Conventional Adversarial Training

ICLR'19 paper review

202111278 컴퓨터공학부 김환희

목차

- Overfitting
- Free
- SLAT (Single-step Latent Adversarial Training)
- GradAlign

Conventional Adversarial Training

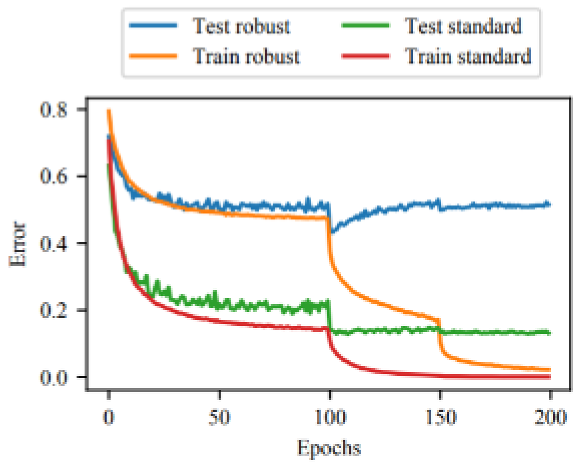
Table 2: Taxonomy of adversarial training covered in this paper.

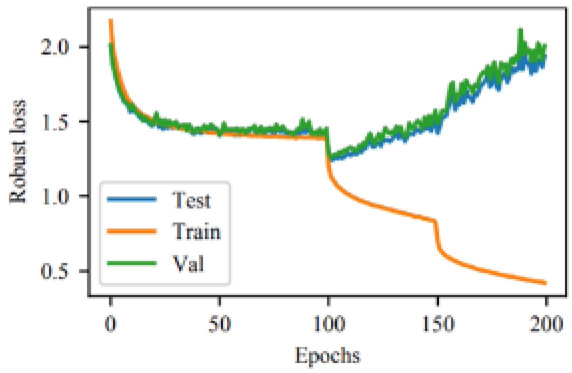
	Defence	Remark
Conventional AT	FGSM-AT [6]	Train a model with FSGM adversarial examples
	PGD-AT [37]	Train a model with PGD examples, import baseline
	TLA [68]	Train a model with the triplet loss
	ANL [22]	Inject adversarial perturbation into latent features
	BAT [23]	Perturb both the image and the label during training
	EAT [73]	Train a model with adversarial examples generated on other pre-trained models
	PED [74]	Force non-maximal predictions as diverse as possible in an ensemble system
	ALP [21]	Force adversarial examples and their corresponding natural samples to have similar output
	FAT [76]	Train a model with friendly adversarial examples which do not cross the decision boundary too much
	Overfitting [81]	leverage early stop to choose the best checkpoint to inference
	Free [69]	Accelerate AT by recycling the gradient information
	SLAT [70]	Accelerate AT with the single-step latent adversarial training (SLAT)
	GradAlign [71]	Accelerate AT with GradAlign that aims to maximize the gradient alignment between x and x'
	Fast AT [72]	Accelerate AT with R+FGSM [46], Cyclic learning rate [102], and Mixed-precision arithmetic [103]

Conventional Adversarial Training

Overfitting

- L. Rice et al., (2020), Overfitting in adversarially robust deep learning, ICML
- Leverage early stop to choose the best checkpoint to inference
- Robust overfitting: epoch가 늘어남에 따라 test adversarial accuracy가 낮아진다.





Reg Method	Robust Test Error (%) Final	Best	Diff
Early stopping w/ val	**46.9**	46.7	0.2
ℓ_1 regularization	53.0 ± 0.39	48.6	4.4
ℓ_2 regularization	55.2 ± 0.4	46.4	55.2
Cutout	48.8 ± 0.79	46.7	2.1
Mixup	49.1 ± 1.32	46.3	2.8
Semi-supervised	47.1 ± 4.32	40.2	6.9

Conventional Adversarial Training

Free

- A. Shafahi et al., (2019), Adversarial training for free!, NeurIPS
- Accelerate AT by recycling the gradient information
- Adversarial training은 일반 training에 비해 3에서 30배까지 시간이 소요된다. 원인은 adversarial example을 만들기 위한 gradient 계산이다.
- Natural training을 웃도는 빠른 adversarial training method를 소개한다.

Table 1: Validation accuracy and robustness of CIFAR-10 models trained with various methods.

Training	Evaluated Against					Train Time (min)
	Nat. Images	PGD-20	PGD-100	CW-100	10 restart PGD-20	
Natural	**95.01%**	0.00%	0.00%	0.00%	0.00%	**780**
Free $m=2$	91.45%	33.92%	33.20%	34.57%	33.41%	816
Free $m=4$	87.83%	41.15%	40.35%	41.96%	40.73%	800
Free $m=8$	85.96%	**46.82%**	**46.19%**	**46.60%**	**46.33%**	785
Free $m=10$	83.94%	46.31%	45.79%	45.86%	45.94%	785
7-PGD trained	87.25%	45.84%	45.29%	46.52%	45.53%	5418

Table 2: Validation accuracy and robustness of CIFAR-100 models trained with various methods.

Training	Evaluated Against			Training Time (minutes)
	Natural Images	PGD-20	PGD-100	
Natural	**78.84%**	0.00%	0.00%	811
Free $m=2$	69.20%	15.37%	14.86%	816
Free $m=4$	65.28%	20.64%	20.15%	**767**
Free $m=6$	64.87%	23.68%	23.18%	791
Free $m=8$	62.13%	**25.88%**	**25.58%**	780
Free $m=10$	59.27%	25.15%	24.88%	776
Madry *et al.* (2-PGD trained)	67.94%	17.08%	16.50%	2053
Madry *et al.* (7-PGD trained)	59.87%	22.76%	22.52%	5157

Algorithm 1 "Free" Adversarial Training (Free-m)

Require: Training samples X, perturbation bound ϵ, learning rate τ, hop steps m

```
Initialize θ
δ ← 0
for epoch = 1 ... N_ep/m do
    for minibatch B ⊂ X do
        for i = 1 ... m do
            Update θ with stochastic gradient descent
                g_θ ← E_{(x,y)∈B}[∇_θ l(x + δ, y, θ)]
                g_adv ← ∇_x l(x + δ, y, θ)]
                θ ← θ − τ g_θ
            Use gradients calculated for the minimization step to update δ
                δ ← δ + ε · sign(g_adv)
                δ ← clip(δ, −ε, ε)
        end for
    end for
end for
```

Conventional Adversarial Training

SLAT (Single-step Latent Adversarial Training)

- G. Y. Park et al., (2021), Reliably fast adversarial training via latent adversarial perturbation, ICCV
- Accelerate AT with the single-step latent adversarial training
- Approximated latent space perturbation을 이용해 빠르게 model을 training한다.

	Method	Standard	PGD-50-10	AutoAttack	Training time (min)
CIFAR-10	PGD-7	84.86±0.16	51.63±0.13	48.65±0.08	383.2
	FGSM-GA	82.88±0.01	48.90±0.37	46.22±0.30	297.9
	YOPO-5-3	82.35±1.78	34.23±3.61	32.79±3.65	62.5
	Free-AT ($m = 8$)	76.57±0.19	44.15±0.30	41.02±0.20	119.4
	FGSM	87.42±1.08	0.01±0.01	0.00±0.00	100.5
	FGSM-RS	90.76±6.36	3.90±4.06	0.44±0.50	99.7
	FGSM-CKPT ($c = 3$)	89.32±0.10	40.83±0.36	39.38±0.24	121.4
	SLAT	85.91±0.31	47.06±0.03	44.62±0.11	104.6
CIFAR-100	PGD-7	59.59±0.17	29.58±0.24	26.00±0.20	392.1
	FGSM-GA	58.63±0.17	27.53±0.10	24.07±0.15	240.5
	YOPO-5-3	51.45±7.33	15.23±2.01	13.94±1.82	65.0
	Free-AT ($m = 8$)	48.02±0.29	22.40±0.19	18.67±0.03	117.1
	FGSM	61.96±2.17	0.00±0.00	0.00±0.00	99.9
	FGSM-RS	50.96±4.57	0.00±0.00	0.00±0.00	100.9
	FGSM-CKPT ($c = 3$)	73.53±0.65	0.66±0.60	0.09±0.09	101.5
	SLAT	59.56±0.50	26.26±0.47	23.02±0.14	101.7
Tiny ImageNet	PGD-7	48.92±0.43	23.05±0.35	18.78±0.14	3098.3
	FGSM-GA	48.73±0.14	22.62±0.11	18.34±0.07	2032.2
	YOPO-5-3	51.45±6.01	15.08±1.78	13.94±1.61	511.5
	Free-AT ($m = 8$)	22.40±0.17	9.05±0.08	6.06±0.18	911.6
	FGSM	36.47±11.75	8.68±12.27	6.63±9.38	779.8
	FGSM-RS	42.13±14.98	10.32±11.93	8.41±9.73	787.5
	FGSM-CKPT ($c = 3$)	61.64±2.24	5.91±6.68	5.26±5.98	753.0
	SLAT	48.77±0.25	20.21±0.16	16.38±0.16	785.5

Algorithm 1: Single-step Latent Adversarial Training method (**SLAT**)

Input: Training iteration T, Number of samples N, Number of layers L, Training set $\mathcal{D} = \{(x_i, y_i)\}_{i=1}^{N}$, Subset of layer indexes K, Layer-wise step size η_k
Output: Adversarially robust network $f_{\boldsymbol{\theta}}$

for $t \leftarrow 1$ ***to*** T **do**
  for $i \leftarrow 1$ ***to*** N **do**
    for $k \in K$ **do**
      // Compute latent adversarial perturbations
      $\delta_k(x_i) = \eta_k \cdot sign\left(\nabla_{h_k(x_i)} \mathcal{L}(f_{\boldsymbol{\theta}}(x_i), y_i)\right)$
    for $l \in \{0, \ldots, L-2\}$ **do**
      if $l \in K$ **then**
        // Propagate adversarial perturbations forward
        $h_{l+1}(x_i) = f_{l+1}(h_l(x_i) + \delta_l(x_i))$
      else
        $h_{l+1}(x_i) = f_{l+1}(h_l(x_i))$
    Optimize $\boldsymbol{\theta}$ by the objective $\mathcal{L}(f_L(h_{L-1}(x_i)), y_i)$ using gradient descent.

Conventional Adversarial Training

GradAlign

- M. Andriushchenko et al., (2020), Understanding and improving fast adversarial training, NeurIPS
- Accelerate AT with GradAlign that aims to maximize the gradient alignment between x and x'
- FastAT을 포함한 AT들엔 catastrophic overfitting 문제가 있다.
- Gradient alignment를 최대화해 catastrophic overfitting을 막는 새로운 정규화 방법을 제안한다.

- Gradient alignment: $\mathbb{E}_{(x,y)\sim D,\ \eta\sim\mathcal{U}([-\varepsilon,\varepsilon]^d)}\left[\cos\left(\nabla_x \ell(x,y;\theta), \nabla_x \ell(x+\eta,y;\theta)\right)\right],$

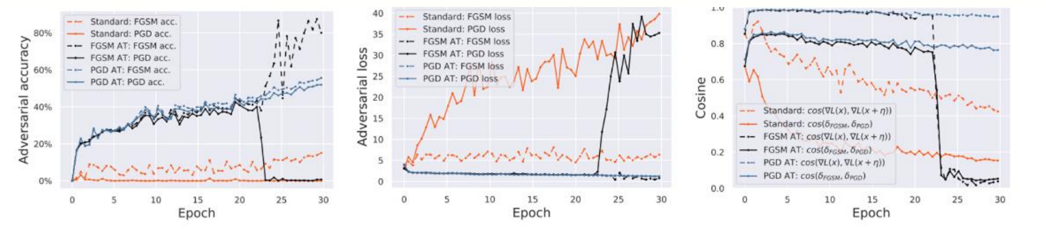


Figure 4: Visualization of the training process of standardly trained, FGSM trained, and PGD-10 trained ResNet-18 on CIFAR-10 with $\varepsilon = 8/255$. All the statistics are calculated on the test set. Catastrophic overfitting for the FGSM AT model occurs around epoch 23 and is characterized by a sudden drop in the PGD accuracy, a gap between the FGSM and PGD losses, and a dramatic decrease of *local linearity*.

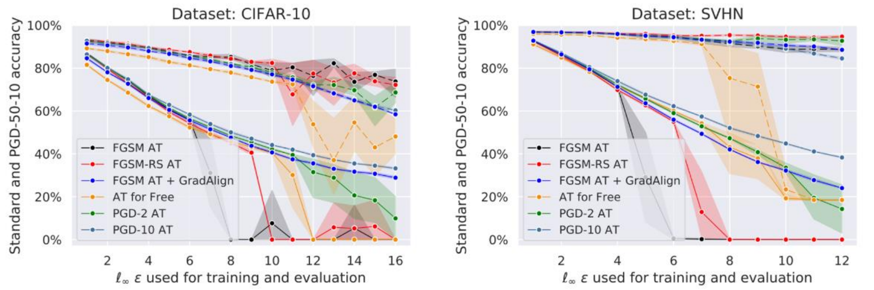


Figure 8: Accuracy (dashed line) and robustness (solid line) of different adversarial training (AT) methods on CIFAR-10 and SVHN with ResNet-18 trained and evaluated with different l_∞-radii. The results are obtained without early stopping, averaged over 5 random seeds used for training and reported with the standard deviation.