Understanding and Mitigating the Tradeoff Between Robustness and Accuracy

Abhirav Gholba; Wissam Kontar

1) Intriguing properties of Neural Networks

- Deep Neural Networks are highly expressive, which is the reason they succeed but also the reason why they produce uninterpretable solutions with counter-intuitive properties.
- First property: Any linear combination of activations of a layer stores feature information invariantly. It is the space rather than individual units of neural networks that contains the semantic information.
- Second property: Input-output mapping in NN is not perfect. Imperceptible perturbations can cause a model to misclassify. Moreover, the specific nature of these perturbations is not a random artifact of learning: the same perturbation can cause a different network, that was trained on a different subset of the dataset, to misclassify the same input.[1]

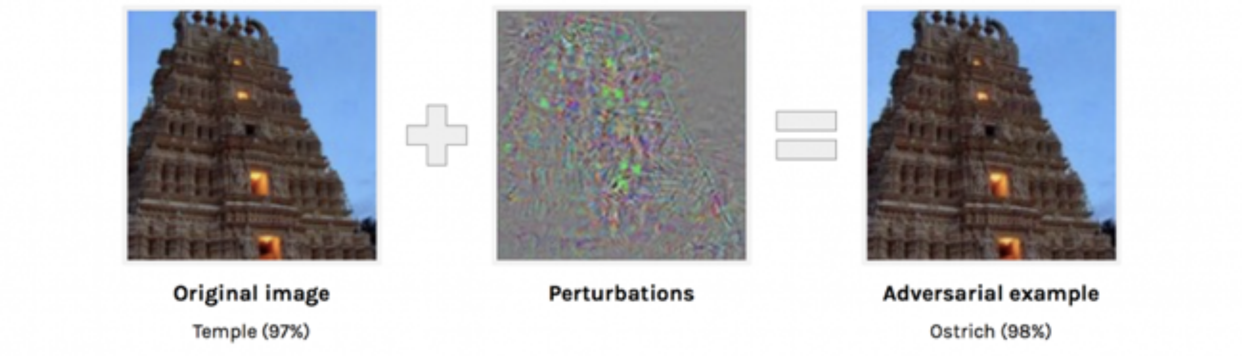
2) Advent of Adversarial Machine Learning

- The existence of the adversarial negatives appears to be in contradiction with the network's ability to achieve high generalization performance. Indeed, if the network can generalize well, how can it be confused by adversarial negatives, which are indistinguishable from the regular examples? Possible explanation is that the set of

[1] "Intriguing properties of neural networks." 19 Feb. 2014, https://arxiv.org/abs/1312.6199.

adversarial negatives is of extremely low probability, and thus is never (or rarely) observed in the test set, yet it is dense (much like the rational numbers), and so it is found near virtually every test case.[2]

- There are two types of adversarial attacks:
  a. Black Box
  b. White Box (our focus)



Common adversarial attacks

1. The Fast Gradient Sign Method (FGSM) attack[3]
   a. One of the earliest attacks.
   b. One shot attack, meaning an adversarial example can be generated in a single step of computation.
   c. Q - Why does a simple linear attack work against non-linear networks?
      A - Hypothesis that NN are too linear to resist linear perturbations. ReLU, LSTMs, maxout are all designed to behave in a linear way. Non-linear sigmoid tuned to be linear.
   d. Perturbation: $x + \varepsilon\ sgn(\square_x\ L(\theta, x, y))$
   e. Some numbers on how effective the attack was at the time on the leading Neural Networks:

[2] "Intriguing properties of neural networks." 19 Feb. 2014, https://arxiv.org/abs/1312.6199.

[3] "Explaining and Harnessing Adversarial Examples." 20 Dec. 2014, https://arxiv.org/abs/1412.6572.

	Error rate	**Confidence**	**ε**
MNIST (softmax)	99.9%	79.3%	0.25
MNIST (maxout)	89.4%	97.6%	0.25
CIFAR-10 (maxout)	87.15%	96.6%	0.1

2. The Projected Gradient Descent (PGD) attack[4]
    a. A very strong first order attack which is used till this day.
    b. Considered a Universally strong adversarial attack.
    c. Iterative

$$x^{t+1} = \Pi_{x+S}(x^t + \alpha\, sgn(\nabla_x\, L(\theta, x, y)))$$

    d. Finds perturbations in $l_2$and l_∞ball around input x.
    e. Experiments[5]: In the $l_\infty(\varepsilon = \frac{2}{255})$ case for a model trained on CIFAR-10 (ResNet), standard accuracy is 99.20% and robust accuracy is 69.10%. We see the same pattern between standard and robust accuracies for other values of ε. We see a clear **trade-off between robustness and accuracy**.

3) Robust Physical-World Attack

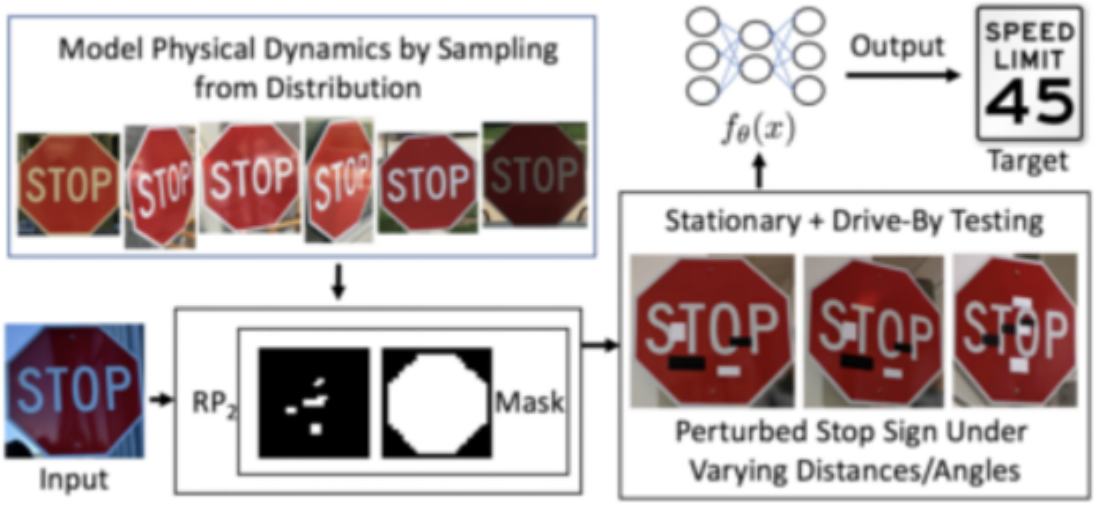
- Given that emerging physical systems are using DNNs in safety-critical situations, adversarial examples could mislead these systems and cause dangerous situations.
- One such kind of dangerous attacks is the Robust Physical Perturbation (RP2) attack.[6]
- Targeted misclassification on real-world example of traffic stop sign.

[4] "Towards Deep Learning Models Resistant to Adversarial Attacks." 4 Sep. 2019, https://arxiv.org/pdf/1706.06083.

[5] "Robustness May Be at Odds with Accuracy." 9 Sep. 2019, https://arxiv.org/pdf/1805.12152.

[6] "Robust Physical-World Attacks on Deep Learning Models." 10 Apr. 2018, https://arxiv.org/abs/1707.08945.

- Generates robust perturbations that achieve high misclassification rates under various environmental conditions, including viewpoints.


Model Physical Dynamics by Sampling from Distribution
STOP
Output
SPEED LIMIT 45
Target
$f_\theta(x)$
Stationary + Drive-By Testing
Input
RP$_2$
Mask
Perturbed Stop Sign Under Varying Distances/Angles

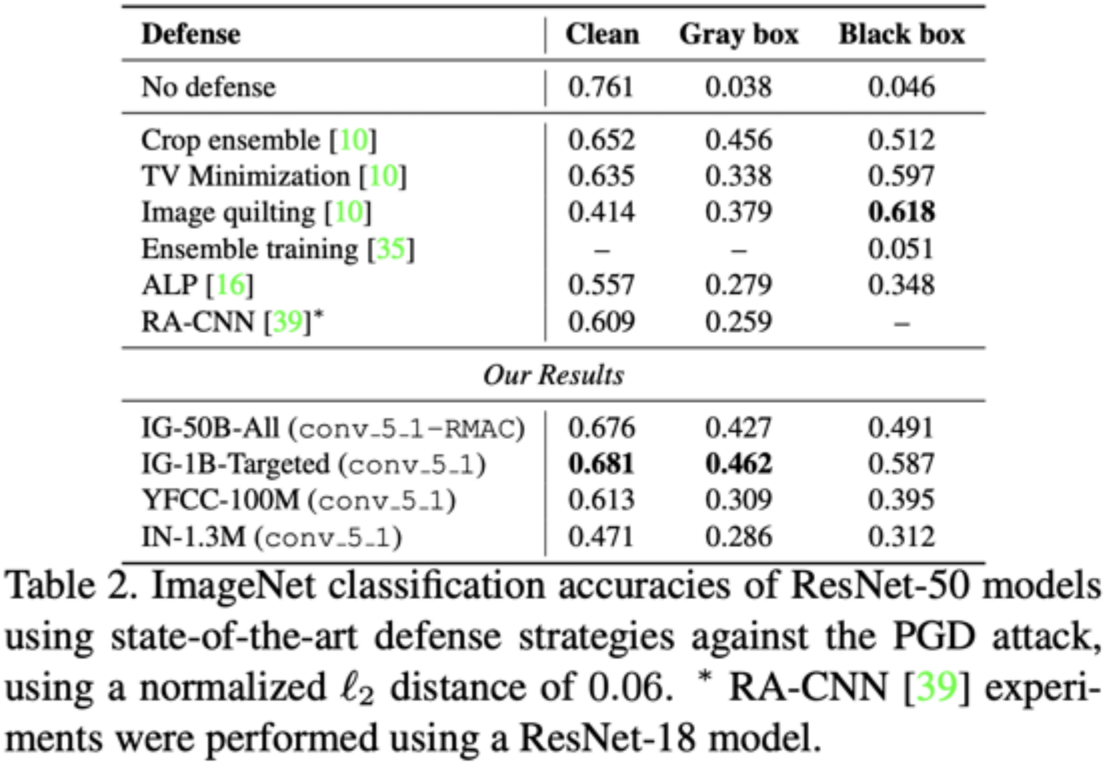
- RP2 pipeline overview. The input is the target Stop sign. RP2 samples from a distribution that models physical dynamics (in this case, varying distances and angles), and uses a mask to project computed perturbations to a shape that resembles graffiti. The adversary prints out the resulting perturbations and sticks them to the target Stop sign.

4) Robust Defense using Web-scale Nearest Neighbor Search

- Many defenses have been proposed against adversarial attacks. Examples include - Hardening of models using adversarial examples in training, defensive distillation (weak), feature squeezing and others.
- One such recent defense was proposed by a team from MIT and Facebook AI called as “Defense Against Adversarial Images using Web-Scale Nearest-Neighbor Search”[7].
- Method

[7] "Defense Against Adversarial Images using Web-Scale ...." 5 Mar. 2019, https://arxiv.org/abs/1903.01612.

    - “Off-manifold” adversarial images.
    - Approximate the projection of an adversarial example onto the image manifold by the finding nearest neighbors in the image database.
    - Classify the “projection” of the adversarial example.
- Pros:
    - Demonstrate the feasibility of web-scale nearest-neighbor search as a defense mechanism.
    - Provides a new avenue for defending methods.
- Cons:
    - Computational cost is large, the results benefit from their powerful hardware.
    - **Unable to mitigate the cost in accuracy for the gain in robustness**.



Defense	Clean	Gray box	Black box
No defense	0.761	0.038	0.046
Crop ensemble [10]	0.652	0.456	0.512
TV Minimization [10]	0.635	0.338	0.597
Image quilting [10]	0.414	0.379	**0.618**
Ensemble training [35]	–	–	0.051
ALP [16]	0.557	0.279	0.348
RA-CNN [39]*	0.609	0.259	–
Our Results			
IG-50B-All (conv_5_1-RMAC)	0.676	0.427	0.491
IG-1B-Targeted (conv_5_1)	**0.681**	**0.462**	0.587
YFCC-100M (conv_5_1)	0.613	0.309	0.395
IN-1.3M (conv_5_1)	0.471	0.286	0.312

Table 2. ImageNet classification accuracies of ResNet-50 models using state-of-the-art defense strategies against the PGD attack, using a normalized ℓ_2 distance of 0.06. * RA-CNN [39] experiments were performed using a ResNet-18 model.

5) Tradeoff between Robustness and Accuracy[8]

Robustness problem

- We are interested in reliable machine learning and thus we expect the trained model to work well on test data that is hard (noisy, adversarial attacks, etc...).
- Empirical analysis shows that standard accuracy decreases when robust training is performed (training on adversarial attacks).

Some explanations on the tradeoff phenomenon

- The optimal accurate predictor is not robust: this does not hold in practice, as with consistent perturbations a model can be both accurate and robust.
- Model class is not expressive enough to contain both robust and accurate predictors even if those exist: currently we design very expressive networks that are capable of representing any arbitrary function.
- The tradeoff is inherent and exists even with infinite data: empirical analysis shows that the gap between standard error of adversarial training and standard training decreases with the availability of more training data. This suggests that with infinite data the tradeoff does not exist, and this is rather a problem of finite complexity than fundamental inherent behavior.

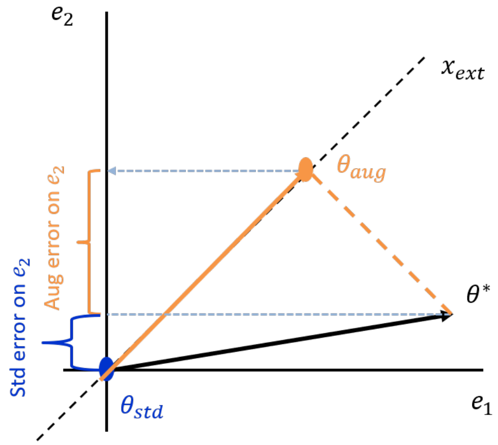
Extra data can hurt sometimes

- The spline example shows us that adding valid data can command local fit at the expense of global fit and thus generalization fails.
- When the inductive bias is not designed to acknowledge the tradeoff between accuracy and robustness, generalization cannot be achieved.

[8] "Understanding and Mitigating the Tradeoff Between ...." 25 Feb. 2020, https://arxiv.org/abs/2002.10716.

Visualization of the tradeoff

- When minimizing the min-norm interpolant the augmented estimator (robust estimator) projects θ^* on the extra data added X_{ext} . This involves a new dimension in the null space e_2, which is a costly dimension as determined by the population covariance Σ. In this case, the robust estimator increases the standard error.



- This tradeoff however is not always costly. We denote that when you augment the entire null space, meaning that you incorporate all information that is possible, then in this case standard error will not increase when adding new data. Another case is when the population covariance matrix is equal to the identity matrix, which means that no dimension is expensive and thus projecting on any dimension cannot increase the standard error.

Mitigating the tradeoff

- The principal idea to mitigate the tradeoff is to: a) regularize your robust estimator towards the standard estimator, b) try to learn which dimension in the null space is expensive and avoid projecting your estimators towards that direction.
- Step (a) is done by changing the inductive bias to regularize towards standard estimator when fitting extra data
- Step (b) is done through the method of Robust Self Training (RST). Essentially, using the standard estimator to pseudo-label unlabeled

data. That gives an idea on the population covariance matrix and allows to study which dimension in the data null space is expensive.

Takeaways

- Sometimes adding valid data can actually hurt the model
- Unlabeled data when added can help in robustness through estimating the population covariance
- We might think neural networks can be very expressive and fit anything, but the key problem remains in the inductive bias and generalization; if done wrong will result in a failing model