Stronger and Faster Wasserstein Adversarial Attacks

Kaiwen Wu
kaiwen.wu@uwaterloo.ca

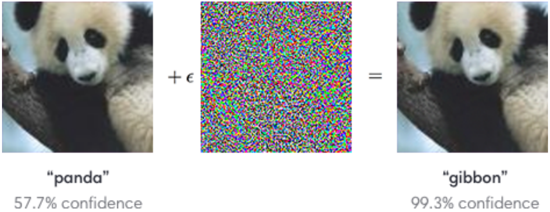
Joint work with Allen Wang and Yaoliang Yu

UNIVERSITY OF WATERLOO

VECTOR INSTITUTE | INSTITUT VECTEUR

Adversarial Examples

- Adversarial examples:



(Goodfellow et al. 2015)

- Generating adversarial examples:

$$\begin{aligned} \underset{\mathbf{x}_{adv}}{\text{maximize}} \quad & \ell(f(\mathbf{x}_{adv}), y) \\ \text{subject to} \quad & \mathbf{x}_{adv} \approx \mathbf{x} \end{aligned}$$

How "Similar" Is Similar?

How to quantify $\mathbf{x}_{adv} \approx \mathbf{x}$?

- $\|\mathbf{x} - \mathbf{x}_{adv}\|_p \leq \epsilon$ (Szegedy et al. 2014)
- point-wise function (Laidlaw et al. 2019)
- geometric transformation (Engstrom et al. 2019)
- Wasserstein distance (Wong et al. 2019)
- ...

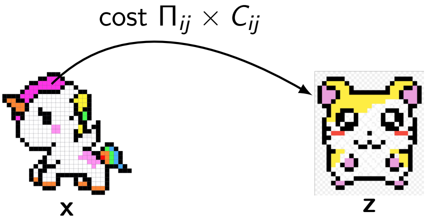
Our contributions

- stronger and faster Wasserstein adversarial attacks
- higher robust accuracy using adversarial training

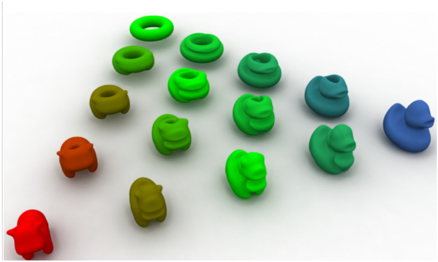
What is Wasserstein Distance?

$$\mathcal{W}(\mathbf{x}, \mathbf{z}) = \min_{\Pi \geq 0} \langle \Pi, C \rangle \quad \text{s.t. } \Pi \mathbf{1} = \mathbf{x}, \Pi^\top \mathbf{1} = \mathbf{z}$$

- $\mathbf{x} \in \mathbb{R}^n$ and $\mathbf{z} \in \mathbb{R}^n$: input images
- $\Pi \in \mathbb{R}^{n \times n}$: transportation matrix
- $C \in \mathbb{R}^{n \times n}$: transportation cost



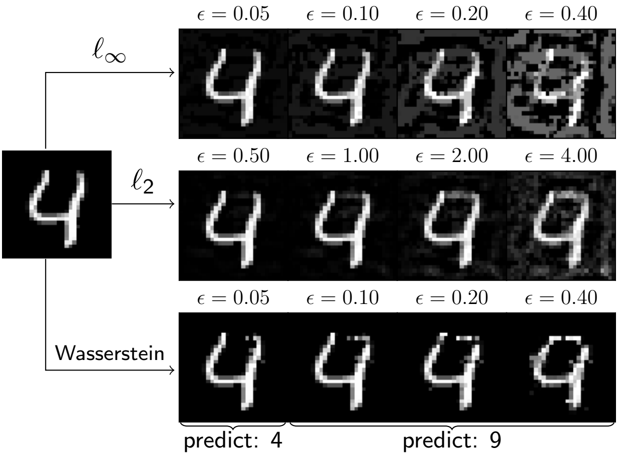
Applications across Different Domains



(Arjovsky et al. 2017; Rabin et al. 2014; Solomon et al. 2015)

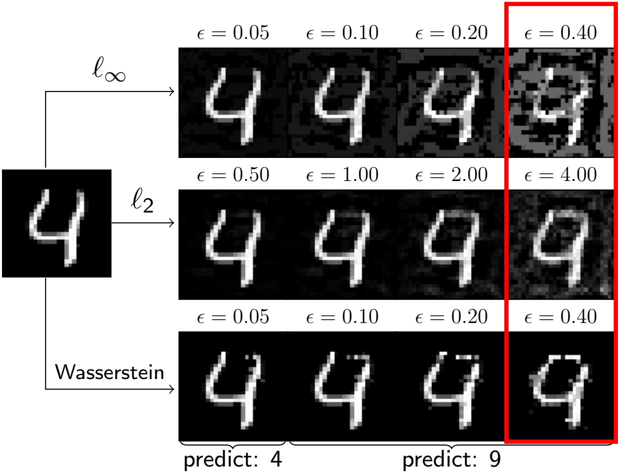
Why Wasserstein Distance?

- Captures geometry in image space, *e.g.* translation, rotation



Why Wasserstein Distance?

- Captures geometry in image space, *e.g.* translation, rotation



Computing Wasserstein Adversarial Examples

Search for adversarial examples:

$$\begin{aligned} \underset{\mathbf{x}_{adv}}{\text{maximize}} \;\; & \ell(\mathbf{x}_{adv}) \\ \text{subject to} \;\; & \mathcal{W}(\mathbf{x}, \mathbf{x}_{adv}) \leq \epsilon \end{aligned}$$

Computing Wasserstein Adversarial Examples

Search for adversarial examples:

$$\begin{aligned} \underset{\mathbf{x}_{adv}}{\text{maximize}} \;\; & \ell(\mathbf{x}_{adv}) \\ \text{subject to} \;\; & \mathcal{W}(\mathbf{x}, \mathbf{x}_{adv}) \leq \epsilon \end{aligned}$$

Alternatively, search for transportation matrix:

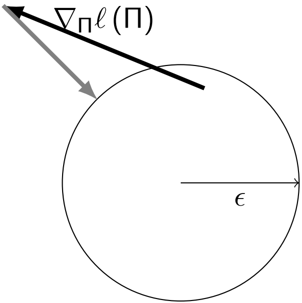
$$\begin{aligned} \underset{\Pi \geq 0}{\text{maximize}} \;\; & \ell(\Pi^\top \mathbf{1}) \\ \text{subject to} \;\; & \Pi \mathbf{1} = \mathbf{x}, \;\; \langle \Pi, C \rangle \leq \epsilon \end{aligned}$$

Then, recover adversarial examples:

$$\mathbf{x}_{adv} = \Pi^\top \mathbf{1}$$

Optimization in Transportation Matrix

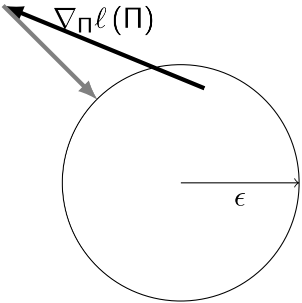
Optimization in Transportation Matrix



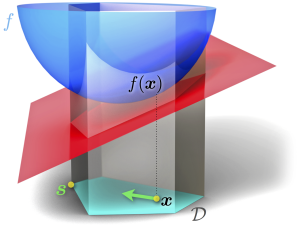
(a) projected gradient

$$\underset{\Pi \geq 0}{\text{minimize}} \quad \frac{1}{2}\|\Pi - G\|_{\mathrm{F}}^2$$
$$\text{subject to } \ \Pi\mathbf{1} = \mathbf{x}, \langle \Pi, \ C \rangle \leq \epsilon$$

Optimization in Transportation Matrix



(a) projected gradient

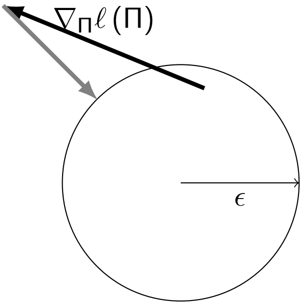


(b) Frank-Wolfe (Jaggi 2011)

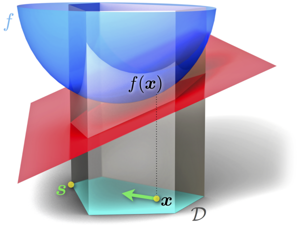
$$\underset{\Pi \geq 0}{\text{minimize}} \quad \frac{1}{2}\|\Pi - G\|_F^2$$
$$\text{subject to} \;\; \Pi\mathbf{1} = \mathbf{x}, \langle \Pi, \; C\rangle \leq \epsilon$$

$$\underset{\Pi \geq 0}{\text{minimize}} \quad \langle \Pi, H\rangle$$
$$\text{subject to} \;\; \Pi\mathbf{1} = \mathbf{x}, \; \langle \Pi, C\rangle \leq \epsilon$$

Optimization in Transportation Matrix



(a) projected gradient



(b) Frank-Wolfe (Jaggi 2011)

$$\underset{\Pi \geq 0}{\text{minimize}} \quad \frac{1}{2}\|\Pi - G\|_{\mathrm{F}}^2$$
$$\text{subject to } \Pi\mathbf{1} = \mathbf{x}, \langle \Pi, C\rangle \leq \epsilon$$

$$\underset{\Pi \geq 0}{\text{minimize}} \quad \langle \Pi, H\rangle$$
$$\text{subject to } \Pi\mathbf{1} = \mathbf{x}, \langle \Pi, C\rangle \leq \epsilon$$

For n dimensional images, Π has n^2 variables...

Solve Projection in PGD

$$\begin{aligned} \underset{\Pi \geq 0}{\text{minimize}} \quad & \tfrac{1}{2}\|\Pi - G\|_{\mathrm{F}}^2 \\ \text{subject to} \quad & \Pi \mathbf{1} = \mathbf{x},\ \langle \Pi, C \rangle \leq \epsilon \end{aligned}$$

Solve Projection in PGD

$$\begin{aligned} \underset{\Pi \geq 0}{\text{minimize}} \quad & \frac{1}{2}\|\Pi - G\|_{\mathrm{F}}^2 \\ \text{subject to} \quad & \Pi \mathbf{1} = \mathbf{x},\ \langle \Pi, C \rangle \leq \epsilon \end{aligned}$$

The Lagrange dual can be simplified as a univariate problem

$$\underset{\lambda \geq 0}{\text{maximize}} \quad g(\lambda)$$

Solve Projection in PGD

$$\begin{aligned} \underset{\Pi \geq 0}{\text{minimize}} \quad & \frac{1}{2}\|\Pi - G\|_{\mathrm{F}}^2 \\ \text{subject to} \quad & \Pi \mathbf{1} = \mathbf{x}, \; \langle \Pi, C \rangle \leq \epsilon \end{aligned}$$

The Lagrange dual can be simplified as a univariate problem

$$\underset{\lambda \geq 0}{\text{maximize}} \quad g(\lambda)$$

- No closed-form expression...

Solve Projection in PGD

$$\begin{aligned} \underset{\Pi \geq 0}{\text{minimize}} \quad & \tfrac{1}{2}\|\Pi - G\|_{\mathrm{F}}^2 \\ \text{subject to} \quad & \Pi \mathbf{1} = \mathbf{x}, \ \langle \Pi, C \rangle \leq \epsilon \end{aligned}$$

The Lagrange dual can be simplified as a univariate problem

$$\underset{\lambda \geq 0}{\text{maximize}} \quad g(\lambda)$$

- No closed-form expression...
- But $g'(\lambda)$ can be evaluated in $O(n^2 \log n)$ time

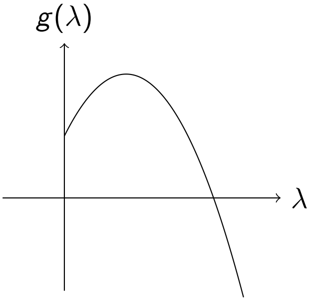
Proposition

$$0 \leq \lambda^\star \leq \frac{2\,\|\mathrm{vec}(G)\|_\infty + \|\mathbf{x}\|_\infty}{\min_{i \neq j}\{C_{ij}\}}$$

Bisection on the Dual

$$\underset{\lambda \geq 0}{\text{maximize}} \quad g(\lambda)$$

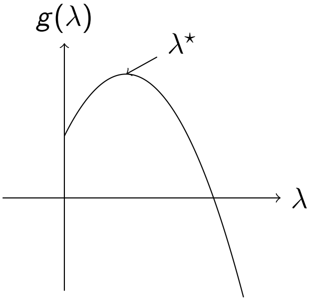
- Converge to high precision ≤ 20 iterations in practice.



Bisection on the Dual

$$\underset{\lambda \geq 0}{\text{maximize}} \quad g(\lambda)$$

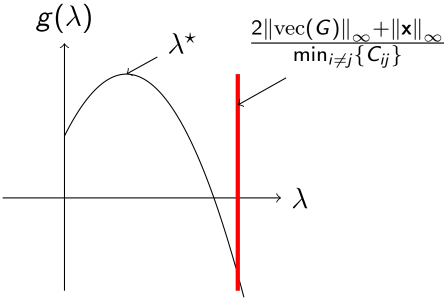
- Converge to high precision ≤ 20 iterations in practice.



Bisection on the Dual

$$\underset{\lambda \geq 0}{\text{maximize}} \quad g(\lambda)$$

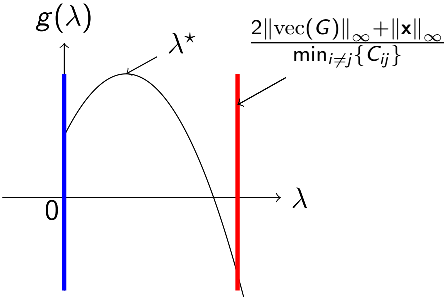
- Converge to high precision ≤ 20 iterations in practice.



Bisection on the Dual

$$\underset{\lambda \geq 0}{\text{maximize}} \quad g(\lambda)$$

- Converge to high precision ≤ 20 iterations in practice.



Solve Linear Minimization in Frank-Wolfe

$$\begin{aligned} \underset{\Pi \geq 0}{\text{minimize}} \quad & \langle \Pi, H \rangle \\ \text{subject to} \quad & \Pi \mathbf{1} = \mathbf{x}, \ \langle \Pi, C \rangle \leq \epsilon \end{aligned}$$

Solve Linear Minimization in Frank-Wolfe

$$\begin{aligned} \underset{\Pi \geq 0}{\text{minimize}} \quad & \langle \Pi, H \rangle \\ \text{subject to} \quad & \Pi \mathbf{1} = \mathbf{x}, \ \langle \Pi, C \rangle \leq \epsilon \end{aligned}$$

The Lagrange dual can be simplified as a univariate problem

$$\underset{\lambda \geq 0}{\text{maximize}} \, g(\lambda)$$

Solve Linear Minimization in Frank-Wolfe

$$\begin{aligned} \underset{\Pi \geq 0}{\text{minimize}} \quad & \langle \Pi, H \rangle \\ \text{subject to} \quad & \Pi \mathbf{1} = \mathbf{x}, \ \langle \Pi, C \rangle \leq \epsilon \end{aligned}$$

The Lagrange dual can be simplified as a univariate problem

$$\underset{\lambda \geq 0}{\text{maximize}}\, g(\lambda)$$

- Bound on the optimum: $0 \leq \lambda^\star \leq \frac{2\|\text{vec}(H)\|_\infty}{\min_{i \neq j}\{C_{ij}\}}$

Solve Linear Minimization in Frank-Wolfe

$$\underset{\Pi \geq 0}{\text{minimize}} \quad \langle \Pi, H \rangle$$
$$\text{subject to} \;\; \Pi \mathbf{1} = \mathbf{x}, \;\; \langle \Pi, C \rangle \leq \epsilon$$

The Lagrange dual can be simplified as a univariate problem

$$\underset{\lambda \geq 0}{\text{maximize}} \, g(\lambda)$$

- Bound on the optimum: $0 \leq \lambda^\star \leq \frac{2\|\mathrm{vec}(H)\|_\infty}{\min_{i \neq j}\{C_{ij}\}}$
- Does not work...
  - difficult to recover primal solution
  - severe numerical instability

Entropic Regularization

$$\underset{\Pi \geq 0}{\text{minimize}} \quad \langle \Pi, H \rangle + \boxed{\gamma \sum_{i=1}^{n} \sum_{j=1}^{n} \Pi_{ij} \log \Pi_{ij}}$$

$$\text{subject to} \;\; \Pi \mathbf{1} = \mathbf{x}, \;\; \langle \Pi, C \rangle \leq \epsilon$$

Entropic Regularization

$$\underset{\Pi \geq 0}{\text{minimize}} \quad \langle \Pi, H \rangle + \boxed{\gamma \sum_{i=1}^{n} \sum_{j=1}^{n} \Pi_{ij} \log \Pi_{ij}}$$

$$\text{subject to} \;\; \Pi \mathbf{1} = \mathbf{x}, \;\; \langle \Pi, C \rangle \leq \epsilon$$

- Closed-form expression to recover primal solution

Entropic Regularization

$$\underset{\Pi \geq 0}{\text{minimize}} \quad \langle \Pi, H \rangle + \boxed{\gamma \sum_{i=1}^{n} \sum_{j=1}^{n} \Pi_{ij} \log \Pi_{ij}}$$

$$\text{subject to} \;\; \Pi \mathbf{1} = \mathbf{x}, \;\; \langle \Pi, C \rangle \leq \epsilon$$

- Closed-form expression to recover primal solution
- Entropic regularization introduces approximation error

Entropic Regularization

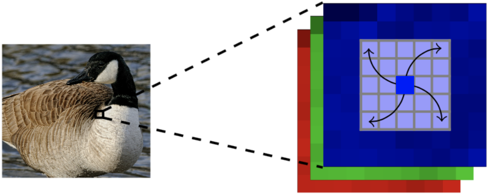
$$\underset{\Pi \geq 0}{\text{minimize}} \quad \langle \Pi, H \rangle + \boxed{\gamma \sum_{i=1}^{n} \sum_{j=1}^{n} \Pi_{ij} \log \Pi_{ij}}$$

$$\text{subject to} \;\; \Pi \mathbf{1} = \mathbf{x}, \;\; \langle \Pi, C \rangle \leq \epsilon$$

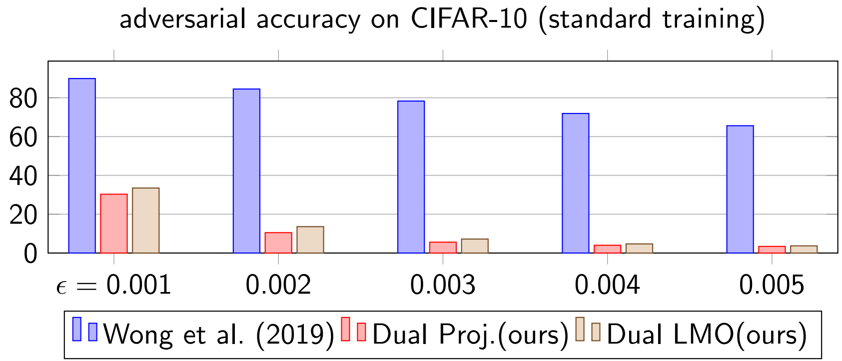
- Closed-form expression to recover primal solution
- Entropic regularization introduces approximation error
- But the approximation error is guaranteed to be small

Exploit Sparsity

- Local transportation constraint (Wong et al. 2019)
  $\Rightarrow$ structured sparsity in Π
- Per iteration cost is reduced to $O(n)$ by exploiting sparsity

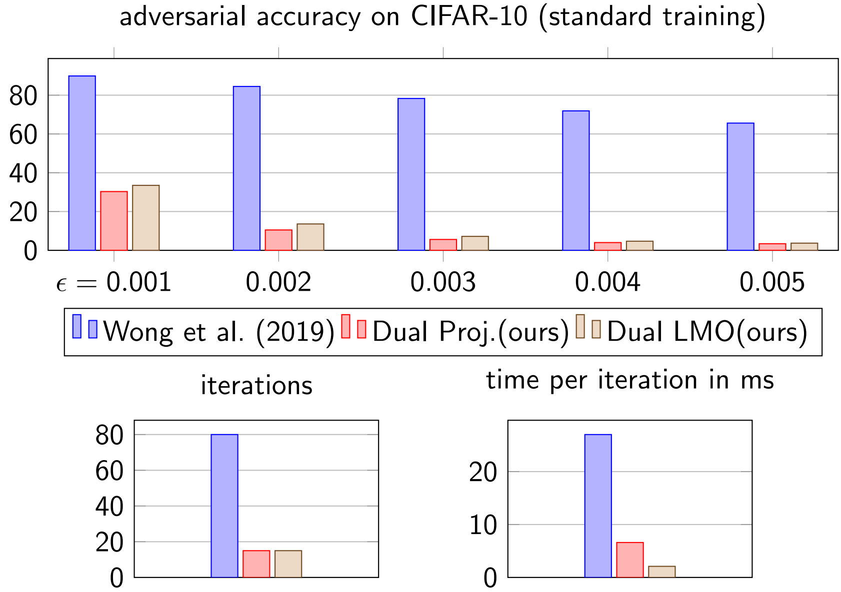


Comparison



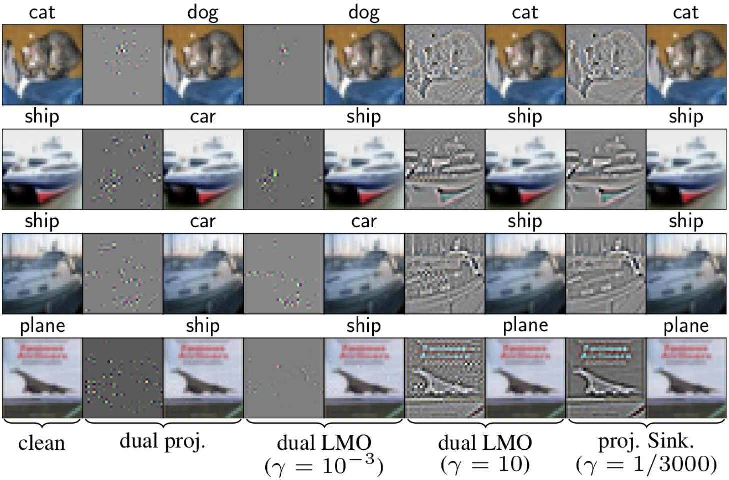
adversarial accuracy on CIFAR-10 (standard training)
80
60
40
20
0
$\epsilon = 0.001$
0.002
0.003
0.004
0.005
Wong et al. (2019)
Dual Proj.(ours)
Dual LMO(ours)

Comparison

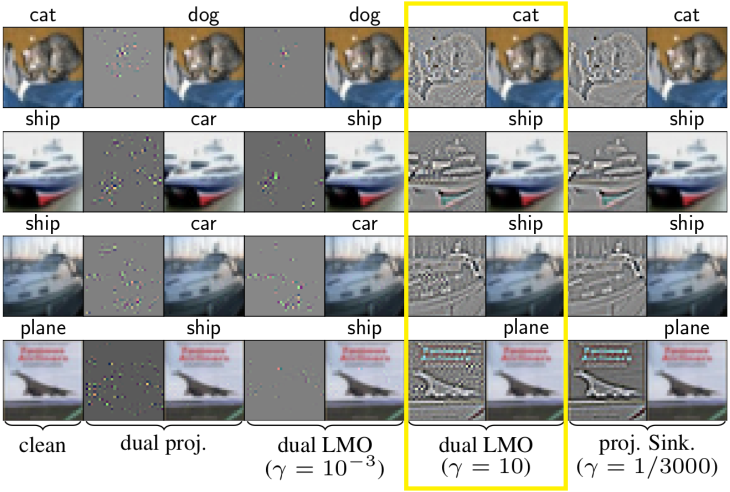


adversarial accuracy on CIFAR-10 (standard training)
80
60
40
20
0
$\epsilon = 0.001$
0.002
0.003
0.004
0.005
Wong et al. (2019)
Dual Proj.(ours)
Dual LMO(ours)
iterations
80
60
40
20
0
time per iteration in ms
20
10
0

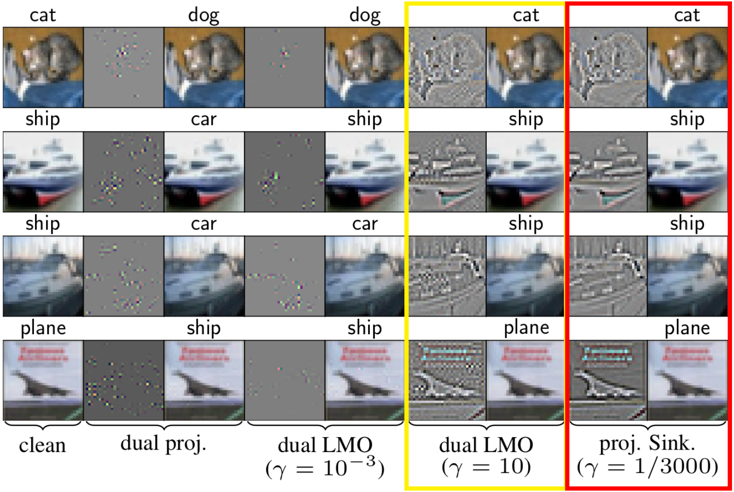
Entropic Regularization Reflects Shapes



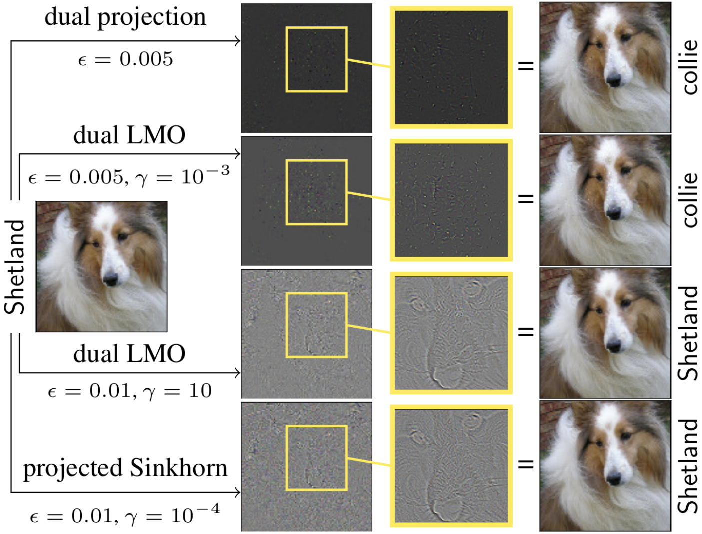
Entropic Regularization Reflects Shapes



Entropic Regularization Reflects Shapes

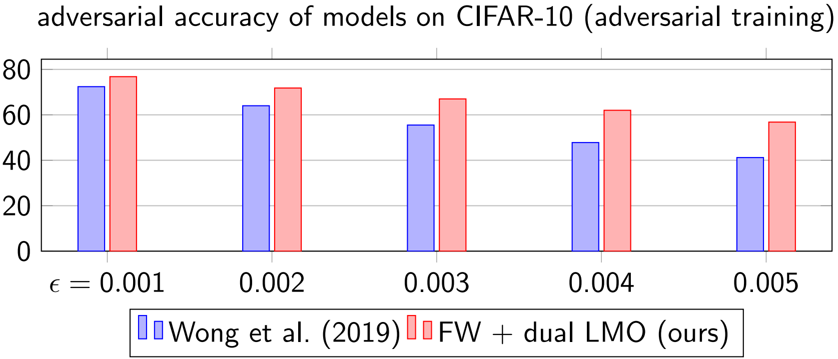


Scalable to High Dimensional Data



Improved Adversarial Training

- Stronger attacks improve adversarial training!



Summary

- PGD and Frank-Wolfe complement each other nicely
- PGD with dual projection is the strongest attack
- Frank-Wolfe with dual LMO is the fastest attack
- Improved adversarial training
- Applicable to any Wasserstein constrained optimization