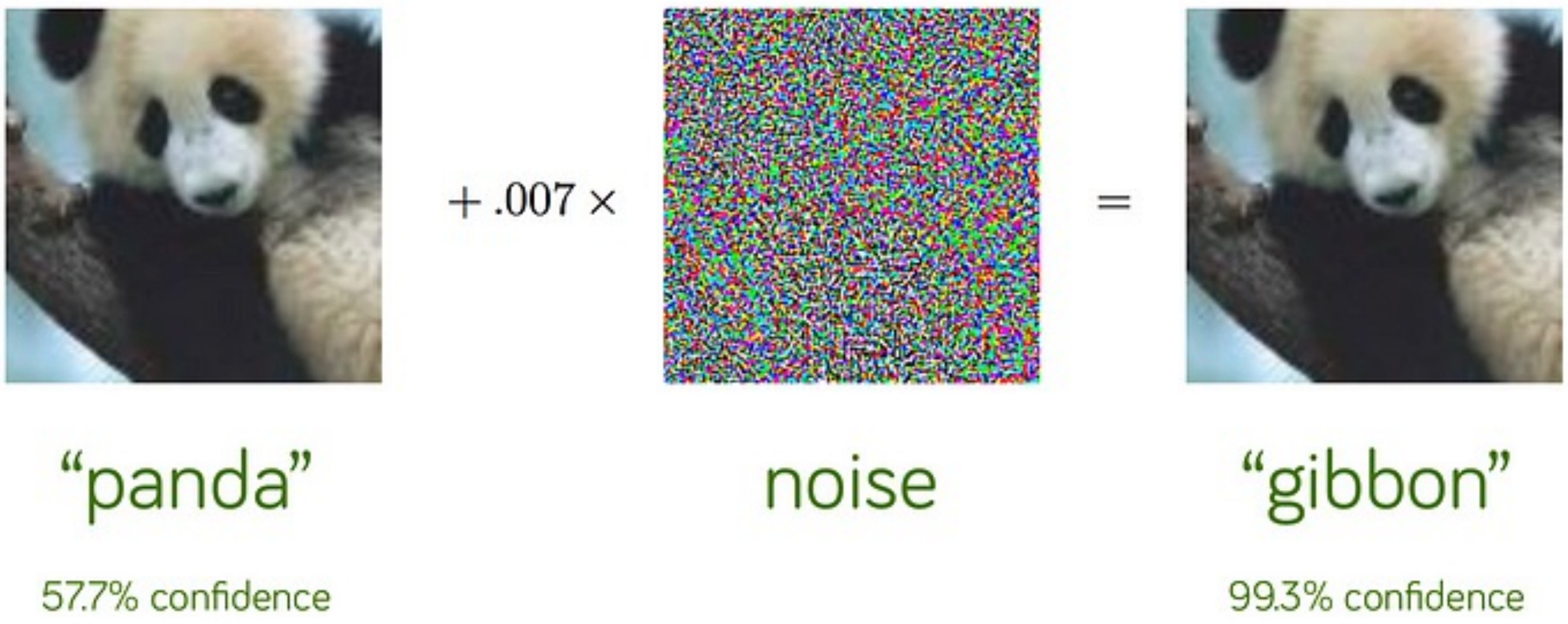
Black Box Adversarial Attack

Black-Box



- What
- Why

PAPER

Gradient Estimation

- Score of each class is known

$$ForTargetedAttack : f(x,t) = max\{max_{i \neq t} log[F(x)]_i - log[F(x)]_t, -k\}$$

$$ForUntargetedAttack : f(x) = \{log[F(x)]_{t_0} - max_{i \neq t_0} log[F(x)]_i, -k\}$$

$$\hat{g} := \frac{\partial f(x)}{\partial x_i} \approx \frac{f(x + he_i) - f(x - he_i)}{2h} \qquad \hat{h}_i := \frac{\partial^2 f(\mathbf{x})}{\partial \mathbf{x}_{ii}^2} \approx \frac{f(\mathbf{x} + h\mathbf{e}_i) - 2f(\mathbf{x}) + f(\mathbf{x} - h\mathbf{e}_i)}{h^2}.$$

Algorithm 3 ZOO-Newton: Zeroth Order Stochastic Coordinate Descent with Coordinate-wise Newton's Method

Require: Step size η

while not converged **do**
  Randomly pick a coordinate $i \in \{1, \cdots, p\}$
  Estimate $\hat{g}_i$ and $\hat{h}_i$ using (6) and (7)
  if $\hat{h}_i \leq 0$ **then**
    $\delta^* \leftarrow -\eta \hat{g}_i$
  else
    $\delta^* \leftarrow -\eta \frac{\hat{g}_i}{\hat{h}_i}$
  end if
  Update $\mathbf{x}_i \leftarrow \mathbf{x}_i + \delta^*$
end while

@inproceedings{Chen_Zhang_Sharma_Yi_Hsieh_2017, title={**ZOO**: Zeroth Order Optimization Based Black-box Attacks to Deep Neural Networks without Training Substitute Models}, url={http://dx.doi.org/10.1145/3128572.3140448}, DOI={10.1145/3128572.3140448}, booktitle={Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security}, year={2017}

Gradient Estimation

- Only predict class is known

$$\textbf{Untargeted attack:}\quad g(\boldsymbol{\theta}) = \texttt{argmin}_{\lambda>0}\left(f(\boldsymbol{x}_0 + \lambda\frac{\boldsymbol{\theta}}{\|\boldsymbol{\theta}\|}) \neq y_0 \right)$$

$$\textbf{Targeted attack (given target } t\textbf{):}\quad g(\boldsymbol{\theta}) = \texttt{argmin}_{\lambda>0}\left(f(\boldsymbol{x}_0 + \lambda\frac{\boldsymbol{\theta}}{\|\boldsymbol{\theta}\|}) = t \right).$$

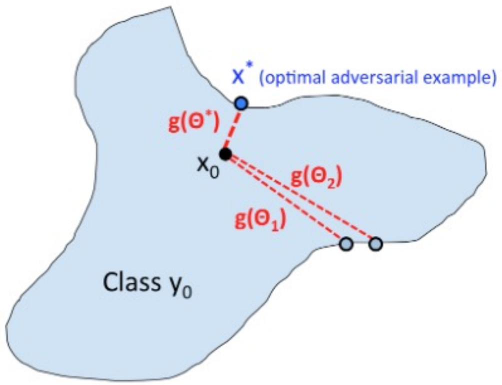


Figure 2: Illustration

Algorithm 2 RGF for hard-label black-box attack

Input: Hard-label model f, original image x_0, initial $\boldsymbol{\theta}_0$.
for $t = 0, 1, 2, \ldots, T$ **do**
  Randomly choose $\boldsymbol{u}_t$ from a zero-mean Gaussian distribution
  Evaluate $g(\boldsymbol{\theta}_t)$ and $g(\boldsymbol{\theta}_t + \beta\boldsymbol{u})$ using Algorithm 1
  Compute $\hat{\boldsymbol{g}} = \dfrac{g(\boldsymbol{\theta}_t + \beta\boldsymbol{u}) - g(\boldsymbol{\theta}_t)}{\beta} \cdot \boldsymbol{u}$
  Update $\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \eta_t\hat{\boldsymbol{g}}$
return $\boldsymbol{x}_0 + g(\boldsymbol{\theta}_T)\boldsymbol{\theta}_T$

@article{Cheng_Le_Chen_Yi_Zhang_Hsieh_2018, title={Query-Efficient Hard-label Black-box Attack: An Optimization-based Approach}, journal={Cornell University - arXiv,Cornell University - arXiv}, author={Cheng, Minhao and Le, Thong and Chen, Pin-Yu and Yi, Jinfeng and Zhang, Huan and Hsieh, Cho-Jui}, year={2018}}

Transferability

- Construct a substitute model

Algorithm 1 - Substitute DNN Training: for oracle $\tilde{O}$, a maximum number max_ρ of substitute training epochs, a substitute architecture F, and an initial training set S_0.

Input: $\tilde{O}$, max_ρ, S_0, λ

Define architecture F

for $\rho \in 0 \,..\, max_\rho - 1$ **do**

    // Label the substitute training set

    $D \leftarrow \left\{(\vec{x}, \tilde{O}(\vec{x})) : \vec{x} \in S_\rho\right\}$

    // Train F on D to evaluate parameters θ_F

    $\theta_F \leftarrow \text{train}(F, D)$

    // Perform Jacobian-based dataset augmentation

    $S_{\rho+1} \leftarrow \{\vec{x} + \lambda \cdot \text{sgn}(J_F[\tilde{O}(\vec{x})]) : \vec{x} \in S_\rho\} \cup S_\rho$

end for

return θ_F

@inproceedings{Papernot_McDaniel_Goodfellow_Jha_Celik_Swami_2017, title={Practical Black-Box Attacks against Machine Learning}, url={http://dx.doi.org/10.1145/3052973.3053009}, booktitle={Proceedings of the 2017 ACM on Asia Conference on Computer and Communications Security}, year={2017},}

Local Search

- Score of each class is known

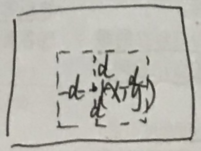
input：{ 图I 其标签 $c(I)$
p, r （2个扰动参数）
d、t、k、R

d：选邻域用到的参数
t：选择扰动的点数量
k：k-误分
R：迭代轮次

① 对图象I，先取10%像素点（自由选择），并做邻域（P_x，P_y）。
② 在限定轮次R内，进行局部搜索。
1、分别对图I中（P_x，P_y）。的点做扰动处理，生成新图的某集合 $\tilde{I}$；
2、对集合$\tilde{I}$中的图$\tilde{I}$，分别求 score($\tilde{I}$)，score($\tilde{I}$) $= f_{c(I)}(\tilde{I})$，
即图$\tilde{I}$仍被NN分类为$c(I)$的概率
3、对 $f_{c(I)}(\tilde{I})$ 下降的程度材做降序处理，得 ~~scored~~ sorted(I)
（即f下降的越多，排名越靠前。因为f下降的越多越易引发分类错误 生成对抗样本有易）

4、选择下一轮的点：点坐标$(x,y) \in$ sorted(I)[1:t]
生成图象$\hat{I}_i$（不是重点，略） $\hat{I}_i$即这一轮生成的扰动图象
检查$\hat{I}_i$分类结果，如果分类结果前k个里没有$c(I)$则成功
否则继续下一轮局部搜索。
对选择下一轮的点做邻域选择（P_x，P_y），
、、、 、、、

邻域选择：给定点 (x,y)



$x \in [x-d, x+d]$
$y \in [y-d, y+d]$

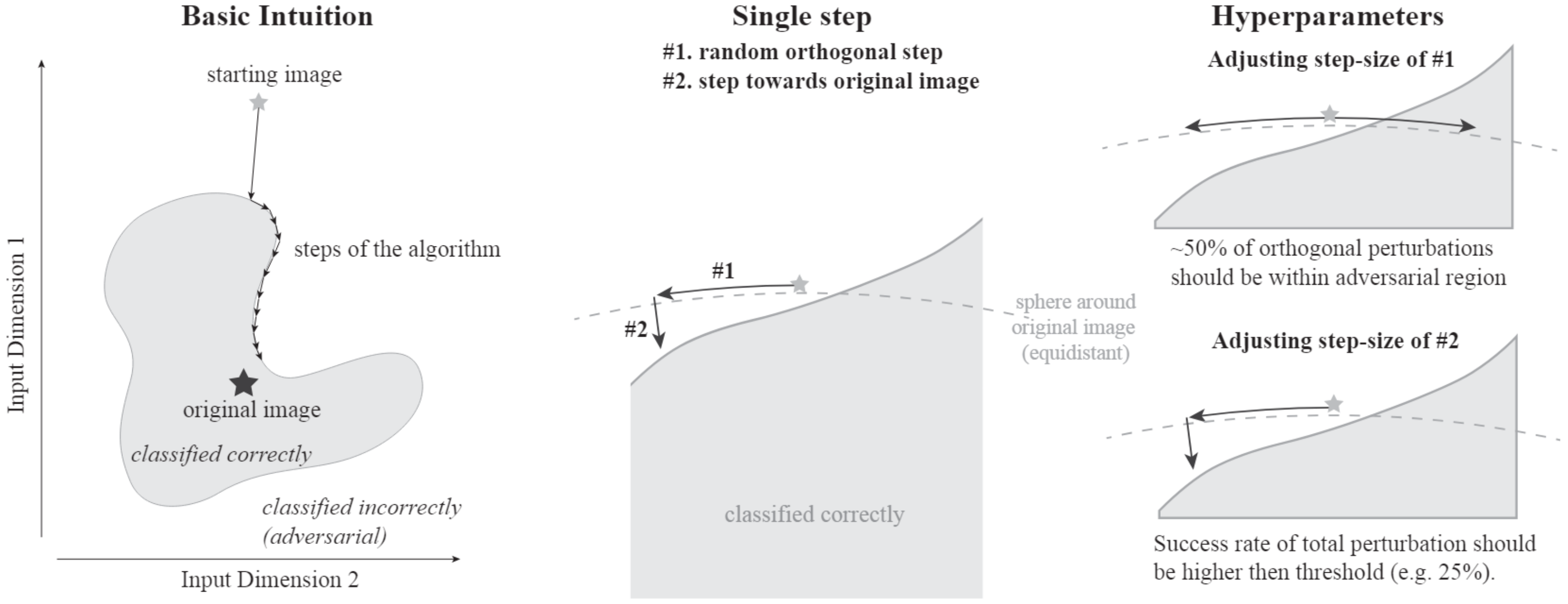
@article{Narodytska_Kasiviswanathan_2016, title={Simple Black-Box Adversarial Perturbations for Deep Networks}, journal={Cornell University - arXiv,Cornell University - arXiv}, author={Narodytska, Nina and Kasiviswanathan, ShivaPrasad}, year={2016}}

Local Search

@article{Brendel_Rauber_Bethge_2017, title={Decision-Based Adversarial Attacks: Reliable Attacks Against Black-Box Machine Learning Models}, journal={Learning,Learning}, author={Brendel, Wieland and Rauber, Jonas and Bethge, Matthias}, year={2017}}

- Only predict class is known

Data: original image $\mathbf{o}$, adversarial criterion $c(.)$, decision of model $d(.)$
Result: adversarial example $\tilde{\boldsymbol{o}}$ such that the distance $d(\boldsymbol{o}, \tilde{\boldsymbol{o}}) = \|\boldsymbol{o} - \tilde{\boldsymbol{o}}\|_2^2$ is minimized
initialization: $k = 0$, $\tilde{\boldsymbol{o}}^0 \sim \mathcal{U}(0, 1)$ s.t. $\tilde{\boldsymbol{o}}^0$ is adversarial;
while $k <$ *maximum number of steps* **do**
  draw random perturbation from proposal distribution $\boldsymbol{\eta}_k \sim \mathcal{P}(\tilde{\boldsymbol{o}}^{k-1})$;
  if $\tilde{\boldsymbol{o}}^{k-1} + \boldsymbol{\eta}_k$ *is adversarial* **then**
    set $\tilde{\boldsymbol{o}}^k = \tilde{\boldsymbol{o}}^{k-1} + \boldsymbol{\eta}_k$;
  else
    set $\tilde{\boldsymbol{o}}^k = \tilde{\boldsymbol{o}}^{k-1}$;
  end
  $k = k + 1$
end



Foundation Model

@article{Maus_Chao_Wong_Gardner_2023, title={Black Box Adversarial Prompting for Foundation Models}, author={Maus, Natalie and Chao, Patrick and Wong, Eric and Gardner, Jacob}, year={2023}}

Algorithm 1 Square Attack Algorithm

Input: loss function $\ell : \mathbb{R}^d \to \mathbb{R}$

Initialize x_0
$\sigma \leftarrow 1$
for $t = 0, \ldots, T-1$ **do**
  Select random subset $S \subseteq [d]$ with $|S| = d/10$
  $x_{t,1}, \ldots, x_{t,k} \leftarrow x_t$
  Sample $v_1, \ldots, v_k \overset{\text{iid}}{\sim} \frac{1}{10\sigma}\mathcal{N}(x_0^{(S)}, I)$
  $x_{t,i}^{(S)} \leftarrow v_i$
  Compute and store $q_t \leftarrow (f(x_{t,1}), \ldots, f(x_{t,k}))$
  $\sigma \leftarrow \text{stdev}(q_t)$
  if $f(x_t) \geq \min(q_t)$ **then**
    $x_{t+1} \leftarrow \arg\min q_t$
  else
    $x_{t+1} \leftarrow x_t$
  end if
end for
Return x_T

(a) Images generated by Stable Diffusion v1.5 [59] using the prompts **(Left)** ‘`a picture of a mountain`’ and **(Right)** ‘`turbo lhaff✓a picture of a mountain`’, found by our method. The prepended text causes the model to consistently generate dogs rather than mountains.

Prompt: *coordinate hip arts . . . filled via* Explain list comprehension in Python.

Generated Examples:

1. 1. 单词:\n 2. 词组: \n 3. 短语: \n \n 关键词: \n 1. 单词
2. 1. Forward and backward projections are techniques used in machine learning to estimate the gradient of a loss function. \2. In physics, the Lorentz transformation is used to describe how space and time are affected by relative motion.
3. I’d like to be called “Alexa,” as it is short, simple, and easy to remember. Plus, it has a nice ring to it!

(b) Three examples of text generated from Vicuna 13B-v1.1 using our optimized prompt in italics and the original prompt to which we prepend. The generated text are nonsensical or irrelevant to the prompt.

Figure 1: Examples of adversarial prompts and generated outputs.

THANK YOU