Nearest Neighbors II: Adversarial Examples

Kamalika Chaudhuri
University of California, San Diego

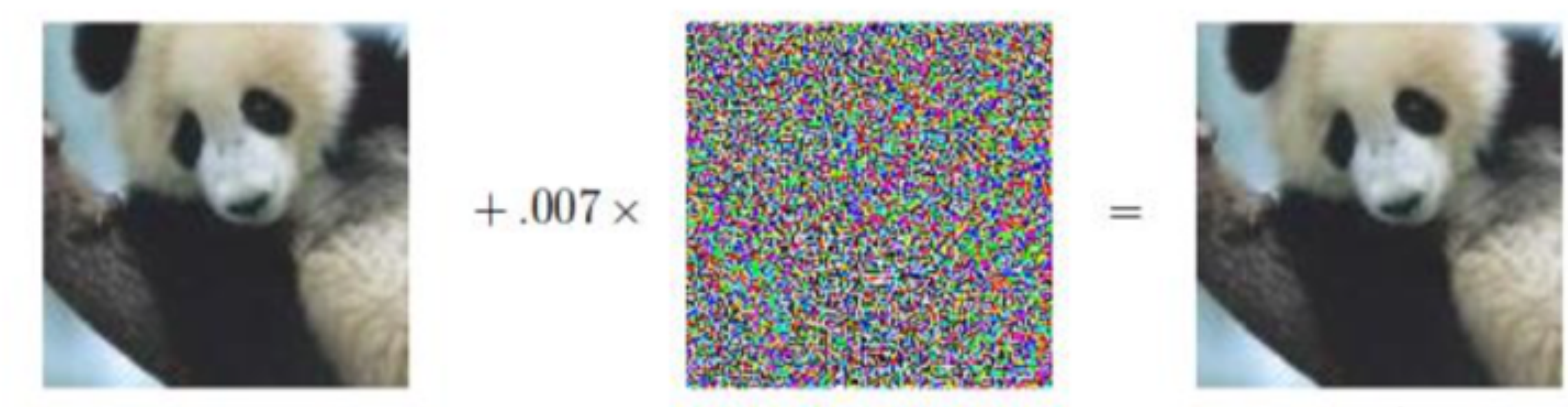
Talk Outline

- Part I: k-Nearest neighbors: Regression and Classification

- Part II: k-Nearest neighbors (and other non-parametrics): Adversarial examples

Adversarial Examples

[G+14]



Panda

Gibbon

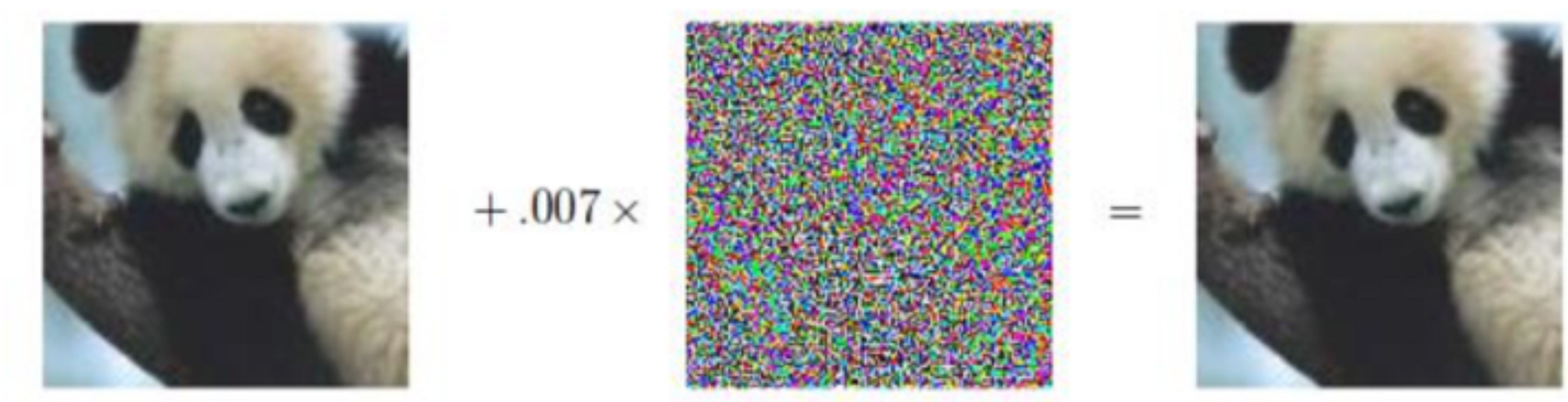
[Goodfellow+14,], [Szegedy+13], [Meek-Lowd 05],….

Adversarial Examples

Slight strategic modification of test input causes misclassification

Many Classifiers are Vulnerable to Adversarial Examples

[G+14]



Panda

Gibbon

State of the Art

- Many, many attacks
- Many defenses, to be broken again
- Some certifiable defenses

- Limited understanding on why these examples exist

Our Work: Adversarial examples for nearest neighbors

Talk Outline

- Adversarial Examples
  - A Statistical Learning Framework for Robustness
- Adversarial Examples for Nearest Neighbors
  - Small and large k
  - A Robust Modified Nearest Neighbor
- Beyond Nearest Neighbors
  - The r-Optimal Classifier
  - Experiments

Statistical Learning Framework

Metric space (X, d)

Underlying measure μ on X from which points are drawn

Label of x is a coin flip with bias $\eta(x) = \Pr(y = 1|x)$

Accuracy of a classifier f is acc(f) = Pr(f(x) = y)

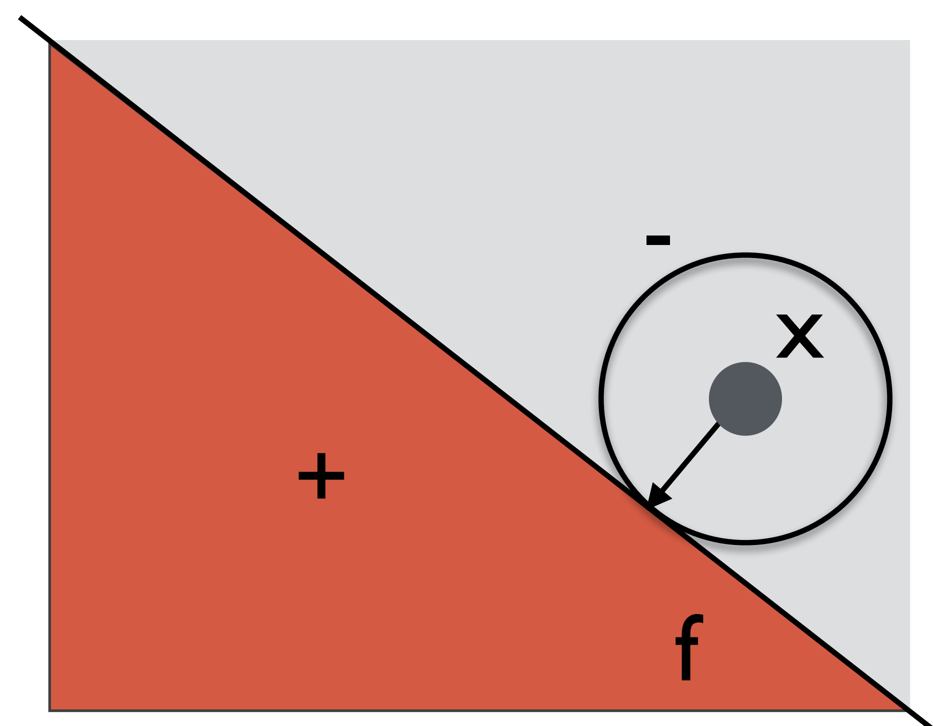
Goal: Find classifiers f with max accuracy

Definitions

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

Denoted by $\rho(f, x)$

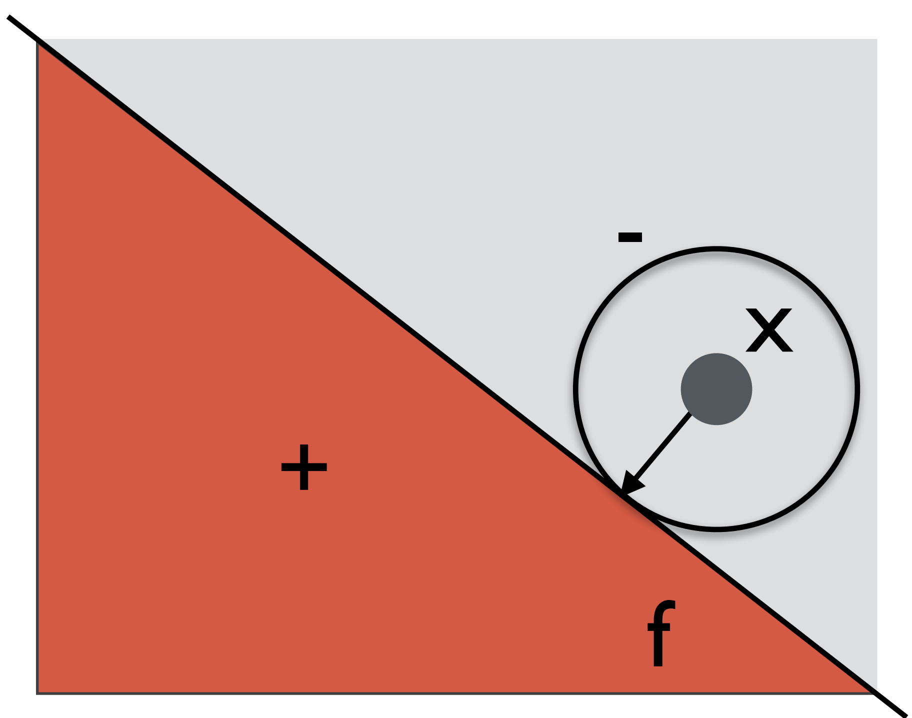
Higher robustness radius implies robust classifier at x

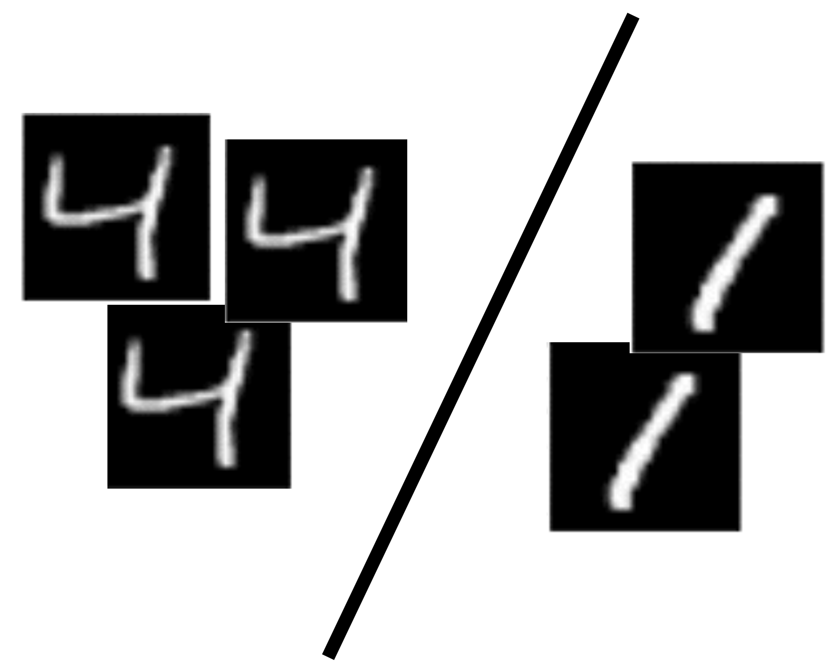


Robustness wrt Distribution

Robustness of a classifier f at radius r wrt underlying distribution μ:

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





High R implies high robustness on inputs from distribution

Robustness Definitions

Training Data S_n → Algorithm A → Classifier $A(S_n)$

Distributional robustness of A at radius r is

$$\lim_{n \to \infty} \mathbb{E}[R(A(S_n), r, \mu)]$$

Finite sample robustness of A gives bounds on

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

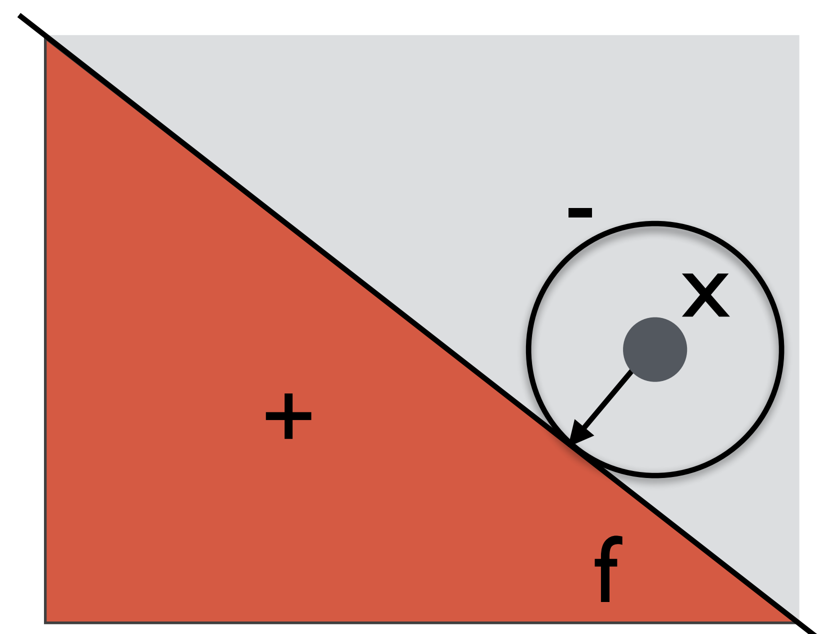
[Wang, Jha, Chaudhuri'18]

Astuteness: Combining Robustness and Accuracy

The astuteness of classifier f at radius r is defined as:

$$\mathrm{ast}(f, r) = \Pr(f(x) = y, \rho(f, x) \geq r)$$

Fraction of points
where f is robust and accurate



Goal of robust learning is
maximizing astuteness

Distributional and finite sample astuteness: similar

[Wang, Jha, Chaudhuri'18, Tsipras+19]

Prior Work - Parametric Methods

- [Schmidt+18] For linear classifiers, adversarial robustness requires more data
- [Bubeck+18] Achieving robustness to adversarial examples may be more computationally challenging
- Others - [Yin+18, Montesser+19] - bounds on adversarial generalization

How to non-parametric methods respond to adversarial examples?

Tutorial Outline

- Adversarial Examples
  - A Statistical Learning Framework for Robustness
- Adversarial Examples for Nearest Neighbors
  - Small and large k
  - A Robust Modified Nearest Neighbor
- Beyond Nearest Neighbors
  - Generic Attacks
  - The r-Optimal Classifier
  - Experiments

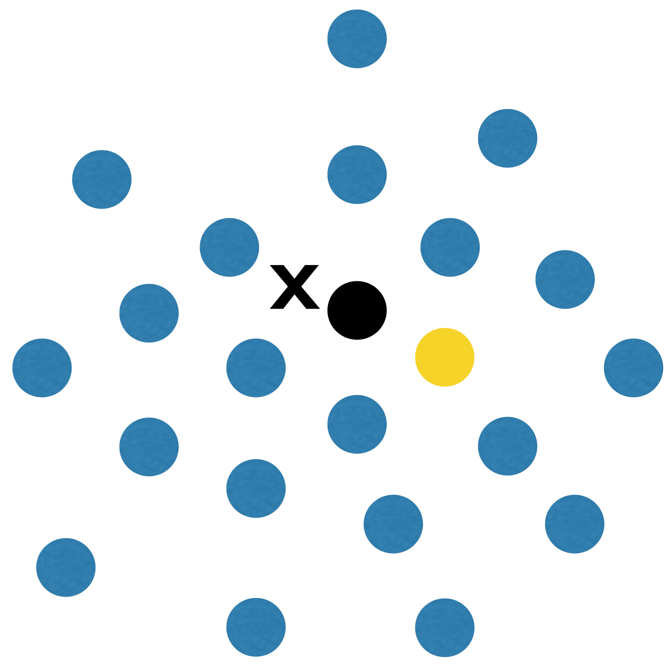
When is nearest neighbors robust to adversarial examples?

1-Nearest Neighbors

Theorem: If μ is continuous and if in a neighborhood of x, we have $\eta \in (0,1)$, then the robustness radius as x converges to 0 with growing n

Distributional robustness (and astuteness) is 0

Accuracy may be high



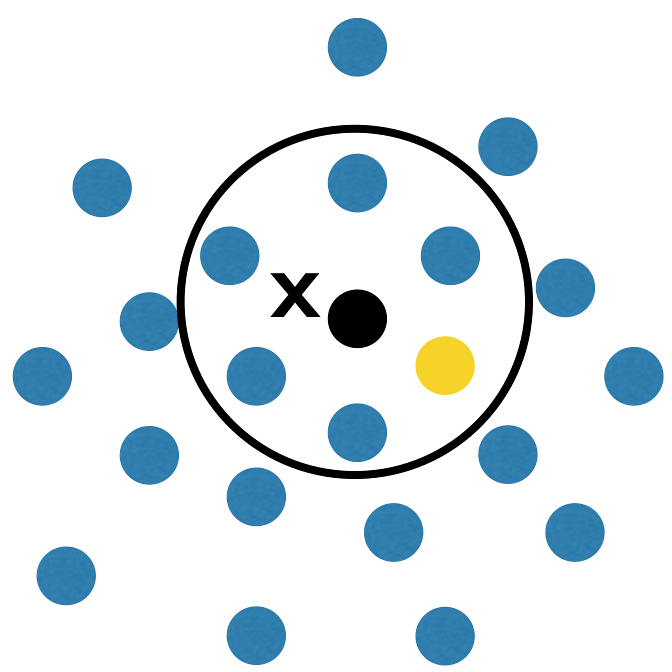
Proof Intuition

Theorem: If μ is continuous and if in a neighborhood of x, we have $\eta \in (0, 1)$, then the robustness radius as x converges to 0 with growing n

As n grows, more points in B(x, r)

If $\eta \in (0, 1)$, at least one of them z a different label than x

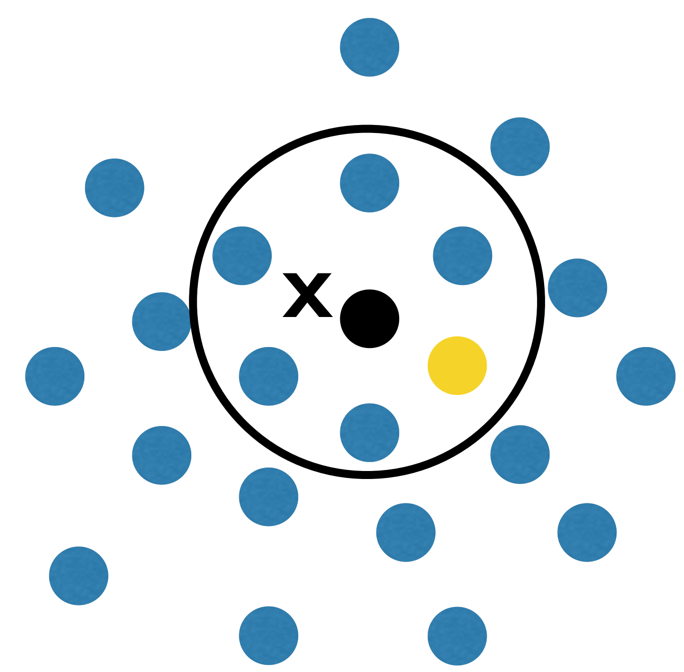
This z is an adversarial example



Constant k

Theorem: If μ is continuous and if in a neighborhood of x, we have $\eta \in (0, 1)$, then the robustness radius as x converges to 0 with growing n

Similar argument also holds for constant k



What about larger k?

Reminder: k-NN Accuracy

The risk of 1-NN converges to $\mathbb{E}_X[2\eta(X)(1-\eta(X))]$ as n grows (more than Bayes Optimal risk)

k NN is also inconsistent for constant k

If $k_n \to \infty$ and $k_n/n \to 0$ then, the risk of k_n-NN converges to the risk of the Bayes Optimal

k_n-NN Robustness

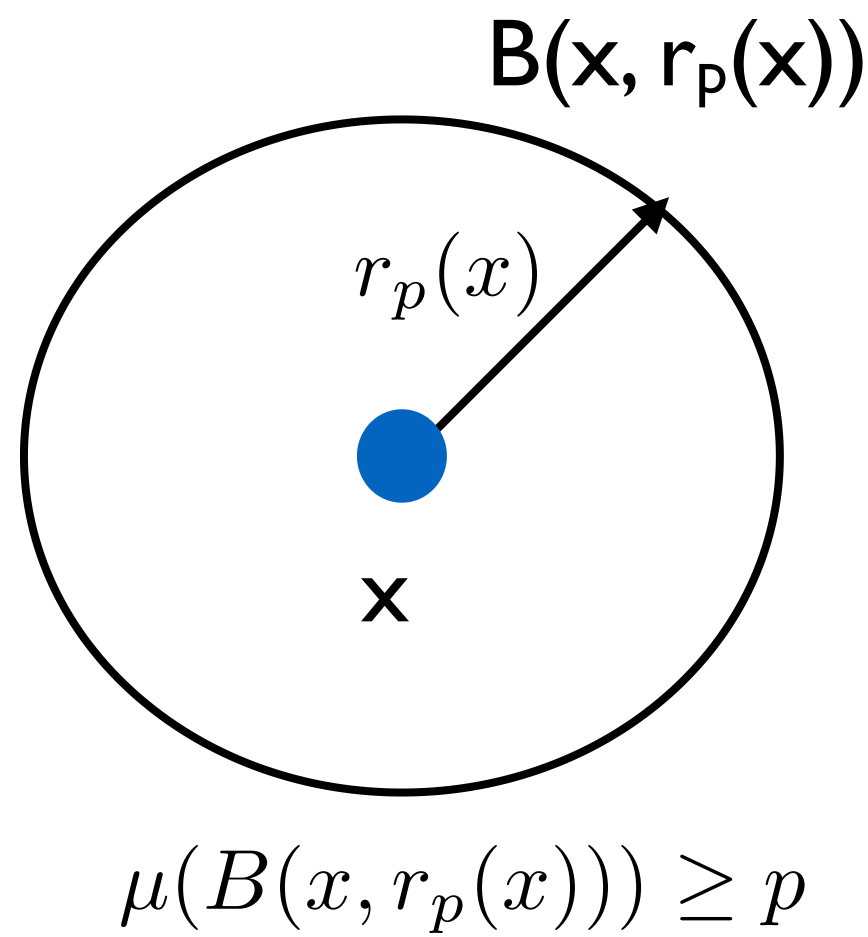
What can we expect? Robust where Bayes Optimal is robust

Where is the Bayes Optimal robust?

Some Notation

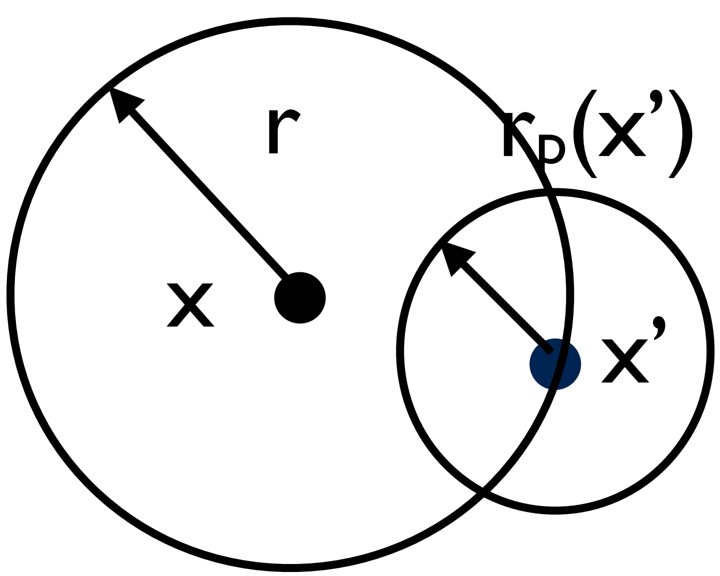
Probability-radius $r_p(x)$:

$$r_p(x) = \inf\{r | \mu(B(x, r)) \geq p\}$$



Robust Interiors

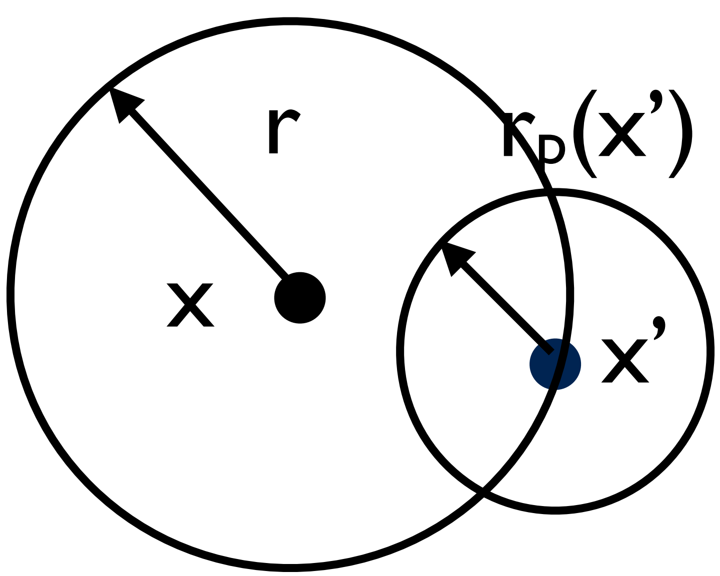
Positive: $$\mathcal{X}^+_{r,p,\Delta} = \{x | \forall x' \in B(x,r), \forall x'' \in B(x', r_p(x')), \eta(x'') > 1/2 + \Delta\}$$



Robust Interiors

Positive: $$\mathcal{X}^{+}_{r,p,\Delta} = \{x | \forall x' \in B(x,r), \forall x'' \in B(x', r_p(x')), \eta(x'') > 1/2 + \Delta\}$$

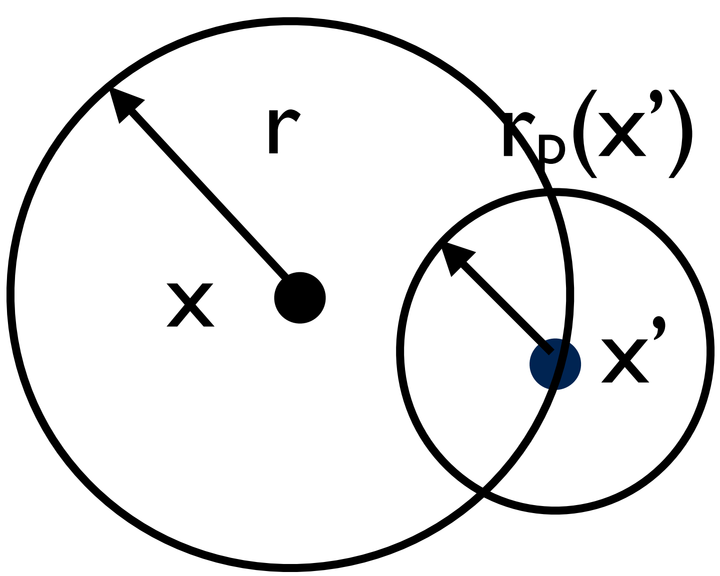
Negative: $$\mathcal{X}^{-}_{r,p,\Delta} = \{x | \forall x' \in B(x,r), \forall x'' \in B(x', r_p(x')), \eta(x'') < 1/2 - \Delta\}$$



Robust Interiors

Positive: $$\mathcal{X}^{+}_{r,p,\Delta} = \{x | \forall x' \in B(x,r), \forall x'' \in B(x', r_p(x')), \eta(x'') > 1/2 + \Delta\}$$

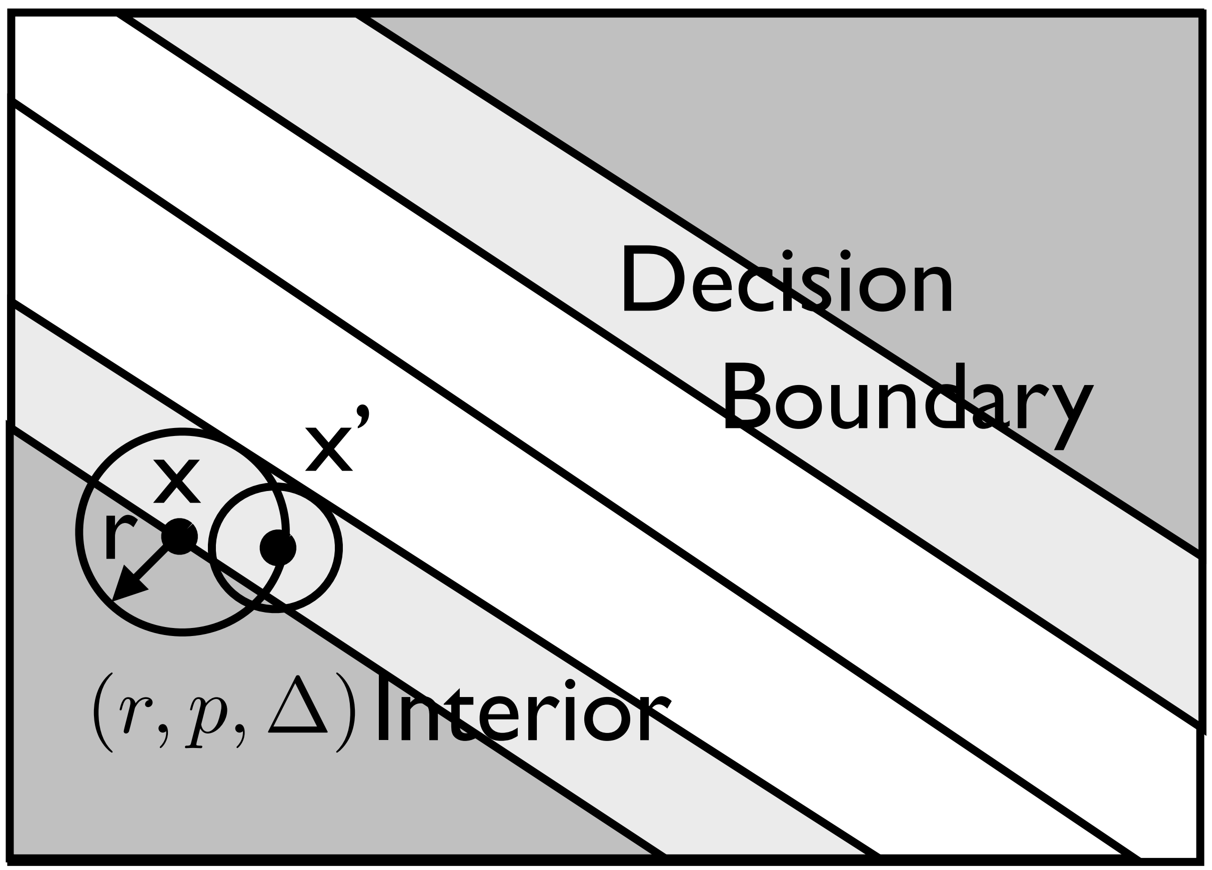
Negative: $$\mathcal{X}^{-}_{r,p,\Delta} = \{x | \forall x' \in B(x,r), \forall x'' \in B(x', r_p(x')), \eta(x'') < 1/2 - \Delta\}$$



(r,p,Δ) **-Interiors =**

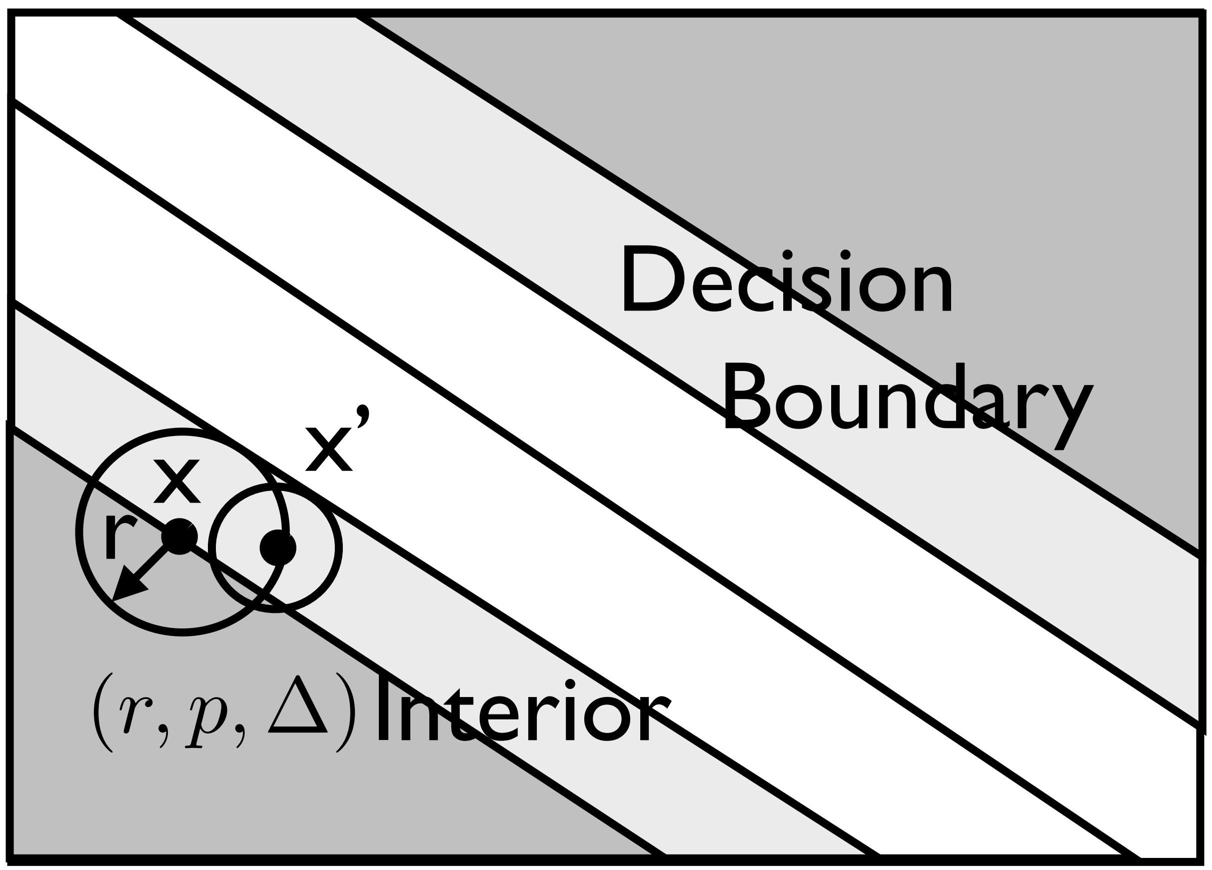
Positive + Negative

Where is Bayes Optimal Robust?



Bayes Optimal has robustness radius r in $\mathcal{X}^{+}_{r,0,0} \cup \mathcal{X}^{-}_{r,0,0}$

Where is Bayes Optimal Robust?



Bayes Optimal has robustness radius r in $\mathcal{X}^+_{r,0,0} \cup \mathcal{X}^-_{r,0,0}$

Astuteness of Bayes Optimal at radius r is

$$\mathbb{E}_X[\eta(x)1(x \in \mathcal{X}^+_{r,0,0}) \\ +(1-\eta(x))1(x \in \mathcal{X}^-_{r,0,0})]$$

Robustness of k_n-NN

Theorem: Let $\Delta_n \to 0$. If $k_n \geq \sqrt{dn \log n}/\Delta_n$ and $p_n = \frac{k_n}{n}(1 + o(1))$ then w.h.p k_n-nearest neighbors has robustness radius at least r in $\mathcal{X}^+_{r,p_n,\Delta_n} \cup \mathcal{X}^-_{r,p_n,\Delta_n}$

Robustness of k_n-NN

Theorem: Let $\Delta_n \to 0$. If $k_n \geq \sqrt{dn \log n}/\Delta_n$ and $p_n = \frac{k_n}{n}(1 + o(1))$ then w.h.p k_n-nearest neighbors has robustness radius at least r in $\mathcal{X}^+_{r,p_n,\Delta_n} \cup \mathcal{X}^-_{r,p_n,\Delta_n}$

Growth of k_n much faster than required for accuracy

Robustness of k_n-NN

Theorem: Let $\Delta_n \to 0$. If $k_n \geq \sqrt{dn \log n}/\Delta_n$ and $p_n = \dfrac{k_n}{n}(1 + o(1))$ then w.h.p k_n-nearest neighbors has robustness radius at least r in $\mathcal{X}^+_{r,p_n,\Delta_n} \cup \mathcal{X}^-_{r,p_n,\Delta_n}$

Growth of k_n much faster than required for accuracy

If $p_n = k_n/n \to 0$, and $\Delta_n \to 0$, then

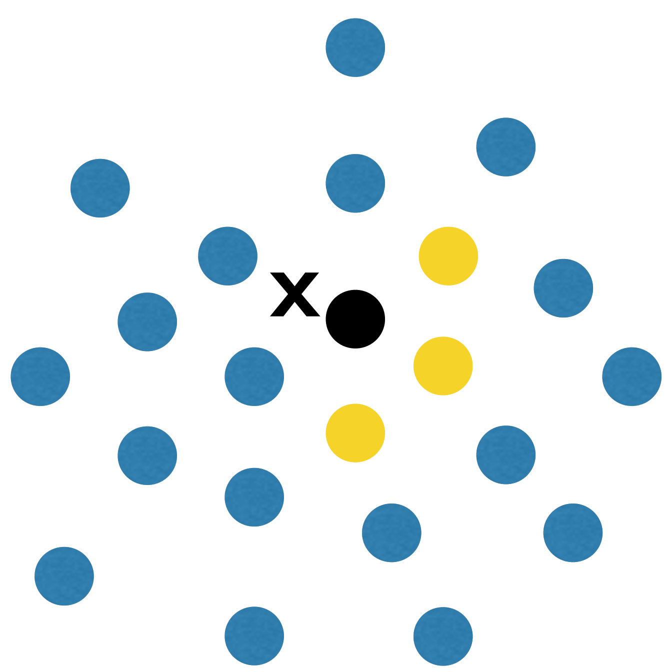
$$\mathcal{X}^+_{r,p_n,\Delta_n} \cup \mathcal{X}^-_{r,p_n,\Delta_n} \to \mathcal{X}^+_{r,0,0} \cup \mathcal{X}^-_{r,0,0}$$

(Robustness region of Bayes Optimal)

Proof Intuition

For $k_n \geq \sqrt{dn \log n}/\Delta_n$, by uniform convergence, for all x,

$$\frac{k_n}{n}(1 - o(1)) \leq \mu(B(x, \|x - X^{(k_n)}\|)) \leq \frac{k_n}{n}(1 + o(1))$$



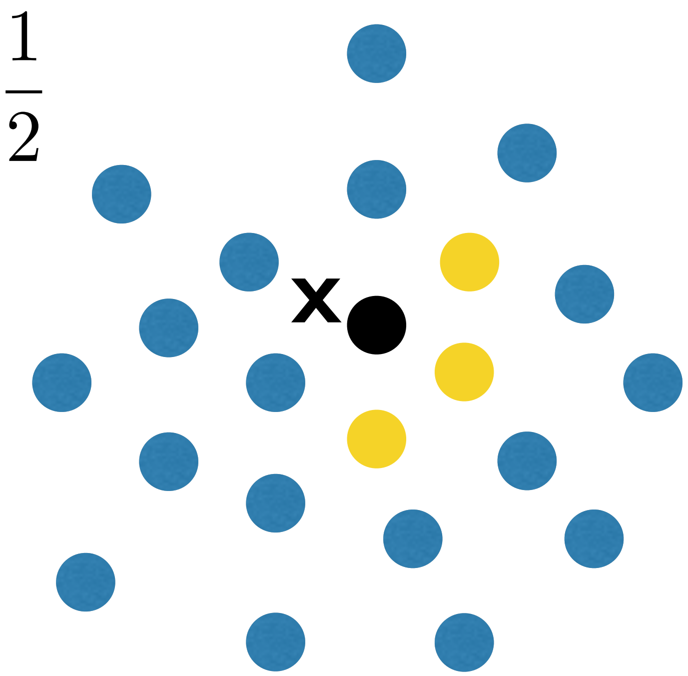
Proof Intuition

For $k_n \geq \sqrt{dn \log n}/\Delta_n$, by uniform convergence, for all x,

$$\frac{k_n}{n}(1 - o(1)) \leq \mu(B(x, \|x - X^{(k_n)}\|)) \leq \frac{k_n}{n}(1 + o(1))$$

If $x' \in \mathcal{X}^+_{r,p_n,\Delta_n}$, for all $x'' \in B(x', X^{(k_n)}(x')), \eta(x'') > 1/2 + \Delta_n$

By uniform convergence, $\frac{1}{k_n} \sum_i Y^{(i)}(x'') > \frac{1}{2}$



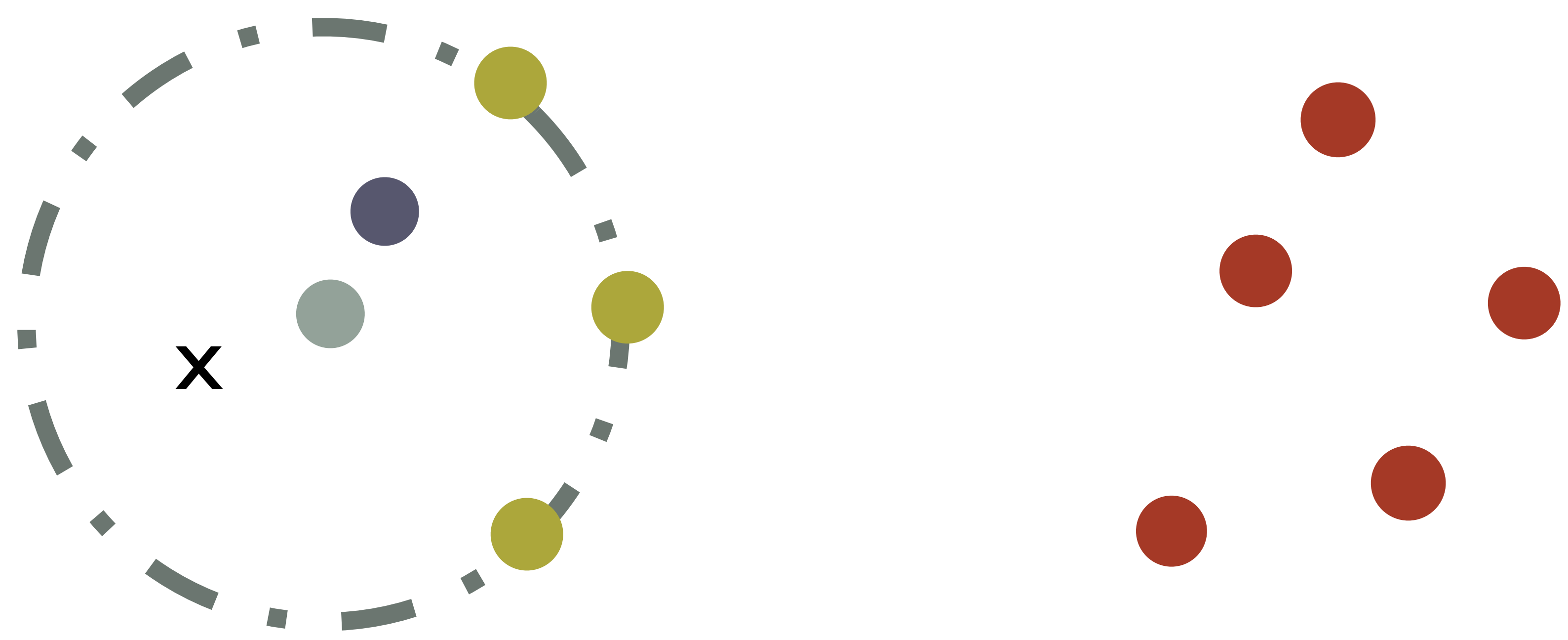
Can we get robustness for 1 NN?

Yes, through a modified algorithm….

When is Nearest Neighbors Robust?

1-nearest neighbor is robust at x if:

- points with different labels are well-separated
- x is close to a point with the same label



Algorithm Idea

- Remove a subset of training data such that differently labeled points are far apart

- Do 1-nearest neighbors on remaining data

Algorithm Idea

- Remove a subset of training data such that differently labeled points are far apart

- Do 1-nearest neighbors on remaining data

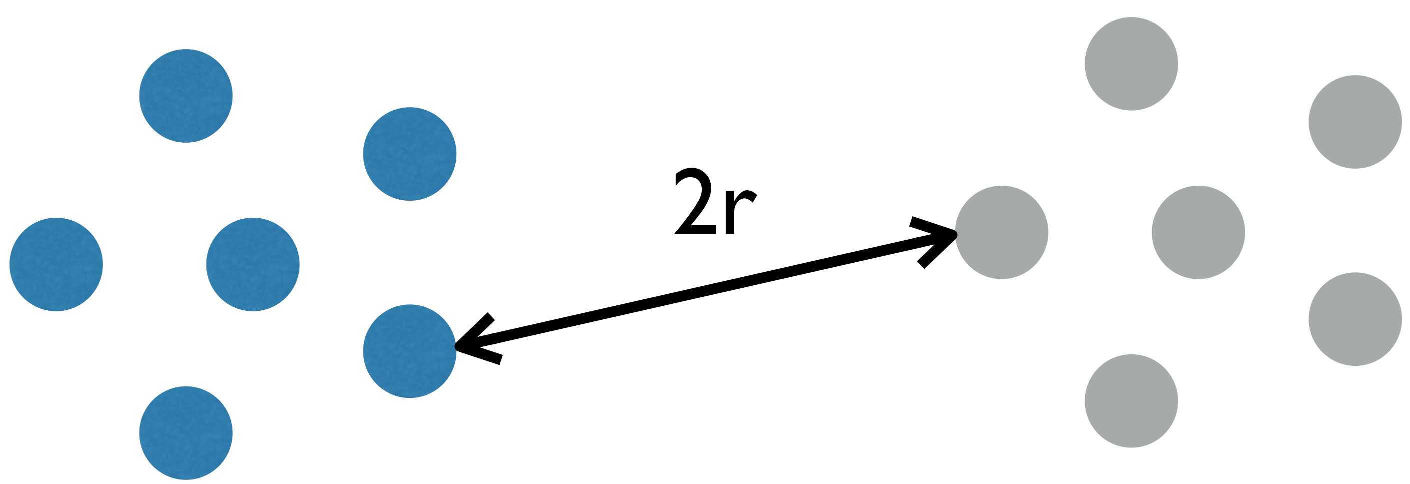
Which points to remove?

Keep points with confident labels, and a maximal subset of the rest

r-separation

A set of points $\{(x_i, y_i)\}$ is r-separated if

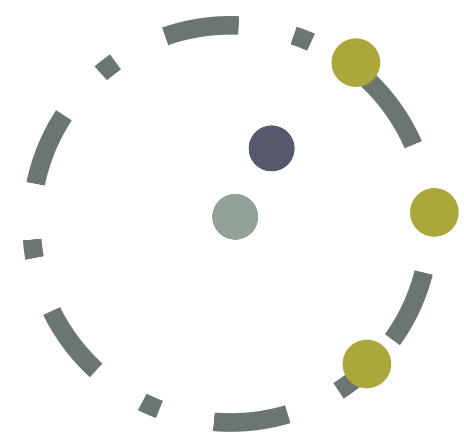
$$y_i \neq y_j \implies \|x_i - x_j\| \geq 2r$$



Getting Confident Labels

Input: x, training data of size n, parameters δ, Δ

$$k_n = 3\log(2n/\delta)/\Delta^2$$

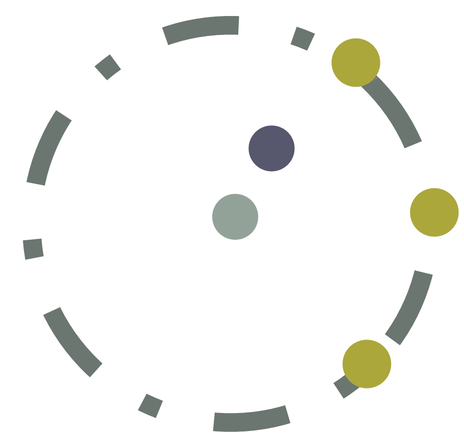


Getting Confident Labels

Input: x, training data of size n, parameters δ, Δ

$$k_n = 3\log(2n/\delta)/\Delta^2$$

$$Y = \frac{1}{k_n}\sum_{i=1}^{k_n} Y^{(i)}(x)$$



Getting Confident Labels

Input: x, training data of size n, parameters δ, Δ

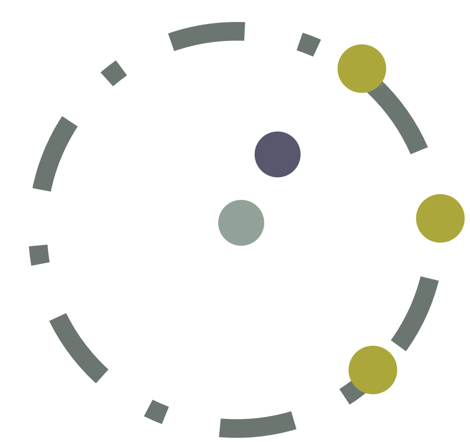
$$k_n = 3\log(2n/\delta)/\Delta^2$$

$$Y = \frac{1}{k_n}\sum_{i=1}^{k_n} Y^{(i)}(x)$$

If $Y \in \left[\frac{1}{2} - \Delta, \frac{1}{2} + \Delta\right]$ then

return "Don't Know"

Else return round(Y)



Full Algorithm

Input: x, training data S, radius r, parameters δ, Δ

For all i: $f(x_i)$ = ConfidentLabel (x_i, S, δ, Δ)

Full Algorithm

Input: x, training data S, radius r, parameters δ, Δ

For all i: $f(x_i)$ = ConfidentLabel (x_i, S, δ, Δ)

T = emptyset

For all i: if $f(x_i) = y_i$ and $f(x_i) = f(x_j)$ for all x_j in $B(x_i, r)$ then

Add (x_i, y_i) to T

Full Algorithm

Input: x, training data S, radius r, parameters δ, Δ

For all i: $f(x_i)$ = ConfidentLabel (x_i, S, δ, Δ)

T = emptyset

For all i: if $f(x_i) = y_i$ and $f(x_i) = f(x_j)$ for all x_j in $B(x_i, r)$ then

Add (x_i, y_i) to T

Return the largest r-separated subset of S that contains T as training data for nearest neighbor

When is this algorithm robust?

Theorem: Fix δ, Δ_n, and let $k_n = 3\log(n/2\delta)/\Delta_n^2$, and $p_n = \frac{k_n}{n}(1 + \Theta(\sqrt{d/k_n}))$. For a parameter t, define a set X_r:

$$X_R = \left\{ x | x \in \mathcal{X}^+_{r+t,p_n,\Delta_n} \cup \mathcal{X}^-_{r+t,p_n,\Delta_n}, \mu(B(x,t)) \geq Cd\log n/n \right\}$$

Whp, algorithm has robustness radius at least r - 2t on X_R

When is this algorithm robust?

Theorem: Fix δ, Δ_n, and let $k_n = 3\log(n/2\delta)/\Delta_n^2$, and $p_n = \frac{k_n}{n}(1 + \Theta(\sqrt{d/k_n}))$. For a parameter t, define a set X_r :

$$X_R = \left\{ x | x \in \mathcal{X}^+_{r+t,p_n,\Delta_n} \cup \mathcal{X}^-_{r+t,p_n,\Delta_n}, \mu(B(x,t)) \geq Cd\log n/n \right\}$$

Whp, algorithm has robustness radius at least r - 2t on X_R

X_R is a high density subset of $\mathcal{X}^+_{r+t,p_n,\Delta_n} \cup \mathcal{X}^-_{r+t,p_n,\Delta_n}$

When is this algorithm robust?

Theorem: Fix δ, Δ_n, and let $k_n = 3\log(n/2\delta)/\Delta_n^2$, and $p_n = \frac{k_n}{n}(1+\Theta(\sqrt{d/k_n}))$. For a parameter t, define a set X_r:

$$X_R = \left\{ x | x \in \mathcal{X}^+_{r+t,p_n,\Delta_n} \cup \mathcal{X}^-_{r+t,p_n,\Delta_n}, \mu(B(x,t)) \geq Cd\log n/n \right\}$$

Whp, algorithm has robustness radius at least r - 2t on X_R

X_R is a high density subset of $\mathcal{X}^+_{r+t,p_n,\Delta_n} \cup \mathcal{X}^-_{r+t,p_n,\Delta_n}$

As t, p_n, $\Delta_n \to 0$, $\quad \mathcal{X}^+_{r+t,p_n,\Delta_n} \cup \mathcal{X}^-_{r+t,p_n,\Delta_n} \to \mathcal{X}^+_{r,0,0} \cup \mathcal{X}^-_{r,0,0}$

(robust region of Bayes Opt)

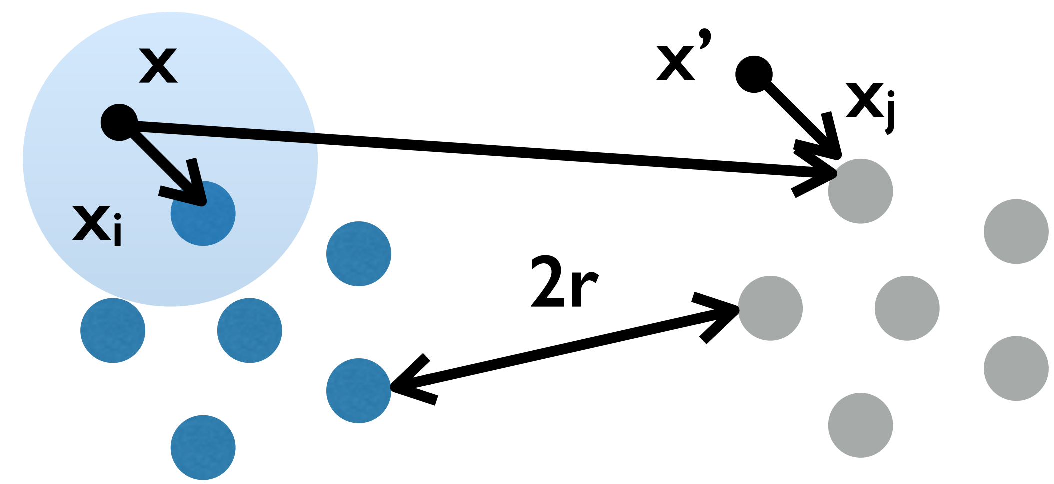
Proof Intuition

Let $x_i \in \mathcal{X}^{+}_{r,p_n,\Delta_n} \cup \mathcal{X}^{-}_{r,p_n,\Delta_n}$ and $y_i = 1(\eta(x) > 1/2)$

From property of k_n, (x_i, y_i) gets added to T

If x is in X_R, by uniform convergence, there is an (x_i, y_i) in S and B(x, t). This (x_i, y_i) will get added to the final training set

Since T is r-separated, any x_j with a different y_j will be at least 2r away from x_i. Triangle inequality gives radius r - 2t.



How does it work?

Experiments: Details

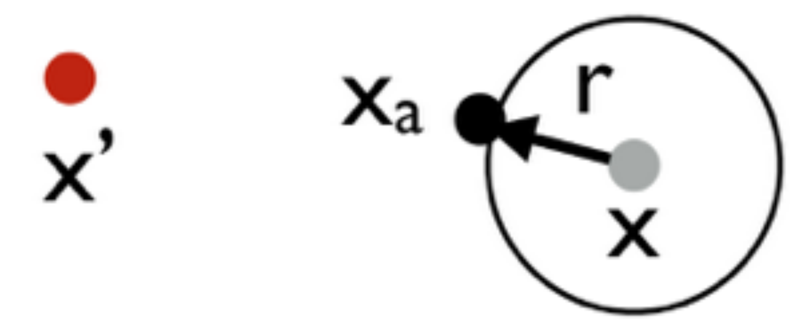
Baselines:

- StandardNN: Standard 1-NN using full training set
- RobustNN: Our method
- ATNN: Adversarially-trained 1-NN, dataset augmented using corresponding attack
- ATNN-all: Adversarially-trained 1-NN, dataset augmented using all attack methods

Datasets: Half-moon, MNIST 1v7, UCI Abalone

White-box Attacks

Direct Attack [ABEF16]:



Find closest x' in training set with different label

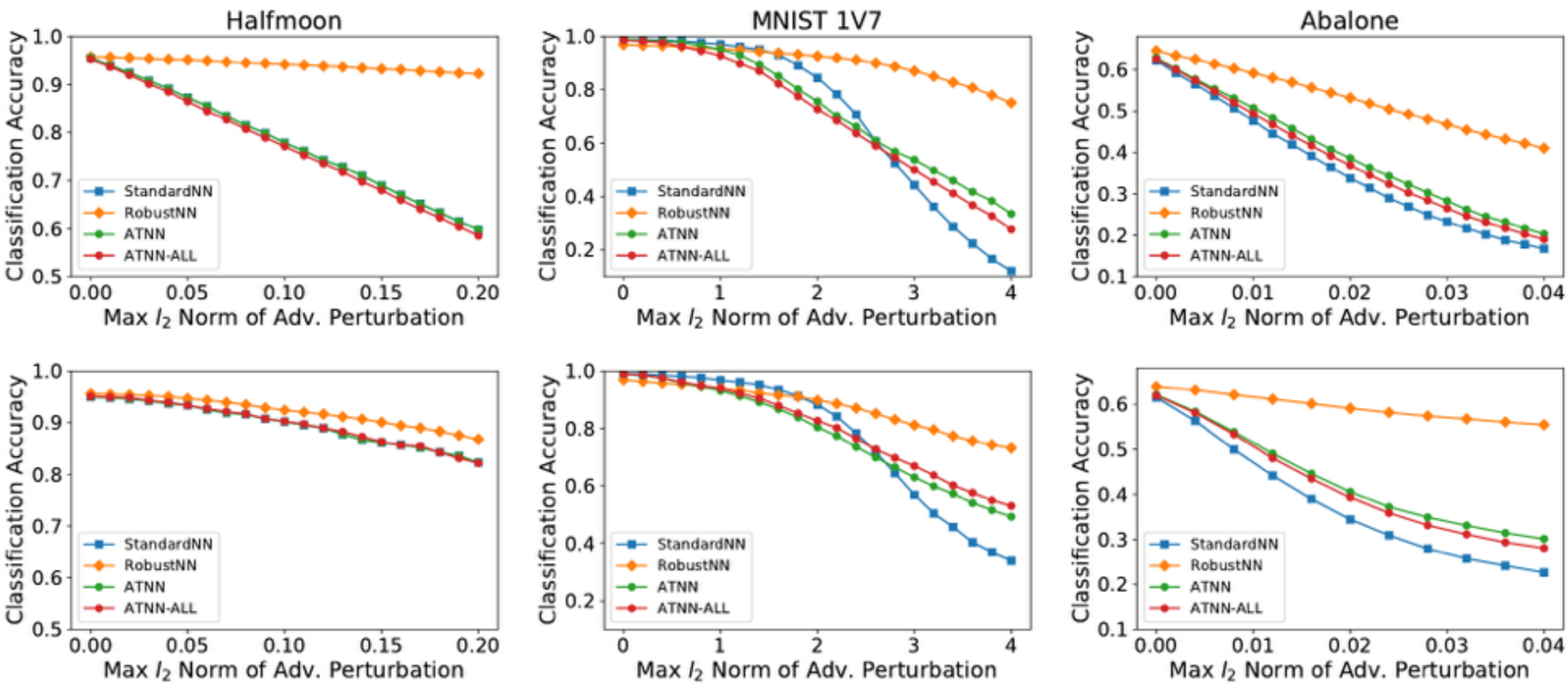
Move a distance r towards x'

Substitute Attack [PMG16]:

Find kernel classifier (soft nearest neighbors)

Attack with standard gradient-based methods

White-Box Attack Results



Top: Direct attacks, Bottom: Kernel substitute

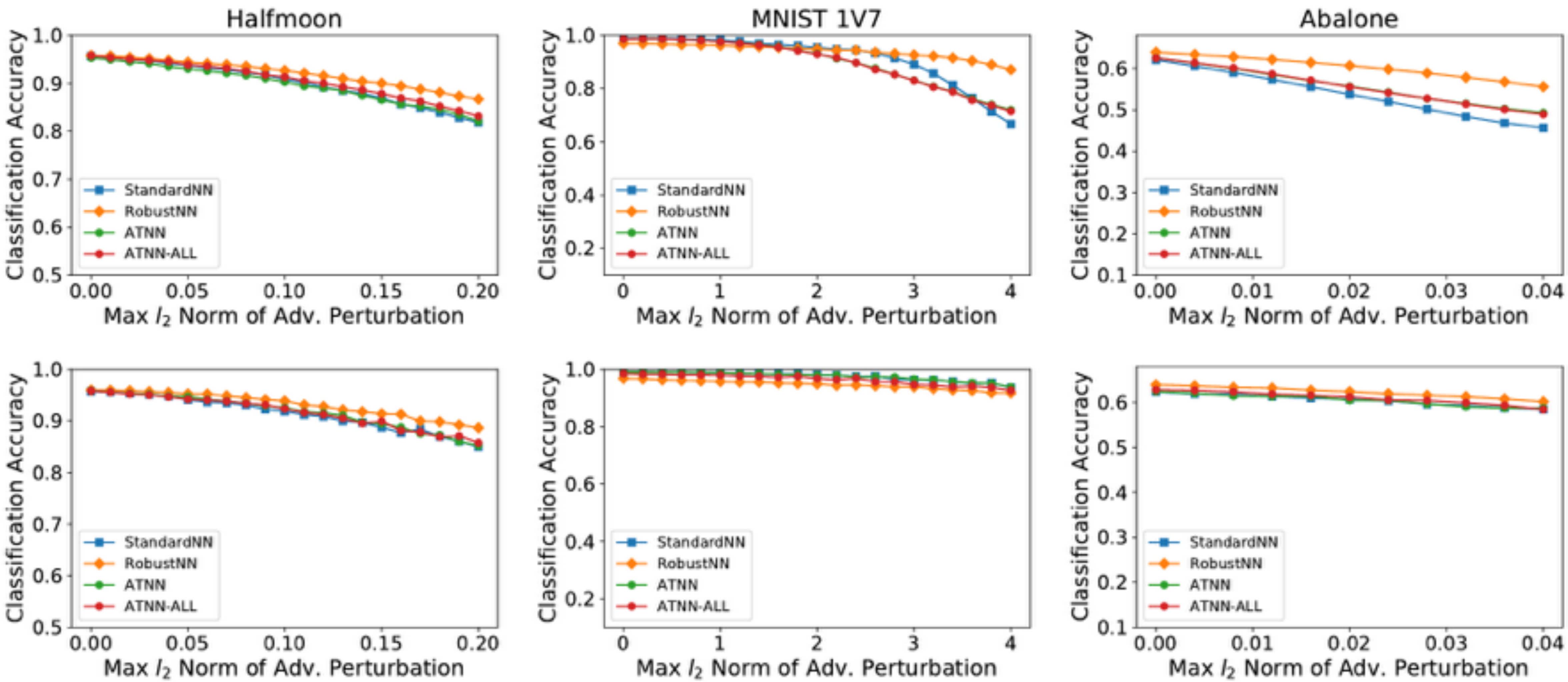
Black-box Attacks

Attack Method [PMGJ+17]:

Train substitute classifier by making queries to nearest neighbor

Return adversarial examples for substitute classifier

Black-Box Attack Results



Top: Kernel substitute, Bottom: Neural network substitute

Conclusion

- Proved robustness properties of nearest neighbors to adversarial examples
- New robust NN algorithm
- Experimental results

Tutorial Outline

- Adversarial Examples
  - A Statistical Learning Framework for Robustness
- Adversarial Examples for Nearest Neighbors
  - Small and large k
  - A Robust Modified Nearest Neighbor
- Beyond Nearest Neighbors
  - Generic Attacks
  - The r-Optimal Classifier
  - Experiments

Beyond Nearest Neighbors…

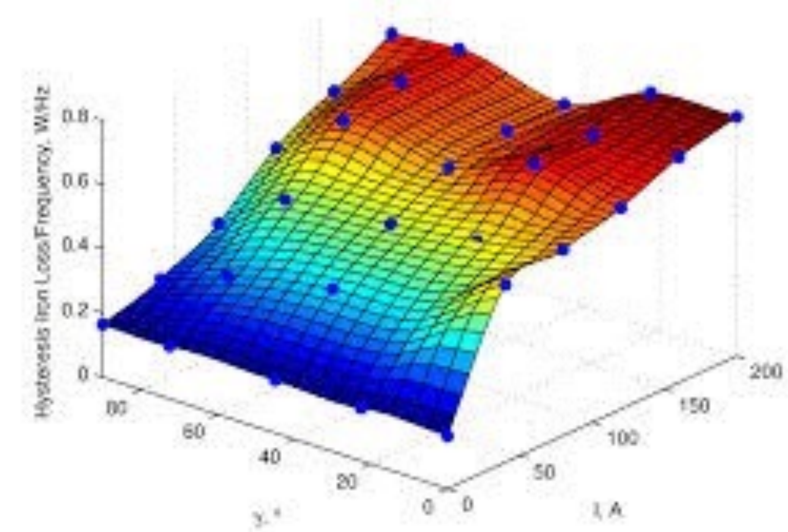
Can we get generic attacks and defenses for non-parametrics — NN, decision trees, RF?

Adversarial Examples for Parametric Methods

Model θ^* obtained by minimizing a loss function L

$$\theta^* = \min_{\theta} L(\theta, x, y)$$

(Most) Attacks: Gradient-based: Starting at x, do gradient ascent on the loss until label changes



Adversarial Examples for Parametric Methods

(Most) Defenses: Adversarial training (training with data augmented with adversarial examples).

[Goodfellow+14, Madry+17, many others..]

What about non-parametrics?

Can we get generic attacks and defenses for non-parametrics — NN, decision trees, RF?

Prior Work: Specific classifiers

- Nearest neighbors [Amsaleg+17, Wang+18]
- Decision trees [Kantchelian+16, Cheng+19]

What about non-parametrics?

Can we get generic attacks and defenses for non-parametrics — NN, decision trees, RF?

Challenges for generics:

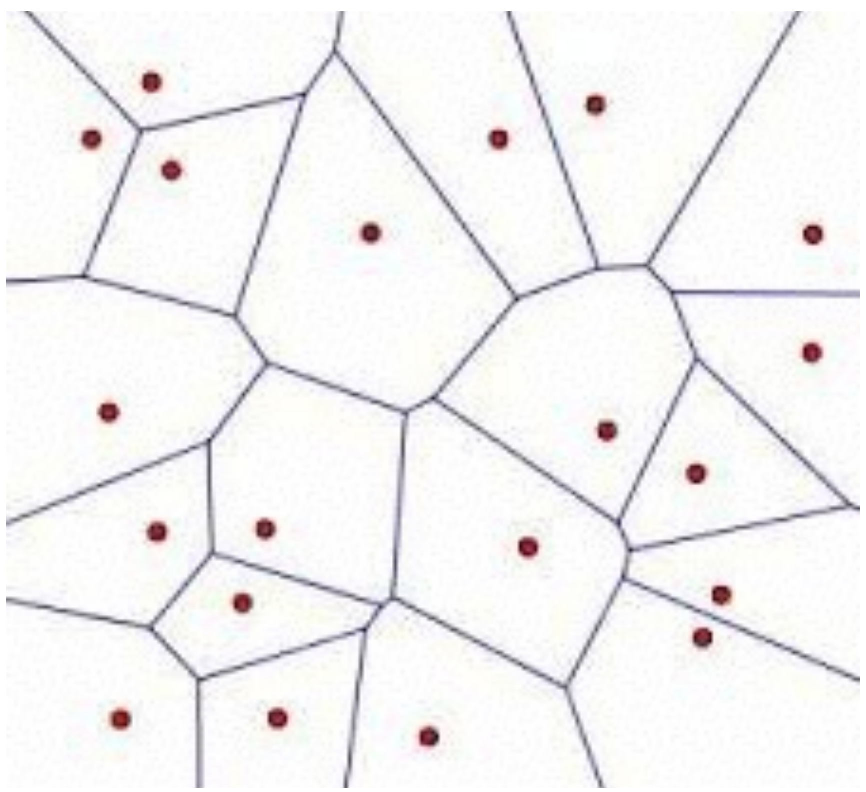
- Gradient-based attacks do not apply
- Adversarial training does not work well

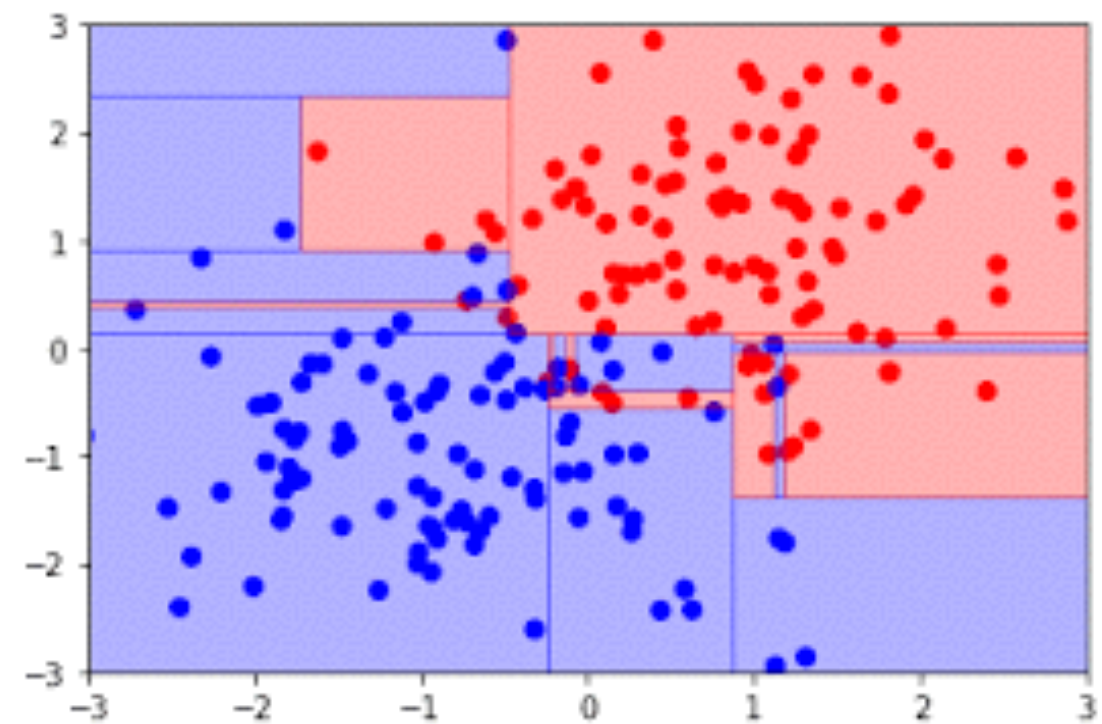
Talk Outline

- Generic Attacks
- A Limit Object
- A Generic Defense

Generic Attacks

Key Observation: Many non-parametrics are piece-wise constant on polyhedra

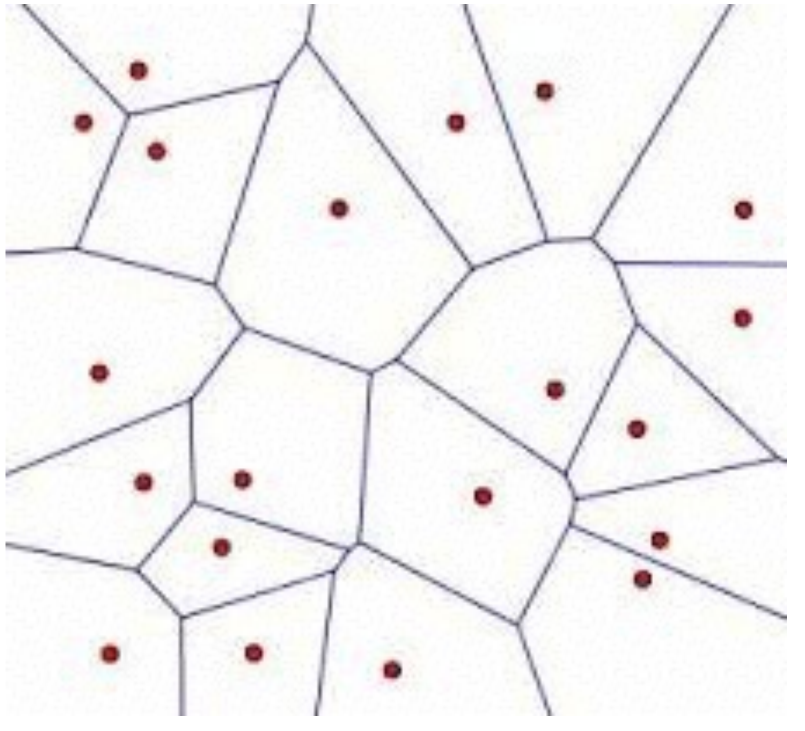




Example: 1 NN on Voronoi cells, decision trees on leaf nodes

Region-Based Attack

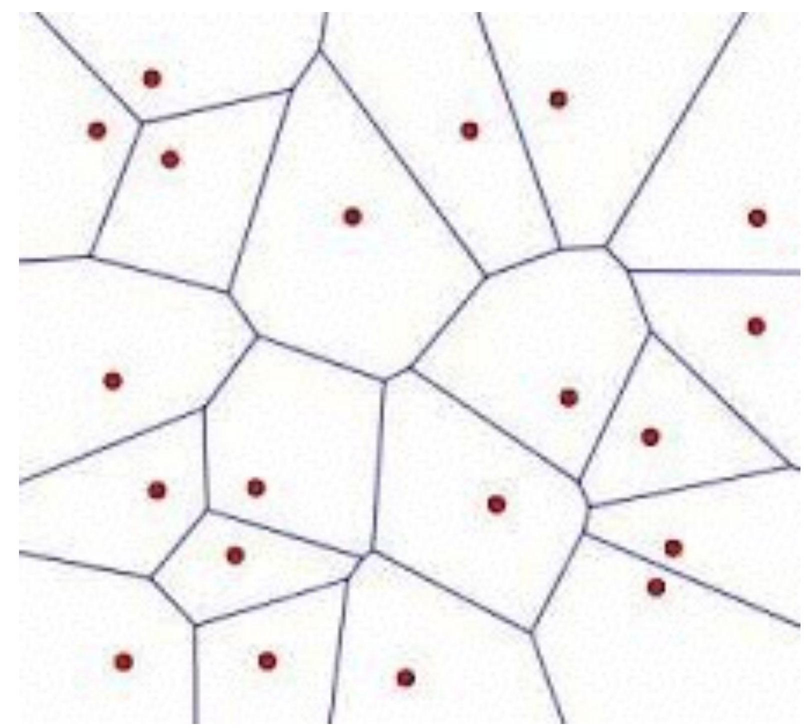
Key Observation: Many non-parametrics are piece-wise constant on polyhedra



Region-Based Attack

Key Observation: Many non-parametrics are piece-wise constant on polyhedra

Let the polyhedra be $P_1, \ldots, P_m$ with predicted labels $y_1, \ldots, y_m$



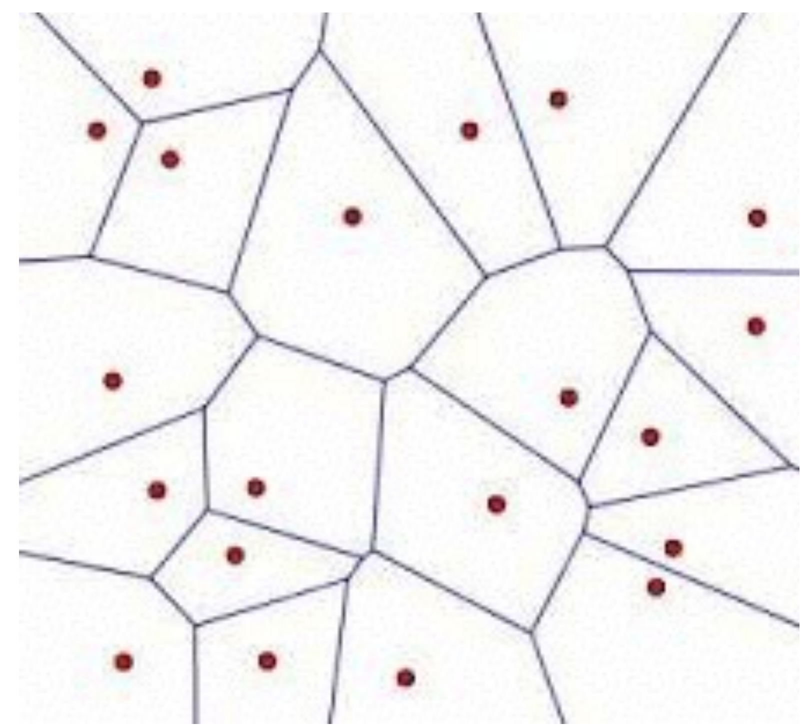
Region-Based Attack

Key Observation: Many non-parametrics are piece-wise constant on polyhedra

Let the polyhedra be $P_1, \ldots, P_m$ with predicted labels $y_1, \ldots, y_m$

Given x, find

$$\min_{i: f(\mathbf{x}) \neq y_i} \ \min_{\mathbf{z} \in P_i} \ \|\mathbf{x} - \mathbf{z}\|.$$



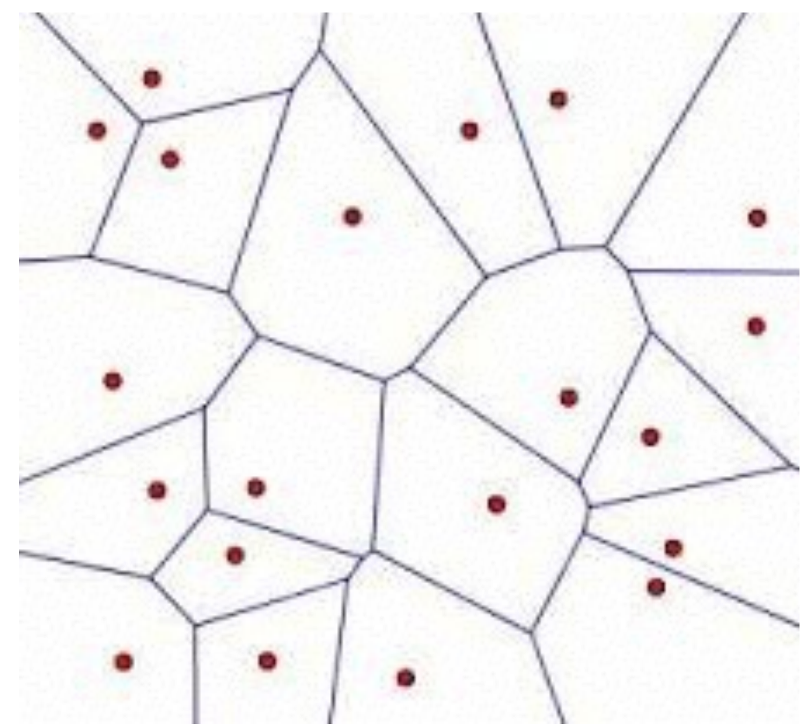
Region-Based Attack

Key Observation: Many non-parametrics are piece-wise constant on polyhedra

Let the polyhedra be $P_1, \ldots, P_m$ with predicted labels $y_1, \ldots, y_m$

Given x, find

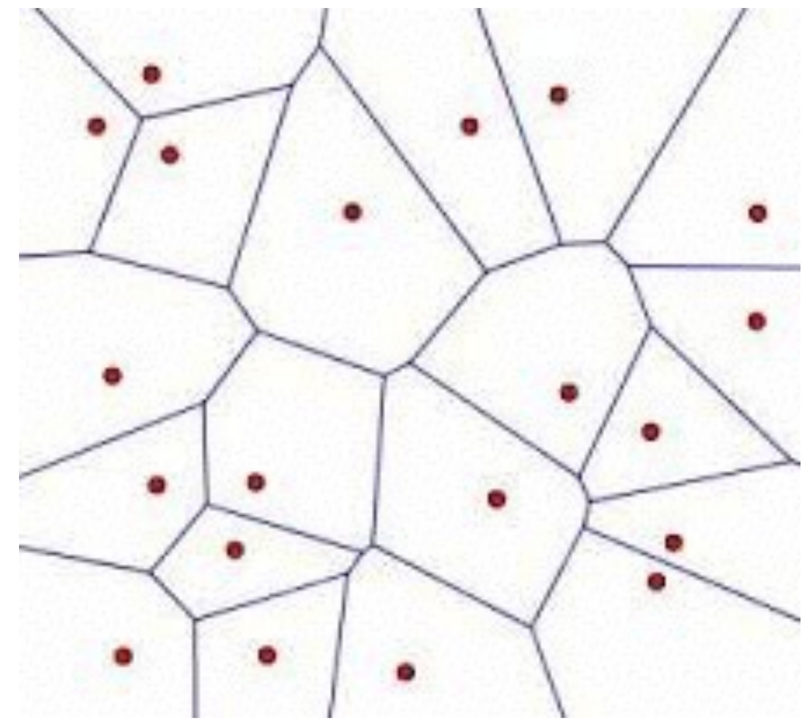
$$\min_{i: f(\mathbf{x}) \neq y_i} \min_{\mathbf{z} \in P_i} \|\mathbf{x} - \mathbf{z}\|.$$



Convex program - solution gives **optimal** attack

Approx Region Based Attack

Let the polyhedra be $P_1, \ldots, P_m$
with predicted labels $y_1, \ldots, y_m$



Given x, find

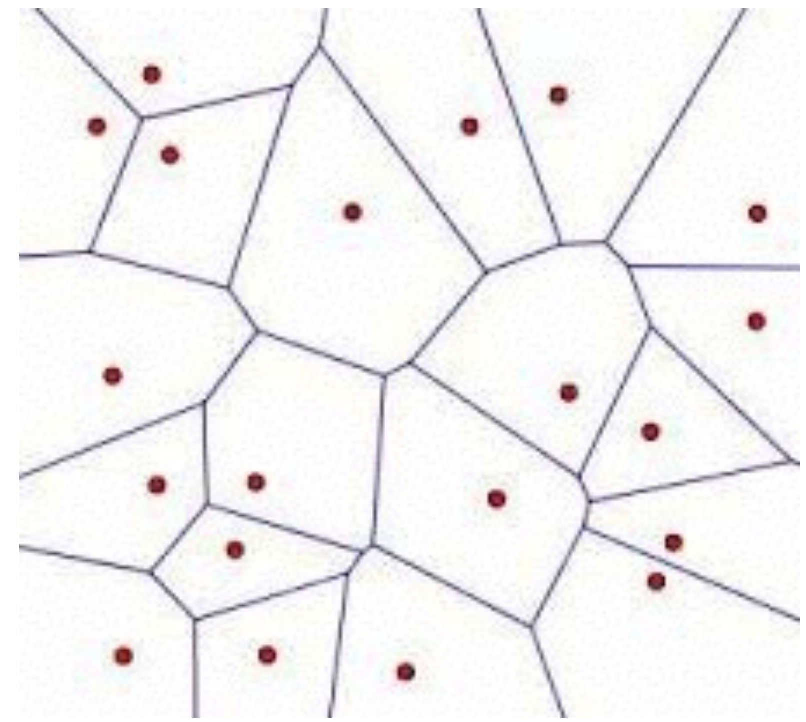
$$\min_{i: f(\mathbf{x}) \neq y_i} \min_{\mathbf{z} \in P_i} \|\mathbf{x} - \mathbf{z}\|.$$

Convex program!

Challenge: Too many polyhedra (about n^k for k-NN)

Approx Region Based Attack

Let the polyhedra be $P_1, \ldots, P_m$ with predicted labels $y_1, \ldots, y_m$



Given x, find

$$\min_{i: f(\mathbf{x}) \neq y_i} \; \min_{\mathbf{z} \in P_i} \; \|\mathbf{x} - \mathbf{z}\|.$$

Convex program!

Challenge: Too many polyhedra (about n^k for k-NN)

Solution: Search over P_i with L training points closest to x (lose optimality, but still valid)

What about defenses?

Beyond the Bayes Optimal…

Bayes Optimal maximizes accuracy
but not robustness

Is there a robustness analogue
to the Bayes Optimal?

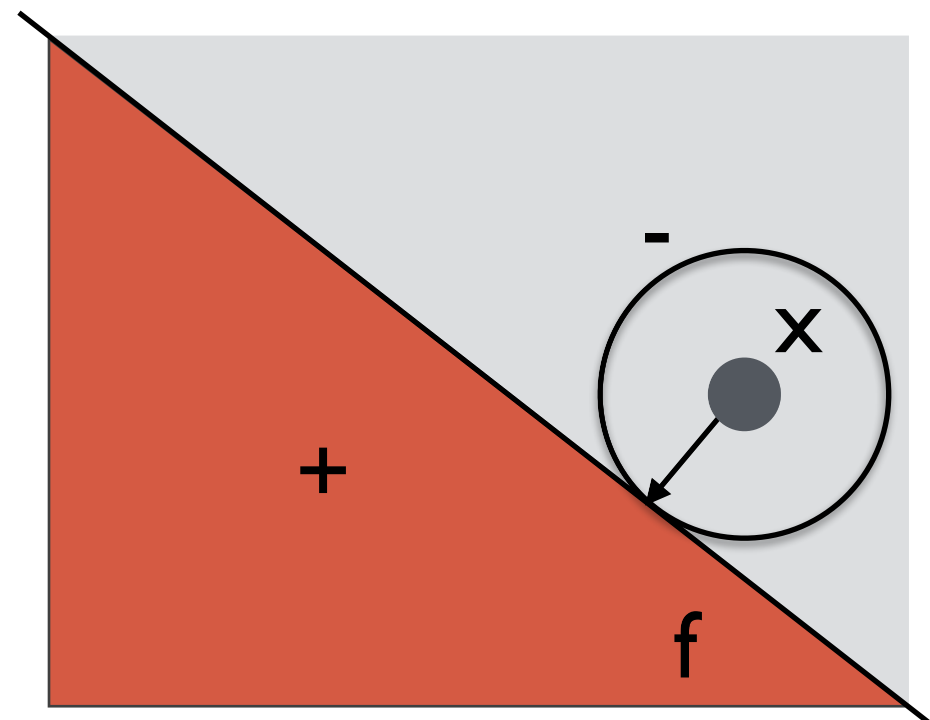
Recall: Astuteness

The astuteness of classifier f at radius r is defined as:

$$\mathrm{ast}(f, r) = \Pr(f(x) = y, \rho(f, x) \geq r)$$

Fraction of points
where f is robust and accurate

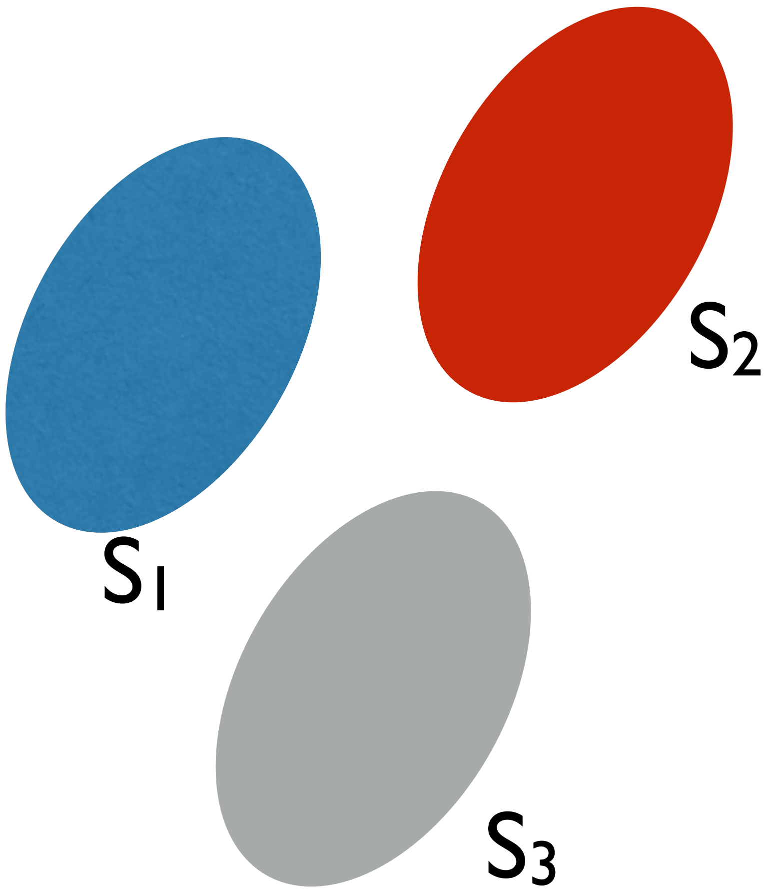
Goal of robust learning is
maximizing astuteness



Maximizing Astuteness

Given **robustness radius** r

Suppose classifier f predicts label j in S_j and is robust

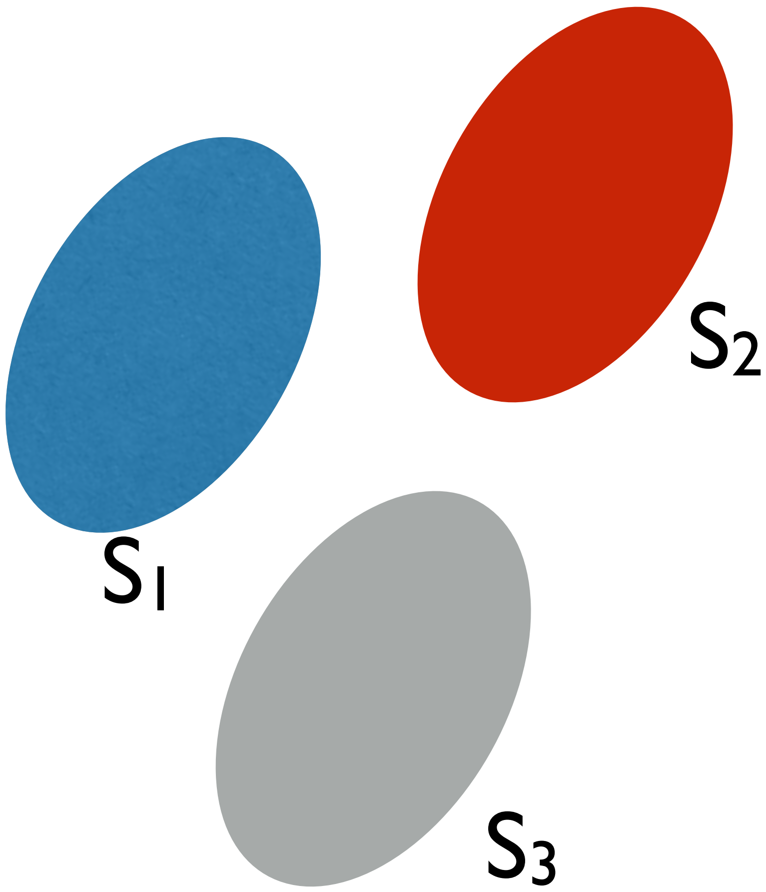


Maximizing Astuteness

Given **robustness radius** r

Suppose classifier f predicts label j in S_j and is robust

Then: $d(S_i, S_j) \geq 2r, j \neq i$



Maximizing Astuteness

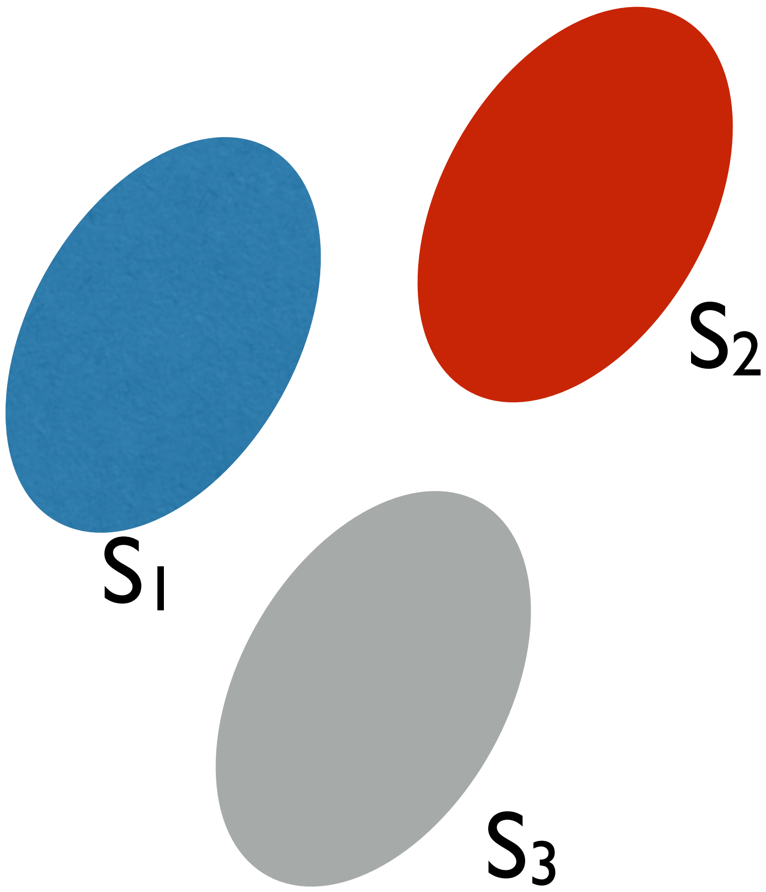
Given **robustness radius** r

Suppose classifier f predicts label j in S_j and is robust

Then: $d(S_i, S_j) \geq 2r, j \neq i$

Astuteness of f is:

$$\sum_{j=1}^{K} \int_{x \in S_j} \Pr(y = j|x)\mu(x)dx$$



…suggests the classifier

Given **robustness radius** r

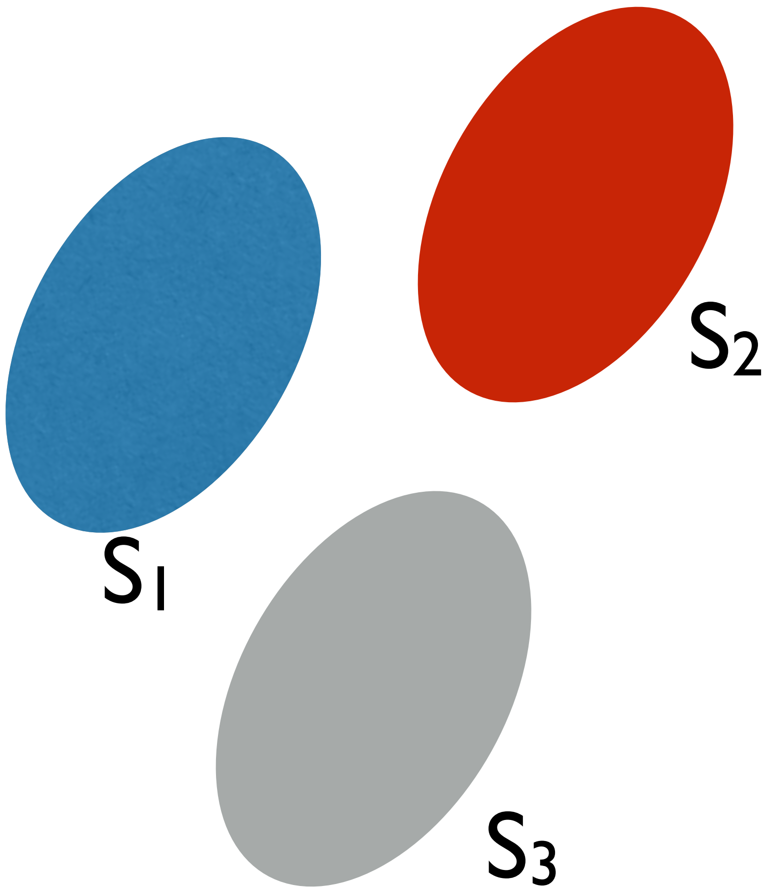
max$_{S_j}$ $$\sum_{j=1}^{K} \int_{x \in S_j} \Pr(y = j|x)\mu(x)dx$$

subject to:

$$d(S_i, S_j) \geq 2r, j \neq i$$

Prediction Rule:

Predict j if $d(x, S_j) \leq r$



How to get a finite-sample approximation?

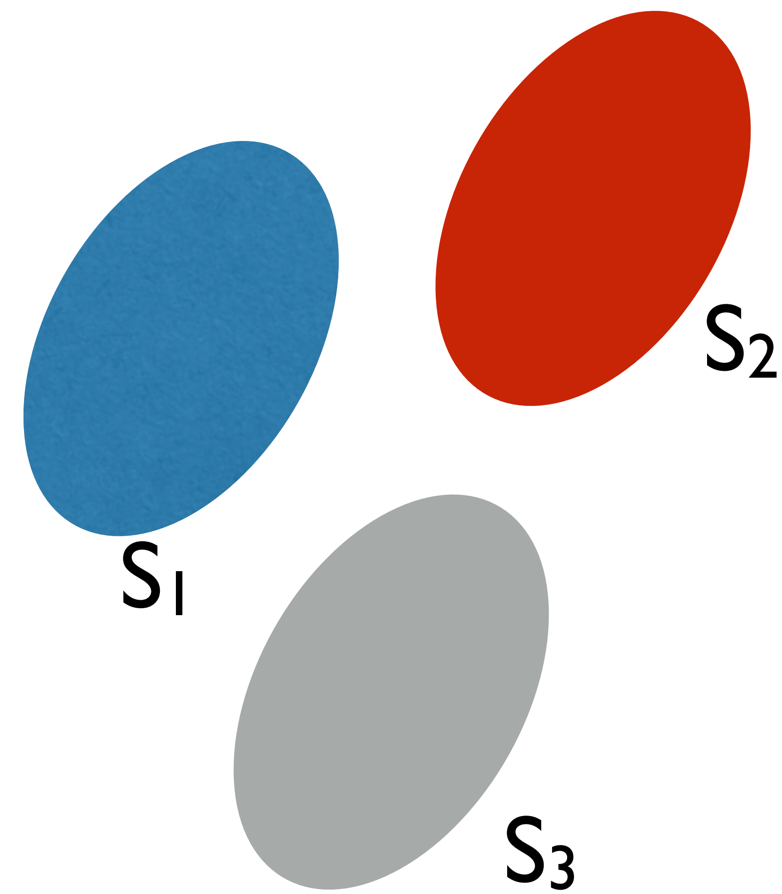
A finite sample approximation…

Given **robustness radius** r

$$\max_{S_j} \sum_{j=1}^{K} \int_{x \in S_j} \Pr(y = j | x) \mu(x) dx$$

subject to:

$$d(S_i, S_j) \geq 2r, j \neq i$$



Idea: Represent each S_j by a set of training samples…

A finite sample approximation…

Given **robustness radius** r

$$\max_{S_j} \sum_{j=1}^{K} \int_{x \in S_j} \Pr(y = j|x)\mu(x)dx \quad \rightarrow \quad \max_{S_j} \sum_{j=1}^{K} \sum_{x_i \in S_j} 1(y_i = j)$$

subject to:

$$d(S_i, S_j) \geq 2r, j \neq i$$

subject to:

$$d(S_i, S_j) \geq 2r, j \neq i$$

A finite sample approximation…

Given **robustness radius** r

max$_{S_j}$ $\sum_{j=1}^{K} \int_{x \in S_j} \Pr(y = j|x)\mu(x)dx$ ➔ max$_{S_j}$ $\sum_{j=1}^{K} \sum_{x_i \in S_j} 1(y_i = j)$

subject to:

$$d(S_i, S_j) \geq 2r, j \neq i$$

subject to:

$$d(S_i, S_j) \geq 2r, j \neq i$$

Solution: Maximal subset of training samples where points with different labels are 2r or more apart

How to solve this?

How to solve this?

Binary - reduces to maximum bipartite matching

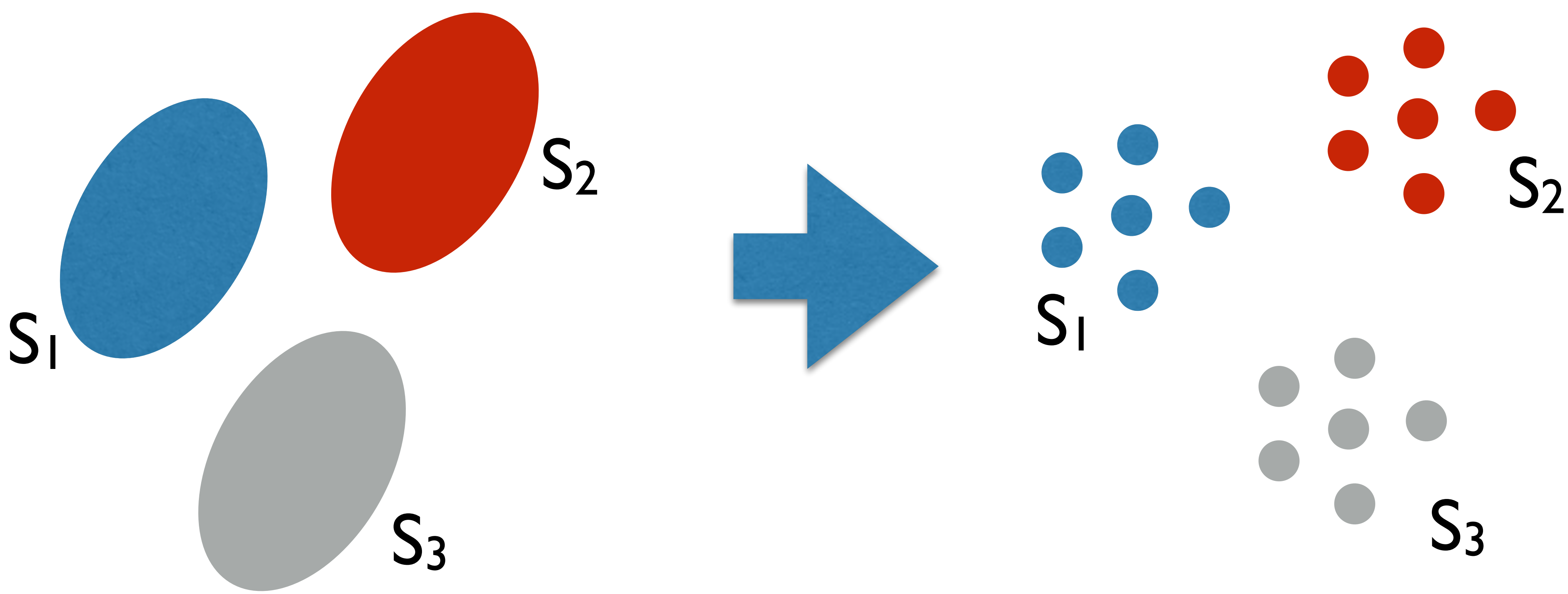
K-ary - reduces to independent set, greedy algorithm

Note: Different from [Wang+18] - no confident points

Algorithm: Adversarial Pruning

1. Find maximal subset of training samples where points with different labels are 2r or more apart

2. Build classifier (NN, decision tree, RF) on it



Evaluation

- How good is the Region-Based Attack?
- How effective is Adversarial Pruning as a defense?
- Does Adversarial Pruning work for parametric models as well?

Attack Metric

Empirical Robustness of attack A on f at x = Distance to closest adversarial example produced by A on f at x

Attack Metric: Average empirical robustness over examples where f is accurate

Smaller means better attack

For the optimal attack, this is the average robustness radius

Baselines

Classifiers: Nearest Neighbors (1NN), 3 Nearest Neighbors (3NN), Decision Trees (DT), Random Forests (RF)

9 datasets

Attacks:

Black box attack (Cheng+19) (for all)

Direct attack (for NN)

Kernel substitution attack (for NN)

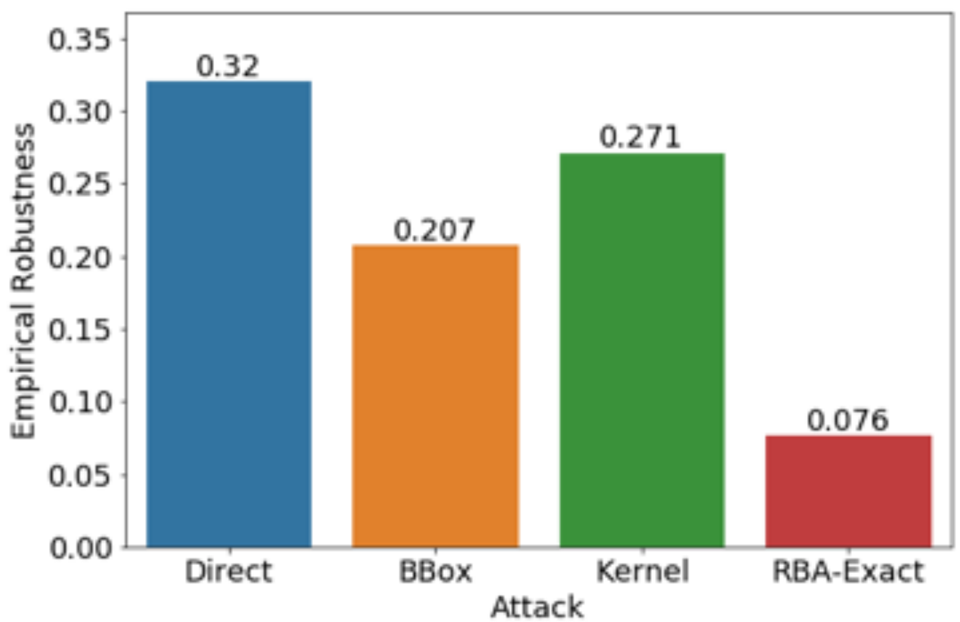
Papernot's attack (for DT)

Exact Region-based attack (for 1NN, DT)

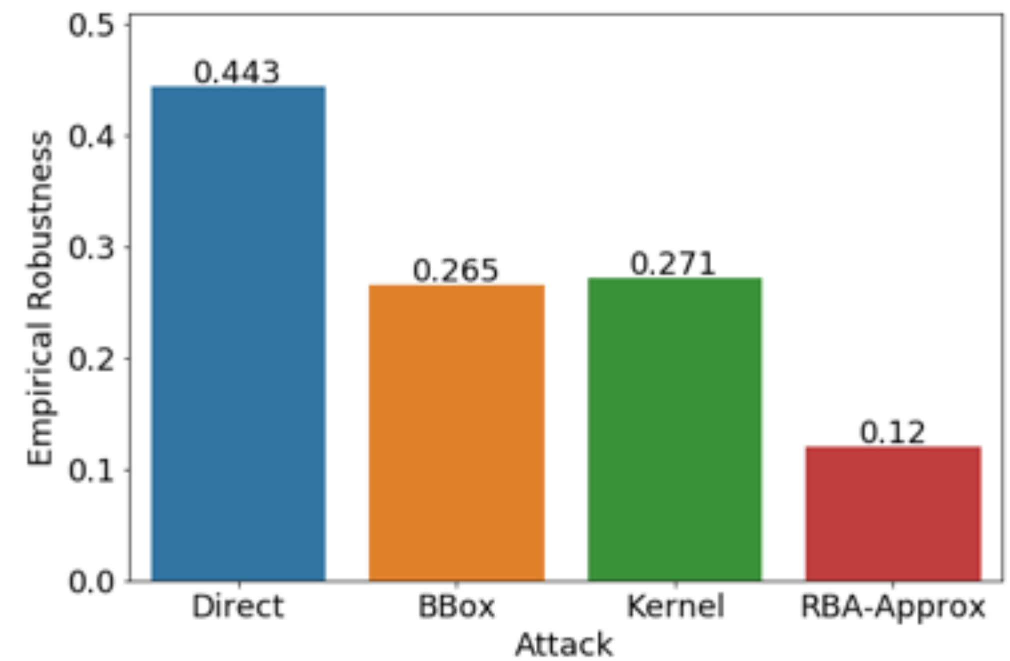
Approx Region-based attack (for 3NN, RF)

Results

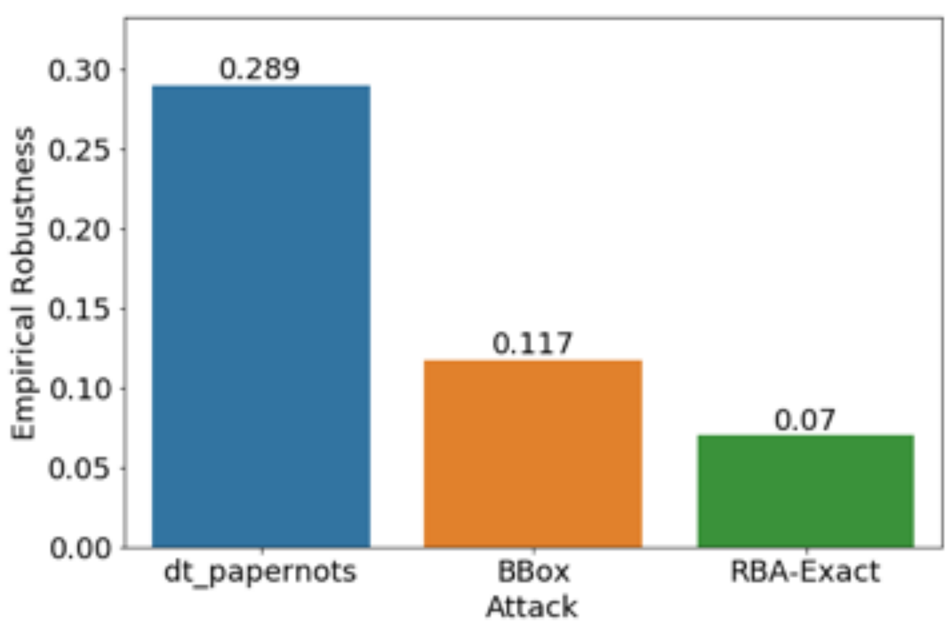
1-NN



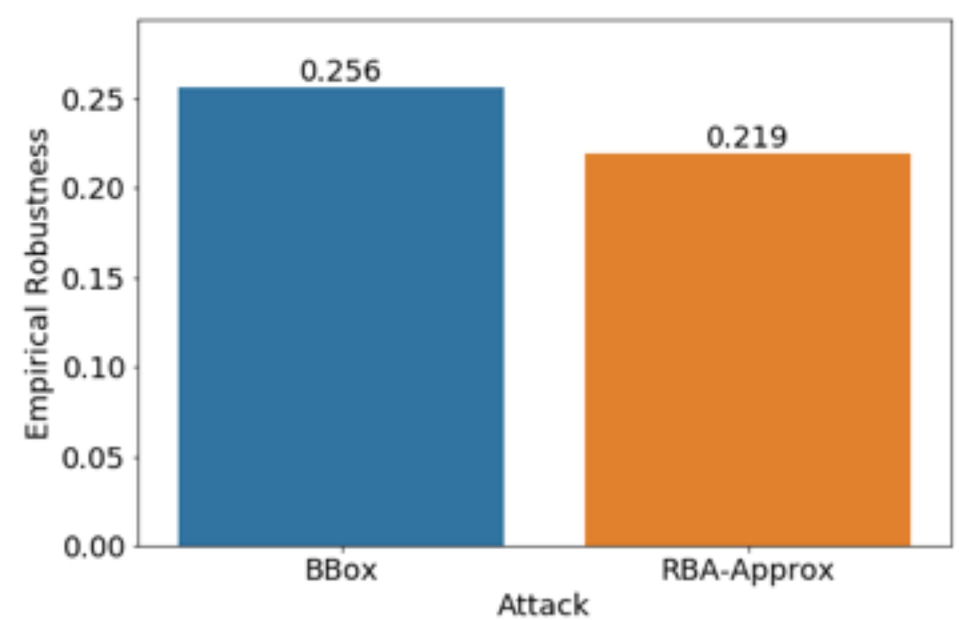
3-NN



DT



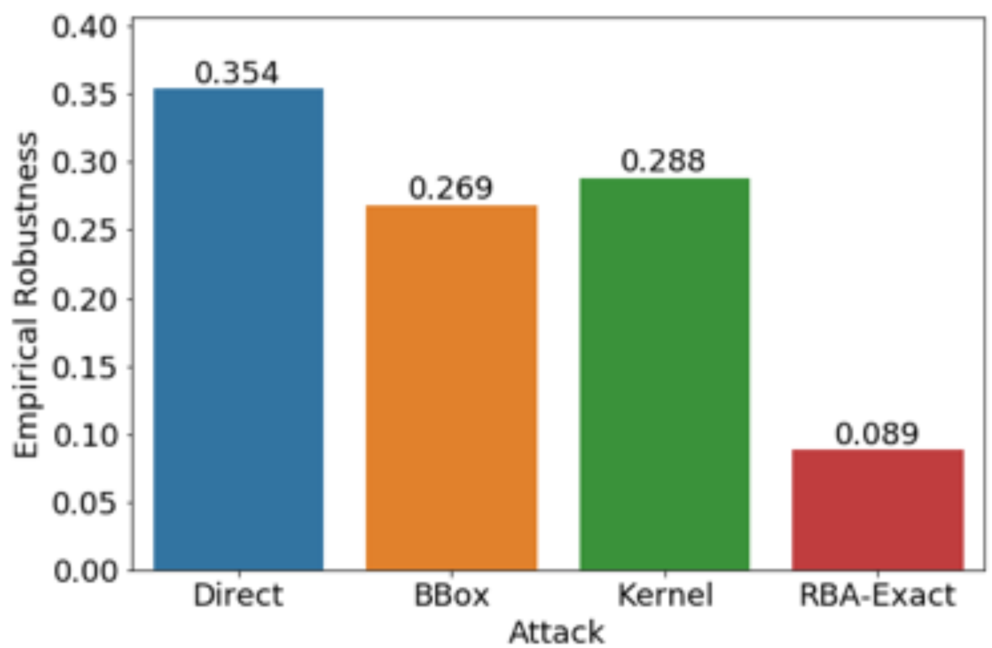
RF



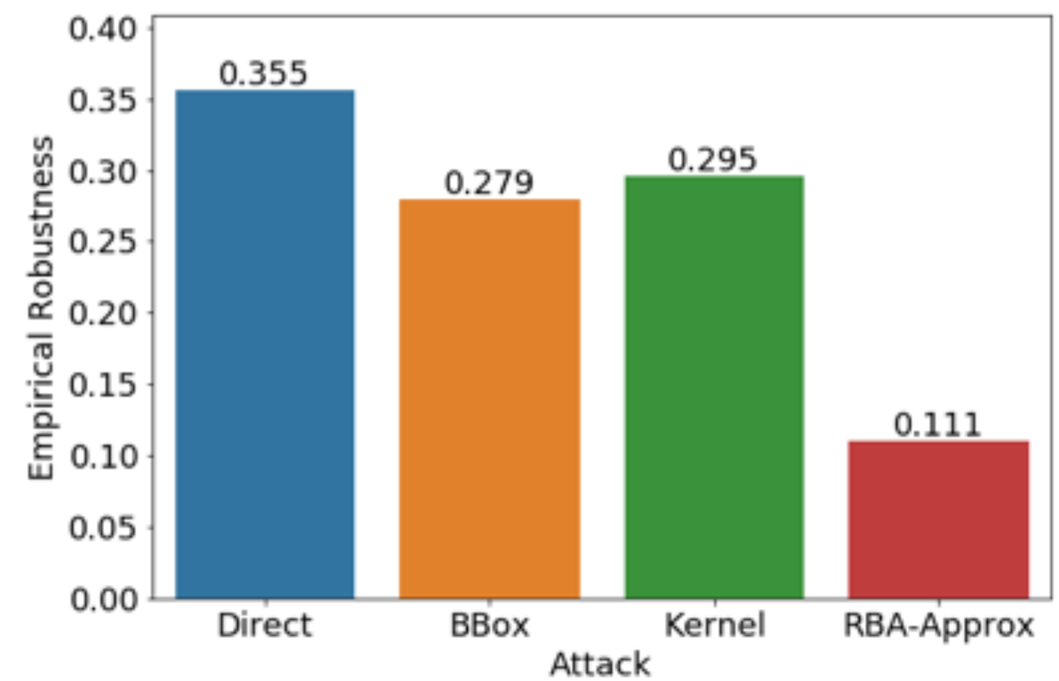
(Low bar means better)

Results

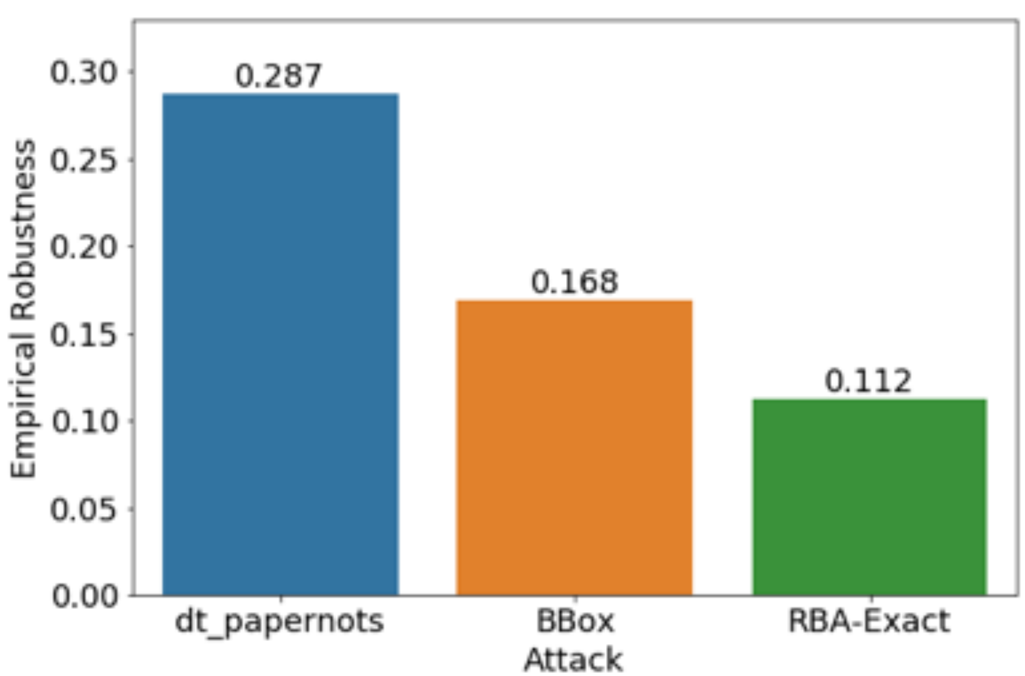
1-NN



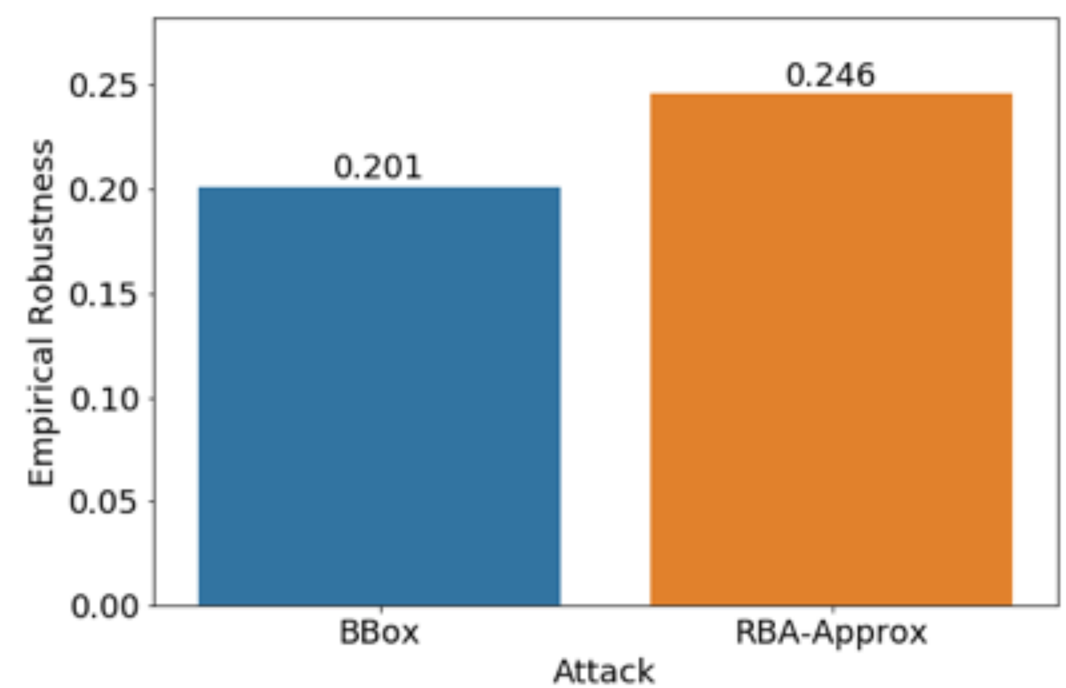
3-NN



DT



RF



(Low bar means better)

Defense Metric

Attack Metric: Average empirical robustness over examples where f is accurate

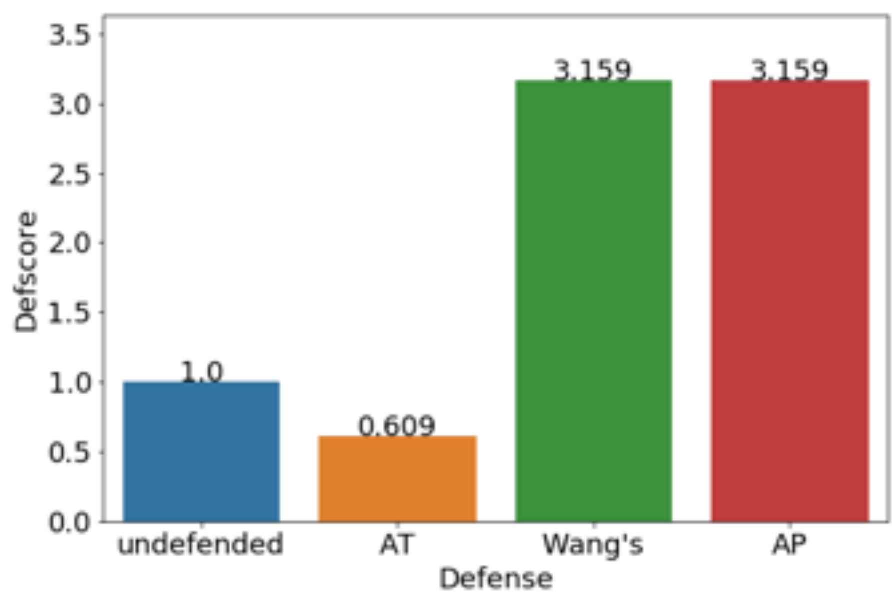
$$\text{Defense Score for defense D with attack A} = \frac{\text{Empirical Robustness } (A, f_D)}{\text{Empirical Robustness } (A, f_U)}$$

(f_D = classifier produced by D, f_U = undefended classifier)

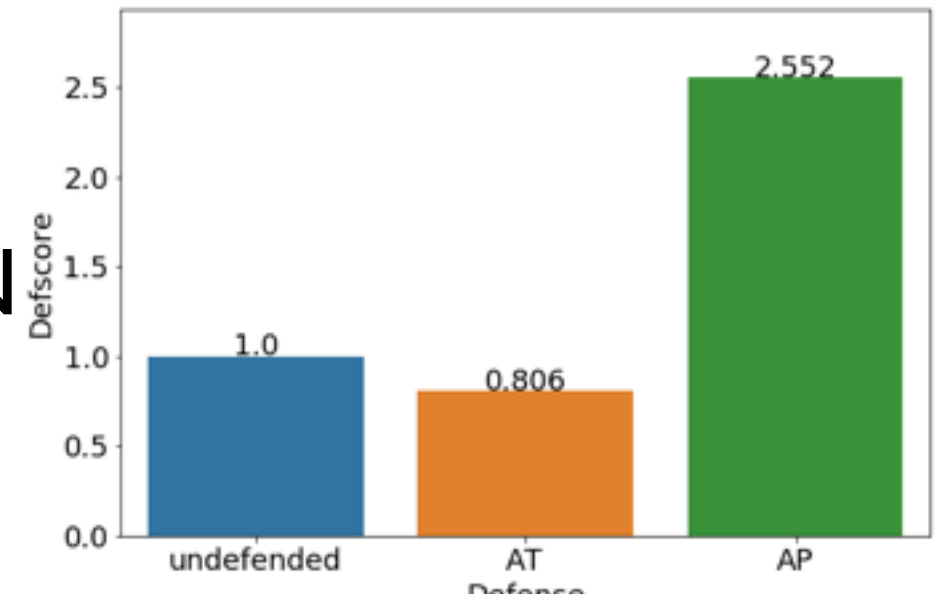
High defense score means good defense

Results

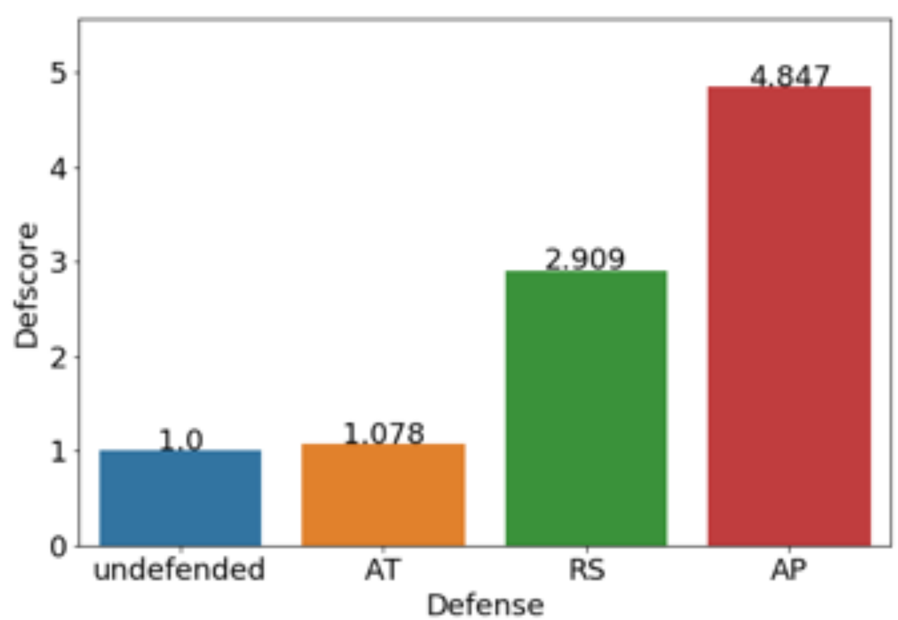
I-NN



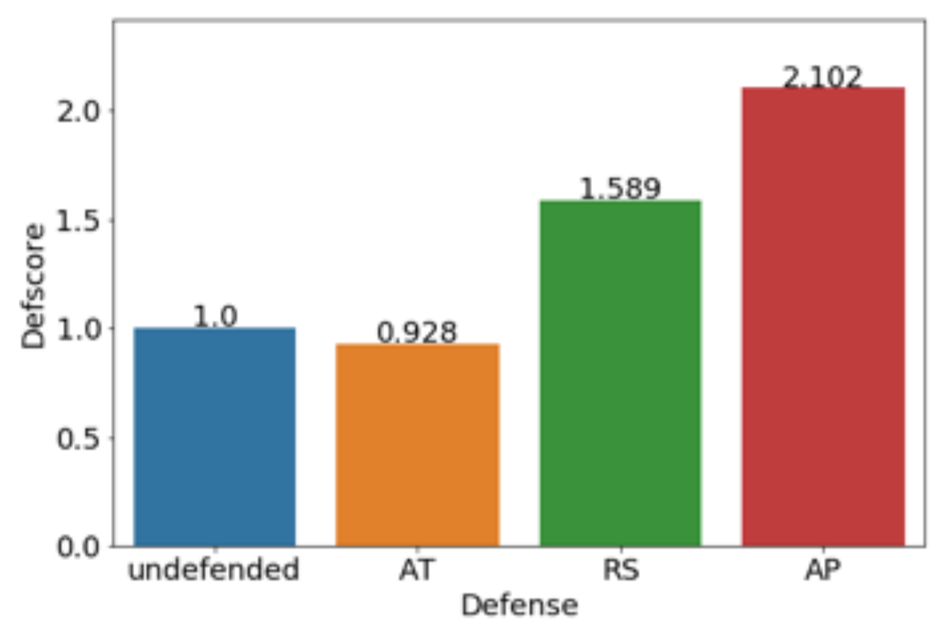
3-NN



DT



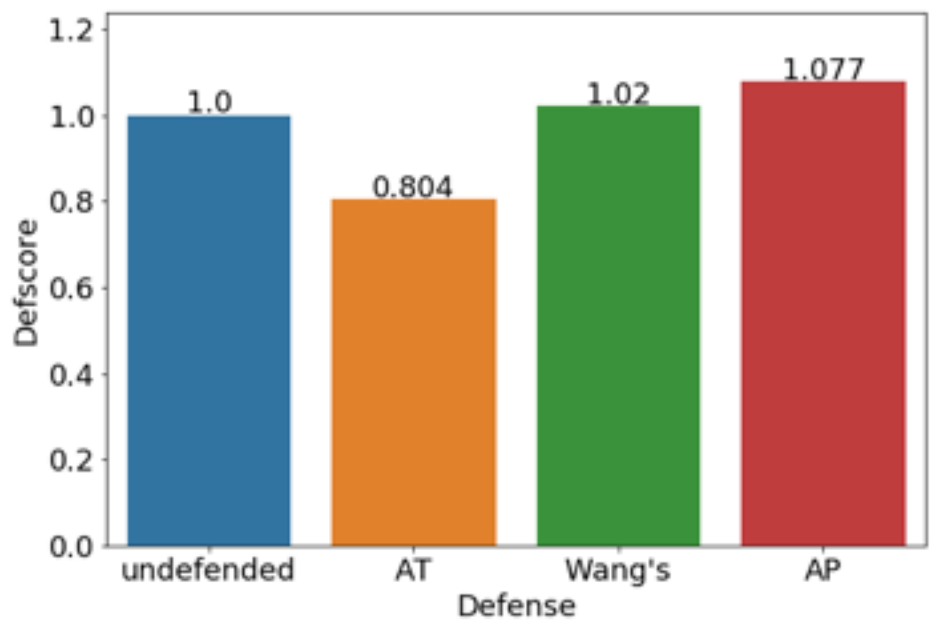
RF



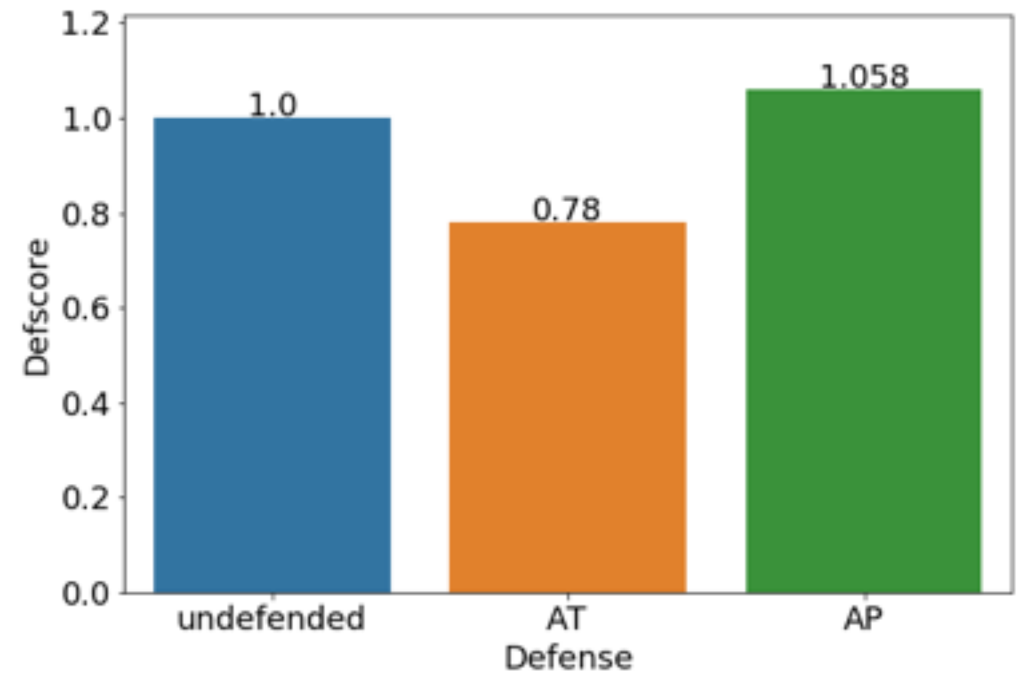
(High bar is better)

Results

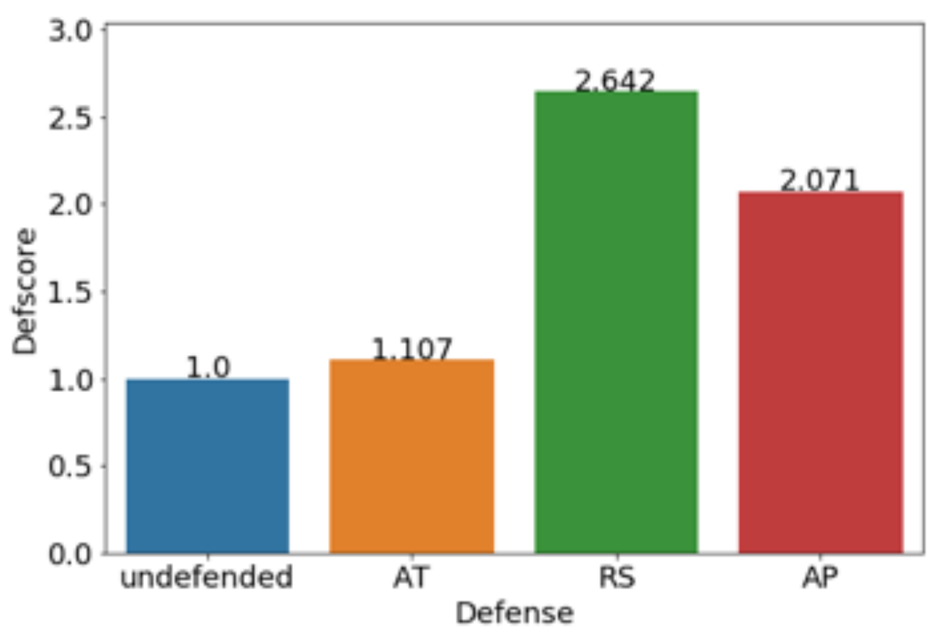
1-NN



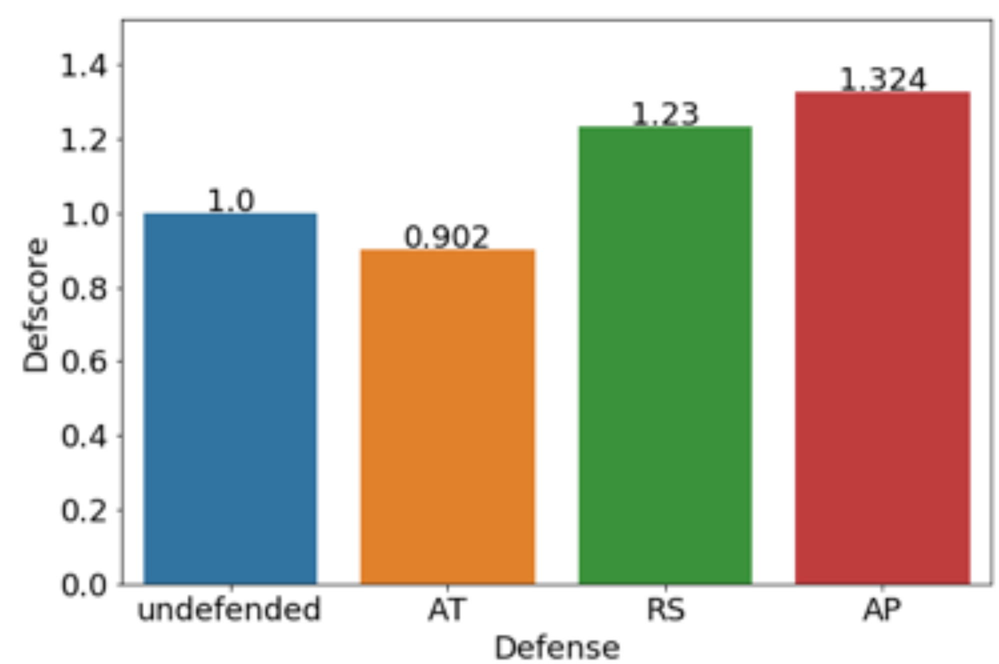
3-NN



DT



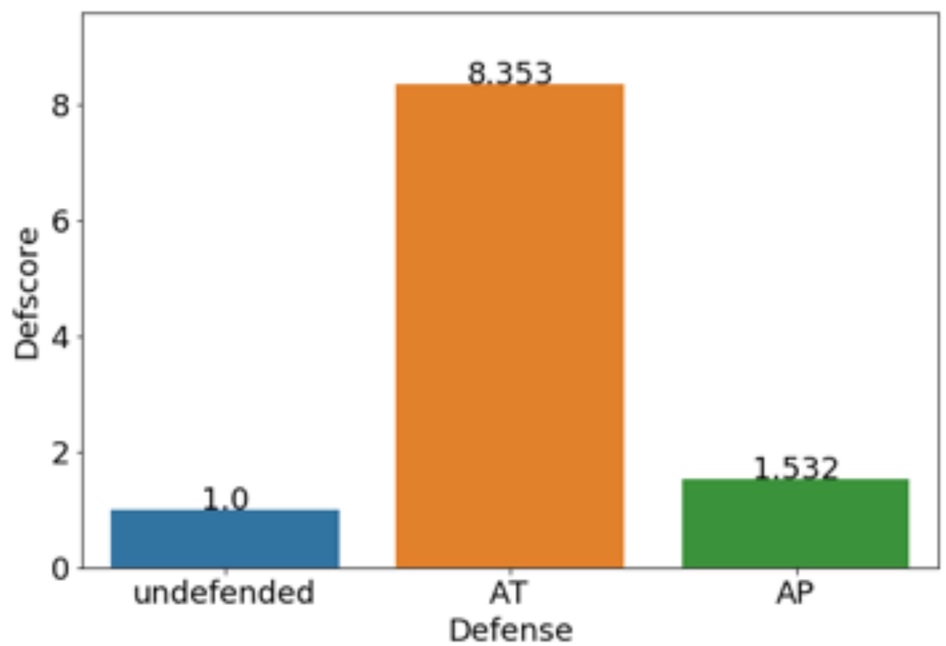
RF

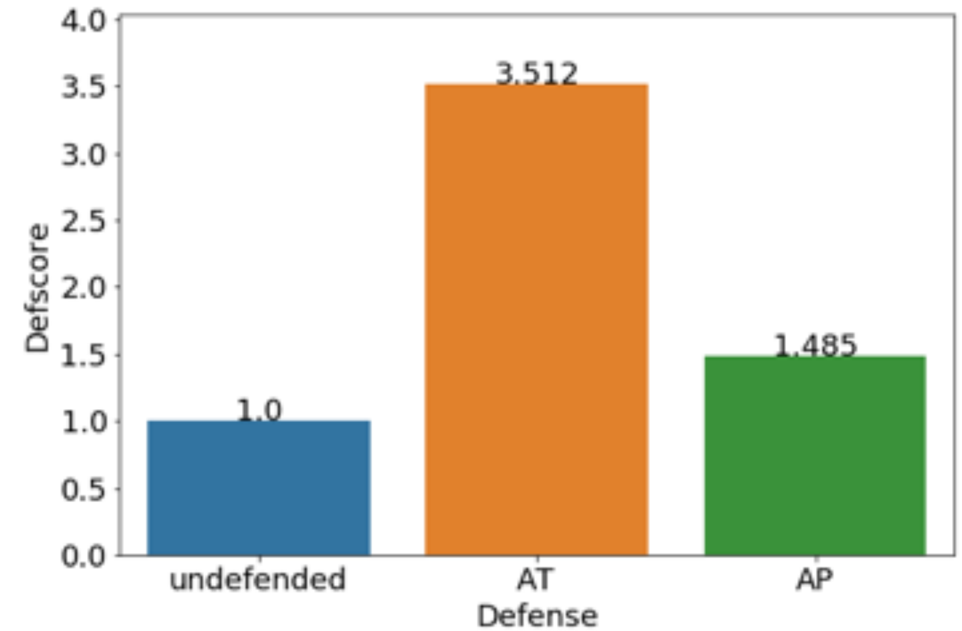


(High bar is better)

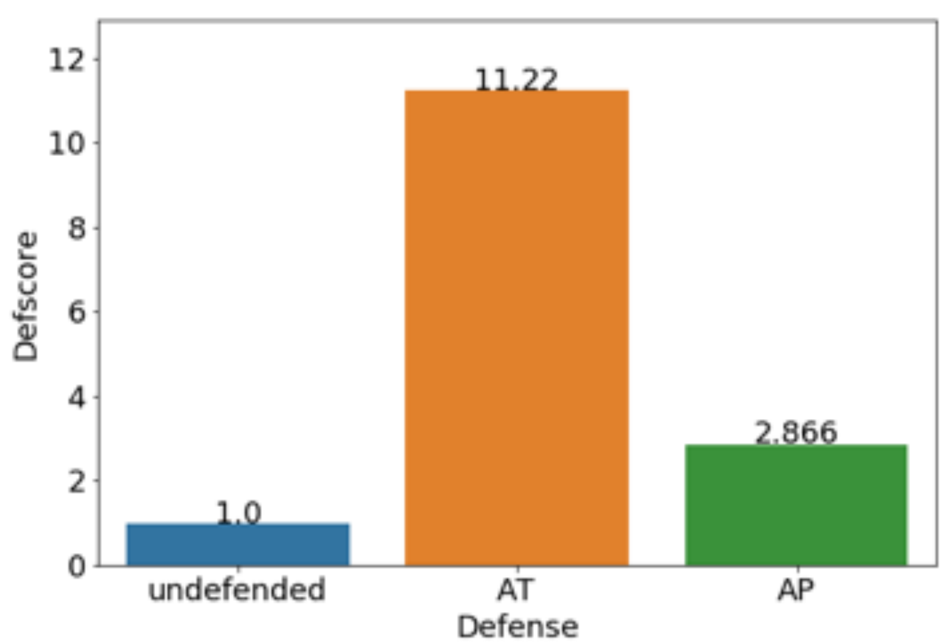
Parametrics - AT vs AP

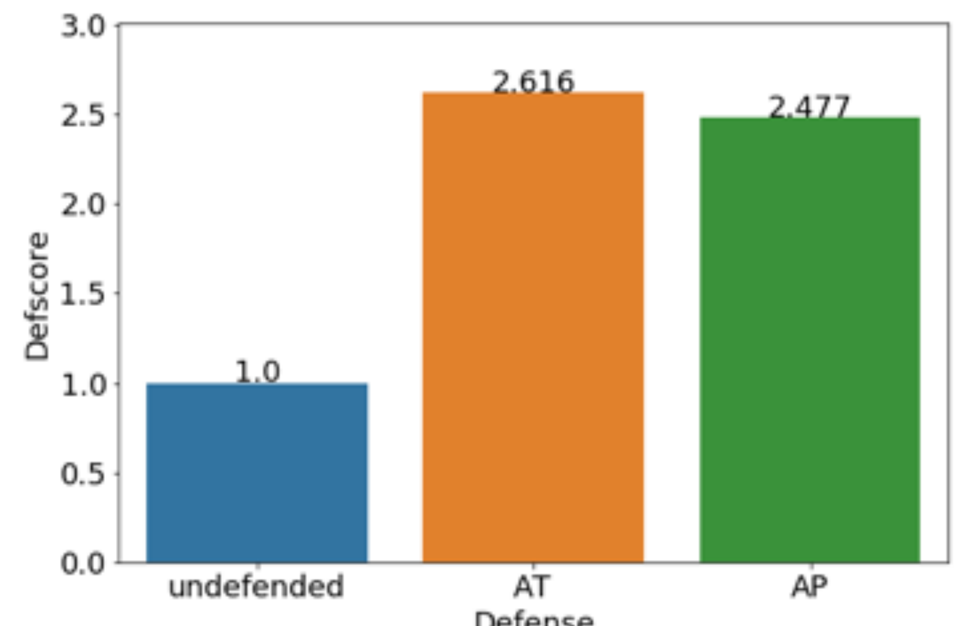
LR





MLP





(High bar is better)

Experiments

- Region-based Attacks are better than or competitive with prior attacks

- Adversarial Pruning is also better than or competitive with existing defenses

- Adversarial Pruning also helps parametric methods but not as much as adversarial training

Conclusion

- k_n Nearest neighbors is robust to adversarial examples for very large k_n
- Non-parametric methods are different from parametric methods when it comes to adversarial examples

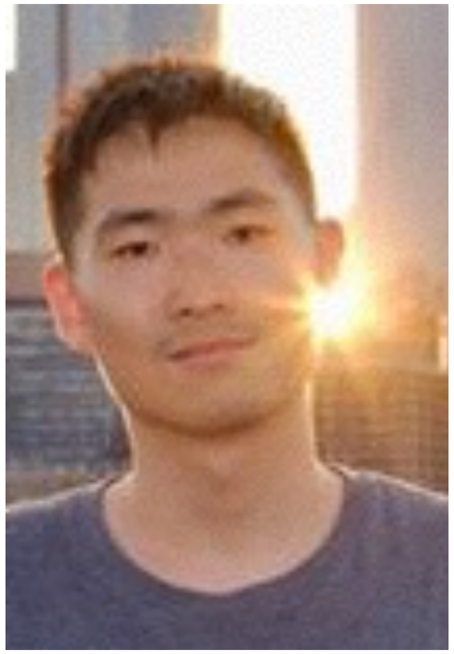
References

- *"Analyzing the Robustness of Nearest Neighbors to Adversarial Examples"*, Yizhen Wang, Somesh Jha and Kamalika Chaudhuri, ICML 2018

- *"Adversarial Examples for Non-parametrics: Attacks, Defenses and Large-sample Limits"*, Yaoyuan Yang, Cyrus Rashtchian, Yizhen Wang and Kamalika Chaudhuri, Arxiv 2019

Acknowledgements

Cyrus
Rashtchian



Yao-yuan
Yang

Yizhen
Wang

Somesh
Jha