DEEL
DEpendable & Explainable Learning

# ACHIEVING ROBUSTNESS IN CLASSIFICATION WITH 1-LIPSCHITZ NN AND OPTIMAL TRANSPORT WITH HINGE REGULARIZATION

**Louis Béthune, Mathieu Serrurier, Franck Mamalet, Thibaut Boissin**, Alberto Gonźalez-Sanz, Jean-Michel Loubes, Eustasio del Barrio

IRIT Institut de Recherche en Informatique de Toulouse CNRS - INP - UT3 - UT1 - UT2J
IRT SAINT EXUPÉRY
INSTITUT de MATHEMATIQUES de TOULOUSE
ANITI ARTIFICIAL & NATURAL INTELLIGENCE TOULOUSE INSTITUTE
Université Fédérale Toulouse Midi-Pyrénées
IVADO
Institut intelligence et données

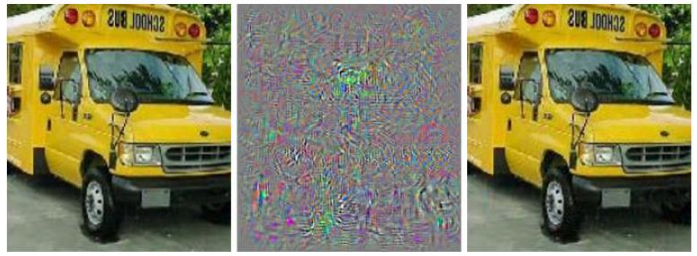
# PUBLICATIONS

**« Achieving robustness in classification using optimal transport with hinge regularization »,** Mathieu Serrurier, Franck Mamalet, Alberto González-Sanz, Thibaut Boissin, Jean-Michel Loubes, Eustasio del Barrio**,**
**CVPR'21 (https://arxiv.org/abs/2006.06520 )**

« **The Many Faces of 1-Lipschitz Neural Networks »,** Louis Béthune, Alberto González-Sanz, Franck Mamalet, Mathieu Serrurier
**(**https://arxiv.org/abs/2104.05097)

# CONTEXT

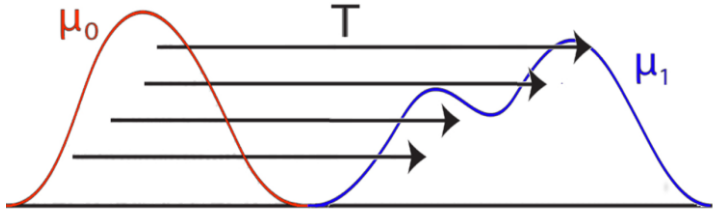
- Adversarial attacks of Neural Networks



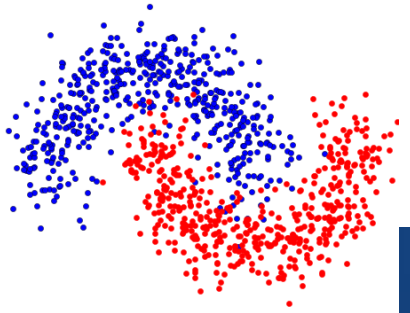
School bus Adversarial noise Ostrich

- Local Lipschitz robustness: $||f(x+\varepsilon) - f(x)|| \leq k.||\varepsilon||$

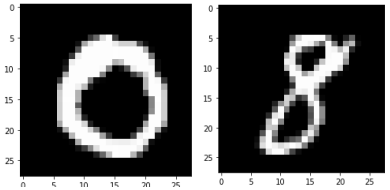
- Classification with Lipschitz Neural Networks and Wasserstein




- Building 1-Lipschitz network

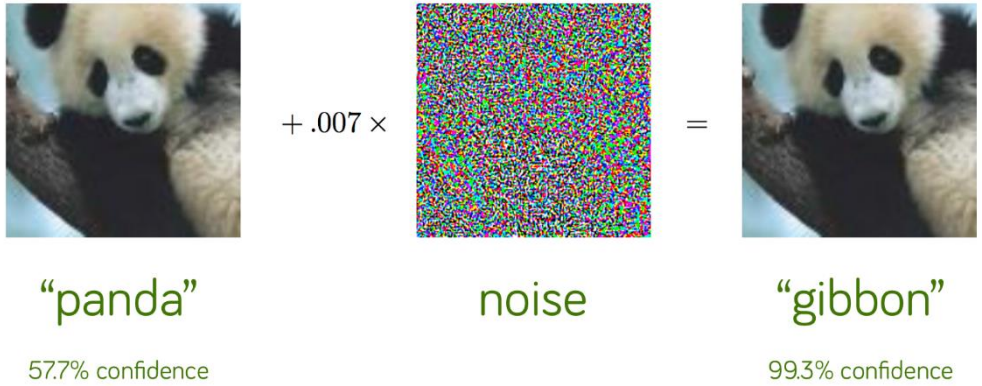


MNIST 0-8



Celeb-A (non)Moustache

# ADVERSARIAL ATTACK




- Adversarial example : closest example with the opposite class

$$adv(f, \mathbf{x}) = \underset{\mathbf{z} \in \Omega | sign(f(\mathbf{z})) = -sign(f(\mathbf{x}))}{argmin} \| \mathbf{x} - \mathbf{z} \|$$

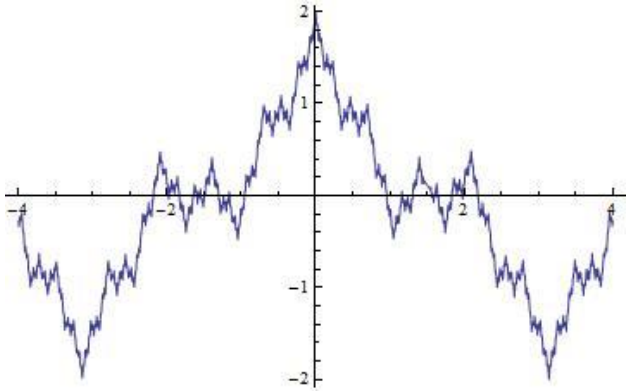
- Why neural networks are structurally weak to attack ?

# LIPSCHITZ FUNCTIONS

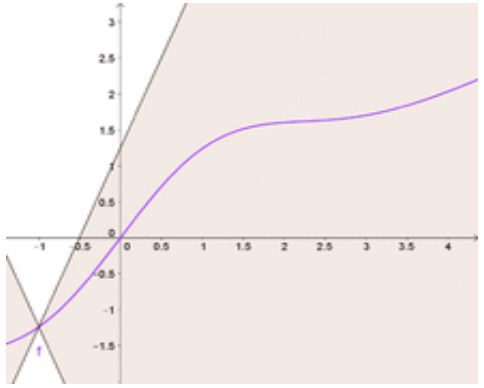
**L-Lipschitz function:** there exists **L** such that for all **x, y**

$$\|f(x) - f(y)\| \leq L\|x - y\|$$

Note: **f is continuous** and almost everywhere (for Lebesgue measure) differentiable:



$$\|\nabla_x f\| \leq L$$

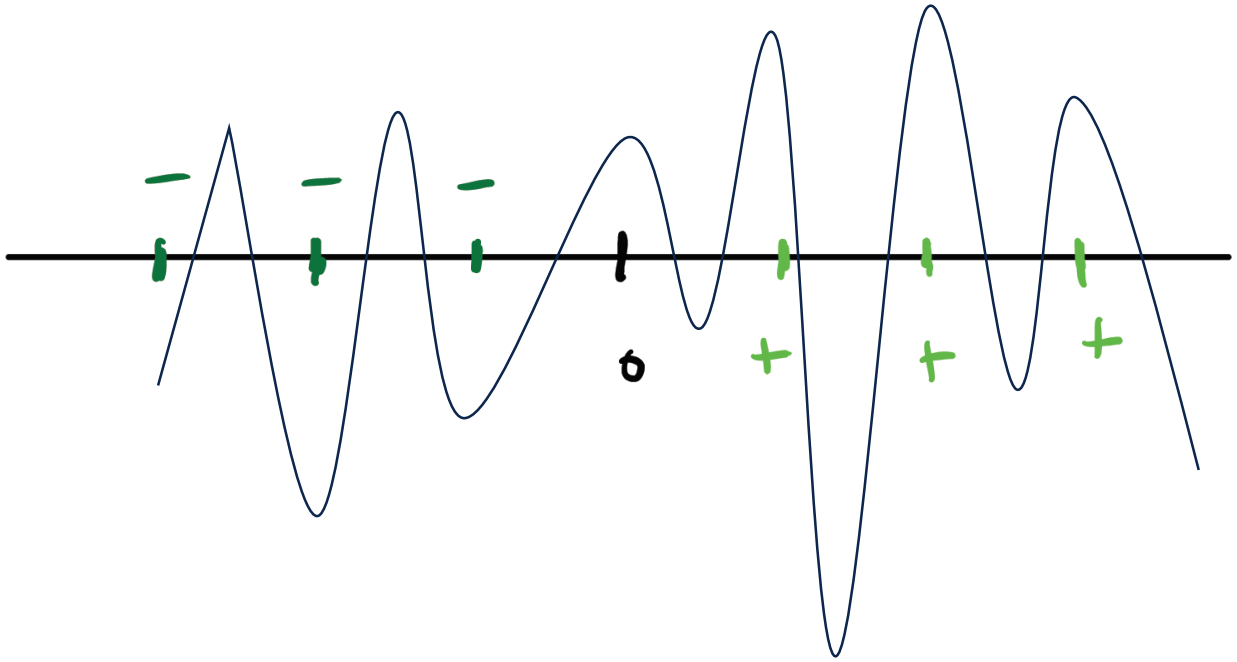


Weierstrass function: continuous but not Lipschitz

Lipschitz continuous function

# WHY ARE NN WEAK TO ATTACK ?

- High Lipschitz constant : small variation of inputs leads to high variation of the output  -> principle of adversarial attack




- Illustration

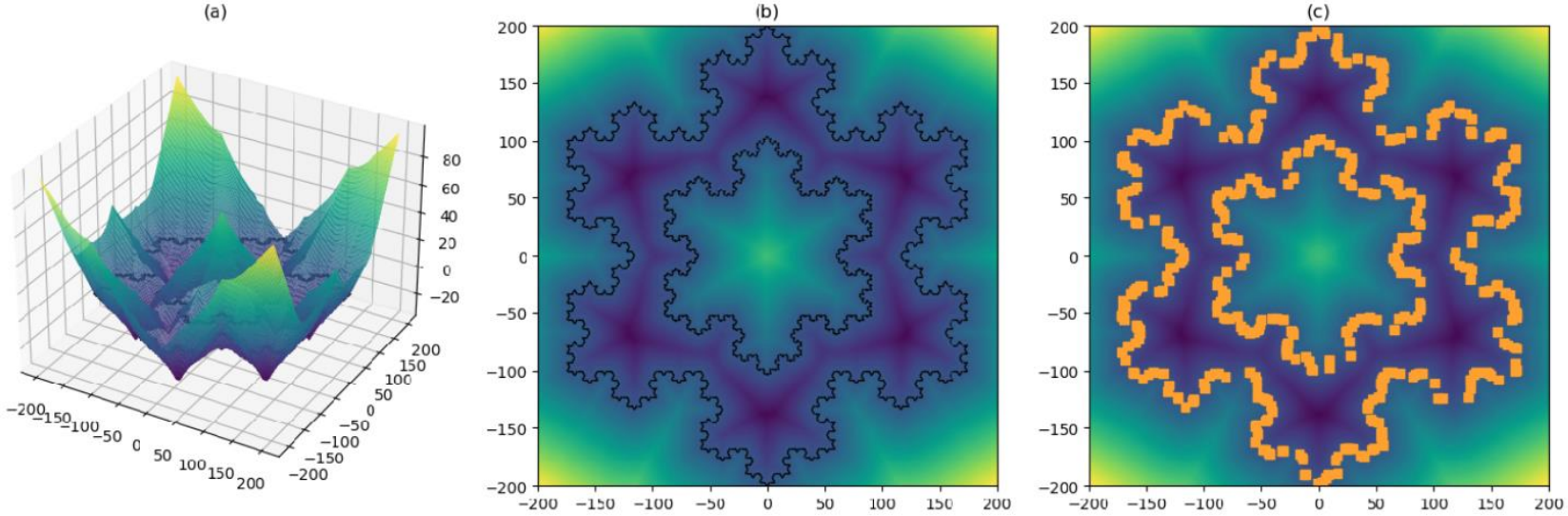
# 1-Lipschitz classifier existence and limitations of Cross-entropy

# LIPSCHITZ NETWORKS: AS CLASSIFIERS, THEY ARE GREAT

With **sign(f)** classifier: can imitate *any* other classifier having closed pre-images

- if a network $f$ is L-Lipschitz, $\frac{1}{L}f$ is 1-Lipschitz and has same decision frontier
- if classes are separable, a 1-Lipschitz neural can attain 100% test-accuracy
- even if classes are not separable, optimal Bayes classifier can be imitated

😀 They at least as powerful as unconstrained neural networks



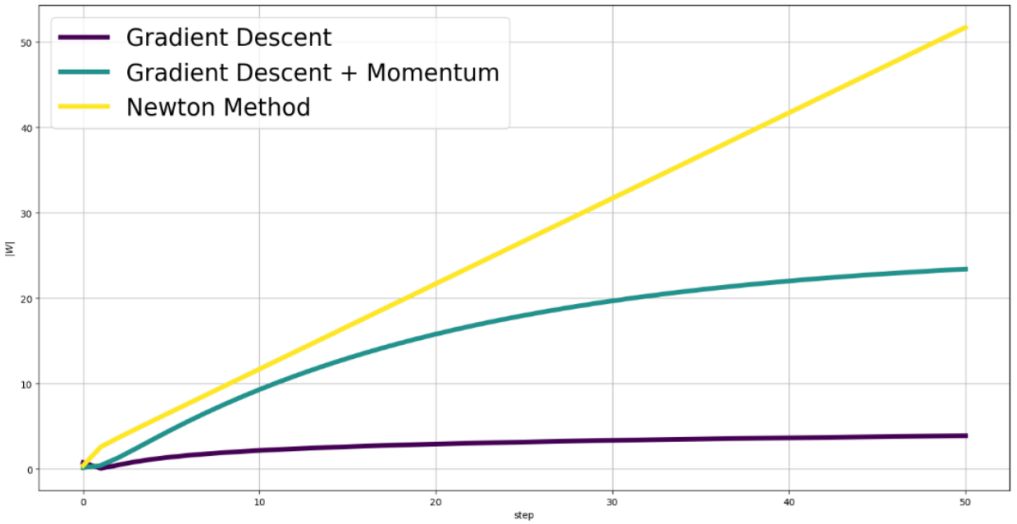
# BINARY CROSS ENTROPY MINIMUM

Probability with sigmoid function: $$\sigma(f(x)) = \frac{1}{1+\exp(-x)}$$

The loss: $$\mathcal{L}(f(x), y) = -\log \sigma(y f(x))$$

Unconstrained neural network: argmin ill-defined ☹

Lipschitz constant might grow indefinitely during optimization...




Training set : 2 points

(x_1, y_1) = (-1, -1)
(x_2, y_2) = (+1, +1)

Vanishing gradient slow down order 1 methods

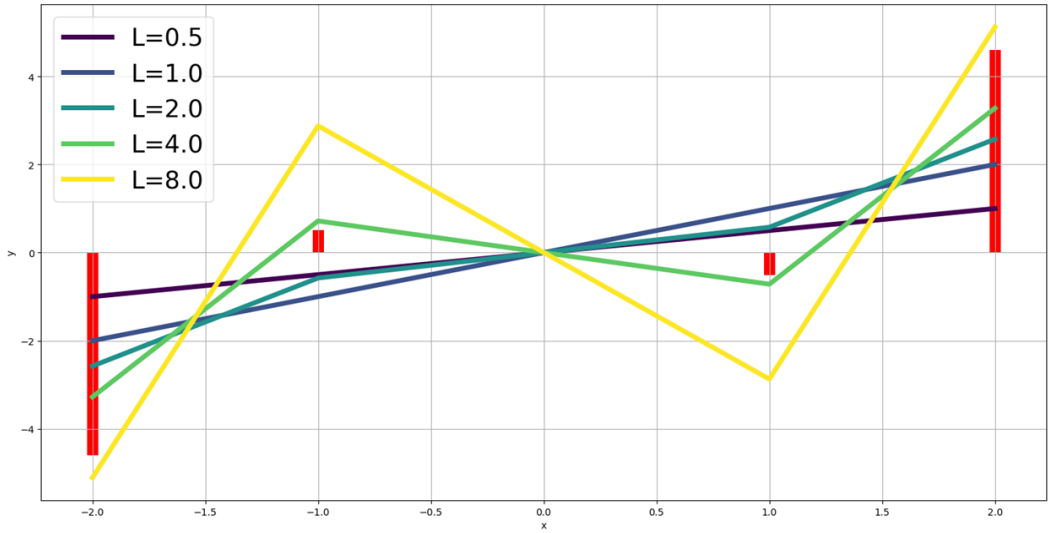
$$y = Wx + b$$

$$W \to \infty \qquad b = 0$$

Lipschitz neural network: argmin attained 😀

Analysis of its properties and generalization capabilities is easier

# LIPSCHITZ CONSTANT AND GENERALIZATION OF BCE

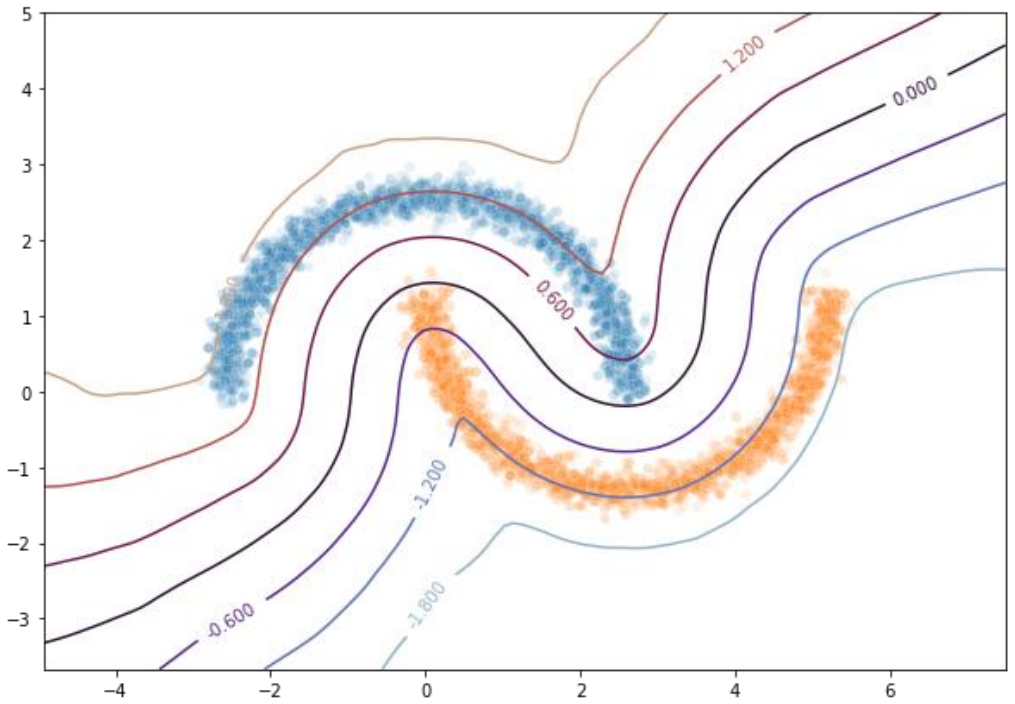



Points = {-2, -1, 1, 2}
Labels = {-1, 1, -1, 1}
Weights = {10, 1, 1, 10}

Small Lipschitz constant ⇔ Poor fitting of training set (underfitting possible)
Big Lipschitz constant ⇔ Tight fitting of training set (overfitting possible)

Lipschitz constant for BCE needs to be tuned just like Margin for Hinge !

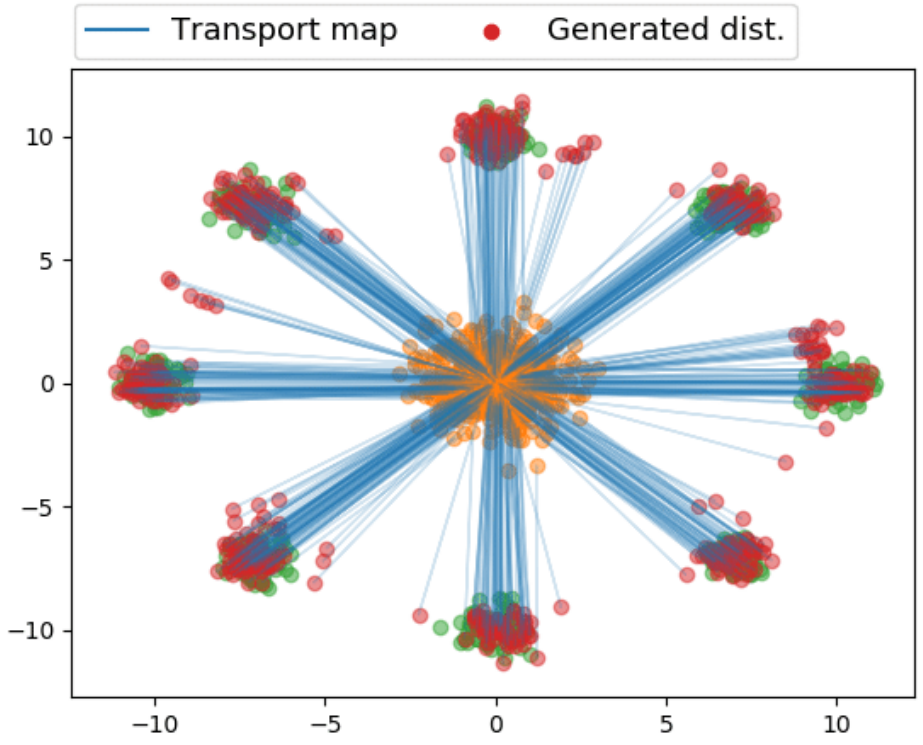
- Training 1-Lipshitz networks



# WHY CLASSIFYING WITH WASSERSTEIN

- Problem : cross entropy not suitable for 1-Lipschitz networks
- Wasserstein distance used as « discriminator » in WGAN
- Solution of the dual problem :
  - 1 Lipschitz function

$$\mathbb{P}_{x,z\sim\gamma^*}(f^*(x) - f^*(z) = ||x - z||) = 1$$



# WASSERSTEIN DISTANCE

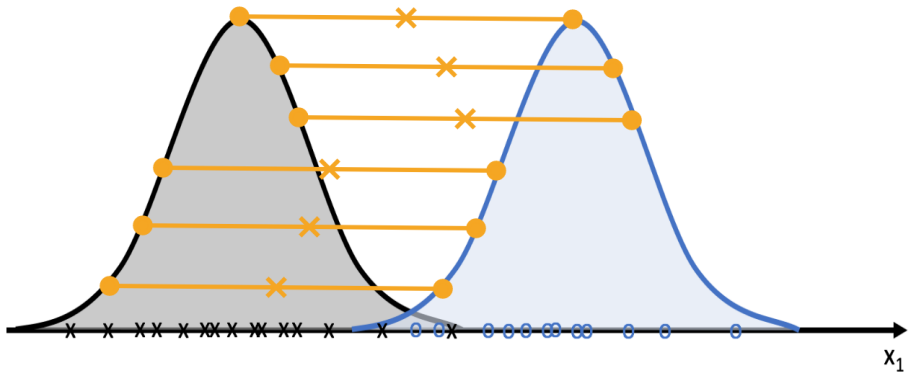
Primal formulation:

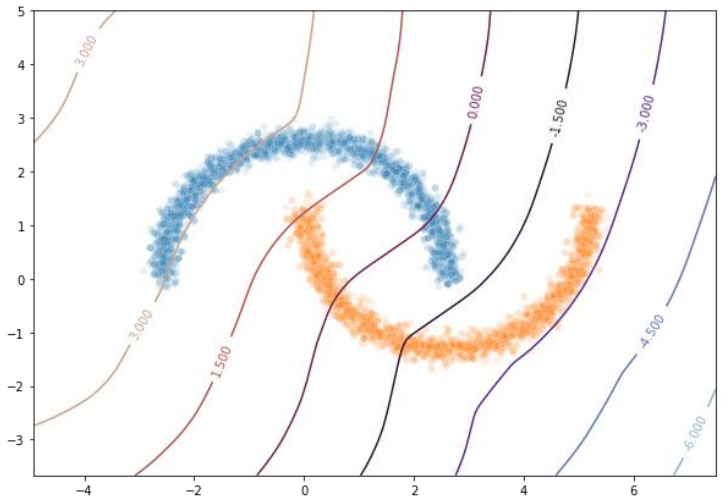
$$\mathcal{W}_p(\mu,\nu) = \inf_{\pi \in \Pi(\mu,\nu)} \mathop{\mathbb{E}}_{x,z\sim\pi} \| x - z \|^p$$

Dual formulation: **f** can be a represented by a 1-lipschitz neural network

$$\mathcal{W}_p(\mu,\nu) = \sup_{f \in Lip_1(\Omega)} \mathop{\mathbb{E}}_{X\sim\mu} [f(X)] - \mathop{\mathbb{E}}_{X\sim\nu} [f(X)]$$

Problem : Wasserstein classifier is weak



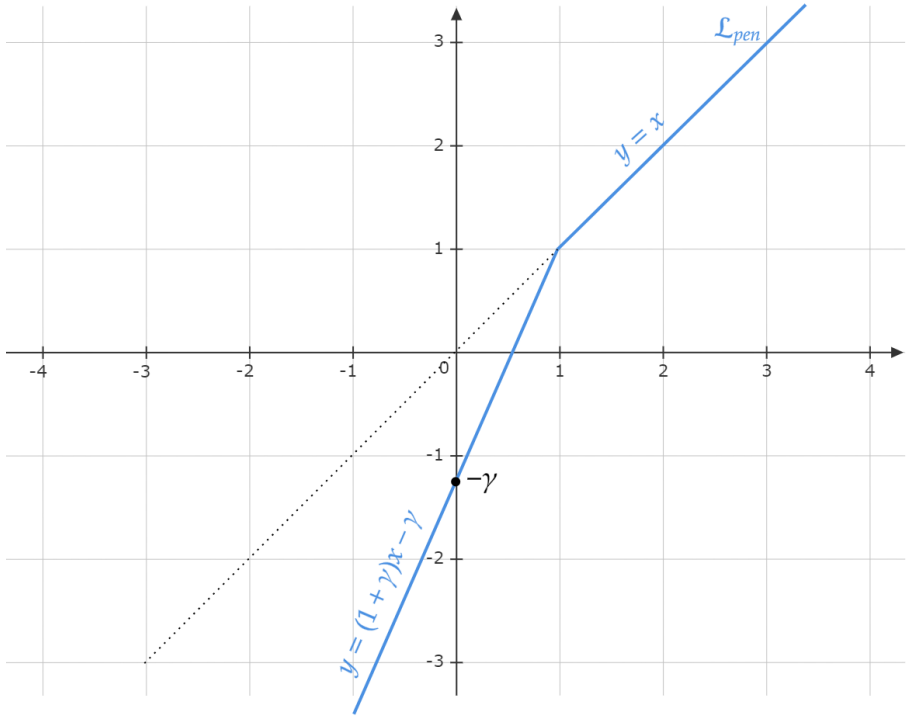



# OUR PROPOSITION: REGULARIZED WASSERSTEIN

- Idea : add an hinge regularization term to enforce classification

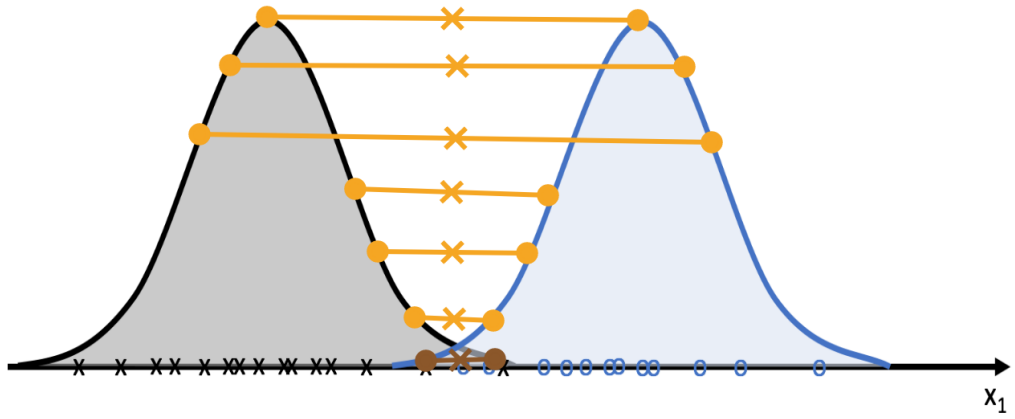
- Regularized dual problem

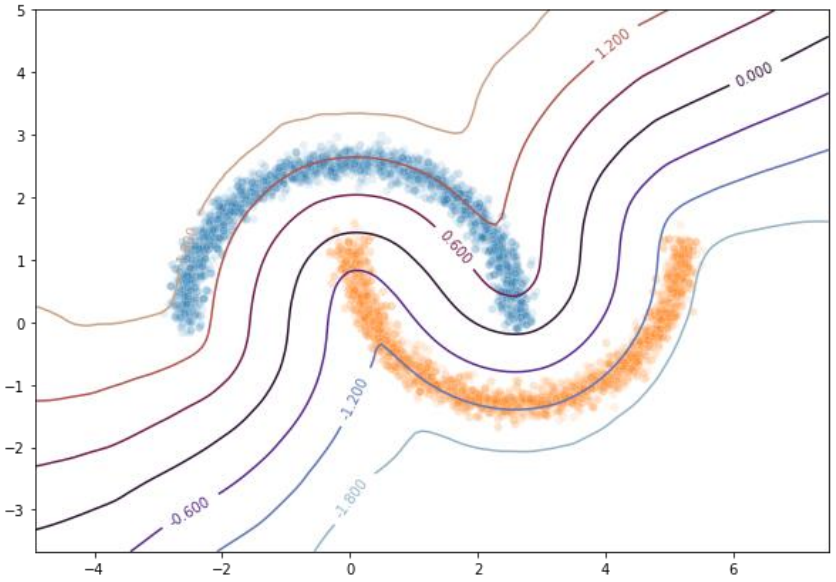
$$\mathcal{W}_{pen} = \sup_{f \in Lip_1(\Omega)} \mathop{\mathbb{E}}_{X,Y} \left( Y f(X) - \gamma (1 - Y f(X))_+ \right)$$



# OUR PROPOSITION: REGULARIZED WASSERSTEIN

- Trade-off accuracy robustness



# THEORETICAL PROPERTIES

- Hinge regularized Wasserstein is still an optimal transport problem

$$\sup_{f \in Lip_1(\Omega)} \mathop{\mathbb{E}}_{X,Y} (Yf(X) - \gamma(1 - Yf(X))_+) =$$

$$\inf_{\pi \in \Pi_\gamma^p(\mu,\nu)} \int_{\Omega \times \Omega} \| x - z \| \, d\pi - \pi_x(\Omega) - \pi_y(\Omega) + 1.$$

- Existence and uniqueness of the solution
- Gradient norm equals to 1 almost everywhere
- Risk bounds (almost OK)

# PRACTICAL PROPERTIES

- Provable robustness

$$\| x - adv(\hat{f}, x) \| \geq |\hat{f}(x)|$$

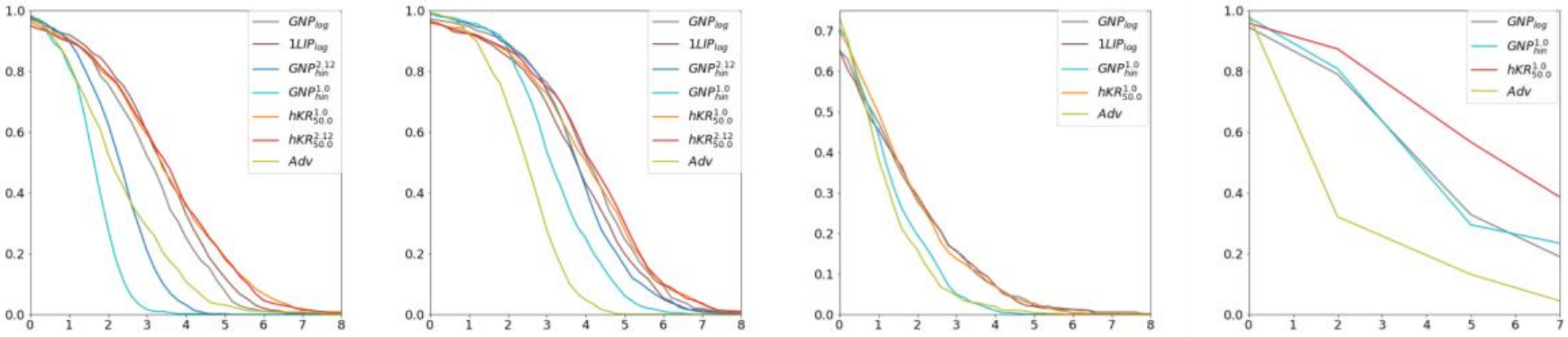
- In practice

$$\| x - adv(\hat{f}, x) \| \simeq |\hat{f}(x)|$$

- Structurally Robust to attack
- Adversarial attack :
  - Follow the transportation path
  - In the direction of : $\nabla_x f^*(\mathbf{x})$
  - Shows explicitelly what you have to change to change class

- Experimentations

# ROBUSTNESS EVALUATION

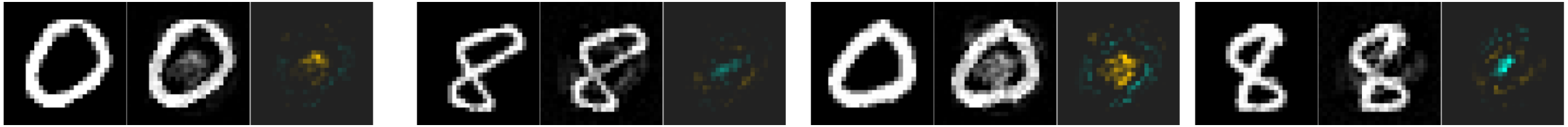



(a) MNIST dense (b) MNIST CNN (c) CIFAR (d) CelebA

Figure 4: Accuracy (Y-axis) w.r.t. of $l_2$ norm of $FGSM$, $l_2PGD$, deepfool and $l_2$ Carlini and Wagner combined attacks on 500 images of the test set

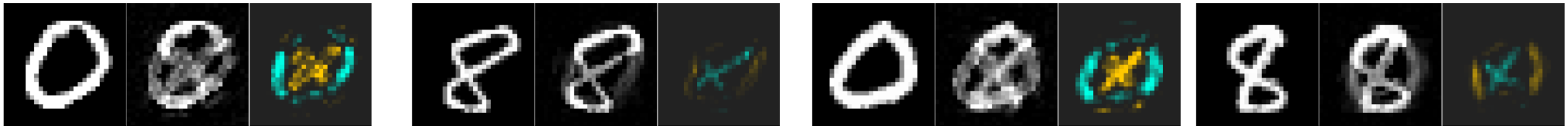
# MNIST 0-8 ADVERSARAL ATTACKS

- Qualitative comparison of adversarial examples (50/50 score)



MLP with binary cross-entropy

1-Lipschitz MLP with binary cross-entropy



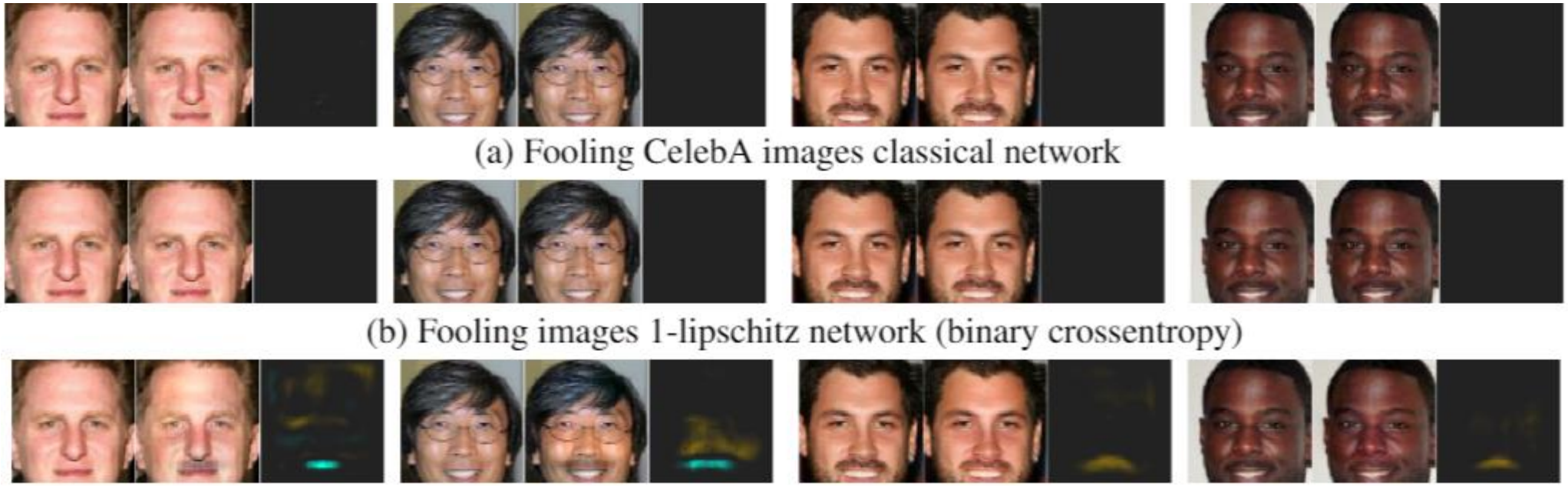
**1-Lipschitz MLP with regularized OT**

# CELEB-A MOUSTACHE ADVERSARAL ATTACKS

- Qualitative comparison of adversarial examples (50/50 score)

Lipschitz VGG with **regularized OT**

Without moustache  In-between with or without moustache



(a) Fooling CelebA images classical network

(b) Fooling images 1-lipschitz network (binary crossentropy)

# CELEB-A GLASS ADVERSARAL ATTACKS

- Qualitative comparison of adversarial examples

Lipschitz VGG with **regularized OT**

- Raw


- Classical approaches


- Our approach

- Building 1-Lipschitz neural networks

# LIPSCHITZ CONSTANT NEURAL NETWORK

- Very hard to evaluate accurately (np-hard)
- Multilayer perceptron

$$f(x) = \phi_k(W_k.(\phi_{k-1}(W_{k-1} \dots \phi_1(W_1.x)))$$

- Lipschitz constant upper-bound

$$L(f) \leq L(\phi_k) * L(W_k) * L(\phi_{k-1}) * L(W_{k-1}) * \dots * L(\phi_1) * L(W_1.x)$$

- Activation function are usualy 1-Lischitz
- If all linear layer are 1-Lischitz, the network is 1-Lischitz (but the constant can be smaller than one)

# 1-LIPSCHITZ NEURAL NETWORKS

- Principles : all the layers have to be 1-lipschitz
- Dense Layer with weights W

$$L(W) = ||W|| \leq ||W||_F \leq \max_{ij}(|W_{ij}|) * \sqrt{nm}$$

- Constraining Lipschitz constant
  - WGAN : weight clipping (last term of the equation)
  - Weight normalization with Frobenius norm $||W||_F$
  - Spectral normalization with spectral norm $||W||$

# 1-LIPSCHITZ OTHERS LAYERS

- ReLU, sigmoid, Tanh: 1-lipschitz by nature
- LeakyReLU : 1-lipschitz if alpha<1.0
- PReLU (Parametric Rectified Linear Unit): need a constraint on scaling factor (=>PReLUlip)

- BatchNormalization: Not lipschitz
- Dropout: Not Lipschitz

# GRADIENT PRESERVING

- Property of the optimal f for the dual wasserstein problem :

$$||\nabla f^*|| = 1 \text{ almost everywhere on the support of } \gamma^*.$$

- i.e. f* is piecewise linear with slope equal to 1 almost every where
- How to achieve that :
  - Use gradient preserving activation function
    - Max min
    - Group sort
    - Full-sort
  - Orthonormalize the eigen vectors of the Weights of each layers (all singular values equals to one)
    - Bjork algorithm during inference
    - Can be time consuming
    - See also previous resentation by Bachour, Malgouyres,Mamalet

# DEEL-LIP LIBRARY

- Library DEEL-LIP based on Tensorflow (Pytorch is coming) for building k-Lipschitz Neural Networks.
- Easy to use, implementing 1-lipschitz and Gradient Norm Preserving (GNP) layers, with GroupSort activation functions.
- hKR multi-class loss implementation
- additional feature to export linear layers to standard layers of TF after learning. (github.com/deel-ai/deel-lip)

SCAN ME

# CONCLUSIONS

- 1-Lipschitz NN are for classification and provide certification robustness radius
- Training 1-Lipschitz network with optimal transport loss
  - New interpretation of classification problem
  - Improve robustness structurally
  - Interpretable
- Deep lip : accessible and optimized librairy to train and use 1-Lipschitz networks
- Future works
  - Outlier detection
  - Semi-supervised/ agnostic learning