Adversarial Examples

https://medium.com/@ageitgey/machine-learning-is-fun-part-8-how-to-intentionally-trick-neural-networks-b55da32b7196

What will I be talking about?

- Adversarial Examples (**AEs**): A kind of weird and scary thing that exists in Neural Networks (**NNs**)

- Disclaimer: I will focus on Image-Classification-NNs (**IC-NNs**)

1. What are IC-NNs?
2. What are AEs? +Cool Examples
3. What are IC-NNs really?
4. What are AEs really?
5. How to make AEs?
6. Stuff that I did not talk about

1. What are IC-NNs?

- An IC-NN is basically a function f

f : Image → Prediction

Input for f:
rgb image

x =

Output of f:
class and confidence

f(x) = „Panda“ with 58% confidence

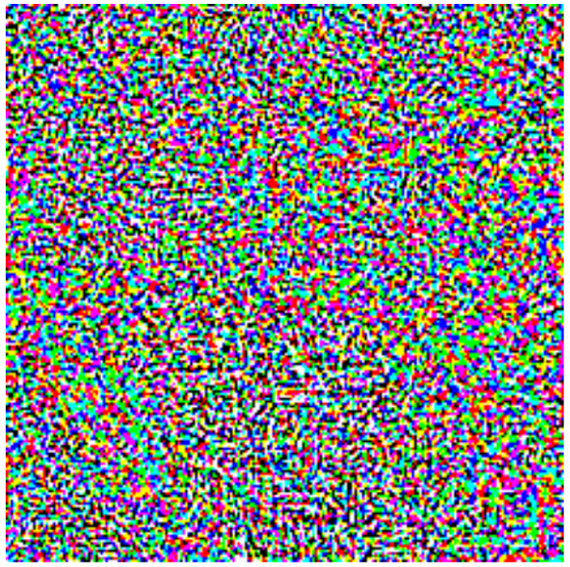
2. What are AEs?

- AEs are Datapoints that make IC-NNs misclassify
- Two Types: Evasion and Impersonation

„Panda“
58% confidence

+ 0.007 x

a little



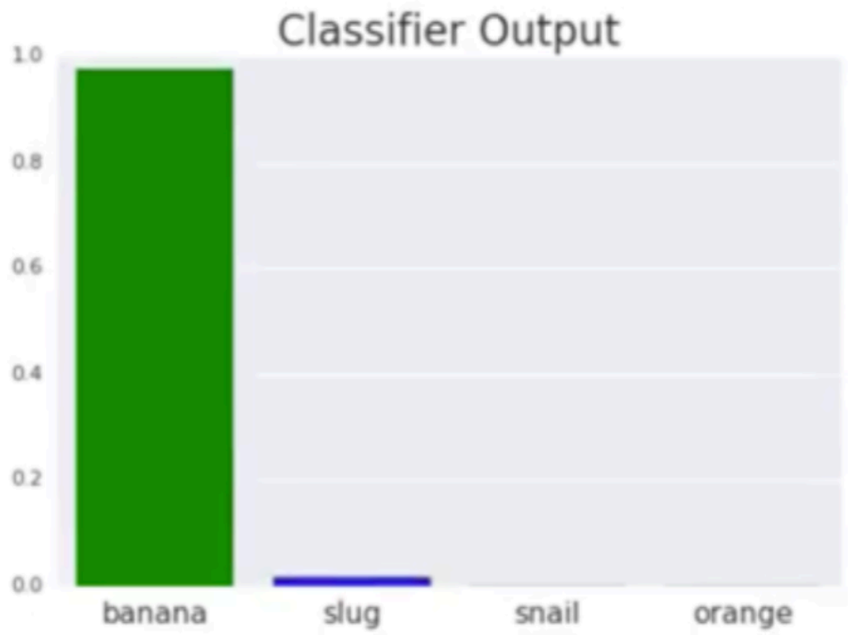
noisy stuff

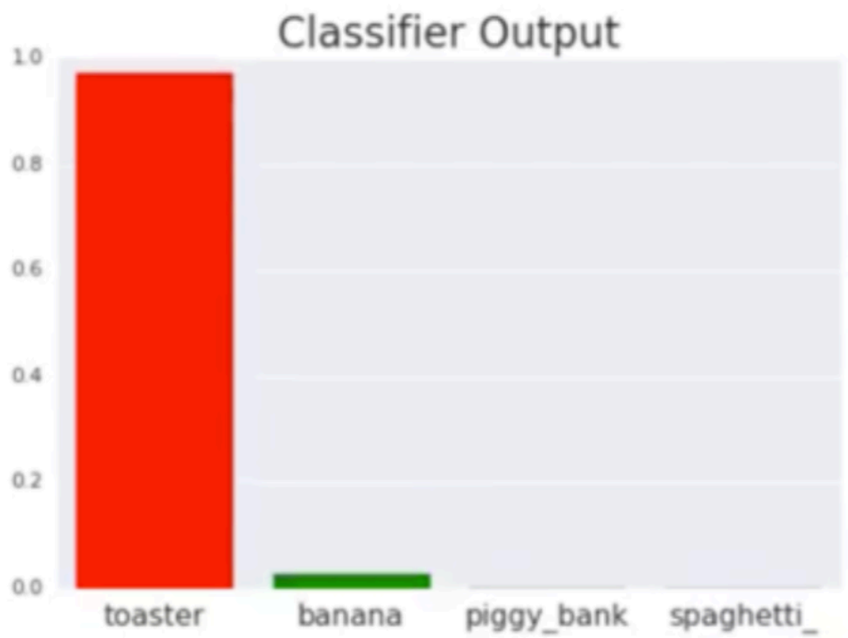
=

„Gibbon“
99% confidence

https://arxiv.org/pdf/1412.6572.pdf

2. Examples of AEs





https://arxiv.org/pdf/1712.09665.pdf

More Examples coming…

3. What are IC-NNs really?

- An IC-NN is a multivariate function f, followed by argmax

$$f : [0,1]^{200k} \rightarrow [0,1]^{20k} \rightarrow [1,20k]x[0,1]$$

Input for f:	Output of f:	Prediction:
256x256 rgb image ~200k pixel values	Confidence value for each of 20k classes	Class with highest confidence

x = 

Array/Vector with 20k values

f(x) = (0.012…. , 0.577, … 0.061)

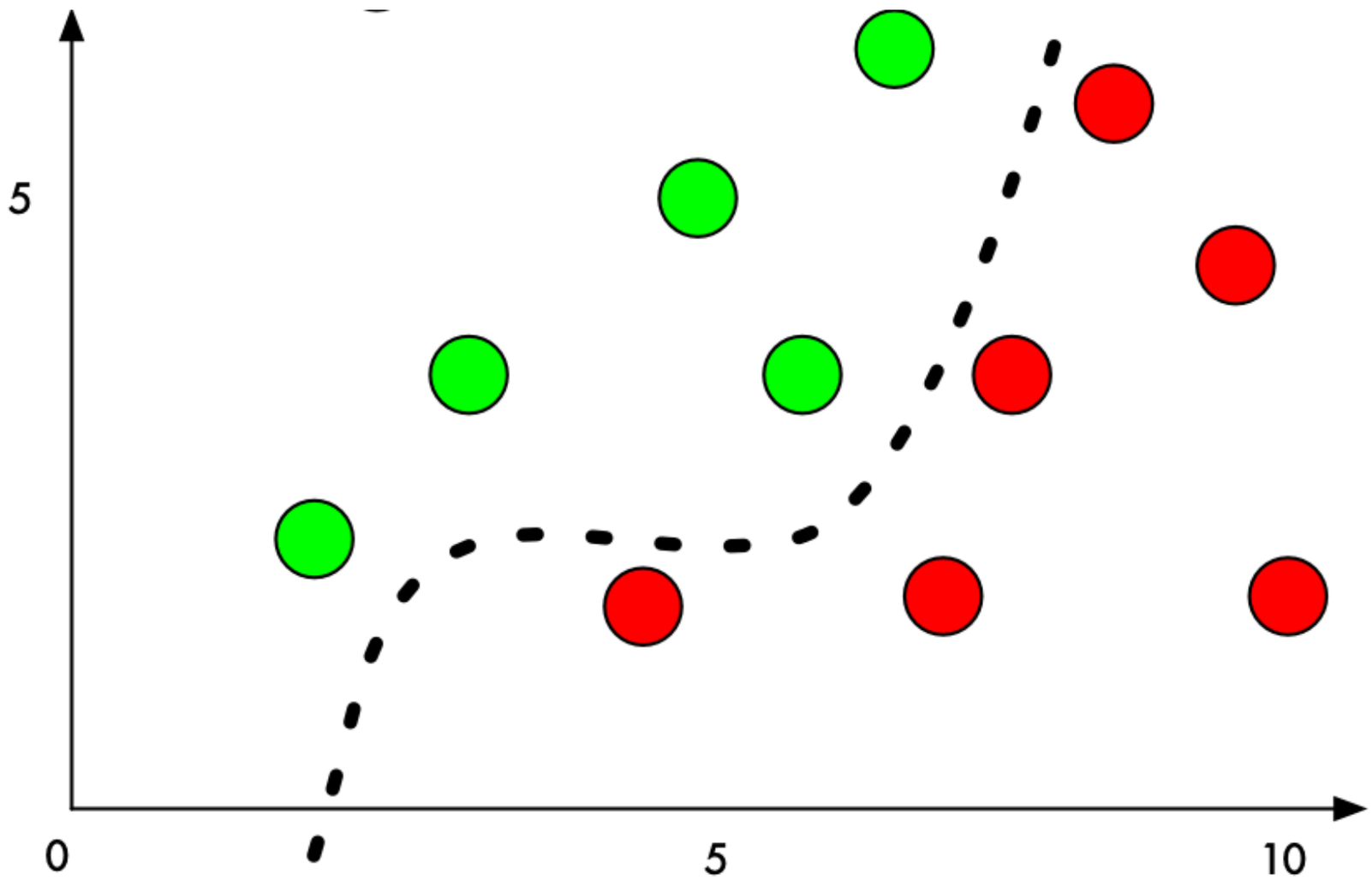
Location for „Panda“

„Panda“
58% confidence

4. What are AEs really?

- Imagine the Input space being 2-dimensional

$$f : [0,1]^{20k} \rightarrow [0,1]^{20k} \rightarrow [1,20k]x[0,1]$$



Pandas:

Gibbons:

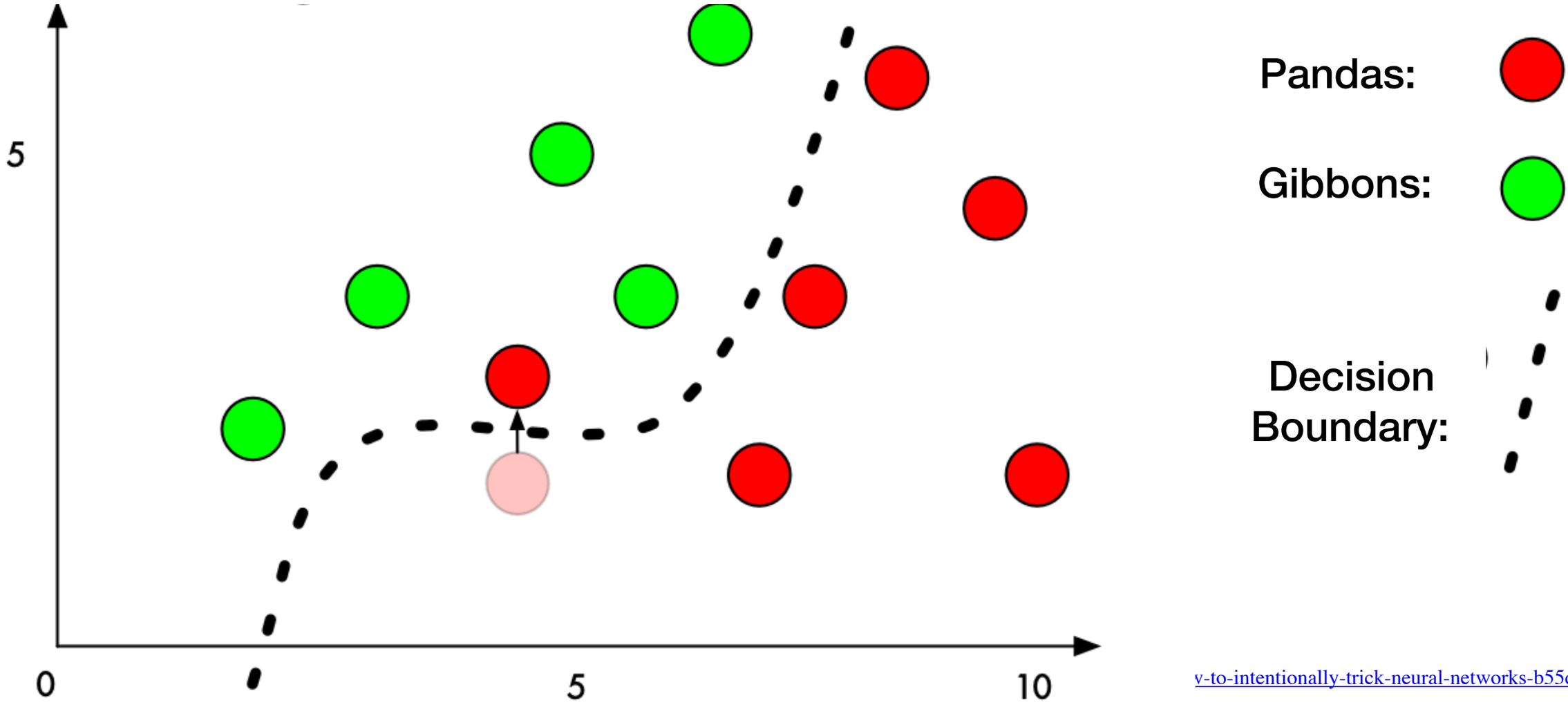
Decision Boundary:

-how-to-intentionally-trick-neural-networks-b55da32b7196

4. What are AEs really?

- AEs push datapoints over the decision boundary

$$f : [0,1]^{20k} \rightarrow [0,1]^{20k} \rightarrow [1,20k]x[0,1]$$



v-to-intentionally-trick-neural-networks-b55da32b7196

4. What are AEs really?

- Problem: Features of high dimensionality

$$f : [0,1]^{200k} \rightarrow [0,1]^{20k} \rightarrow [1,20k]x[0,1]$$

- There is basically decision boundary everywhere
  -> enables evasion

- Every class is closer to all other classes
  -> enables impersonation

It would be cool to visualize the dimensionality problem here…

5. How to make AEs?

- For Evasion: Minimize current class
- For Impersonation: Maximize target class
- If you want non-detection by human eyes: Minimize pixel changes (with L0 Linf etc.)
- (This slide needs more work and will probably become 3 slides with images)
- Also: What are the challenges in creating AEs (White-/Black-Box) and how easy isit to overcome them (e.g. Black-Box transferability)

6. Stuff I did not talk about

- AE for different tasks
    - Spam/Malware Detection
    - Image Segmentation
- Countermeasures
    - Adversarial retraining, Input Reconstruction, Ensembling
    - No „cure for everything“, most countermeasures are broken within a short time frame (mention „Arms Race“ and AE-competitions like kaggle)
- This Slide will also get some more attention, but probably stay 1 slide