technische universität dortmund

fakultät für informatik

Seminar „Uncertainty quantification in machine learning“

Adversarial Examples can be Effective Data Augmentation for Unsupervised Machine Learning

Waldemar Voos

11.06.2022

technische universität dortmund

fakultät für informatik

Outline

technische universität dortmund

fakultät für informatik

Motivation

What is the article about

What are we doing?

- Creating framework to generate supervised and unsupervised adversial examples

What do we reach?

- Higher robustness and visuality compared to other frameworks so far

How do we get there?

- Mutual information neural estimator (MINE)
- A new MinMax optimization algorithm

Generate Supervised Adversial Examples

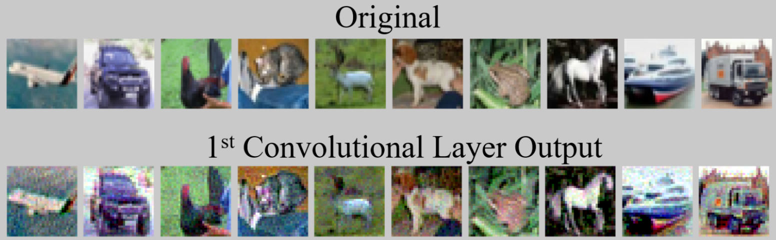


Abbildung: Supervised Adversial Examples from our framework

Generate Unsupervised Adversial Examples

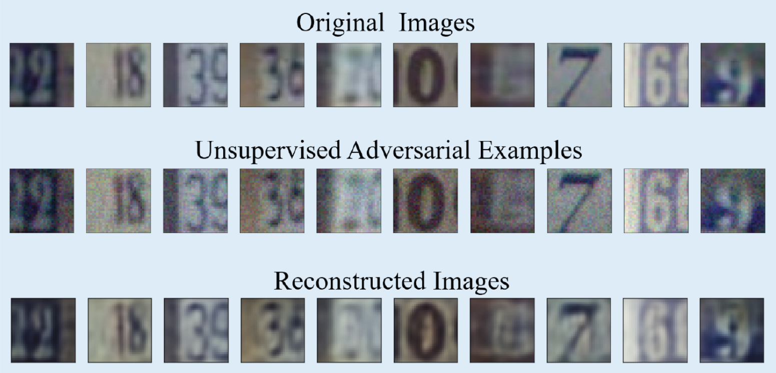


Abbildung: Unsupervised Adversial Examples in Data Reconstruction Task

technische universität dortmund

fakultät für informatik

Outline

Mutual information

What do we calculate?

- Dependency between two random variables X and Z (I(X,Z))

X \ Y	0	1
0	0,25	0,25
1	0,25	0,25

I(X,Y) = 0

X \ Y	0	1
0	0,5	0
1	0	0,5

I(X,Y) = 1

Abbildung: Example for Mutual Information

Convolutional Neural Network (CNN)

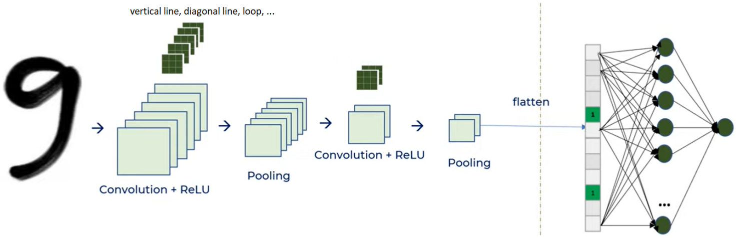


Abbildung: Convolutional Neural Network
($https://www.youtube.com/watch?v=zfiSAzpy9NM$)

Convolution Layer

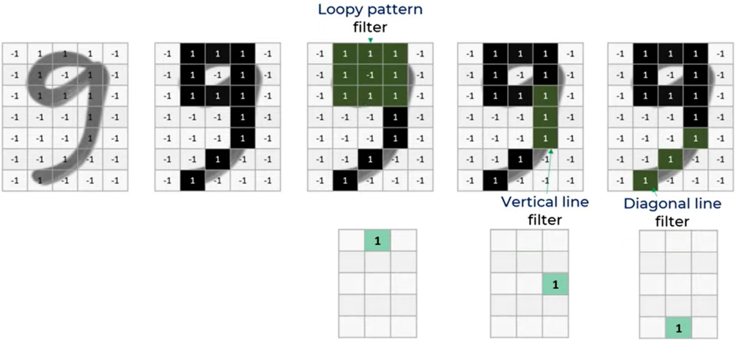


Abbildung: Filters and Feature Maps of a convolutional layer ($https://www.youtube.com/watch?v=zfiSAzpy9NM$)

MINE Algorithm

Require: input sample x, perturbed sample $x+\delta$, 1st convolution layer output $conv(\cdot)$, MI neural estimator $I(\theta)$

Initialize neural network parameters θ

Get $\{conv(x)_k\}_{k=1}^{K}$ and $\{conv(x+\delta)_k\}_{k=1}^{K}$ via 1^{st} convolution layer

for t in T_I iterations **do**

  Take K samples from the joint distribution: $\{conv(x)_k, conv(x+\delta)_k\}_{k=1}^{K}$

  Shuffle K samples from $conv(x+\delta)$ marginal distribution: $\{conv(x+\delta)_{(k)}\}_{k=1}^{K}$

  Evaluate the mutual information estimate $I(\theta) \leftarrow \frac{1}{K}\sum_{k=1}^{K} T_\theta(conv(x)_k, conv(x+\delta)_k) - \log\left(\frac{1}{K}\sum_{k=1}^{K}\exp[T_\theta(conv(x)_k, conv(x+\delta)_{(k)})]\right)$

  $\theta \leftarrow \theta + \nabla_\theta I(\theta)$

Return $I(\theta)$

Abbildung: Per-sample MINE via Convolution

technische universität dortmund

fakultät für informatik

Outline

technische universität dortmund

fakultät für informatik

Convex Optimization Function

Definition

$$\underset{\delta: x+\delta \in [0,1]^d,\ \delta \in [-\epsilon,\epsilon]^d}{\mathrm{Min}} \quad \underset{c \geq 0}{\mathrm{Max}} \quad F(\delta, c) \triangleq c \cdot f_x^+(x+\delta) - I_\Theta(x, x+\delta)$$

Abbildung: MinMax function to optimize

- δ-constraints caused by L_p-Norm bounded perturbation and normalization
- If attack criterion $f_x(x+\delta) \leq 0$ adversial example found
- For UAE mutual information sign changes

MINMAX Algorithm

Algorithm 1: MinMax Attack Algorithm

Require: data sample x, attack criterion $f_x(\cdot)$, step sizes α and β, perturbation bound ϵ, # of iterations T

Initialize $\delta_0 = 0$, $c_0 = 0$, $\delta^* = \text{null}$, $I_\Theta^* = -\infty$, $t = 1$

for t in T iterations **do**

- $\delta_{t+1} = \delta_t - \alpha \cdot (c \cdot \nabla f_x^+(x + \delta_t) - \nabla I_\Theta(x, x + \delta_t))$
- Project δ_{t+1} to $[-\epsilon, \epsilon]$ via clipping
- Project $x + \delta_{t+1}$ to $[0, 1]$ via clipping
- Compute $I_\Theta(x, x + \delta_{t+1})$
- Perform $c_{t+1} = (1 - \frac{\beta}{t^{1/4}}) \cdot c_t + \beta \cdot f_x^+(x + \delta_{t+1})$
- Project c_{t+1} to $[0, \infty]$
- **if** $f_x(x+\delta_{t+1}) \leq 0$ and $I_\Theta(x, x+\delta_{t+1}) > I_\Theta^*$ **then**
  - update $\delta^* = \delta_{t+1}$ and $I_\Theta^* = I_\Theta(x, x + \delta_{t+1})$

Return δ^*, I_Θ^*

Abbildung: MINMAX Algorithm

technische universität dortmund

fakultät für informatik

Outline

technische universität dortmund

fakultät für informatik

Outline

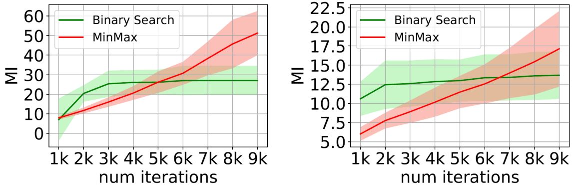
technische universität dortmund

fakultät für informatik

Mutual Information and Attack Success Rate

	MNIST		CIFAR-10	
	ASR	MI	ASR	MI
Penalty-based	100%	28.28	100%	13.69
MinMax	100%	**51.29**	100%	**17.14**

Abbildung: ASR and MI value over 1000 samples



(a) MNIST (b) CIFAR-10

Abbildung: Mean and Standard Deviation of MI value over 1000 samples

MinMax attacks vs. PGD attacks

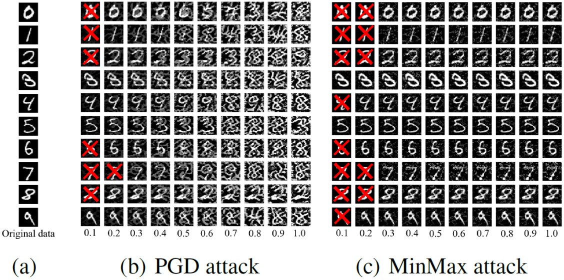


Abbildung: ASR and MI value over 1000 samples

technische universität dortmund

fakultät für informatik

Outline

technische universität dortmund

fakultät für informatik

Reconstruction Loss

MNIST									
	Reconstruction Error (test set)					ASR (training set)			
Autoencoder	Original	MINE-UAE	L_2-UAE	GA ($\sigma = 0.01$)	GA ($\sigma = 10^{-3}$)	MINE-UAE	L_2-UAE	GA ($\sigma = 0.01$)	GA ($\sigma = 10^{-3}$)
Sparse	0.00561	**0.00243** (↑ 56.7%)	0.00348 (↑ 38.0%)	0.00280±2.60e-05 (↑ 50.1%)	0.00280±3.71e-05 (↑ 50.1%)	100%	99.18%	54.10%	63.95%
Dense	0.00258	**0.00228** (↑ 11.6%)	0.00286 (↓ 6.0%)	0.00244±0.00014 (↑ 5.4%)	0.00238±0.00012 (↑ 7.8%)	92.99%	99.94%	48.53%	58.47%
Convolutional	0.00294	**0.00256** (↑ 12.9%)	0.00364 (↓ 23.8%)	0.00301±0.00011 (↓ 2.4%)	0.00304±0.00015 (↓ 3.4%)	99.86%	99.61%	68.71%	99.61%
Adversarial	0.04785	**0.04581** (↑ 4.3%)	0.06098 (↓ 27.4%)	0.05793±0.00501 (↓ 21%)	0.05544±0.00567 (↓ 15.86%)	98.46%	43.54%	99.79%	99.83%
SVHN									
Sparse	0.00887	**0.00235** (↑ 73.5%)	0.00315 (↑ 64.5%)	0.00301±0.00137 (↑ 66.1%)	0.00293±0.00078 (↑ 67.4%)	100%	72.16%	72.42%	79.92%
Dense	0.00659	**0.00421** (↑ 36.1%)	0.00550 (↑ 16.5%)	0.00858±0.00232 (↓ 30.2%)	0.00860±0.00190 (↓ 30.5%)	99.99%	82.65%	92.3%	93.92%
Convolutional	0.00128	**0.00095** (↑ 25.8%)	0.00121 (↑ 5.5%)	0.00098 ± 3.77e-05 (↑ 25.4%)	0.00104±7.41e-05 (↑ 18.8%)	100%	56%	96.40%	99.24%
Adversarial	0.00173	**0.00129** (↑ 25.4%)	0.00181 (↓ 27.4%)	0.00161±0.00061 (↑ 6.9%)	0.00130±0.00037 (↑ 24.9%)	94.82%	58.98%	97.31%	99.85%

Abbildung: Reconstruction Loss comparison with different Autoencoders

Questions ?