Beijing Forest Studio
北京理工大学信息系统及安全对抗实验中心

模型窃取

博士研究生 鲁川

2021年05月9日

内容提要

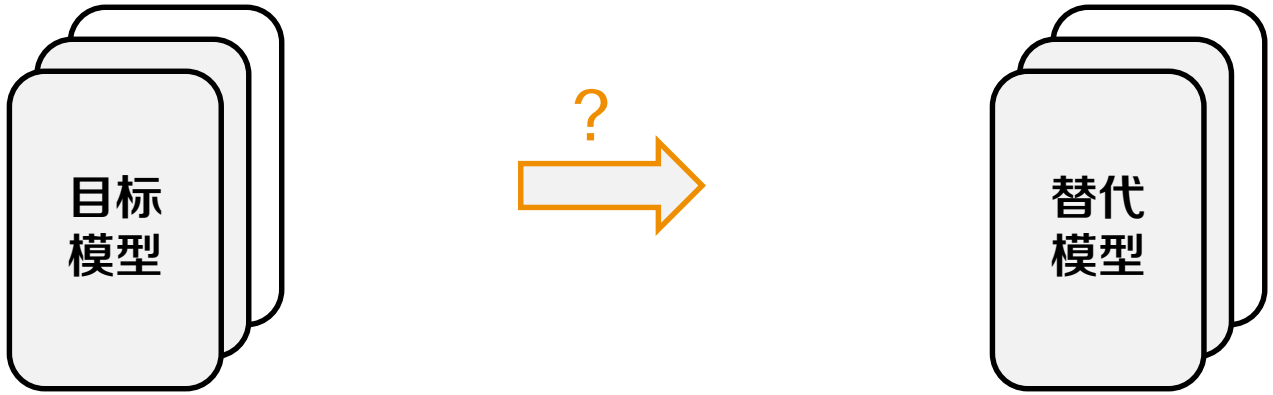
- 背景简介
- 基本概念
- 算法原理
- 优劣分析
- 应用总结
- 参考文献

背景简介

- 预期收获
  - 1. 了解模型窃取的主要方法分类
  - 2. 理解模型窃取两类方法的技术原理
  - 3. 了解模型窃取在网络安全领域的应用

背景简介

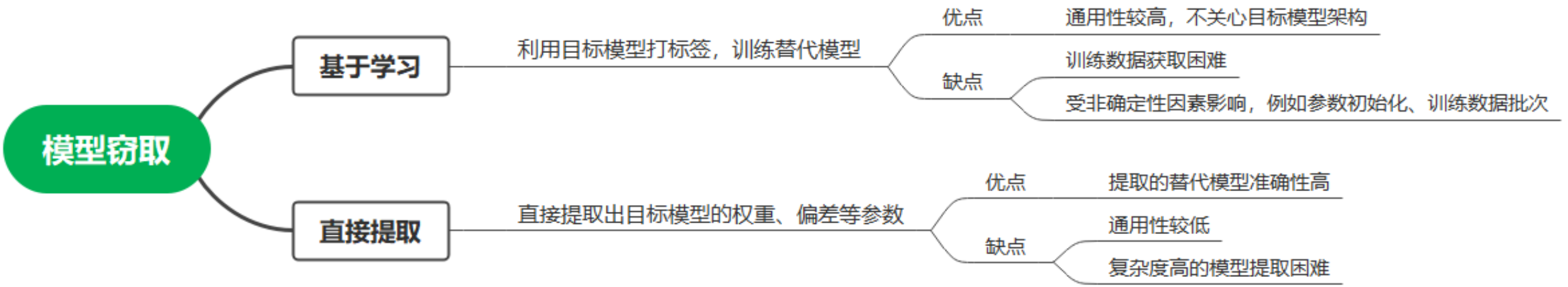
- 模型窃取提取背景
  - 许多云运营商提供了机器学习服务（MLaaS），可通过API提供服务，或者云用户提供训练好的模型来赚取利益
  - 一些通过专有数据训练的模型，如金融交易、医疗信息、用户交易信息等



基本概念

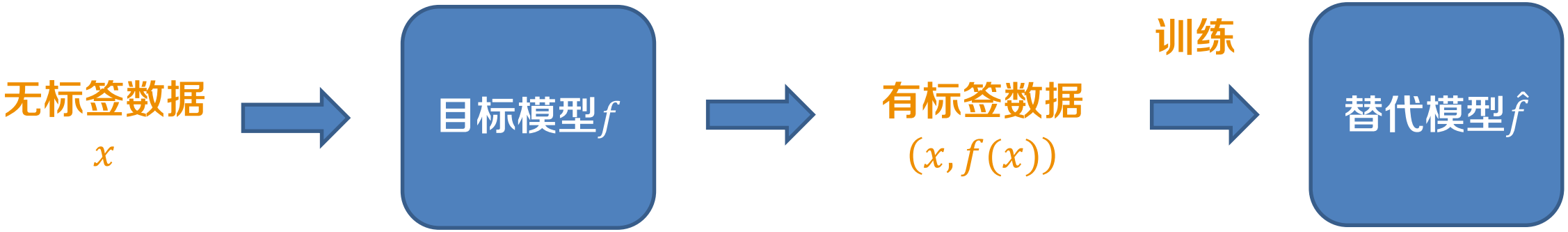
基本概念

- **模型窃取**
  - **针对目标模型，复制一个功能相似甚至完全相同的替代模型**



基本概念

- **基于学习的模型提取**



基本概念

- **直接提取**
  - **考虑逻辑回归模型**

$$f(\boldsymbol{x}) = \sigma(\boldsymbol{w} \cdot \boldsymbol{x} + \beta)$$

其中$\sigma(t) = \frac{1}{1+e^{-t}}$，$\boldsymbol{w} \in \mathbb{R}^d$，$\beta \in \mathbb{R}$

  - **因为是线性模型，可以通过**$d+1$**线性无关的输入构造公式求解**w**和**β

$$\boldsymbol{w} \cdot \boldsymbol{x} + \beta = \sigma^{-1}\big(f(\boldsymbol{x})\big)$$

$$\begin{cases} w_1 \cdot x_1 + \beta = \sigma^{-1}\big(f(x_1)\big) \\ w_2 \cdot x_2 + \beta = \sigma^{-1}\big(f(x_2)\big) \\ \cdots\cdots \\ w_d \cdot x_d + \beta = \sigma^{-1}\big(f(x_d)\big) \end{cases} \Rightarrow \begin{cases} w_1 \\ w_2 \\ \cdots\cdots \\ w_d \\ \beta \end{cases}$$

基本概念

- **激活函数为$ReLU$的单隐层神经网络**

$$O_L(x) = A^{(1)}ReLU\left(A^{(0)}x + B^{(0)}\right) + B^{(1)}$$

  - **如果$A^{(0)}$、 $B^{(0)}$放大c倍， $A^{(1)}$缩小c^{-1}倍，相同输入下， $O_L(x)$不变，即**

$$O'_L(x) = c^{-1} \cdot A^{(1)}ReLU\left(c \cdot \left(A^{(0)}x + B^{(0)}\right)\right) + B^{(1)} = O_L(x)$$

  - **若$A^{(0)}x + B^{(0)} \leq 0$，则**

$$O'_L(x) = c^{-1} \cdot A^{(1)} \cdot 0 + B^{(1)} = O_L(x) = B^{(1)}$$

  - **若$A^{(0)}x + B^{(0)} > 0$，则**

$$O'_L(x) = c^{-1} \cdot A^{(1)} \cdot c \cdot \left(A^{(0)}x + B^{(0)}\right) + B^{(1)} = O_L(x)$$

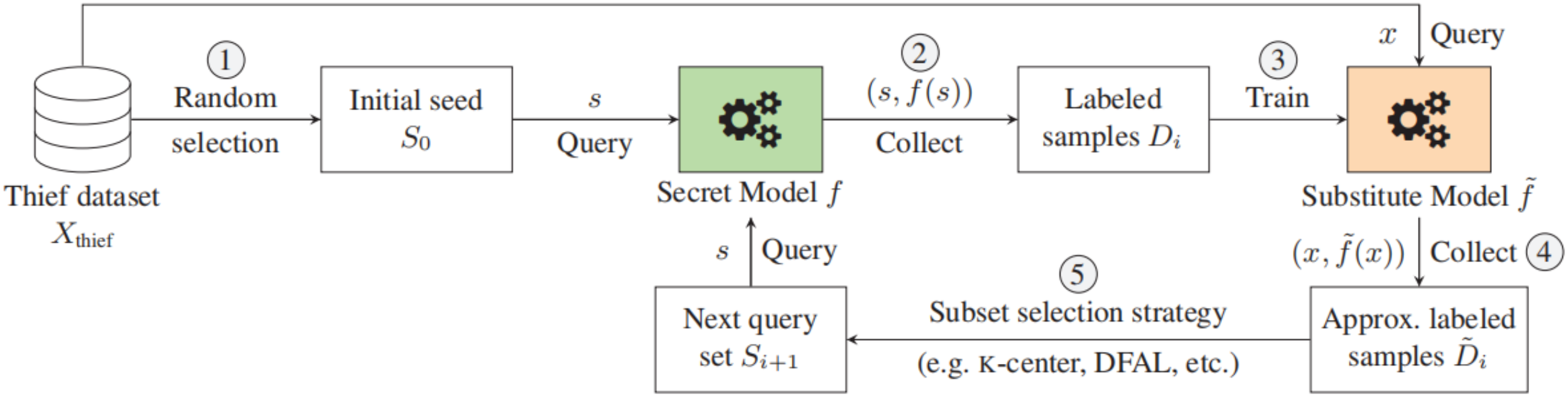
算法原理

算法原理

T	在无目标模型训练集的情况下进行模型窃取
I	无标签公共数据集、目标模型
P	1.在无标签公共数据集选取一组数据集 2.利用目标模型f打标签，并且训练替代模型$\tilde{f}$ 3.根据选择策略，选取下一次迭代的数据集
O	替代模型

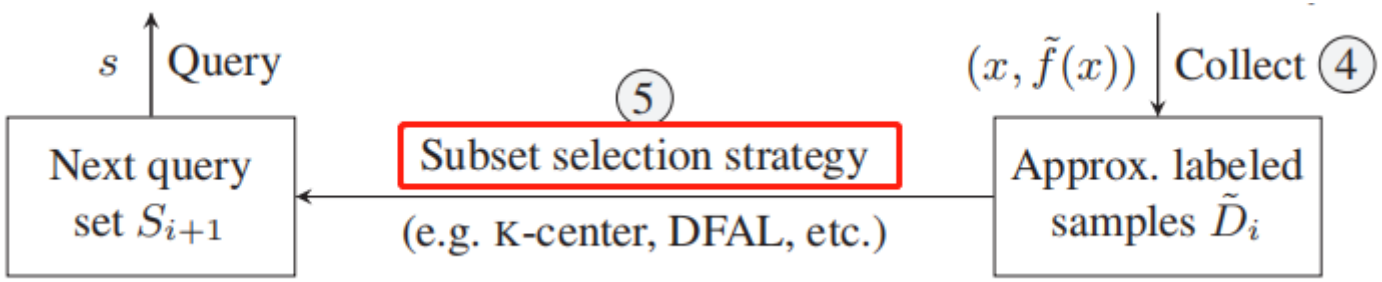
P	无目标模型训练集
C	可以自由访问目标模型
D	每一次迭代训练集的选择策略
L	AAAI 2020

算法框架



X_{thief}：公共数据集　　f：攻击模型　　$\tilde{f}$：替代模型

选择策略



- **Random strategy**
  - **随机选择下一组训练数据**
- **Uncertainty strategy（Lewis and Gale 1994）**
  - **根据公式选择$\mathcal{H}_n$最大的**

$$\mathcal{H}_n = -\sum_j \tilde{y}_{n,j} log \tilde{y}_{n,j}$$

  - **$\tilde{y}_{n,j}$：替代模型$\tilde{f}$的预测概率向量**
  - **$\tilde{y}_{n,j}$越大，$\mathcal{H}_n$越小，代表替代模型$\tilde{f}$对该组训练集的预测效果越好**

选择策略

- **K-center strategy（Sener and Savarese 2018）**

$$(x_0^*, \tilde{y}_0^*) = arg \max_{(x_n,\tilde{y}_n)\in\tilde{D}_i} \min_{(x_m,y_m)\in D_{i-1}} \left\|\tilde{y}_n - \tilde{f}(x_m)\right\|_2^2$$

  - D_{i-1}：替代模型$\tilde{f}$上一轮使用过的数据集　$\tilde{D}_i$：未使用过的数据
  - 找出和上一轮最不相似的数据集
- **DFAL strategy：DeepFool-based Active Learning（Ducoffe and Precioso 2018）**
  - 使替代模型$\tilde{f}$错误分类$(\tilde{f}(x_n) \neq \tilde{f}(\hat{x}_n))$，且扰动$\alpha_n$最小的数据集

$$\alpha_n = \left\|x_n - \tilde{f}(\hat{x}_n)\right\|_2^2$$

  - x_n：扰动前的数据集 $\hat{x}_n$：扰动后的数据集
  - 找出最靠近决策边界的数据集，然后使用扰动前的数据
- **DAFL+K-center strategy**

评估标准

- **评估标准**

$$Agreement(f, \tilde{f}) = \frac{1}{|X_{secert}^{test}|} \sum_{x \in X_{secert}^{test}} I\left(f(x) = \tilde{f}(x)\right)$$

  - X_{secert}^{test}：测试数据集
- **测试集**
  - 图像：MNIST、CIFAR-10、GTSRB
  - 文本：MR、IMDB、AG News
- **训练集**
  - 图像：ILSVRC2012-14(100K/20K)
  - 文本：WikiText-2(80K/9K)

实验结果

- 同构模型
  - DFAL+K-center效果最好，并且训练预算越多，效果越好

MNIST	10K (8%)	15K (12%)	20K (17%)	25K (21%)	30K (25%)
Random	91.64	95.19	95.90	97.48	97.36
Uncertainty	94.64	97.43	96.77	97.29	97.38
K-center	**95.80**	95.66	96.47	**97.81**	97.95
DFAL	95.75	95.59	96.84	97.74	97.80
DFAL+K-center	95.40	**97.64**	**97.65**	97.60	**98.18**
Using the full thief dataset (120K):					98.54
Using uniform noise samples (100K):					20.56

CIFAR-10	10K (8%)	15K (12%)	20K (17%)	25K (21%)	30K (25%)
Random	63.75	68.93	71.38	75.33	76.82
Uncertainty	63.36	69.45	72.99	74.22	76.75
K-center	**64.20**	70.95	72.97	74.71	78.26
DFAL	62.49	68.37	71.52	**77.41**	77.00
DFAL+K-center	61.52	**71.14**	**73.47**	74.23	**78.36**
Using the full thief dataset (120K):					84.99
Using uniform noise samples (100K):					10.62

MR	10K (11%)	15K (17%)	20K (22%)	25K (28%)	30K (33%)
Random	76.45	78.24	79.46	81.33	82.36
Uncertainty	**77.19**	80.39	81.24	**84.15**	83.49
K-center	77.12	**81.24**	**81.96**	83.95	**83.96**
Using the full thief dataset (89K):					86.21
Using discrete uniform noise samples (100K):					75.79

IMDB	10K (11%)	15K (17%)	20K (22%)	25K (28%)	30K (33%)
Random	71.67	78.79	74.70	80.71	79.23
Uncertainty	73.48	78.12	**81.78**	**82.10**	82.17
K-center	**77.67**	**78.96**	80.24	81.58	**82.90**
Using the full thief dataset (89K):					86.38
Using discrete uniform noise samples (100K):					53.23

实验结果-同构模型

- 迭代次数的影响
  - 相同条件下迭代次数越多，效果越好
  - 迭代越多，选择更合适的数据概率越高

Dataset	Substitute model agreement (%)	
	10 iterations	20 iterations
MNIST	95.80	**96.74**
CIFAR-10	64.20	**64.23**
GTSRB	72.71	**72.78**

- 输出类别的影响
  - 结果概率分数和只给结果的对比
  - 结果概率能够更好的帮助训练

Dataset	Substitute model agreement (%)	
	Top-1 prediction	Probability scores
MNIST	95.80	**98.61**
CIFAR-10	64.20	**77.29**
GTSRB	72.71	**86.90**

- 对抗样本可迁移性
  - Papernot（2017）：模型越相似，可迁移性越高

Dataset	Papernot	ACTIVETHIEF			
		Rand	K-cen	DFAL	DF+K
MNIST	40.28	49.77	47.75	**59.55**	53.08
CIFAR-10	82.61	**85.76**	85.26	84.25	84.30
GTSRB	84.83	93.41	93.45	**93.83**	93.34

实验结果

- 不同模型架构的影响
  - 测试目标模型和替代模型不同架构的影响
  - 异构情况下，对结果影响较小

Dataset	Secret model	Substitute model CNN	Substitute model RNN
MR	CNN	**86.21**	85.18
	RNN	84.80	**89.12**
IMDB	CNN	**86.38**	85.53
	RNN	87.06	**90.22**
AG News	CNN	90.07	**92.62**
	RNN	90.96	**93.01**

- 相同模型，不同参数
  - l：卷积块

(a) MNIST dataset

Secret model	Substitute model l = 2	l = 3	l = 4
l = 2	**98.73**	98.15	97.63
l = 3	97.21	**98.81**	98.10
l = 4	96.75	98.05	**98.36**

(b) CIFAR-10 dataset

Secret model	Substitute model l = 2	l = 3	l = 4
l = 2	**78.34**	76.83	74.48
l = 3	80.66	81.57	**81.80**
l = 4	74.34	**79.17**	78.82

(c) GTSRB dataset

Secret model	Substitute model l = 2	l = 3	l = 4
l = 2	**95.02**	92.30	86.88
l = 3	90.08	**91.42**	91.28
l = 4	80.95	**86.50**	84.69

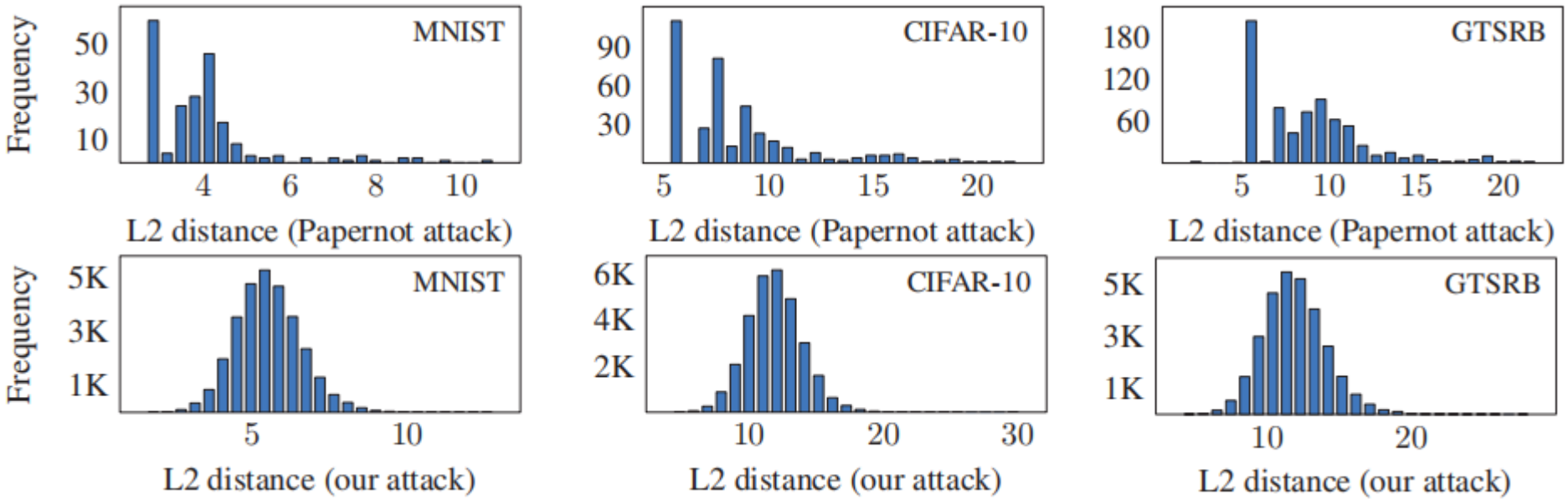
同样l的情况一般是最好的，l过小或者过大可能造成过拟合或欠拟合

实验结果

- 对抗模型窃取检测方法效果
  - PRADA（Juuti 2019）：被分类到同一类别的两次查询之间的距离d_i分布服从高斯分布，合成或者扰动后的数据会影响这个分布

$$d_i = \min_{j<i, y_j=y_i} \|x_i - x_j\|_2$$

这篇文章的方法明显更符合高斯分布，因为选取的是自然数据，且没有扰动



算法原理

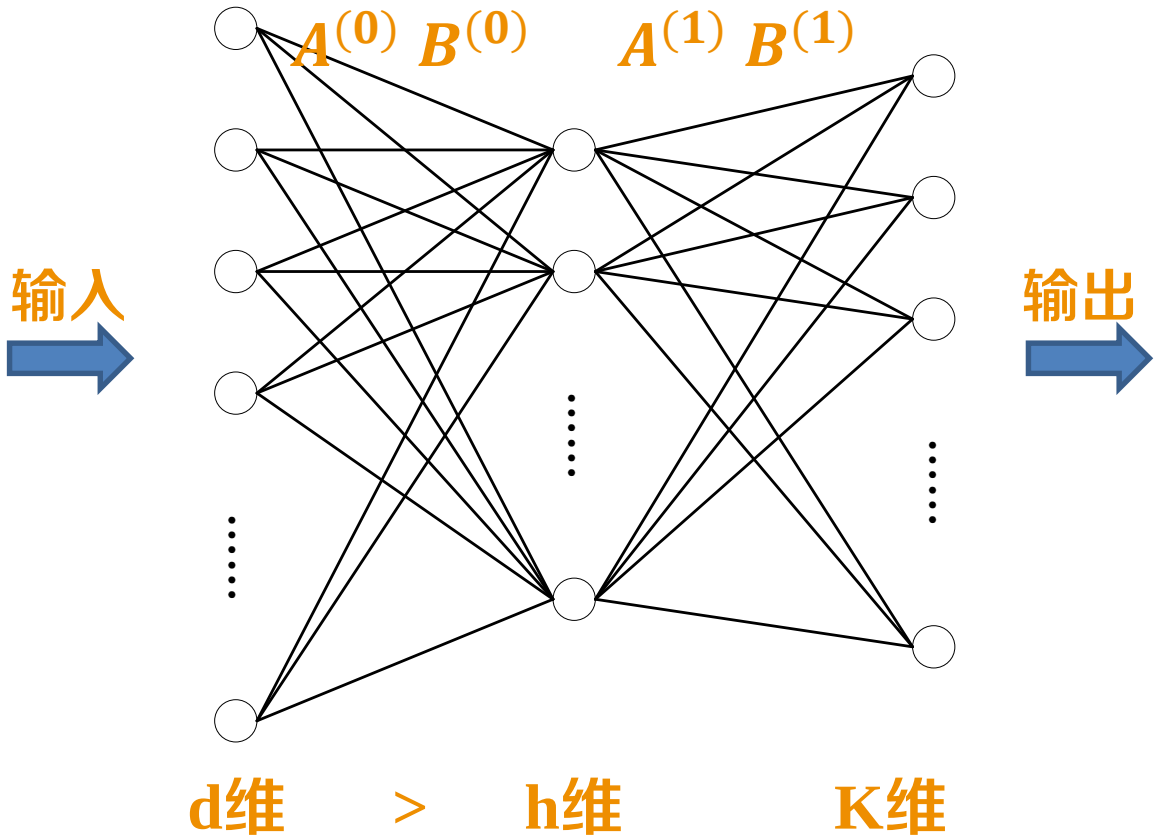
T	单隐层神经网络的模型窃取
I	目标模型
P	1.临界点寻找 2.权重绝对值计算 3.符号恢复 4.利用最小二乘法提取最后一层参数
O	替代模型

P	如何得到神经网络的权重和偏差
C	单隐层ReLU神经网络
D	参数多，线性和非线性相结合
L	USENIX Security Symposium 2020

模型架构

- logit function（假定可以获得logit输出）

$$O_L(x) = A^{(1)}ReLU\left(A^{(0)}x + B^{(0)}\right) + B^{(1)}$$



Symbol	Definition
d	Input dimensionality
h	Hidden layer dimensionality ($h < d$)
K	Number of classes
$A^{(0)} \in \mathbb{R}^{d \times h}$	Input layer weights
$B^{(0)} \in \mathbb{R}^{h}$	Input layer bias
$A^{(1)} \in \mathbb{R}^{h \times K}$	Logit layer weights
$B^{(1)} \in \mathbb{R}^{K}$	Logit layer bias

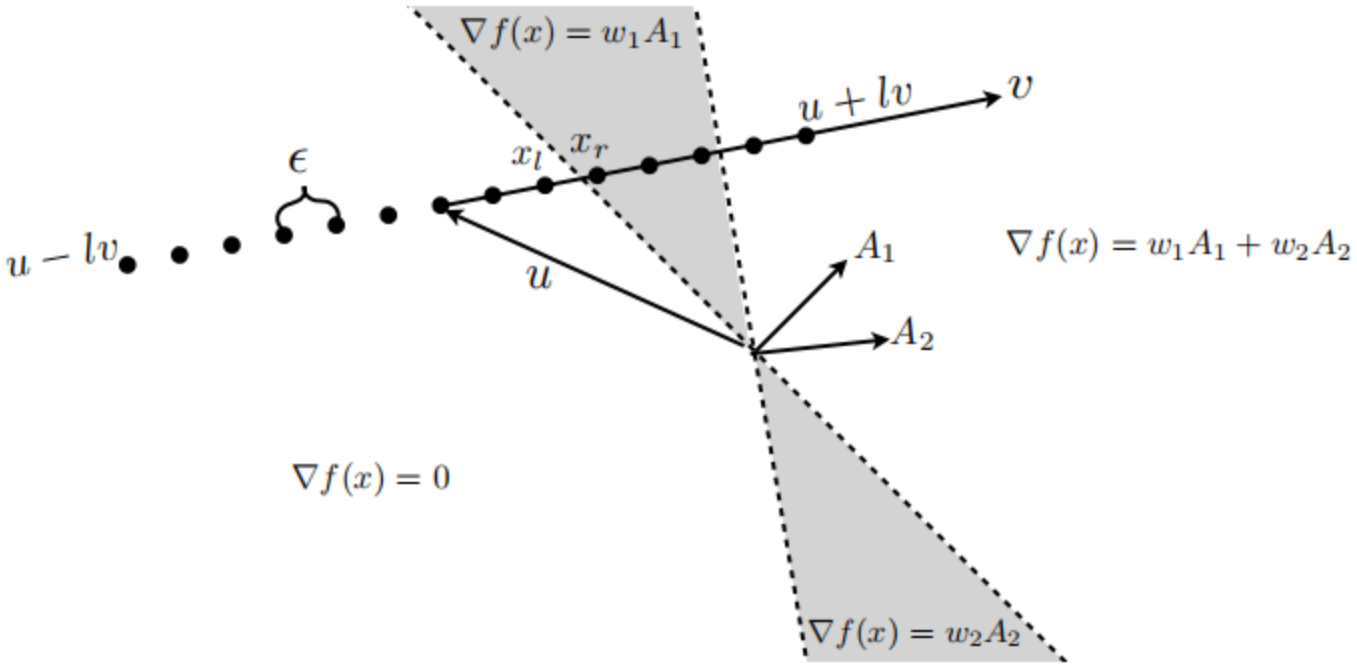
算法原理

- $ReLU$神经网络
  - 一个两层$ReLU$神经网络，忽略偏差项，可以得到

$$f(x) = \sum_{i=1}^{h} g(x)_i w_i A_i^T x$$

  - 其中，$g(x)_i = 1\{A_i^T x > 0\}$，则梯度为

$$\nabla f(x) = \sum_{i=1}^{h} g(x)_i w_i A_i$$

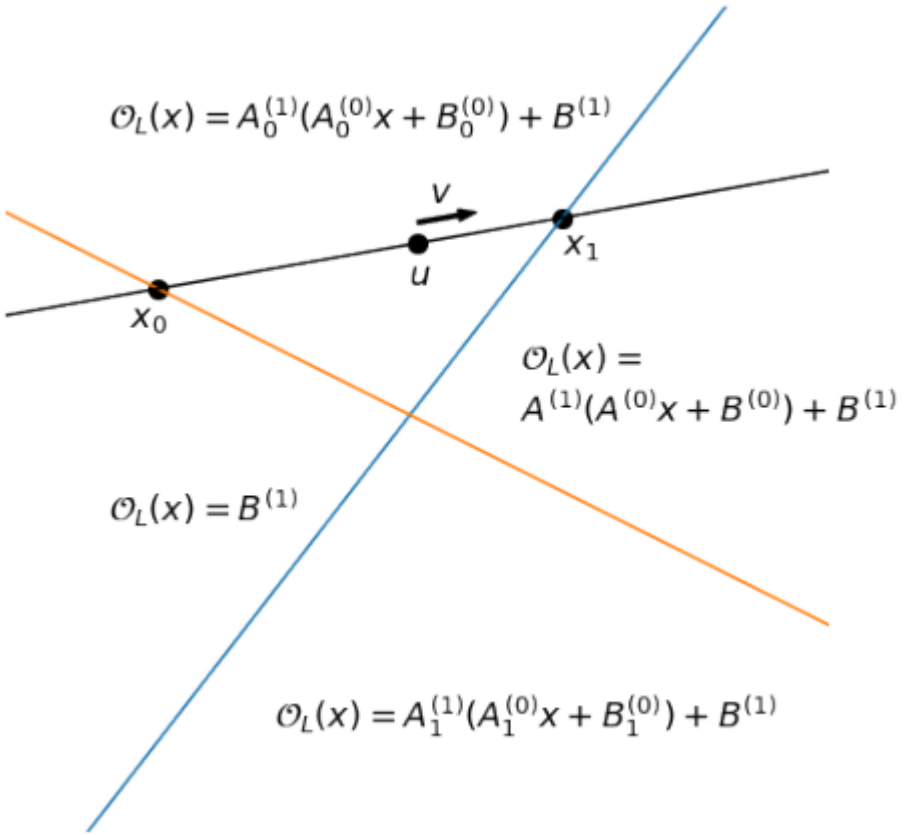


$d = h = 2$

将平面分为了四个区域，每个区域梯度相同，当跨区域时，梯度会有一个突变，例如从x_l移动到x_r

算法原理

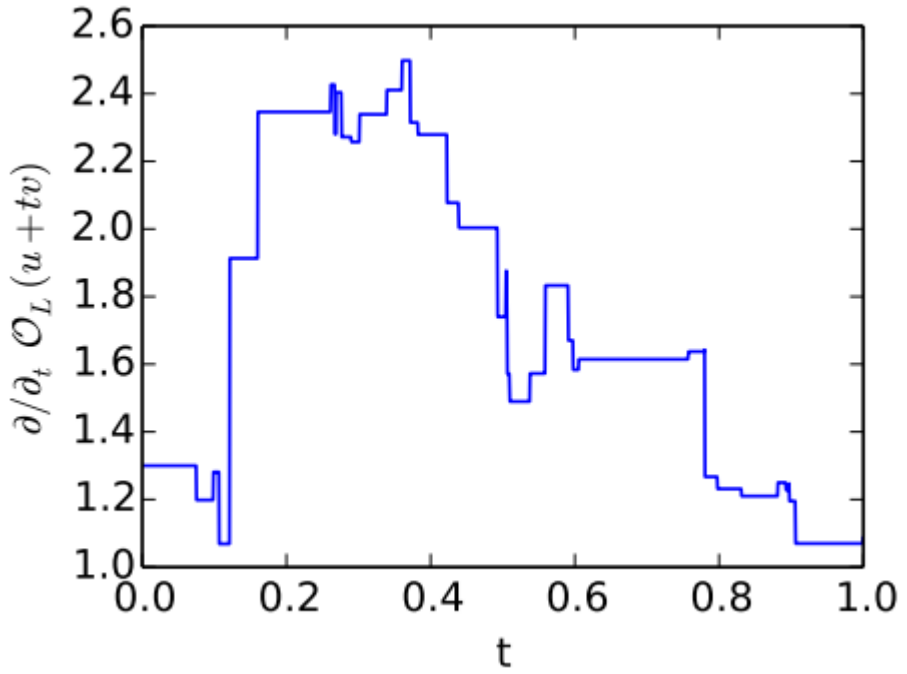
- 二维



- 多维
  - 对两个随机向量$u, v \in \mathbb{R}^d$进行采样，并考虑函数

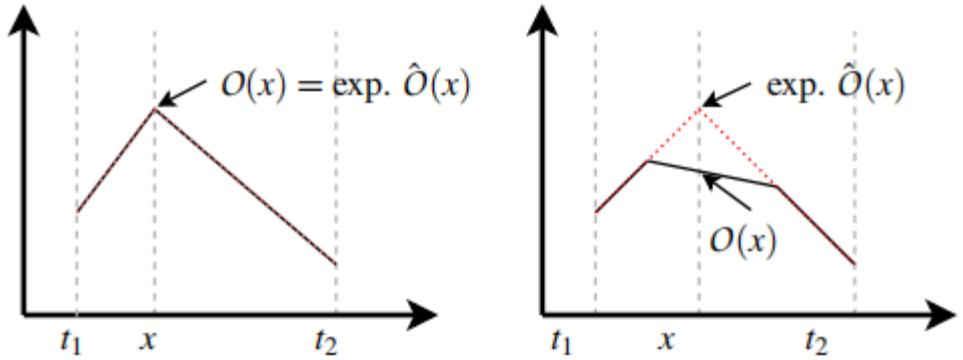
$$L(t; u, v, O_L) = O_L(u + vt)$$

$$= A^{(1)} ReLU\left(A^{(0)}x + B^{(0)}\right) + B^{(1)}$$



算法原理

- **临界点搜索**



多维情况下， O_L局部情况

Algorithm 1 Algorithm for 2-linearity testing. Computes the location of the only critical point in a given range or rejects if there is more than one.

Function f, range $[t_1, t_2]$, ε

$m_1 = \frac{f(t_1+\varepsilon)-f(t_1)}{\varepsilon}$ ▷ Gradient at t_1

$m_2 = \frac{f(t_2)-f(t_2-\varepsilon)}{\varepsilon}$ ▷ Gradient at t_2

$y_1 = f(a), y_2 = f(b)$

$x = a + \frac{y_2-y_1-(b-a)m_2}{m_1-m_2}$ ▷ Candidate critical point

$\hat{y} = y_1 + m_1 \frac{y_2-y_1-(b-a)m_2}{m_1-m_2}$ ▷ Expected value at candidate

$y = f(x)$ ▷ True value at candidate

if $\hat{y} = y$ **then return** x

else return "More than one critical point"

end if

算法原理

- **权重绝对值计算**
  - **由原始模型O_L、使$ReLU_i$为零的临界点x_i、输入空间内随机方向e_j：**

$$\frac{\partial^2 O_L}{\partial e_j^2}|_{x_i} = \frac{\partial O_L}{\partial e_j}|_{x_i+c\cdot e_j} - \frac{\partial O_L}{\partial e_j}|_{x_i-c\cdot e_j} = \pm\left(A_{ji}^{(0)} A_i^{(1)}\right)$$

  - **$c>0$且很小，x_i是临界点，所以在$x_i \pm c\cdot e_j$，$\frac{\partial O_L}{\partial e_j}$只在$ReLU_i$有差别**
  - **在e_1和e_2两个方向查询计算出$\left|A_{1i}^{(0)} A^{(1)}\right|$和$\left|A_{2i}^{(0)} A^{(1)}\right|$，可以得到比值$\left|A_{1i}^{(0)} / A_{ki}^{(0)}\right|$**
  - **根据$ReLU$的缩放，可以设$\left|A_{1i}^{(0)}\right| = 1$，得到其他绝对值**

算法原理

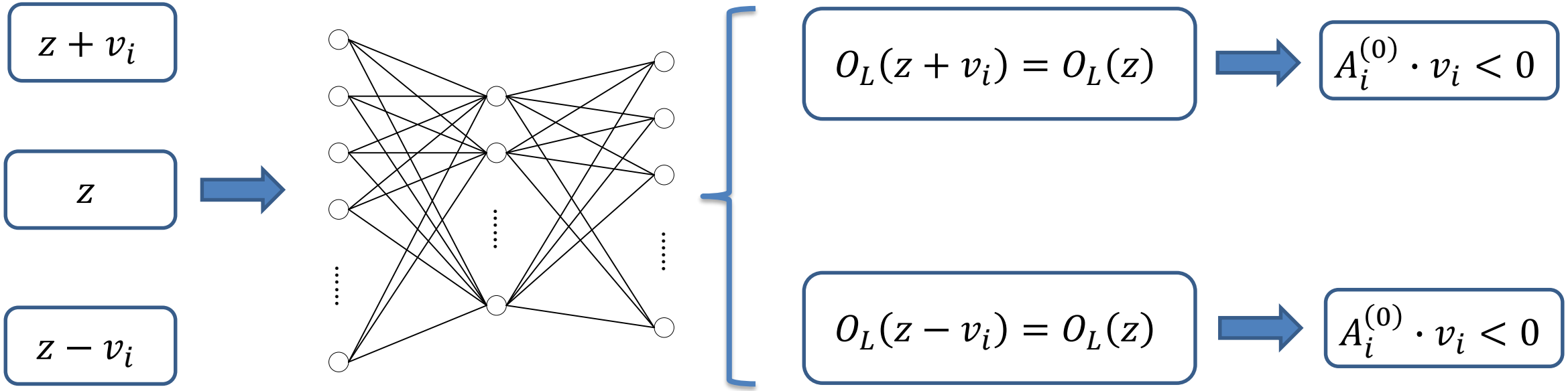
- **相对符号计算**
  - **考虑从方向$(e_j + e_k)$求二元偏导数**

$$\frac{\partial^2 O_L}{\partial (e_j + e_k)^2}|_{x_i} = \pm \left(A_{ji}^{(0)} A_i^{(1)} \pm A_{ki}^{(0)} A_i^{(1)} \right)$$

  - **如果$A_{ji}^{(0)}$和$A_{ki}^{(0)}$对梯度的贡献相互抵消，则符号相反**
  - **如果$A_{ji}^{(0)}$和$A_{ki}^{(0)}$对梯度的贡献互相复合，则符号相同**
  - **因此，对不同方向求导，可以得到每个ReLU内权重相互之间的符号关系**

算法原理

- **符号恢复**
  - **向量z：因为$h < d$，所以存在$\hat{A}^{(0)}z = \vec{0}$**
  - **向量v_i：对于每个$ReLU_i$，$v_i A^{(0)} = e_i$，即沿v_i方向移动只会更改$ReLU_i$的输入值**



实验结果

- 数据集：MNIST、CIFAR-10
- 模型：单隐层全连接神经网络，隐藏单元16~512，参数个数12000~100000

# of Parameters	12,500	25,000	50,000	100,000
Fidelity	100%	100%	100%	99.98%
Queries	$2^{17.2}$	$2^{18.2}$	$2^{19.2}$	$2^{20.2}$

- MNIST与CIFAR-10实验结构类似，因为该方法与训练数据无关

模型窃取

优劣分析

优劣分析

- 基于学习的模型窃取
  - 优势
    - 通用性高，不需要关注目标模型的结构
  - 劣势
    - 训练数据获取困难
    - 受非确定性参数影响，如参数初始化、训练数据顺序等
- 直接提取
  - 优势
    - 替代模型准确性高
  - 劣势
    - 复杂模型提取困难
    - 通用性低，扩展困难

应用总结

- 模型窃取应用
  - 重构具有经济效益的模型，如MLssA
  - 绕过安全检测，如垃圾邮件检测，恶意软件检测等
  - 对抗样本可迁移性，模型越相似，可迁移性越高
- 未来发展
  - 如何对抗防御模型窃取的方法。如Ian Goodfellow提出将数据分区输入到多个模型，并且还对输出根据差分隐私添加噪声
  - 如何解决直接提取对复杂神经网络的困难。例如多层ReLU神经网络无法确定是哪一层产生的临界点；其他激活函数（Sigmoid、tanh等）

参考文献

[1] Pal S , Y Gupta, Shukla A , et al. ActiveThief: Model Extraction Using Active Learning and Unannotated Public Data[C]. Proceedings of the AAAI Conference on Artificial Intelligence, 2020, 34(1):865-872.
[2] Jagielski M , Carlini N , Berthelot D , et al. High Accuracy and High Fidelity Extraction of Neural Networks[C]. USENIX Security Symposium 2020.
[3] Milli S , Schmidt L , Dragan A D , et al. Model Reconstruction from Model Explanations[J]. arXiv, 2018.
[4] F Tramèr, Fan Z , Juels A , et al. Stealing Machine Learning Models via Prediction APIs[C]. 25th USENIX Security Symposium, USENIX Security 16, Austin, TX, USA, August 10-12, 2016. 2016.

知人者智，自知者明。
胜人者有力，自胜者
强。知足者富。强行
者有志。不失其所者
久。死而不亡者，寿。

谢谢！