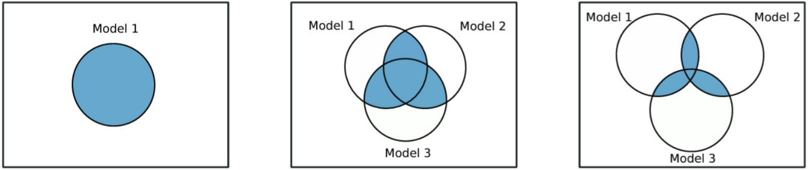
LOTOS: Layer-wise Orthogonalization for Training rObust enSembles

A. Ebrahimpour-Boroojeny, H. Sundaram, and V. Chandrasekaran

University of Illinois at Urbana-Champaign

Why ensembles?

- Ensembles of models is known to be more robust to adversarial attacks.
  - It is harder to fool multiple models!



(a) Single Model (b) Ensemble of 3 Models (c) Diverse Ensemble

- The adversarial examples are transferable!
  - this reduces the effectiveness of using multiple models
  - introducing diversity among the models helps to counteract this effect
  - this is the basis of methods for training robust ensembles

[1] Kariyappa, S., & Qureshi, M. K. (2019). Improving adversarial robustness of ensembles with diversity training. arXiv preprint arXiv:1901.09981.

Transferability of adversarial examples

- For model $\mathcal{F}: \mathcal{X} \to \mathcal{Y}$, untargeted attack $\mathcal{A}(x) = x + \delta_x$, and a sample (x, y), $\mathcal{A}$ maximizes $\ell_{\mathcal{F}}(x + \delta_x, y)$, s.t. $\|\delta_x\|_2 \leqslant \epsilon$.

Transferability of adversarial examples

- For model $\mathcal{F}:\mathcal{X}\to\mathcal{Y}$, untargeted attack $\mathcal{A}(x)=x+\delta_x$, and a sample (x,y), $\mathcal{A}$ maximizes $\ell_{\mathcal{F}}(x+\delta_x,y)$, s.t. $\|\delta_x\|_2\leqslant\epsilon$.
- Given an untargetted attack $\mathcal{A}(x)$ on a surrogate model $\mathcal{F}$, transferability between $\mathcal{F}$ and the target model $\mathcal{G}$ is defined as:
  - Consider the samples for which:
    1. $\mathcal{F}$ makes correct prediction,
    2. $\mathcal{G}$ makes correct prediction,
    3. $\mathcal{F}$ makes wrong prediction on $\mathcal{A}(x)$
  - And compute the ratio of them on which $\mathcal{G}$ makes wrong prediction on $\mathcal{A}(x)$ as well.

Transferability of adversarial examples

- For model $\mathcal{F}: \mathcal{X} \to \mathcal{Y}$, untargeted attack $\mathcal{A}(x) = x + \delta_x$, and a sample (x, y), $\mathcal{A}$ maximizes $\ell_{\mathcal{F}}(x + \delta_x, y)$, s.t. $\|\delta_x\|_2 \leqslant \epsilon$.
- Given an untargetted attack $\mathcal{A}(x)$ on a surrogate model $\mathcal{F}$, transferability between $\mathcal{F}$ and the target model $\mathcal{G}$ is defined as:
  - Consider the samples for which:
    1. $\mathcal{F}$ makes correct prediction,
    2. $\mathcal{G}$ makes correct prediction,
    3. $\mathcal{F}$ makes wrong prediction on $\mathcal{A}(x)$
  - And compute the ratio of them on which $\mathcal{G}$ makes wrong prediction on $\mathcal{A}(x)$ as well.
- More specifically:

$$T_{rate}(\mathcal{A}_{\mathcal{F}}, \mathcal{F}, \mathcal{G}) = \mathbb{P}_{(x,y)\in\mathcal{X}\times\mathcal{Y}}[\mathcal{G}(\mathcal{A}_{\mathcal{F}}(x)) \neq y \mid \mathcal{F}(x) = \mathcal{G}(x) = y \wedge \mathcal{F}(\mathcal{A}_{\mathcal{F}}(x)) \neq y],$$

Controling the Lipschitz constant

- Function f is L-Lipschitz if:

$$\|f(x_1) - f(x_2)\|_2 \leqslant L\|x_1 - x_2\|_2.$$

- Consider a model

$$f(x,\theta) = W^{L+1}\sigma_L(W^L(\sigma_{L-1}(W^{L-1}(\ldots\sigma_1(W^1x)\ldots)))),$$

  - Then:

$$\|f\|_{\text{Lip}} \leqslant \|(W^{L+1})\|_{\text{Lip}} \cdot \|\sigma_L\|_{\text{Lip}} \cdot \|(W^L)\|_{\text{Lip}} \cdots \|\sigma_1\|_{\text{Lip}} \cdot \|(W^1)\|_{\text{Lip}}$$

Effect of Lipschitz continuity

Controlling the Lipschitz factor of a model makes it more robust. [2]

$\boxed{\text{We investigated how it affects the transferability rate among models!}}$

[2] Ebrahimpour Boroojeny, A., Telgarsky, M., & Sundaram, H. (2024, April). Spectrum extraction and clipping for implicitly linear layers. In International Conference on Artificial Intelligence and Statistics (pp. 2971-2979). PMLR.

Effect of Lipschitz continuity

Controlling the Lipschitz factor of a model makes it more robust. [2]

We investigated how it affects the transferability rate among models!

Proposition

If $\ell_{\mathcal{F}}$ and $\ell_{\mathcal{G}}$ are L-Lipschitz w.r.t the inputs, and $\|\delta\|_2 \leqslant r$, for attack $\mathcal{A}$:

$$|R_{\mathcal{F}}(A(x), y) - R_{\mathcal{G}}(A(x), y)| \leqslant 2Lr + |R_{\mathcal{F}}(x, y) - R_{\mathcal{G}}(x, y)|.$$

[2] Ebrahimpour Boroojeny, A., Telgarsky, M., & Sundaram, H. (2024, April). Spectrum extraction and clipping for implicitly linear layers. In International Conference on Artificial Intelligence and Statistics (pp. 2971-2979). PMLR.

Effect of Lipschitz continuity

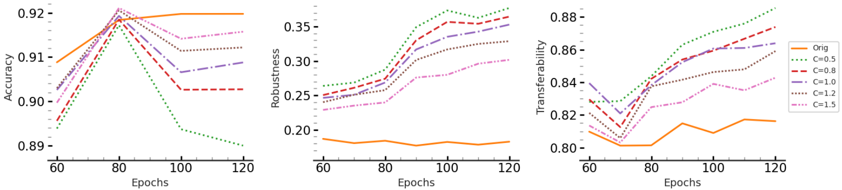
Controlling the Lipschitz factor of a model makes it more robust. [2]

We investigated how it affects the transferability rate among models!

Proposition

If $\ell_{\mathcal{F}}$ and $\ell_{\mathcal{G}}$ are L-Lipschitz w.r.t the inputs, and $\|\delta\|_2 \leqslant r$, for attack $\mathcal{A}$:

$$|R_{\mathcal{F}}(A(x), y) - R_{\mathcal{G}}(A(x), y)| \leqslant 2Lr + |R_{\mathcal{F}}(x, y) - R_{\mathcal{G}}(x, y)|.$$



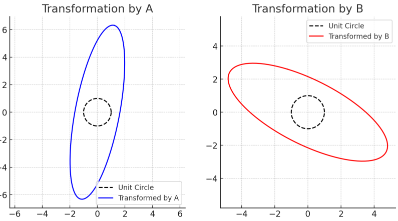
[2] Ebrahimpour Boroojeny, A., Telgarsky, M., & Sundaram, H. (2024, April). Spectrum extraction and clipping for implicitly linear layers. In International Conference on Artificial Intelligence and Statistics (pp. 2971-2979). PMLR.

What can we do?

- How to preserve the robustness of individual models *and* increase the diversity?
- We consider the layers-wise transformations of linear layers!

What can we do?

- How to preserve the robustness of individual models *and* increase the diversity?
- We consider the layers-wise transformations of linear layers!
  - Consider two corresponding layers from the two models in the ensemble:
  - The direction along the largest singular vector of each one should cause small changes in the other!



New component in the loss function

- We define a new notion of similarity between the jth layers of models f and g, based on their top-k sub-spaces:

$$S_k^{(j)}(f^{(j)}, g^{(j)}, \texttt{mal}) := \sum_{i=1}^{k} w_i(\mathrm{ReLU}(\|f^{(j)}(v_i')\|_2 - \texttt{mal}) + \mathrm{ReLU}(\|g^{(j)}(v_i)\|_2 - \texttt{mal})),$$

  - where `mal` is a threshold hyperparameter

New component in the loss function

- We define a new notion of similarity between the jth layers of models f and g, based on their top-k sub-spaces:

$$S_k^{(j)}(f^{(j)}, g^{(j)}, \mathtt{mal}) := \sum_{i=1}^{k} w_i(\mathrm{ReLU}(\|f^{(j)}(v_i')\|_2 - \mathtt{mal}) + \mathrm{ReLU}(\|g^{(j)}(v_i)\|_2 - \mathtt{mal})),$$

  - where `mal` is a threshold hyperparameter

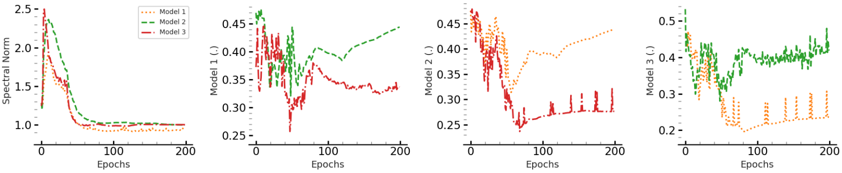
- LOTOS uses a new component in the loss that penalizes the similarity for all pairs of corresponding layers for the models of the ensemble:

$$\mathcal{L}_{\textsf{train}} = \frac{1}{N}\sum_{i=1}^{N}\mathcal{L}_{\textsf{CE}}(\mathcal{F}_i(x), y) + \frac{\lambda}{M\,N\,(N-1)}\sum_{z=1}^{N-1}\sum_{j=z+1}^{N}\sum_{l=1}^{M} S_k^{(l)}(f_z^{(l)}, f_j^{(l)}, \mathtt{mal})$$

Does the new loss function work?

The effect of orthogonalization with different values of `mal`:

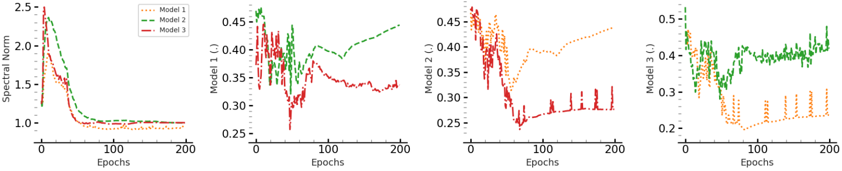
- When `mal` $= 0.5$:



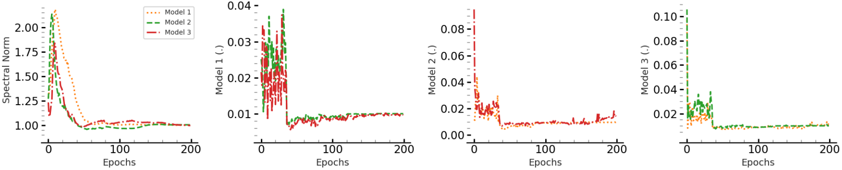
Does the new loss function work?

The effect of orthogonalization with different values of `mal`:

- When `mal` $= 0.5$:



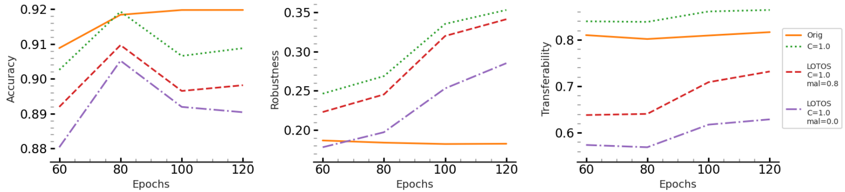
- When `mal` $= 0.01$:



Does this help?

Experimental setup: ensembles of three ResNet-18 models

- LOTOS effectively *decreases the transferability rate* among pairs of models
- LOTOS still benefits from *robustness of individual models*
- By decreasing *the value of* `mal` transferability rate decreases, but individual accuracy and robust accuracy decreases, and therefore, *introduces a trade-off*



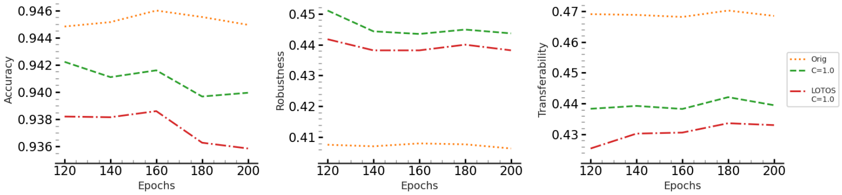
More robust ensembles!

- Comparing the effectiveness in overall robustness of the ensembles:
  - Two different architectures (ResNet-18 and DLA)
  - Two different datasets (CIFAR-10 and CIFAR-100)

	CIFAR-10			CIFAR-100		
	Orig	$C = 1.0$	LOTOS	Orig	$C = 1.0$	LOTOS
	Ensembles of ResNet-18 models					
Test Acc	**95.3 ± 0.06**	94.7 ± 0.24	94.6 ± 0.19	**77.2 ± 0.17**	76.6 ± 0.01	76.6 ± 0.10
Robust Acc	30.3 ± 1.63	35.2 ± 0.72	**36.3 ± 0.88**	15.2 ± 0.45	18.9 ± 0.40	**20.2 ± 0.47**
	Ensembles of DLA models					
Test Acc	**95.4 ± 0.12**	95.2 ± 0.05	95.05 ± 0.09	77.1 ± 0.09	**78.8 ± 0.31**	78.3 ± 0.38
Robust Acc	26.7 ± 0.58	32.8 ± 1.28	**34.5 ± 0.63**	16.5 ± 0.78	19.4 ± 0.32	**21.0 ± 0.39**

What about heterogeneous ensembles?

- The singular vectors of the first affine layer have the same dimension, which is equal to the input dimension!
  - They can be made orthogonal using LOTOS
- Ensembles of ResNet-18, ResNet-34, and DLA trained on CIFAR-10:
  - LOTOS still effective, but not as much as homogeneous ensembles



Works with other methods!

- Comparing the effectiveness of LOTOS when used along with TRS[3], which is one of the prior SOTA in ensemble robust training
  - Two different architectures (ResNet-18 and DLA)
  - Two different datasets (CIFAR-10 and CIFAR-100)

	CIFAR-10			**CIFAR-100**		
	TRS	TRS + $C = 1$	TRS + LOTOS	TRS	TRS + $C = 1$	TRS + LOTOS
			ENSEMBLES OF RESNET-18 MODELS			
TEST ACC	**94.4** ± **0.05**	94.1 ± 0.17	92.7 ± 0.09	**73.28** ± **0.46**	72.94 ± 0.29	67.23 ± 1.22
ROBUST ACC	30.8 ± 0.65	35.9 ± 1.35	**41.5** ± **1.04**	12.3 ± 0.53	16.3 ± 0.57	**20.7** ± **0.99**
			ENSEMBLES OF DLA MODELS			
TEST ACC	**94.72** ± **0.06**	92.79 ± 0.13	93.18 ± 0.14	**72.6** ± **0.54**	63.3 ± 1.20	66.8 ± 1.26
ROBUST ACC	31.2 ± 0.80	32.9 ± 0.77	**35.3** ± **0.39**	23.2 ± 0.41	23.7 ± 2.36	**24.3** ± **1.67**

- Similar results with DVERGE[4], another prior SOTA

[3] Yang, Z., Li, L., Xu, X., Zuo, S., Chen, Q., Zhou, P., ... Li, B. (2021). Trs: Transferability reduced ensemble via promoting gradient diversity and model smoothness. Advances in Neural Information Processing Systems, 34, 17642-17655.

[4] Yang, H., Zhang, J., Dong, H., Inkawhich, N., Gardner, A., Touchet, A., ... Li, H. (2020). Dverge: diversifying vulnerabilities for enhanced robust generation of ensembles. Advances in Neural Information Processing Systems, 33, 5505-5515.

Is it efficient?

- LOTOS promotes orthogonality among the top-k sub-spaces.
- The $(k+1)$st singular vectors can still be highly correlated!
- The $(k+1)$st singular value can be as large as the 1st one!

Theorem

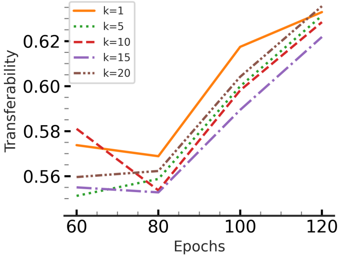
Given two convolutional layers, M_1 and M_2 with a single input and output channel and circular padding for which **f** *is the vectorized form of the filter with a length of T, and considering n to be the length of the vectorized input, if $\|Av'_1\|_2 \leqslant \epsilon$, then:*

$$\|Av'_p\|_2 \leqslant \sqrt{\epsilon^2 + \pi \|\mathbf{f}\|_2^2 T^2 \frac{p}{n}},$$

where A is the corresponding linear transformation of M_1 and v'_p is singular vector of M_2 corresponding to its p-th largest singular value.

Is it efficient?

- Observing the effect of k in transferability rate:



- Increasing k slightly decreases the transferability rate, but up to $k = 15$
- k larger than 15 does not show further improvements due to further optimization constraints and restrictions of convolutional layer's spectrum

Thank You!

Our Paper

Other results

- Observing the effect of k in transferability rate:

	Orig	Orig+ $C = 1$	Orig + LOTOS
TRS	10.3 ± 0.33	13.2 ± 1.12	$\mathbf{14.5} \pm \mathbf{0.81}$
DVERGE	19.7 ± 2.34	26.8 ± 0.75	$\mathbf{29.2} \pm \mathbf{0.56}$

- Comparing the running times:

	Orig	$C = 1$	LOTOS	TRS	TRS + $C = 1$	TRS + LOTOS	Adv	Adv + $C = 1$	Adv + LOTOS
ResNet-18	33.2	74.9 $\times 2.3$	79.3 $\times 2.4$	158.2	224.4 $\times 1.4$	227.3 $\times 1.4$	312.6	479.2 $\times 1.5$	485.2 $\times 1.5$
DLA	63.1	155.4 $\times 2.5$	165.4 $\times 2.6$	326.2	466.1 $\times 1.4$	477.4 $\times 1.5$	758.6.2	942.5 $\times 1.2$	949.2 $\times 1.2$

Transferability of adversarial examples

- For model $\mathcal{F} : \mathcal{X} \to \mathcal{Y}$ and, attack $\mathcal{A}(x) = x + \delta_x$, and a sample (x, y):
  - for untargetted attack: $\mathcal{A}$ maximizes $\ell_{\mathcal{F}}(x + \delta_x, y)$, s.t. $\|\delta_x\|_2 \leqslant \epsilon$.
  - and targeted attack for y_t: $\mathcal{A}_t(x) = \min_{\delta\ x}[\ell_{\mathcal{F}}(x + \delta_x . y_t)]$
- Given an untargetted attack $\mathcal{A}(x)$ on a surrogate model $\mathcal{F}$, transferability between $\mathcal{F}$ and the target model $\mathcal{G}$ is defined as:

$$T_{rate}(\mathcal{A}_{\mathcal{F}}, \mathcal{F}, \mathcal{G}) = \mathbb{P}_{(x,y)\in\mathcal{X}\times\mathcal{Y}}[\mathcal{G}(\mathcal{A}_{\mathcal{F}}(x)) \neq y \mid \mathcal{F}(x) = \mathcal{G}(x) = y \\ \wedge \mathcal{F}(\mathcal{A}_{\mathcal{F}}(x)) \neq y],$$

Transferability of adversarial examples

- For model $\mathcal{F}: \mathcal{X} \to \mathcal{Y}$ and, attack $\mathcal{A}(x) = x + \delta_x$, and a sample (x, y):
  - for untargetted attack: $\mathcal{A}$ maximizes $\ell_{\mathcal{F}}(x + \delta_x, y)$, s.t. $\|\delta_x\|_2 \leqslant \epsilon$.
  - and targeted attack for y_t: $\mathcal{A}_t(x) = \min_{\delta\ x}[\ell_{\mathcal{F}}(x + \delta_x.y_t)]$
- Given an untargetted attack $\mathcal{A}(x)$ on a surrogate model $\mathcal{F}$, transferability between $\mathcal{F}$ and the target model $\mathcal{G}$ is defined as:

$$T_{rate}(\mathcal{A}_{\mathcal{F}}, \mathcal{F}, \mathcal{G}) = \mathbb{P}_{(x,y)\in\mathcal{X}\times\mathcal{Y}}[\mathcal{G}(\mathcal{A}_{\mathcal{F}}(x)) \neq y \mid \mathcal{F}(x) = \mathcal{G}(x) = y \\ \wedge \mathcal{F}(\mathcal{A}_{\mathcal{F}}(x)) \neq y],$$

  - And for targeted attack $\mathcal{A}_t(x)$, it is defined as:

$$T_{rate}(\mathcal{A}_{\mathcal{F}}^{(t)}, \mathcal{F}, \mathcal{G}) = \mathbb{P}_{(x,y)\in\mathcal{X}\times\mathcal{Y}}[\mathcal{G}(\mathcal{A}_{\mathcal{F}}^{(t)}(x)) = y_t \mid \mathcal{F}(x) = \mathcal{G}(x) = y \\ \wedge \mathcal{F}(\mathcal{A}_{\mathcal{F}}^{(t)}(x)) = y_t].$$