Analyzing the Robustness of Nearest Neighbors to Adversarial Examples

Kamalika Chaudhuri
University of California, San Diego

Based on joint work with

Yizhen Wang and Somesh Jha

Adversarial Learning

How to design classifiers that are robust to adversarial examples?

Classification

Traffic sign images

Which sign

STOP

Speed Limit

YIELD

How to do Classification

$$\begin{bmatrix} 0.523 \\ 0.214 \\ \vdots \\ 0.702 \\ \vdots \\ 0 \\ 0.995 \end{bmatrix}, \quad 22$$

STOP

Image,
Annotation

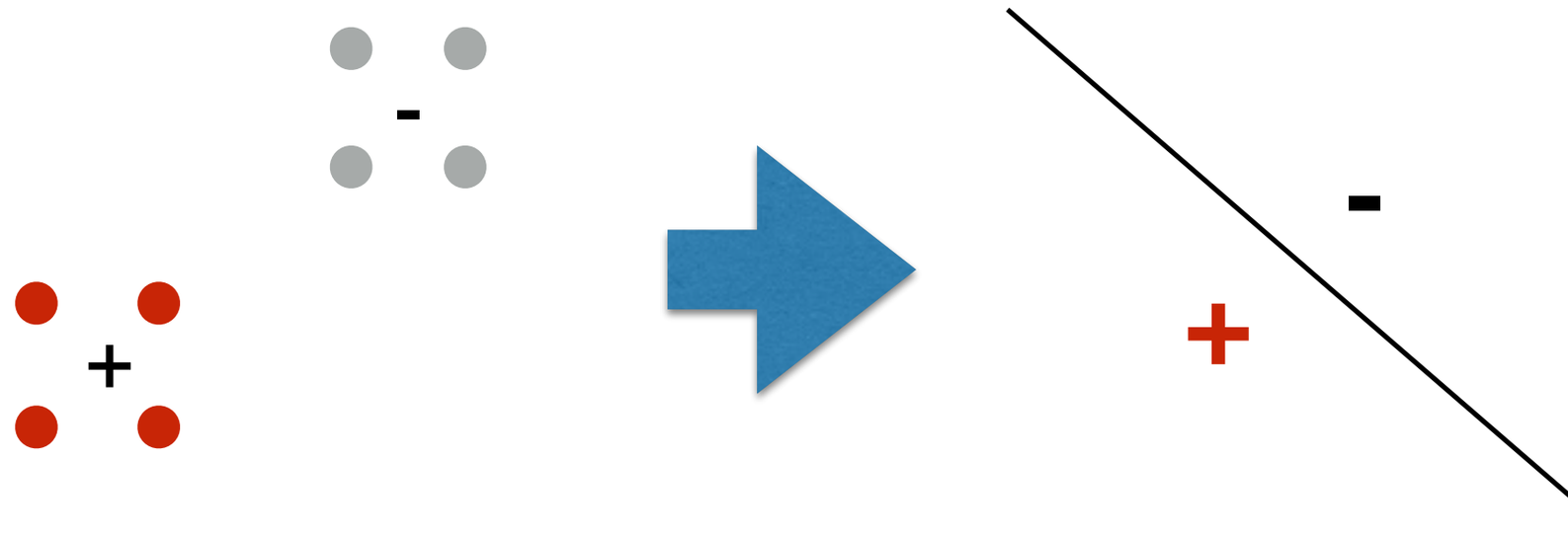
Feature
Vector

(X)

Label

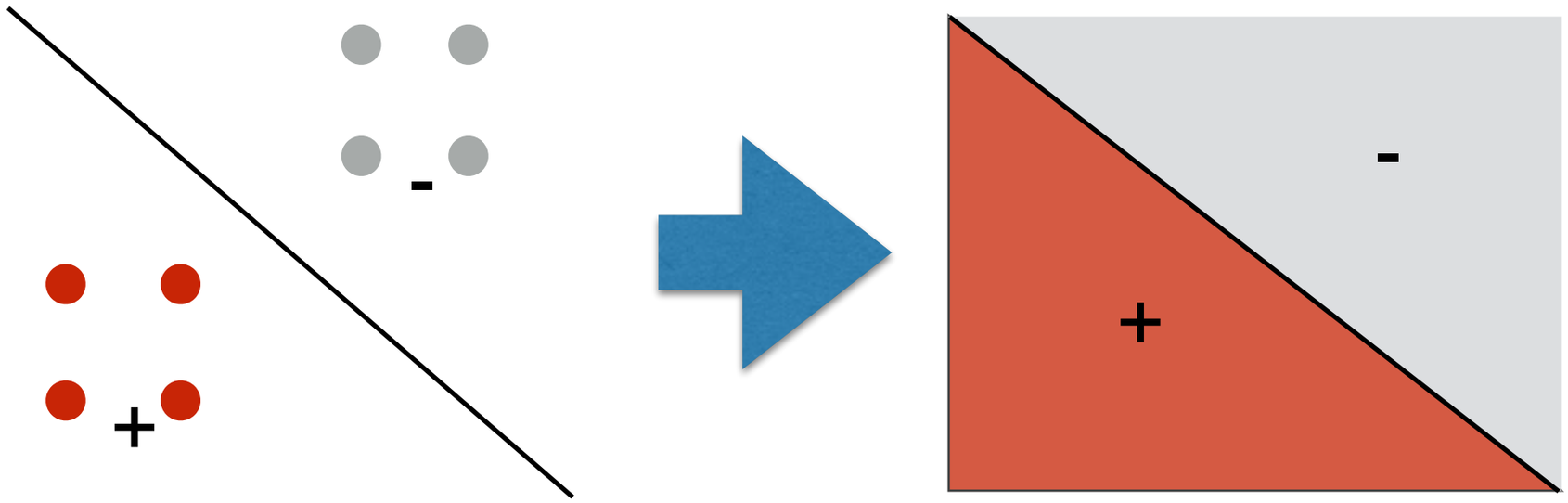
(Y)

How to do Classification



Given labeled training examples (x_1, y_1), …. (x_n, y_n),
Find a prediction rule f to predict y from x

How to do Classification



Given labeled **training examples** $(x_1, y_1), \ldots (x_n, y_n)$,
Find a prediction rule f to predict y from x

Key: Generalization (f should work on **test examples** coming from an underlying distribution)

Adversarial Examples

[ML05, S+13, G+14]

Threat Model

Learner: Builds a classifier from training data

Threat Model

Learner: Builds a classifier from training data

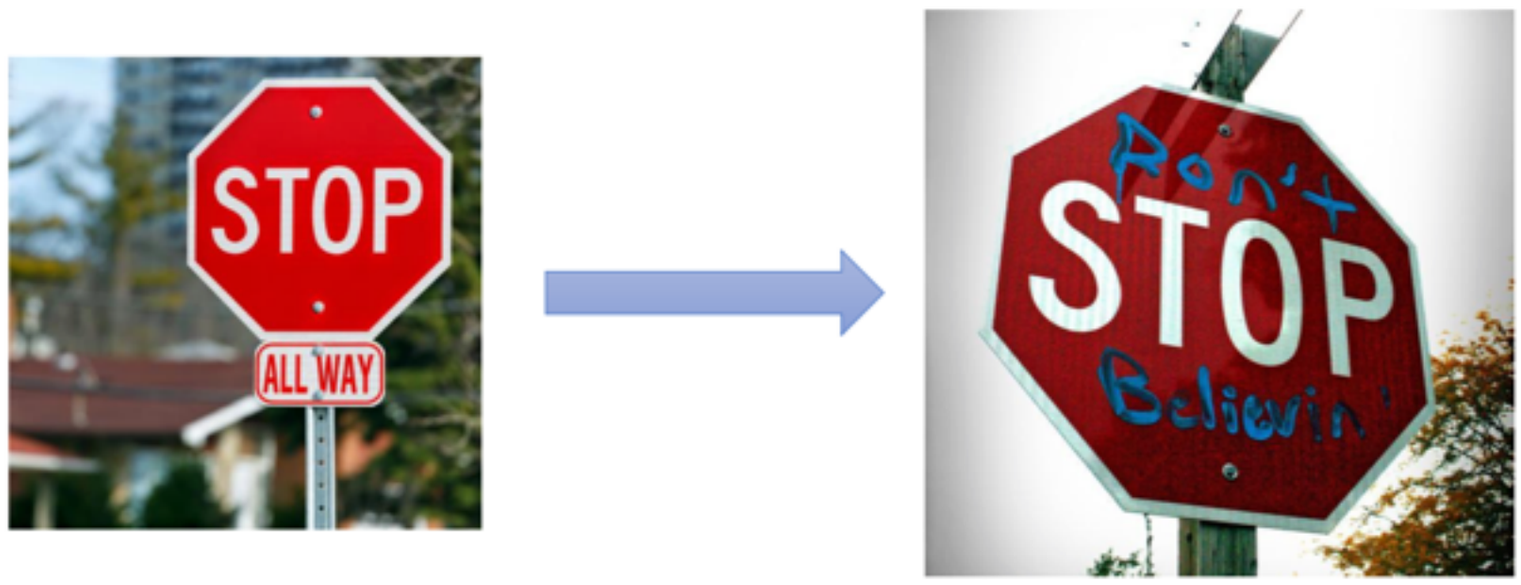
User: Uses a classifier

Threat Model

Learner: Builds a classifier from training data

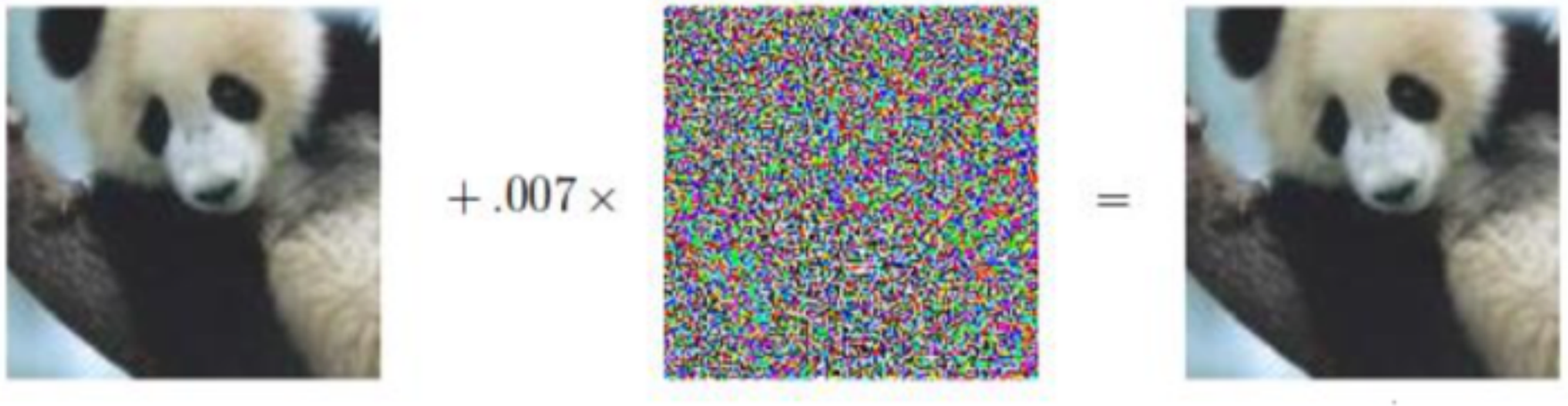
User: Uses a classifier

Adversary: Wants to make user misclassify by perturbing test examples



Many classifiers are vulnerable to adversarial examples …

[G+14]



Many classifiers are vulnerable to adversarial examples …

[P+16]

Adversarial Examples - State of the Art

- Many many attacks
- Many defenses, to be broken again by other attacks
- Only defense that has (sort of) held up - training using adversarial examples

Adversarial Examples - State of the Art

- Many many attacks

- Many defenses, to be broken again by other attacks

- Only defense that has (sort of) held up so far - training using adversarial examples

- Not much understanding on why these examples exist

Talk Outline

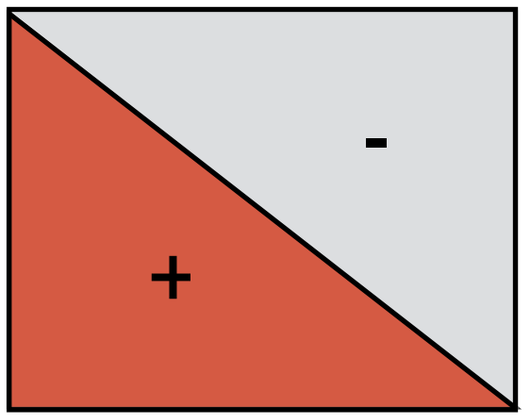
Adversarial Examples

- Background
- Definitions

Why do we have adversarial examples?

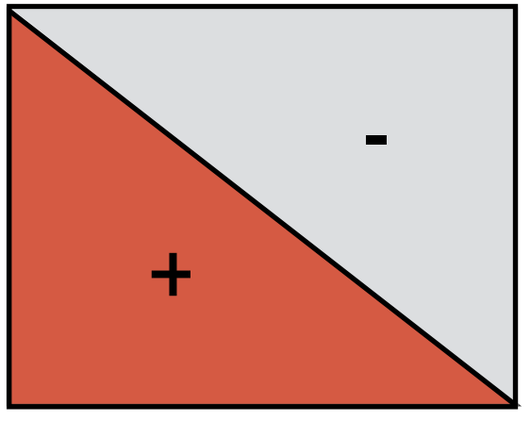
Why do we have adversarial examples?

Data distribution

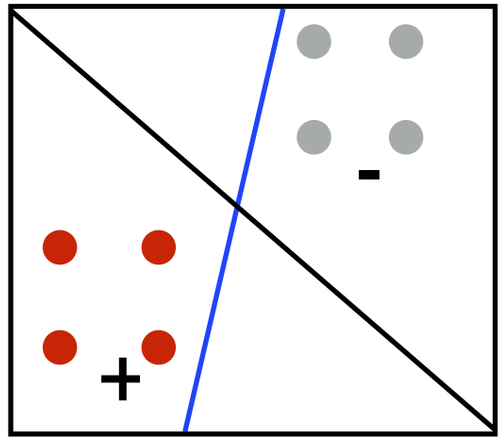


Why do we have adversarial examples?

Data distribution

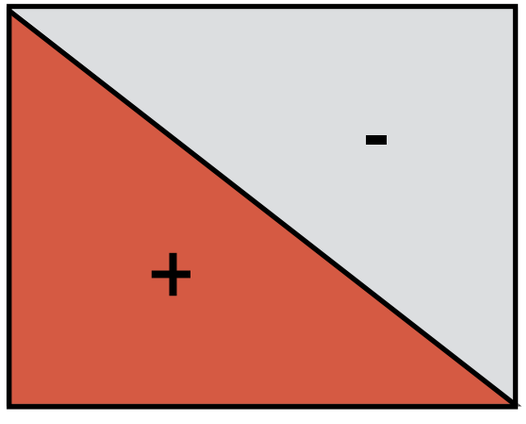


Too few samples

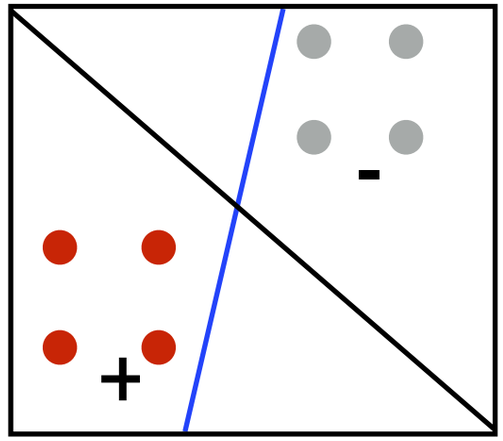


Why do we have adversarial examples?

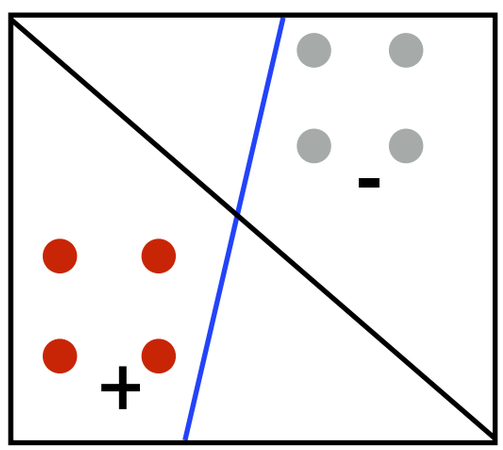
Data distribution



Too few samples

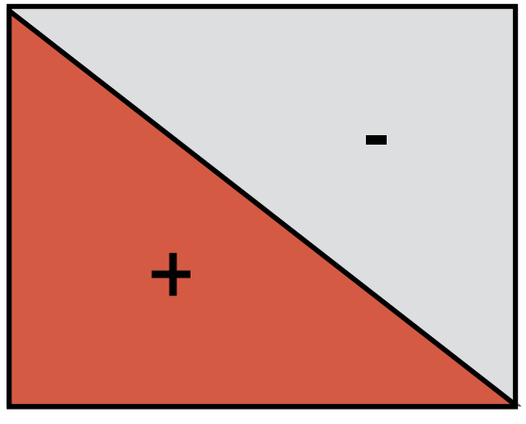


Bad algorithm



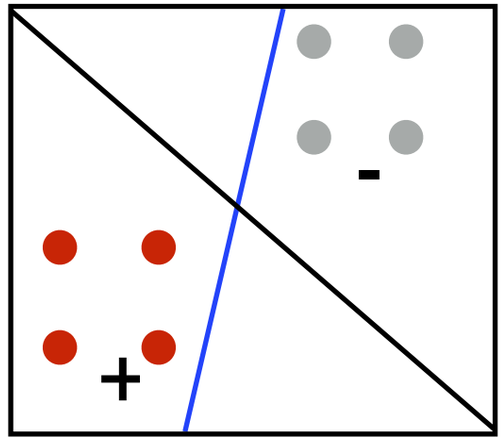
Why do we have adversarial examples?

Data distribution



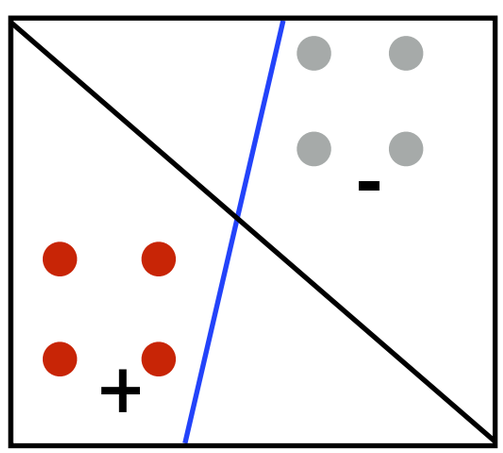
Distributional Robustness

Too few samples



Finite Sample Robustness

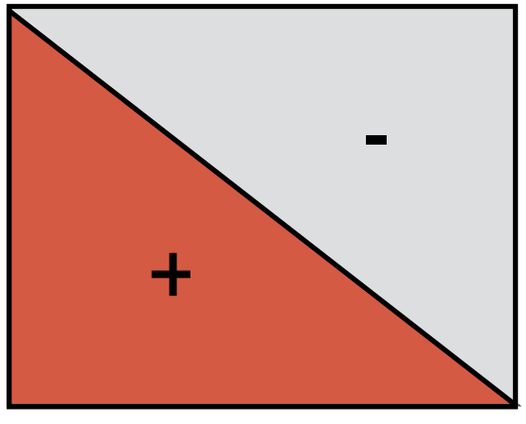
Bad algorithm



Algorithmic Robustness

Why do we have adversarial examples?

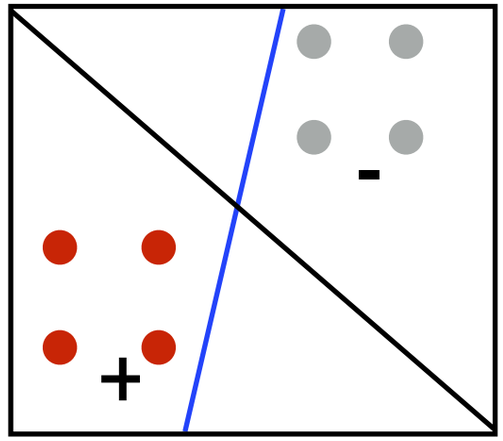
Data distribution



Distributional Robustness

(bias)

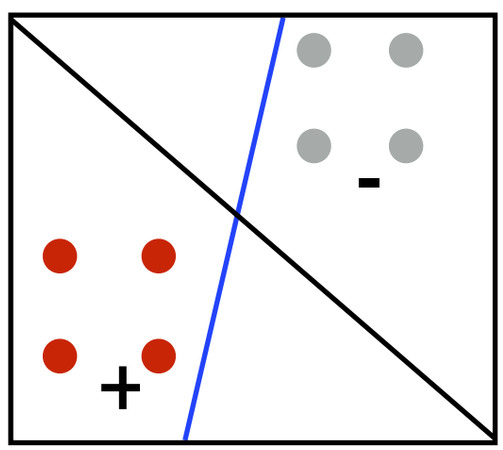
Too few samples



Finite Sample Robustness

(variance)

Bad algorithm

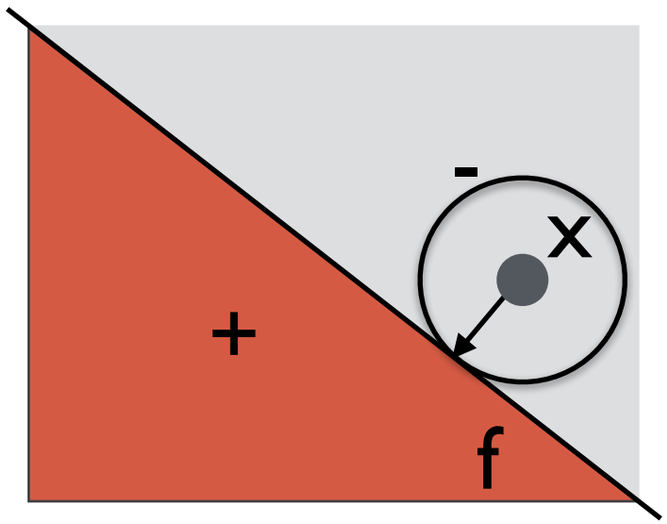


Algorithmic Robustness

Definitions

Robustness Radius

Robustness Radius $\rho(f, x)$ of a classifier f at x is the distance to closest z such that $f(x) \neq f(z)$

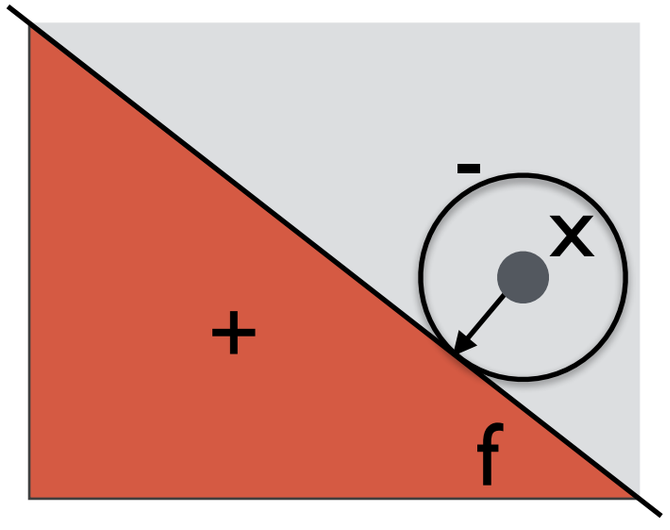


High robustness radius at x means classifier robust at x

Robustness wrt Distribution

Robustness of f at x at radius r wrt distribution μ

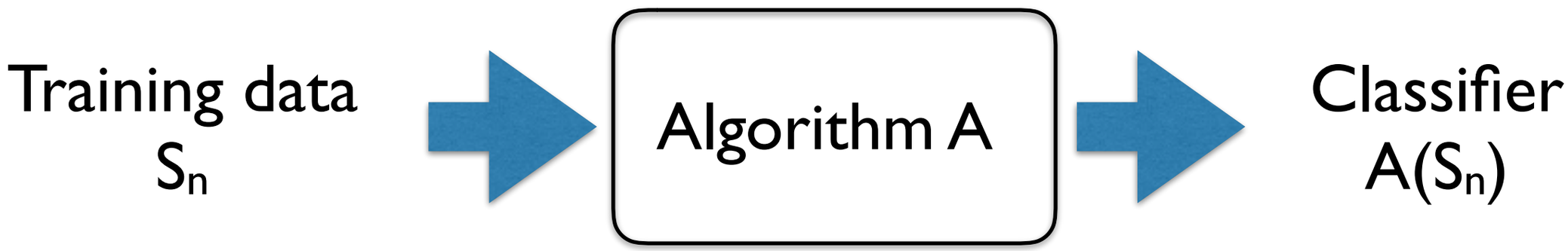
$$R(f, r, \mu) = \Pr_{x \sim \mu}(\{x | \rho(f, x) \geq r\})$$



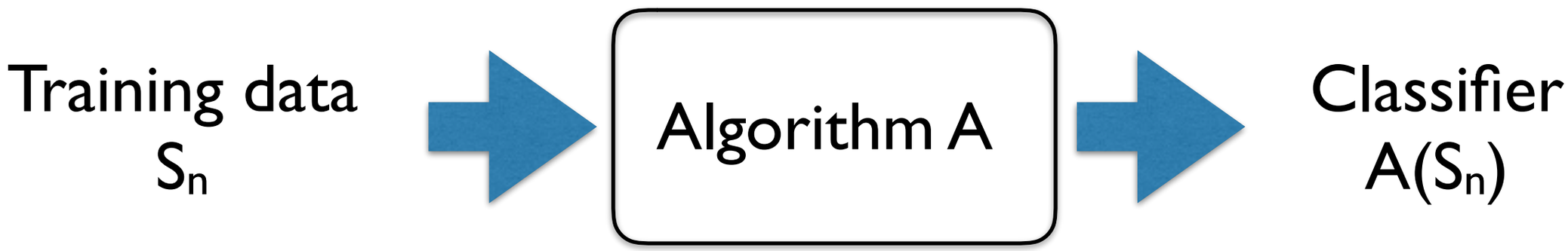
High robustness means robust classifier

μ = distribution over input instances

Robustness Definitions



Robustness Definitions

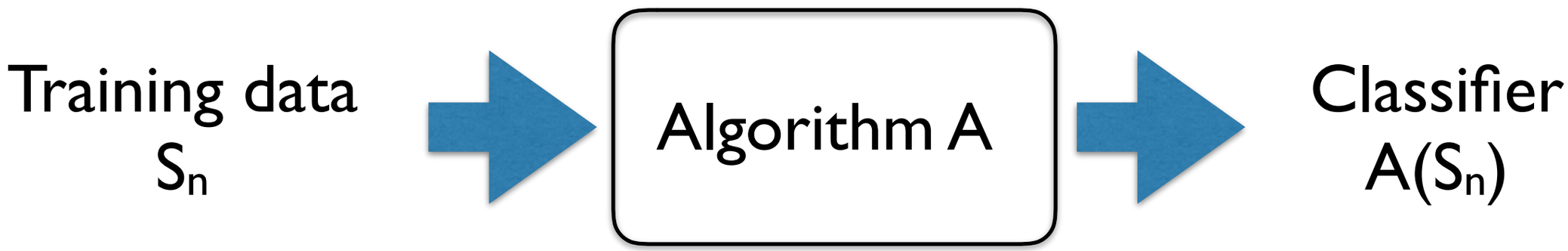


D_x = marginal of data distribution D over x

Distributional robustness of A wrt D at radius r is

$$\lim_{n\to\infty} \mathbb{E}_{S_n\sim D}[R(A(S_n), r, D_x)]$$

Robustness Definitions



D_x = marginal of data distribution D over x

Distributional robustness of A wrt D at radius r is

$$\lim_{n\to\infty} \mathbb{E}_{S_n\sim D}[R(A(S_n), r, D_x)]$$

Finite sample robustness of A wrt D at radius r bounds

$$\mathbb{E}_{S_n\sim D}[R(A(S_n), r, D_x)] \qquad \text{for finite n}$$

Talk Outline

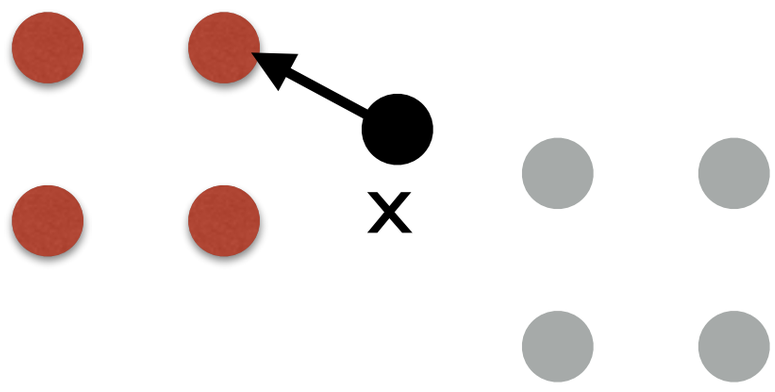
Adversarial Examples

- Background
- Definitions
- Analysis

How to analyze robustness to adversarial examples?

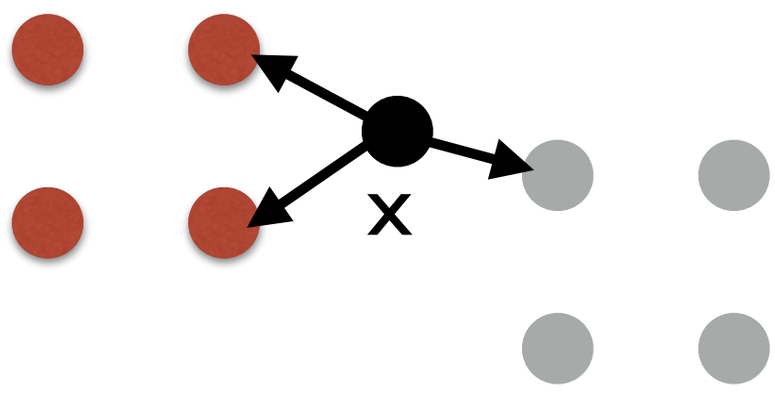
Our work - analysis for nearest neighbors

Nearest Neighbor Classifiers



Given training data $(x_1, y_1), \ldots, (x_n, y_n)$, test example x, find x_i in training data closest to x. Return y_i.

k-Nearest Neighbor Classifiers



Given training data $(x_1, y_1), \ldots, (x_n, y_n)$, test example x,
find k closest points $x_{i1}, .., x_{ik}$. Return majority$(y_{i1},..,y_{ik})$.

What is known about Nearest Neighbors?

Bayes optimal classifier g: $g(x) = 2\mathbb{I}(p(y = +|x) \geq 1/2) - 1$

Let R* = expected error of Bayes optimal classifier g

What is known about Nearest Neighbors?

Bayes optimal classifier g: $g(x) = 2\mathbb{I}(p(y = +|x) \geq 1/2) - 1$

Let R* = expected error of Bayes optimal classifier g

Asymptotic [CH67, DGL96]:

Error of 1-NN ➡ 2R*(1 - R*) as $n \to \infty$

Error of k-NN ➡ R* as $n \to \infty, k_n \to \infty, k_n/n \to 0$

Finite sample rates: highly distribution dependent

What about robustness of nearest neighbors?

Robustness of 1-Nearest Neighbor

$A_1(S_n)$ = 1-Nearest Neighbor classifier on training set S_n

Theorem: If, in some neighborhood of x,

- D_x is absolutely continuous
- p(y=+|x) is continuous
- 0 < p(y=+|x) < 1 **(impure region)**

then $\rho(A_1(S_n), x) \to 0$ as $n \to \infty$

Robustness of 1-Nearest Neighbor

$A_1(S_n)$ = 1-Nearest Neighbor classifier on training set S_n

Theorem: If, in some neighborhood of x,

- D_x is absolutely continuous
- p(y=+|x) is continuous
- 0 < p(y=+|x) < 1 **(impure region)**

then $\rho(A_1(S_n), x) \to 0$ as $n \to \infty$

Distributional robustness of NN in "impure" regions is 0

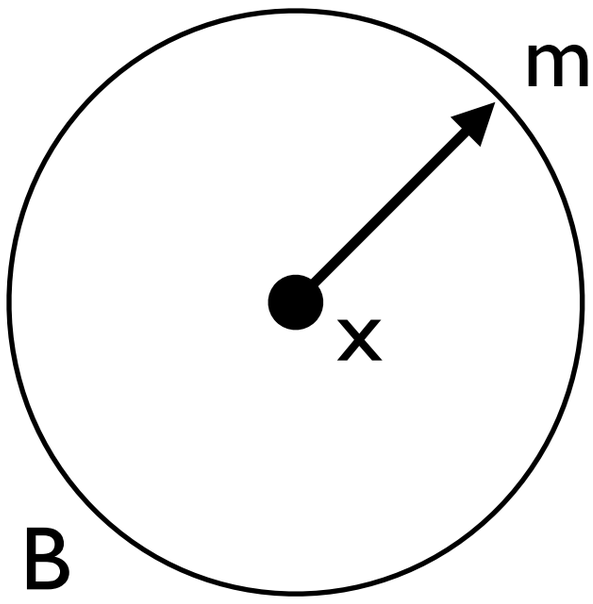
Accuracy is non-zero!

Proof Ideas

Theorem: If, in some neighborhood of x,

- D_x is absolutely continuous
- p(y=+|x) is continuous
- 0 < p(y=+|x) < 1

then $\rho(A_1(S_n), x) \to 0$ as $n \to \infty$

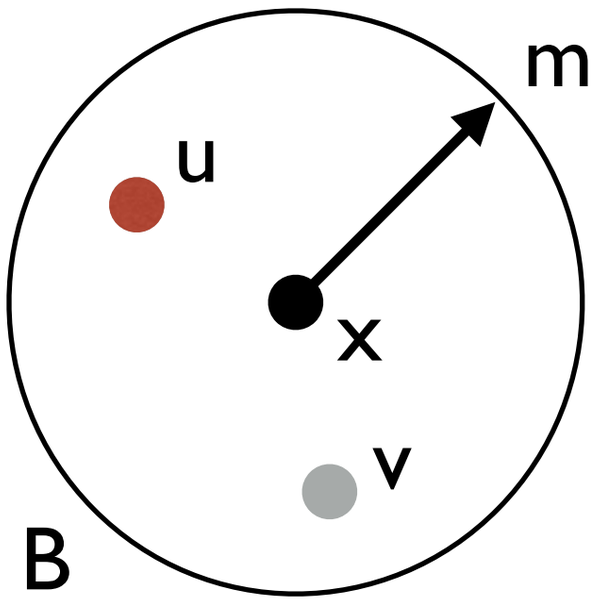


Proof Ideas

Theorem: If, in some neighborhood of x,

- D_x is absolutely continuous
- p(y=+|x) is continuous
- 0 < p(y=+|x) < 1

then $\rho(A_1(S_n), x) \to 0$ as $n \to \infty$



min(r/2, diam(nbd(x))

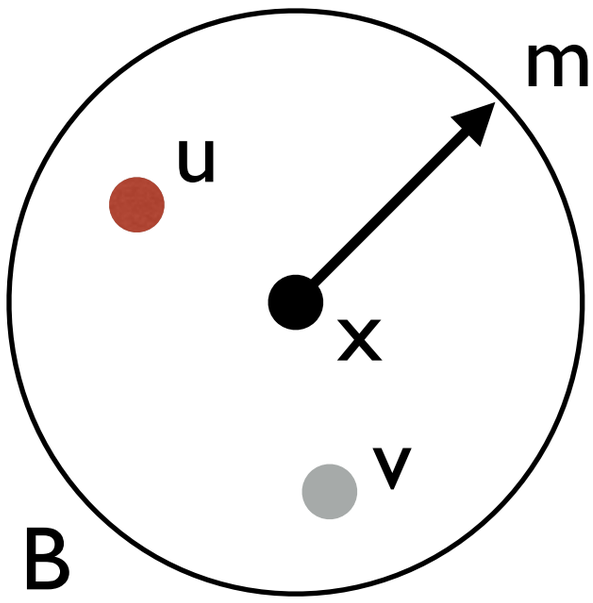
For large enough n, there are at least two points u, v with different labels in B

Proof Ideas

Theorem: If, in some neighborhood of x,

- D_x is absolutely continuous
- p(y=+|x) is continuous
- 0 < p(y=+|x) < 1

then $\rho(A_1(S_n), x) \to 0$ as $n \to \infty$



min(r/2, diam(nbd(x))

For large enough n, there are at least two points u, v with different labels in B

One of them is adversarial example for x

Robustness of k-Nearest Neighbor

$A_k(S_n)$ = k-Nearest Neighbor classifier on training set S_n

Theorem: If, in some neighborhood of x,

- D_x is absolutely continuous
- p(y=+|x) is continuous
- 0 < p(y=+|x) < 1 **(impure region)**

then $\rho(A_k(S_n), x) \to 0$ as $n \to \infty$

Robustness of k-Nearest Neighbor

$A_k(S_n)$ = k-Nearest Neighbor classifier on training set S_n

Theorem: If, in some neighborhood of x,

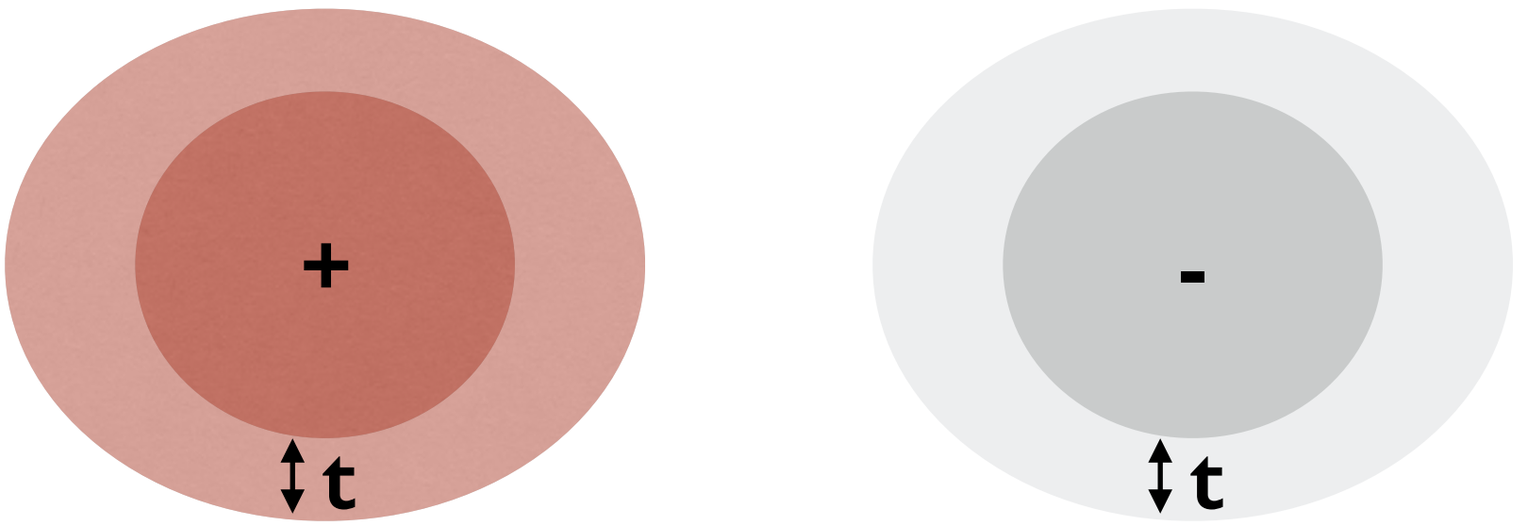
- D_x is absolutely continuous
- p(y=+|x) is continuous
- 0 < p(y=+|x) < 1 **(impure region)**

then $\rho(A_k(S_n), x) \to 0$ as $n \to \infty$

k-NN does not help!

(unlike accuracy where kNN is better than 1NN)

Interiors



t-interior of + region = all x s.t B(x, t) has p(y=+|x) = 1
t-interior of - region = all x s.t B(x, t) has p(y=-|x) = 1

1-NN has non-zero robustness radius in the interiors of the + and - regions

Robustness Bounds

Let X_t = (t-interior of + region) U (t-interior of - region)

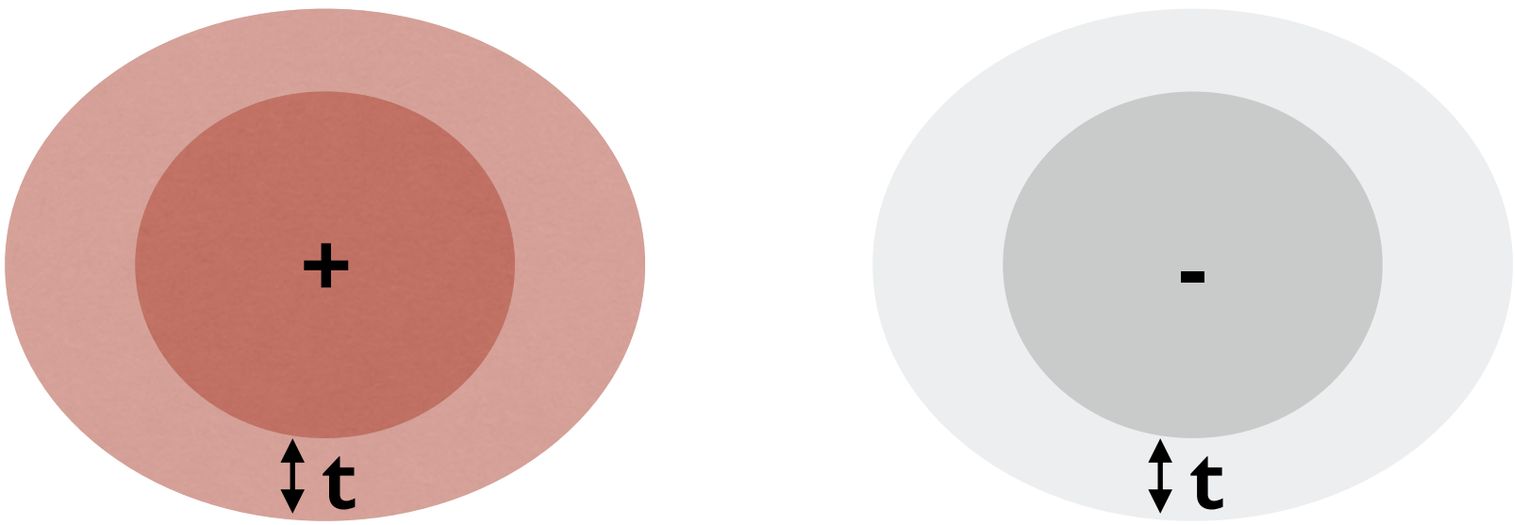
Theorem:

$$\mathbb{E}[R(A(S_n), r, D_x)] \geq P(X_{2r+t}) - d_{t,n}$$

where $d_{t,n} = \mathbb{E}_{x_0 \sim D_x}[(1 - P(B(x_0, t)))^n]$

1. $d_{t,n}$ = distribution dependent quantity
   For continuous D_x, fixed t, goes to 0 with large enough n
2. X_{2r+t} goes to X_{2r} as t goes to 0

Summary: Robustness of NN



NN is non-robust in “impure” regions

NN is robust in the interior of “pure” regions

What happens in between depends on data distribution

Talk Outline

Adversarial Examples

- Background
- Definitions
- Analysis

- Defense

When is NN robust?

Let g(x) = the Bayes optimal classifier

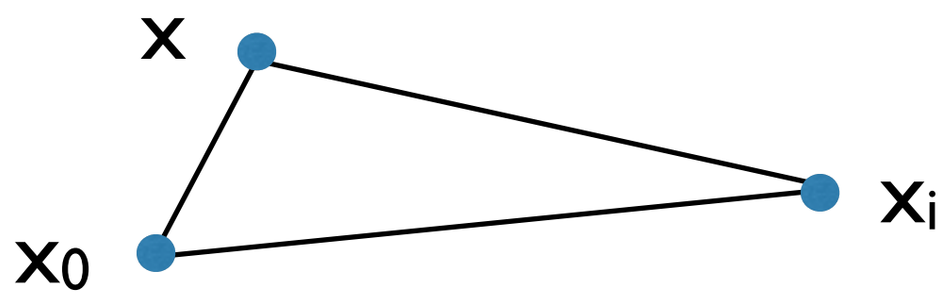
Theorem: If there is a training point (x_0, y_0) s.t. if

(a) $g(x) = y_0$

(b) For all (x_i, y_i) in training set with $y_i \neq g(x)$ implies:

$$d(x_0, x_i) > 2r + 2d(x, x_0)$$

then NN has robustness radius at least r at x.



Robust if differently labeled points are far apart

Algorithm Idea

- Remove a subset of training data s.t differently labeled points are far apart
- Do NN on the remaining data

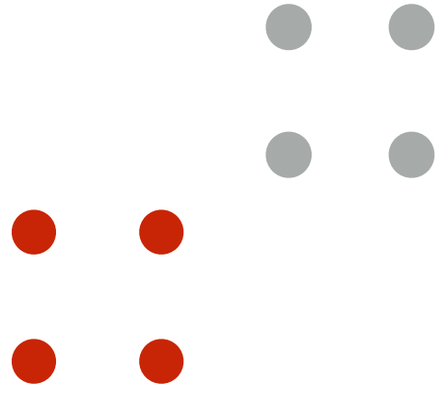
Which points to remove?

Algorithm

r-separated set:

Set of points $(x_1, y_1), \ldots (x_m, y_m)$ s.t

$$\|x_i - x_j\| \leq r \implies y_i = y_j$$

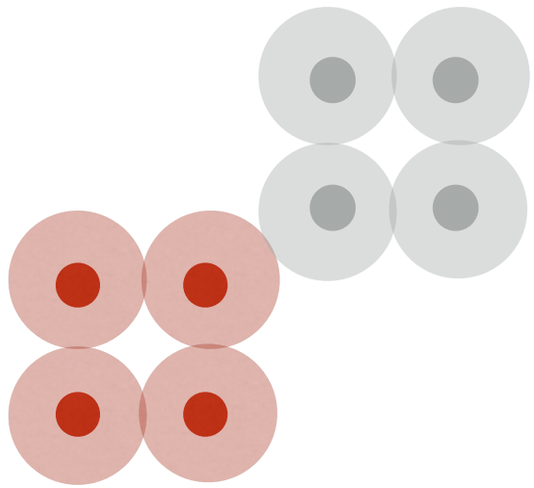


Algorithm

r-separated set:

Set of points $(x_1, y_1), \ldots (x_m, y_m)$ s.t

$$\|x_i - x_j\| \leq r \implies y_i = y_j$$



t-Cover induced by a set S:

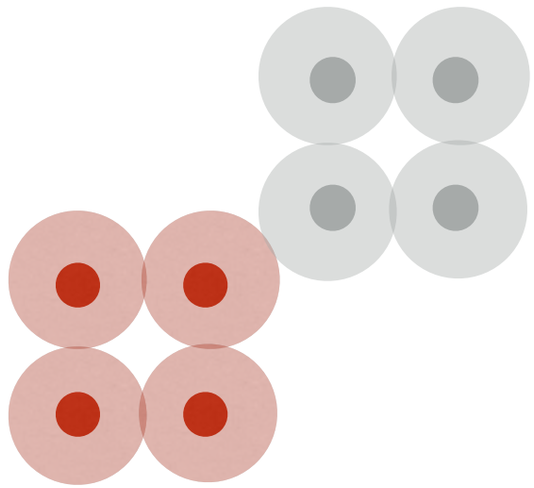
$$Cov(S, t) = U_{x \text{ in } S}\ B(x, t)$$

Algorithm

r-separated set:

Set of points $(x_1, y_1),\ldots(x_m, y_m)$ s.t

$$\|x_i - x_j\| \leq r \implies y_i = y_j$$



t-Cover induced by a set S:

$Cov(S, t) = U_{x \text{ in } S} B(x, t)$

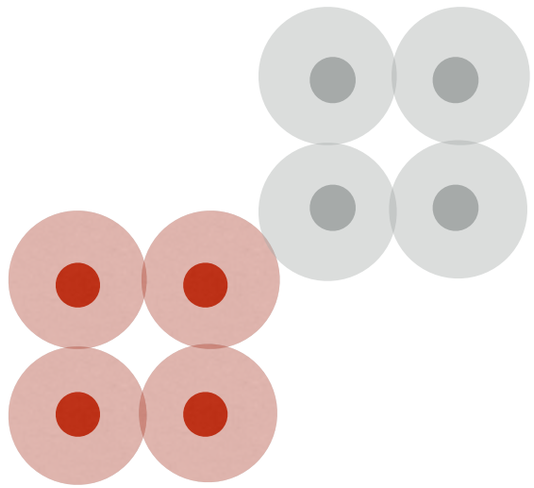
Main Idea: For robustness radius r, keep the r-separated subset S of the training set with max $Pr(Cov(S, 2r + t))$

Algorithm

r-separated set:

Set of points $(x_1, y_1),\ldots(x_m, y_m)$ s.t

$$\|x_i - x_j\| \leq r \implies y_i = y_j$$



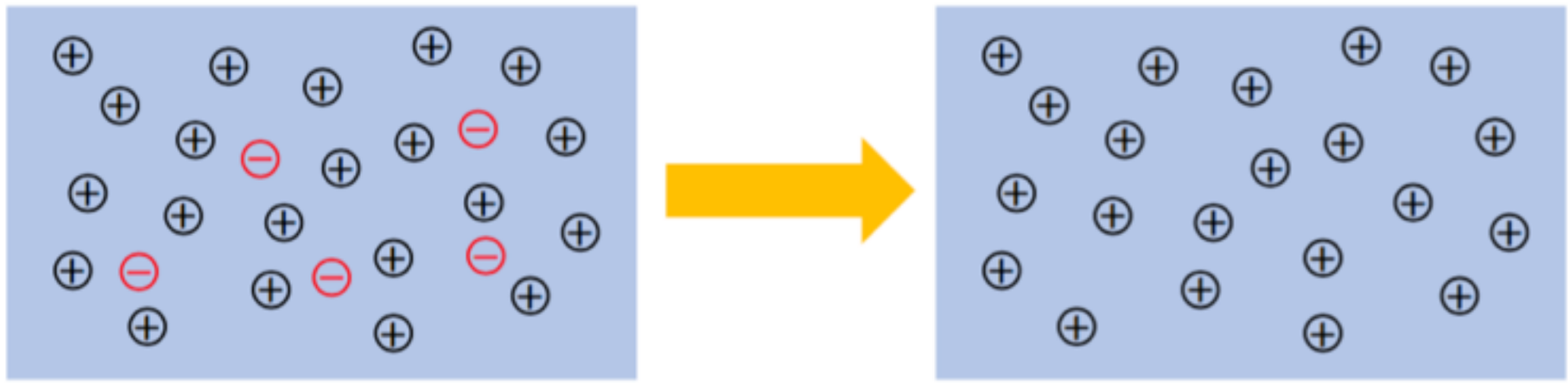
t-Cover induced by a set S:

$Cov(S, t) = U_{x \text{ in } S} B(x, t)$

Main Idea: For robustness radius r, keep the r-separated subset S of the training set with max $Pr(Cov(S, 2r + t))$

Estimate $Pr(Cov(S, 2r + t))$ with extra unlabeled dataset

An Example of how this helps



(70% +, 30% -)

Distributional Robustness of NN = 0 (impure region)

Distributional Robustness of Robust-NN = 1

Performance Guarantees: Definitions

$A_{IR}(S)$ = Robust 1NN classifier on training data S

t-separated set wrt Bayes Optimal classifier:

A set of points S are t-separated wrt Bayes Optimal classifier g if for x and z in S, $\|x - z\| \leq t \implies g(x) = g(z)$

$V_{max}(t)$ = t-separated set V wrt Bayes Optimal classifier with largest Pr(V)

Performance Guarantees

$A_{1R}(S)$ = Robust 1NN classifier on training data S

Theorem:

$$\mathbb{E}[R(A_{1R}(S_n), r, D_x)] \geq \Pr(V_{\max}(2r + 2t)) - [2\delta + \epsilon(n, m, \delta) + c_{t,n}]$$

where:

$$\epsilon(n, m, \delta) \to 0 \quad \text{as} \quad m, n \to \infty \quad \text{(m = \#unlabeled pts)}$$
$$c_{t,n} \to 0 \quad \text{as} \quad n \to \infty$$

A little better than NN

Open Question: Can the rates be improved?

Talk Outline

Adversarial Examples

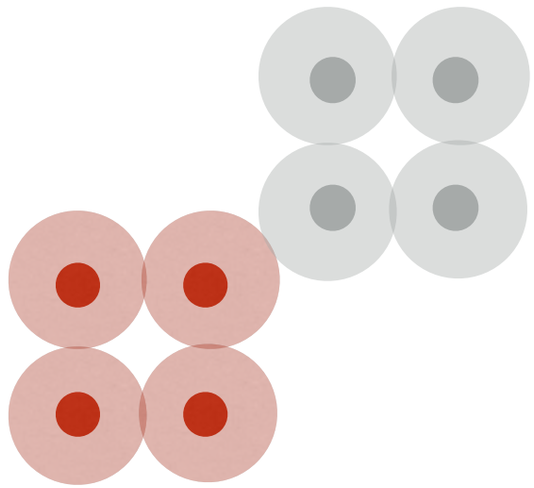
- Background
- Definitions
- Analysis
- Defense
- Validation

Recall: Algorithm

r-separated set:

Set of points $(x_1, y_1),\ldots(x_m, y_m)$ s.t

$$\|x_i - x_j\| \leq r \implies y_i = y_j$$



t-Cover induced by a set S:

$Cov(S, t) = U_{x \text{ in } S} B(x, t)$

Main Idea: For robustness radius r, keep the r-separated subset S of the training set with max $Pr(Cov(S, 2r + t))$

Estimate $Pr(Cov(S, 2r + t))$ with extra unlabeled dataset

Algorithm in Practice

In practice:
— extra unlabeled data usually unavailable
— hard to find r-separated set with max Pr(Cov(S, 2r + t))
— choosing t is challenging

So, pick **largest** r-separated subset of training set to keep

For 2 labels, reduces to max matching in bipartite graph
(for higher labels, reduces to min vertex cover)

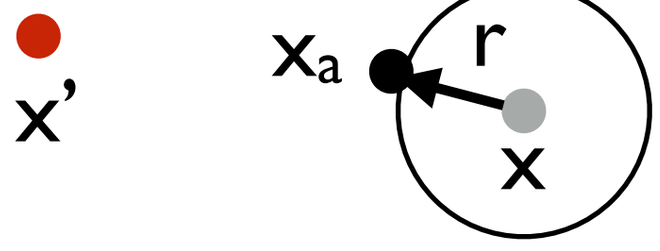
Methodology - White Box Attacks

Attack Method: Given x and radius r,

y = label of x assigned by NN

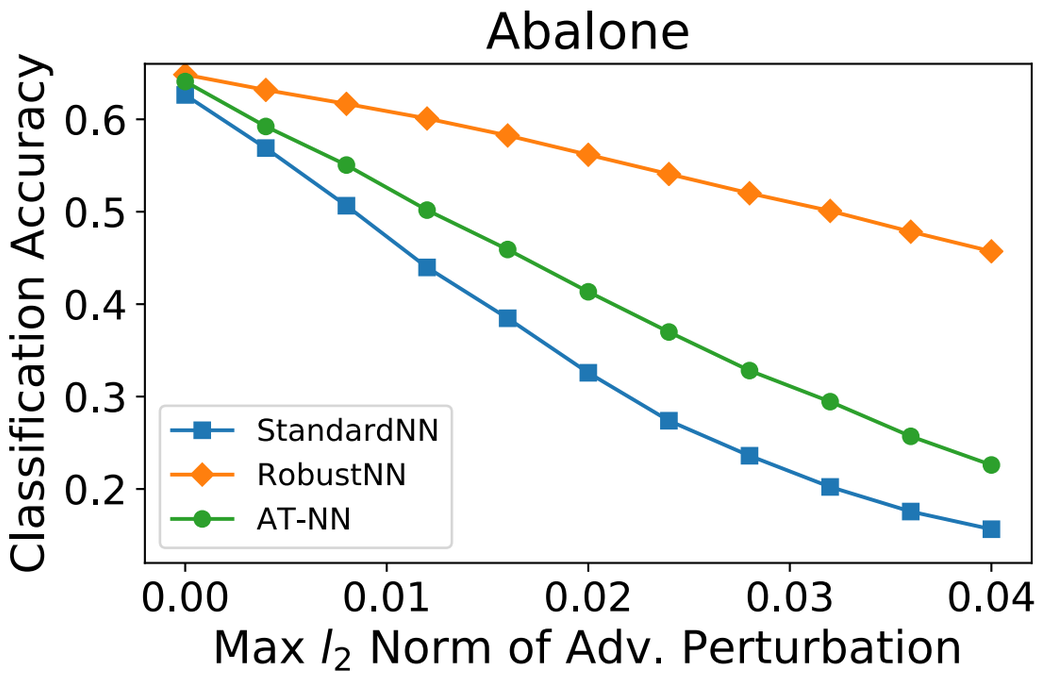
Find x' closest to x with label different from y

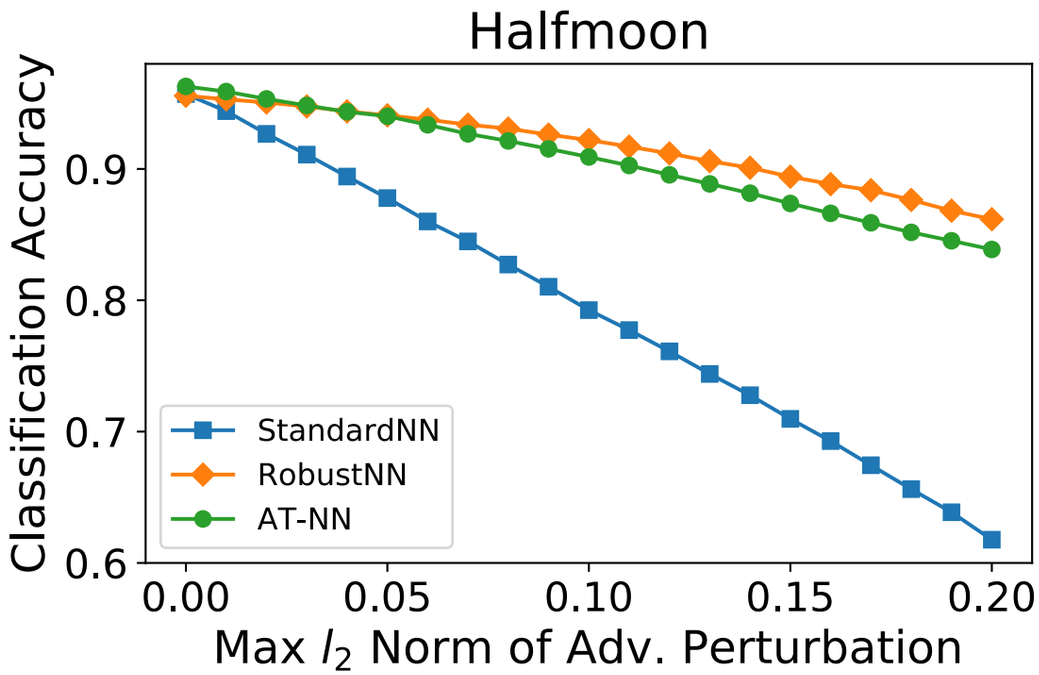
Return $x_a = x + r \frac{x - x'}{\|x - x'\|}$

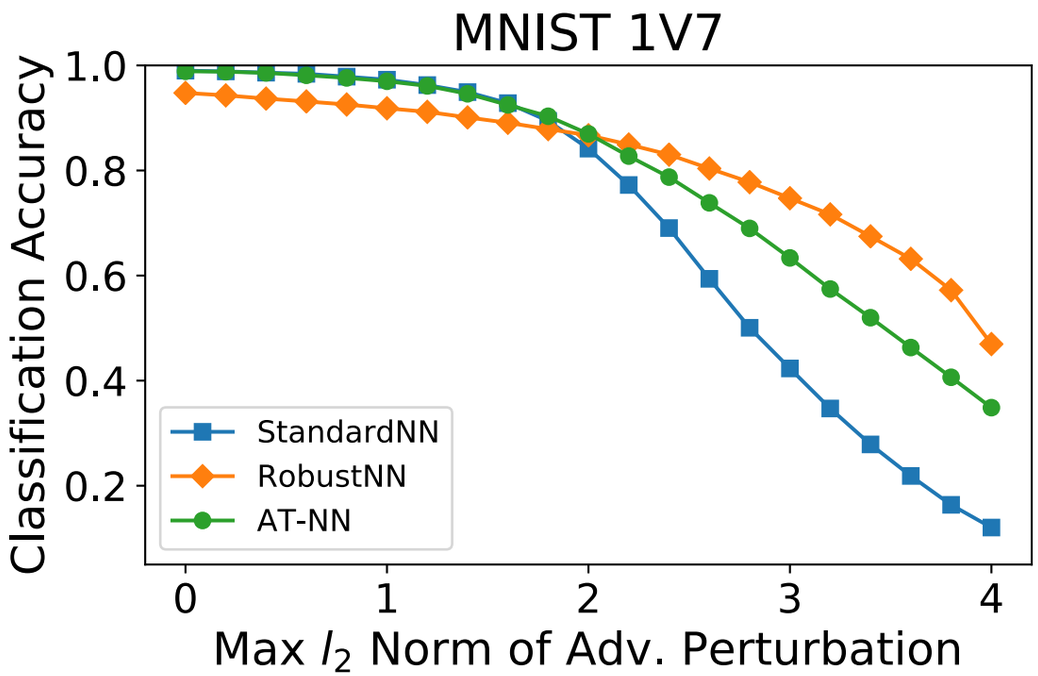


Baselines: StdNN, RobustNN, AdversariallyTrainedNN

Results







Datasets (clockwise):

Abalone, Halfmoon, MNIST 1v7

Methodology - Black Box Attacks

Attack Method [P+17]: Given x and radius r,

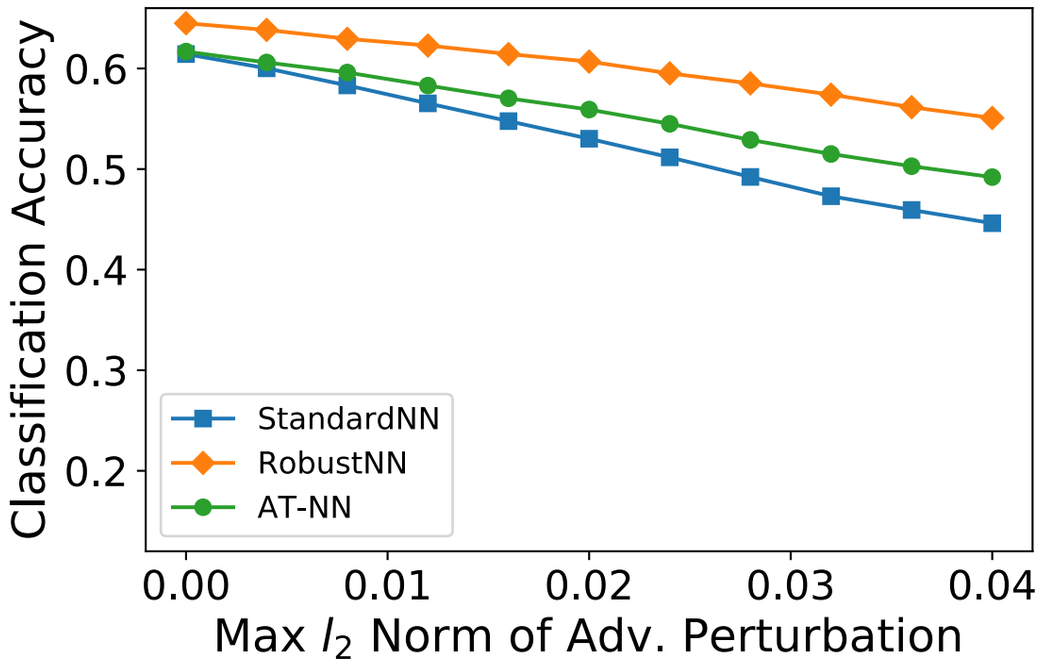
Train a *substitute classifier* f by querying the NN classifier as an oracle

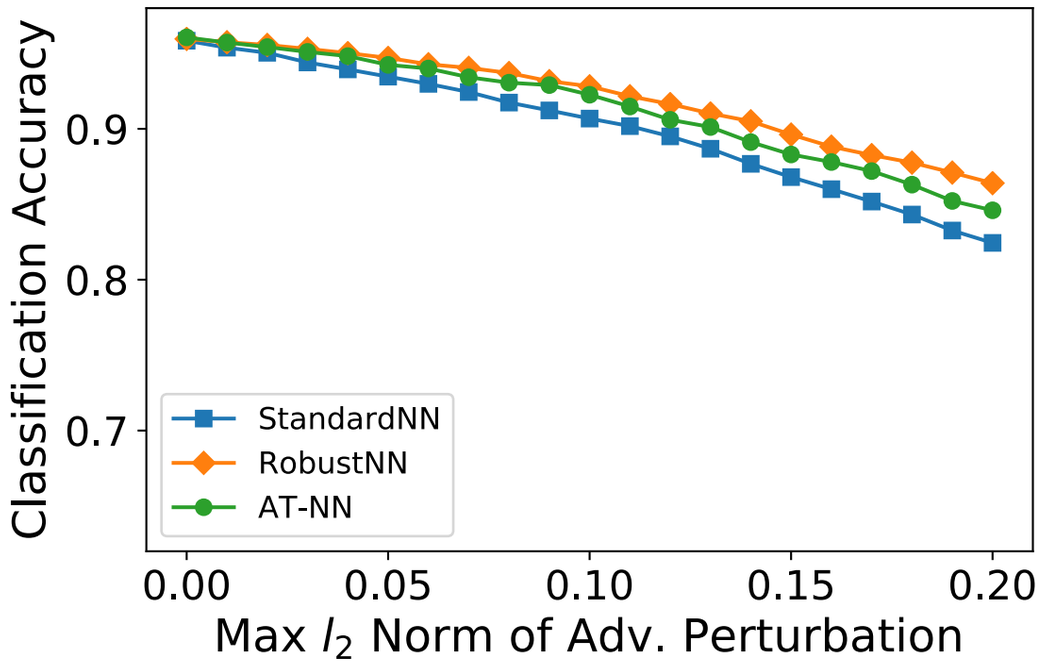
Find adversarial example for f around x

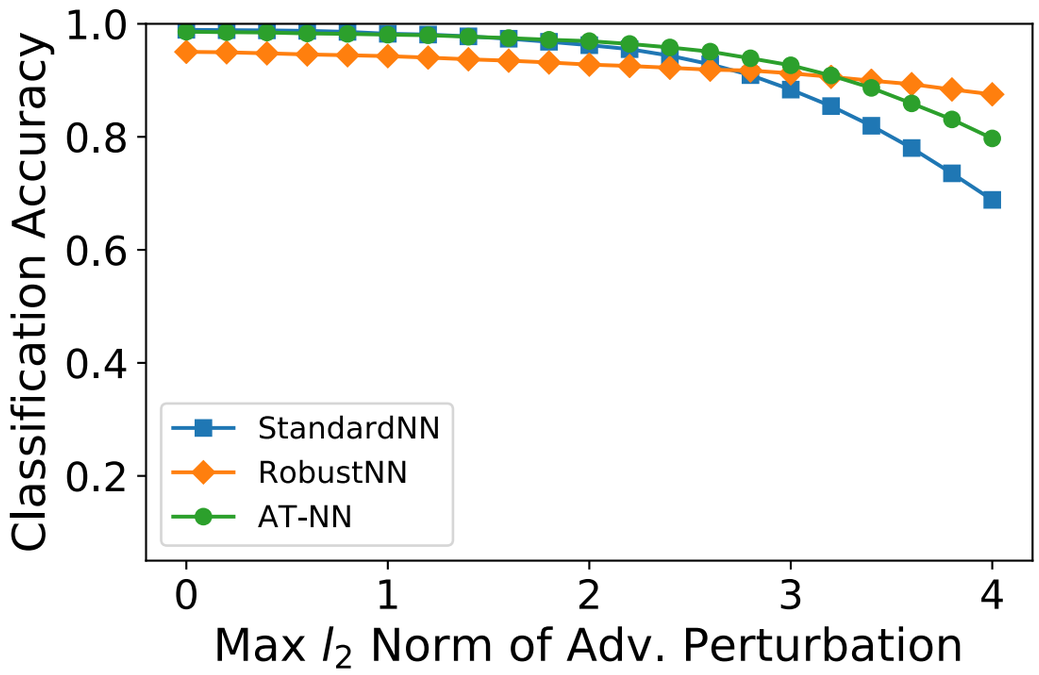
Classifiers used: NeuralNet, Kernel

Baselines: StdNN, RobustNN, AdversariallyTrainedNN

Results - Kernel



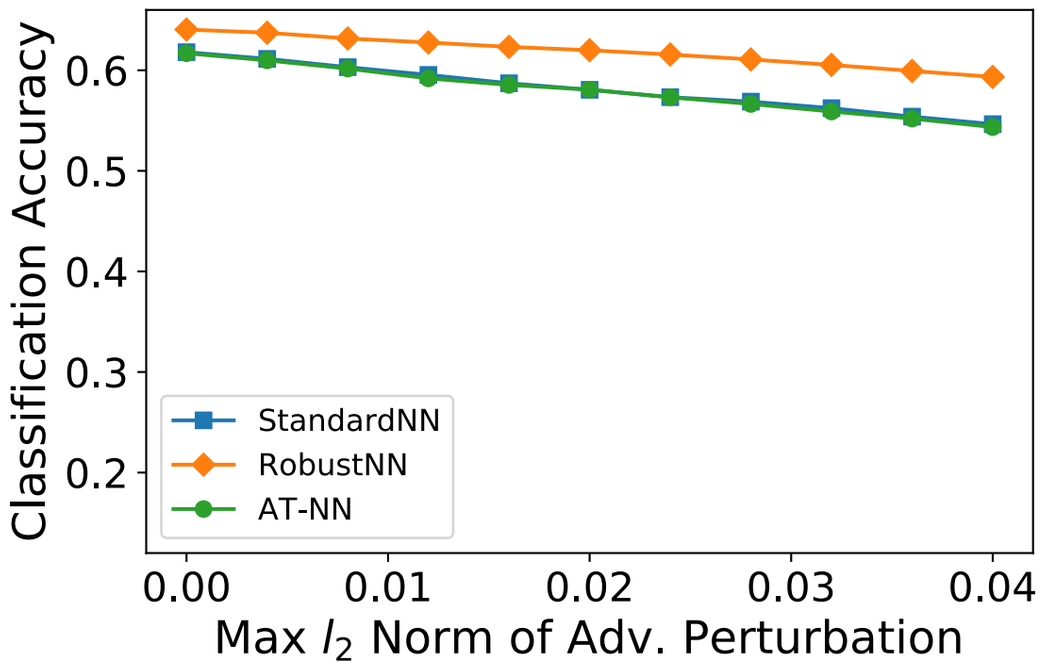


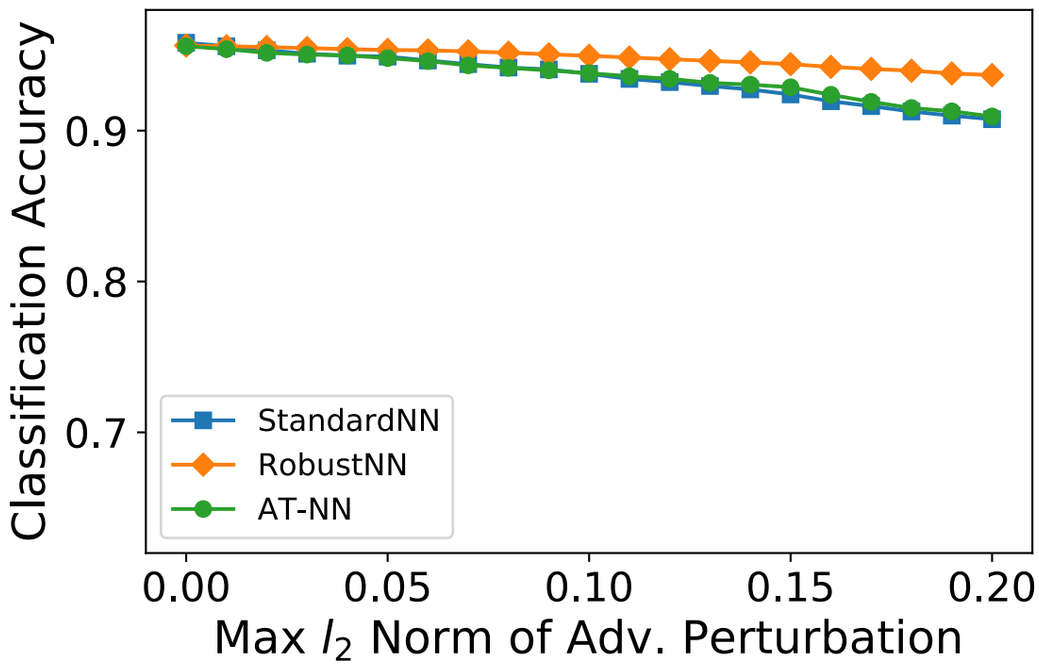


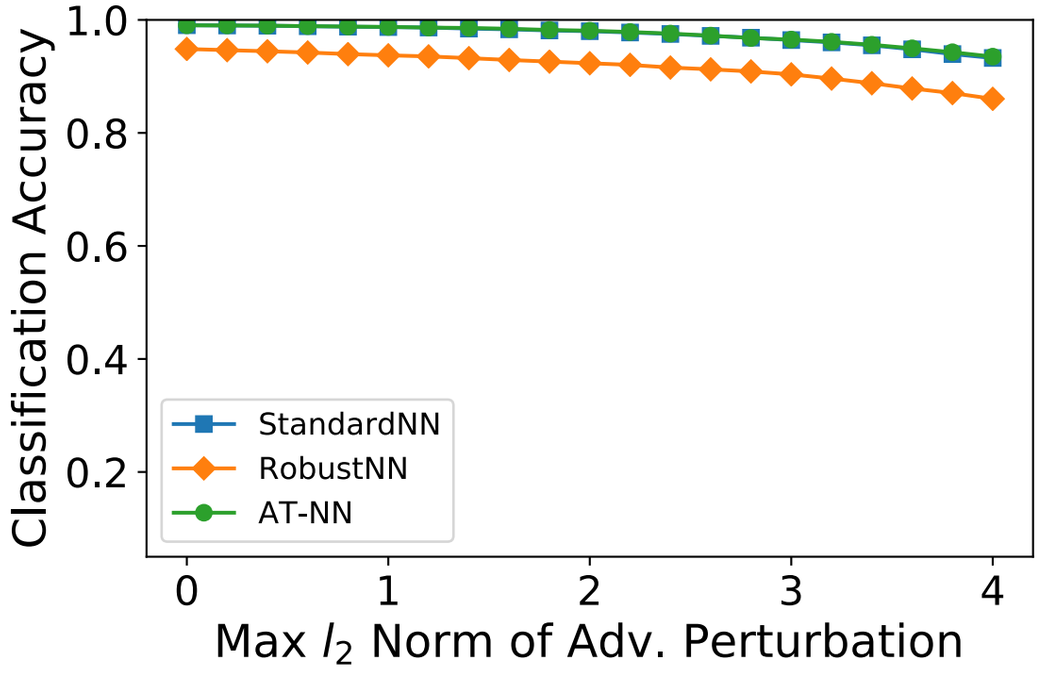
Datasets (clockwise):

Abalone, Halfmoon, MNIST 1v7

Results - Neural Nets







Datasets (clockwise):

Abalone, Halfmoon, MNIST 1v7

Conclusions

— We provide new definitions for distributional and finite sample robustness

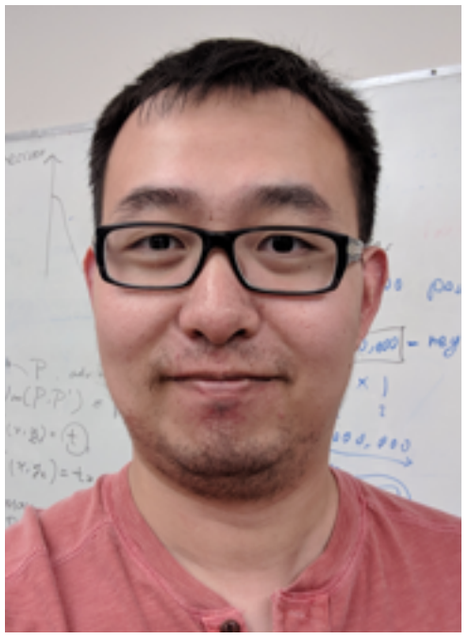
— Analyze the robustness of nearest neighbors

— Analysis leads to a new defense

Reference

"Analyzing the Robustness of Nearest Neighbors to Adversarial Examples", Yizhen Wang, Somesh Jha and Kamalika Chaudhuri, arXiv:1706.03922

Acknowledgments



Yizhen Wang

Somesh Jha