Certified Robustness to Adversarial Examples with Differential Privacy

Mathias Lécuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, Suman Jana

Columbia University

Code: https://github.com/columbia/pixeldp
Contact: mathias@cs.columbia.edu
Deep Learning

• Deep Neural Networks (DNNs) deliver remarkable performance on many tasks.

• DNNs are increasingly deployed, including in attack-prone contexts:

The New York Times

Taylor Swift Said to Use Facial Recognition to Identify Stalkers

By Sopan Deb, Natasha Singer - Dec. 13, 2018
Example

![Diagram of a neural network with layers labeled as input, layer 1, layer 2, layer 3, and softmax. The softmax output shows probabilities for ticket 1, ticket 2, ticket 3, and no ticket. The bar chart on the right shows the distribution of $Q(x)$ with tickets 1, 2, 3, and no ticket.]
Example

But DNNs are vulnerable to adversarial example attacks.
Example

But DNNs are vulnerable to adversarial example attacks.
Accuracy (top 1)

Size of attack $\alpha$

Accuracy under attack Inception-v3 DNN on ImageNet dataset.

- Teddy bear: $\|\alpha\|_2 = 0.52$
- Giant panda: $\|\alpha\|_2 = 1.06$
- Teapot
1. Evaluate accuracy under attack:
   • Launch an attack on examples in a test set.
   • Compute accuracy on the attacked examples.
2. Improve accuracy under attack:
   • Many approaches: e.g. train on adversarial examples.
     (e.g Goodfellow+ '15; Papernot+ '16; Buckman+ '18; Guo+ '18)

Problem: both steps are attack specific, leading to an arms race that attackers are winning.
     (e.g Carlini-Wagner '17; Athalye+ '18)
Key questions

• **Guaranteed accuracy**; what is my minimum accuracy under any attack?
• **Prediction robustness**; given a prediction can any attack change it?
Key questions

• ** Guaranteed accuracy**: what is my minimum accuracy under any attack?

• **Prediction robustness**: given a prediction can any attack change it?

• A few recent approaches with provable guarantees.  
  (e.g. Wong-Kolter '18; Raghunathan+ '18; Wang+ '18)

• **Poor scalability** in terms of:
  • Input dimension (e.g. number of pixels).
  • DNN size.
  • Size of training data.
Key questions

• **Guaranteed accuracy**: what is my minimum accuracy under any attack?
• **Prediction robustness**: given a prediction can any attack change it?

• My defense **PixelDP** gives answers for norm bounded attacks.
• Key idea: novel use of **differential privacy** theory at prediction time.
• The **most scalable** approach: first provable guarantees for large models on ImageNet!
PixelDP outline

Motivation

Design

Evaluation
Key idea

• Problem: small input perturbations create large score changes.
Key idea

- Problem: small input perturbations create large score changes.
- Idea: design a DNN with bounded maximum score changes (leveraging Differential Privacy theory).

\[ \sqrt{\sum \alpha_i^2} = \| \alpha \|_2 \leq L \]

- Diagram showing a DNN with layers and softmax function, with input perturbations and classification results.
- Bar chart showing classification results for different tickets.
Differential Privacy

• Differential Privacy (DP): technique to randomize a computation over a database, such that changing one data point can only lead to bounded changes in the distribution over possible outputs.

• For \((\varepsilon, \delta)\)-DP randomized computation \(A_f\):

\[
P(A_f(d) \in S) \leq e^\varepsilon P(A_f(d') \in S) + \delta
\]

• We prove the Expected Output Stability Bound. For any DP mechanism with bounded outputs in \([0, 1]\) we have:

\[
\mathbb{E}(A_f(d)) \leq e^\varepsilon \mathbb{E}(A_f(d')) + \delta
\]
Key idea

- Problem: small input perturbations create large score changes.
- Idea: design a DNN with bounded maximum score changes (leveraging Differential Privacy theory).

![Diagram of a neural network with layers and softmax output showing prediction and no ticket with values 0.1, 0.6, 0.1, 0.2, and 0.1]
Key idea

• Problem: small input perturbations create large score changes.

• Idea: design a DNN with bounded maximum score changes (leveraging Differential Privacy theory).

Make prediction DP

\[ \text{input } x \rightarrow \text{layer 1} \rightarrow \text{layer 2} \rightarrow \text{layer 3} \rightarrow \text{softmax} \]

\[ A_Q(x) \rightarrow \mathbb{E}(A_Q(x)) \]

stability bounds
Key idea

• Problem: small input perturbations create large score changes.
• Idea: design a DNN with bounded maximum score changes (leveraging Differential Privacy theory).
PixelDP architecture

1. Add a new noise layer to make DNN DP.
2. Estimate the DP DNN's mean scores.
3. Add estimation error in the stability bounds.
1. Add a new noise layer to make DNN DP.
2. Estimate the DP DNN's mean scores.
3. Add estimation error in the stability bounds.
PixelDP architecture

1. Add a new noise layer to make DNN DP.
2. Estimate the DP DNN's mean scores.
3. Add estimation error in the stability bounds.
PixelDP architecture

Resilience to *post-processing*: any computation on the output of an \((\varepsilon, \delta)-\text{DP}\) mechanism is still \((\varepsilon, \delta)-\text{DP}\).

1. Add a new noise layer to make DNN DP.
2. Estimate the DP DNN's mean scores.
3. Add estimation error in the stability bounds.
1. Add a new noise layer to make DNN DP.
2. Estimate the DP DNN's mean scores.
3. Add estimation error in the stability bounds.

Compute empirical mean with standard Monte Carlo estimate.
1. Add a new noise layer to make DNN DP.
2. Estimate the DP DNN's mean scores.
3. Add estimation error in the stability bounds.
1. Add a new noise layer to make DNN DP.
2. Estimate the DP DNN's mean scores.
3. Add estimation error in the stability bounds.
Further challenges

• Train DP DNN with noise.
• Control pre-noise sensitivity during training.
• Support various attack norms ($L_1$, $L_2$, $L_∞$).
• Scale to large DNNs and datasets.
Scaling to Inception on ImageNet

• Large dataset: image resolution is 300x300x3.

• Large model:
  • 48 layers deep.
  • 23 millions parameters.
  • Released pre-trained by Google on ImageNet.
1 Draws for prediction
Given an image to classify, we want to detect the highest probability label.

1.1 Fixed bounds
We first ask how many draws we need to distinguish the highest probability with probability at least 0.99. We start from Hoeffding's inequality applied to a Bernoulli variable that

\[ P(\bar{X} - p | \bar{X}) < 2e^{2\frac{p^2}{n}} \]

We now note the difference between the highest and second highest label probability, and rewrite the variable so that the bounds do not overlap

\[ P(\bar{X} - p | \bar{X}) < 2ke^{\frac{p^2}{2n}} \]

Finally, we apply a union bound over the k possible labels, and end up with:

\[ P(\bar{X} - p | \bar{X}) < 2ke^{\frac{p^2}{2n}} \]

For instance in datasets with k = 10, distinguishing the top label with probability at least 0.99 when it is bigger than the second one by 0.1 requires n ⇐ 1500 draws.

---

Scaling to Inception on ImageNet
Scaling to Inception on ImageNet

PixelDP auto-encoder

Post-processing

Inception-v3

 Draws for prediction

Given an image to classify, we want to detect the highest probability label.

1. Fixed bounds

We first ask how many draws we need to distinguish the highest probability with probability at least $1/2$. We start from Hoeffding's inequality applied to a Bernoulli variable that...

$$\bar{X} = \frac{1}{n} \sum_{i=1}^{n} X$$

$$P \left( | \bar{X} - p | > \epsilon \right) \leq \frac{2e^{2\epsilon^2}}{n}$$

We now note the difference between the highest and second highest label probability, and rewrite the variable so that the bounds do not overlap.

Finally, we apply a union bound over the $k$ possible labels, and end up with:

$$P \left( | \bar{X} - p | > \epsilon \right) \leq \frac{2ke^{2\epsilon^2}}{n^2} \leq \frac{1}{n}$$

For instance in datasets with $k = 10$, distinguishing the top label with probability at least 0.99 when it is bigger than the second one by 0.1 requires $n \approx 1500$ draws.
PixelDP Outline

Motivation

Design

Evaluation
Evaluation:

1. Guaranteed accuracy on large DNNs/datasets
2. Are robust predictions harder to attack in practice?
3. Comparison with other defenses against state-of-the-art attacks.
Methodology

Five datasets:

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Image size</th>
<th>Number of Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ImageNet</td>
<td>299x299x3</td>
<td>1000</td>
</tr>
<tr>
<td>CIFAR-100</td>
<td>32x32x3</td>
<td>100</td>
</tr>
<tr>
<td>CIFAR-10</td>
<td>32x32x3</td>
<td>10</td>
</tr>
<tr>
<td>SVHN</td>
<td>32x32x3</td>
<td>10</td>
</tr>
<tr>
<td>MNIST</td>
<td>28x28x1</td>
<td>10</td>
</tr>
</tbody>
</table>

Three models:

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Number of Layers</th>
<th>Number of Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inception-v3</td>
<td>48</td>
<td>23M</td>
</tr>
<tr>
<td>Wide ResNet</td>
<td>28</td>
<td>36M</td>
</tr>
<tr>
<td>CNN</td>
<td>3</td>
<td>3M</td>
</tr>
</tbody>
</table>

Metrics:

- Guaranteed accuracy.
- Accuracy under attack.

Attack methodology:

- State of the art attack [Carlini and Wagner S&P'17].
- Strengthened against our defense by averaging gradients over multiple noise draws.
Guaranteed accuracy on ImageNet with Inception-v3

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy (%)</th>
<th>Guaranteed accuracy (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0.05</td>
</tr>
<tr>
<td>Baseline</td>
<td>78</td>
<td>-</td>
</tr>
<tr>
<td>PixelDP: L=0.25</td>
<td>68</td>
<td>63</td>
</tr>
<tr>
<td>PixelDP: L=0.75</td>
<td>58</td>
<td>53</td>
</tr>
</tbody>
</table>

More DP noise

Meaningful **guaranteed accuracy** for ImageNet!
What if we only act on robust predictions?
(e.g. if not robust, check ticket)

Dataset: CIFAR-10

Accuracy on robust predictions

- Baseline
- Precision: threshold 0.05

Attack size (2-norm) vs Accuracy (top 1)
Accuracy on robust predictions

- Baseline
- Precision: \textbf{threshold 0.1}
- Recall: \textbf{threshold 0.1}

\textbf{Dataset}: CIFAR-10

\textbf{Comparison}: Madry+ '17

If we increase the robustness threshold: better accuracy, less predictions.
Comparison with other provable defenses

PixelDP scales to larger models, yielding better accuracy and robustness.
PixelDP summary

• PixelDP is the first defense that:
  • Gives **attack-independent guarantees** against norm-bounded adversarial attacks.
  • And **scales** to the largest models and datasets.

• Already extensions by others!
  • Improve the bounds at a given noise level (Li+ '18; Cohen+ '19).
  • Use other noise distributions (Pinot+ '19).
  • Adapt optimization (Rakin+ '18).