ON THE PRIVACY RISKS OF ALGORITHMIC RECOUSE

Pawelczyk, Lakkaraju, Neel

Presented by Christina Xiao, Lucas Monteiro Pas, Catherine Huang
MOTIVATION

Gradient Based

Lime

Counterfactuals
MOTIVATION

Can these methods leak:

1 - User’s sensitive information?
2 - Model’s weights?
3 - Training dataset?
Motivation

Can these methods leak:

1 - User’s sensitive information?
2 - Model’s weights?
3 - Training dataset?

Counterfactuals
PREVIOUS WORKS

Membership Inference.

Given access to an instance + Loss information = Instance is in training
Membership Inference.

Given access to an instance, how to leverage XAI?

Given access to an instance + Feature attribution = Instance is in training

PREVIOUS WORKS

Model Extraction.

Given access to predictions

Reconstruct the model
Previous Works


Given access to predictions + Counterfactual explanations = Reconstruct the model


The authors assume that the adversary can query the models multiple times!
CONTRIBUTIONS

Given access to an instance  \[ \begin{align*}
\text{Counterfactual explanations} \\
\end{align*} \]
Contributions

Membership Inference.

Given access to an instance + Counterfactual explanations = Instance is in training
Contributions

Given access to an instance, the adversary can query the models a unique time!
CONTRIBUTIONS

1. Define a **new class of attacks** called counterfactual distance-based attacks

2. Provide two examples of attacks in the class
CONTRIBUTIONS

1. Define a **new class of attacks** called counterfactual distance-based attacks

2. Provide two examples of attacks in the class

\[ c ("Instance", "Counterfactual") \]

Instance is in training
Preliminaries: Algorithmic Recourse

\[ x' = \arg\min_{x' \in A^p} \ell(f_{\theta}(x'), 1) + \lambda \cdot c(x, x') \]

Wachter et al.
PRELIMINARIES: MI ATTACKS
PRELIMINARIES: MI ATTACKS

THRESHOLDING ON LOSS
(YEOM ET AL.)

\[ M_{\text{Loss}}(x) = \begin{cases} \text{MEMBER} & \text{if } \ell(\theta, z) \leq \tau_L \\ \text{NON-MEMBER} & \text{if } \ell(\theta, z) > \tau_L. \end{cases} \]

very powerful and simple, but only feasible when can access instance’s \( y \);
model’s \( \ell, \Theta \); underlying data distribution \( \mathcal{D} \) (to practically get \( \tau \))
Preliminaries: MI Attacks

Thresholding on Loss (Yeom et al.)

- Loss of model with params $\theta$ on instance $z = (x, y)$
- Threshold

$$M_{\text{Loss}}(x) = \begin{cases} \text{MEMBER} & \text{if } \ell(\theta, z) \leq \tau_L \\ \text{NON-MEMBER} & \text{if } \ell(\theta, z) > \tau_L. \end{cases}$$

Very powerful and simple, but only feasible when can access instance’s $y$; model’s $\ell$, $\theta$; underlying data distribution $\mathcal{D}$ (to practically get $\tau$)

Loss Likelihood Ratio Attack (Carlini et al.)

Given: sample access to underlying data distribution $\mathcal{D}$

1. Adversary trains shadow models
2. Computes confidence in each model $f_\theta$ when $z$ in/out train set
3. Fits normal distributions to these in/out confidences
4. Computes approximate likelihood ratio $\Lambda$
5. Predicts MEMBER when $\Lambda > \tau$
Setting: Recourse-based MI Game

owner $\mathcal{O}$ and adversary $\mathcal{A}$
**Setting: Recourse-based MI Game**

**owner \( \emptyset \):**
1. Draws training set \( D_t \) from underlying data distribution \( \mathcal{D} \)
2. Trains model \( f_\emptyset \)
3. Labels every point \( z \) in \( D_t \) with binary label \( f_\emptyset(z) \)
4. Flips coin to determine where to sample \( x \) from
   a. If heads, conditional distribution \( \mathcal{D} | f_\emptyset(z) = 0 \)
   b. If tails, subset of \( D_t \) with label 0
5. Generates recourse \( x' \) for \( x \)
6. Sends \( (x', x) \)

[all data labelled 0 (unfavorable outcome), but combination of training data or not]
**Setting: Recourse-based MI Game**

**Owner** $\mathcal{O}$:
1. Draws training set $D_t$ from underlying data distribution $\mathcal{D}$
2. Trains model $f_\theta$
3. Labels every point $z$ in $D_t$ with binary label $f_\theta(z)$
4. Flips coin to determine where to sample $x$ from
   a. If heads, conditional distribution $\mathcal{D} \mid f_\theta(z) = 0$
   b. If tails, subset of $D_t$ with label 0
5. Generates recourse $x'$ for $x$
6. Sends $(x', x)$

**Adversary** $\mathcal{A}$:
1. Can also query $\mathcal{D}$, knows implementation details of $\mathcal{O}$
2. Guesses whether $x$ is MEMBER (in $D_t$) or NON-
Loss-based attacks are good at determining MEMBER, because model typically overfits to training points.

This may be because, during training, decision boundary is pushed away from training points (Shroki et al.).

Points in training set should be further from boundary than points in test set.

Counterfactual distance-based attacks!
ATTACK 1: THRESHOLDING ON CFD

Assume that $\mathcal{A}$ knows a priori optimal threshold $\tau_\alpha$ that maximizes TPR given FPR $\alpha$; in practice, will plot TPR v. FPR over all $\tau_D$.

$M_{\text{Distance}}(x) = \begin{cases} \text{MEMBER} & \text{if } c(x, x') \geq \tau_D \\ \text{NON-MEMBER} & \text{if } c(x, x') < \tau_D \end{cases}$

(Recall: $M_{\text{Loss}}(x) = \begin{cases} \text{MEMBER} & \text{if } \ell(\theta, z) \leq \tau_L \\ \text{NON-MEMBER} & \text{if } \ell(\theta, z) > \tau_L \end{cases}$)
ATTACK 2: CFD LRT

again, similar to preliminaries, but more adjustments

Algorithm 1 One-sided Distance-based Likelihood Ratio Test (CFD LRT)

1: Inputs: point \((x, y)\), recourse output \(s = \text{GetRecourse}(x, f_\theta), \mathcal{D}\); FP-Rate: \(\alpha\), # Shadow Models: \(N\), \(\mathcal{T} = \text{TrainClassifier}()\)
2: teststats = []
3: Compute: \(t_0 = T(s) = c(x, x')\) \hspace{1cm} \text{compute CFD on input}
ATTACK 2: CFD LRT

again, similar to preliminaries, but more adjustments

**Algorithm 1** One-sided Distance-based Likelihood Ratio Test (CFD LRT)

1: **Inputs:** point \((x,y)\), recourse output \(s = \text{GetRecourse}(x, f_\theta), D\); FP-Rate: \(\alpha\), # Shadow Models: \(N, T = \text{TrainClassifier}(\cdot)\)
2: \[\text{teststats} = []\]
3: **Compute:** \(t_0 = T(s) = c(x, x')\)
4: **for** \(i = 1 : N\) **do**
5: \[\text{Sample } D_t^{(i)} \sim D\]
6: \[f_{\theta^{(i)}} = \text{TrainClassifier}(D^{(i)})\]
7: \[s^{(i)} = \text{GetRecourse}(x, f_{\theta^{(i)}})\]
8: \[\text{teststats} \leftarrow T(s^{i}) = c(x, x'^{(i)})\]
9: **end for**

---

- compute CFD on input
- train shadow models (only need to do once!) and recourses, collect their CFDs on input
ATTACK 2: CFD LRT

Algorithm 1 One-sided Distance-based Likelihood Ratio Test (CFD LRT)

1: Inputs: point \((x, y)\), recourse output \(s = \text{GetRecourse}(x, \theta, D)\); FP-Rate: \(\alpha\), # Shadow Models: \(N\), \(T = \text{TrainClassifier}()\)
2: teststats = []
3: Compute: \(t_0 = T(s) = c(x, x')\)
4: for \(i = 1 : N\) do
5: Sample \(D_t^{(i)} \sim D\)
6: \(f_{\theta^{(i)}} = \text{TrainClassifier}(D^{(i)})\)
7: \(s^{(i)} = \text{GetRecourse}(x, f_{\theta^{(i)}})\)
8: teststats \(\leftarrow T(s^i) = c(x, x''^{(i)})\)
9: end for
10: \(\hat{\mu}_{\text{MLE}} = \frac{1}{N} \sum_{i=1}^{N} \left( \log c(x, x''^{(i)}) \right)\)
11: \(\hat{\sigma}_{\text{MLE}}^2 = \frac{1}{N} \sum_{i=1}^{N} \left( \hat{\mu}_{\text{MLE}} - \log (c(x, x''^{(i)})) \right)^2\)

again, similar to preliminaries, but more adjustments

compute CFD on input

train shadow models (only need to do once!) and recourses, collect their CFDs on input

estimate params of normal distribution
ATTACK 2: CFD LRT

again, similar to preliminaries, but more adjustments

Algorithm 1 One-sided Distance-based Likelihood Ratio Test (CFD LRT)

1: **Inputs:** point \((x,y)\), recourse output \(s = \text{GetRecourse}(x,f_\theta),\mathcal{D};\) FP-Rate: \(\alpha\), # Shadow Models: \(N, \mathcal{T} = \text{TrainClassifier}(\cdot)\)
2: teststats = []
3: **Compute:** \(t_0 = T(s) = c(x,x')\)
4: for \(i = 1 : N\) do
5: Sample \(\mathcal{D}_t^{(i)} \sim \mathcal{D}\)
6: \(f_\theta^{(i)} = \text{TrainClassifier}(\mathcal{D}^{(i)})\)
7: \(s^{(i)} = \text{GetRecourse}(x,f_\theta^{(i)})\)
8: teststats \(\leftarrow T(s^i) = c(x,x'^{(i)})\)
9: end for
10: \(\hat{\mu}_{\text{MLE}} = \frac{1}{N} \sum_{i=1}^{N} \left( \log c(x,x'^{(i)}) \right)\)
11: \(\hat{\sigma}^2_{\text{MLE}} = \frac{1}{N} \sum_{i=1}^{N} \left( \hat{\mu}_{\text{MLE}} - \log \left( c(x,x'^{(i)}) \right) \right)^2\)
12: if \(t_0 > z_{1-\alpha}\) then
13: **Output:** \(G = \text{NON-MEMBER}\)
14: else
15: **Output:** \(G = \text{MEMBER}\)
16: end if

\(\triangleright z_{1-\alpha}\) is the \(1-\alpha\)-quantile of \(Z \sim \mathcal{L}\mathcal{N}(\hat{\mu}_{\text{MLE}}, \hat{\sigma}^2_{\text{MLE}})\)

threshold \(\tau\)
Is privacy leakage through recourses inevitable?
Is privacy leakage through recourses inevitable?

Privacy community thinks DP in training can bound the success of any adversary.
**Bounding Success of \(\mathcal{A}\) with DP**

**Theorem 1.** Let \(\mathcal{T} : (\mathcal{X} \times \mathcal{Y})^n \to \Theta\) denote the training algorithm, draw \(D_t \sim D^n\) and and \(\mathcal{A}\) be an arbitrary adversary that receives \(z = (x, y)\), \(s \sim \mathcal{R}(f_\theta, x, D_t)\) from the recourse inference game, and produces a guess \(G \in \{\text{MEMBER, NON-MEMBER}\}\). Then, if \(\mathcal{R}\) is \((\epsilon, 0)\)-differentially private, we have for all \(\mathcal{A}\):

\[
BA_{\mathcal{A}} \leq \frac{1}{2} + \frac{1 - e^{-\epsilon}}{2}.
\]

**Implications:**
- Using DP in training, we can strongly bound the adversary’s **balanced accuracy** success ((TPR + TNR) / 2) — not just excess accuracy broadly
- For a small FPR \(\alpha\), TPR of \(\mathcal{A}\) is also close to \(\alpha\)
Bounding success of $\mathcal{A}$ with DP

**Theorem 1.** Let $\mathcal{T} : (\mathcal{X} \times \mathcal{Y})^n \to \Theta$ denote the training algorithm, draw $D_t \sim \mathcal{D}^n$ and $A$ be an arbitrary adversary that receives $z = (x, y)$, $s \sim \mathcal{R}(f_\theta, x, D_t)$ from the recourse inference game, and produces a guess $G \in \{\text{MEMBER, NON-MEMBER}\}$. Then, if $\mathcal{R}$ is $(\epsilon, 0)$-differentially private, we have for all $A$:

$$BA_A \leq \frac{1}{2} + \frac{1 - e^{-\epsilon}}{2}.$$

**Implications:**
- Using DP in training, we can strongly bound the adversary’s **balanced accuracy** success ($(\text{TPR} + \text{TNR}) / 2$) — not just excess accuracy broadly
- For a small FPR $\alpha$, TPR of $\mathcal{A}$ is also close to $\alpha$

**Proof:**
- Appendix A; mostly applies definitions and expands integrals
Bounding Success of $\mathcal{A}$ with DP

**Theorem 1.** Let $T : (\mathcal{X} \times \mathcal{Y})^n \to \Theta$ denote the training algorithm, draw $D_t \sim \mathcal{D}^n$ and and $A$ be an arbitrary adversary that receives $z = (x, y), s \sim \mathcal{R}(f_\theta, x, D_t)$ from the recourse inference game, and produces a guess $G \in \{\text{MEMBER}, \text{NON-MEMBER}\}$. Then, if $\mathcal{R}$ is $(\epsilon, 0)$-differentially private, we have for all $A$:

$$BA_A \leq \frac{1}{2} + \frac{1 - e^{-\epsilon}}{2}.$$ 

**Implications:**
- Using DP in training, we can strongly bound the adversary’s **balanced accuracy** success ($(TPR + TNR) / 2$) — not just excess accuracy broadly
- For a small FPR $\alpha$, TPR of $A$ is also close to $\alpha$

**Proof:**
- Appendix A; mostly applies definitions and expands integrals

**However:**
- DP is not a silver bullet! Training with DP causes significant drop in accuracy
Experimental Evaluation: Setup

Datasets
1. Adult (A)  
   - Label: whether income > 50,000
2. Home Equity Line of Credit (H)  
   - Label: whether individuals will repay HELOC
3. Diabetes (D)  
   - Label: whether patient will be readmitted within next 30 days
4. Synthetic  
   - Label: comes from Gaussian samples

Recourse Algorithms
1. SCFE (Wachter et al.)  
   - Gradient-based objective
2. Growing Spheres (GS)  
   - Random search in the input space
3. CCHVAE  
   - Trains a variational autoencoder (VAE)  
   - VAE searches in a lower-dimensional latent space
## EXPERIMENTAL EVALUATION: PROCEDURE

<table>
<thead>
<tr>
<th><strong>SUBSAMPLING</strong></th>
<th><strong>TRAINING</strong></th>
<th><strong>EVALUATION</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>- Subsampling 10,000 data points</td>
<td>- Fully connected classifier neural network</td>
<td>- Balanced accuracy</td>
</tr>
<tr>
<td>- 5,000 points: owner trains private model</td>
<td>- 1 hidden layer: 1000 nodes, ReLu activation</td>
<td>- Receiver operating characteristic AUC score</td>
</tr>
<tr>
<td>- 5,000 points: adversary trains shadow model, for CFD likelihood ratio (LRT) attack</td>
<td>- ADAM optimizer (lr=0.0001)</td>
<td>- Log-scale ROC curves</td>
</tr>
<tr>
<td></td>
<td>- 250 epochs</td>
<td>- True positive rates (TPRs) at low false positive rates (FPRs)</td>
</tr>
</tbody>
</table>
ATTACK EFFICIENCY

(a) SCFE

(b) GS

(c) CCHVAE

log-log transformation
ATTACK EFFICIENCY: TAKEAWAYS

- Both methods (CFD, CFD LRT) often outperform the random baseline across all metrics
- CFD LRT generally outperforms CFD

Small upwards deviations from the diagonal at low FPRs (i.e. 0.01)

A small fraction of points are accurately identified as members

Successful MI attack
**EFFECTS OF # FEATURES**

- Higher dimensionality ⇒ greater MI attack success
- At the interpolation threshold (d = n = 5000), the baseline loss-based and distance-based attacks start outperforming the LRT-based attacks.
**Effects of Model Architecture**

Models

(a) 2 layers, 1000 hidden notes

(b) 3 layers, 100 hidden nodes

(c) 3 layers, 333 hidden nodes

(d) 3 layers, 1000 hidden nodes

Complex model architecture (i.e. overfitting) \(\Rightarrow\) greater MI attack success

(especially the CFD LRT attack)
When data dimensionality is high (# of features)
- When underlying model overfits to training data (model architecture)
- Combination of the above ⇒ increased vulnerability of recourses to CFD-based MI attacks
NOVEL ATTACKS

- **Idea:** we can leverage recourses to infer private training data membership information

- **Contribution:** MI attacks that leverage counterfactual distances output by recourse methods

EVIDENCE OF PRIVACY LEAKAGE

- **Implications:** privacy leakage is a risk of recourse algorithms; explainability-privacy tradeoff

- **Relevance:** proposed MI attacks are effective in diverse domains (lending, healthcare, law)
**LIMITATIONS**

- CFD is only a heuristic—an approximation of the distance of data point $x$ to the model boundary
  - Recall: CFD = distance from $x$ to its recourse
- Attacks operate under assumption that adversary can only query recourse algorithm once
- Must assume adversary knows optimal threshold that maximizes TPR given a fixed FPR
- This paper highlights a problem (privacy leakage), but not yet a solution
- Assessed on binary classification tasks only
  - Generalizability of results (broadly)
**Future Work**

- **Generalization:** whether recourse exposes us to other forms of privacy leakage
  - Can algorithmic recourse lead to successful reconstruction attacks? How about attacks on sensitive summary statistics of the training data (or anything else about the data distribution)?
- **Generalization:** which other XAI mechanisms involve privacy violations?
- **Solutions to protect privacy:** whether we can train models that provide recourse while mitigating privacy risks
  - How do we construct faithful model explanations that also do not leak too much information about the underlying training data? What is the privacy-utility trade-off of such models?
Discussion Questions

1. Given the explainability-privacy tradeoff highlighted in this paper, what is the role of each of the following in determining explainability and privacy benchmarks when training a model?
   a. ML practitioners (model builders and model breakers)
   b. End users (model consumers)

2. Both privacy and explainability can cultivate user trust in an ML model, and the lack thereof of any of these can break this trust. Both pillars are crucial but cannot fully coexist (this is the crux of our paper)—in what situations would you care about one pillar over the other?

3. Besides recourse, what other XAI mechanisms do you think might lead to privacy violations?

4. Is it even possible to have private explanations? Is this even worth going for?