CS 7775

Seminar in Computer Security: Machine Learning Security and Privacy
Fall 2023

Alina Oprea
Associate Professor
Khoury College of Computer Science

September 28 2023

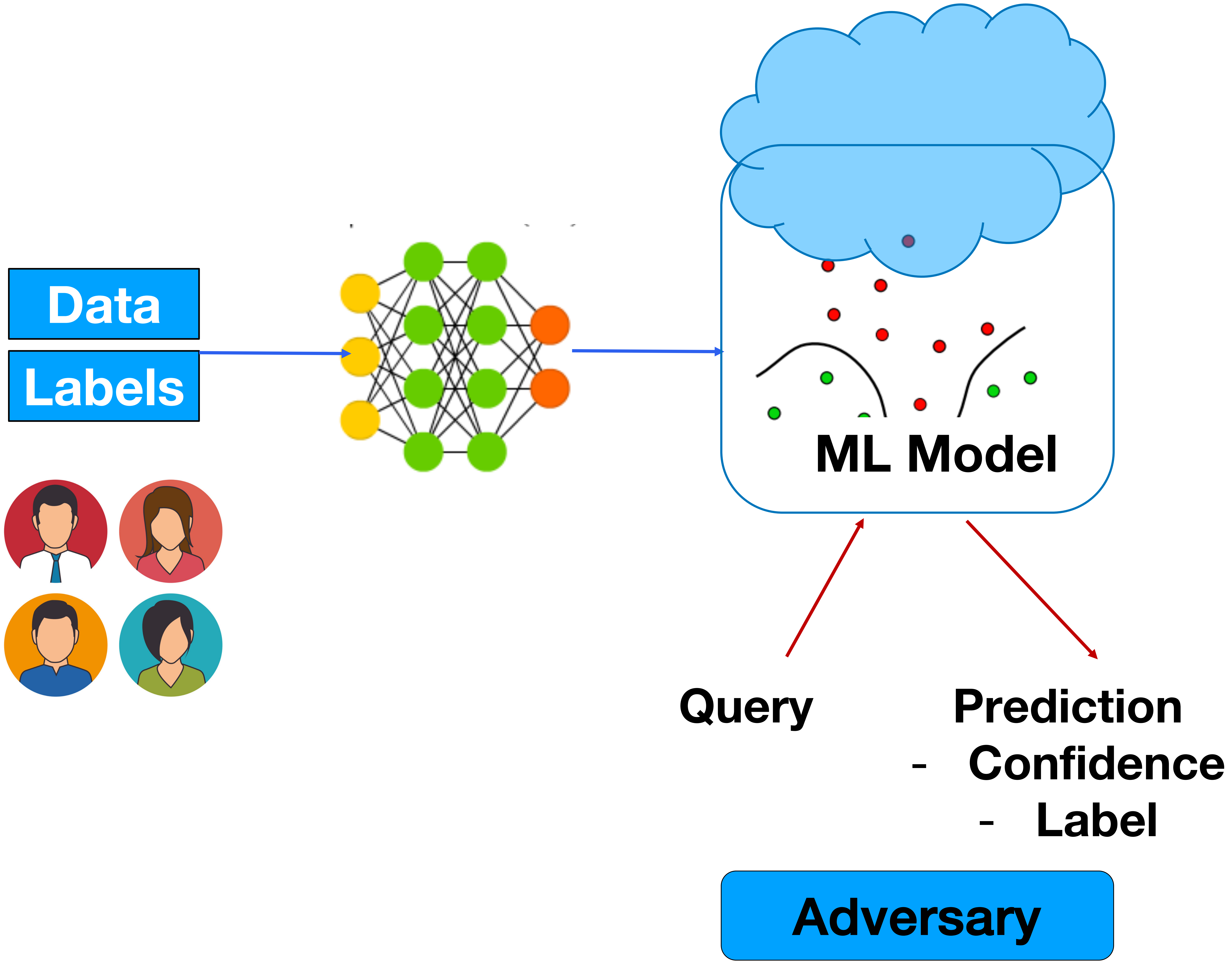
Adversarial Machine Learning: Taxonomy

Attacker's Objective

	Integrity Target small set of points	**Availability** Target entire model	**Privacy** Learn sensitive information
Training	Targeted Poisoning Backdoor Poisoning Subpopulation Poisoning	Poisoning Availability Model Poisoning	-
Testing	Evasion Attacks	Sponge Adversarial Examples	Reconstruction Membership Inference Model Extraction

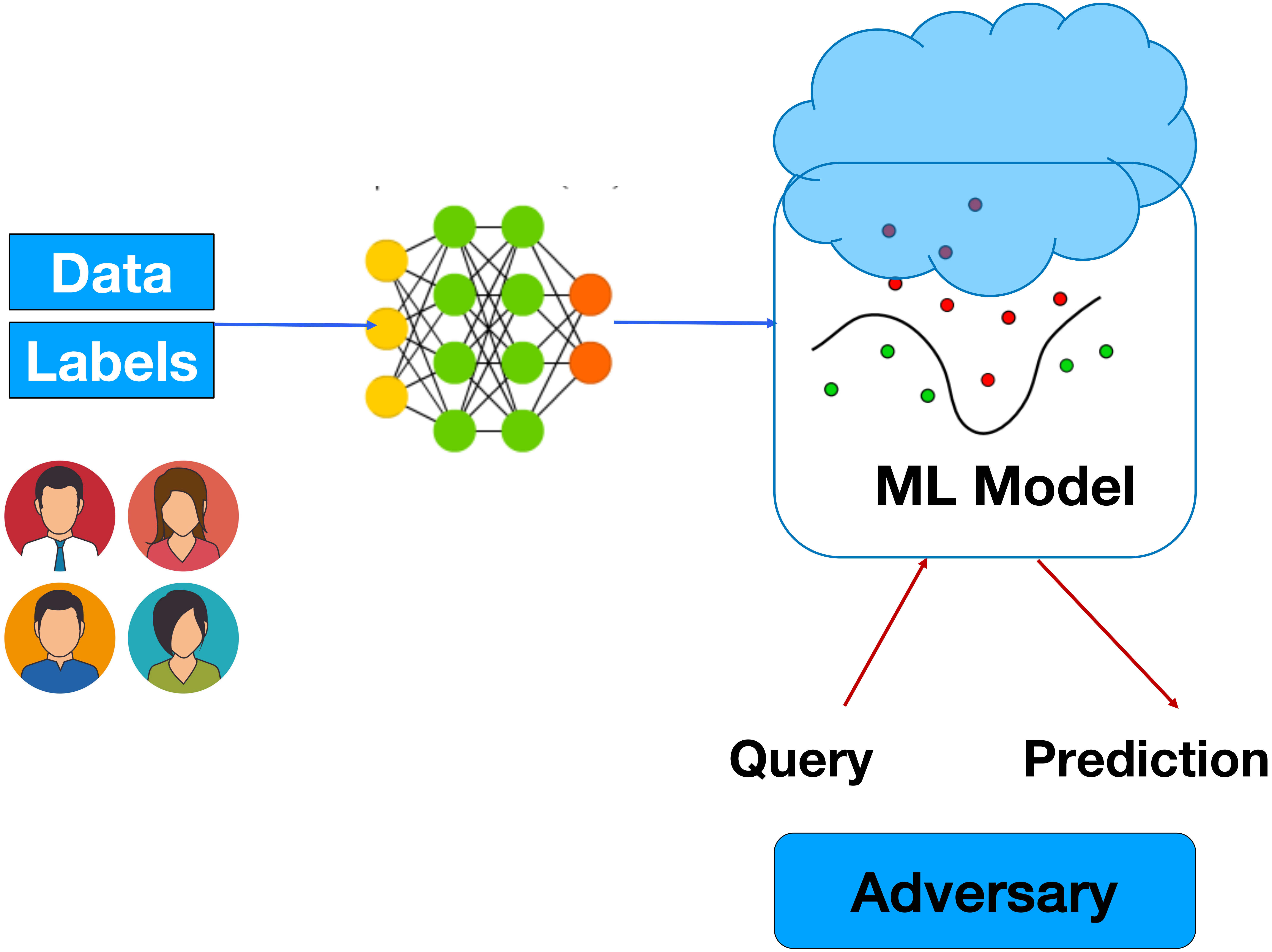
Learning Stage

Privacy Attacks in ML



- ML model is trained by third-party collecting user data
- **Black-box**
  - Query access to model
  - Model returns confidence (probability of prediction) or only predicted label
- What can the adversary learn about the training set?

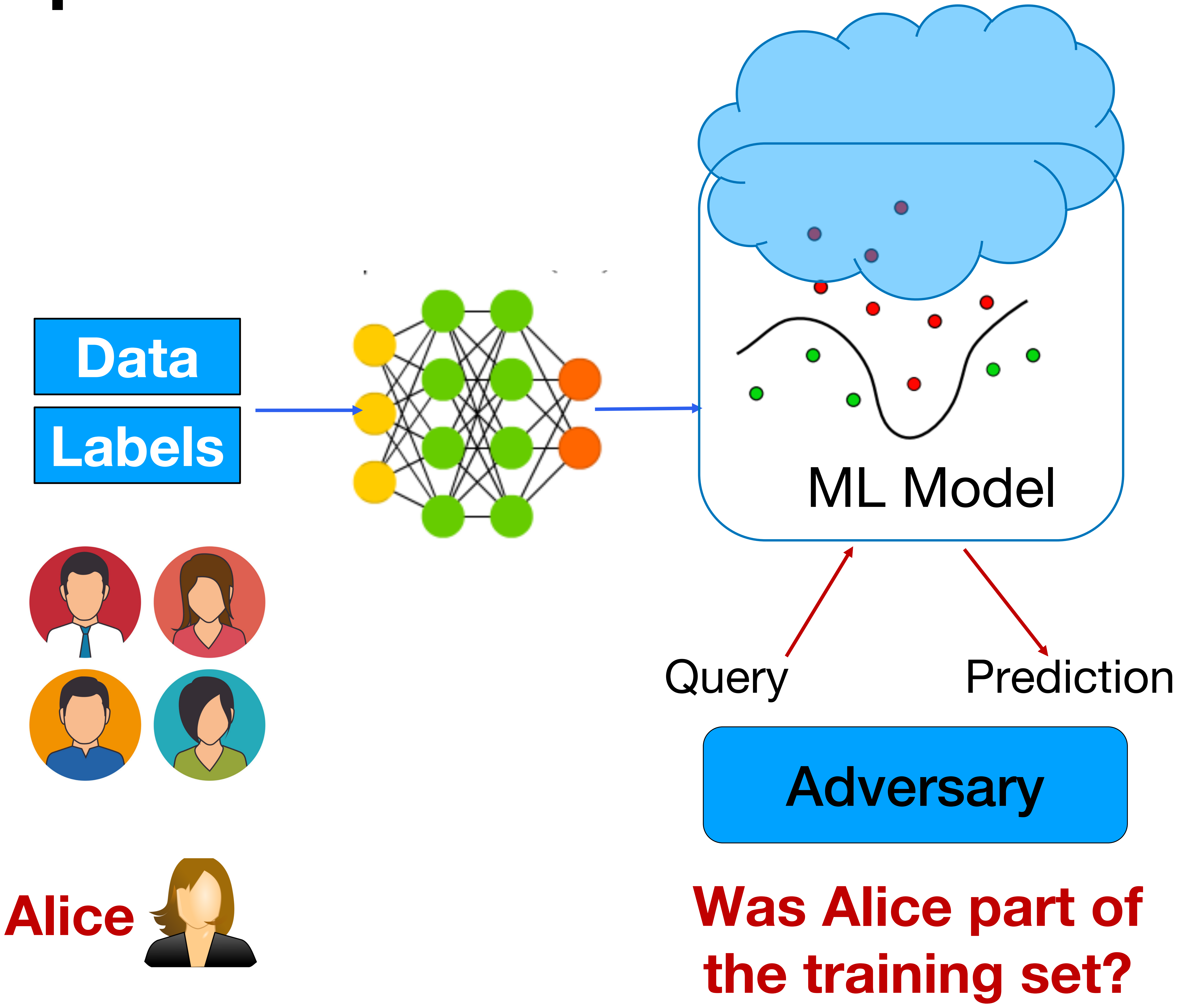
Privacy Attacks in ML



- Reconstruction: Extract sensitive data from training sets
  - Statistical databases: [DN03]
  - DNNs: [HVY22]
  - LLM memorization: [CTW21]
- Membership Inference: Determine if data sample was in training set
  - [SSS17], [YGF18], [CCN22]
- Property Inference: Learn global properties about the training set
  - [MGC22], [CAO23]

Membership Inference

- Learn if a user participated in training set of model
  - Being part of ML training set might be sensitive
- Introduced for statistical computations on genomic data [HSR08]
- First membership inference attack on DNNs [SSS17]
- More efficient attacks [YGF18], [CCN22]



Membership Inference Attacks From First Principles

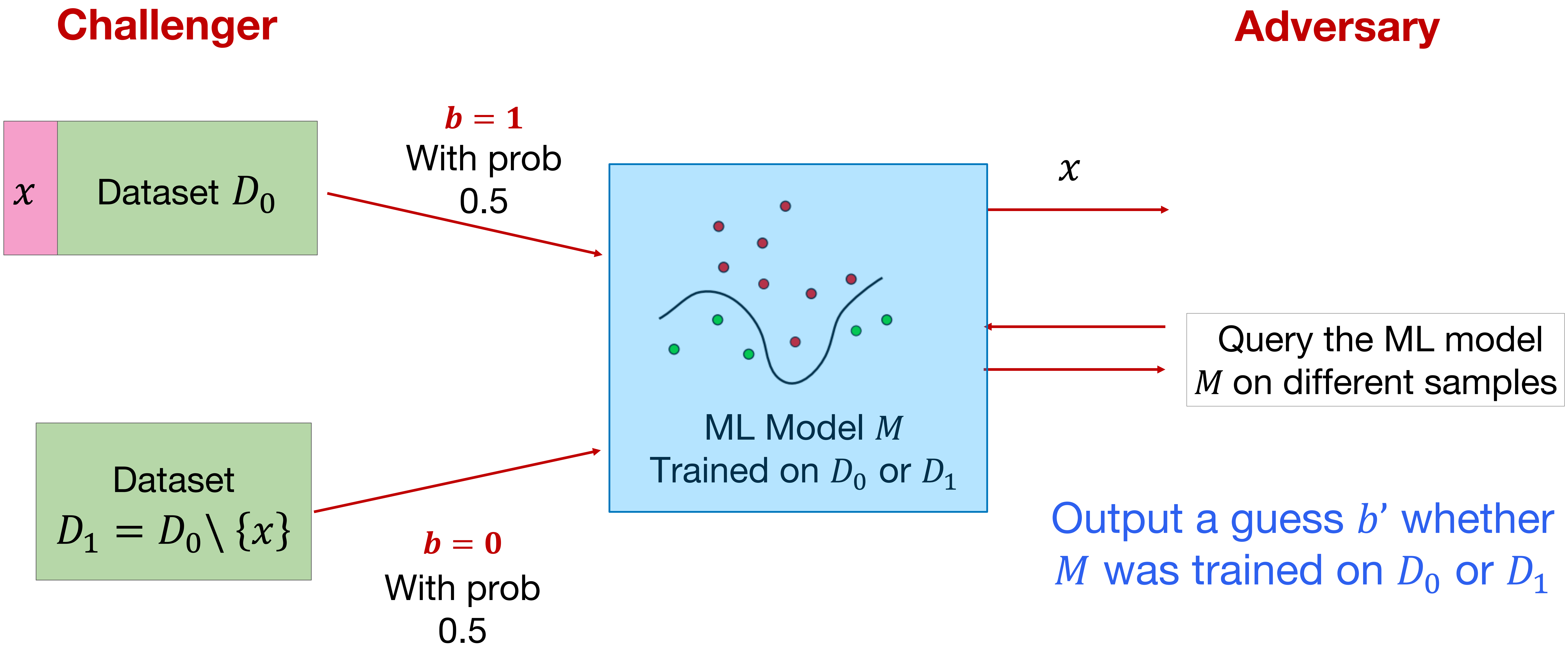
Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, Florian Tramèr

Slides adapted from John Abascal

Overview

- Membership Inference (MI) Overview
- Motivating Example for Current Attacks
- Problem with Current Attacks
- Online Likelihood Ratio Attack (LiRA)
- An Offline Variant
- Empirical Results and Practical Considerations

Membership Inference (MI)



Membership Inference Attacks

Definition 1 (Membership inference security game). *The game proceeds between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$:*

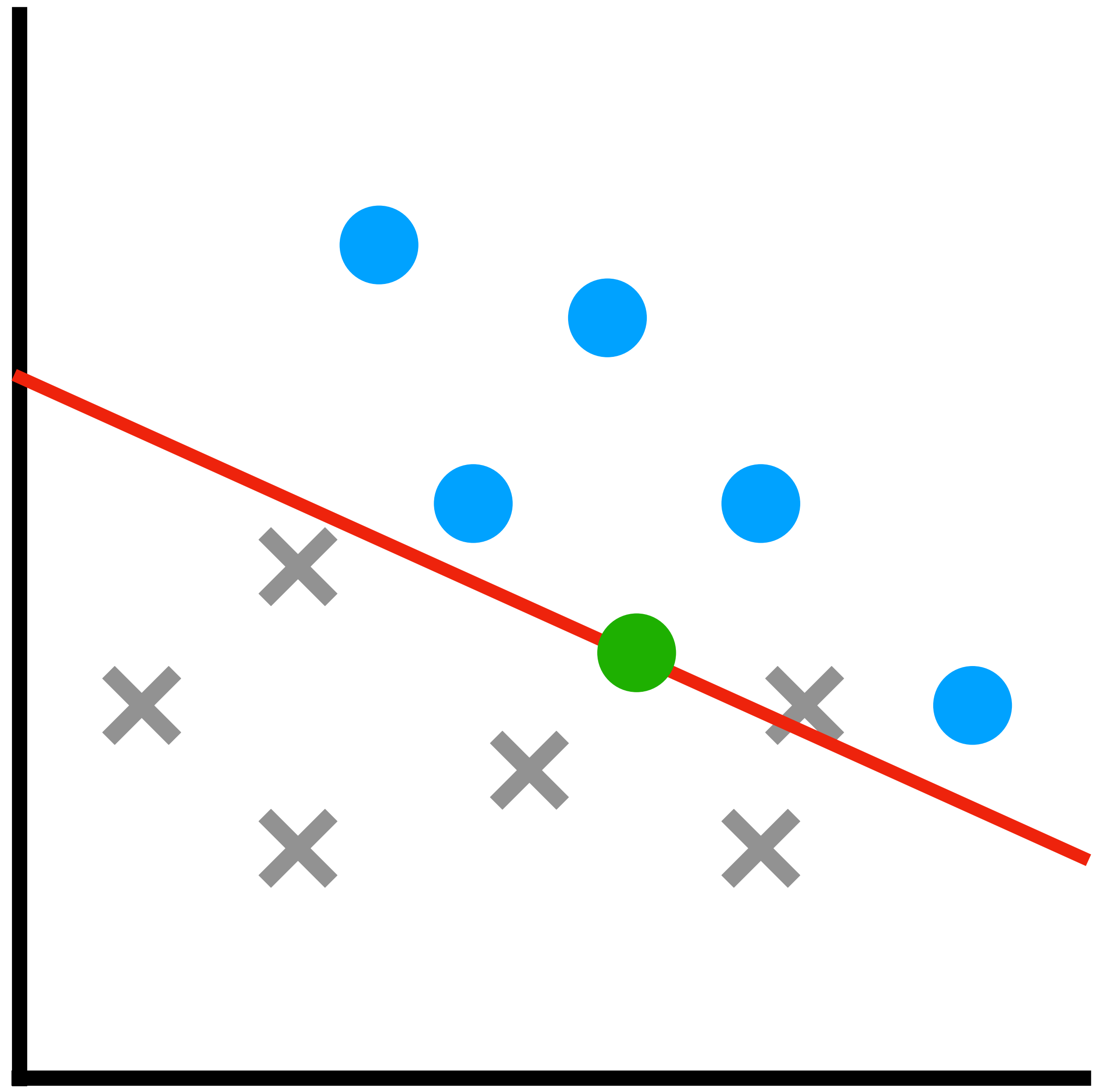
1) *The challenger samples a training dataset $D \leftarrow \mathbb{D}$ and trains a model $f_\theta \leftarrow \mathcal{T}(D)$ on the dataset D.*
2) *The challenger flips a bit b, and if $b = 0$, samples a fresh challenge point from the distribution $(x, y) \leftarrow \mathbb{D}$. Otherwise, the challenger selects a random challenge point from the training set $(x, y) \leftarrow^{\$} D$.*
3) *The challenger sends (x, y) to the adversary.*
4) *The adversary gets query access to the distribution $\mathbb{D}$, and to the model f_θ, and outputs a bit $\hat{b} \leftarrow \mathcal{A}^{\mathbb{D},f}(x, y)$.*
5) *Output 1 if $\hat{b} = b$, and 0 otherwise.*

Overview

- Membership Inference (MI) Overview
- Motivating Example for Current Attacks
- Problem with Current Attacks
- Online Likelihood Ratio Attack (LiRA)
- An Offline Variant
- Empirical Results and Practical Considerations

A Motivating Example

- Suppose we have a predictive model with **low capacity**, $\mathcal{M}$, which outputs the probability of an individual, x, having a disease ($\mathcal{M}(x) \in [0,1]$)
  - $\mathcal{M}$ is trained on dataset D, and when we make the query $\mathcal{M}_D(Bob)$, the output is 0.55



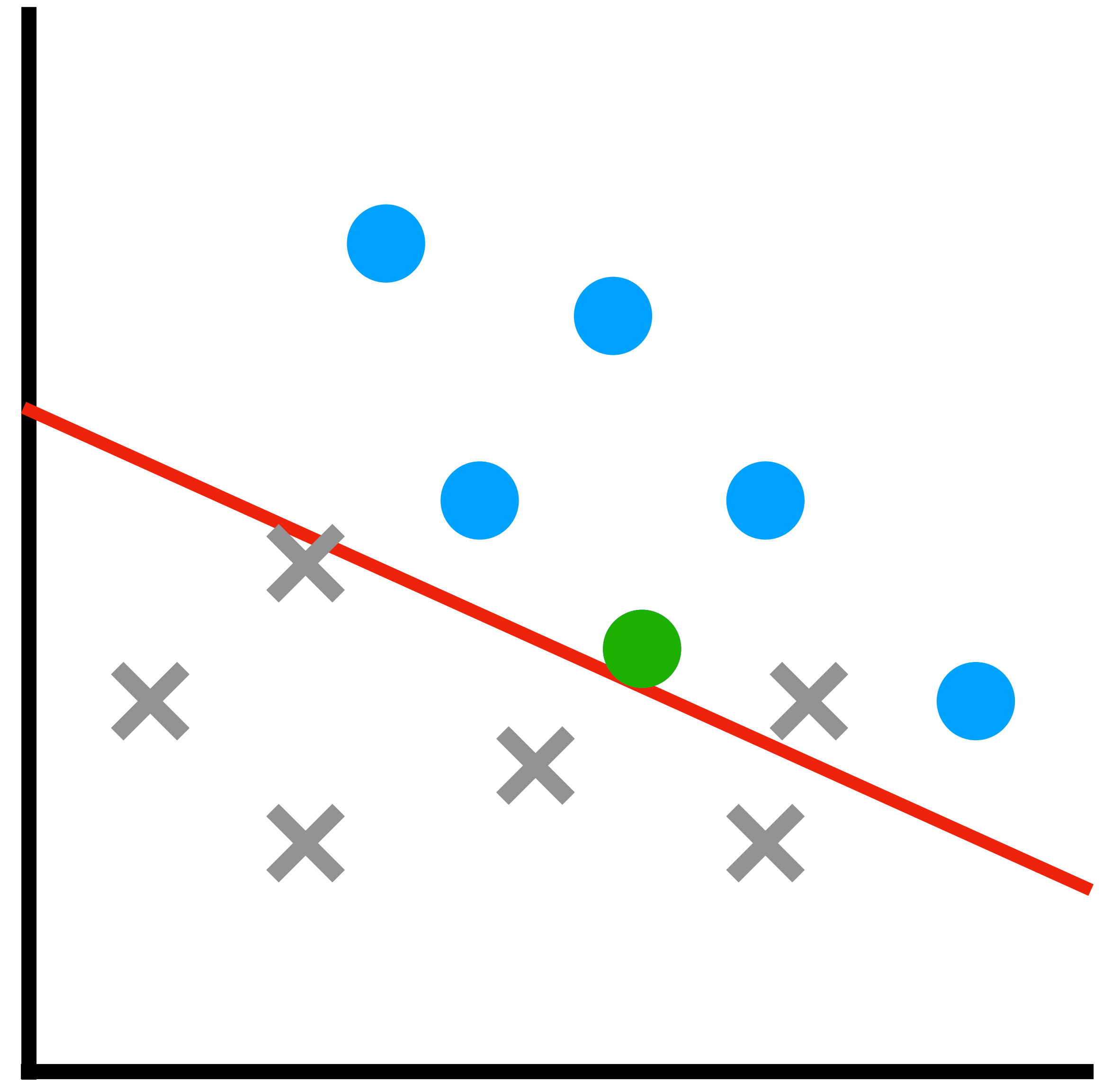
A Motivating Example

- Suppose we have a predictive model with **low capacity**, $\mathcal{M}$, which outputs the probability of an individual, x, having a disease ($\mathcal{M}(x) \in [0,1]$)

  - $\mathcal{M}$ is trained on dataset D, and when we make the query $\mathcal{M}_D(Bob)$, the output is 0.55

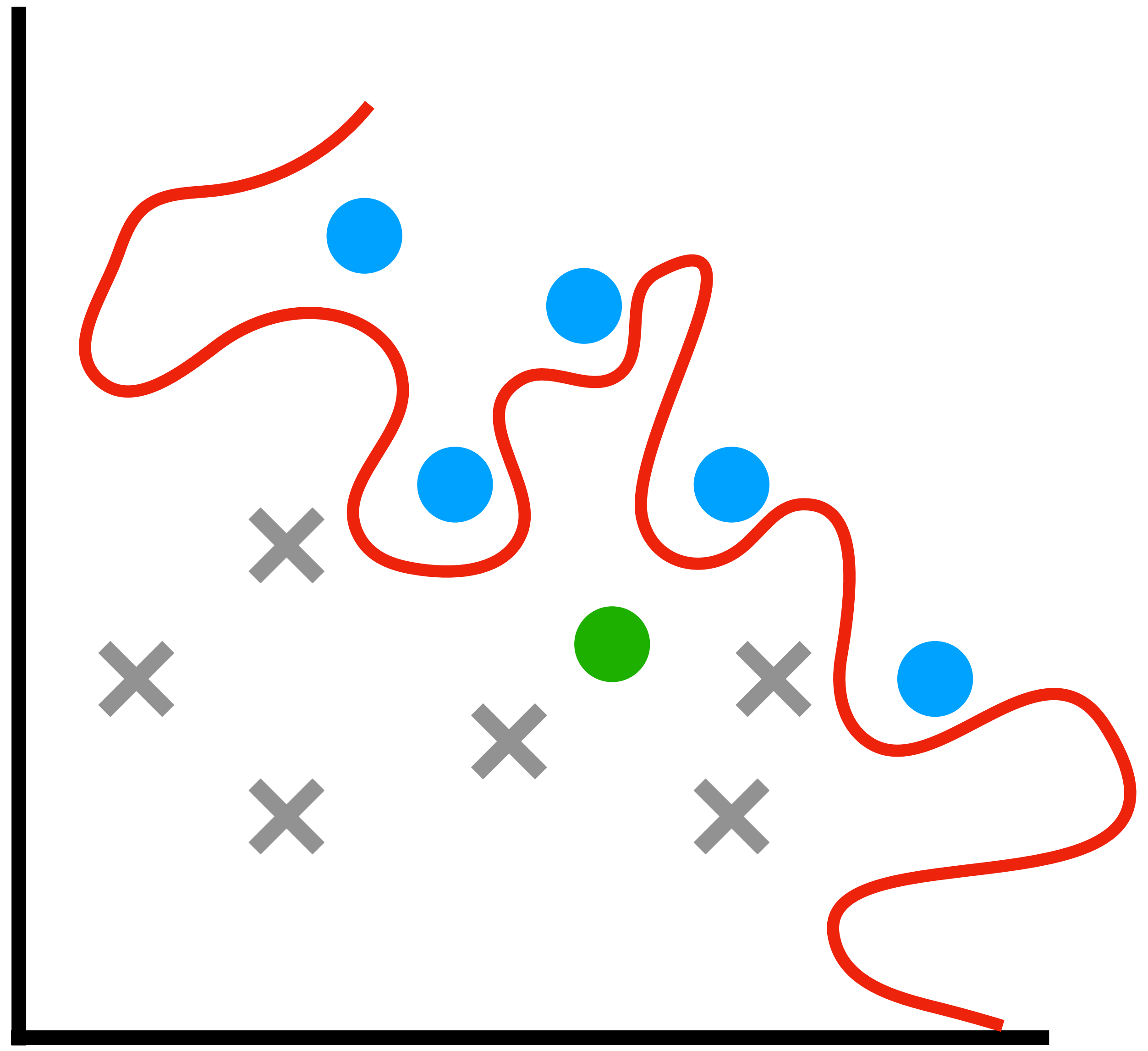
  - $\mathcal{M}$ is trained on dataset D + Bob, and when we make the query $\mathcal{M}_{D+Bob}(Bob)$, the output is 0.57

    - **It is unclear whether Bob has this medical condition**



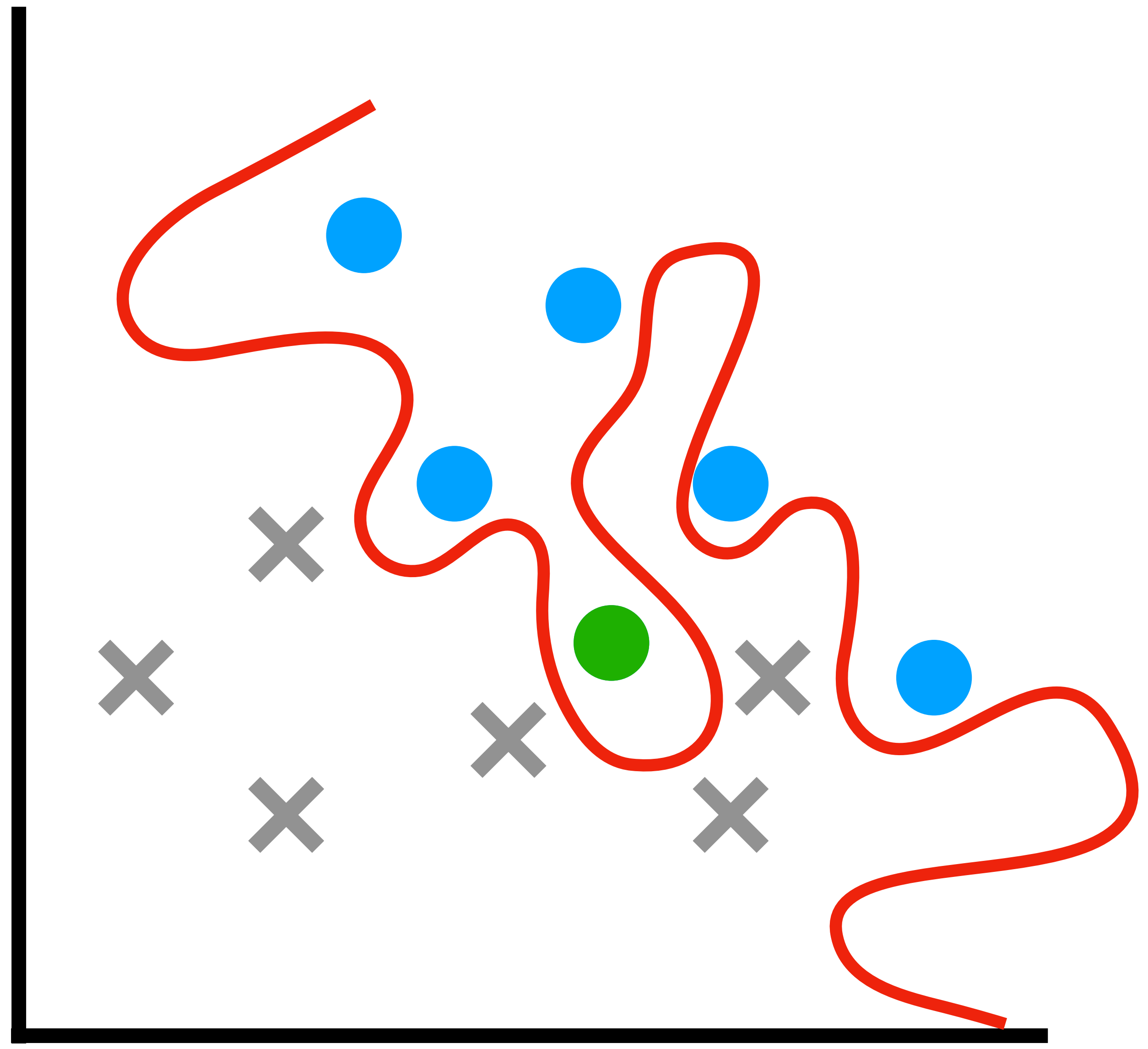
A Motivating Example

- Suppose we have a predictive model with **high capacity**, $\mathcal{M}$, which outputs the probability of an individual, x, having a disease ($\mathcal{M}(x) \in [0,1]$)
  - $\mathcal{M}$ is trained on dataset D, and when we make the query $\mathcal{M}_D(Bob)$, the output is 0.36



A Motivating Example

- Suppose we have a predictive model with **high capacity**, $\mathcal{M}$, which outputs the probability of an individual, x, having a disease ($\mathcal{M}(x) \in [0,1]$)
  - $\mathcal{M}$ is trained on dataset D, and when we make the query $\mathcal{M}_D(Bob)$, the output is 0.36
  - $\mathcal{M}$ is trained on dataset D + Bob, and when we make the query $\mathcal{M}_{D+Bob}(Bob)$, the output is 0.70
    - **Although we have 70% confidence that Bob has the disease, the drastic change in confidence tells us that Bob *definitely* has the disease**

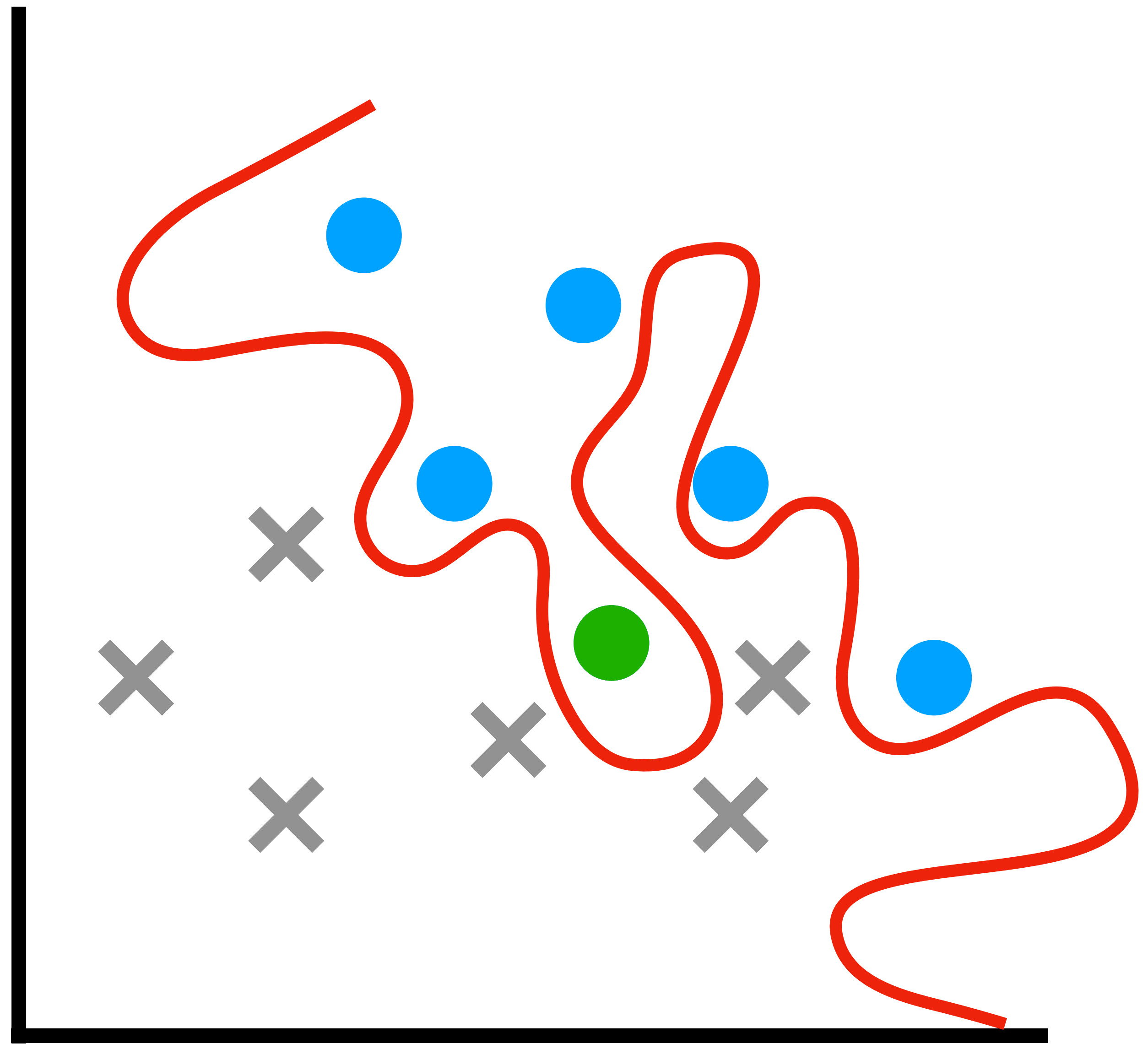


Loss-Based Membership Inference

- We can use the model's confidence (or loss) on a target point as a test statistic **[Yeom et al. '18]**

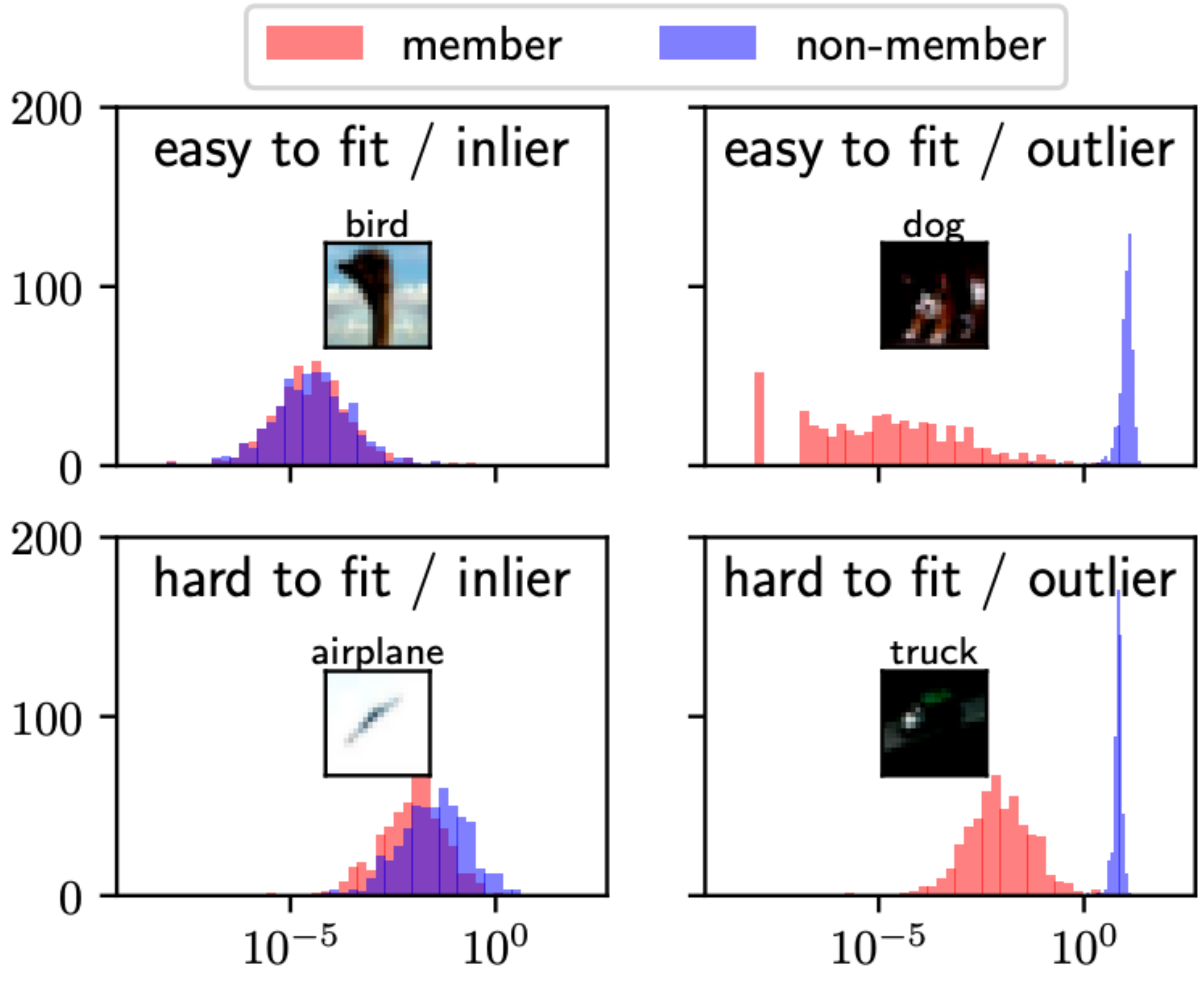
- Determine that a point is a member if its loss < T; otherwise the point is non-member

  - Global threshold: average loss of training points



The Problem with Current Attacks

- Prior work evaluates attacks using average-case success metrics (i.e., accuracy over a dataset)
  - The attacks themselves typically involve computing a single test statistic and thresholding the **IN** vs. **OUT** classification **[Yeom et al. '18]**
- **Privacy is not an average case metric**
  - Certain examples are "harder" to overfit than others
  - Assuming that confidences are on an equal scale ignores the reality of per-example hardness

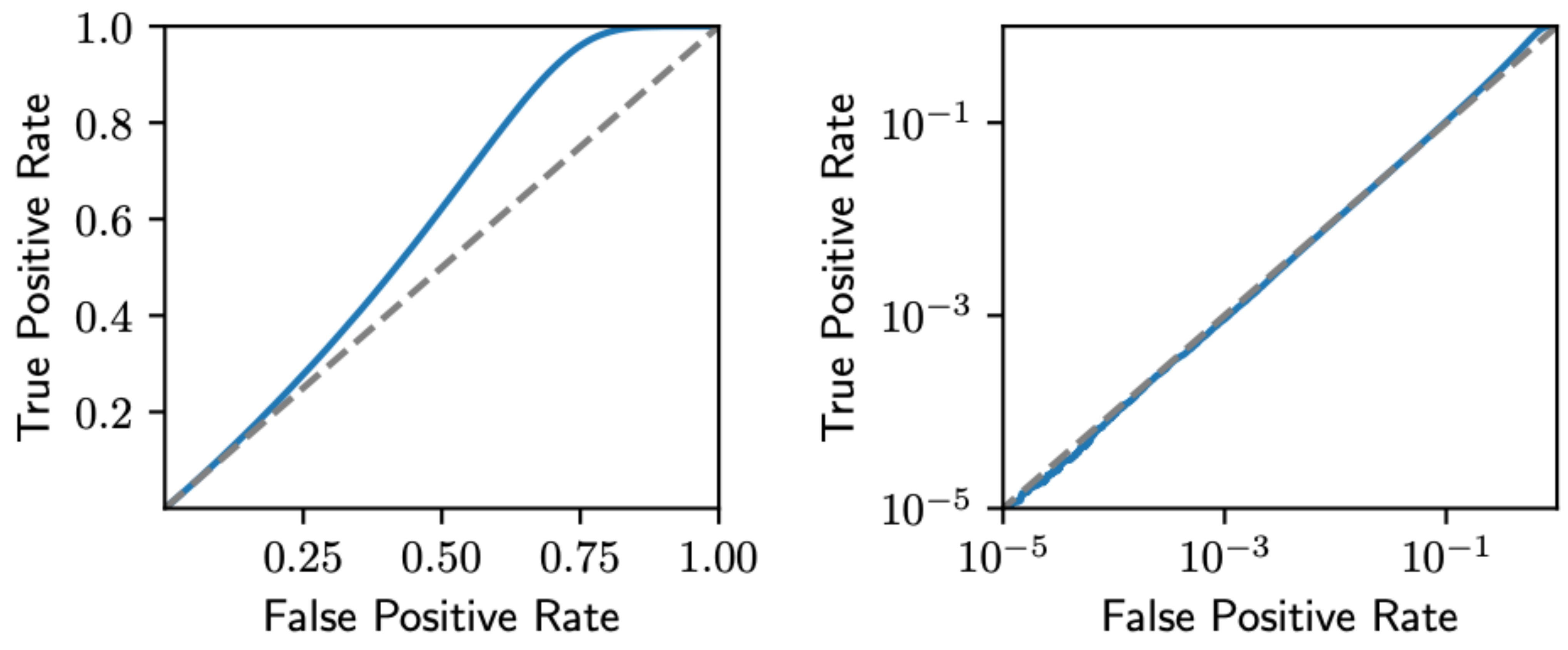


The Problem with Current Attacks

- Balanced Accuracy of the **LOSS** Attack

$$\mathbb{P}_{x,y,f,b}[\mathcal{A}^{\mathbb{D},f}(x,y) = b]$$

  - This accuracy is *symmetric* (Equal cost to FP and FN)
    - Depending on the setting, we might care more about FP or FN
  - This accuracy is an *average-case* metric
    - **Attack A** perfectly targets 0.1% of the data and guesses on the rest. **Attack B** succeeds with 50.05% on any given user
    - Both have same accuracy



(a) linear scale (b) log scale

Paper’ Goals

Main Objectives

1. **Create an attack that can effectively measure when someone *is* a member**

- Have a high true positive rate for a fixed false positive rate

2. **Design the attack in a principled manner**

Overview

- Membership Inference (MI) Overview
- Motivating Example for Current Attacks
- Problem with Current Attacks
- Online Likelihood Ratio Attack (LiRA)
- An Offline Variant
- Empirical Results and Practical Considerations

MI as Hypothesis Testing

- MI requires the adversary to distinguish between two worlds
  - **IN: The world where the model was trained on the target point**
  - **OUT: The world where the model wasn't trained on the target point**
- Both of these worlds induce distributions over the trained model
  - $\mathbb{Q}_{in}(x, y) = \{f \leftarrow \mathcal{T}(D \cup \{(x, y)\})\}$
  - $\mathbb{Q}_{out}(x, y) = \{f \leftarrow \mathcal{T}(D)\}$

MI as Hypothesis Testing

- Given a model f, and a target example (x, y) we want to distinguish between these two distributions
    - We can view this task as a **hypothesis test** between two hypotheses: f was sampled from $\mathbb{Q}_{in}$ or $\mathbb{Q}_{out}$
- The **Neyman-Pearson Lemma** states that the best hypothesis test at a fixed FPR is obtained by thresholding the *Likelihood-ratio Test between the two hypotheses*

$$\Lambda(f; x, y) = \frac{p(f \mid \mathbb{Q}_{in}(x, y))}{p(f \mid \mathbb{Q}_{out}(x, y))}$$

Likelihood Ratio Test

- The exact likelihood ratio is typically intractable since $\mathbb{Q}_{in}$ and $\mathbb{Q}_{out}$ are not analytically known and f is high dimensional and unknown to the adversary

- Instead, we can measure $p(\ell(f(x), y) | \tilde{\mathbb{Q}}_{in/out}(x, y))$ where $\tilde{\mathbb{Q}}$ is the distribution of losses on (x, y) for models trained (**IN**) or not trained (**OUT**) on (x, y)

    - We individually model separate pairs of distributions $\tilde{\mathbb{Q}}_{in}$ and $\tilde{\mathbb{Q}}_{out}$ for each example (x, y)

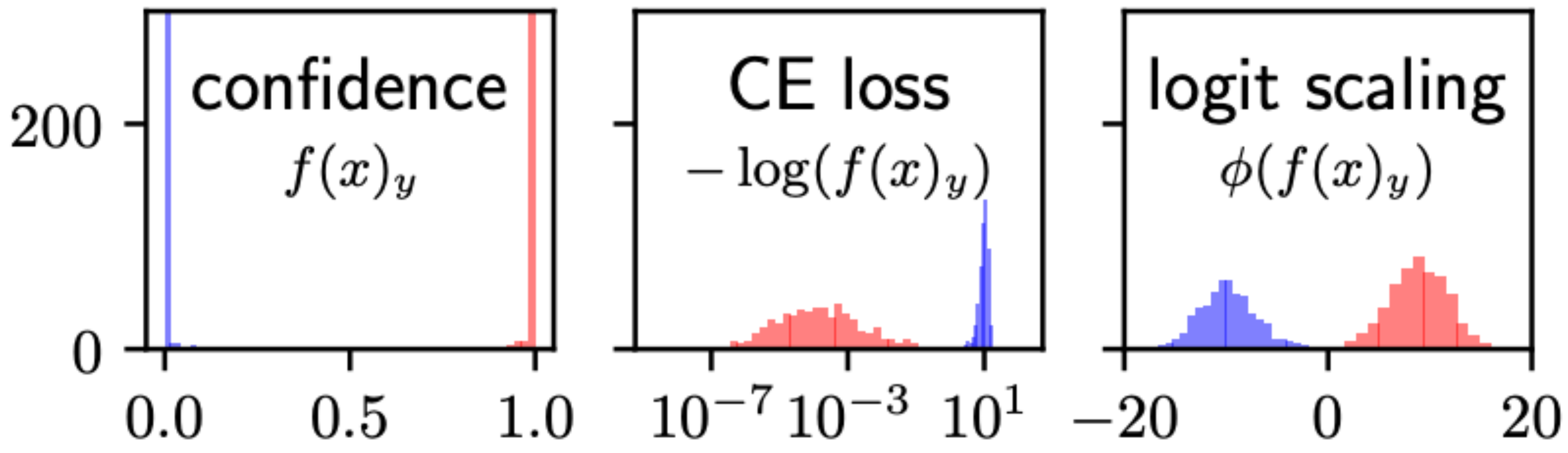
Likelihood Ratio Test

- To improve performance at low FPR, the authors take a *parametric* approach by approximating the distributions using Gaussians

- Instead of using the loss directly, they use a scaled version of the model's prediction "confidence"

    - First look at the model's confidence $f(x)_y = \exp(-\ell(f(x), y))$ which is in $[0,1]$

    - Then scale it to obtain a statistic in $(-\infty, \infty)$ using $\phi(p) = \log(\frac{p}{1-p})$



Likelihood Ratio Attack (LiRA) Strategy

1. Assume we have black-box access to some model f, a target example (x, y), and access to the underlying distribution, $\mathbb{D}$, where f 's training set was drawn from

2. Train several shadow models on datasets (sampled from $\mathbb{D}$) with and without (x, y) to mimic the worlds where (x, y) is **IN** and **OUT**

3. Aggregate the shadow models' prediction scores on (x, y) and compute sample mean/variance

4. Compare these Gaussians to the target model's scaled confidence by using the **likelihood ratio test**

Online LiRA Algorithm

Require: model f, example (x, y), data distribution $\mathbb{D}$

$\text{confs}_{\text{in}} = \{\}$

$\text{confs}_{\text{out}} = \{\}$

for N times **do**

  $D_{\text{attack}} \xleftarrow{\$} \mathbb{D}$

  $f_{\text{in}} \leftarrow \mathcal{T}(D_{\text{attack}} \cup \{(x, y)\})$ ▷ *train IN model*

  $\text{confs}_{\text{in}} \leftarrow \text{confs}_{\text{in}} \cup \{\phi(f_{\text{in}}(x)_y)\}$

  $f_{\text{out}} \leftarrow \mathcal{T}(D_{\text{attack}})$ ▷ *train OUT model*

  $\text{confs}_{\text{out}} \leftarrow \text{confs}_{\text{out}} \cup \{\phi(f_{\text{out}}(x)_y)\}$

end for

$\mu_{\text{in}} \leftarrow \texttt{mean}(\text{confs}_{\text{in}})$

$\mu_{\text{out}} \leftarrow \texttt{mean}(\text{confs}_{\text{out}})$

$\sigma^2_{\text{in}} \leftarrow \texttt{var}(\text{confs}_{\text{in}})$

$\sigma^2_{\text{out}} \leftarrow \texttt{var}(\text{confs}_{\text{out}})$

$\text{conf}_{\text{obs}} = \phi(f(x)_y)$ ▷ *query target model*

return $\Lambda = \dfrac{p(\text{conf}_{\text{obs}} \mid \mathcal{N}(\mu_{\text{in}}, \sigma^2_{\text{in}}))}{p(\text{conf}_{\text{obs}} \mid \mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}}))}$

Overview

- Membership Inference (MI) Overview
- Motivating Example for Current Attacks
- Problem with Current Attacks
- Online Likelihood Ratio Attack (LiRA)
- An Offline Variant
- Empirical Results and Practical Considerations

Motivation for an Offline Variant

- The online variant of LiRA has a *significant* usability limitation
  - **We need to train 2N *new* machine learning models for every set of membership inference queries**
  - **Assumes the queries are known in advance**
  - Doing this is very computationally expensive
- The authors provide an offline variant of the attack that uses a **one-sided hypothesis test**

Offline Attack Algorithm

- This attack trains the shadow models on randomly sampled datasets ahead of time and **never trains the shadow models on the target points**
- Same as LiRA but remove lines 5, 6, 10, and 12 (the steps where we would consider $\tilde{\mathbb{Q}}_{in}$)
- Lastly, line 15 becomes a **one-sided hypothesis test**
  - Measure probability of observing a confidence as high as the target model's under the null-hypothesis: (x, y) is **OUT**

Require: model f, example (x, y), data distribution $\mathbb{D}$

1: $\text{confs}_{\text{in}} = \{\}$
2: $\text{confs}_{\text{out}} = \{\}$
3: **for** N times **do**
4: $\quad D_{\text{attack}} \xleftarrow{\$} \mathbb{D}$
5: $\quad \cancel{f_{\text{in}} \leftarrow \mathcal{T}(D_{\text{attack}} \cup \{(x, y)\})}$ ▷ *train IN model*
6: $\quad \cancel{\text{confs}_{\text{in}} \leftarrow \text{confs}_{\text{in}} \cup \{\phi(f_{\text{in}}(x)_y)\}}$
7: $\quad f_{\text{out}} \leftarrow \mathcal{T}(D_{\text{attack}})$ ▷ *train OUT model*
8: $\quad \text{confs}_{\text{out}} \leftarrow \text{confs}_{\text{out}} \cup \{\phi(f_{\text{out}}(x)_y)\}$
9: **end for**
10: $\cancel{\mu_{\text{in}} \leftarrow \texttt{mean}(\text{confs}_{\text{in}})}$
11: $\mu_{\text{out}} \leftarrow \texttt{mean}(\text{confs}_{\text{out}})$
12: $\cancel{\sigma^2_{\text{in}} \leftarrow \texttt{var}(\text{confs}_{\text{in}})}$
13: $\sigma^2_{\text{out}} \leftarrow \texttt{var}(\text{confs}_{\text{out}})$
14: $\text{conf}_{\text{obs}} = \phi(f(x)_y)$ ▷ *query target model*
15: **return** $\Lambda = \cancel{\dfrac{p(\text{conf}_{\text{obs}} \mid \mathcal{N}(\mu_{\text{in}}, \sigma^2_{\text{in}}))}{p(\text{conf}_{\text{obs}} \mid \mathcal{N}(\mu_{\text{out}}, \sigma^2_{\text{out}}))}}$

$$\Lambda = 1 - \mathbb{P}[Z > \phi(f(x)_y)], Z \sim \mathcal{N}(\mu_{out}, \sigma^2_{out})$$

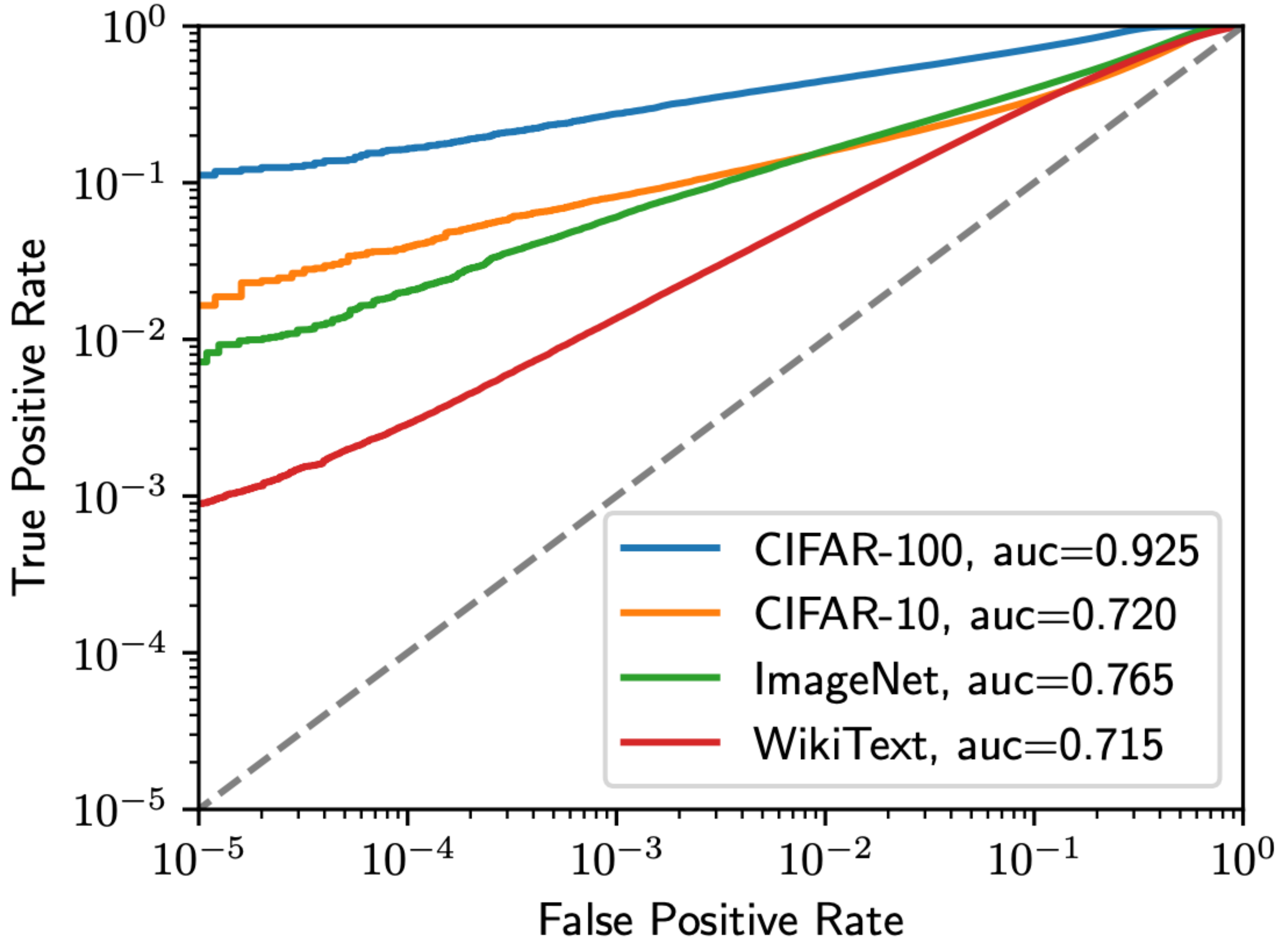
Overview

- Membership Inference (MI) Overview
- Motivating Example for Current Attacks
- Problem with Current Attacks
- Online Likelihood Ratio Attack (LiRA)
- An Offline Variant
- Empirical Results and Practical Considerations

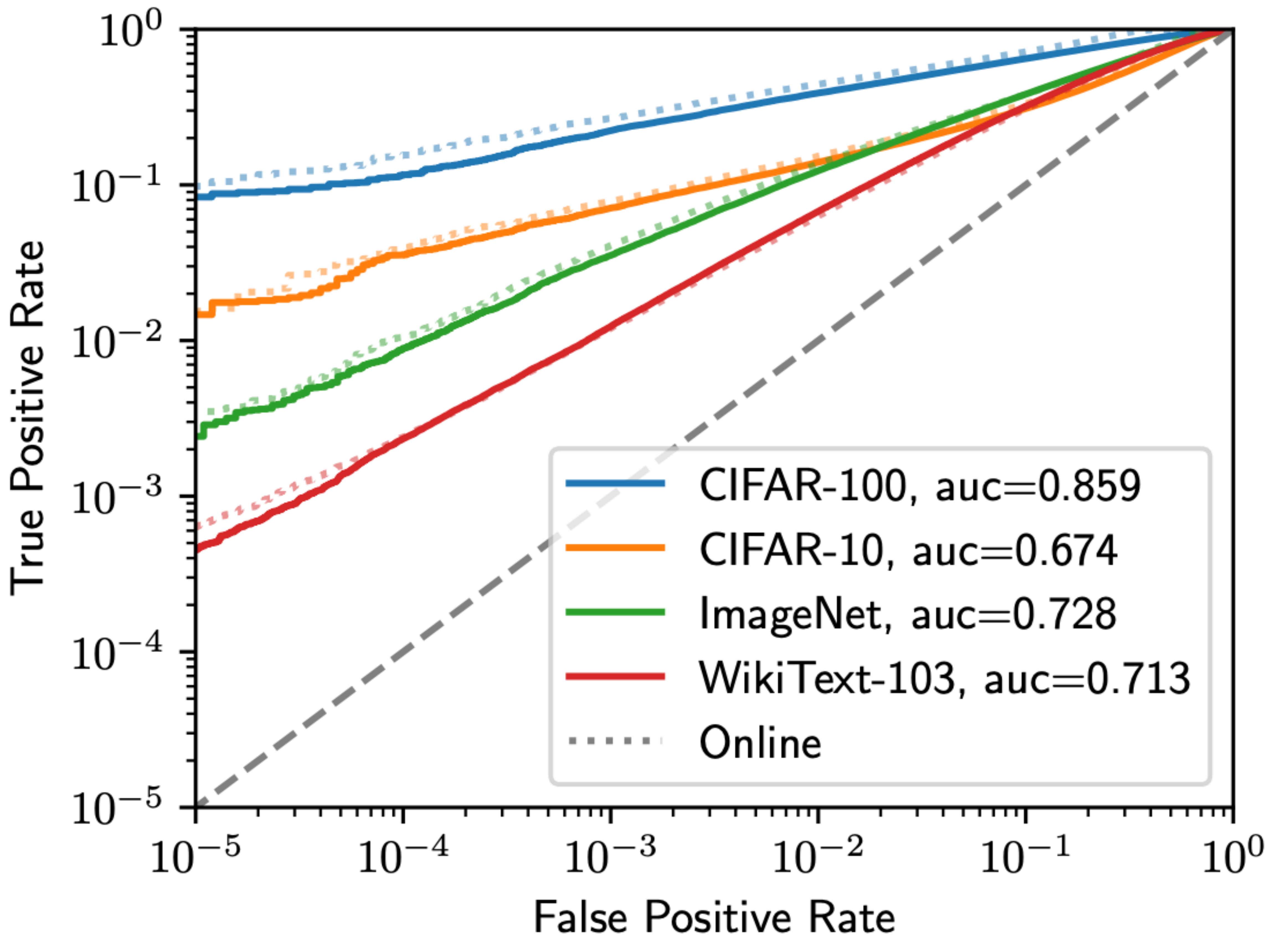
Attack Evaluation (TPR/FPR)

- The authors evaluate the attacks’ **TPR** and **FPR** over several complex model architectures and datasets
  - **Models:** ResNet(18, 34, 50), DenseNet121, MobileNetV2, etc.
  - **Datasets:** CIFAR-10, CIFAR-100, ImageNet, etc.
- Depending on the dataset and model architecture, the number of shadow models the authors trained differs
  - ImageNet: **N = 64**, CIFAR-10: **N = 256**

Attack Evaluation (TPR/FPR)



Online Attack



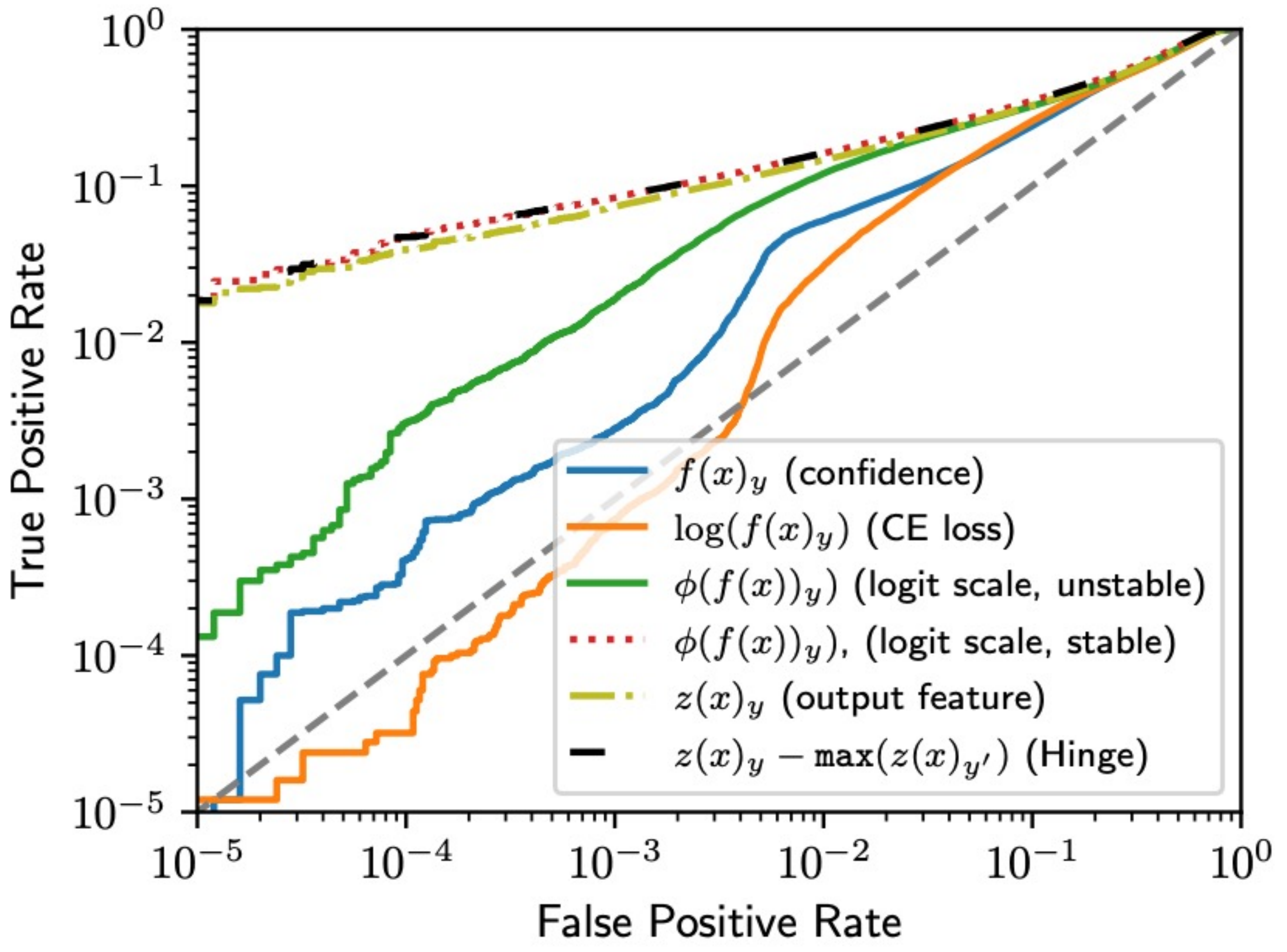
Offline Attack

Attack Evaluation (Stable Scaling)

- The authors found the original logit scaling $\phi(p) = \log(\frac{p}{1-p})$ to be unstable in practice

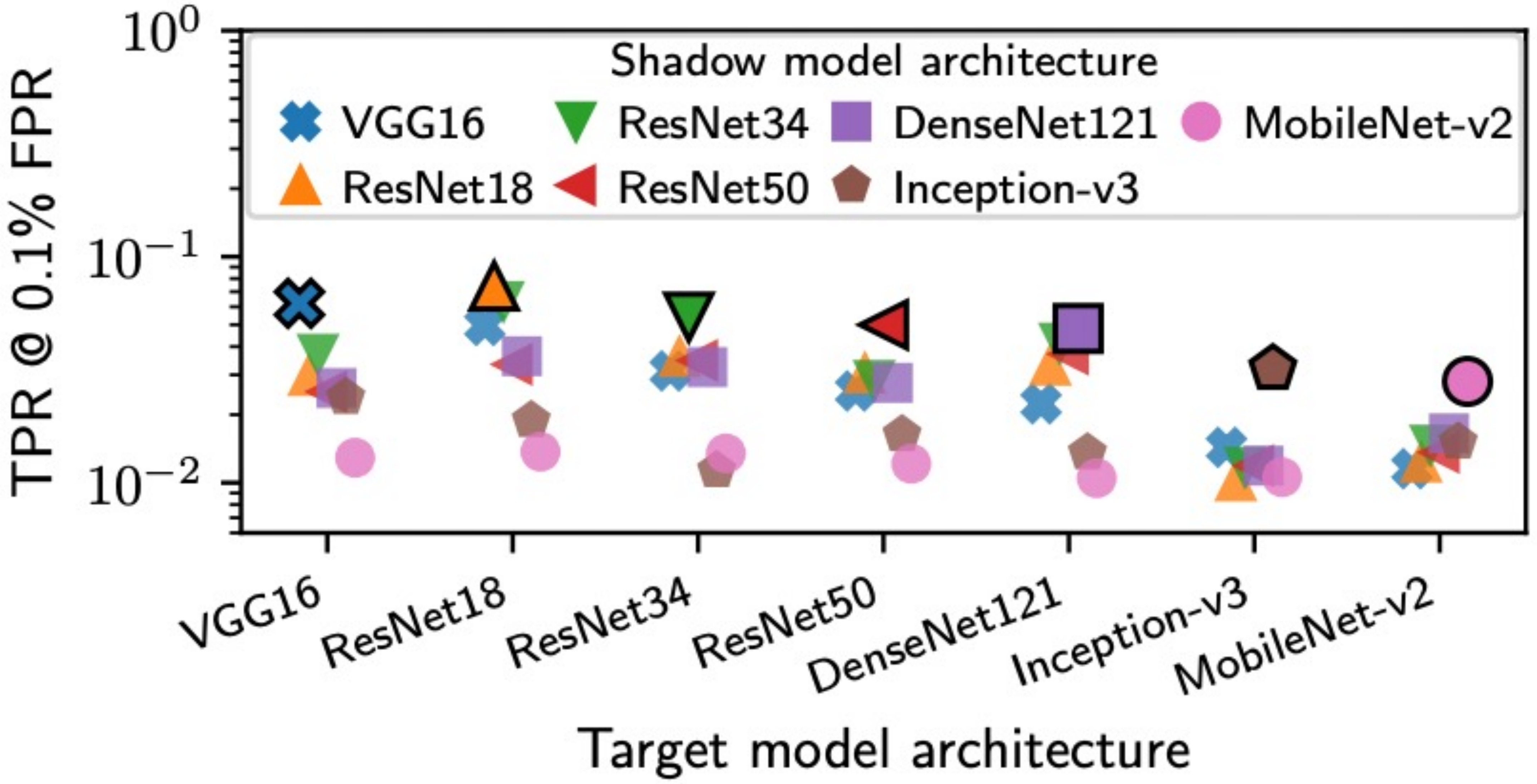
- Instead, we see an increase in attack success when using a stable variant

$$\phi_{stable} = \log(\frac{f(x)_y}{\sum_{y' \neq y} f(x)_{y'}})$$



Attack Evaluation (Architectures)

- **The attack succeeds against state-of-the-art CIFAR-10 models**
    - The degree to which the attack succeeds depends on the shadow model architecture
- Empirically, the attack performs best when the shadow models have the *same architecture* as the target model



Attack Success vs Model Accuracy

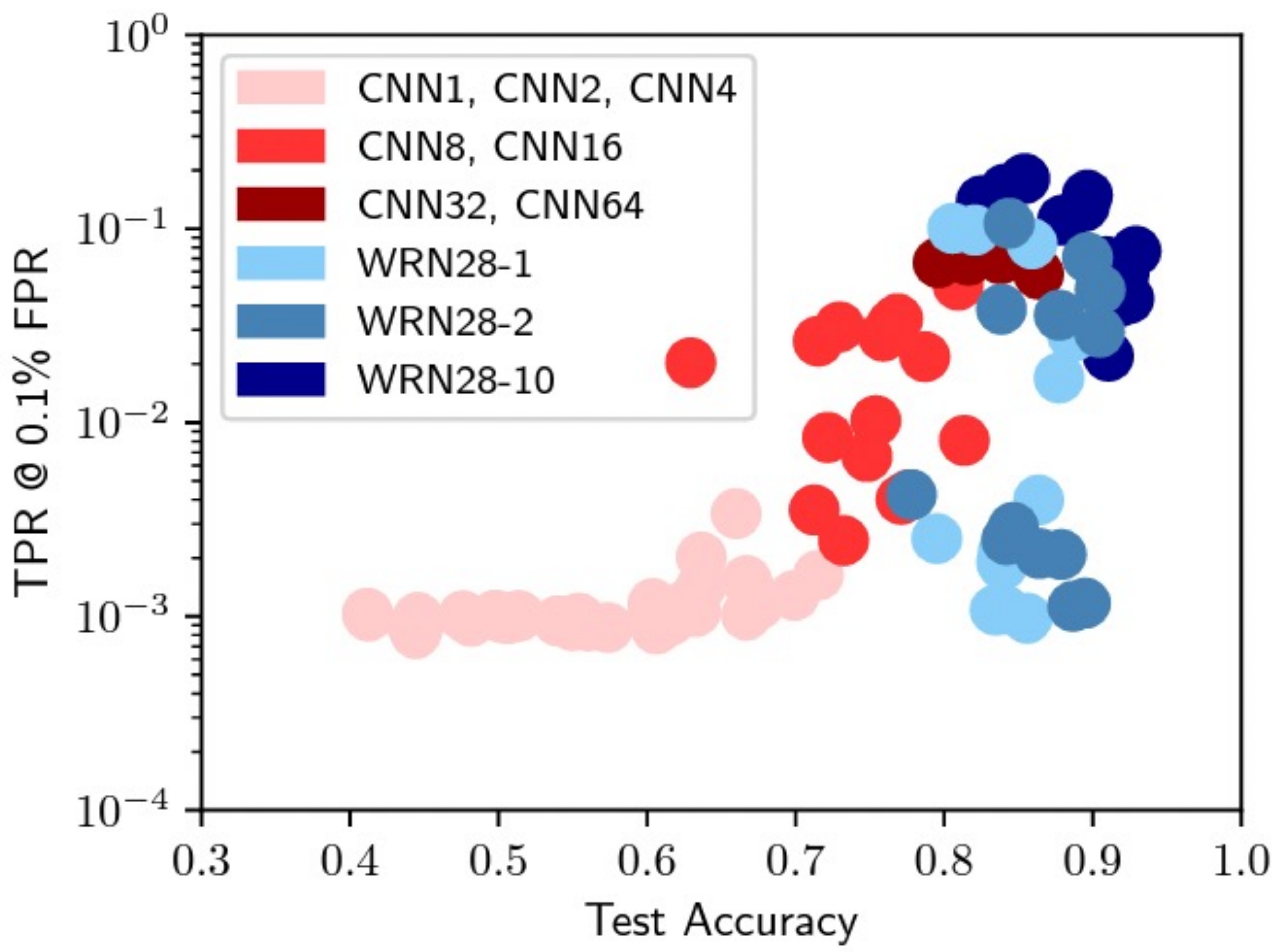
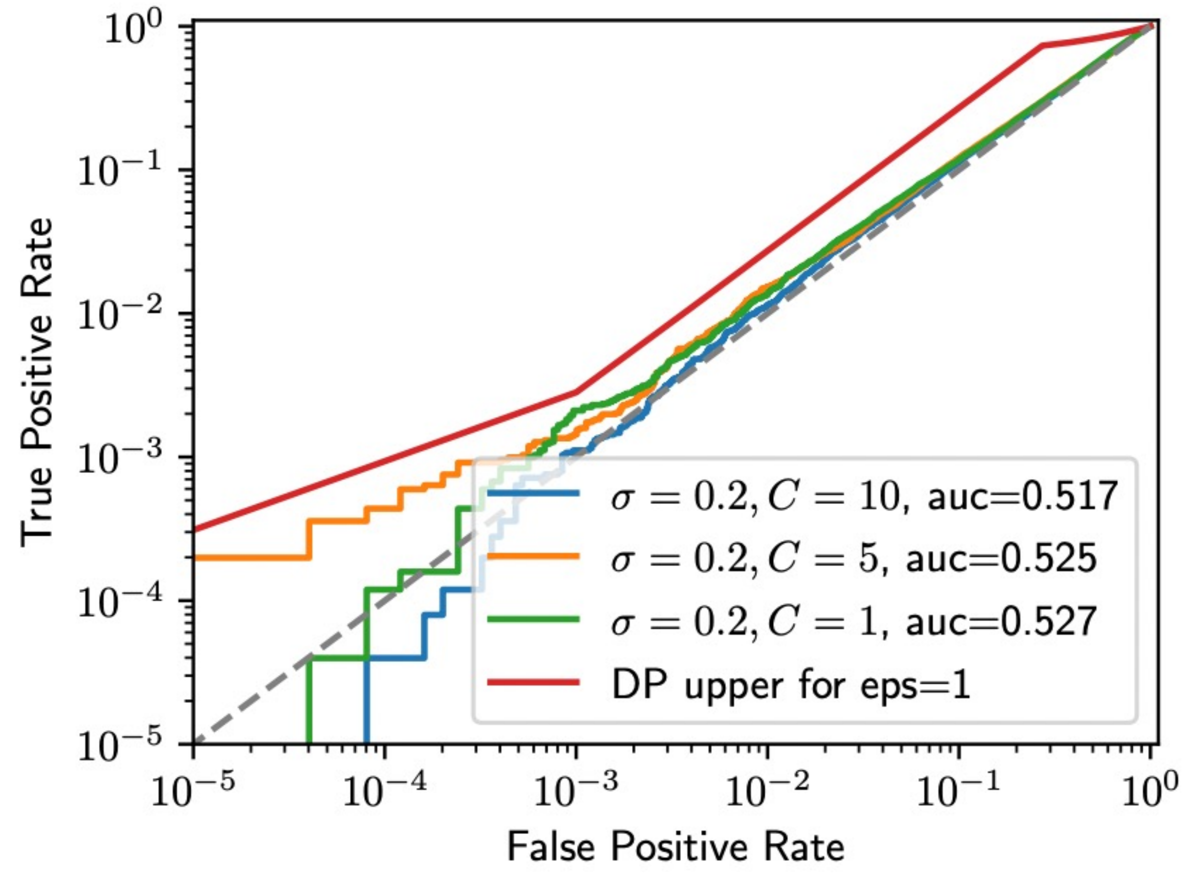


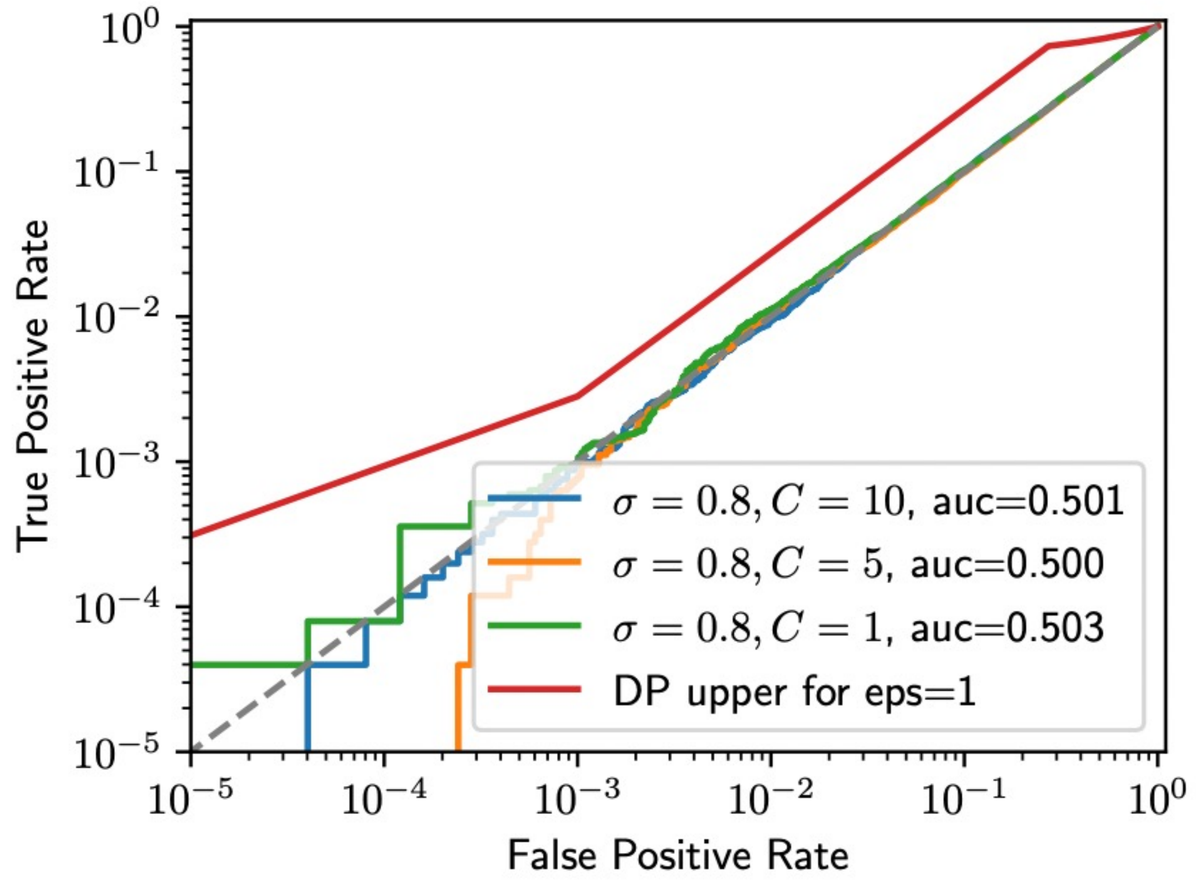
Fig. 16: Attack true-positive rate versus model test accuracy.

Attack Evaluation (DP-SGD)

- **Differentially Private SGD** is the main defense mechanism against MI attacks on machine learning models
    - DP gives an upper bound on the success of any MI attack
- Even when little noise is added, small clipping norms significantly reduces the performance of the attack



(b) $\epsilon > 5000$



(c) $\epsilon = 8$

Strengths

- Introduced new metrics for evaluating MI attack success
- Viewing membership inference as a **hypothesis test between IN and OUT loss distributions** can achieve much better true positive rates than prior work
- Several new ideas: learn per-sample thresholds, fit Gaussians to logit distribution
- Good attack performance
- Comprehensive experiments

Limitations

- The attack comes with a **sizable computational overhead**
- Some of the assumptions might not be true
  - There are no statistical tests performed to determine if the logits are Gaussians
- Why white-box attack does not perform better?