)	MobileNet v2 (2018)	PNASNet (2018)	SENet (2018)	Average
ImageNet	ILA+I-FGSM	0.1	**100.00%**	86.28%	97.14%	97.48%	89.62%	95.28%	93.16%
		0.05	**100.00%**	56.00%	80.00%	81.00%	60.00%	66.00%	68.6%
		0.03	**100.00%**	41.00%	68.00%	65.00%	41.00%	54.00%	53.8%
	LinBP+I-FGSM+ILA	0.1	**100.00%**	97.00%	100.00%	99.00%	97.00%	98.00%	98.2%
		0.05	**100.00%**	74.00%	94.00%	92.00%	72.00%	87.00%	83.8%
		0.03	**100.00%**	51.00%	73.00%	76.00%	51.00%	62.00%	62.6%
	ConBP+I-FGSM+ILA	0.1	**100.00%**	98.00%	100.00%	100.00%	97.00%	98.00%	**98.6%**
		0.05	**100.00%**	79.00%	92.00%	94.00%	80.00%	86.00%	**86.2%**
		0.03	**100.00%**	55.00%	83.00%	80.00%	55.00%	70.00%	**68.6%**

4. Experiments

Following prior arts on transferability, we adopt ImageNet as the dataset for comparing ConBP and LinBP. As suggested in [9], applying LinBP to ResNet requires additional re-normalization of gradients. To exclude the influence of re-normalization, we first compare their performance on a famous VGG architecture with batch normalization. Not losing generality, we adopt the continuous derivative of the widely known softplus function in the backward pass for alleviating the non-continuous derivative issue in the backward pass. Another variant of softplus function is parametric softplus that can control the shape of the resulting derivative with a parameter β. With an infinitely large β, its derivative approaches that of ReLU, and on the contrary an infinitely small derivative β leads to a constant, thus linear, derivative. Note that LinBP has no additional hyperparameter for making it flexible. Following normalization LinBP [9], we apply ConBP only to some ReLUs in the latter part of the DNN. With VGG19 as the source model, the results are shown in Table 1 with β in ConBP set to 0.5. We observe that applying LinBP to a few (up to 7) ReLUs in the network indeed improves the transferability, however, applying it to all ReLUs decreases the transferability. ConBP outperforms LinBP for most cases, especially when applied to more ReLUs. Our result suggests that the improved transferability stems from the continuous approximation of the gradient, regardless of being linear or not.

We further compare LinBP and ConBP in the ResNet. Here, we perform the comparison in two setups: (1) without re-normalization (see Table 2) and (2) with re-normalization (see Table 3). The results suggest that in both cases, our ConBP outperforms LinBP by a large margin. We further report the results combined with ILA [10] in Table 4. Our ConBP also outperforms LinBP in this setup.

5. Conclusion

The linear nature of DNNs is often hypothesized to explain the existence of adversarial examples and their transferability property. One recent work shows that LinBP improves transferability, constituting new evidence for supporting this rather controversial hypothesis. This work proposes ConBP based on which we discover that continuous derivative approximation for the gradient is sufficient enough for improving the transferability. Our extensive experiments show that regardless of being linear or not, approximating the ReLU derivative with a continuous derivative consistently improves the transferability and overall outperforms linear approximation. Thus, our finding clearly shows that the success of LinBP improving the transferability should not be attributed to the linearity hypothesis. As a byproduct, our investigation also leads to a new backpropagation method for improving transferability.

References

[1] W. Brendel, J. Rauber, and M. Bethge. Decision-based adversarial attacks: Reliable attacks against black-box machine learning models. *arXiv preprint arXiv:1712.04248*, 2017. 2

[2] N. Carlini and D. Wagner. Towards evaluating the robustness of neural networks. In *Symposium on Security and Privacy (SP)*, 2017. 2

[3] J. Chen, M. I. Jordan, and M. J. Wainwright. Hopskipjumpattack: A query-efficient decision-based attack. In *ieee symposium on security and privacy (sp)*, 2020. 2

[4] P.-Y. Chen, H. Zhang, Y. Sharma, J. Yi, and C.-J. Hsieh. Zoo: Zeroth order optimization based black-box attacks to deep neural networks without training substitute models. In *ACM workshop on artificial intelligence and security*, 2017. 2

[5] Y. Dong, F. Liao, T. Pang, H. Su, J. Zhu, X. Hu, and J. Li. Boosting adversarial attacks with momentum. In *CVPR*, 2018. 1, 2

[6] Y. Dong, T. Pang, H. Su, and J. Zhu. Evading defenses to transferable adversarial examples by translation-invariant attacks. In *CVPR*, 2019. 1, 2

[7] Y. Dong, H. Su, B. Wu, Z. Li, W. Liu, T. Zhang, and J. Zhu. Efficient decision-based black-box adversarial attacks on face recognition. In *CVPR*, 2019. 2

[8] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. In *ICLR*, 2015. 1, 2

[9] Y. Guo, Q. Li, and H. Chen. Backpropagating linearly improves transferability of adversarial examples. *arXiv preprint arXiv:2012.03528*, 2020. 1, 2, 4

[10] Q. Huang, I. Katsman, H. He, Z. Gu, S. Belongie, and S.-N. Lim. Enhancing adversarial example transferability with an intermediate level attack. In *ICCV*, 2019. 2, 4

[11] A. Ilyas, S. Santurkar, D. Tsipras, L. Engstrom, B. Tran, and A. Madry. Adversarial examples are not bugs, they are features. In *NeurIPS*, 2019. 1

[12] N. Inkawhich, W. Wen, H. H. Li, and Y. Chen. Feature space perturbations yield more transferable adversarial examples. In *CVPR*, 2019. 2

[13] D. Krotov and J. Hopfield. Dense associative memory is robust to adversarial inputs. *Neural computation*, 2018. 1, 2

[14] A. Kurakin, I. Goodfellow, and S. Bengio. Adversarial machine learning at scale. In *ICLR*, 2017. 2

[15] Y. Liu, X. Chen, C. Liu, and D. Song. Delving into transferable adversarial examples and black-box attacks. *ICLR*, 2017. 2

[16] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu. Towards deep learning models resistant to adversarial attacks. In *ICLR*, 2018. 2

[17] S.-M. Moosavi-Dezfooli, A. Fawzi, and P. Frossard. Deepfool: a simple and accurate method to fool deep neural networks. In *CVPR*, 2016. 2

[18] N. Narodytska and S. Kasiviswanathan. Simple black-box adversarial attacks on deep neural networks. In *CVPRW*, 2017. 2

[19] T. Pang, K. Xu, and J. Zhu. Mixup inference: Better exploiting mixup to defend adversarial attacks. *ICLR*, 2020. 2

[20] N. Papernot, P. McDaniel, I. Goodfellow, S. Jha, Z. B. Celik, and A. Swami. Practical black-box attacks against machine learning. In *ACM on Asia conference on computer and communications security*, 2017. 2

[21] M. Sotoudeh and A. V. Thakur. Computing linear restrictions of neural networks. *arXiv preprint arXiv:1908.06214*, 2019. 1, 2

[22] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013. 1

[23] P. Tabacof and E. Valle. Exploring the space of adversarial images. In *IJCNN*, 2016. 2

[24] T. Tanay and L. Griffin. A boundary tilting persepective on the phenomenon of adversarial examples. *arXiv preprint arXiv:1608.07690*, 2016. 1, 2

[25] D. Wu, Y. Wang, S.-T. Xia, J. Bailey, and X. Ma. Skip connections matter: On the transferability of adversarial examples generated with resnets. *arXiv preprint arXiv:2002.05990*, 2020. 2

[26] C. Xie, M. Tan, B. Gong, A. Yuille, and Q. V. Le. Smooth adversarial training. *arXiv preprint arXiv:2006.14536*, 2020. 2

[27] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. L. Yuille. Improving transferability of adversarial examples with input diversity. In *CVPR*, 2019. 1, 2

[28] Z. Yan, Y. Guo, and C. Zhang. Subspace attack: Exploiting promising subspaces for query-efficient black-box attacks. *NeurIPS*, 2019. 2

[29] W. Zhou, X. Hou, Y. Chen, M. Tang, X. Huang, X. Gan, and Y. Yang. Transferable adversarial perturbations. In *ECCV*, 2018. 2