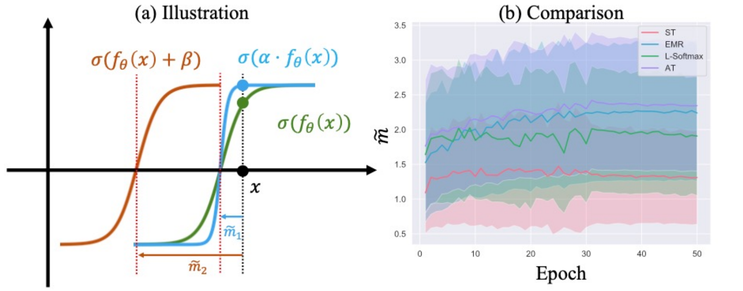
Boosting Adversarial Robustness From The Perspective of Effective Margin Regularization

Ziquan Liu and Antoni B. Chan

CityU 香港城市大學 City University of Hong Kong 專業 創新 胸懷全球 Professional·Creative For The World

BMVC 2022

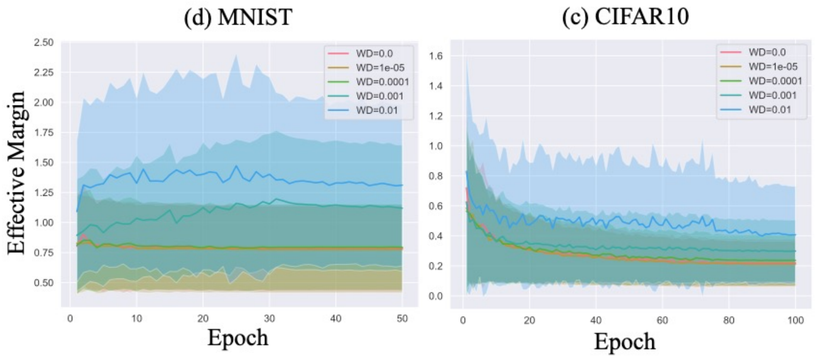
Motivation



$$XE_i = -\sum_{k=1}^{K} y_{ik} \log \frac{\exp(l_{ik})}{\sum_j \exp(l_{ij})} \qquad \tilde{m}_i = \min_{j \neq y_i} \frac{f_{\boldsymbol{\theta}}^{(y_i)}(\boldsymbol{x}_i) - f_{\boldsymbol{\theta}}^{(j)}(\boldsymbol{x}_i)}{\|\boldsymbol{w}_i^{(y_i)} - \boldsymbol{w}_i^{(j)}\|_2}$$

(a) The scale-variant property of cross-entropy loss. The loss can be reduced while keeping the actual margin the same.
(b) The adversarial attack is not variant to the scaling.
(c) Only training with cross-entropy loss does not effectively increase the actual margin and improves adversarial robustness

Can weight decay come to rescue?



Training	λ_{WD}	Clean Acc.	PGD20	$\tilde{m}_{\text{train}}$	$\tilde{m}_{\text{test}}$
ST	0.1	77.88	34.06	1.00±0.69	1.02±0.70
ST	0.01	96.54	48.41	1.31±0.69	1.31±0.67
ST	0.001	98.41	24.41	1.11±0.51	1.12±0.53
ST	0.0001	98.33	4.69	0.79±0.34	0.79±0.36
ST+LSoftmax	0.1	17.83	17.27	3.13±1.11	3.21±1.16
ST+LSoftmax	0.01	98.00	60.59	1.91±0.86	1.91±0.85
ST+LSoftmax	0.001	**98.63**	51.07	1.82±0.63	1.81±0.66
ST+LSoftmax	0.0001	98.50	28.91	1.34±0.44	1.34±0.47
ST+EMR$_{0.1}$	0.001	97.50	**87.56**	4.41±1.24	2.24±0.98
AT	0.01	97.66	90.11	1.18±0.69	2.27±1.07
AT	0.001	98.68	92.62	1.21±0.69	2.34±0.94
AT	0.0001	**98.98**	92.24	1.09±0.61	2.13±0.77
AT+LSoftmax	0.01	97.57	89.88	1.18±0.69	2.28±1.07
AT+LSoftmax	0.001	98.72	92.60	1.21±0.68	2.35±0.94
AT+LSoftmax	0.0001	99.02	92.50	1.10±0.63	2.12±0.78
AT+EMR$_{3e-4}$	0.001	98.68	**92.78**	3.83±1.27	2.42±0.99

Training	λ_{WD}	Clean Acc	PGD10	$\tilde{m}_{\text{train}}$	$\tilde{m}_{\text{test}}$
ST	0.01	55.77	4.19	0.39±0.31	0.41±0.32
ST	0.001	82.10	0.68	0.28±0.19	0.30±0.21
ST	0.0001	**83.00**	0.50	0.22±0.15	0.23±0.17
ST+LSoftmax	0.01	55.06	4.90	0.41±0.33	0.43±0.34
ST+LSoftmax	0.001	82.39	0.81	0.27±0.18	0.28±0.19
ST+LSoftmax	0.0001	82.73	0.29	0.22±0.15	0.23±0.16
ST+EMR$_{0.01}$	0.0005	69.78	**16.37**	0.66±0.48	0.70±0.50
AT	0.01	34.73	22.98	0.90±0.82	1.22±1.04
AT	0.001	60.27	32.47	0.76±0.65	1.09±0.84
AT	0.0001	**63.59**	32.58	0.71±0.60	1.03±0.78
AT+LSoftmax	0.01	34.62	22.87	0.91±0.82	1.21±1.02
AT+LSoftmax	0.001	60.47	32.29	0.76±0.65	1.08±0.84
AT+LSoftmax	0.0001	63.09	32.96	0.71±0.61	1.03±0.78
AT+EMR$_{0.001}$	0.0005	62.79	**33.41**	0.74±0.64	1.08±0.83

In both standard and adversarial training, increasing weight decay does not always lead to improved adversarial robustness. There are two major problems with weight decay:

- Uniform regularization for all parameters
- Does not consider local structures

Proposed Method

We propose to use the effective margin regularization (EMR) to control the effective margin during training. Difference from weight decay, the EMR regularization is dependent on input and controls the gradient norm of each local neural network.

$$l_{ij} = \boldsymbol{x}_i^T \boldsymbol{w}_i^{(j)} = \|\boldsymbol{x}_i\| \|\boldsymbol{w}_i^{(j)}\| \cos(\phi_{ij})$$

$$\mathcal{L} = \frac{1}{B} \sum_i^B XE(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i) + \lambda_{EMR} \frac{1}{B} \sum_i^B \sum_j^K \|\boldsymbol{w}_i^{(j)}\|^2$$

Computation: For a neural network without batch normalization, the samples in a batch are independent. Thus, we can take the gradient of logits $\sum_i l_{ij}$ with respect to input tensor and get the gradient for jth class.

The previous table shows the effectiveness of EMR.

Approximation of EMR in large-scale models (with batch norm)

Define $\boldsymbol{l}_i$ as the logit vector for the ith sample, and $h_i = \sum_j p_{ij} l_{ij}(\boldsymbol{x}_i)$ as the weighted logit mean, where $\sum_j p_{ij} = 1$ is a constant weight vector in a (K − 1)-dim simplex. The gradient of h_i with respect to $\boldsymbol{x}_i$ is $\nabla_x h_i = \sum_j p_{ij} w_i^{(j)}$ and its squared l2 norm is

$$\hat{\mathcal{L}}_{EMR}(\boldsymbol{x}_i) = \|\nabla_{\boldsymbol{x}} \sum_j p_{ij} l_{ij}(\boldsymbol{x}_i)\|_2^2 = \sum_j \sum_k p_{ij} p_{ik} \langle \boldsymbol{w}_i^{(j)}, \boldsymbol{w}_i^{(k)} \rangle$$

We set the p as the output of a softmax function with input logit l divided by a temperature parameter, which controls the penalty on logits.

Algorithm 1: Adv. Training with Effective Margin Regularization

Input: Training data $\mathcal{D}_{tr}$, EMR parameter λ_{EMR}, EMR temperature t, learning rate η, beta of TRADES β
Output: Model parameters $\boldsymbol{\theta}$
Initialize model parameters;
for $i = 1, \ldots, N_e$ **do**
  Adjust η and λ_{EMR};
  Split $\mathcal{D}_{tr}$ into N_B =ceil(N_{tr}/B) batches;
  for $b = 1, \ldots, N_B$ **do**
    Generate adversarial examples $\{\tilde{\boldsymbol{x}}_i, y_i\}_{i=1}^B$;
    if *AT* **then**
      $\mathcal{L} = \frac{1}{B} \sum_{i=1}^B XE(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_i), y_i)$;
    end
    if *TRADES* **then**
      $\mathcal{L} = \frac{1}{B} \sum_i XE(f_{\boldsymbol{\theta}}(\boldsymbol{x}_i), y_i) + \beta \frac{1}{B} \sum_i \mathcal{D}_{KL}(f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_i), f_{\boldsymbol{\theta}}(\boldsymbol{x}_i))$;
    end
    % EMR:
    $\boldsymbol{p}_i$=softmax($f_{\boldsymbol{\theta}}(\tilde{\boldsymbol{x}}_i)/t$) and detach the gradient;
    $\mathcal{L}_{EMR} = \frac{1}{B} \sum_i^B \|\nabla_{\boldsymbol{x}} \sum_{j=1}^K p_{ij} l_{ij}(\tilde{\boldsymbol{x}}_i)\|_2^2$ **(eval mode)**;
    $\boldsymbol{\theta} := \boldsymbol{\theta} - \eta \nabla_{\boldsymbol{\theta}} (\mathcal{L} + \lambda_{EMR} \mathcal{L}_{EMR})$
  end
end

Main Algorithm

Experiment Result

We use WideResNet-34-10 as the backbone for the experiment and test the performance on CIFAR10. The adversarial attack is l_∞ bounded with an epsilon of 8/255.

	Clean Acc.	FGSM	PGD10	PGD100	AutoAttack
AT	87.35	59.97	53.55	52.30	50.31
AT+IGR[41]	**87.49**	60.32	53.49	52.42	50.42
AT+HE[37]	84.53	64.07	60.36	59.80	51.88
AT+EMR (ours)	85.74	60.67	55.43	54.62	**52.20**
TRADES	82.95	60.65	56.71	56.17	52.19
TRADES+IGR[41]	**84.18**	61.16	56.67	55.90	52.41
TRADES+HE[37]	79.61	60.95	58.23	57.99	51.49
TRADES+EMR (ours)	83.03	60.89	57.27	56.89	**52.73**
AT+MAIL [26]	86.96	60.90	55.42	54.53	45.07
AT+MAIL+EMR (ours)	**87.33**	61.32	56.77	56.00	**46.25**
TRADES+MAIL [26]	84.82	60.44	55.35	54.69	51.97
TRADES+MAIL+EMR (ours)	**85.37**	61.67	56.82	56.21	**53.29**
MART [48]	**83.62**	61.83	57.32	56.43	51.40
MART+EMR (ours)	83.55	62.87	58.09	57.43	**52.16**

Table 3. Evaluation of adversarial robustness using WideResNet-34-10 on CIFAR10. The best result under the strongest attack is emphasized with bold text.