

NOTES

- Call for actions
 - 5/18,20 lectures will be offered as recordings
 - 5/25: Memorial Day – no class
 - 6/03: Final term-project presentation
 - 12-13 min presentation + 2-3 min Q&A
 - Presentation **MUST** cover:
 - 1-2 slides on your research motivation and goals
 - 1-2 slides on your hypothesis and evaluation design
 - 3-4 slides on your most interesting/promising results
 - 1 slide on your implications and conclusion
 - 6/8–10: Final exam will be online (Canvas) and available
 - 6/10: Homework 1 – 4 late submissions

AI 539: TRUSTWORTHY ML
PART IV – OVERVIEW AND MEMBERSHIP INFERENCE

Sanghyun Hong

sanghyun.hong@oregonstate.edu

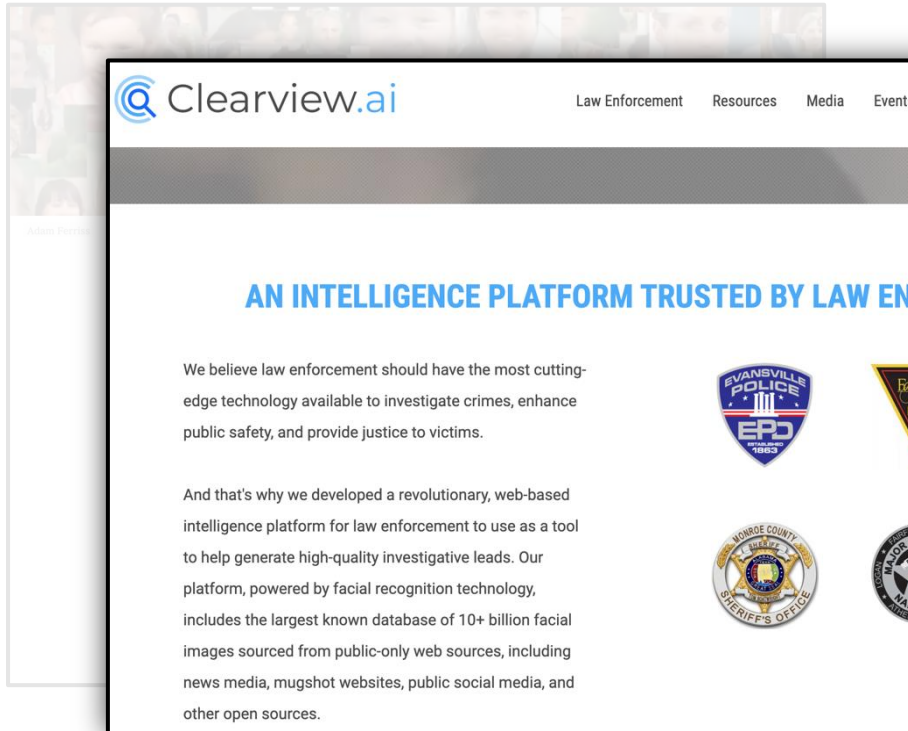


Oregon State
University



TRUE AI
Trustworthy and Responsible AI

WHY PRIVACY MATTERS?



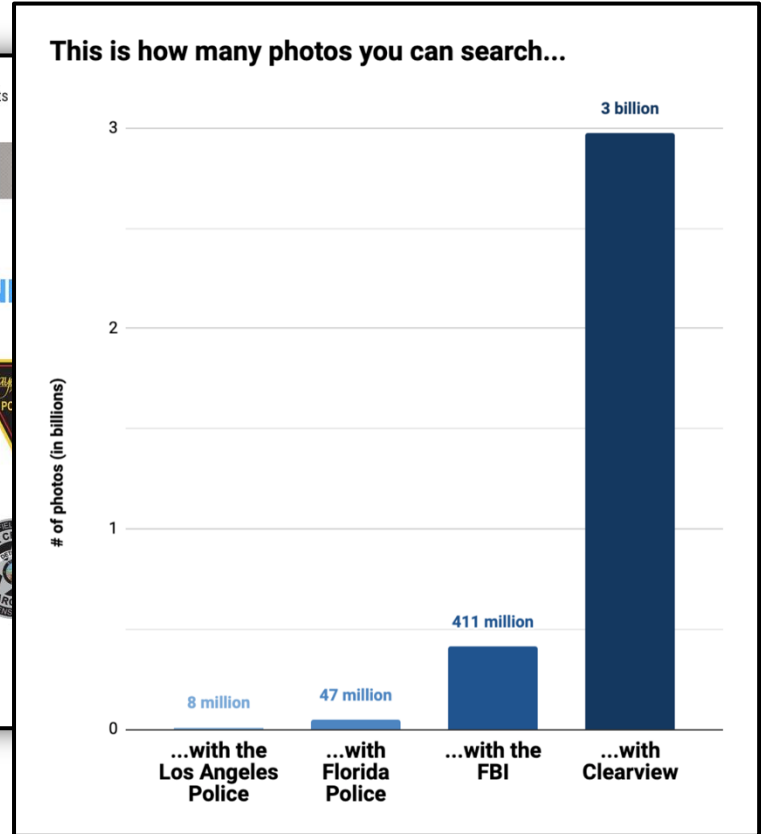

Clearview.ai

Law Enforcement Resources Media Events

AN INTELLIGENCE PLATFORM TRUSTED BY LAW ENFORCEMENT

We believe law enforcement should have the most cutting-edge technology available to investigate crimes, enhance public safety, and provide justice to victims.

And that's why we developed a revolutionary, web-based intelligence platform for law enforcement to use as a tool to help generate high-quality investigative leads. Our platform, powered by facial recognition technology, includes the largest known database of 10+ billion facial images sourced from public-only web sources, including news media, mugshot websites, public social media, and other open sources.



¹<https://www.nytimes.com/2020/01/18/technology/clearview-privacy-facial-recognition.html>

²<https://www.muckrock.com/news/archives/2020/jan/18/clearview-ai-facial-recognition-records/>

WHY PRIVACY MATTERS?

- Let's think about
 - What is privacy?
 - Our right to control information about ourselves
 - Context dependent: it should match the norms of the original context
 - In ML: models should not reveal any individual training records
 - What does privacy matter?
 - Your personal info matters: discrimination, identity theft
 - Chilling effect (e.g., no freedom of speech)
 - In ML: we cannot establish trust
 - How is it different from security?
 - Security is about “secure” – an absolute value
 - Privacy is about “norms” – context dependent, knobs matter

HOW CAN WE MAKE ML PRIVATE?

- Perfect, yet not interesting solution:
 - No learning
 - But this is *not* what we want

HOW CAN WE MAKE ML PRIVATE: DE-ANONYMIZATION

- Setup
 - **Attacker**: de-anonymize anonymized records
 - **Victim** : holds and releases anonymized sensitive data

- Knowledge
 - Auxiliary (side) information about the data

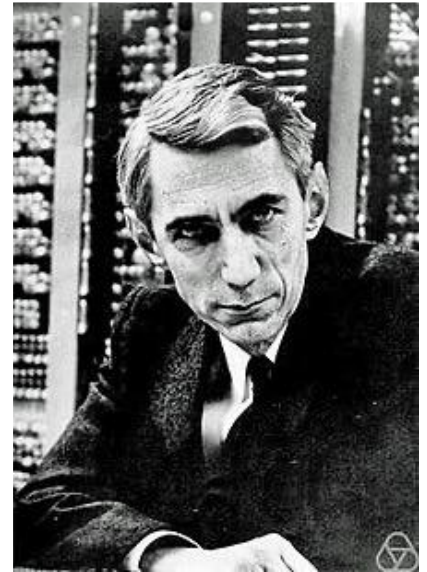
- Capability
 - Query your data with chosen inputs
 - Post-process query outputs q
 - Cross-reference with external data
 - ...

HOW CAN WE MAKE ML PRIVATE: DE-ANONYMIZATION

- In ML
 - Model trains on statistical patterns across records
 - Anonymizing individual fields (e.g., SSN) does not prevent leakage
 - Examples of sensitive data:
 - Cancer diagnoses (medical records)
 - Demographic attributes
 - Financial transactions
 - Location histories
 - ...
- Note:
 - “Anonymization of a data record might seem easy to implement. Unfortunately, it is *increasingly easy to defeat* anonymization by the very techniques that are being developed for many legitimate applications of big data.” [1]

HOW CAN WE MAKE ML PRIVATE: CRYPTO-BASED SOLUTIONS

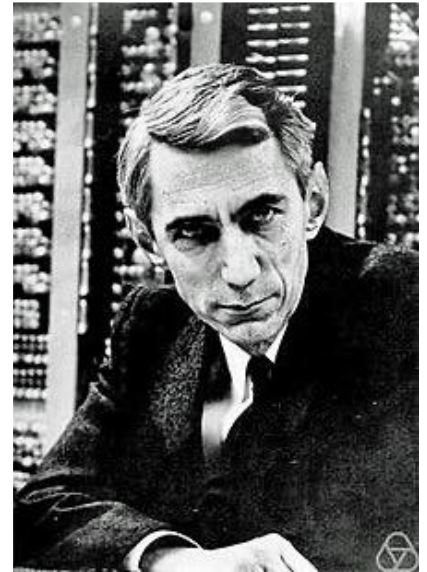
- Shannon's perfect security
 - An adversary should not distinguish a message M from a random text R



Claude Shannon (1916 ~ 2001)
A Father of Information Theory
and Modern Cryptography

HOW CAN WE MAKE ML PRIVATE: CRYPTO-BASED SOLUTIONS

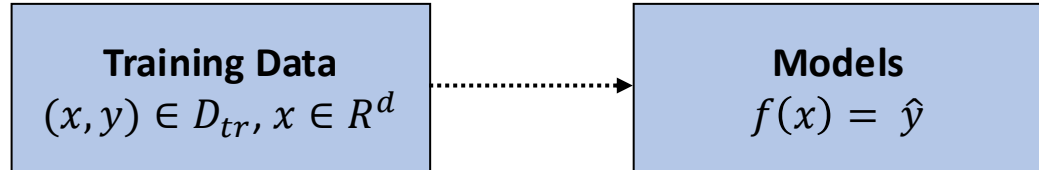
- Shannon's perfect security
 - An adversary should not distinguish a message M from a random text R
 - Formally:
 - $\Pr[M = m | C = c] = \Pr[M = m]$
 - where
 - m is a message (from a set M)
 - c is a ciphertext (from a set of all ciphertexts C)
 - $\Pr[C = c | M = m] = \Pr[C = c]$
 - It means:
 - Ciphertext provides no additional information
 - Observing c does not help with guessing $M = m$
 - c is independent of the message m



Claude Shannon (1916 ~ 2001)
A Father of Information Theory
and Modern Cryptography

HOW CAN WE MAKE ML PRIVATE: CRYPTO-BASED SOLUTIONS

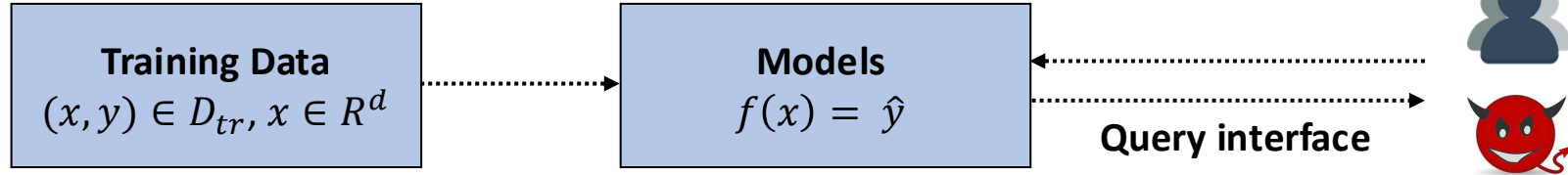
- Perfect security in model training



- Potential solutions:
 - **Encrypt-decrypt:** encrypt the training data and decrypt it to train a model
 - **Homomorphic encryption:** encrypt the training data and train a model on it
 - ...

PRIVACY IS ABOUT HOW TO SHAPE THE OUTPUTS

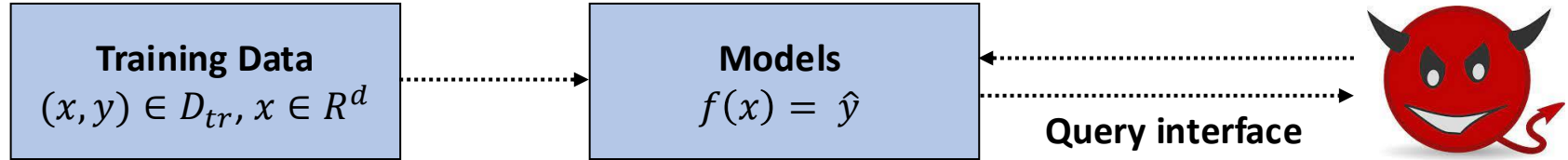
- Inferences with such model(s)



- Potential problems:
 - Prohibitively expensive (than standard training)
 - Restricted to key-holders only: once leaked, then done
 - Key-holders can still access private information through queries

WHAT AN ADVERSARY CAN DO WITH MODEL OUTPUTS?

- ML Pipeline

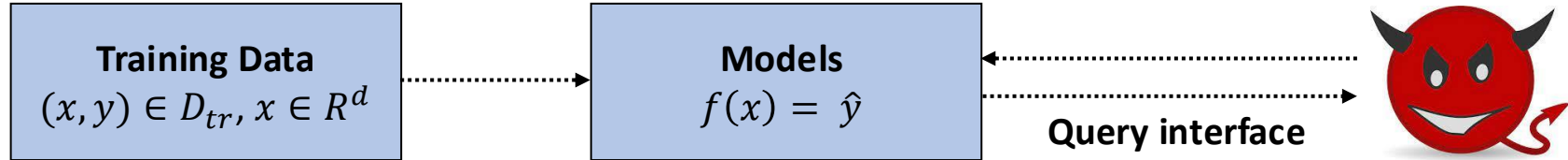


- Privacy risks

- Identify your membership in the training data
- Identify (sensitive) properties of your training data
- Identify (sensitive) attribute of a person that you know
- Reconstruct a sample completely
- Reconstruct a model behind the query interface
- ...

WHAT AN ADVERSARY CAN DO WITH MODEL OUTPUTS?

- ML Pipeline



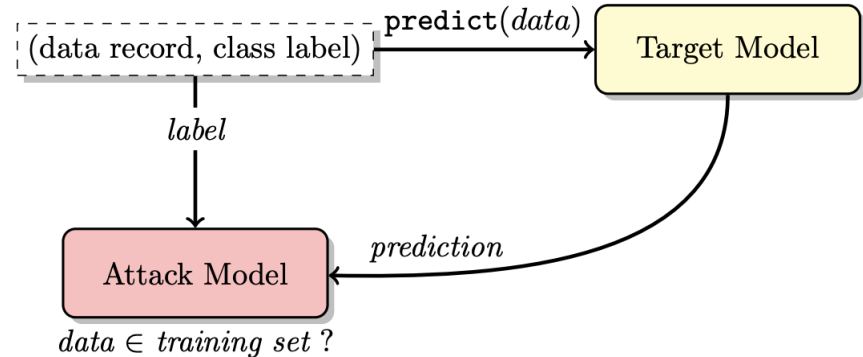
- Privacy threat models

- Tracing attack : Identify your membership in the training data
- Reconstruction : Identify (sensitive) properties of your training data
- De-anonymization: Identify (sensitive) attribute of a person that you know
- Reconstruction : Reconstruct a sample completely
- Reconstruction : Reconstruct a model behind the query interface
- ...

MEMBERSHIP INFERENCE ATTACKS

MEMBERSHIP INFERENCE ATTACKS

- Threat model
 - An adversary \mathcal{A} wants to know
 - if a sample $(x, y) \sim D$ is the member of
 - the training set S of an ML model f or not



MEMBERSHIP INFERENCE ATTACKS

- Threat model
 - Suppose
 - $(x, y) \sim D$; x is a set of features, y is a response
 - S is a training set drawn from D^n
 - A is a learning algorithm, l is the loss function
 - A_S is a model trained on S
 - \mathcal{A} is an adversary

MEMBERSHIP INFERENCE ATTACKS

- Threat model

- Suppose

- $(x, y) \sim D$; x is a set of features, y is a response
 - S is a training set drawn from D^n
 - A is a learning algorithm, l is the loss function
 - A_S is a model trained on S
 - \mathcal{A} is an adversary

- Membership experiment¹

- Sample $S \sim D^n$, and let $A_S = A(S)$
 - Choose $b \leftarrow \{0, 1\}$ *uniformly* at random
 - Draw $z \sim S$ if $b = 0$, or $z \sim D$ if $b = 1$
 - $\text{Exp}^M(\mathcal{A}, A, n, D)$ is 1 if $\mathcal{A}(z, A_S, n, D) = b$ and 0 otherwise. \mathcal{A} must output 0 or 1

MEMBERSHIP INFERENCE ATTACKS

- Threat model

- Membership experiment¹

- Sample $S \sim D^n$, and let $A_S = A(S)$
 - Choose $b \leftarrow \{0, 1\}$ *uniformly* at random
 - Draw $z \sim S$ if $b = 0$, or $z \sim D$ if $b = 1$
 - $\text{Exp}^M(\mathcal{A}, A, n, D)$ is 1 if $\mathcal{A}(z, A_S, n, D) = b$ and 0 otherwise. \mathcal{A} must output 0 or 1

- Membership advantage¹

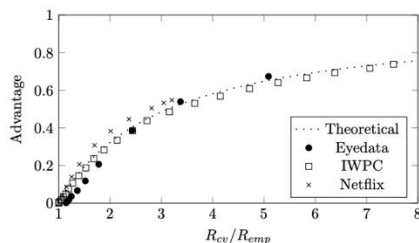
- $\text{Adv}^M(\mathcal{A}, A, n, D) = \Pr[\mathcal{A} = 0 | b = 0] - \Pr[\mathcal{A} = 0 | b = 1]$
 $= 2 \Pr[\text{Exp}^M(\mathcal{A}, A, n, D) = 1] - 1$

MEMBERSHIP INFERENCE ATTACKS

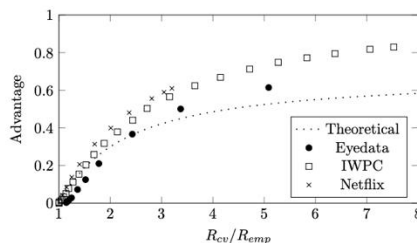
- Yeom *et al.* attack
 - \mathcal{A}_1 : Bounded loss function
 - Suppose the loss function is bounded on B
 - For $z = (x, y)$
 - The attacker returns 1 with the probability $l(A_S, z)/B$
 - Otherwise, the attacker outputs 0

MEMBERSHIP INFERENCE ATTACKS

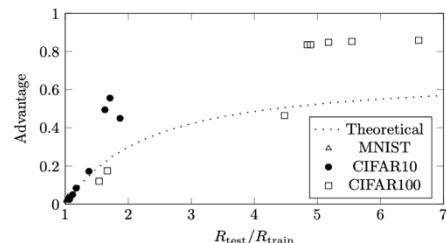
- Yeom *et al.* attack
 - \mathcal{A}_1 : Bounded loss function
 - Suppose the loss function is bounded on B
 - For $z = (x, y)$
 - The attacker returns 1 with the probability $l(A_S, z)/B$
 - Otherwise, the attacker outputs 0
 - (Theorem 2) \mathcal{A}_1 's advantage is $R_{\text{gen}}(A)/B$



(a) Regression and tree models assuming knowledge of σ_S and σ_D .



(b) Regression and tree models assuming knowledge of σ_S only.



(c) Deep CNNs assuming knowledge of average training loss L_S .

MEMBERSHIP INFERENCE ATTACKS

- Yeom *et al.* attack
 - \mathcal{A}_1 : Bounded loss function
 - Suppose the loss function is bounded on B
 - For $z = (x, y)$
 - The attacker returns 1 with the probability $l(A_s, z)/B$
 - Otherwise, the attacker outputs 0
 - \mathcal{A}_2 : Threshold
 - Suppose the attacker knows
 - The conditional probability density functions of the error
 - $f(\epsilon | b = 0)$ and $f(\epsilon | b = 1)$
 - such as the avg. loss over the training data (and over the test data)
 - For $z = (x, y)$
 - Let $\epsilon = y - A_s(x)$
 - The attacker outputs $\operatorname{argmax}_{b \in \{0,1\}} f(\epsilon | b)$

MEMBERSHIP INFERENCE ATTACKS

- Evaluation

	Our work	Shokri et al. [7]
Attack complexity	Makes only one query to the model	Must train hundreds of shadow models
Required knowledge	Average training loss L_S	Ability to train shadow models, e.g., input distribution and type of model
Precision	0.505 (MNIST) 0.694 (CIFAR-10) 0.874 (CIFAR-100)	0.517 (MNIST) 0.72-0.74 (CIFAR-10) > 0.99 (CIFAR-100)
Recall	> 0.99	> 0.99

Table 1: Comparison of our membership inference attack with that presented by Shokri et al. While our attack has slightly lower precision, it requires far less computational resources and background knowledge.

HOW CAN WE ENHANCE THE THRESHOLD ATTACK?

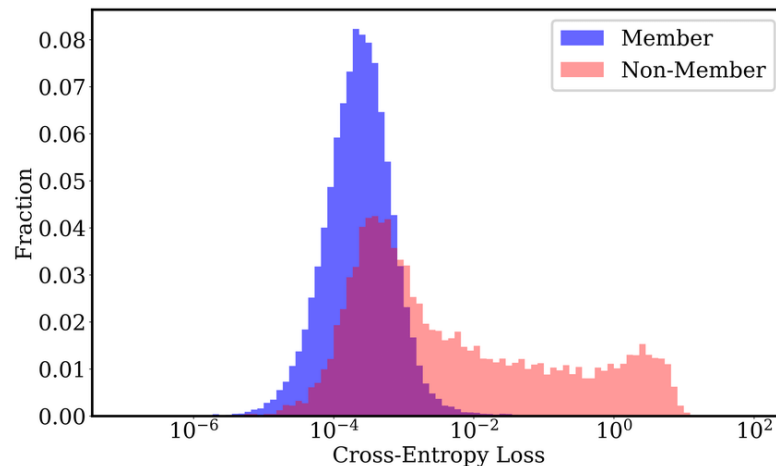
MEMBERSHIP INFERENCE ATTACKS AGAINST MACHINE LEARNING MODELS, SHOKRI ET AL., OAKLAND 2017

REVISITING YEOM ET AL. ATTACK

- Yeom *et al.* attack
 - \mathcal{A}_1 : Bounded loss function
 - Suppose the loss function is bounded on B
 - For $z = (x, y)$
 - The attacker returns 1 with the probability $l(A_s, z)/B$
 - Otherwise, the attacker outputs 0
 - \mathcal{A}_2 : Threshold
 - Suppose the attacker knows
 - The conditional probability density functions of the error
 - $f(\epsilon | b = 0)$ and $f(\epsilon | b = 1)$
 - such as the avg. loss over the training data (and over the test data)
 - For $z = (x, y)$
 - Let $\epsilon = y - A_s(x)$
 - The attacker outputs $\operatorname{argmax}_{b \in \{0,1\}} f(\epsilon | b)$

REVISITING YEOM ET AL. ATTACK

- Yeom *et al.* attack
 - \mathcal{A}_2 : Threshold
 - Suppose the attacker knows
 - The conditional probability density functions of the error
 - $f(\epsilon | b = 0)$ and $f(\epsilon | b = 1)$
 - such as the avg. loss over the training data (and over the test data)
 - For $z = (x, y)$
 - Let $\epsilon = y - A_S(x)$
 - The attacker outputs $\operatorname{argmax}_{b \in \{0,1\}} f(\epsilon | b)$
- Challenge:
 - How to compute an optimal threshold?



MEMBERSHIP INFERENCE ATTACKS

- Shokri *et al.* attack

- Key idea: shadow models

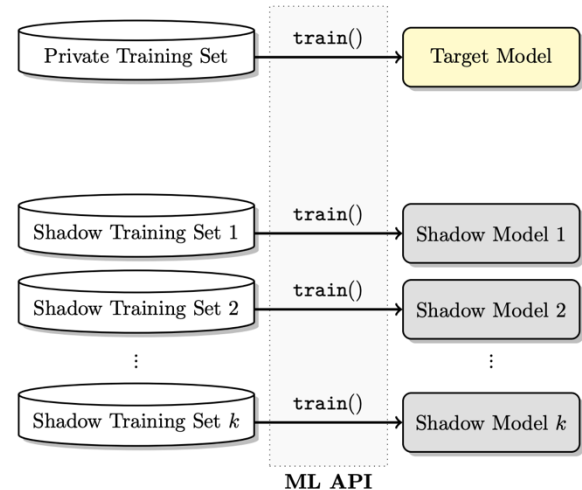
- The attacker has some data samples from D
 - If the attacker trains models with those samples, we know their memberships!
 - If shadow models are trained similarity, we can exploit the membership info.!

- Attacker's data:

- Know the labeled records: (x, y)
 - Query them to the target model and collect its predictions: $((x, y), \hat{y})$

- How to train?

- Create a train and test split
 - Use the *train* data to train the shadow models



MEMBERSHIP INFERENCE ATTACKS

- Shokri *et al.* attack
 - What if the attacker does not have data?
 - (x, y) from a distribution like the victim's...
 - Data generation strategies:
 - Model-based synthesis
 - Statistics-based synthesis
 - Noisy real-data

Algorithm 1 Data synthesis using the target model

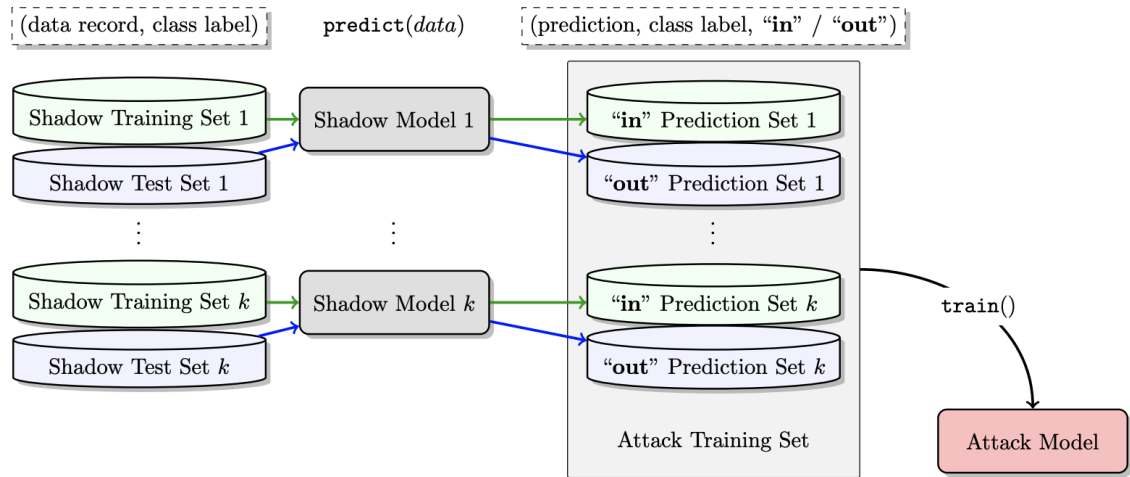
```
1: procedure SYNTHESIZE(class : c)
2:    $\mathbf{x} \leftarrow \text{RANDRECORD}()$   $\triangleright$  initialize a record randomly
3:    $y_c^* \leftarrow 0$ 
4:    $j \leftarrow 0$ 
5:    $k \leftarrow k_{max}$ 
6:   for iteration = 1  $\dots$  itermax do
7:      $\mathbf{y} \leftarrow f_{target}(\mathbf{x})$   $\triangleright$  query the target model
8:     if  $y_c \geq y_c^*$  then  $\triangleright$  accept the record
9:       if  $y_c > \text{conf}_{min}$  and  $c = \arg \max(\mathbf{y})$  then
10:        if  $\text{rand}() < y_c$  then  $\triangleright$  sample
11:          return  $\mathbf{x}$   $\triangleright$  synthetic data
12:        end if
13:      end if
14:       $\mathbf{x}^* \leftarrow \mathbf{x}$ 
15:       $y_c^* \leftarrow y_c$ 
16:       $j \leftarrow 0$ 
17:    else
18:       $j \leftarrow j + 1$ 
19:      if  $j > rej_{max}$  then  $\triangleright$  many consecutive rejects
20:         $k \leftarrow \max(k_{min}, \lceil k/2 \rceil)$ 
21:         $j \leftarrow 0$ 
22:      end if
23:    end if
24:     $\mathbf{x} \leftarrow \text{RANDRECORD}(\mathbf{x}^*, k)$   $\triangleright$  randomize  $k$  features
25:  end for
26:  return  $\perp$   $\triangleright$  failed to synthesize
27: end procedure
```

MEMBERSHIP INFERENCE ATTACKS

- Shokri *et al.* attack

- Attack model

- Data format $((x, y), \hat{y})$
 - Some of them are “IN” the shadow train, otherwise “OUT”
 - Combine three info. $(y, \hat{y}, \mathbf{IN})$ or $(y, \hat{y}, \mathbf{OUT})$
 - Make the attack model predict **IN** or **OUT**

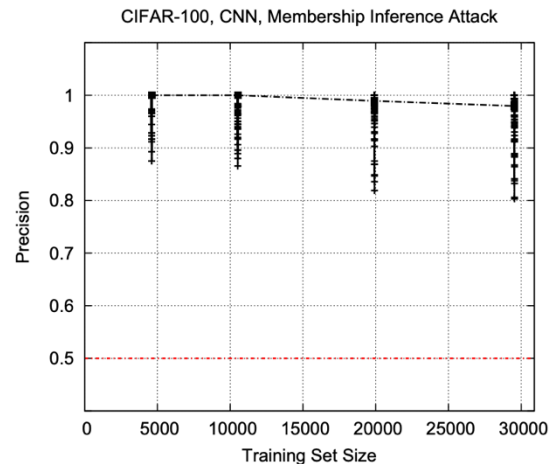
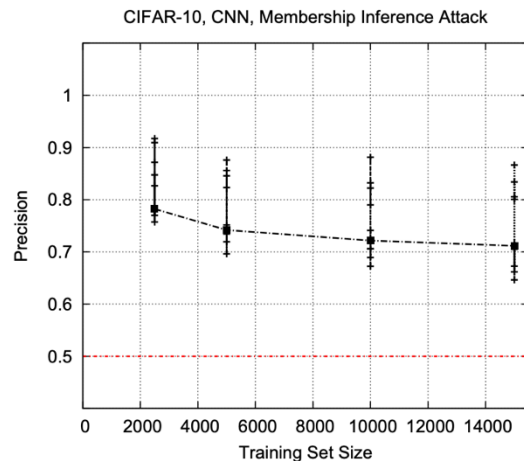
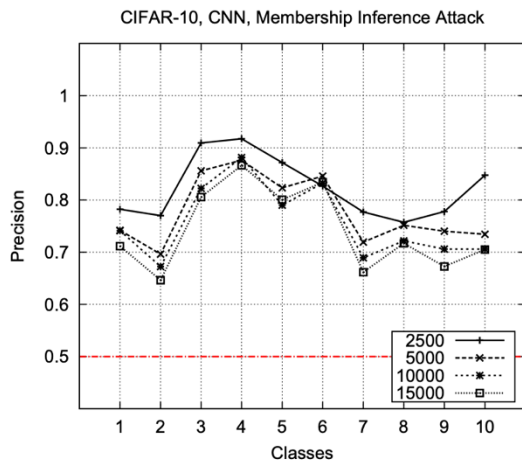


EVALUATION

- Setup
 - Datasets:
 - MNIST | CIFAR-10/100
 - Purchases | Locations | Texas-100 | UCI Adult
 - Models
 - MLaaS: Google Prediction API | Amazon ML | NNs
 - MI Attack
 - Shadow models: 20 – 100 models
 - Defenses
 - Heuristics: Top-k | Precision | Regularization
 - [?!] In theory: DP

EVALUATION

- MI Attacks on CIFAR
 - Shadow models: 100
 - Training set (for targets):
 - CIFAR-10: {2.5, 5, 10, 15}k samples
 - CIFAR-100: {4.5, 10, 20, 30}k samples
 - **In-short:** MI attacks work with a pretty reasonable acc.



EVALUATION

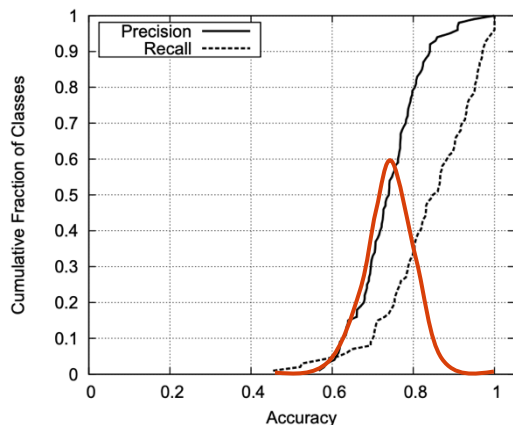
- MI Attacks w. Different Models

- Dataset: Purchase-100
- Models (trained on 10k records):
 - Amazon ML
 - Google's Prediction API

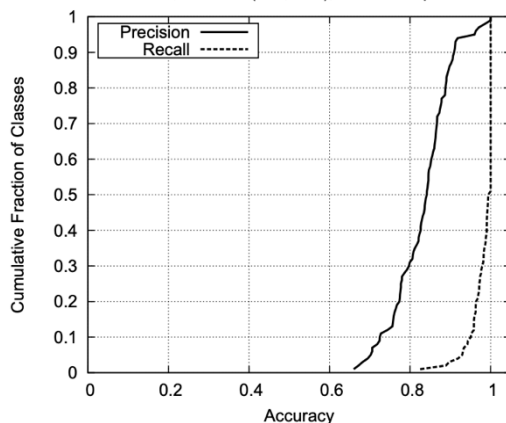
– **In-short:** across all models, MI attacks work with a pretty reasonable acc.

<i>ML Platform</i>	<i>Training</i>	<i>Test</i>
Google	0.999	0.656
Amazon (10,1e-6)	0.941	0.468
Amazon (100,1e-4)	1.00	0.504
Neural network	0.830	0.670

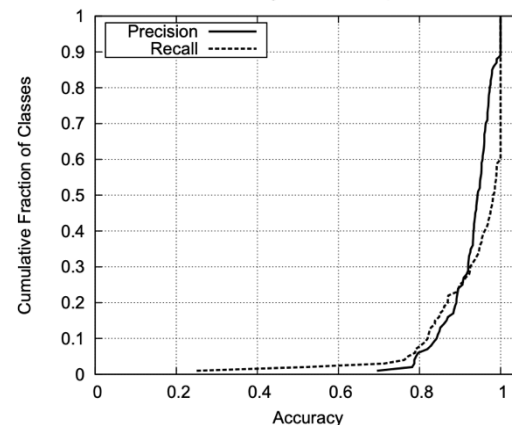
Purchase Dataset, Amazon (10,1e-6), Membership Inference Attack



Purchase Dataset, Amazon (100,1e-4), Membership Inference Attack

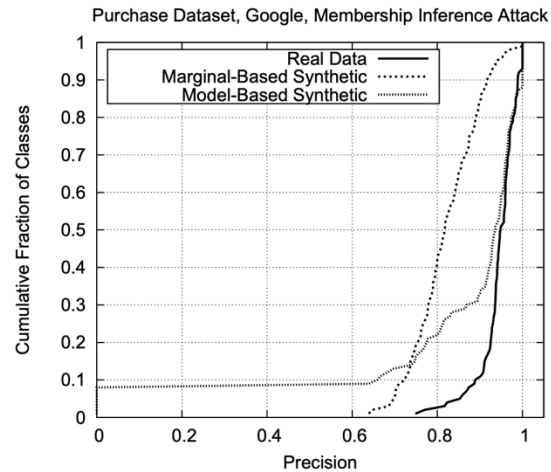
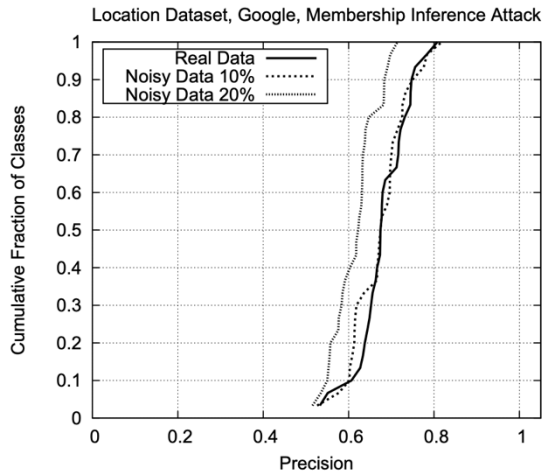


Purchase Dataset, Google, Membership Inference Attack



EVALUATION

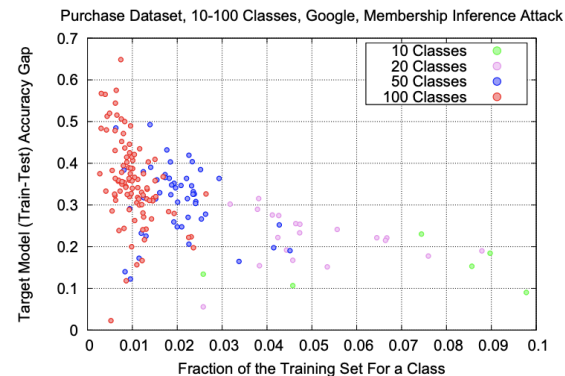
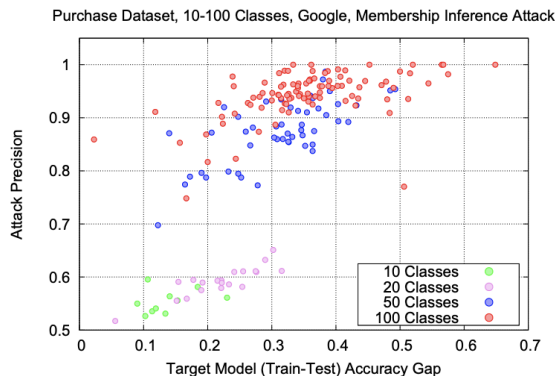
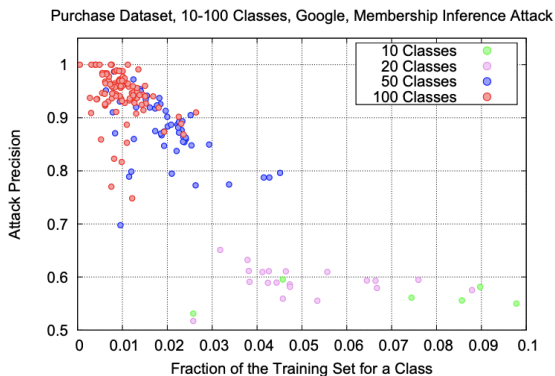
- MI Attacks w. Different Shadow Models
 - Dataset: Location
 - Modification:
 - Noisy shadow training data
 - No data (synthesize it!)
 - **In-short:** MI attacks show robust acc. under the weak approximation of the dist.



EVALUATION

- MI Attacks w. Different # classes
 - Dataset: Purchase
 - Modification:
 - # Classes: 10 – 100 classes (keep $N(D_{tr})$ the same)
 - Google Prediction API
 - **In-short:** More supporting data samples in the c

<i>Dataset</i>	<i>Training Accuracy</i>	<i>Testing Accuracy</i>	<i>Attack Precision</i>
Adult	0.848	0.842	0.503
MNIST	0.984	0.928	0.517
Location	1.000	0.673	0.678
Purchase (2)	0.999	0.984	0.505
Purchase (10)	0.999	0.866	0.550
Purchase (20)	1.000	0.781	0.590
Purchase (50)	1.000	0.693	0.860
Purchase (100)	0.999	0.659	0.935
TX hospital stays	0.668	0.517	0.657



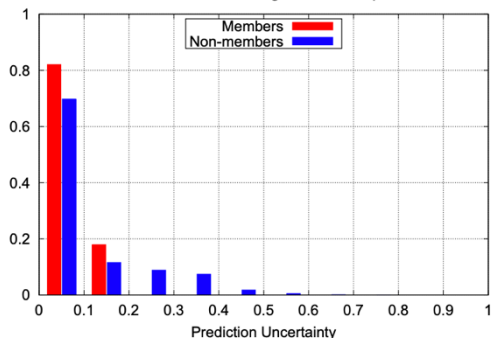
EVALUATION

- MI Attacks, Why Do They Work?
 - Dataset: Purchase
 - Modification:
 - # Classes: 10 – 100 classes (keep $N(D_{tr})$ the same)
 - Google Prediction API
 - **In-short:** It may depend on a model's ability to distinguish members and non-members

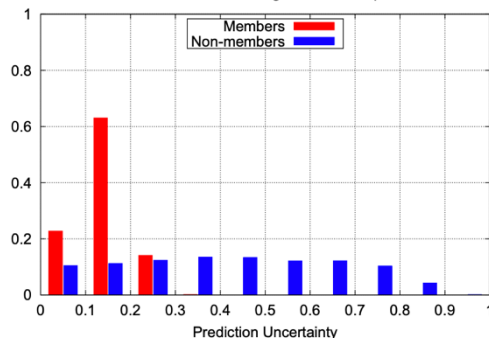
EVALUATION

- MI Attacks, Why Do They Work?

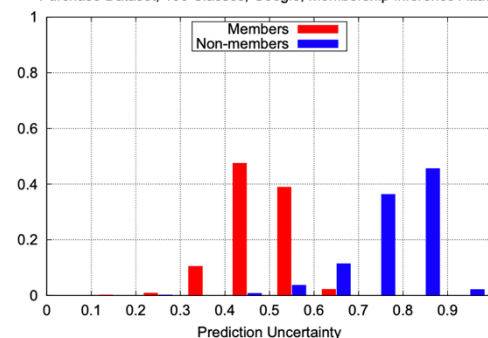
Purchase Dataset, 10 Classes, Google, Membership Inference Attack



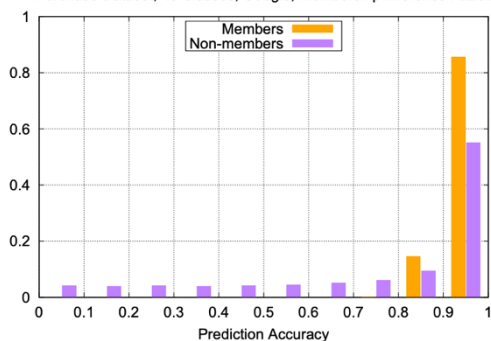
Purchase Dataset, 20 Classes, Google, Membership Inference Attack



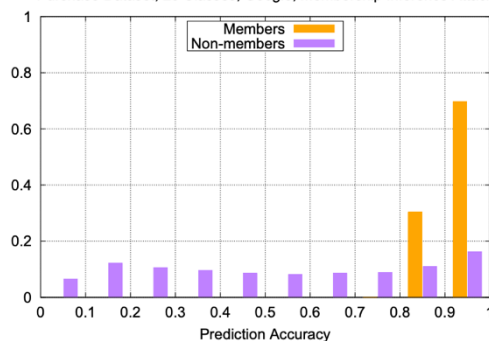
Purchase Dataset, 100 Classes, Google, Membership Inference Attack



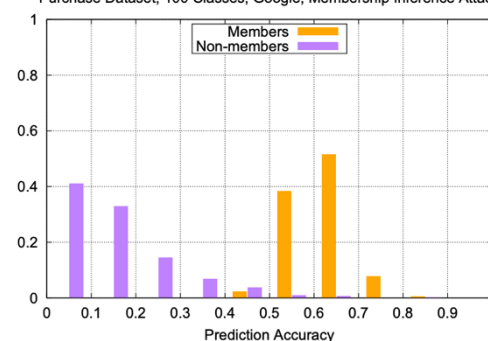
Purchase Dataset, 10 Classes, Google, Membership Inference Attack



Purchase Dataset, 20 Classes, Google, Membership Inference Attack



Purchase Dataset, 100 Classes, Google, Membership Inference Attack



EVALUATION

- Defenses

- Top-k
- Precision (round-ups)
- Regularization (L_2)

- Results (on NNs)

- Still MI attack works
 - in $k = 1$ (label)
 - with less precision ($d = 1$)
- Regularization somewhat effective but care must be taken for a model's acc.

Purchase dataset	<i>Testing Accuracy</i>	<i>Attack Total Accuracy</i>	<i>Attack Precision</i>	<i>Attack Recall</i>
No Mitigation	0.66	0.92	0.87	1.00
Top $k = 3$	0.66	0.92	0.87	0.99
Top $k = 1$	0.66	0.89	0.83	1.00
Top $k = 1$ label	0.66	0.66	0.60	0.99
Rounding $d = 3$	0.66	0.92	0.87	0.99
Rounding $d = 1$	0.66	0.89	0.83	1.00
Temperature $t = 5$	0.66	0.88	0.86	0.93
Temperature $t = 20$	0.66	0.84	0.83	0.86
L2 $\lambda = 1e - 4$	0.68	0.87	0.81	0.96
L2 $\lambda = 1e - 3$	0.72	0.77	0.73	0.86
L2 $\lambda = 1e - 2$	0.63	0.53	0.54	0.52

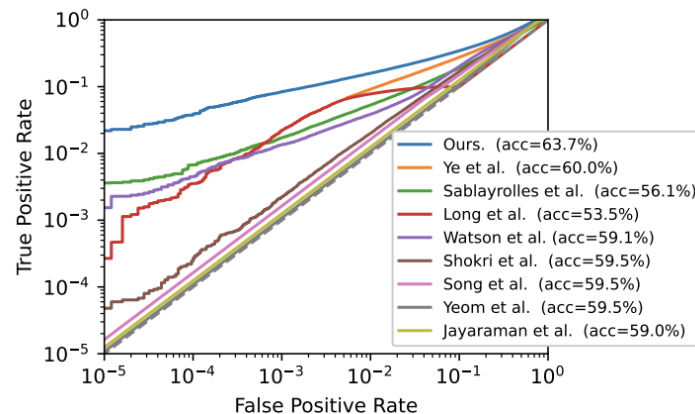
Hospital dataset	<i>Testing Accuracy</i>	<i>Attack Total Accuracy</i>	<i>Attack Precision</i>	<i>Attack Recall</i>
No Mitigation	0.55	0.83	0.77	0.95
Top $k = 3$	0.55	0.83	0.77	0.95
Top $k = 1$	0.55	0.82	0.76	0.95
Top $k = 1$ label	0.55	0.73	0.67	0.93
Rounding $d = 3$	0.55	0.83	0.77	0.95
Rounding $d = 1$	0.55	0.81	0.75	0.96
Temperature $t = 5$	0.55	0.79	0.77	0.83
Temperature $t = 20$	0.55	0.76	0.76	0.76
L2 $\lambda = 1e - 4$	0.56	0.80	0.74	0.92
L2 $\lambda = 5e - 4$	0.57	0.73	0.69	0.86
L2 $\lambda = 1e - 3$	0.56	0.66	0.64	0.73
L2 $\lambda = 5e - 3$	0.35	0.52	0.52	0.53

HOW SHOULD WE MEASURE MEMBERSHIP INFERENCE SUCCESS?

MEMBERSHIP INFERENCE ATTACKS FROM FIRST PRINCIPLE, CALINI ET AL., OAKLAND 2022

REVISITING YEOM ET AL. AND SHOKRI ET AL. ATTACK

- Metrics for measuring the attack success
 - Membership advantage (Yeom et al.)
 - Precision (Shokri et al.)
 - AUROC (Jayaraman et al.)
 - ...



REVISITING YEOM ET AL. AND SHOKRI ET AL. ATTACK

- Metrics for measuring the attack success
 - Problem of existing metrics
 - Symmetric: equal cost to false-positives and false-negatives
 - Average-case metric: often in security, we are interested in a certain subset
 - LOSS attack
 - Metrics:
 - Membership advantage
 - Precision
 - AUROC
 - Problem: perform **at random at low-FPR**

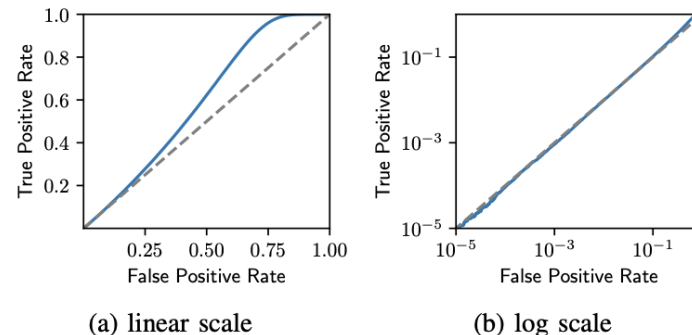


Fig. 2: ROC curve for the LOSS baseline membership inference attack, shown with both linear scaling (left), also and log-log scaling (right) to emphasize the low-FPR regime.

REVISITING YEOM ET AL. AND SHOKRI ET AL. ATTACK

- Metrics for measuring the attack success
 - Problem of existing metrics
 - Symmetric: equal cost to false-positives and false-negatives
 - Average-case metric: often in security, we are interested in a certain subset
 - LOSS attack
 - Metrics: membership advantage or precision
 - Problem: perform at random at low-FPR

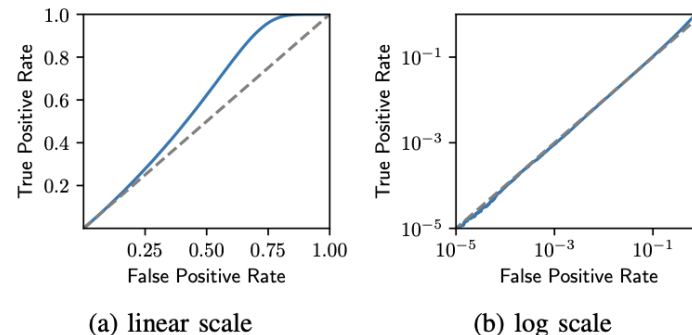


Fig. 2: ROC curve for the LOSS baseline membership inference attack, shown with both linear scaling (left), also and log-log scaling (right) to emphasize the low-FPR regime.

MEMBERSHIP INFERENCE ATTACK

- LiRA (The likelihood ratio attack)
 - Per-sample hardness score
 - Not all examples are equal
 - Some samples are easier to fit
 - Some samples have a larger separability
 - It does not matter if it is an inlier or outlier

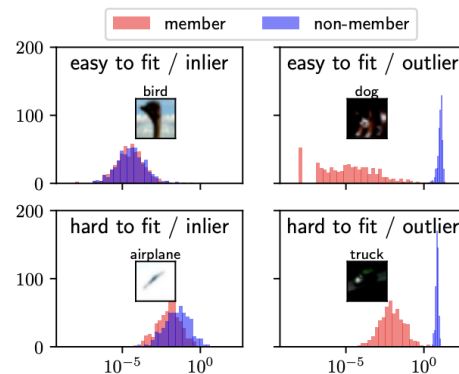


Fig. 3: Some examples are easier to fit than others, and some have a larger separability between their losses when being a member of the training set or not. We train 1024 models on random subsets of CIFAR-10 and plot the losses for four examples when the example is a member of the training set ($\hat{Q}_{in}(x, y)$, in red) or not ($\hat{Q}_{out}(x, y)$, in blue).

MEMBERSHIP INFERENCE ATTACK

- LiRA (The likelihood ratio attack)
 - Per-sample hardness score
 - Not all examples are equal
 - Some samples are easier to fit
 - Some samples have a larger separability
 - It does not matter if it is an inlier or outlier
 - Proposed attack
 - Compute per-sample hardness scores
 - Use parametric modeling

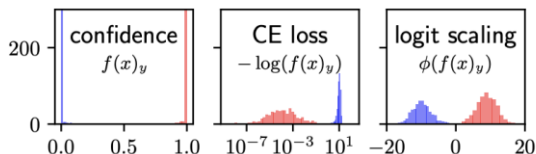


Fig. 4: The model’s confidence, or its logarithm (the cross-entropy loss) are not normally distributed. Applying the logit function yields values that are approximately normal.

Algorithm 1 Our online Likelihood Ratio Attack (LiRA). We train shadow models on datasets with and without the target example, estimate mean and variance of the loss distributions, and compute a likelihood ratio test. (In our **offline** variant, we omit lines 5, 6, 10, and 12, and instead return the prediction by estimating a single-tailed distribution, as is shown in Equation (4).)

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: $D_{\text{attack}} \leftarrow \mathbb{S} \mathbb{D}$ ▷ Sample a shadow dataset
- 5: $f_{\text{in}} \leftarrow \mathcal{T}(D_{\text{attack}} \cup \{(x, y)\})$ ▷ train IN model
- 6: $\text{confs}_{\text{in}} \leftarrow \text{confs}_{\text{in}} \cup \{\phi(f_{\text{in}}(x)_y)\}$
- 7: $f_{\text{out}} \leftarrow \mathcal{T}(D_{\text{attack}} \setminus \{(x, y)\})$ ▷ train OUT model
- 8: $\text{confs}_{\text{out}} \leftarrow \text{confs}_{\text{out}} \cup \{\phi(f_{\text{out}}(x)_y)\}$
- 9: **end for**
- 10: $\mu_{\text{in}} \leftarrow \text{mean}(\text{confs}_{\text{in}})$
- 11: $\mu_{\text{out}} \leftarrow \text{mean}(\text{confs}_{\text{out}})$
- 12: $\sigma_{\text{in}}^2 \leftarrow \text{var}(\text{confs}_{\text{in}})$
- 13: $\sigma_{\text{out}}^2 \leftarrow \text{var}(\text{confs}_{\text{out}})$
- 14: $\text{conf}_{\text{obs}} = \phi(f(x)_y)$ ▷ query target model
- 15: **return** $\Lambda = \frac{p(\text{conf}_{\text{obs}} \mid \mathcal{N}(\mu_{\text{in}}, \sigma_{\text{in}}^2))}{p(\text{conf}_{\text{obs}} \mid \mathcal{N}(\mu_{\text{out}}, \sigma_{\text{out}}^2))}$

EVALUATION

- Setup
 - Datasets: CIFAR-10, CIFAR-100, ImageNet and WikiText
 - Models
 - Wide-ResNet (CIFAR-10 and -100)
 - ResNet-50 (ImageNet)
 - GPT-2 small (WikiText)
 - LiRA setup
 - Shadow models: 65 for ImageNet and 256 for others
 - Repeat the attack 10 times
 - Metric
 - TPR at 1% FPR
 - ROC curve

EVALUATION

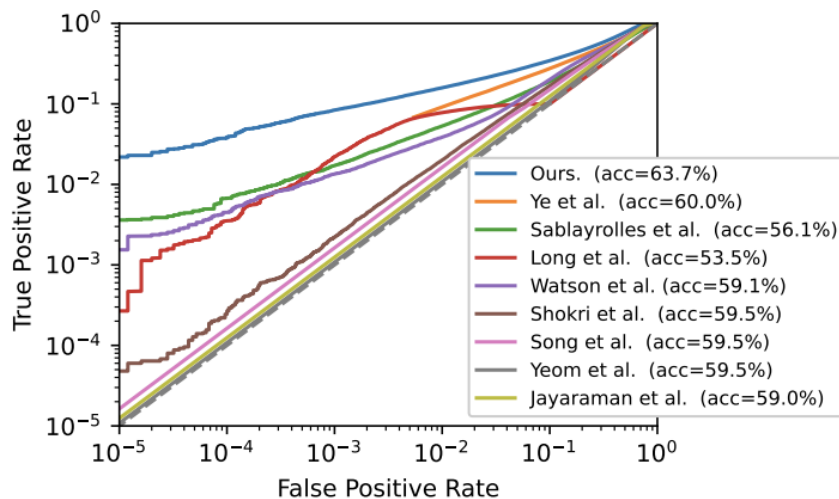
- LiRA (online) attack vs others

Method	shadow models	multiple queries	class hardness	example hardness	TPR @ 0.001% FPR			TPR @ 0.1% FPR			Balanced Accuracy		
					C-10	C-100	WT103	C-10	C-100	WT103	C-10	C-100	WT103
Yeom et al. [70]	○	○	○	○	0.0%	0.0%	0.00%	0.0%	0.0%	0.1%	59.4%	78.0%	50.0%
Shokri et al. [60]	●	○	●	○	0.0%	0.0%	–	0.3%	1.6%	–	59.6%	74.5%	–
Jayaraman et al. [25]	○	●	○	○	0.0%	0.0%	–	0.0%	0.0%	–	59.4%	76.9%	–
Song and Mittal [61]	●	○	●	○	0.0%	0.0%	–	0.1%	1.4%	–	59.5%	77.3%	–
Sablayrolles et al. [56]	●	○	●	●	0.1%	0.8%	0.01%	1.7%	7.4%	1.0%	56.3%	69.1%	65.7%
Long et al. [37]	●	○	●	●	0.0%	0.0%	–	2.2%	4.7%	–	53.5%	54.5%	–
Watson et al. [68]	●	○	●	●	0.1%	0.9%	0.02%	1.3%	5.4%	1.1%	59.1%	70.1%	65.4%
Ye et al. [69]	●	○	●	●	–	–	–	–	–	–	60.3%	76.9%	65.5%
Ours	●	●	●	●	2.2%	11.2%	0.09%	8.4%	27.6%	1.4%	63.8%	82.6%	65.6%

TABLE I: **Comparison of prior membership inference attacks** under the same settings for well-generalizing models on CIFAR-10, CIFAR-100, and WikiText-103 using 256 shadow models. Accuracy is only presented for completeness; we do not believe this is a meaningful metric for evaluating membership inference attacks. Full ROC curves are presented in Appendix A.

EVALUATION

- LiRA (online) attack vs others
 - 10x more successful than the prior attacks at the low-FPR region (0.001 - 0.1 FPR)



EVALUATION

- LiRA (online) attack and the generalization gap
 - Overfitted models tend to be vulnerable to the attack
 - There are models with the identical gaps 100x times vulnerable
 - More accurate models are more vulnerable to the attack

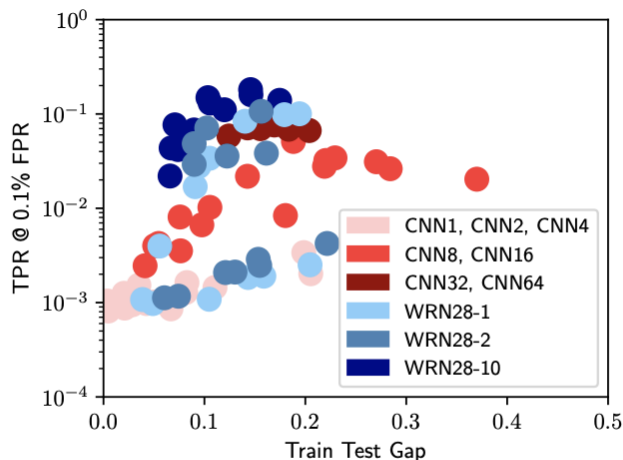
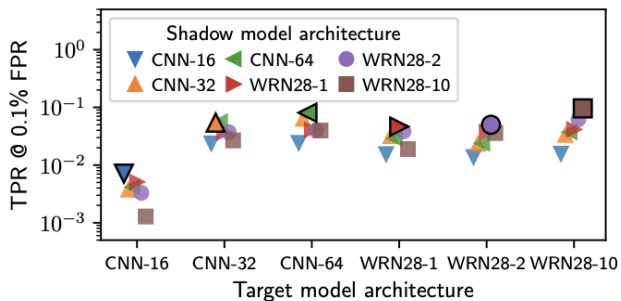


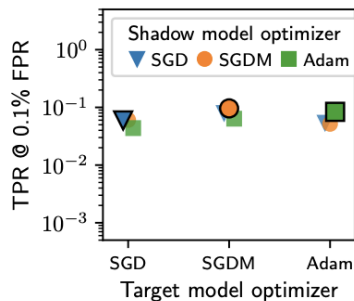
Fig. 7: Attack true-positive rate versus model train-test gap for a variety of CIFAR-10 models.

EVALUATION

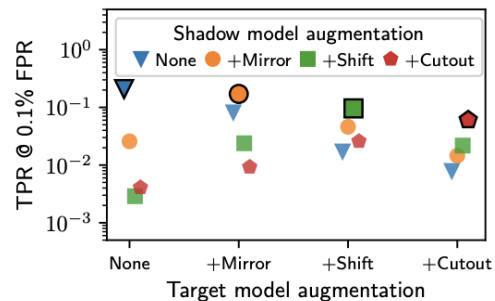
- LiRA (online) attack with different settings
 - While the training configurations are different from shadow models to the target
 - LiRA attack performs consistently; the attack is agnostic to the training setups



(a) Vary model architecture.



(b) Vary training optimizer.



(c) Vary data augmentation.

Fig. 11: Our attack succeeds when the adversary is uncertain of the target model's training setup. We vary the target model's architecture (a), the training optimizer (b) and the data augmentation (c), as well as the adversary's guess of each of these properties when training shadow models. The attack performs best when the adversary guesses correctly (black-lined markers).

Thank You!

Sanghyun Hong

<https://secure-ai.systems/courses/MLSec/current>



Oregon State
University



TRUE AI
Trustworthy and Responsible AI