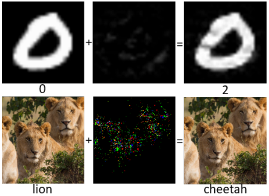
Margin Maximization for Robust Classification Using Deep Learning

Matyasko Alexander, Lap-Pui Chau

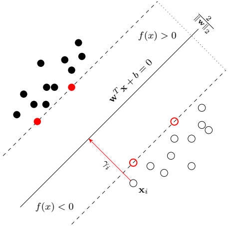
School of Electrical and Electronic Engineering
Nanyang Technological University
Singapore

May 15, 2017

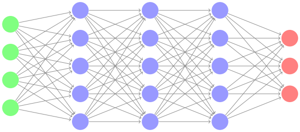
Overview



Adversarial examples

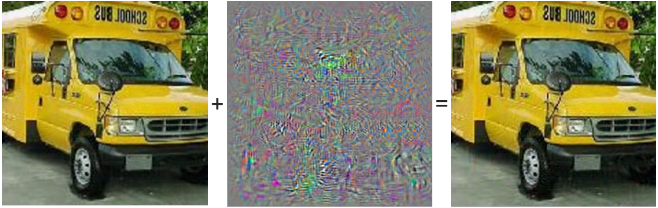


SVM and its robustness



Deep margin maximization

Adversarial Examples



School bus

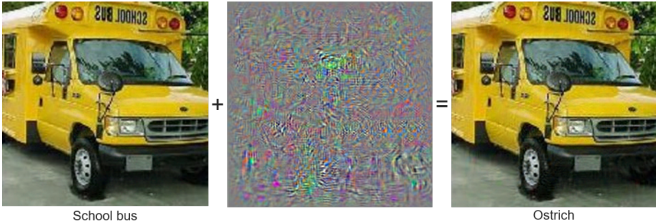
Ostrich

Szegedy et al. 2013

Importance of model robustness:

- Lack of robustness is counter-intuitive and undesirable.
- Improve classifier generalization (Xu et al. 2011).
- Limits applications of deep neural networks in adversarial settings (Papernot et al. 2016).

Adversarial Examples



Szegedy et al. 2013

Importance of model robustness:

- Lack of robustness is counter-intuitive and undesirable.
- Improve classifier generalization (Xu et al. 2011).
- Limits applications of deep neural networks in adversarial settings (Papernot et al. 2016).

Related work

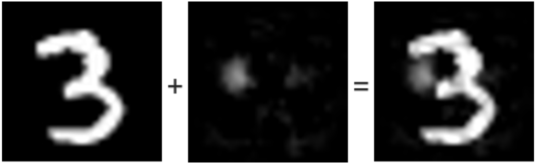
- Attacks:
  - Gradient-based attacks:
    - Fast Gradient Sign (Goodfellow et al. 2015).
    - DeepFool (Moosavi-Dezfooli et al. 2016).
  - Black-box attacks (Papernot et al. 2016).
- Defenses:
  - Data regularization:
    - Adversarial training (Goodfellow et al. 2015).
    - Virtual Adversarial training (Miyato et al. 2015).
  - Model-based regularization:
    - Layer-wise Contractive penalty (Gu et al. 2014).
    - Parseval networks (Moustapha et al. 2017).

Related work

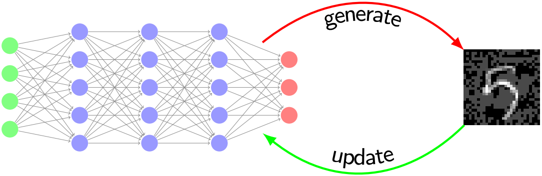
- Attacks:
  - Gradient-based attacks:
    - Fast Gradient Sign (Goodfellow et al. 2015).
    - DeepFool (Moosavi-Dezfooli et al. 2016).
  - Black-box attacks (Papernot et al. 2016).
- Defenses:
  - Data regularization:
    - Adversarial training (Goodfellow et al. 2015).
    - Virtual Adversarial training (Miyato et al. 2015).
  - Model-based regularization:
    - Layer-wise Contractive penalty (Gu et al. 2014).
    - Parseval networks (Moustapha et al. 2017).

Limitations of data regularization

- Perturbation should be label non-changing:

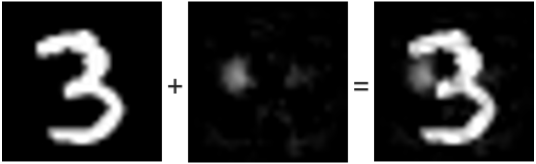


- Model fails to anticipate changes in the adversary:

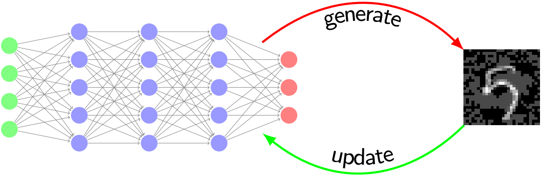


Limitations of data regularization

- Perturbation should be label non-changing:



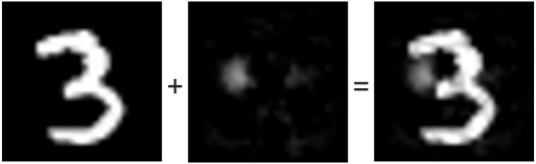
- Model fails to anticipate changes in the adversary:



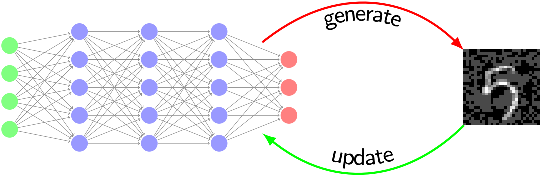
NANYANG
TECHNOLOGICAL
UNIVERSITY

Limitations of data regularization

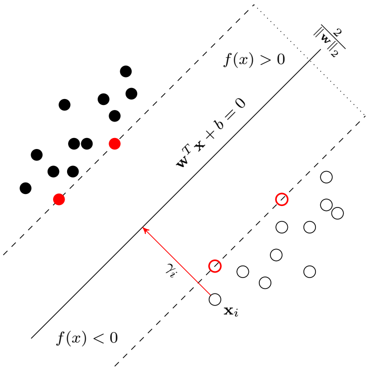
- Perturbation should be label non-changing:



- Model fails to anticipate changes in the adversary:



SVM margin maximization

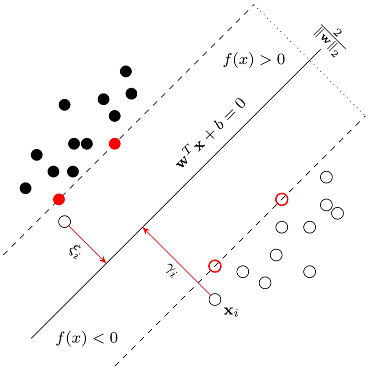


Theorem (Xu et al. 2009)

$$\min : \max_{(\mathbf{r}_1,\ldots,\mathbf{r}_m)\in\mathcal{T}} \sum_{i=1}^{m} \left(1 - y_i\left(\mathbf{w}^T(\mathbf{x}_i - \mathbf{r}_i) + b\right)\right)_+$$

where $\mathcal{T} = \{(\mathbf{r}_i, \ldots, \mathbf{r}_m) \mid \sum_{i=1}^{m} \|\mathbf{r}_i\|^* \leq C\}$.

SVM margin maximization

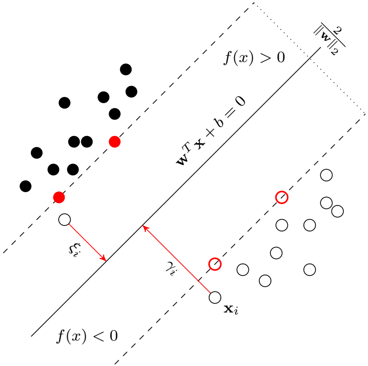


Theorem (Xu et al. 2009)

$$\min : \max_{(\mathbf{r}_1,\ldots,\mathbf{r}_m)\in\mathcal{T}} \sum_{i=1}^{m} \left(1 - y_i\left(\mathbf{w}^T(\mathbf{x}_i - \mathbf{r}_i) + b\right)\right)_+$$

where $\mathcal{T} = \{(\mathbf{r}_i, \ldots, \mathbf{r}_m) \mid \sum_{i=1}^{m} \|\mathbf{r}_i\|^* \leq C\}$.

SVM margin maximization



Theorem (Xu et al. 2009)

$$\min : \max_{(\mathbf{r}_1,\ldots,\mathbf{r}_m)\in\mathcal{T}} \sum_{i=1}^{m} \left(1 - y_i \left(\mathbf{w}^T(\mathbf{x}_i - \mathbf{r}_i) + b\right)\right)_+$$

where $\mathcal{T} = \{(\mathbf{r}_i, \ldots, \mathbf{r}_m) \mid \sum_{i=1}^{m} \|\mathbf{r}_i\|^* \leq C\}$.

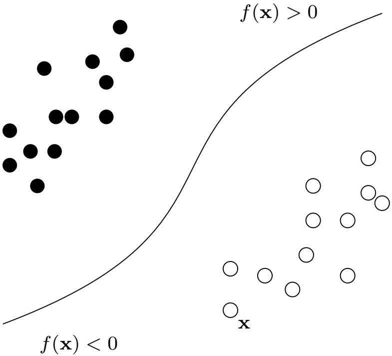
Deep network margin maximization

Geometric margin:

$$\gamma = \min\{\|\mathbf{r}\|_2 \mid f(\mathbf{x}+\mathbf{r}) = 0\}$$

Using first-order approximation:

$$\hat{\gamma} = \frac{|f(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f(\mathbf{x})\|_2}$$



Binary margin maximization

$$\min \sum_{i=1}^{m} (1 - y_i f(\mathbf{x}_i))_+ + C\|\nabla_{\mathbf{x}} f(\mathbf{x}_i)\|_2 \tag{1}$$

Related work: Drucker et al. (1991), Rifai et al. (2011).

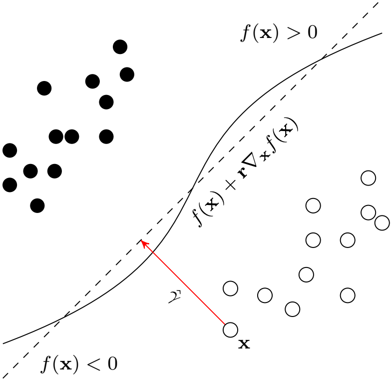
Deep network margin maximization

Geometric margin:

$$\gamma = \min\{\|\mathbf{r}\|_2 \,|\, f(\mathbf{x}+\mathbf{r}) = 0\}$$

Using first-order approximation:

$$\hat{\gamma} = \frac{|f(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f(\mathbf{x})\|_2}$$



Binary margin maximization

$$\min \sum_{i=1}^{m} (1 - y_i f(\mathbf{x}_i))_+ + C\|\nabla_{\mathbf{x}} f(\mathbf{x}_i)\|_2 \quad (1)$$

Related work: Drucker et al. (1991), Rifai et al. (2011).

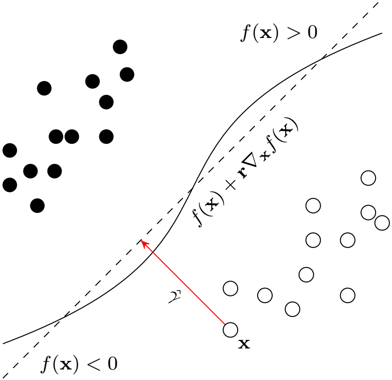
Deep network margin maximization

Geometric margin:

$$\gamma = \min\{\|\mathbf{r}\|_2 \mid f(\mathbf{x}+\mathbf{r}) = 0\}$$

Using first-order approximation:

$$\hat{\gamma} = \frac{|f(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f(\mathbf{x})\|_2}$$



Binary margin maximization

$$\min \sum_{i=1}^{m} (1 - y_i f(\mathbf{x}_i))_+ + C\|\nabla_{\mathbf{x}} f(\mathbf{x}_i)\|_2 \qquad (1)$$

Related work: Drucker et al. (1991), Rifai et al. (2011).

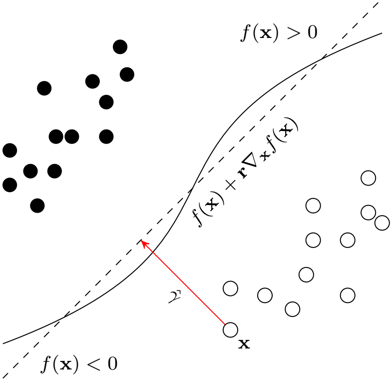
Deep network margin maximization

Geometric margin:

$$\gamma = \min\{\|\mathbf{r}\|_2 \mid f(\mathbf{x}+\mathbf{r}) = 0\}$$

Using first-order approximation:

$$\hat{\gamma} = \frac{|f(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f(\mathbf{x})\|_2}$$



Binary margin maximization

$$\min \sum_{i=1}^{m} \boxed{(1 - y_i f(\mathbf{x}_i))_+} + C\|\nabla_{\mathbf{x}} f(\mathbf{x}_i)\|_2 \qquad (1)$$

Related work: Drucker et al. (1991), Rifai et al. (2011).

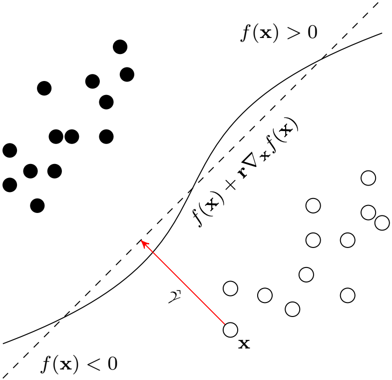
Deep network margin maximization

Geometric margin:

$$\gamma = \min\{\|\mathbf{r}\|_2 \,|\, f(\mathbf{x}+\mathbf{r}) = 0\}$$

Using first-order approximation:

$$\hat{\gamma} = \frac{|f(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f(\mathbf{x})\|_2}$$



Binary margin maximization

$$\min \sum_{i=1}^{m} (1 - y_i f(\mathbf{x}_i))_+ + C\boxed{\|\nabla_{\mathbf{x}} f(\mathbf{x}_i)\|_2} \quad (1)$$

Related work: Drucker et al. (1991), Rifai et al. (2011).

Deep network margin maximization

Geometric margin:

Theorem (See paper for details)

Let $\mathcal{T}_i = \{\mathbf{r}_i \,|\, \|\mathbf{r}_i\|^* \leq C\}$ be an uncertainty set where $\mathbf{r}_i$ is the perturbation for $\mathbf{x}_i$. Then, the optimization problem in eq. (1) approximately minimizes the following robust optimization problem:

$$\min : \sum_{i=1}^{m} \max_{\mathbf{r}_i \in \mathcal{T}_i} (1 - y_i f(\mathbf{x}_i - \mathbf{r}_i))_+$$

Binary margin maximization

$$\min \sum_{i=1}^{m} (1 - y_i f(\mathbf{x}_i))_+ + C \|\nabla_{\mathbf{x}} f(\mathbf{x}_i)\|_2 \tag{1}$$

Related work: Drucker et al. (1991), Rifai et al. (2011).

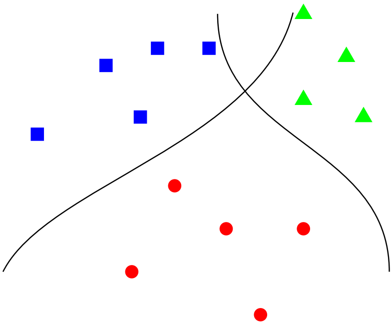
Multiclass DNN margin maximization

Margin between class i and j:

$$\gamma_{i,j} = \frac{|f_i(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$

Datapoint margin:

$$\hat{\gamma} = \min_{j \neq y} \frac{|f_y(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_y(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$



Multiclass deep margin maximization (Theorem IV.2)

$$\min \sum_{i=1}^{m} \max_{j \neq y_i} \left(1 + f_{y_i}(\mathbf{x}_i) - f_j(\mathbf{x}_i)\right)_+ + C \max_{j \neq i} \|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|$$

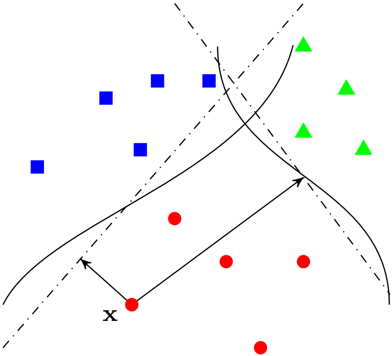
Multiclass DNN margin maximization

Margin between class i and j:

$$\gamma_{i,j} = \frac{|f_i(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$

Datapoint margin:

$$\hat{\gamma} = \min_{j \neq y} \frac{|f_y(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_y(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$



Multiclass deep margin maximization (Theorem IV.2)

$$\min \sum_{i=1}^{m} \max_{j \neq y_i} \left(1 + f_{y_i}(\mathbf{x}_i) - f_j(\mathbf{x}_i)\right)_+ + C \max_{j \neq i} \|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|$$

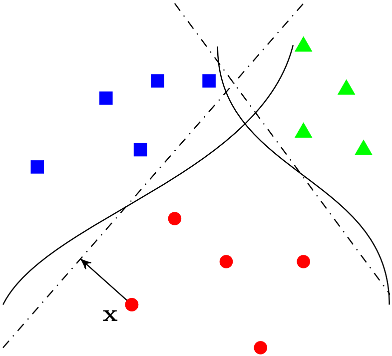
Multiclass DNN margin maximization

Margin between class i and j:

$$\gamma_{i,j} = \frac{|f_i(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$

Datapoint margin:

$$\hat{\gamma} = \min_{j \neq y} \frac{|f_y(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_y(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$



Multiclass deep margin maximization (Theorem IV.2)

$$\min \sum_{i=1}^{m} \max_{j \neq y_i} (1 + f_{y_i}(\mathbf{x}_i) - f_j(\mathbf{x}_i))_+ + C \max_{j \neq i} \|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|$$

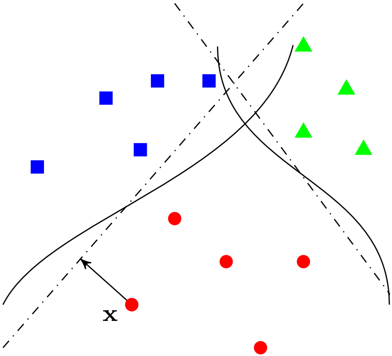
Multiclass DNN margin maximization

Margin between class i and j:

$$\gamma_{i,j} = \frac{|f_i(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$

Datapoint margin:

$$\hat{\gamma} = \min_{j \neq y} \frac{|f_y(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_y(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$



Multiclass deep margin maximization (Theorem IV.2)

$$\min \sum_{i=1}^{m} \max_{j \neq y_i} \left(1 + f_{y_i}(\mathbf{x}_i) - f_j(\mathbf{x}_i)\right)_+ + C \max_{j \neq i} \|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|$$

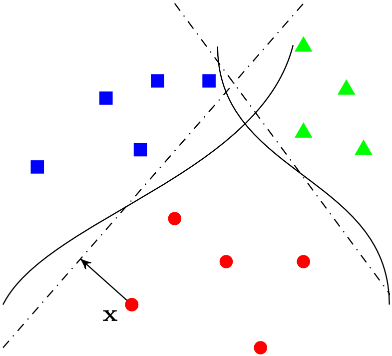
Multiclass DNN margin maximization

Margin between class i and j:

$$\gamma_{i,j} = \frac{|f_i(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$

Datapoint margin:

$$\hat{\gamma} = \min_{j \neq y} \frac{|f_y(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_y(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$



Multiclass deep margin maximization (Theorem IV.2)

$$\min \sum_{i=1}^{m} \boxed{\max_{j \neq y_i} \left(1 + f_{y_i}(\mathbf{x}_i) - f_j(\mathbf{x}_i)\right)_+} + C \max_{j \neq i} \|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|$$

See Crammer et al. (2002).

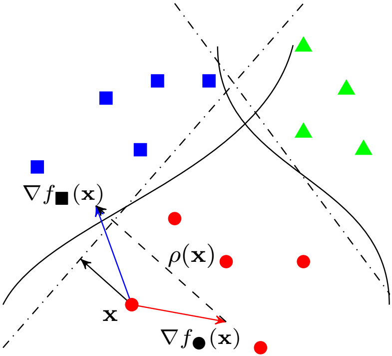
Multiclass DNN margin maximization

Margin between class i and j:

$$\gamma_{i,j} = \frac{|f_i(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$

Datapoint margin:

$$\hat{\gamma} = \min_{j \neq y} \frac{|f_y(\mathbf{x}) - f_j(\mathbf{x})|}{\|\nabla_{\mathbf{x}} f_y(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$



Multiclass deep margin maximization (Theorem IV.2)

$$\min \sum_{i=1}^{m} \max_{j \neq y_i} (1 + f_{y_i}(\mathbf{x}_i) - f_j(\mathbf{x}_i))_+ + C \boxed{\max_{j \neq i} \|\nabla_{\mathbf{x}} f_i(\mathbf{x}) - \nabla_{\mathbf{x}} f_j(\mathbf{x})\|}$$

Experiments: MNIST

Network architectures:

- Fully-connected network (784-1000-1000-1000-10)
- Lenet-5 convolutional network

Average robustness:

$$\rho_{\mathsf{adv}}(f) = \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \frac{\|\mathbf{r}(\mathbf{x})\|_2}{\|\mathbf{x}\|_2}$$

Algorithms:

- Baseline
- Dropout (Srivastava et al. 2014)
- AT (Goodfellow et al. 2015)
- VAT (Miyato et al. 2015)
- Our l_1-margin maximization
- Our l_2-margin maximization

Experiments: MNIST

Network architectures:

- Fully-connected network (784-1000-1000-1000-10)
- Lenet-5 convolutional network

Average robustness:

$$\rho_{\text{adv}}(f) = \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \frac{\|\mathbf{r}(\mathbf{x})\|_2}{\|\mathbf{x}\|_2}$$

Algorithms:

- Baseline
- Dropout (Srivastava et al. 2014)
- AT (Goodfellow et al. 2015)
- VAT (Miyato et al. 2015)
- Our l_1-margin maximization
- Our l_2-margin maximization

Network	Error %	$\rho_{\text{adv}} \times 10^{-1}$
Baseline	1.42 ± 0.08	1.14 ± 0.01
Dropout	1.34 ± 0.05	1.20 ± 0.01
AT	1.19 ± 0.06	1.60 ± 0.05
VAT	$\mathbf{0.87 \pm 0.04}$	$\mathbf{2.69 \pm 0.02}$
Our l_1	$\mathbf{0.84 \pm 0.03}$	$\mathbf{2.73 \pm 0.08}$
Our l_2	$\mathbf{0.86 \pm 0.04}$	2.59 ± 0.05

Experiments: MNIST

Network architectures:

- Fully-connected network (784-1000-1000-1000-10)
- Lenet-5 convolutional network

Average robustness:

$$\rho_{\mathsf{adv}}(f) = \frac{1}{|\mathcal{D}|} \sum_{\mathbf{x} \in \mathcal{D}} \frac{\|\mathbf{r}(\mathbf{x})\|_2}{\|\mathbf{x}\|_2}$$

Algorithms:

- Baseline
- Dropout (Srivastava et al. 2014)
- AT (Goodfellow et al. 2015)
- VAT (Miyato et al. 2015)
- Our l_1-margin maximization
- Our l_2-margin maximization

Network	Error %	$\rho_{\mathsf{adv}} \times 10^{-1}$
Baseline	0.72 ± 0.06	1.54 ± 0.04
Dropout	$\mathbf{0.58 \pm 0.03}$	1.70 ± 0.05
AT	0.73 ± 0.05	2.00 ± 0.03
Our l_1	0.64 ± 0.02	$\mathbf{2.22 \pm 0.05}$
Our l_2	0.62 ± 0.04	$\mathbf{2.17 \pm 0.06}$

Experiments: MNIST (cont.)

Qualitative comparison

Proposition

Ideally, images which are adversarial for neural network should be visually confusing for humans.

Experiments: MNIST (cont.)

Qualitative comparison

Proposition

Ideally, images which are adversarial for neural network should be visually confusing for humans.

Experiments: MNIST (cont.)

Qualitative comparison

Proposition

Ideally, images which are adversarial for neural network should be visually confusing for humans.

Experiments: MNIST (cont.)

Qualitative comparison

Proposition

Ideally, images which are adversarial for neural network should be visually confusing for humans.

Experiments: MNIST (cont.)

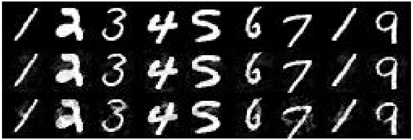
Qualitative comparison

Proposition

Ideally, images which are adversarial for neural network should be visually confusing for humans.

Conclusion

- We extended margin maximization to deep neural networks. We theoretically showed that the proposed objective is equivalent to the robust optimization problem.
- The proposed objective improves network robustness both quantitatively and qualitatively.

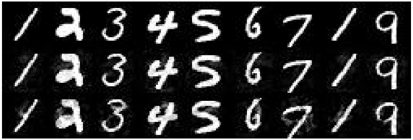


Future work

- Extensions to other problems.
- Address scalability issues.
- Comparison of algorithms based on how humans perceive visually confusing images.

Conclusion

- We extended margin maximization to deep neural networks. We theoretically showed that the proposed objective is equivalent to the robust optimization problem.
- The proposed objective improves network robustness both quantitatively and qualitatively.

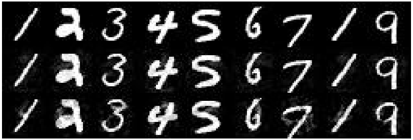


Future work

- Extensions to other problems.
- Address scalability issues.
- Comparison of algorithms based on how humans perceive visually confusing images.

Conclusion

- We extended margin maximization to deep neural networks. We theoretically showed that the proposed objective is equivalent to the robust optimization problem.
- The proposed objective improves network robustness both quantitatively and qualitatively.

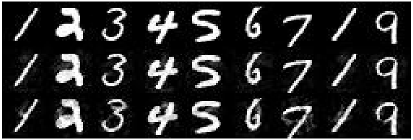


Future work

- Extensions to other problems.
- Address scalability issues.
- Comparison of algorithms based on how humans perceive visually confusing images.

Conclusion

- We extended margin maximization to deep neural networks. We theoretically showed that the proposed objective is equivalent to the robust optimization problem.
- The proposed objective improves network robustness both quantitatively and qualitatively.

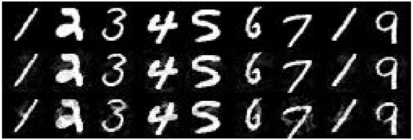


Future work

- Extensions to other problems.
- Address scalability issues.
- Comparison of algorithms based on how humans perceive visually confusing images.

Conclusion

- We extended margin maximization to deep neural networks. We theoretically showed that the proposed objective is equivalent to the robust optimization problem.
- The proposed objective improves network robustness both quantitatively and qualitatively.



Future work

- Extensions to other problems.
- Address scalability issues.
- Comparison of algorithms based on how humans perceive visually confusing images.

Thank you for your attention! Any questions?

Contributions

- We proposed novel margin maximization framework for deep neural networks.
- We theoretically showed that the proposed objective is equivalent to the robust optimization problem.
- The proposed objective improves network robustness both quantitatively and qualitatively.

Future work

- Extensions to other problems.
- Address scalability issues.
- Comparison based on how humans perceive visually confusing images.