Deep Learning

Exercise 11: Adversarial Training

Instructor: Manuel Günther
Email: guenther@ifi.uzh.ch
Office: AND 2.54

Friday, May 21, 2021

Outline

Outline

Adversarial Examples via FGS/FGV

Gradient Calculation

- Loss function categorical $\mathcal{J}^{\mathrm{CE}}$
- Gradient w.r.t. $\mathcal{X}$: $\nabla_{\mathcal{X}} = \frac{\partial \mathcal{J}^{\mathrm{CE}}}{\partial \mathcal{X}}$
  - → Enable gradient for input: `X.requires_grad_(True)`
  - → Compute loss: `J = loss(X,t)`
  - → Compute gradient: `J.backward()`
  - → Access gradient: `X.grad`

Fast Gradient Sign

- Adversarial input:
$$\check{\mathcal{X}}_{\mathrm{FGS}} = \mathcal{X} + \alpha \operatorname{sign}\left(\nabla_{\mathcal{X}}\right)$$
- Clip to pixel range $[0, 1]$

Fast Gradient Value

- Adversarial input:
$$\check{\mathcal{X}}_{\mathrm{FGV}} = \mathcal{X} + \alpha \frac{\nabla_{\mathcal{X}}}{\max\left|\nabla_{\mathcal{X}}\right|}$$
- Clip to pixel range $[0, 1]$

Adversarial Examples via FGS/FGV

Task 1: Network

1. Small convolutional network
   → Copy from last exercises
2. With or without BatchNorm

Task 2: FGS

1. Implement `FGS(X,T,alpha)`
2. One-step with fixed $\alpha = 0.3$

Task 2(b, optional): FGV

1. Implement `FGV(X,T,alpha)`
2. One-step with fixed $\alpha = 0.6$

Task 3: Evaluation

1. Accuracy on original test samples
2. Accuracy on adversarial test samples
   → Only correctly classified originals
   → Compare to original targets

Outline

Adversarial Training

Training Schedule

1. Obtain batch:
$$\mathcal{B} = \left\{(\mathcal{X}^{[n_1]}, t^{[n_1]}), \ldots, (\mathcal{X}^{[n_B]}, t^{[n_B]})\right\}$$
2. Compute gradient ∇_{Θ} for $\mathcal{B}$
3. Create adversarial samples:
$$\check{\mathcal{B}} = \left\{(\check{\mathcal{X}}^{[n_1]}, t^{[n_1]}), \ldots, (\check{\mathcal{X}}^{[n_B]}, t^{[n_B]})\right\}$$
4. Compute gradient $\check{\nabla}_{\Theta}$ for $\check{\mathcal{B}}$
5. Update weights:
$$\Theta = \Theta - \eta(\nabla_{\Theta} + \check{\nabla}_{\Theta})$$

Task 4: Training

1. Dataset: MNIST
2. Categorical cross-entropy loss
3. Implement training schedule
4. Train for $\sim$50 epochs
5. Evaluate after each epoch
   - $\rightarrow$ Accuracy on original images
   - $\rightarrow$ Accuracy on adversarial images

Task 4(b, optional): Plot

1. Plot accuracies over epoch

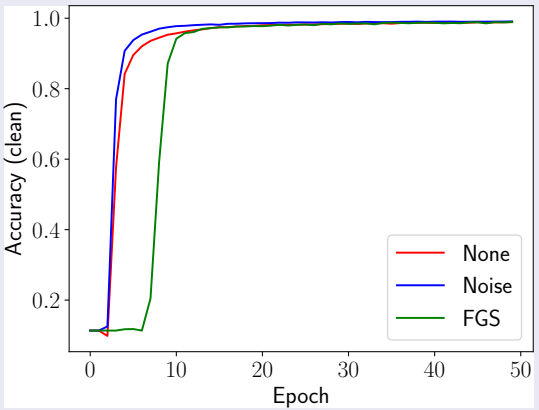
Advarsarial Training

Variations (optional)

- Standard training
  - → Train on clean images only
  - → Evaluate original and adversarial
- Training with noise
  - → $\hat{\mathcal{X}} = \mathcal{X} + \alpha\{-1, +1\}^{D \times E}$
  - → Evaluate original and adversarial
- Network w/o and w/ BatchNorm

Results might not transfer to other datasets!

Accuracy on Original Images



Advarsarial Training

Variations (optional)

- Standard training
  - → Train on clean images only
  - → Evaluate original and adversarial
- Training with noise
  - → $\hat{\mathcal{X}} = \mathcal{X} + \alpha\{-1, +1\}^{D \times E}$
  - → Evaluate original and adversarial
- Network w/o and w/ BatchNorm

Results might not transfer to other datasets!

Accuracy on FGS Samples