Singularities in Deep Neural Networks:

A Brief Discussion about Decision Boundary and Classifiers

August 12th, 2020

FAPESP SPRINT Project

Title: Applications of Singularity Theory on Deep Neural Networks

Scientific Research Student: Alan Gonelli Miranda (INCTMat/CNPq fellowship)

ICMC coordinator: Raimundo N. Araújo dos Santos (SMA-ICMC)

i-PRoBe Lab / MSU coordinator: Arun Ross (MSU)

ICMC USP SÃO CARLOS

1. Introduction

❑1.1 Motivation to Study Adversarial Examples

- Deep neural networks (DNNs) are vulnerable against adversarial examples: crafted instances whose goal is to cause errors and false predictions on the classifiers.
- Adversary examples pose security concerns as they can perform an attack to machine learning systems even if they do not have access to the model
- DNNs have impacted different research areas such as computer vision, speech recognition and natural language processing. DNNs are vulnerable against adversarial manipulations at testing time.
- An attacker can add such a small carefully crafted noise to the testing example so the DNN classifier predicts an incorrect label. In this case the crafted testing examples is called adversarial exampled and the attack is evasion attack.
- Evasion attacks are a real risk for developing DNNs in security and safety contexts, so our main goal here is to set up methods to defend against evasion attacks

1. Introduction

❑1.2 What kinds of neighborhood to study?

- Let us **Analyse local neighborhoods in the input space of DNN models.** Previous works considered small balls or low-dimensional subspaces. *Liu et al (*2017) and *Tramèr et al* (2017) proposed limited regions around benign examples. Main goal: explain why some adversarial examples transfer across different models.

- However the relationship between DNN models and adversarial examples is better characterized when **considering larger neighborhoods.** This was confirmed by *Cao & Gong* (2017) that suggest to consider the region around the input produces more robust classification than considering the input as a single point. **[1]**

❑ 1.3 How to analyse?

- Some attacks (*OPTMargin*) evade region classification based in a small ball around an input instance (*Cao & Gong*). A possible approach consists of looking at properties of **surrounding decision boundaries in the input space of the model**, which are: distance to the boundaries and the adjacent classes. Therefore, we can characterize the robustness of the attack and compare the decision boundaries between the adversarial and benign by metrics such distances from the examples to adjacent classes.

1. Introduction

❑ 1.4 What will be analysed?

- Defense Methods : Adversarial Training, *projected gradient descent* (PGD) and region classification;
- Attacks Methods: OPTMARGIN, OPTBRITTLE and *Fast Gradient Sign Method* (FGSM)
- Databases, Academic Image Classification:
  - MNIST, image's pixel in range [0,1];
  - CIFAR-10, image's pixel in range [0,255];

❑1.5 Main Approach

- Demonstrate attacks (OPTMARGIN), able to evade region classification domains with low-distortion adversarial examples;
- Analysis of decision boundaries around an input to talk about how effective are adversarial examples
- The importance of decision boundary information: Useful to train a classifier to differentiate the information which comes from different kinds of input instances.

2. CIFAR-10 Dataset

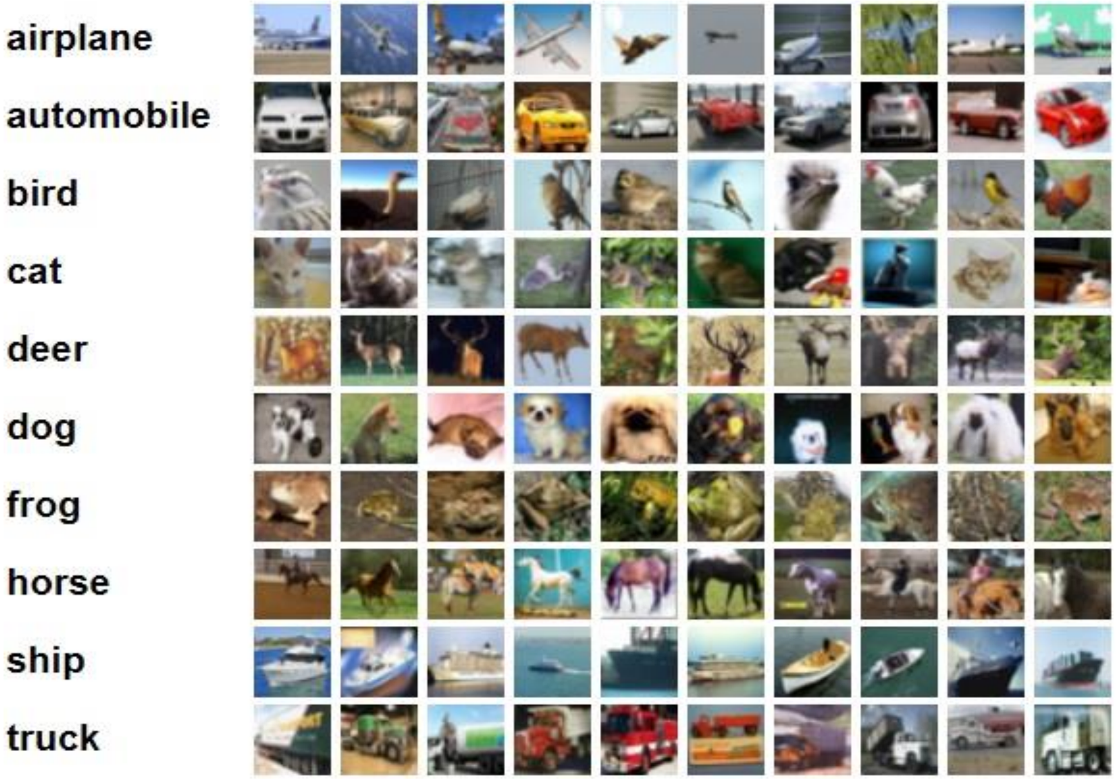
❑ 2.1 Main Elements

The CIFAR-10 dataset consists of:

- 60000 32x32 colour images in 10 classes, with 6000 images per class.
- There are 50000 training images and 10000 test images.
- The dataset is divided into five training batches and one test batch, each with 10000 images.
- The test batch has 1000 randomly-selected images from each class.
- The training batches have the remaining images in a random order and contain exactly 5000 images from each class.
- The classes are mutually exclusive, so there is no overlap between them.
- *Versions: Matlab, Python and Binary for C programs.*

2. CIFAR-10 Dataset

Figure 1: CIFAR-10 Dataset **[4]**



Source: *Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton****. CIFAR 10 Dataset****. Computer Science, University of Toronto. Available at:https://www.cs.toronto.edu/~kriz/cifar.html*

3. MNIST Dataset

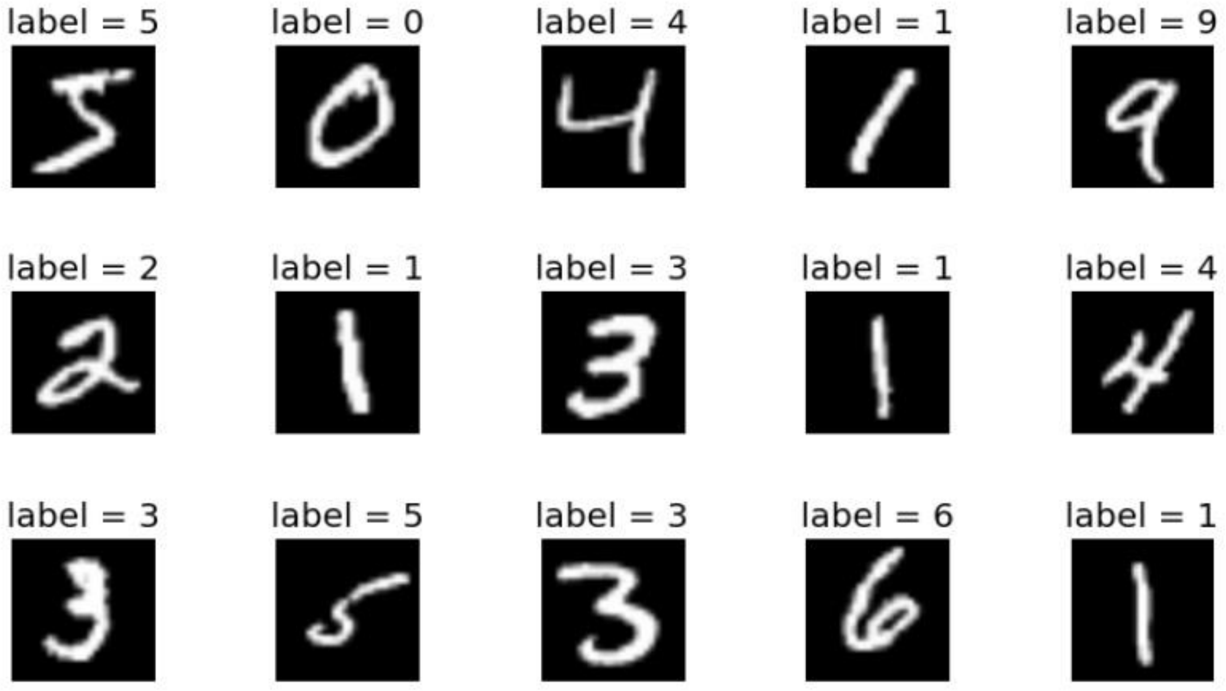
❑ 3.1 Main Elements

MNIST is a set of small images of handwritten digits.

- Acronym that stands for Modified National Institute of Standards and Technology database
- 60,000 small square 28x28 pixel grayscale images of handwritten numbers sorted from 0 to 9;
- **Basic Task:** Classify a given image of handwritten digit into 1 of 10 classes available by returning integers values from 0 to 9;
- It is a large database of handwritten digits that is commonly used for training various image processing systems. The database is also widely used for training and testing.
- Top performing- models are convolutional neural networks (CNNs) with a classification accuracy above 95% and an error rate of 0.3% on the dataset.

3. MNIST Dataset

Figure 2: Example of MNIST Dataset



Source*: Available at www.towardsdatascience.com*

4. Mathematical Notation

4.1 Image Classification main notation

- ❑ $f: model\ - point\ classifier$
- ❑ $x \in \mathbb{R}^{height\ x\ width\ x\ channels}: image\ representation$
- ❑ $f(x): label\ of\ model\ evaluated\ by\ the\ classifier$
- ❑ $C: set\ of\ classes$

  ↳ $Input\ instance\ come\ from\ a\ continuous\ high - dimensional\ space\ while\ the\ output\ is\ discrete$

- ❑ $x_{RMS}: root\ mean\ square\ of\ a\ set\ of\ values\ \{x_1, x_2, \dots \dots, x_n\}$
- ❑ $f_{RMS}: root\ mean\ square\ for\ a\ fucntion\ over\ all\ time$
- ❑ $l: loss\ term\ for\ each\ model\ in\ the\ ensemble$

4. Mathematical Notation

- v_i: perturbations applied to input
- x': Adversarial example, perceptually indistinguible from benign example
- j: incorrectly classified: $j \neq y_{true}$
- y_{true}: true class for the image x
- k: confidence margin (logits)
- ε: magnitude of random orthogonal (uniform);
- $Z(x)$: $|C|$ dimensional vector of class weights (logits) used by f to classify the image
- Percentage numbers in study: Success rate of adversariak examples generated

5. Problem Setup

- **5.1 Kinds of Adversarial Examples:**

- *Targeted*: Examples which are incorrectly classified as an attacker chosen-class

- *Untargeted*: Examples which are misclassified as any class other than the correct

- **5.2 Distortion:**

- Amount of perturbation used to generate an adversarial example from the original input instance.

- Measured by the *root-mean-square* **(RMS)** distance metric between the *original input distance* and the *adversarial example*.

- **5.3 Kinds of Defenses:**

- *Adversarial Training with examples generated by projected gradient descent (PGD);*

- *Region Classification*;

5. Problem Setup

❑ 5.4 Region Classification

- A defense against adversarial examples which takes the *majority prediction* on a lot of slightly *perturbed versions* of input, sampled uniformly from a hypercube around it. In this case, the majority prediction across a neighborhood around an input is considered as a region.

- It is opposite to the traditional method of classifying only the input instance: *point classification*.

- According to Cao & Gong, region classification is a good defense against low-distortion adversarial examples generated by existing attacks.

❑ 5.5 OPTMARGIN Attack

- Generate low-distortion adversarial examples, robust to small perturbations.

- Classifies a small number of perturbed inputs points ---> Ensemble of models:

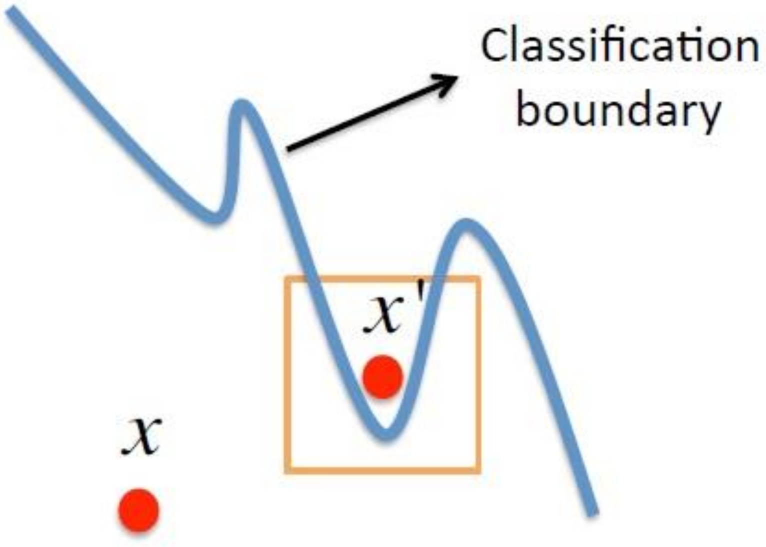
$$f_i(x) = f(x + v_i) \qquad (1)$$

$f: point\ classifier\ used\ in\ the\ region\ classifier$

$v_i: perturbations\ applied\ to\ the\ input\ x$

5. Problem Setup

Figure 3: Illustration of our region-based classification. x is a testing benign example and x' is the corresponding adversarial example. The hypercube centered at x' intersects the most with the class region that has the true label



Source: *Xiaoyu Cao & Neil Zhenqiang Gong.* ***Mitigating Evasion Attacks to Deep Neural Networks via Region-based Classification.*** *Duke University, 31 December 2019.*

5. Problem Setup

□ 5.6 Fundamental Equations

- Utilize optimization attack techniques to craft an example able to fool the entire ensemble while minimizing the distortion
- Similar with *Carlini & Wagner* $L_2's$ attack:

$$l_i(x') = l(x' + v_i) = \max\left(-k, Z\,(x' + v_i)_{\,y} - \max\left\{Z(x' + v_i)_{\,j} : j \neq y\right\}\right) \qquad (3)$$

- The loss term increases when model f_i predicts the correct class y over the next most likely class
- In OPTMARGIN, it is set $k = 0$ ----→ the model barely misclassifies the input
- This approach can be extended to an objective function related to the sum of the terms:

$$minimize \qquad ||x' - x||_2^2 + c * (l_1(x') + \cdots + l_n(x')) \qquad (4)$$

- 20 classifiers in the attacker's ensemble:

  $v_1\,, \ldots \ldots . . v_{19}: random\ orthogonal\ vectors\ with\ uniform\ \varepsilon\ magnitude$

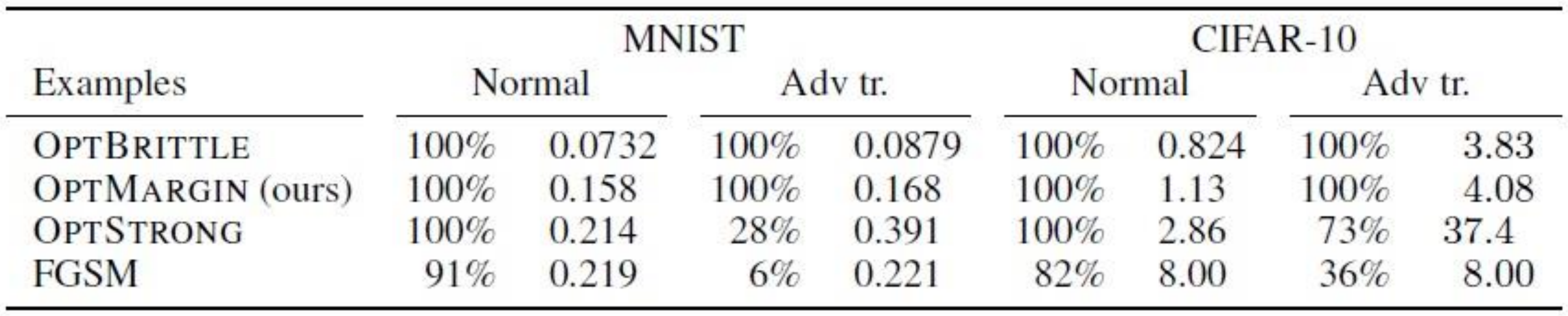
  $v_{20}: 0$

- This choice is related to make it likely for a random perturbation to lie in the region between the v_i
- *For stability in optimization*: fixed values of v_i throughout the optimization of the attack

6. Results and Discussions

❑6.1 Distortion Evaluation

Table 1: *Success rate* (%) and *average distortion* (RMS) of adversarial examples generated by different attacks. On *MNIST*, the level of distortion in OPTMARGIN examples is visible to humans, but the original class is still distinctly visible



	MNIST				CIFAR-10			
Examples	Normal		Adv tr.		Normal		Adv tr.	
OPTBRITTLE	100%	0.0732	100%	0.0879	100%	0.824	100%	3.83
OPTMARGIN (ours)	100%	0.158	100%	0.168	100%	1.13	100%	4.08
OPTSTRONG	100%	0.214	28%	0.391	100%	2.86	73%	37.4
FGSM	91%	0.219	6%	0.221	82%	8.00	36%	8.00

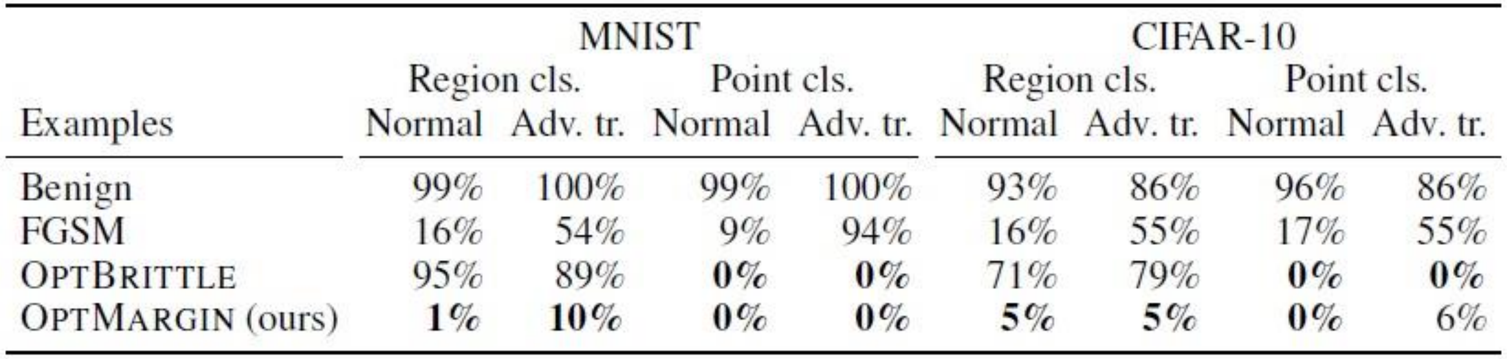
Source: Warren, Bo Li & Dawn Song. **Decision Boundary Analysis of Adversarial Examples**. Computer Science Division. University of California, Berkeley, International Conference on Learning Representations, 2018.

- Average distortion is averaged over the successful adversarial examples
- FGSM samples are also less successful on the PGD adversary trained models

6. Results and Discussions

❑6.2 Evading Region Classification

Table 2: Accuracy of region classification and point classification on examples from diferente attacks



	MNIST				CIFAR-10			
	Region cls.		Point cls.		Region cls.		Point cls.	
Examples	Normal	Adv. tr.	Normal	Adv. tr.	Normal	Adv. tr.	Normal	Adv. tr.
Benign	99%	100%	99%	100%	93%	86%	96%	86%
FGSM	16%	54%	9%	94%	16%	55%	17%	55%
OPTBRITTLE	95%	89%	**0%**	**0%**	71%	79%	**0%**	**0%**
OPTMARGIN (ours)	**1%**	**10%**	**0%**	**0%**	**5%**	**5%**	**0%**	6%

Source: Warren, Bo Li & Dawn Song. **Decision Boundary Analysis of Adversarial Examples**. Computer Science Division. University of California, Berkeley, International Conference on Learning Representations, 2018.

- More effective attacks result in lower accuracy. The attacks that achieve the lowest accuracy for each configuration of defenses are shown in **bold**.

6. Results and Discussions

❑6.3 Decision Boundary

- Surfaces in the model's input space where the output prediction changes among the classes.

6.3.1 Decision Boundary Distance

- Estimation of the distance to the decision boundary in a sample of random direction's in the model's input space, starting from a given input point. So, in each direction the distance can be calculated by the model's prediction on the perturbed inputs atpoints along the direction.

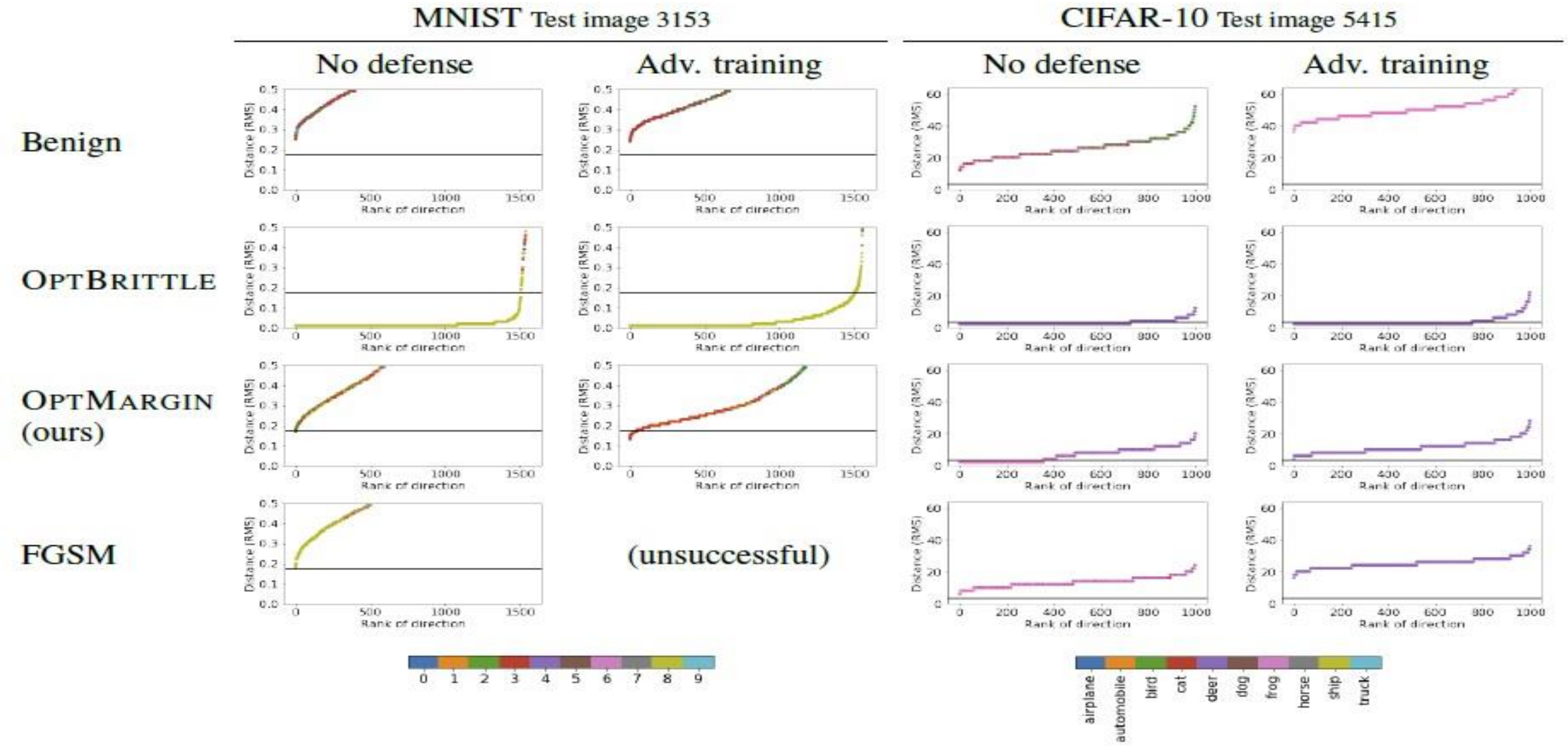
- When the model's prediction on the perturbed image changes from the prediction on the original image, the distance is used asa measure of how far the decision boundary is in that direction.

6.3.2 Adjacent Class Purity

- Adversarial examples tend to have most directions lead to a boundary adjacent to a single class;

- $Purity\ of\ the\ top\ k\ classes$ around na input image: It is the largest cumulative fraction of random directions which encounter a boundary adjacent to one of the k classes.

Figure 4: Decision Boundary Distances for Benign and Adversarial Examples



Source: Warren, Bo Li & Dawn Song. **Decision Boundary Analysis of Adversarial Examples**. Computer Science Division. University of California, Berkeley, International Conference on Learning Representations, 2018.

6. Results and Discussions

❑ 6.4 Individual Instances

- OPTMARGIN generate robust adversarial examples;

- Attacks applied to models: trained normally (1) and trained with PGD adversarial examples (2).

- Color represent the adjacent class to a specific match boundary.

- The black line is drawn according to the expected distance of an image sorted during region classification

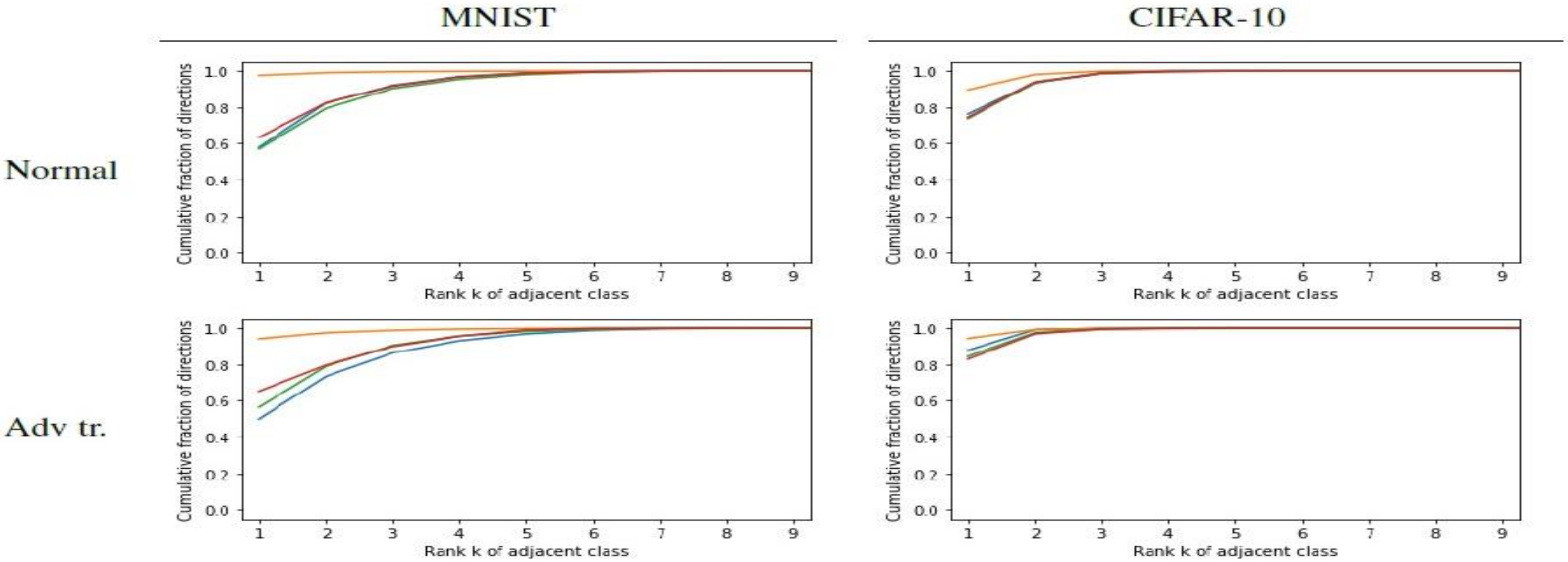
- Optimization attack generate plots different from benign examples because they seek to become closer as possible to boundary adjacent to the original class. This happens in a majority of directions.

- Reflects the advantages of region classification: a small perturbation in nearly every direction crosses the boundary to the original class.

Figure 5: Average Purity of Adjacent classes around Benign and Adversarial Examples

❑ 6.5 Adjacent Class Purity

- Adversarial Examples tend to have most directions lead to a boundary adjacent to a single class
- Curves lowered on the left indicate images surrounded by decision regions of multiple classes



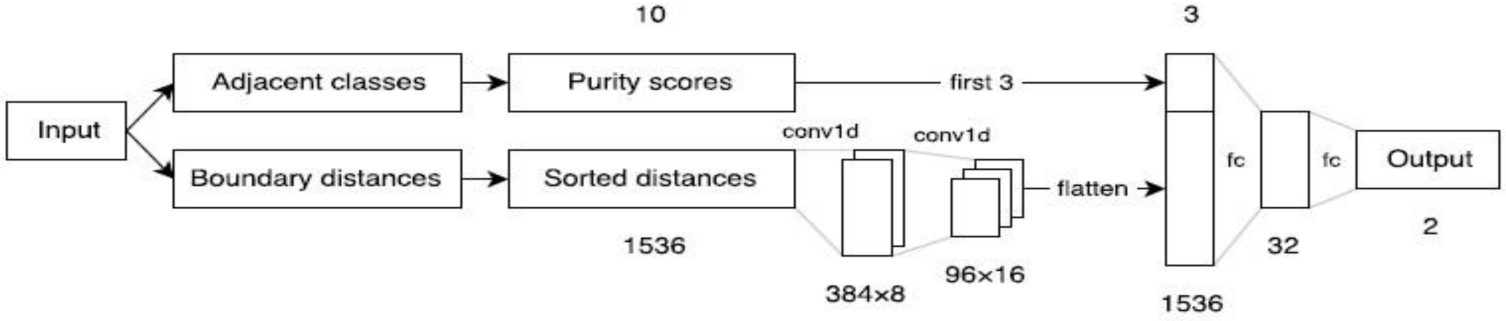
Source: Warren, Bo Li & Dawn Song. **Decision Boundary Analysis of Adversarial Examples**. Computer Science Division. University of California, Berkeley, International Conference on Learning Representations, 2018.

6. Results and Discussions

❑ 6.6 Decision Boundary Classification

- This approach is considering the distribution of distances to a decision boundary in a set of randomly chosen directions and distribution of adjacent classes.
- The information to make this classification possible is supplied by a neural network which deal with decision boundary information.
- The NN processes the distribution of boundary distances by applying 2 1-D convolutional layers to a sorted array of distances. So the results are filtered, it adds the first 3 purity scores and applies 2 fully connected layers -→ binary classification.

Figure 6: Architecture of our decision boundary classifier. Sizes are shown for our MNIST experiments. **[1]**



Source: Warren, Bo Li & Dawn Song. **Decision Boundary Analysis of Adversarial Examples**. Computer Science Division. University of California, Berkeley, International Conference on Learning Representations, 2018.

7. Conclusion

- Benefits of considering large neighborhoods around a given input in input space;
- Analysis by examining the decision boundaries around benign examples (originals) and around adversarial ones.
- Challenge to solve: How generate adversarial examples that better mimic benign examples' surrounding decision boundaries;
- Adversarial examples are close to the classification boundary and the hypercube around an adversarial intersects with the class region that has the true label of the adversarial example
- Proposal of a DNN region-based classifier which ensembles information in the hypercube around an example to predict its label.
- New possibilities can be explored in future works such as modifying regions, different methods to ensemble information in a region and structure attacks able to create robust adversarial examples.

8. Bibliographic References

- ❑ **[1]** Warren, Bo Li & Dawn Song. **Decision Boundary Analysis of Adversarial Examples**. Computer Science Division. University of California, Berkeley, International Conference on Learning Representations, 2018.

- ❑ **[2]** Xiaoyu Cao & Neil Zhenqiang Gong. **Mitigating Evasion Attacks to Deep Neural Networks via Region-based Classification.** Duke University, 31 December 2019.

- ❑ **[3]** Alex Krizhevsky, Vinod Nair, and Geoffrey Hinton**. CIFAR 10 Dataset**. Computer Science, University of Toronto.

Available at:https://www.cs.toronto.edu/~kriz/cifar.html