Lecture 5: Adversarial Robustness

Trustworthy Machine Learning

Spring 2024

Deep Neural Networks
are “mostly” accurate, yet brittle

“car”

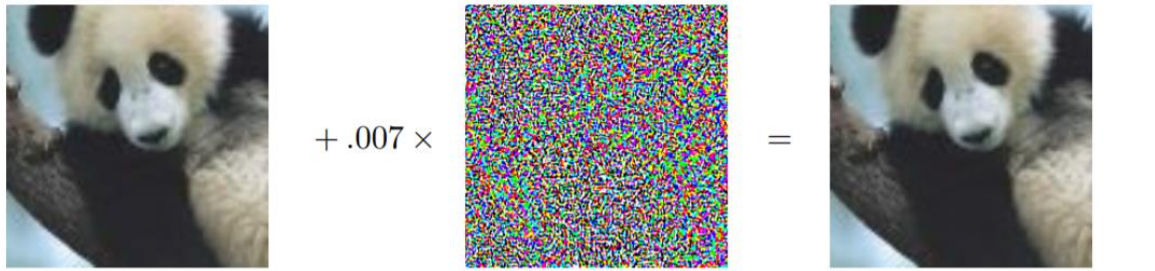
“truck”

Deep Neural Networks
are “mostly” accurate, yet brittle

Prediction:
Stop sign

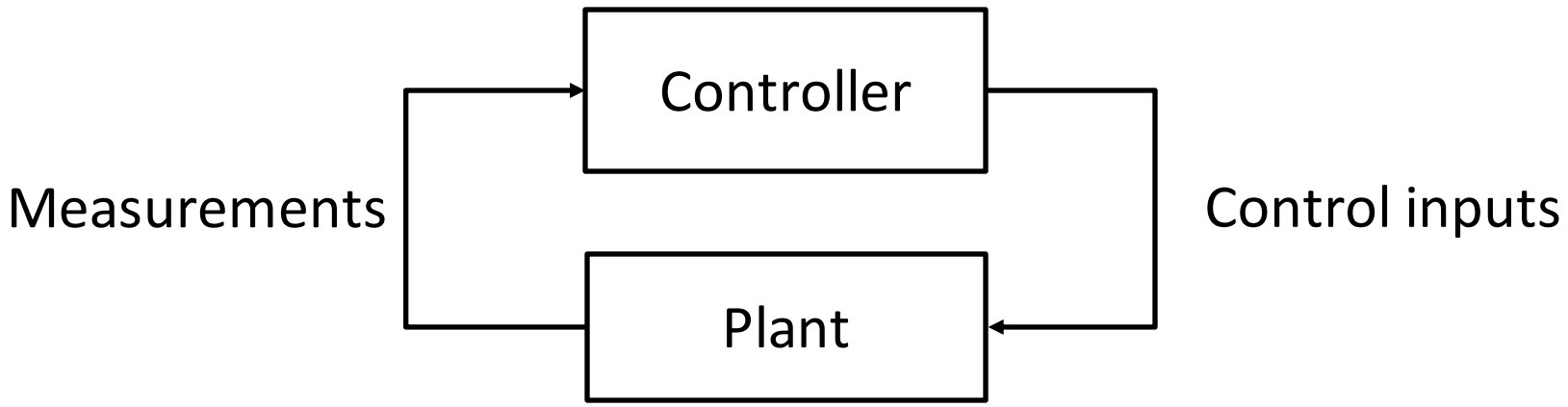
Prediction:
Speed Limit 45

Deep Neural Networks
are “mostly” accurate, yet brittle



robustness: similar images ⇒ same label

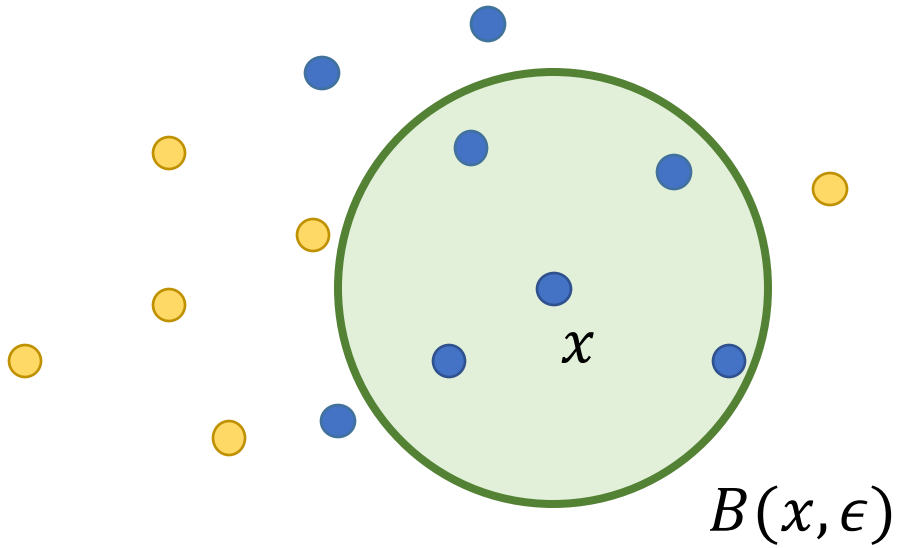
Historical Context: Robust Control



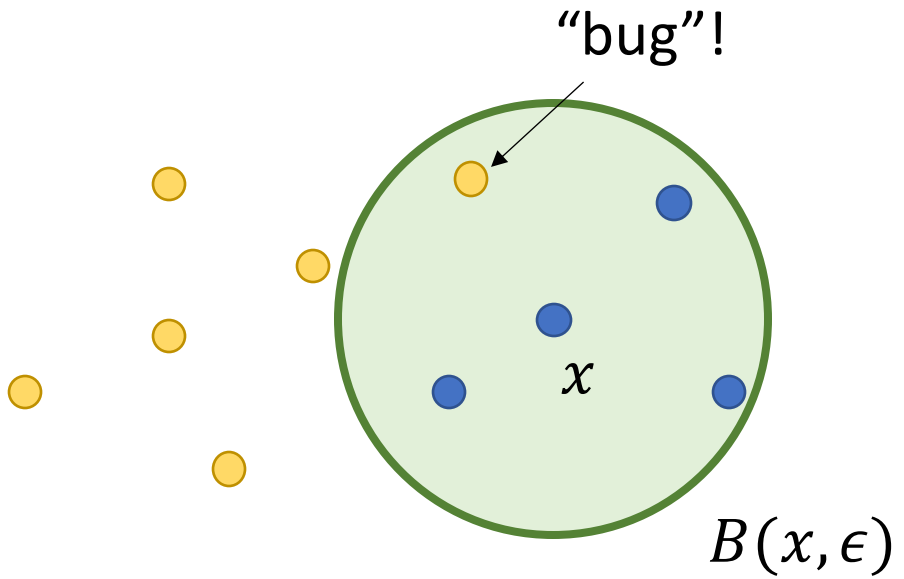
Desired robustness property of the controller:

When sensor measurements change slightly, control inputs should not change drastically

robustness: $\|x - x'\|_\infty \leq \epsilon \Rightarrow$ same label



ϵ-robust at x



not ϵ-robust at x

Research Directions

- **How to make neural networks robust?**
- **Can we “fool” neural networks to misclassify?**
- **Can we design learning algorithms to get robustness guarantees?**
- **Can we verify that a given model is robust?**
- **What about LLMs?**

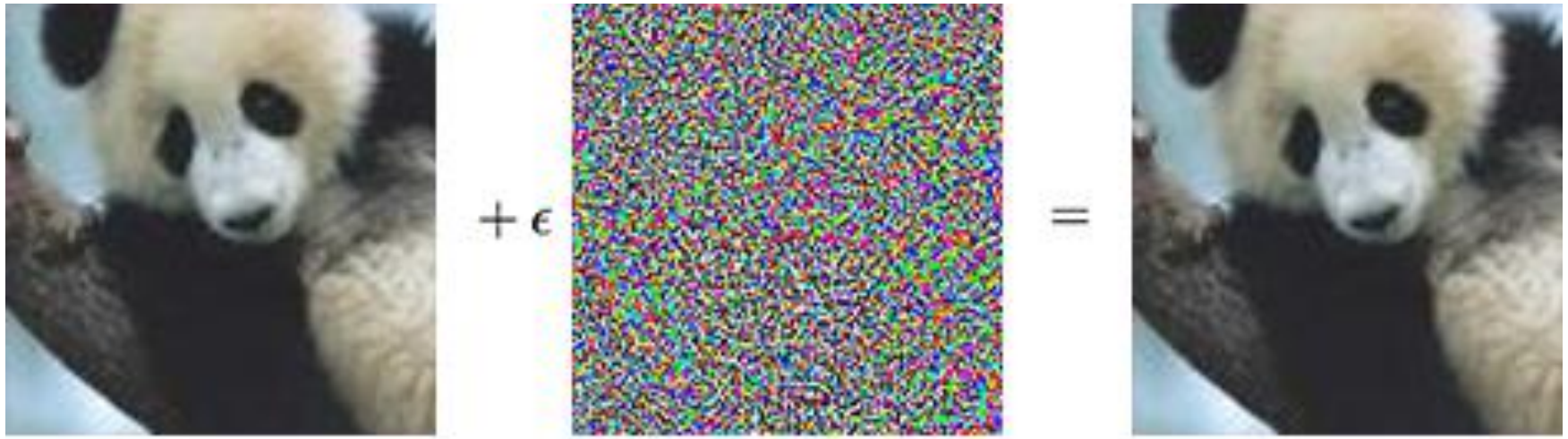
Agenda

- **Today: Attack: Adversarial Examples**
- **Feb 7: Defense: Adversarial training and randomized smoothing**
- **Feb 12: Guest lecture by Alex Robey on robustness for LLMs**
- **Feb 14, 19 (and maybe 21): Formal methods for verified robustness**
- **Homework 1 on adversarial robustness**

Today: Adversarial Examples

- **Key publications:**
  - **Intriguing properties of neural networks; Szegedy et al, 2014**
  - **Explaining and harnessing adversarial examples; Goodfellow et al, 2015**
- **Acknowledgement for slides:**
  - **Osbert's lecture in CIS 5190**
  - **Eric Wong's lectures in "Debugging Data and Models"**
  - **Tutorial: Adversarial robustness: Theory and practice; Kolter and Madry**

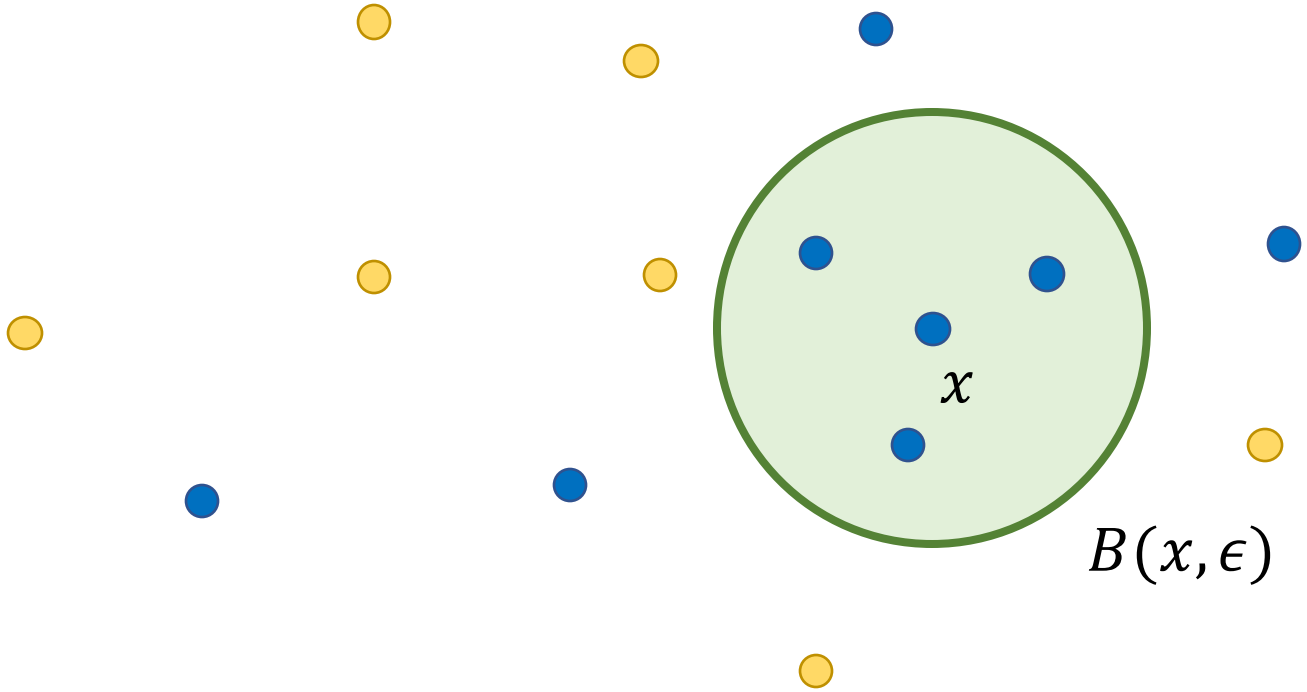
A Legendary Adversarial Example



Szegedy et al., Intriguing Properties of Neural Networks, 2014

Is there a simple fix using data augmentation ?

Doesn't work to ensure robustness!
In theory as well as practice!!

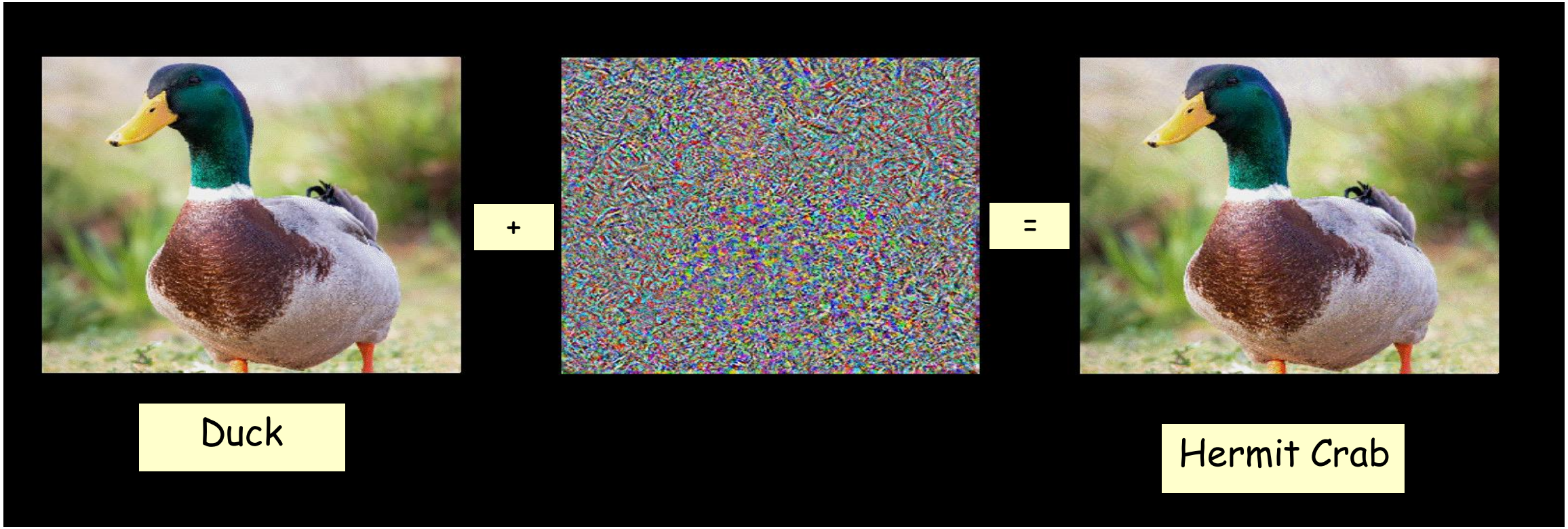


- Sample multiple points close to x
- Assign them same label as x
- Add them to training data set and retrain

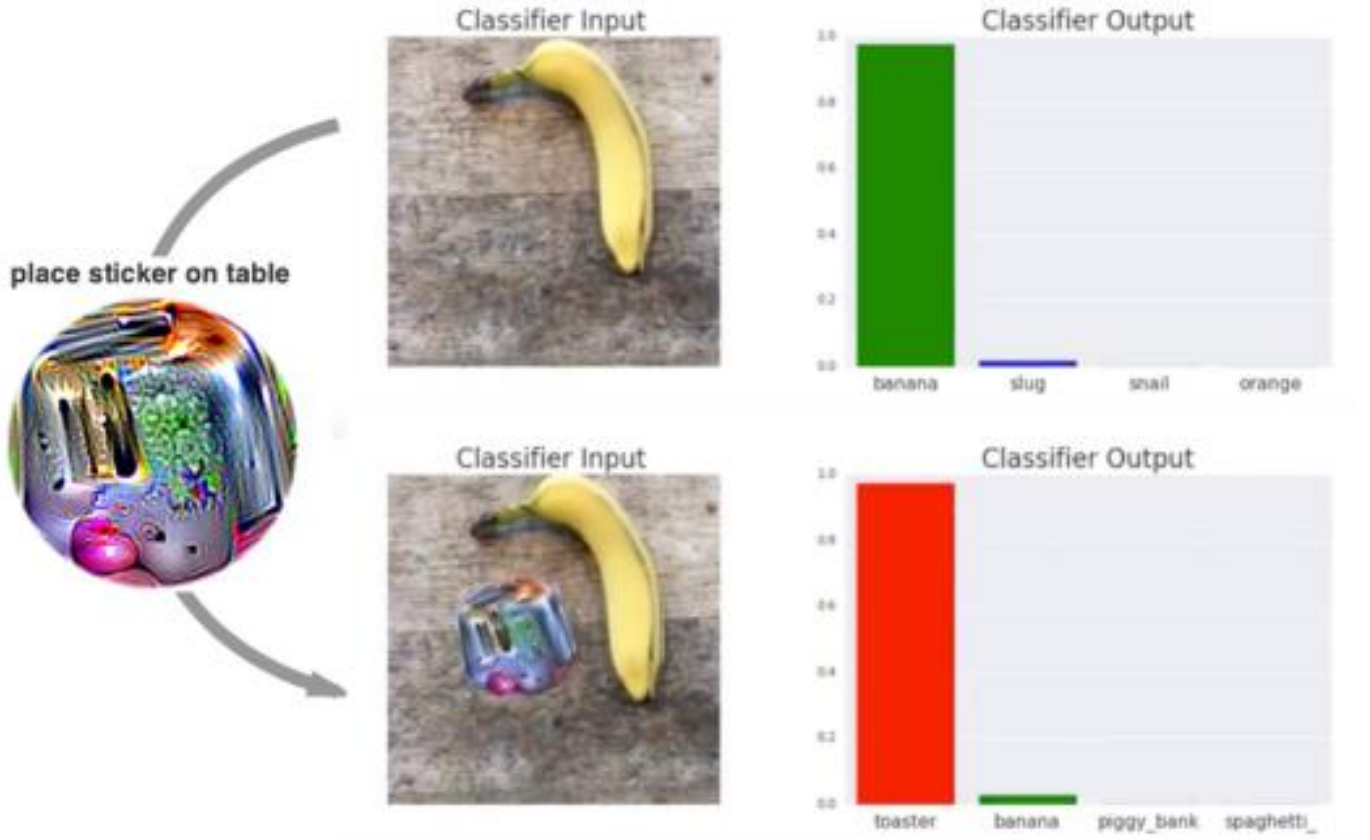
Szegedy et al (2014) discovery

A surprisingly robust strategy for finding adversarial examples

Adversarial examples everywhere …



Patch attack



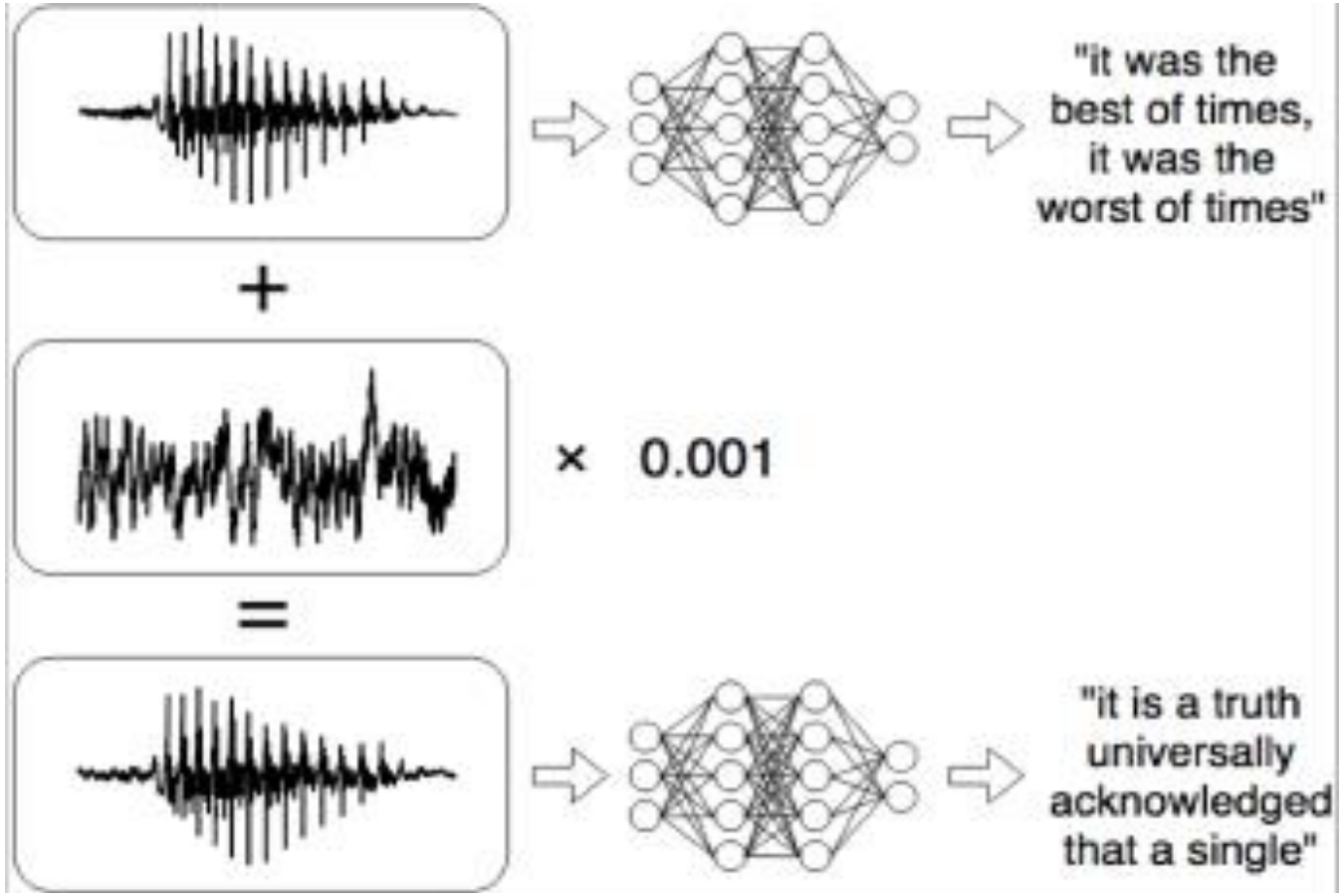
Brown et al, 2017 “Adversarial Patch”

Sentences and language models

Label	Sentence
P	I am currently trying to give this company another chance. I have had the same scheduling experience as others have written about. Wrote to them today
N	I am currently trying to give this company another review. I have had the same dental experience about others or written with a name. Thanks to them today

Hsieh et al. 2019 “Natural Adversarial Sentence Generation with Gradient based Perturbation”

Speech recognition



Carlini, Wagner, 2018

Adversarial Perturbations can be dangerous …

- **Task:**
  - Photo ID verification
  - Goal is to check whether uploaded photo matches a photo ID
- **Attack:**
  - User perturbs their image to match the photo in the ID
  - Challenge for machine learning in online identity verification!

(Valid photo ID from Papesh 2018)

Finding Adversarial Examples

Szegedy et al, 2014

Supervised Learning

- Given a model f parameterized by θ
- Loss(x, y; θ) denotes the error of f_θ on input x with respect to desired output y

- Learning as optimization:

  Given a training set of labeled input/output pairs (x, y),

  find θ to minimize the average training loss

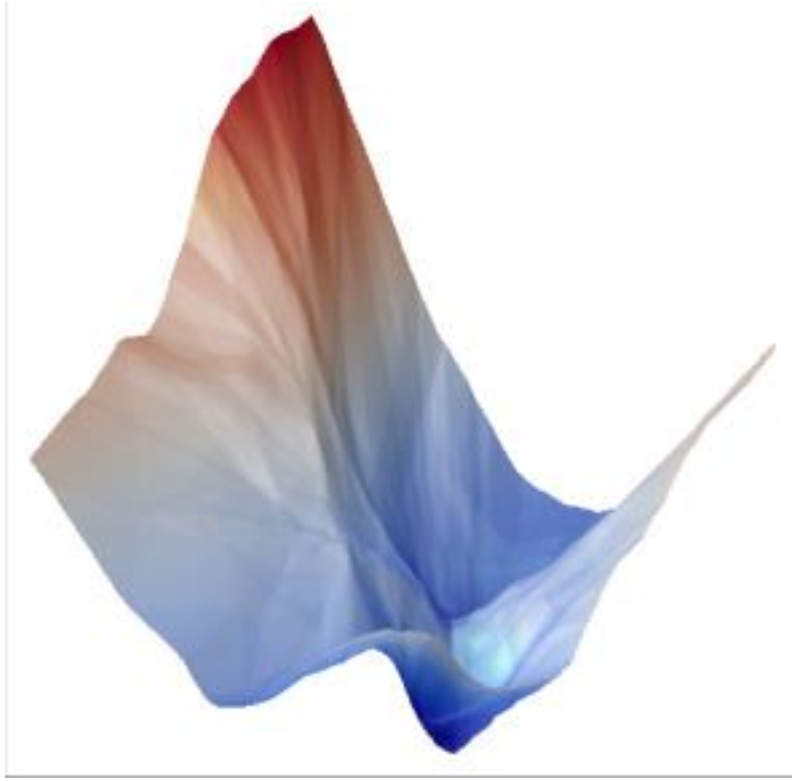
Adversarial Example Computation

- Given a (trained) model f with parameters θ
- Fix input x and corresponding output $y = f_\theta(x)$
- Loss($x+\delta$, y; θ) denotes the "change" in output with respect to δ–perturbation in input
- How can we formalize searching for adversarial example as optimization?

Given a bound Δ on input perturbation,

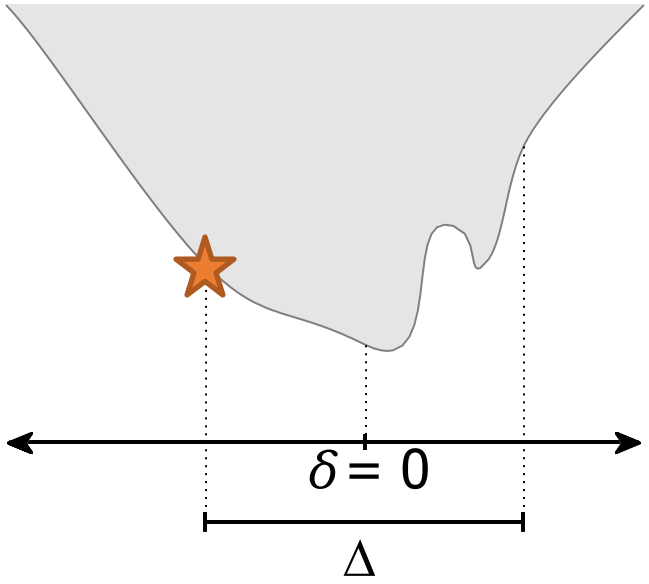
find $0 < \delta < \Delta$ to maximize Loss($x+\delta$, y; θ)

Challenge: Complexity of model / loss function



$\max_{\delta < \Delta} \text{Loss}(x+\delta, y; \theta)$

Solution: Local Search using Gradient Descent



$\max_{\delta < \Delta} \text{Loss}(x+\delta, y; \theta)$

How to implement desired gradient descent ?

- Key step in learning:

  Computing the gradient ∇_θ Loss(x, y; θ) using backpropagation

- Question: How will you compute ∇_δ Loss(x+δ, y; θ) ?

- Question: To find the (locally) optimal value, is it ok to repeatedly update x to

  x + α ∇_δ Loss(x+δ, y; θ), where α is the learning rate ?

- No! We want $\delta < \Delta$, so this is an instance of "constrained optimization"

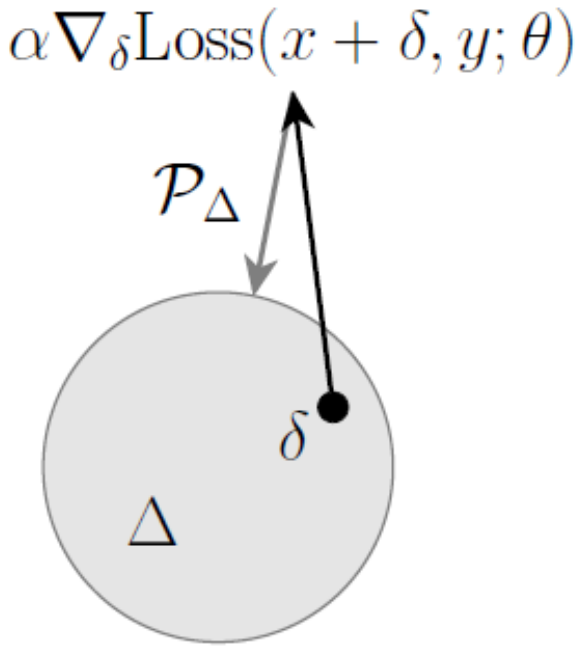
Projected gradient descent

Recall we are optimizing

$$\max_{\delta \in \Delta} \text{Loss}\,(x + \delta, y; \theta)$$

We can employ a projected gradient descent method, take gradient step and project back into feasible set Δ

$$\delta := \mathcal{P}_{\Delta}[\delta + \nabla_{\delta}\text{Loss}(x + \delta, y; \theta)]$$



Source: Tutorial on Adversarial robustness by Kolter and Madry

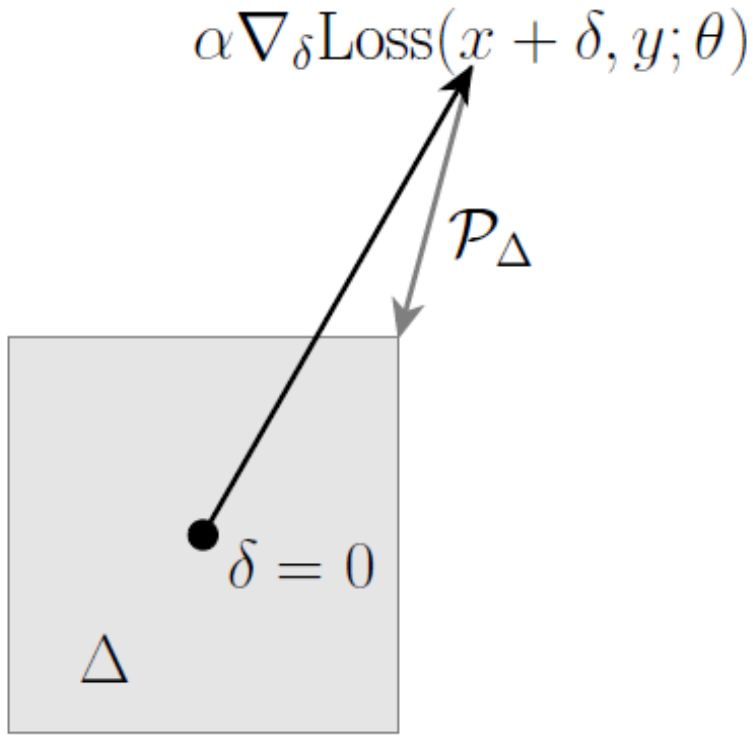
Fast Gradient Sign Method (FGSM)

To be more concrete, take Δ to be the ℓ_∞ ball, $\Delta = \{\delta : \|\delta\|_\infty \leq \epsilon\}$, so projection takes the form

$$P_\Delta(\delta) = \text{Clip}(\delta, [-\epsilon, \epsilon])$$

As $\alpha \to \infty$, we always reach "corner" of the box, called fast gradient sign method (FGSM) [Goodfellow et al., 2014]

$$\delta = \epsilon \cdot \text{sign}\big(\nabla_\delta \text{Loss}(x + \delta, y; \theta)\big)$$

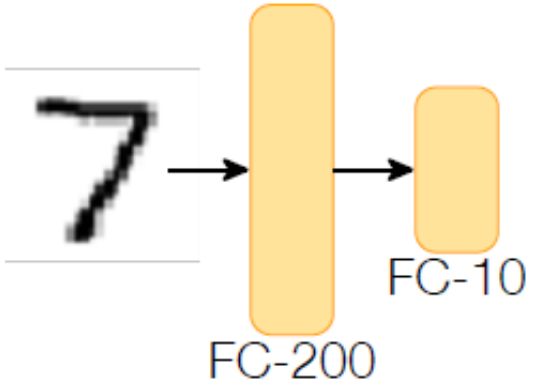


Source: Tutorial on Adversarial robustness by Kolter and Madry

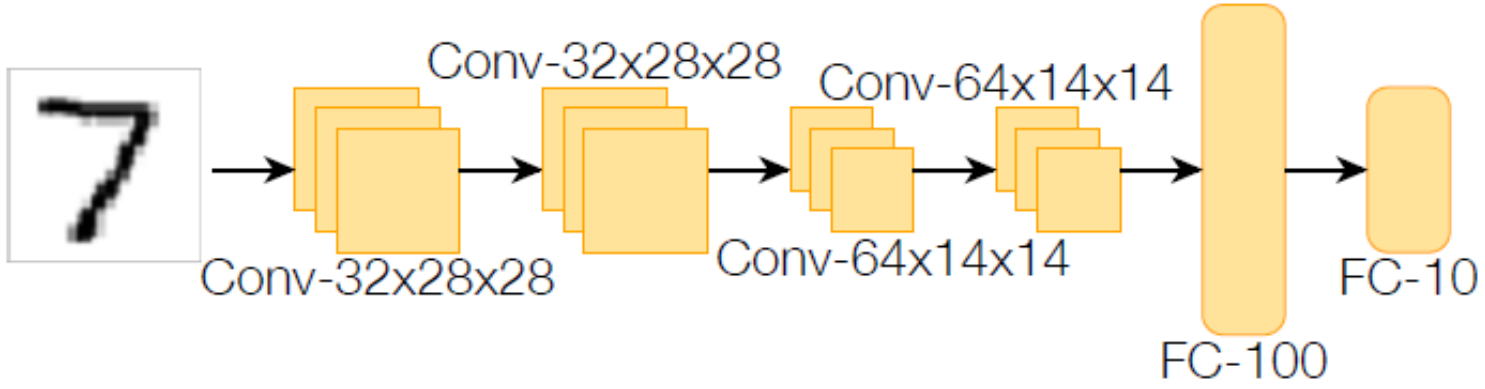
Empirical Evaluation

Will apologies to everyone, you are going to see MNIST examples in the tutorial … it is the best dataset for demonstrating some of the more computationally intensive methods

2-layer fully connected MLP

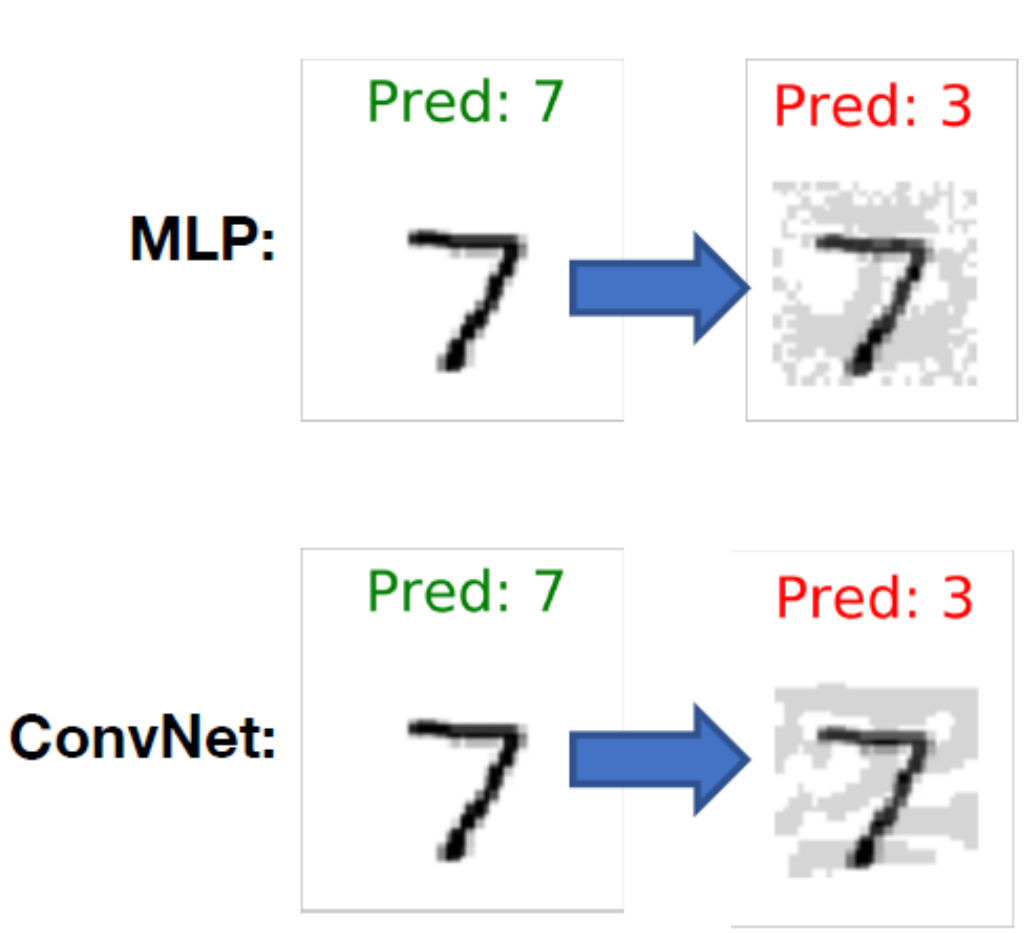


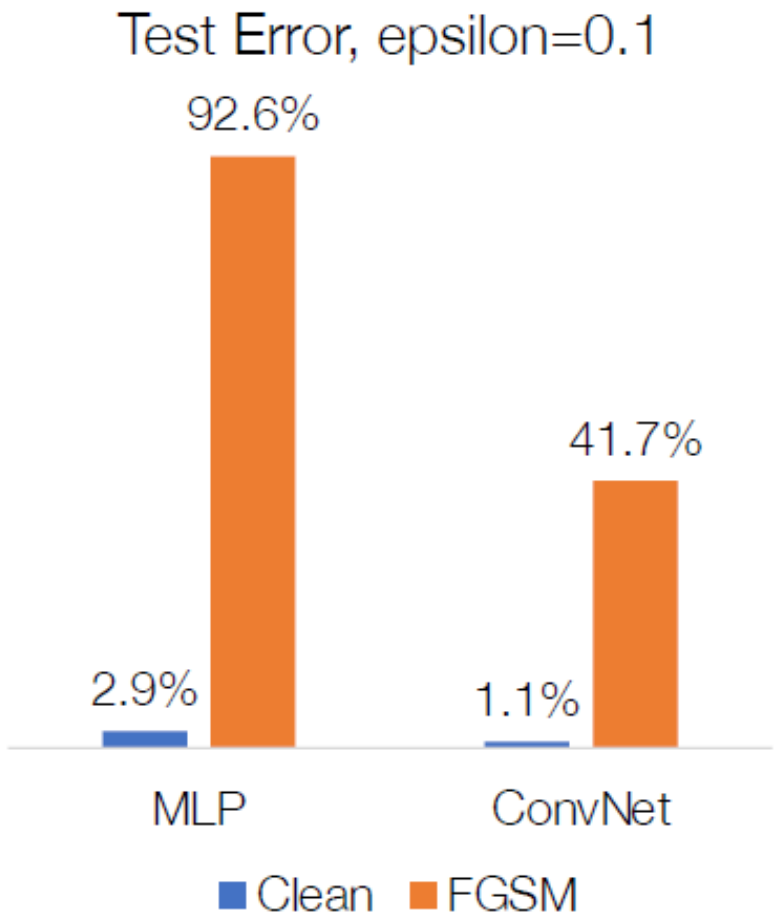
6 layer ConvNet



Source: Tutorial on Adversarial robustness by Kolter and Madry

Evaluation of FGSM





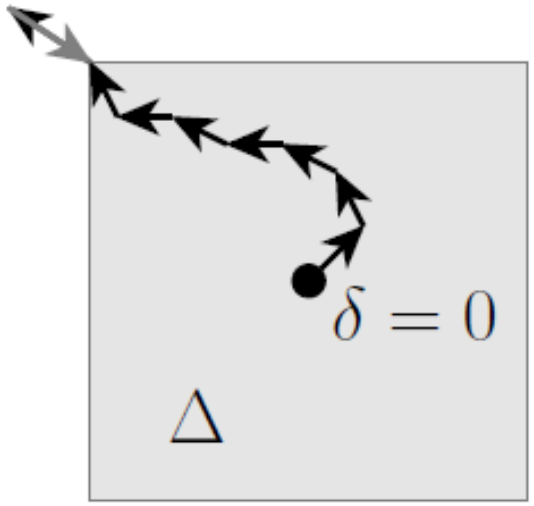
Source: Tutorial on Adversarial robustness by Kolter and Madry

Projected Gradient Descent

Projected gradient descent applied to ℓ_∞ ball, repeat:

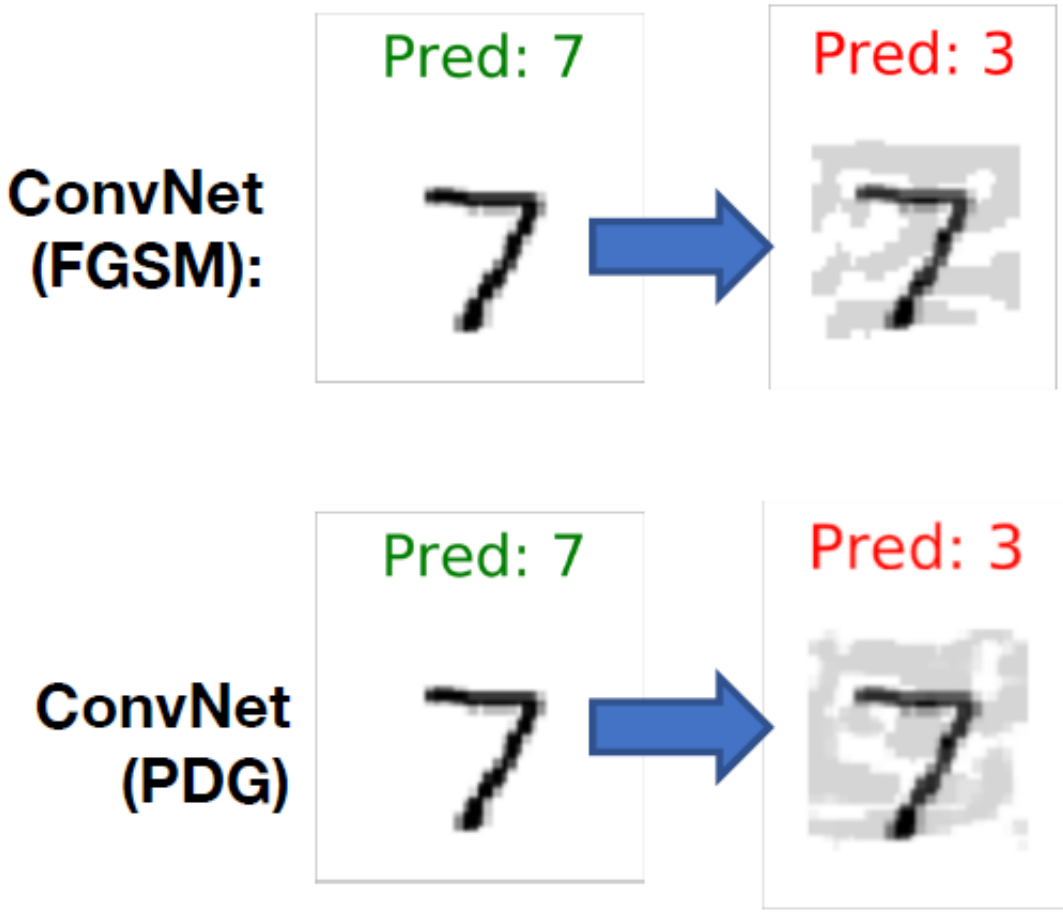
$$\delta := \text{Clip}_\epsilon[\delta + \alpha \nabla_\delta J(\delta)]$$

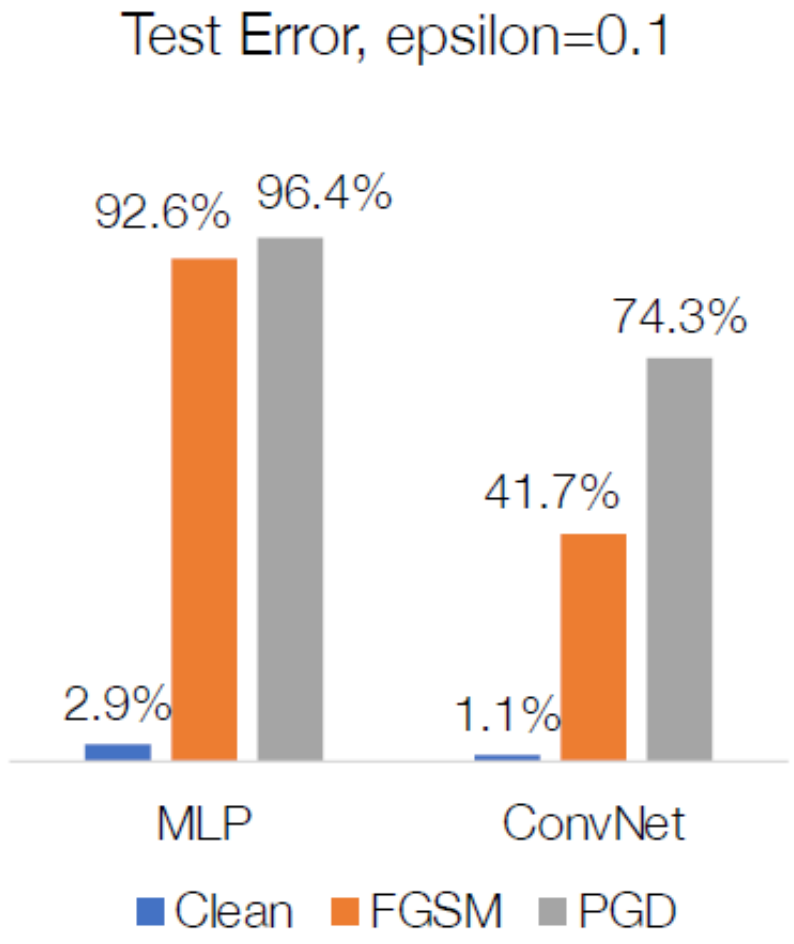
Slower than FGSM (requires multiple iterations), but typically able to find better optima



Source: Tutorial on Adversarial robustness by Kolter and Madry

PGD Evaluation





Source: Tutorial on Adversarial robustness by Kolter and Madry

Targeted Attack

Also possible to explicitly try to change label to a *particular* class

$$\max_{\delta \in \Delta} \left(\text{Loss}(x + \delta, y; \theta) - \text{Loss}(x + \delta, y_{\text{targ}}; \theta) \right)$$

Consider multi-class cross entropy loss

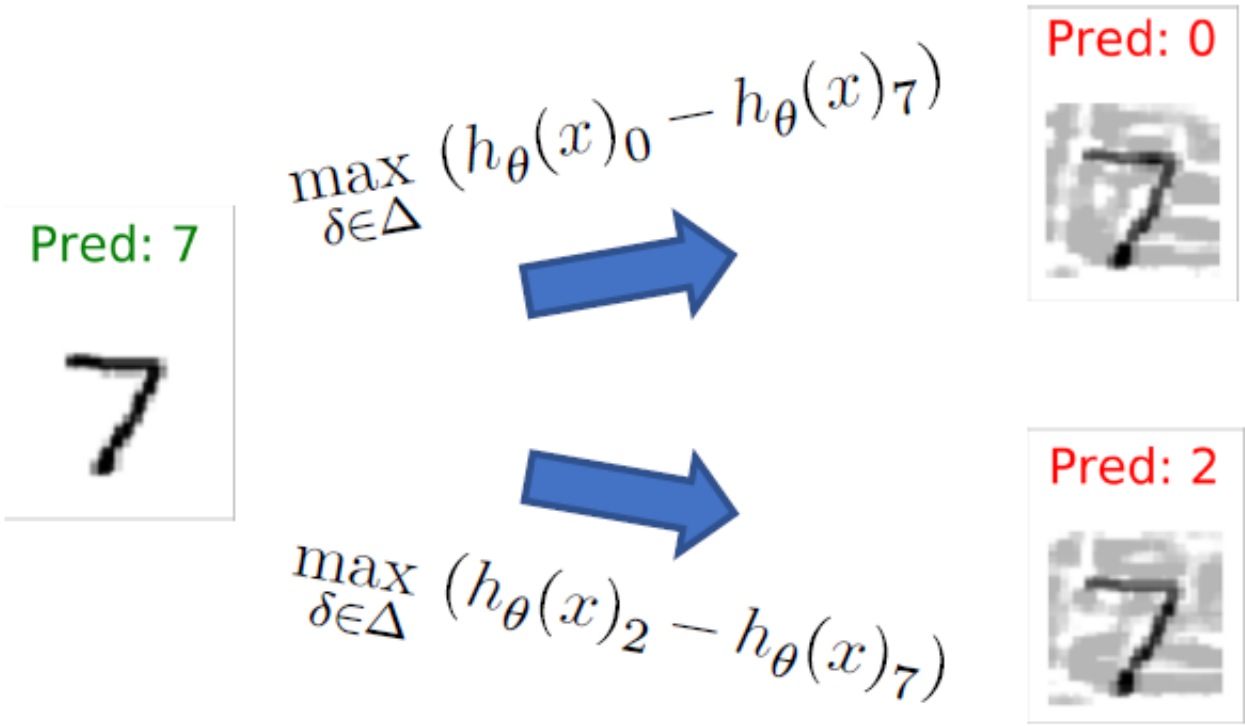
$$\text{Loss}(x + \delta, y; \theta) = \log \sum_i \exp h_\theta(x + \delta)_i - h_\theta(x)_y$$

Then note that above problem simplifies to

$$\max_{\delta \in \Delta} \left(h_\theta(x)_{y_{\text{targ}}} - h_\theta(x)_y \right)$$

Source: Tutorial on Adversarial robustness by Kolter and Madry

Targeted Attack Example



Note: A targeted attack can succeed in "fooling" the classifier, but change to a different label than target

Source: Tutorial on Adversarial robustness by Kolter and Madry

Alternative ways to solve the optimization problem

- Goal: Solve max $_{\delta < \Delta}$ Loss(x+δ, y; θ)
- Another approach: encode the problem using constraints and use specialized and optimized constraint solver such as ReluPLEX
- Another approach: Use convex relaxation for approximate solving
- We will revisit when we discuss verification/certification for robustness

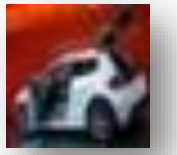
Beyond adding noise to images

- Adversarial patches: Given a “patch” p find an optimal position within given image x so as to maximize the loss on the augmented image

- Text substitution: Given a set of allowed substitutions (e.g. words by their synonyms) find the modified sentence to maximize the loss

Geometric Transformations

x



Rotate x between α° and β°

$$x_0 = f_0(\alpha, \beta)$$
$$x_1 = f_1(\alpha, \beta)$$
.....

$perturbation\ set\ \alpha = -10^{\circ}$ and $\beta = 10^{\circ}$

Intriguing Properties of Neural Networks

- Adversarial examples can be computed efficiently using Fast Gradient Sign Method
- On image classification benchmarks, adversarial examples are so close to original examples that the difference is imperceptible to human eye
- Same adversarial example is often misclassified by alternative classifiers with different architectures or trained using different data set !

Agenda

- **Today: Adversarial Examples**
- **Next class: Defense: Adversarial training and randomized smoothing**
- **Feb 12: Guest lecture by Alex Robey on robustness for LLMs**
- **Feb 14, 19 (and maybe 21): Formal methods for verified robustness**
- **Homework 1 on adversarial robustness**