FIT5230 Malicious AI

Adversarial Machine Learning I

Overview

- Benign vs Adversarial: attacks on INTegrity
- Semantic adversarial attack
- Noise attack
- Fast Gradient Sign (FGS)
- Fast Gradient Value (FGV)
- Recent adversarial attacks on AI

Security attacks AI

- AI with benign samples
  - all samples correct or (at worst) have random errors

vs

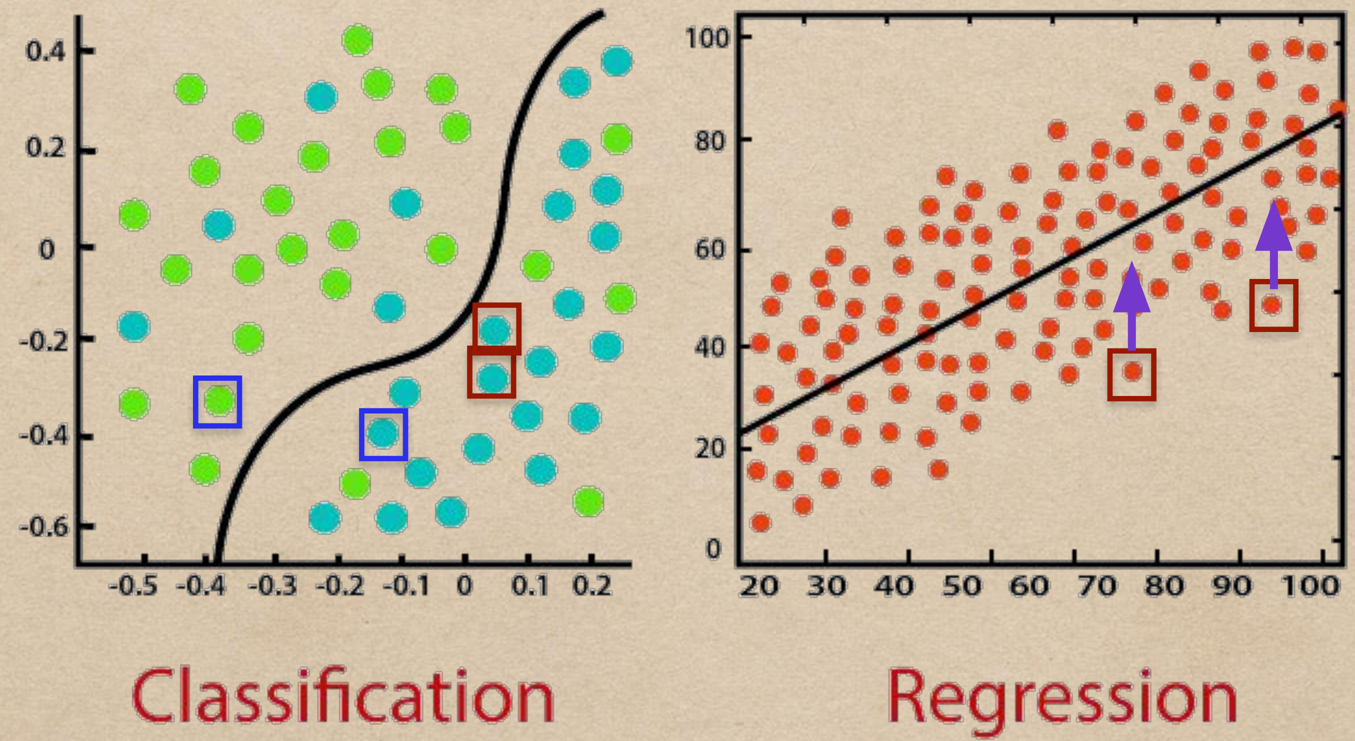
- AI with malicious samples
  - some corrupted samples s.t.
    - bias the learning outcome
    - designed to be undetectable/innocent-looking/indistinguishable. Q: Why?

Security attacks AI

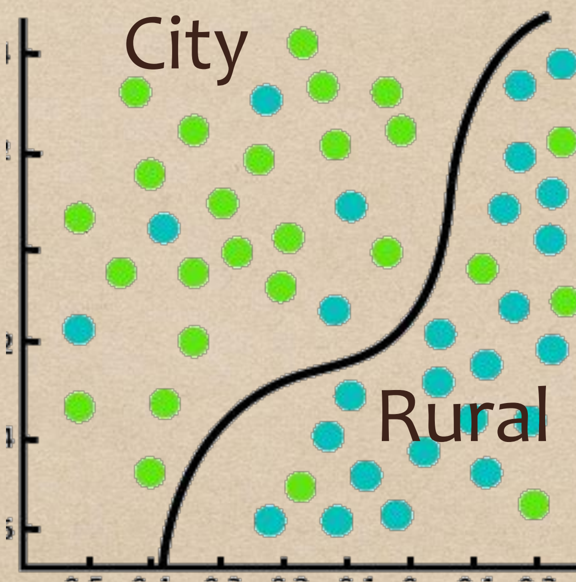
- AI without adversarial attack:
  - samples may be changed due to errors
- AI with adversarial attack:
  - samples intentionally corrupted
  - ?

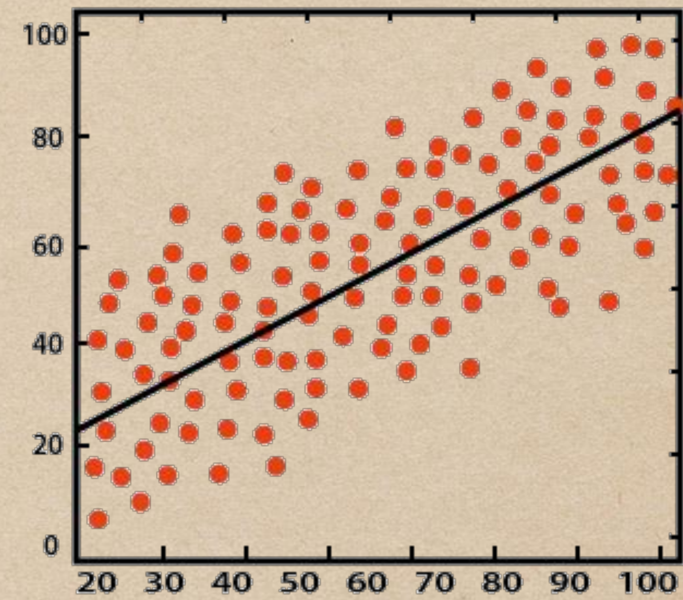
Security attacks AI

- AI with adversarial attack:
    - samples intentionally corrupted



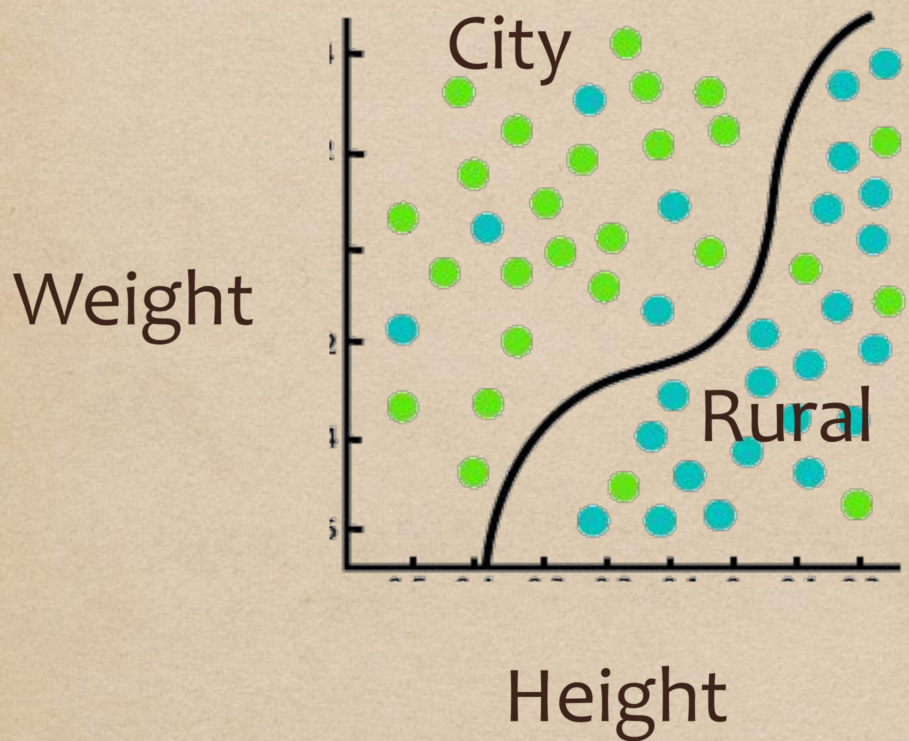
Security attacks AI

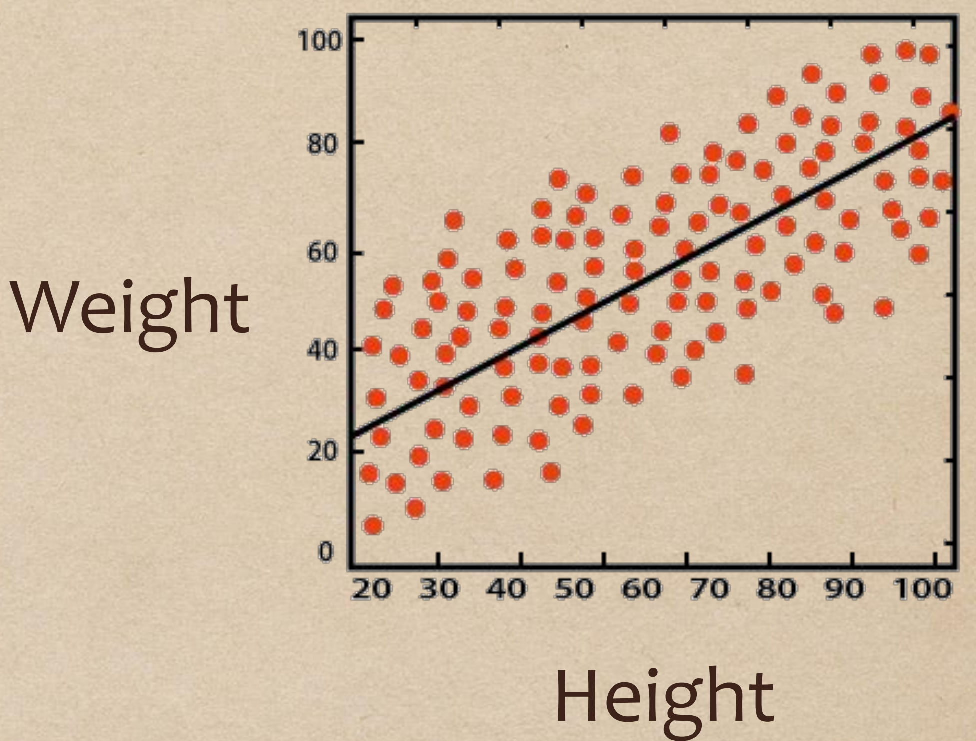




- Classification:
  - learn the boundary, separating classes
  - given unknown class, predict its class based on observed features/attributes

Security attacks AI

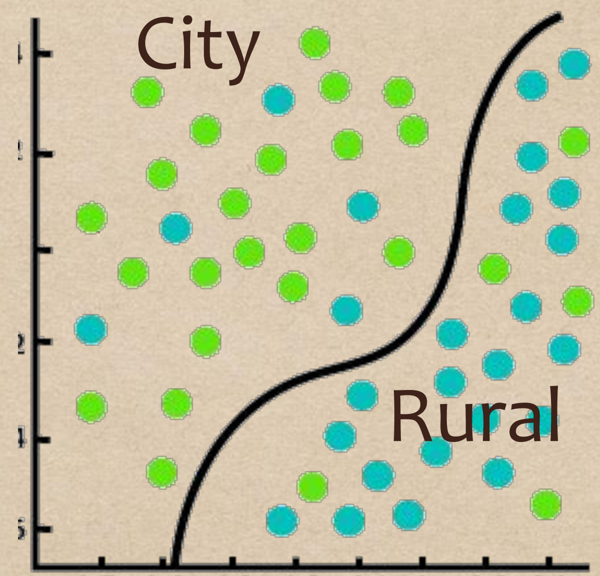




- Classification:
  - e.g. given observed height and weight, predict class/category: s/he’s from city or rural?
  - predict discrete value: 0 or 1

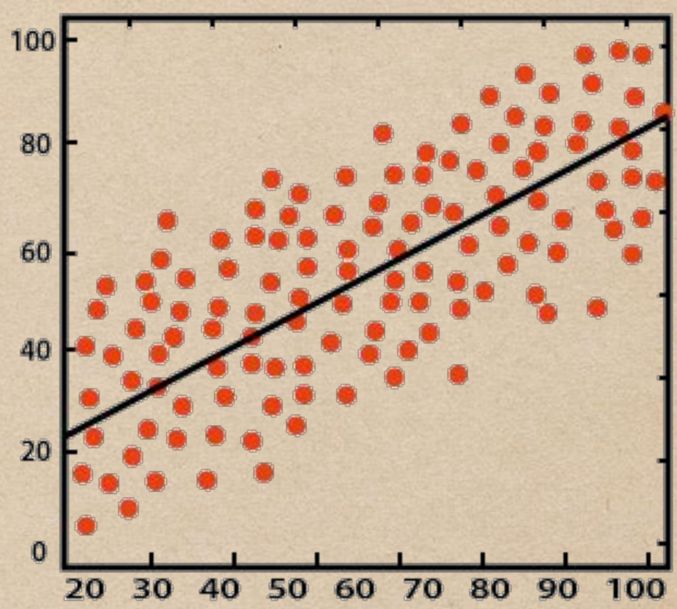
Security attacks AI

Weight



Height

Weight

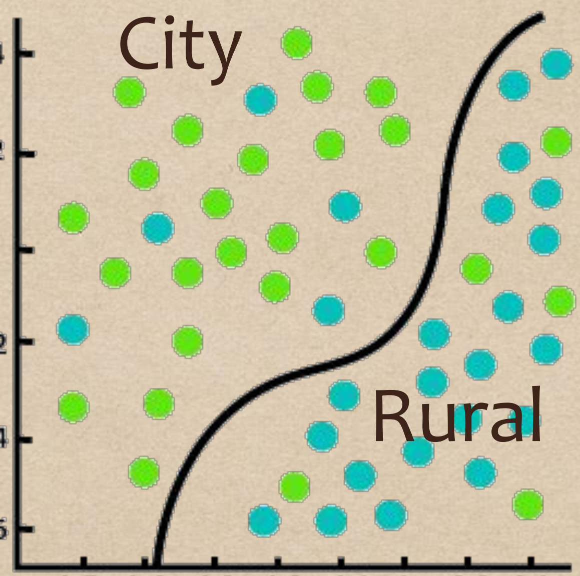


Height

- Regression:
  - learn the function that best represents observed data
  - e.g. Weight depends on Height

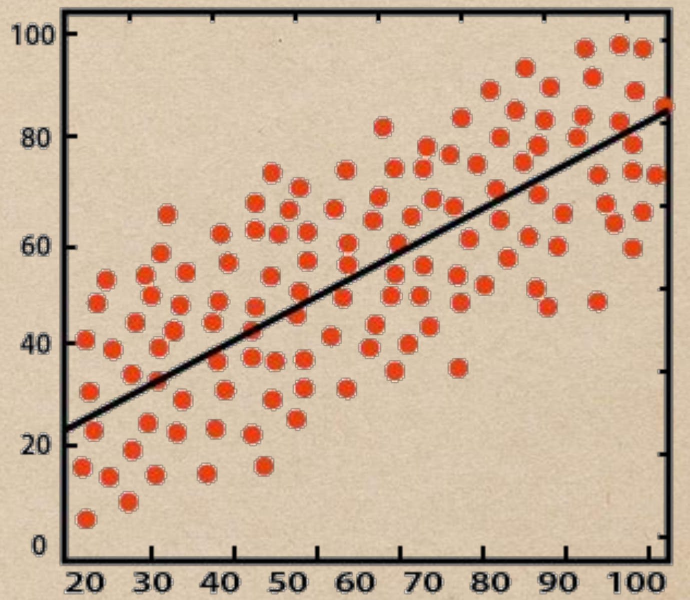
Security attacks AI

Weight



Height

Weight

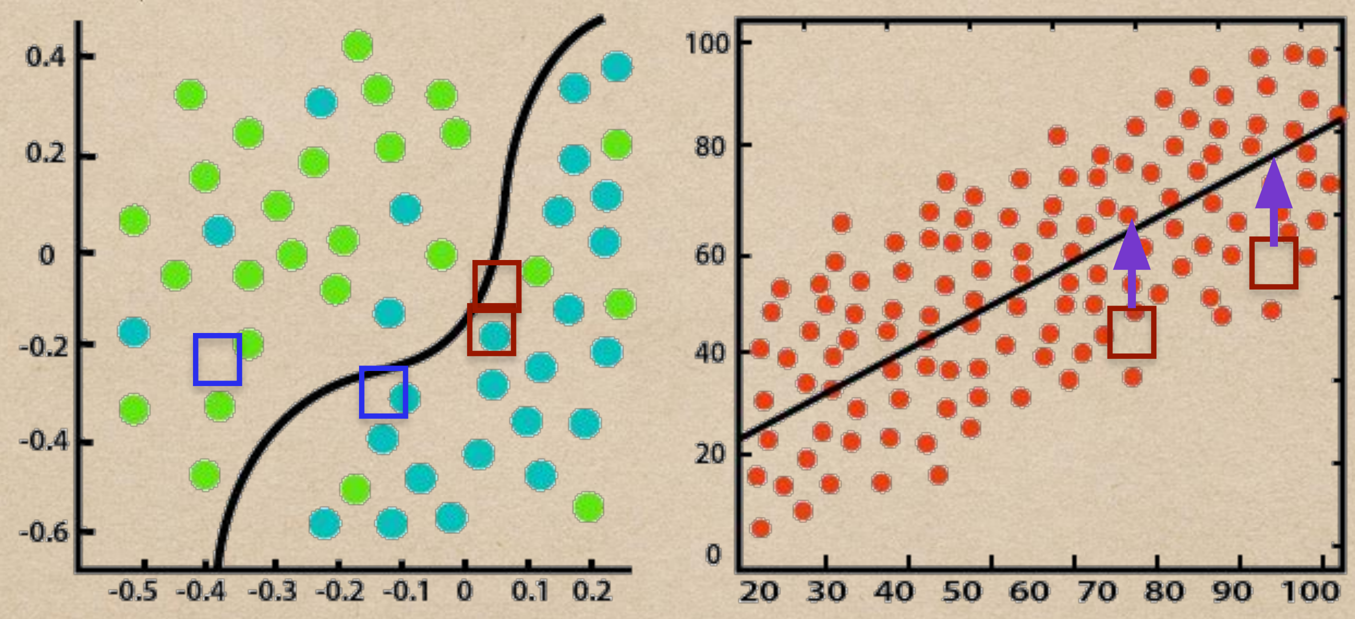


Height

- Regression:
  - e.g. given observed Height, predict Weight
  - predict continuous value: Weight

Security attacks AI

- AI with adversarial attack:
  - samples intentionally corrupted



Classification Regression

- Q: what is the difference between the two cases?
samples changed in both cases

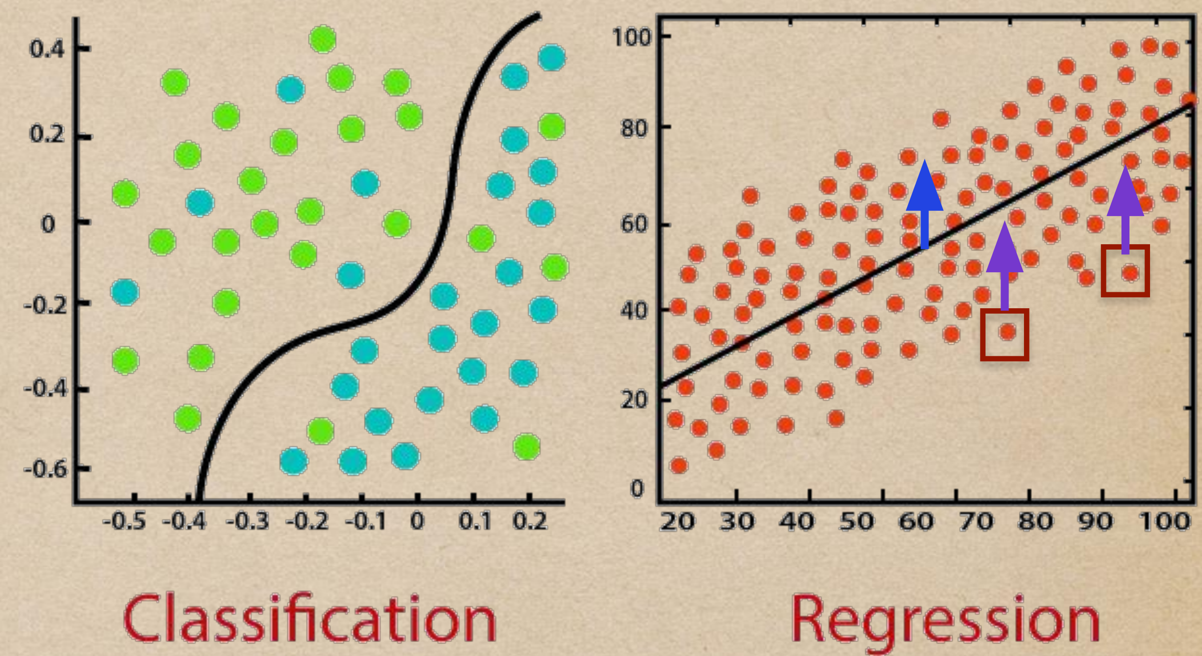
Security attacks AI

- AI without adversarial attack: samples may be changed due to errors
  - learning outcome will change in a random manner
  - e.g. class 1 samples misclassified as class 2, class 2 samples misclassified as class 1

- AI with adversarial attack: samples intentionally corrupted
  - learning outcome will be biased / targetted

Security attacks AI

- An adversarial ML attack
  - is an attack on the INT security property of the AI/ML algorithm



  - the Means: attacks on AI samples' INT, i.e. samples modified
  - the End Goal: attack on AI learning outcome's INT i.e. outcome changed

Adversarial ML

- Adversarial Classification: Dalvi et al. @KDD 2004
- Jul 2024: 1304 citations, 130/year

Adversarial classification
N Dalvi, P Domingos, Mausam, S Sanghai… - Proceedings of the tenth …, 2004 - dl.acm.org
… We define **adversarial classification** … **Adversary**, which attempts to make Classifier **classify** positive instances in T as negative by modifying those instances from x to x = A(x). (**Adversary** …
☆ Save 99 Cite Cited by 1304 Related articles All 10 versions

- Can machine learning be secure?: Barreno et al. 2006
- Jul 2024: 1186 citations, 148/year

Can machine learning be **secure**?
M Barreno, B Nelson, R Sears, AD Joseph… - Proceedings of the 2006 …, 2006 - dl.acm.org
… However, **machine learning** algorithms themselves **can** … "**Can machine** learning be **secure**?" Novel contributions of this paper include a taxonomy of different types of attacks on **machine** …
☆ Save 99 Cite Cited by 1186 Related articles All 16 versions

Adversarial DL

- Intriguing properties of neural networks: Szegedy et al. (incl Goodfellow) @ICLR 2014
- Jul 2024: 16847 citations, 1684/year

Intriguing properties of neural networks

C Szegedy, W Zaremba, I Sutskever, J Bruna… - arXiv preprint arXiv …, 2013 - arxiv.org

… **neural networks** learn input-output mappings that are fairly discontinuous to a significant extent. We can cause the **network** to … D has the following **intriguing properties** which we will sup…

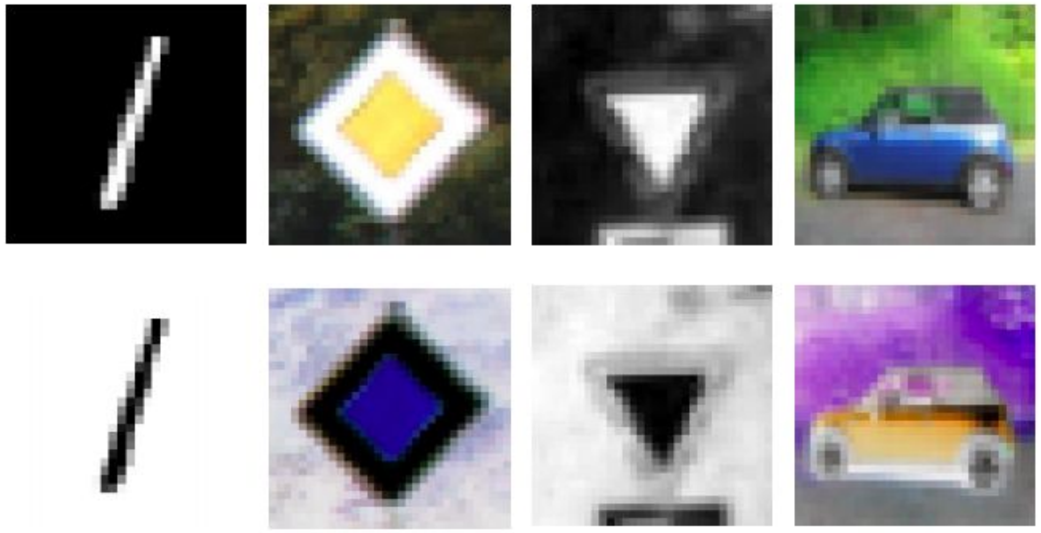
☆ Save 99 Cite Cited by 16847 Related articles All 22 versions »

Semantic Adversarial Attack

On the Limitation of Convolutional Neural Networks in Recognizing Negative Images

Hossein Hosseini, Baicen Xiao, Mayoore Jaiswal and Radha Poovendran
Network Security Lab (NSL), Department of Electrical Engineering, University of Washington, Seattle, WA
{hosseinh, bcxiao, mayoore, rp3}@uw.edu

Abstract—Convolutional Neural Networks (CNNs) have chieved state-of-the-art performance on a variety of computer ision tasks, particularly visual classification problems, where ew algorithms reported to achieve or even surpass the human erformance. In this paper, we examine whether CNNs are apable of learning the semantics of training data. To this end, e evaluate CNNs on negative images, since they share the same tructure and semantics as regular images and humans can lassify them correctly. Our experimental results indicate that hen training on regular images and testing on negative images, ne model accuracy is significantly lower than when it is tested n regular images. This leads us to the conjecture that current aining methods do not effectively train models to generalize the

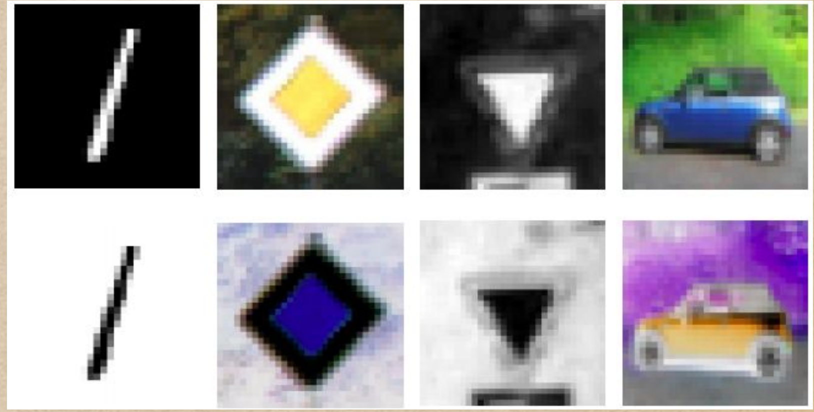


https://arxiv.org/pdf/1703.06857.pdf

Semantic Adversarial Attack

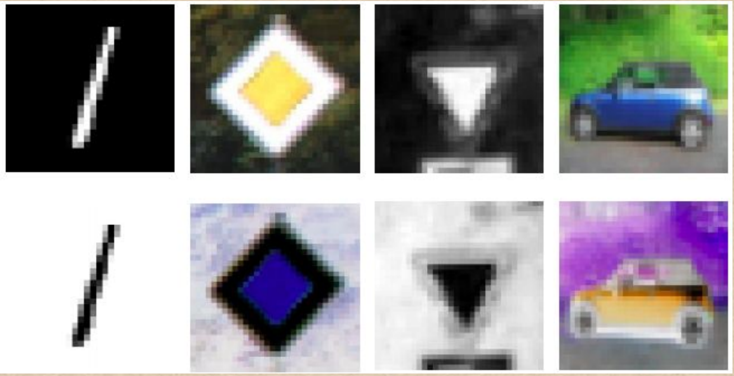
- Can Convolutional Neural Network (CNN) learn semantics of training samples?

- Gist of Semantic adversarial samples: semantically the same as original, but otherwise quite different, e.g. negative images



- Negative image:
    - share same semantics & structure as regular image, humans can easily classify
    - reversed brightness, large pixel-wise perturbation

Semantic Adversarial Attack



- Negative image:
  - reversed brightness, large pixel-wise perturbation, though semantically same
  - e.g. consider a grayscale pixel:
    - 0 = black, (max) 255 = white
    - reverse the brightness: max-pixelvalue
      - black⟶white: 0 ⟶ 255-0 = 255
      - white⟶black: 255 ⟶ 255-255 = 0

Semantic Adversarial Attack

- Limitation of DL based training
  - networks trained to memorise inputs, but not really learn the object structures, so cannot semantically differentiate
  - if test samples not distributed like training samples, won’t work well: aka out-of-distribution (OOD) attack
  - Q: for negative images case, why it is OOD?
- Accuracy (measure) as performance metric not adequate
- Q: what is a good metric to measure performance in this case?

Noise Attack

- naïve untargetted black box attack
  - simply add random noise to affect the learning outcome
  - untargetted: not aiming to bias the outcome towards something specific, just be different
  - black box: not need info on the ML model

  - Q: is the semantic adversarial attack
    - an untargeted attack?
    - a black box attack?

Fast Gradient Signed (FGS) Method

- Goodfellow et al. Explaining & Harnessing Adversarial Examples @ICLR 2015
- Jul 2024: 21192 citations, 2354/year

Explaining and harnessing adversarial examples

IJ Goodfellow, J Shlens, C Szegedy

arXiv preprint arXiv:1412.6572, 2014 · arxiv.org

Ian Goodfellow

DeepMind

Verified email at deepmind.com

Deep Learning

> Several machine learning models, including neural networks, consistently misclassify adversarial examples---inputs formed by applying small but intentionally worst-case perturbations to examples from the dataset, such that the perturbed input results in the model outputting an incorrect answer with high confidence. Early attempts at explaining this phenomenon focused on nonlinearity and overfitting. We argue instead that the primary cause of neural networks' vulnerability to adversarial perturbation is their linear
>
> SHOW MORE

Save · Cite · Cited by 21192 · Related articles · All 19 versions

Generative Adversarial Networks

Generative adversarial nets

I Goodfellow, J Pouget-Abadie… - Advances in neural …, 2014 - proceedings.neurips.cc

… We propose a new framework for estimating **generative** models via **adversarial nets**, in which we simultaneously train two models: a **generative** model G that captures the data …

☆ Save 99 Cite Cited by 70193 Related articles All 67 versions »

Ian Goodfellow

DeepMind

Verified email at deepmind.com

Deep Learning

- Aug 2023: 58748 citations, +12000/year
- Aug 2022: 47467 citations, +14000/year

Fast Gradient Sign (FGS) Method

- Paper: Goodfellow et al.: Explaining and Harnessing Adversarial Examples @ICLR 2015

Explaining and harnessing adversarial examples

IJ Goodfellow, J Shlens, C Szegedy

arXiv preprint arXiv:1412.6572, 2014 · arxiv.org

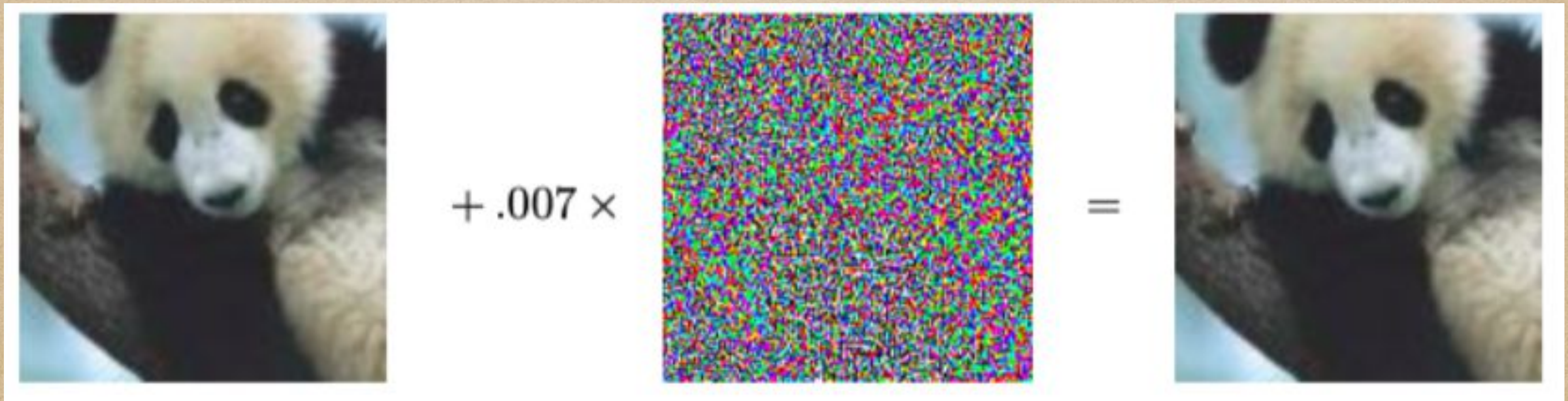
Several machine learning models, including neural networks, consistently misclassify adversarial examples---inputs formed by applying small but intentionally worst-case perturbations to examples from the dataset, such that the perturbed input results in the model outputting an incorrect answer with high confidence. Early attempts at explaining this phenomenon focused on nonlinearity and overfitting. We argue instead that the primary cause of neural networks' vulnerability to adversarial perturbation is their linear

SHOW MORE

Save · Cite · Cited by 21192 · Related articles · All 19 versions

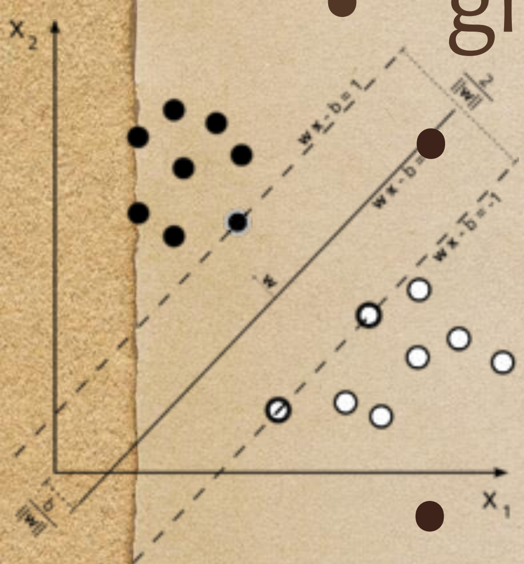
Fast Gradient Sign (FGS) Method

- a white box attack:
  - attacker has complete access to the victim model
- add small perturbations/distortions δ until the classifier labels it as different class/category:
  - $x' = x + \delta$



Machine Learning (ML) points

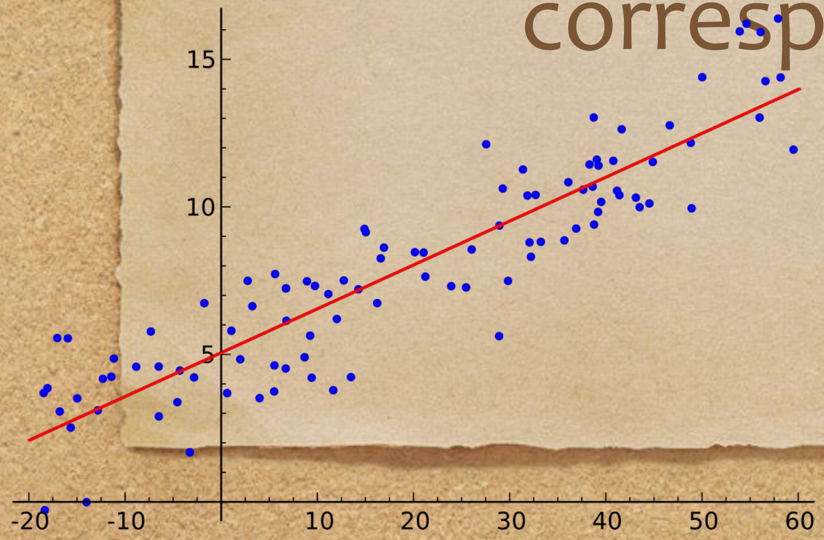
- supervised learning
  - given labelled samples $\{ (\boldsymbol{x}_i , y_i) \}$



    - classification (y_i denotes the label of $\boldsymbol{x}_i$): learn the mapping f: $X \rightarrow$ Y, & given test sample $\boldsymbol{x}$ with ground truth label y, predict its label ŷ
    - regression (y_i denotes the variable dependent on $\boldsymbol{x}_i$): learn the mapping f: $X \rightarrow$ Y, & given test sample $\boldsymbol{x}$ with ground truth variable y, predict its corresponding dependent variable ŷ

Machine Learning (ML) points

- supervised learning
  - given labelled samples $\{ (\boldsymbol{x}_i, y_i) \}$
    - classification (y_i denotes the label of $\boldsymbol{x}_i$): learn the mapping $f: X \rightarrow Y$, & given test sample $\boldsymbol{x}$ with ground truth label y, predict its label $\hat{y}$
    - regression (y_i denotes the variable dependent on $\boldsymbol{x}_i$): learn the mapping $f: X \rightarrow Y$, & given test sample $\boldsymbol{x}$ with ground truth variable y, predict its corresponding dependent variable $\hat{y}$

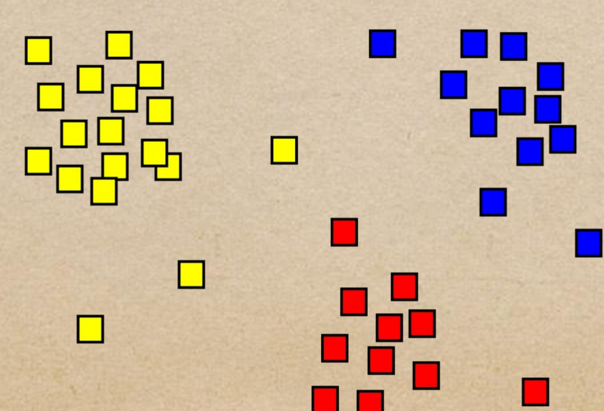


Machine Learning (ML) points

- unsupervised
  - given n unlabelled samples $\{x_i\}$, look for patterns
    - clustering: partition into k subsets S_i s.t.

      minimize: $$\sum_{j=1}^{k}\sum_{i=1}^{n}\left\|x_i^{(j)} - c_j\right\|^2$$

      where c_i is the centroid for cluster j



Machine Learning (ML) points

- loss(distance)/cost function J
  - distance between actual y & predicted ŷ
- Mean Squared Error (MSE)

$$MSE = \frac{1}{n} \sum (y_i - \hat{y}_i)^2$$

- Root MSE (L2 norm)

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

Machine Learning (ML) points

- loss/cost function J
  - distance between actual y & predicted ŷ
- Mean Absolute Error (MAE) (based on L1 norm)

$$MAE = \frac{\sum_{i=1}^{n} | y_i - \hat{y}_i |}{n}$$

- Mean Bias Error (MBE)

$$MBE = \frac{\sum_{i=1}^{n} (y_i - \hat{y}_i)}{n}$$

Machine Learning (ML) points

- loss/cost function J
    - distance between actual & predicted

- Cross Entropy (Negative Log Likelihood) loss

$$-\sum_i y_i \log(\hat{y}_i)$$

where y_i is the actual and $\hat{y}_i$ is the predicted

Fast Gradient Sign (FGS) Method

- perturbation: use gradient ∇ of the loss J wrt the input image $\boldsymbol{x}$, aiming to maximize that loss J

- $\boldsymbol{x}' = \boldsymbol{x} + \delta = \boldsymbol{x} + \varepsilon * \text{sign}(\nabla_{\boldsymbol{x}} J(\theta, \boldsymbol{x}, y))$

  - y = original label of input image $\boldsymbol{x}$
  - ε = multiplier (learning rate) to keep the perturbation small
  - θ = model parameters
  - J = loss

Machine Learning (ML) points

- partial derivative (in the direction of the axes)

  e.g. $\frac{\partial f}{\partial x}$ means rate of change along direction of x-axis.

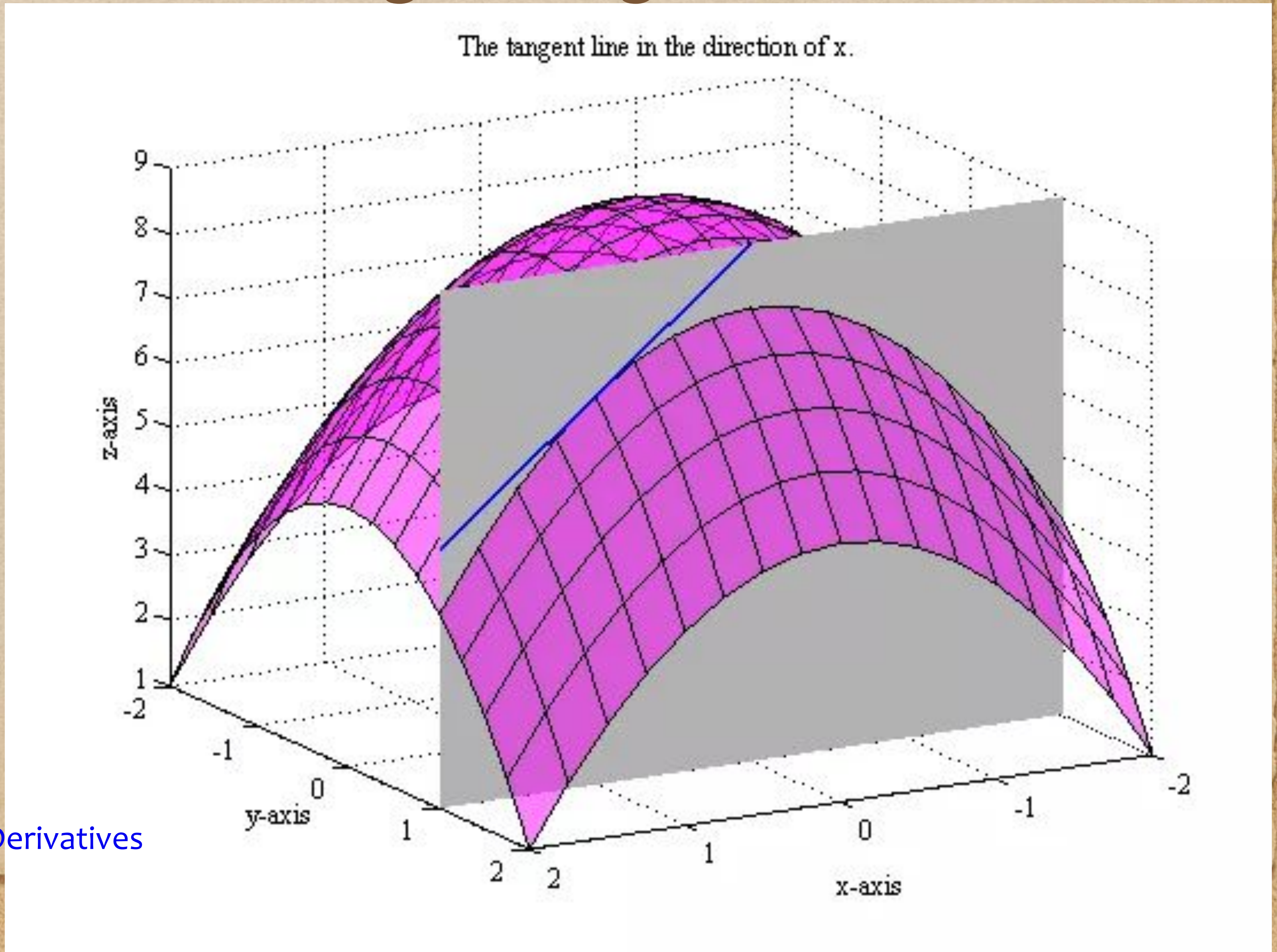
Q: rate of change in direction not along axes?

- directional derivative (in the direction denoted by u)
  - u = unit vector i.e. $||u|| = 1$ representing any direction

Machine Learning (ML) points

- partial derivative (in the direction of the axes)

  e.g. $\frac{\partial f}{\partial x}$ means rate of change along direction of x-axis:

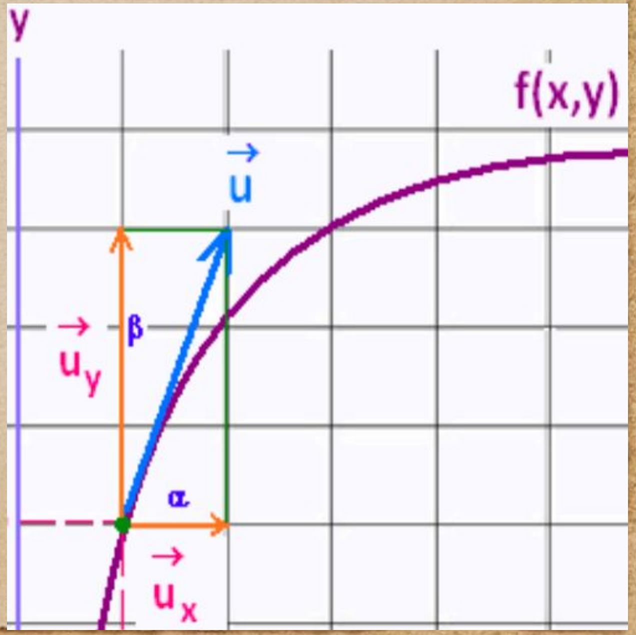


https://www.wikihow.com/Take-Partial-Derivatives

Machine Learning (ML) points

- partial derivative (in the direction of the axes)
  e.g.   means rate of change along direction of
  x-axis. Q: rate of change in direction not along axes?

- directional derivative (in the direction denoted by u)
  - u = unit vector i.e. ||u|| = 1, representing any direction
  - $$D_{\mathbf{u}} f(\mathbf{a}) = \lim_{h \to 0} \frac{f(\mathbf{a} + h\mathbf{u}) - f(\mathbf{a})}{h}$$
  - i.e. @ point a: when tiny change h in the direction of u, f(a) changes to f(a + hu)



Machine Learning (ML) points

- directional derivative

  $$D_{\mathbf{u}} f(\mathbf{a}) = \nabla f(\mathbf{a}) \cdot \mathbf{u} = \|\nabla f(\mathbf{a})\| \cos\theta$$

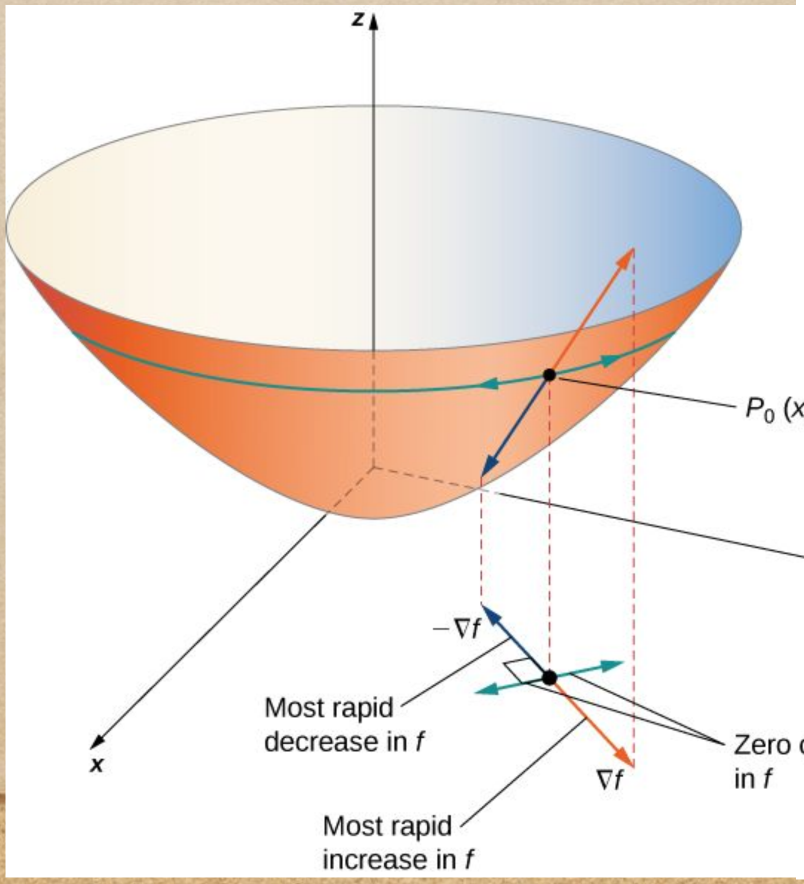
  - this has max value when θ = 0 (i.e. gradient ∇f in the same direction as u) since max of cos θ = cos 0 = 1
    - ∇f (–∇f) points in direction of greatest increase of f, i.e. direction of steepest ascent (descent)

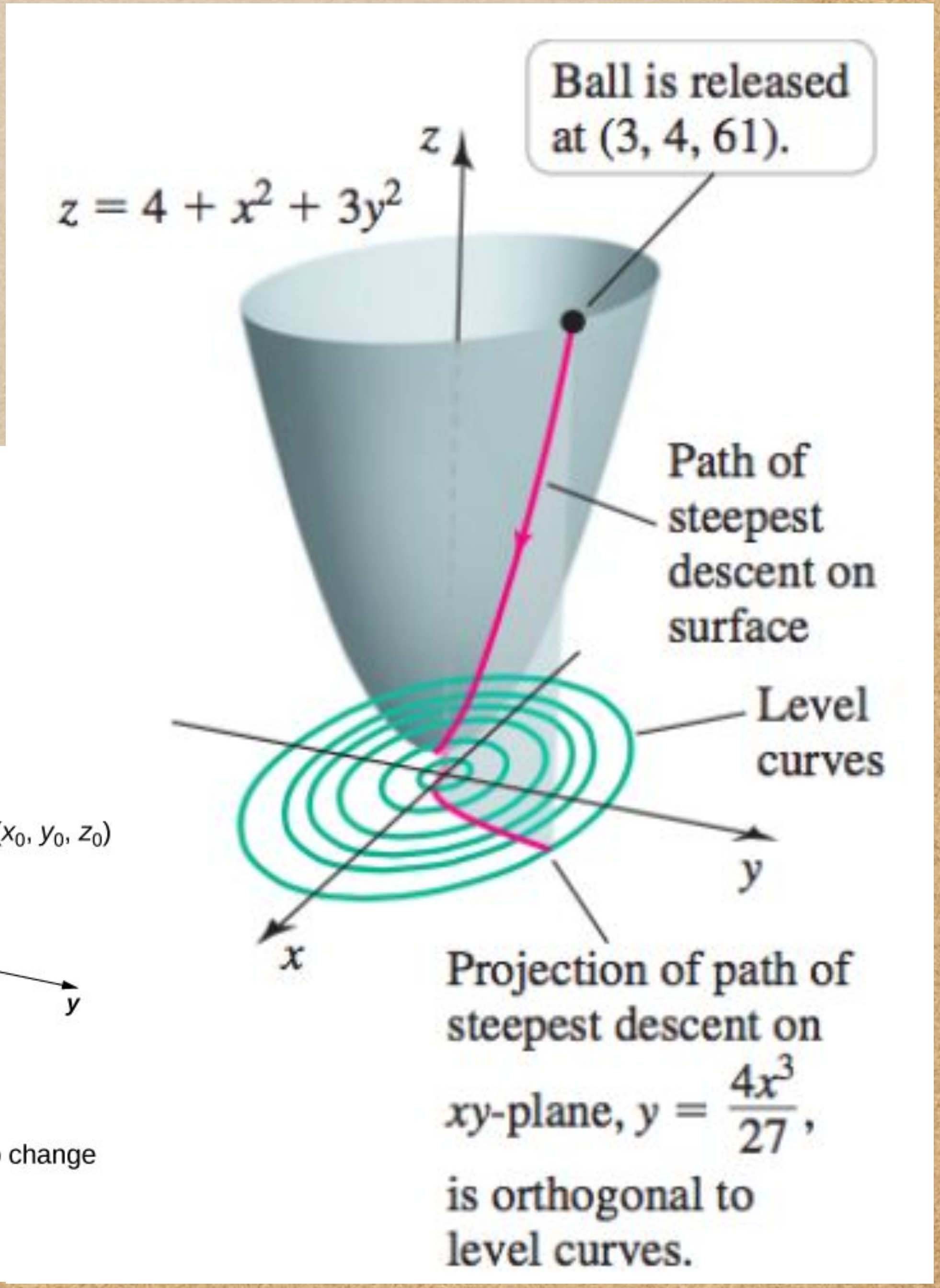
- Note: gradient ∇f(a) (is a vector)

  $$\nabla f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$$

Machine Learning (ML) points

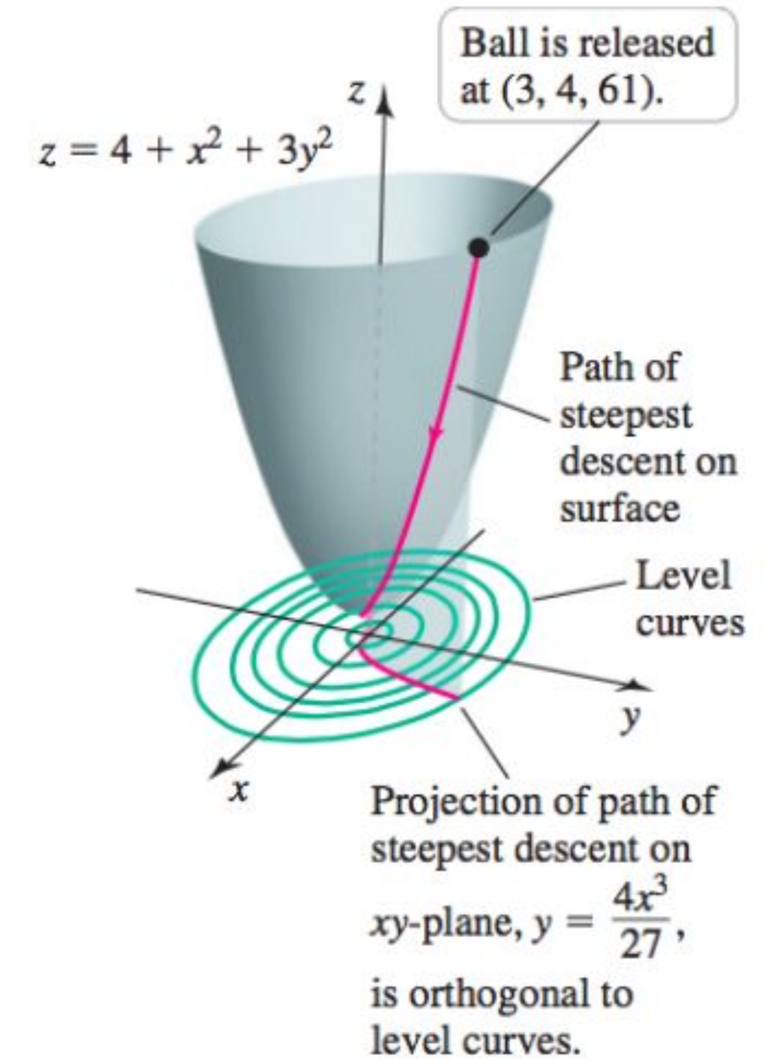
- direction of steepest ascent/descent
    - ∇f ($-\nabla f$) points in direction of greatest increase of f, i.e. direction of steepest ascent (descent)





ML points*

- direction of steepest ascent/descent



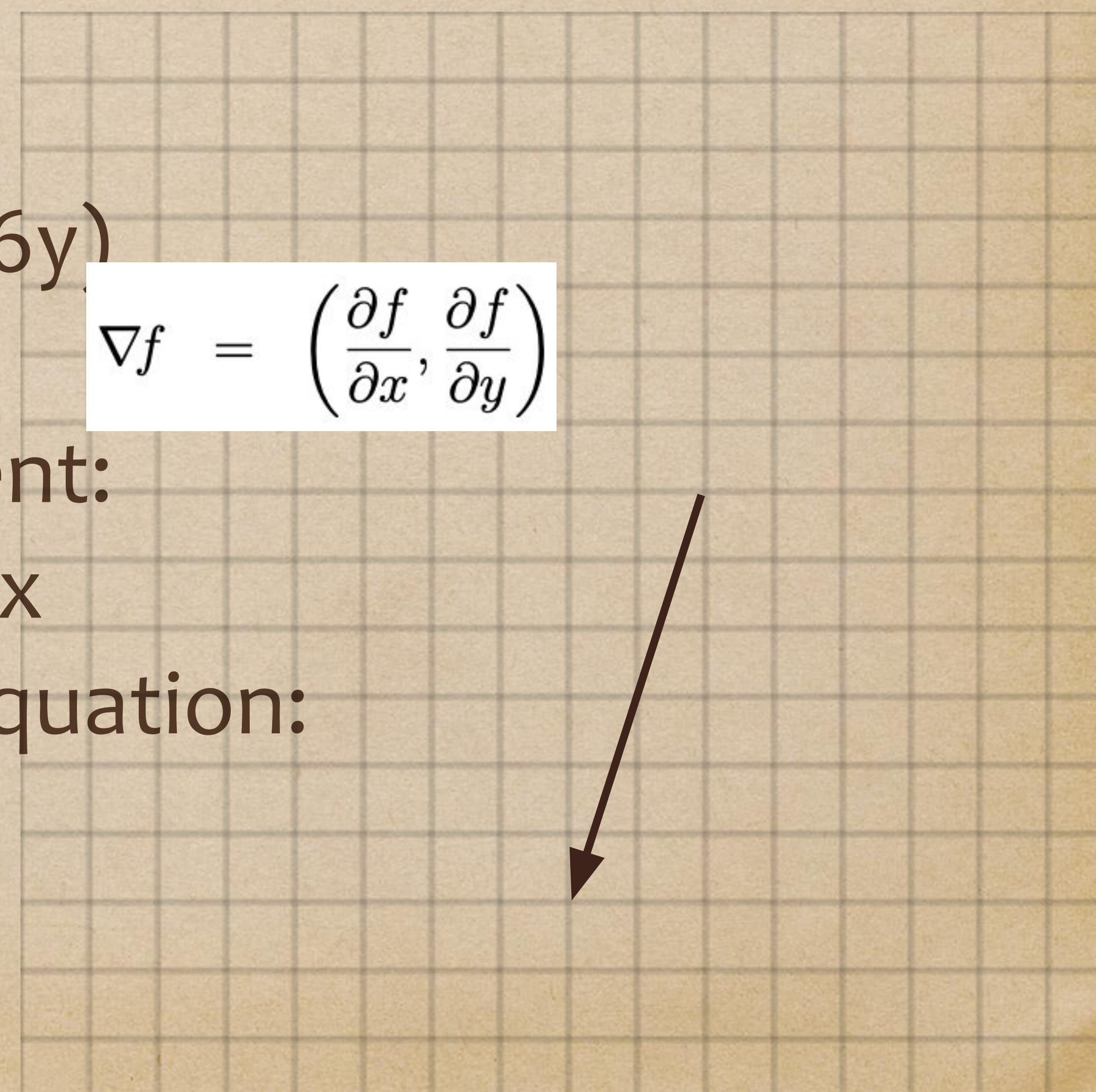
EXAMPLE 7 **Path of steepest descent** Consider the paraboloid $z = f(x, y) = 4 + x^2 + 3y^2$ (Figure 13.73). Beginning at the point $(3, 4, 61)$ on the surface, find the path in the xy-plane that points in the direction of steepest descent on the surface.

SOLUTION Imagine releasing a ball at $(3, 4, 61)$ and assume that it rolls in the direction of steepest descent at all points. The projection of this path in the xy-plane points in the direction of $-\nabla f(x, y) = \langle -2x, -6y \rangle$, which means that at the point (x, y) the line tangent to the path has slope $y'(x) = (-6y)/(-2x) = 3y/x$. Therefore, the path in the xy-plane satisfies $y'(x) = 3y/x$ and passes through the initial point $(3, 4)$. You can verify that the solution to this differential equation is $y = 4x^3/27$ and the projection of the path of steepest descent in the xy-plane is the curve $y = 4x^3/27$. The descent ends at $(0, 0)$, which corresponds to the vertex of the paraboloid (Figure 13.73). At all points of the descent, the curve in the xy-plane is orthogonal to the level curves of the surface.

Machine Learning (ML) points*

- direction of steepest ascent/descent
  - ∇f ($-\nabla f$) points in direction of greatest increase of f, i.e. direction of steepest ascent (descent)
  - e.g. $f(x,y) = 4 + x^2 + 3y^2$
  - $\nabla f = (\partial f/\partial x, \partial f/\partial y) = (2x,6y)$
  - $-\nabla f = (-2x,-6y)$
  - slope dy/dx of this gradient:
    - $dy/dx = (-6y)/(-2x) = 3y/x$
    - solve this differential equation:
      - ...
      - $y = (4x^3)/(27)$

$$\nabla f = \left(\frac{\partial f}{\partial x}, \frac{\partial f}{\partial y}\right)$$



Fast Gradient Sign (FGS) Method

- perturbation: use gradient ∇ of the loss J wrt the input image $\boldsymbol{x}$, aiming to maximize that loss
  - $\boldsymbol{x}' = \boldsymbol{x} + \delta = \boldsymbol{x} + \varepsilon$*sign($\nabla_{\boldsymbol{x}}$J(θ,$\boldsymbol{x}$,y))

```
adv_x = x + ε*signedGrad
```

- $\nabla_{\boldsymbol{x}}$: Gradient only wrt $\boldsymbol{x}$ because model already trained, so parameters θ constant
- $\nabla_{\boldsymbol{x}}$J(·,$\boldsymbol{x}$,·):

```
gradient = tape.gradient(J,x)
```

- sign($\nabla_{\boldsymbol{x}}$J(·,$\boldsymbol{x}$,·)):

```
signedGrad = tf.sign(gradient)
```

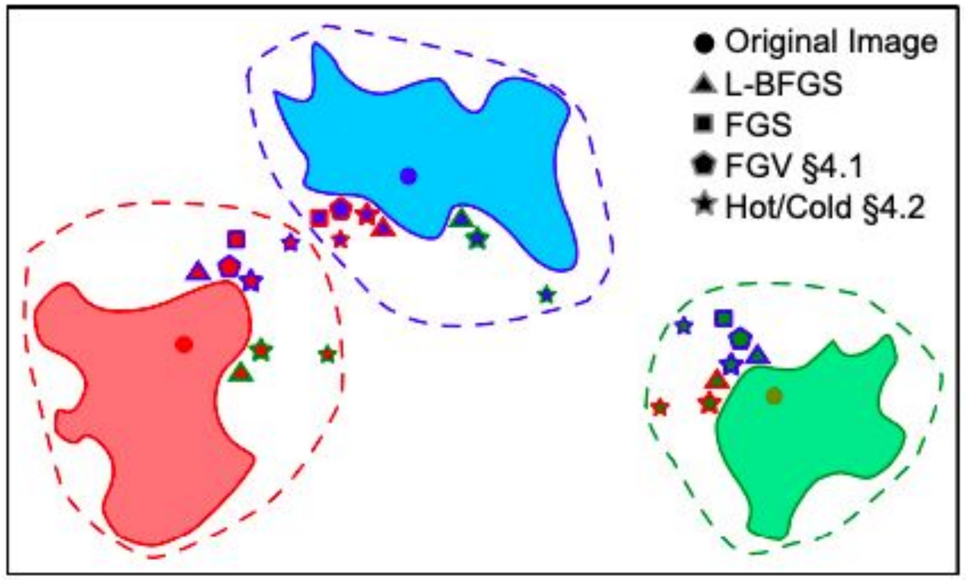
Fast Gradient Value (FGV) Method

Adversarial Diversity and Hard Positive Generation

Andras Rozsa, Ethan M. Rudd, and Terrance E. Boult*
University of Colorado at Colorado Springs
Vision and Security Technology (VAST) Lab
{arozsa,erudd,tboult}@vast.uccs.edu

Abstract

State-of-the-art deep neural networks suffer from a fundamental problem – they misclassify adversarial examples formed by applying small perturbations to inputs. In this paper, we present a new psychometric perceptual adversarial similarity score (PASS) measure for quantifying adversarial images, introduce the notion of hard positive generation, and use a diverse set of adversarial perturbations – not just the closest ones – for data augmentation. We introduce a novel hot/cold approach for adversarial example generation, which provides multiple possible adversarial pertur-



https://arxiv.org/abs/1605.01775

Fast Gradient Value (FGV) Method

- perturbation: use gradient ∇ of the loss J wrt the input image $\boldsymbol{x}$, aiming to maximise that loss

- $\boldsymbol{x}' = \boldsymbol{x} + \delta = \boldsymbol{x} + \varepsilon * \mathrm{sign}(\nabla_{\boldsymbol{x}} J(\theta, \boldsymbol{x}, y))$ ↔ FGS

- $\boldsymbol{x}' = \boldsymbol{x} + \delta = \boldsymbol{x} + \varepsilon * \nabla_{\boldsymbol{x}} J(\theta, \boldsymbol{x}, y)$ ↔ FGV

Recent Adversarial Attacks on AI

- https://proceedings.neurips.cc/paper_files/paper/2023/hash/a97b58c4f7551053b0512f92244b0810-Abstract-Conference.html?
- https://ojs.aaai.org/index.php/AAAI/article/view/26739
- https://arxiv.org/abs/2403.09766
- https://arxiv.org/abs/2402.09132
- https://arxiv.org/abs/2402.15911
- https://openaccess.thecvf.com/content/CVPR2024/html/Li_One_Prompt_Word_is_Enough_to_Boost_Adversarial_Robustness_for_CVPR_2024_paper.html
- https://arxiv.org/abs/2305.14950
- https://arxiv.org/abs/2406.04031
- https://proceedings.mlr.press/v239/schwinn23a.html