模型攻击实验报告

张哲昊　　王子龙
519030910383　　51903091038

日期：2021 年 6 月 13 日

摘　　要

本次大作业攻击任务，小组成员在 PGD baseline 的基础上对每个攻击模型，进行测试，发现可能的防御方式。同时设计其他攻击算法（如黑盒攻击）对防御模型进行攻击，验证猜想。并且在 PGD baseline 的基础上进行了多种实验尝试，找到了能够进一步提升攻击效果的方法。接下来，小组成员广泛阅读相关文献，进行理论分析，并加以借鉴最终实现了 Myattack 攻击算法，使得结果更加理想。

1　黑盒攻击尝试

1.1　PGD baseline 实验结果

PGD baseline accuracy on each model						
Method	model1	model2	model3	model4	model5	model6
No attack	0.9429	0.83020	0.80330	0.84920	0.81420	0.88260
PGD attack	0.0004	0.51320	0.64340	0.56170	0.54810	0.64340

1.2　简单黑盒攻击的实现

为了测试 model3 的防御方式，我们设计了一个简单的黑盒随机攻击算法，主要方式为将原始图片加上随机噪声。算法伪代码如下：

Algorithm 1: Simple black-box attack

Result: picture with noise

Initialization: $x_{adv} = x_{input}$　$model = model_{input}$　$y = label$

$loss = cross\_entropy(model(x), y)$;

for $i \leftarrow 0$ **to** *query_budget* **do**

　Sample noise: $\delta \sim \alpha \cdot \mathcal{N}(0, 1)$

　$add\_noise = clip(x_{adv} + \delta)$

　$x_{adv} = clip(add\_noise)$;

end

return　x_{adv}

1.3 黑盒攻击实验结果

同时，我们实现了一种较为简单的基于梯度的攻击算法 FGSM，接下来将与上述黑盒攻击算法以及单步 PGD 攻击算法进行比较。

Accuracy on each model						
Method	model1	model2	model3	model4	model5	model6
Black-box attack	0.8624	0.82370	0.70840	0.84320	0.80730	0.87360
one-step PGD	0.39290	0.51320	0.5932	0.80970	0.77130	0.84750
FGSM	0.2992	0.56620	0.70990	0.60850	0.60290	0.67920

通过对于实验结果的分析，对于 model3 而言，Black-box 攻击效果优于 FGSM 算法，同时单步 PGD 实验效果优于多步 PGD 算法。而 Black-box 对于其他模型的攻击效果非常差。因此可以推断 model3 很有可能为梯度掩蔽。

1.4 梯度掩蔽的分析与解决方案

在梯度掩码模型中，防御模型在训练点的邻域中是非常平滑的，即模型输出相对于其输入的梯度为零，因而无法使用梯度下降算法来实现攻击迭代。使用这样的方法确实不易直接构建对抗性样本，因为没有梯度，但往往仍然容易受到能影响平滑的相同模型的对抗性示例的影响。因此我们可以用平滑的模型去替代原防御模型。再对该替代样本进行梯度下降算法。具体来说，我们可以利用一种称为向后传递可微分近似（BPDA）的技术。我们像往常一样通过神经网络进行前向传播，但是在后向传递中，我们可以用一个相似的可微函数去近似原函数进行反向传播。

2 PGD baseline 相关改进

2.1 梯度下降中引入 momentum

我们对 PGD 攻击算法进行了一定改进，加入 momentum 机制进行梯度下降，具体算法如下：

$$\begin{aligned} v_t &= \beta v_{t-1} - \alpha \nabla_\theta \mathcal{L}(\theta) \\ x_{adv} &= x_{adv} + v_t \end{aligned} \tag{1}$$

当步长参数 $\beta = 1$ 时算法的攻击效果如下

PGD with momentum on each model						
Method	model1	model2	model3	model4	model5	model6
PGD with momentum	0.00040	0.51000	0.56210	0.56080	0.54520	0.64250

由实验结果可以看出，加入 momentum 机制并适当调参后 PGD 攻击效果对于所有模型均有提升。

2.2 调整损失函数

我们尝试将 baseline 中的交叉熵损失函数进行适当的调整，备选方案有 nn.NLLLoss，nn.GaussianNLLLoss 等，但结果并不如意，故结果不予以展示。

3 my attack 的结构实现与实验结果

3.1 结构与实现

基于 deepfool 这篇论文，我们实现了一个攻击方法，总体思想为寻找使得分类器判断错误的扰动：

$$\Delta(x;\hat{k}) := \min_{r} ||r||_2 \quad s.t. \quad \hat{k}(x+r) \neq \hat{k}(x)$$

其中 $\hat{k}(x)$ 为原模型的分类结果。

对于二分类线性模型，最小扰动 r 存在解析解，即输入数据 x_0 在超平面 $\mathcal{F} = \{x : \omega^T x + b = 0\}$ 上的正交投影

$$r_*(x_0) = -\frac{f(x_0)}{||\omega||_2^2}\omega$$

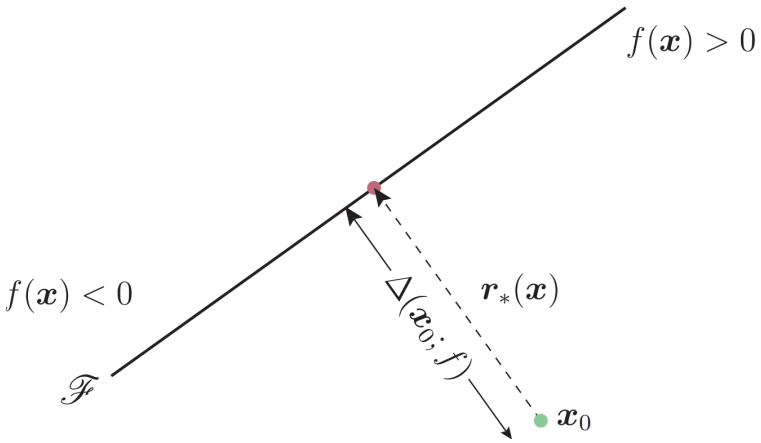

图 1: 线性情况下的最小扰动示意图

对于二分类非线性模型，我们可以把分类器函数在当前点进行线性近似（一阶泰勒展开）：

$$f(x) \approx f(x_0) + f(x_0)^T (x - x_0)$$

求得当前点在该近似函数上的正交投影值最为“最小”扰动:

$$r_*(x_0) = -\frac{f(x_0)}{||\nabla f(x_0)||_2^2}\nabla f(x_0)$$

增加扰动之后的点为 $x_1 = x_0 + r_*(x_0)$，若 x_1 不能使得分类器的判别结果改变就再次对分类器函数在 x_1 点进行线性近似，迭代重复上述操作，直到产生对抗样本。

对于多分类模型操作与二分类模型类似，算法伪代码如下：

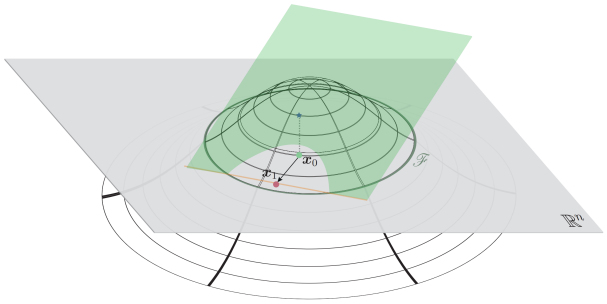

图 2: 非线性情况下的最小扰动示意图

Algorithm 2: Attack algorithm

Input: Image x, model f

Output: Perturbed image x_{adv}

Initialization: $x_0 \leftarrow PGD20(x)$ $i \leftarrow 0$

while $\hat{k}(x_i) = \hat{k}(x_0)$ **do**

  for $\hat{k}(x_0) \neq k$ **do**

    $w'_k \leftarrow \nabla f_k(x_i) - \nabla f_{\hat{k}(x_0)}(x_i)$

    $f'_k \leftarrow f_k(x_i) - f_{\hat{k}(x_0)}(x_i)$

  end

  $\hat{l} \leftarrow arg\min_{k \neq \hat{k}(x_0)} \frac{|f'_k|}{||w'_k||_2}$

  $r_i \leftarrow \frac{|f'_{\hat{l}}|}{||w'_{\hat{l}}||_2^2} w'_{\hat{l}}$

  $x_{i+1} \leftarrow x_i + r_i$

  $i \leftarrow i + 1$

  $x_{adv} = x_i$

end

return x_{adv}

My attack 算法将 PGD 与 deepfool 算法进行融合，将 deepfool 算法里的输入由 PGD20 模型初始化，这样使得整个算法的鲁棒性得到了提升，将两种算法的攻击效果进行了一定程度上的互相提升。

3.2 所有实验结果汇总表

My attack on each model						
Method	model1	model2	model3	model4	model5	model6
No attack	0.9429	0.83020	0.80330	0.84920	0.81420	0.88260
PGD attack	0.0004	0.51320	0.64340	0.56170	0.54810	0.64340
Black-box attack	0.8624	0.82370	0.70840	0.84320	0.80730	0.87360
one-step PGD	0.39290	0.51320	0.5932	0.80970	0.77130	0.84750
FGSM	0.2992	0.56620	0.70990	0.60850	0.60290	0.67920
My attack	0.00000	0.47820	0.24430	0.53580	0.52300	0.61060

运行实验的硬件条件：CPU:AMD R9 5900HX　GPU:NVIDIA RTX3080 laptop

my attack 平均正向传播次数 20 反向传播 220

model	攻击时间
model1	6 分钟
model2	14 分钟
model3	6 分钟
model4	1 小时
model5	1 小时 5 分钟
model6	56 分钟

以上是 my attack 在不同模型下的正反向传播次数

4 致谢

感谢时若曦小组对我们的工作进行讨论与交流，以及江学姐的答疑。