AIモデルの説明可能性Shapley値からの属性推定リスクの評価とその対策

當麻 僚太郎[1,a)]　菊池 浩明[1]

概要：機械学習モデルの公平性や学習の透明性を保証し，ユーザに納得感を与えるために機械学習モデルの出力を説明する説明可能性技術が注目されている．機械学習モデルを用いたサービスの多くは Machine Learning as a Service（MLaaS）と呼ばれるプラットフォーム上で提供されており，これらの MLaaS プラットフォームでは，モデルの出力に加えて，モデル説明するいくつかの指標を提供している．特に Shapley 値は Amazon Web Services や Google Cloud Platform，Microsoft Azure などの主要な MLaaS プラットフォームで提供されている主流の手法である．しかし，2022 年に Luo らが Shapley 値による説明からモデルへのプライベートな入力を推論出来ることを示した．これにより，Shapley 値を用いた説明可能性指標にはモデルからプライバシー情報が漏洩するリスクがあることが分かった．ただし，Shapley 値からの属性推定リスクがデータや推定アルゴリズムにどれくらい依存するかは明らかでない．そこで，本研究では，各属性間の相関の強さや攻撃者が採用するアルゴリズムに応じた属性推定リスクの変化を明らかにし，このリスクに対する対策について検討する．

キーワード：Shapley 値，説明可能性，属性推定

Mitigation and Evaluation the Risk of Feature Inference Attack from AI Model Explainability, Shapley Values

Ryotaro Toma[1,a)]　Hiroaki Kikuchi[1]

Abstract: Explainability has gained attention to ensure fairness and transparency in machine learning models, providing users with a sense of understanding. Most services for machine learning models are offered in a style of Machine Learning as a Service (MLaaS) platforms, which provide several methods to explain model outputs. Particularly, the explanation on the Shapley values is widely available on major MLaaS platforms such as Amazon Web Services, Google Cloud Platform, and Microsoft Azure. Luo et al. (2022) demonstrated that Shapley value-based explanations provided from MLaaS could lead to inference of private inputs to the model, posing privacy risks of private information leakage from models. Nevertheless, it remains unclear how the attribute inference risk from Shapley values depends on the data and the estimation algorithms. Hence, this study investigates how the attribute inference risk varies with the strength of correlations between attributes and the algorithms adopted by attackers varies and consider possible mitigation to this threat.

Keywords: Shapley values, explainability, attribute inference

1. はじめに

近年，機械学習モデルは金融や雇用などの重要な意思決定の場面で活用されることが増えている．それらの多くのモデルはニューラルネットワークやアンサンブルモデルな

[1] 明治大学総合数理学部
School of Interdisciplinary Mathematical Sciences, Meiji University
[a)] ev200598@meji.ac.jp

どの複雑な構造を持ち，入力に対してブラックボックスであった．そのため，モデルの公平性や透明性を保証し，モデルの出力に対して説明を与えるための説明可能性技術 eXplainable AI（XAI）が注目されている [3], [4], [5]．機械学習モデルを用いた商品サービスを提供するための基盤である Machine Learning as a Service（以下，MLaaS）プラットフォームでは，様々な説明可能性技術を用いた説明を提供している．特に Shapley 値 [9] を基にした説明は，Amazon Web Services [6] や Google Cloud Platform [7], Microsoft Azure [8] などの主要な MLaaS プラットフォームで提供されている手法である．

しかし，2022 年に Luo ら [1] は Shapley 値に基づく説明から本来秘匿されているモデルへの入力属性を推論出来ることを示した．

しかしながら，[1] では，最小勾配法による属性推定アルゴリズム ψ を提案しているが，説明モデル f に多くの機械学習アルゴリズムがあるように，ψ にも多くの可能性がある．特に，モデル f と ψ の間には相関があり，属性推定リスクの評価は自明ではない．そこで，本研究では，Luo ら [1] の手法を基にして，各説明変数と目的変数間の相関や攻撃者が採用するアルゴリズム ψ の違いに対してどのように属性推定リスクが変化するかを明らかにする．特に，f と ψ が線形回帰モデルのときには，Shapley 値から正確にプライベートな特徴ベクトルの推定が可能であることを示す．この提案方式を，Adult データセット [10] について適用した結果を報告する．

2. 基本定義

2.1 Shapley 値

Shapley 値 [9] は協力ゲーム理論において連携プレイヤー間で利益を分配するための協調作業を定量化するために，Shapley に提案されたものである．本研究では，n 特徴の入力 $\boldsymbol{x} = (x_1, \ldots, x_n)$ に対するモデルの出力 $f(\boldsymbol{x})$ の局所的な説明として Shapley 値ベクトル $\boldsymbol{s} = (s_1, \ldots, s_n)$ を与える．特徴量のインデックス集合を $N = \{1, 2, \ldots, n\}$，N の部分集合を S，Shapley 値を計算するために参照するデータのサンプルを $\boldsymbol{x}^0$ とする．

S に対応する入力サンプル $\boldsymbol{x}_{[S]} = ((x_{[S]})_1, \ldots, (x_{[S]})_n)$ とする．ここで，$i = 1, \ldots, n$ について，

$$(x_{[S]})_i = \begin{cases} x_i & \text{if } i \in S, \\ x_i^0 & \text{otherwise.} \end{cases} \tag{1}$$

例えば，$\boldsymbol{x} = (\mathbf{2}, \mathbf{5}, \mathbf{1}, \mathbf{3})$，$\boldsymbol{x}^0 = (0, 3, 2, 1)$，$S = \{2, 3\}$ としたとき，$\boldsymbol{x}_{[S]} = [0, \mathbf{5}, \mathbf{1}, 1]$ である．このとき，特徴量に対する Shapley 値 s_i は，

$$s_i = \frac{1}{n!} \sum_{S \subseteq N \setminus \{i\}} |S|!(n - |S| - 1)! \left(f(\boldsymbol{x}_{[S \cup \{i\}]}) - f(\boldsymbol{x}_{[S]}) \right) \tag{2}$$

で定められる．$\boldsymbol{s} = \phi(\boldsymbol{x}; \boldsymbol{x}^0, f) = (s_1, \ldots, s_n)$ を Shapley 値を与える写像とする．

2.2 Shapley sampling values

式 (2) の時間計算量は $O(2^n)$ であるため，多くの MLaaS プラットフォームでは近似的に Shapley 値を計算するサンプリング手法を採用している．本研究では，$n!$ 個の順列のうちランダムに $v = 50$ 個サンプリングして Shapley 値 s_i を計算している．

2.3 Feature Inference Attack on Shapley Values [1]

2.3.1 モデル

Luo ら [1] は，サービス事業者が機密データセット $\mathcal{X}_{train}$ に基づいてブラックボックスモデル f を訓練し，MLaaS プラットフォーム上に展開するシステムモデルを想定している．その実験概要図を図 1 に示す．

ユーザはプライベートな入力サンプル $\boldsymbol{x}$ を送信し，モデルの出力 $\hat{\boldsymbol{y}} = (y_1, \ldots, y_c)$ と n 個の説明値のベクトル $\boldsymbol{s} = (s_1, \ldots, s_n)$ を得る．ただし，c は正解ラベルの数である．$c > 2$ のとき対応する説明ベクトルは本来 c 個分得られるが，ここでは $c = 1$ とする．

2.3.2 攻撃者 1

攻撃者 1 は $\mathcal{X}_{train}$ と同じ分布に従う補助データセット $\mathcal{X}_{aux}$ を持っていると仮定する．全ての $\boldsymbol{x}_{aux} \in \mathcal{X}_{aux}$ をモデル f に送信し，対応する説明データ $\mathcal{S}_{aux}$ を得る．そして，$\psi : \mathcal{S}_{aux} \to \mathcal{X}_{aux}$ が誤差 $L(\psi(\mathcal{S}_{aux}), \mathcal{X}_{aux})$ を最小化するように訓練する．プライベートな入力 $\boldsymbol{x}$ の推測値は，与えられた Shapley 値 $\boldsymbol{s}$ を用いて $\hat{\boldsymbol{x}} = \psi(\boldsymbol{s})$ とする．攻撃者 1 の属性推論を Algorithm 1 に示す．

Algorithm 1 補助データセットを用いた推定 [1]

Input: ブラックボックスモデル f，補助データセット $\mathcal{X}_{aux}$，学習率 α，攻撃対象の Shapley 値ベクトル $\boldsymbol{s}$
Output: 推論されたプライベートな入力 $\hat{\boldsymbol{x}}$

```
1:  S_aux ← φ(X_aux; f)
2:  θ_ψ ← N(0, 1)
3:  for each epoch do
4:      for each batch do
5:          loss ← 0
6:          B ← randomly select a batch of samples
7:          for j ∈ 1, ..., |B| do
8:              (x̂_aux)^j ← ψ((s_aux)^j; θ_ψ)
9:              loss ← loss + L((x̂_aux)^j, (x_aux)^j)
10:         end for
11:         θ_ψ' ← θ_ψ − α∇_{θ_ψ} loss
12:     end for
13: end for
14: x̂ ← ψ(s; θ_ψ)
15: return x̂
```

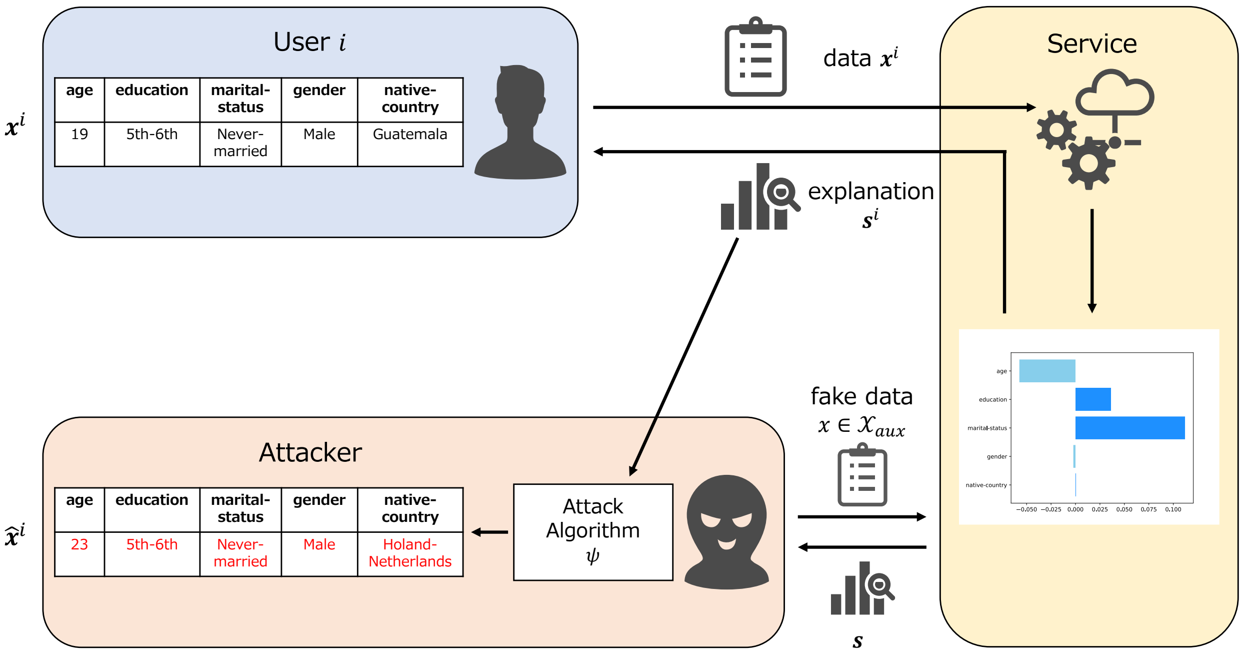

図 1　全体概要図

2.3.3　攻撃者 2

攻撃者 2 は補助データセットを持たず，データの分布や変数間の相関を知らないと仮定する．そのため，ランダムに生成した $\mathcal{X}_{rand}$ について対応する説明データセット $\mathcal{S}_{rand}$ を得る．プライベートな入力 $\boldsymbol{x}$ の特徴量 i の推測値 $\hat{x}_i$ は，与えられた Shapley 値 $\boldsymbol{s}$ を用いて，$|(s_{rand})_i^j - s_i| \leq \xi$ を満たす k 行のサンプル $\{(x_{rand})_i^j\}_{j=1}^k$ について，$\hat{x}_i = \frac{1}{k}\sum_{j=1}^k (x_{rand})_i^j$ とする．攻撃者 2 の属性推論を Algorithm 2 に示す．

3.　基本原理

10 行 5 列のトイデータに対して属性推論攻撃を行う．データの生成は，標準正規分布に従う独立な 3 つの乱数列 $n_1 \sim \mathcal{N}(0,1)$，$n_2 \sim \mathcal{N}(0,1)$，　$n_3 \sim \mathcal{N}(0,1)$ を用いて，$x_1 = n_1$，$x_2 = n_2$，$x_3 = n_1 n_2$，$x_4 = n_2 n_3$，$y = x_1 - x_3 x_4$ とする．生成したデータを表 1，y に対する各列の相関係数を表 2 に示す．

データの 1〜5 行目を $\mathcal{X}_{test}$，6〜10 行目を $\mathcal{X}_{train}$ とする．Shapley 値は $\mathcal{X}_{train}$ の各行を参照サンプルとし，それぞれ求めた Shapley 値の平均，すなわち $\boldsymbol{s} = \frac{1}{5}\sum_{j=6}^{10} \phi(\boldsymbol{x}, \boldsymbol{x}^j)$ とする．

例として，$\mathcal{X}_{train}$ をフィッティングした線形回帰モデル f について，$\mathcal{X}_{test}$ に対する Shapley 値 $\mathcal{S}_{test}$ を表 3 に示す．

また，モデル f と推定アルゴリズム ψ の組み合わせに対する MAE を表 4 に，SR を表 5 に示す．ここで，攻撃者 2 が使うランダムデータセット $\mathcal{X}_{rand}$ は標準正規分布に従う乱数を 100 行生成した．

Algorithm 2 補助データセットを用いない攻撃 [1]

Input: ブラックボックスモデル f，サンプリングエラーの閾値 ξ，最小サンプル数 m_C，推論値の幅の閾値 τ，攻撃対象の Shapley 値ベクトル $\boldsymbol{s}$，入力特徴の個数 n

Output: 推論されたプライベートな入力 $\hat{\boldsymbol{x}}$

```
x̂ ← ∅
X_rand ← m 行 n 列のランダムデータセット
S_rand ← φ(X_rand; f)
for i = 1, 2, ..., n do
    D ← ∅
    for j = 1, 2, ..., m do
        dist ← |s_i − (s_rand)_i^j|
        x̃ ← (x_rand)_i^j
        D ← D ∪ (dist, x̃)
    end for
    Sort D on dist in an ascending order
    C ← ∅
    for j = 1, 2, ..., m do
        (dist, x̃) ← D_j
        if |C| < m_C or dist < ξ then
            C ← C ∪ {x̃}
        end if
    end for
    if max C − min C > τ then
        x̂_i ←⊥
    else
        x̂_i ← (1/|C|) ΣC
    end if
    x̂ ← x̂ ∪ {x̂_i}
end for
return x̂
```

表 1　トイデータ

	x_1	x_2	x_3	x_4	y
$\mathcal{X}_{test}$	1.8	0.1	0.3	−0.4	1.9
	0.4	1.5	0.6	1.0	−0.2
	1.0	0.8	0.7	0.7	0.5
	2.2	0.1	0.3	−0.1	2.2
	1.9	0.4	0.8	1.0	1.1
$\mathcal{X}_{train}$（$\mathcal{X}_{aux}$）	−1.0	0.3	−0.3	−0.5	−1.2
	1.0	1.5	1.4	0.1	0.9
	−0.2	−0.2	0.0	0.0	−0.2
	−0.1	0.3	0.0	0.5	−0.1
	0.4	−0.9	−0.4	−1.3	−0.1

表 2　トイデータの y に対する各列の相関係数

x_1	x_2	x_3	x_4
0.95	0.02	0.44	0.08

表 3　Shapley 値 $\mathcal{S}_{test}$

	s_1	s_2	s_3	s_4
$\boldsymbol{x}^1$	1.30	0.02	0.06	−0.04
$\boldsymbol{x}^2$	0.28	−0.29	0.18	0.34
$\boldsymbol{x}^3$	0.72	−0.13	0.21	0.26
$\boldsymbol{x}^4$	1.59	0.02	0.06	0.04
$\boldsymbol{x}^5$	1.37	−0.04	0.25	0.34

表 4　モデル f と推定アルゴリズム ψ の組み合わせに対する MAE

攻撃	f	ψ	MAE				
			x_1	x_2	x_3	x_4	平均
1	線形回帰	線形回帰	0.00	0.00	0.00	0.00	0.00
	線形回帰	決定木	0.82	1.24	0.74	1.18	1.00
	決定木	線形回帰	0.69	0.52	0.41	0.53	0.54
	決定木	決定木	0.68	1.16	0.82	0.54	0.80
2	線形回帰	N/A	0.06	0.00	0.00	0.00	0.02
	決定木	N/A	0.39	0.54	0.46	0.69	0.52
平均			0.44	0.58	0.41	0.49	

表 5　モデル f と推定アルゴリズム ψ の組み合わせに対する SR

攻撃	f	ψ	SR				
			x_1	x_2	x_3	x_4	平均
1	線形回帰	線形回帰	1.00	1.00	1.00	1.00	1.00
	線形回帰	決定木	0.40	0.00	0.00	0.00	0.00
	決定木	線形回帰	0.00	0.20	0.00	0.00	0.05
	決定木	決定木	0.00	0.20	0.00	0.00	0.05
2	線形回帰	N/A	0.40	1.00	1.00	1.00	0.85
	決定木	N/A	0.00	0.00	0.00	0.00	0.00
平均			0.33	0.37	0.33	0.33	

結果として，目的変数との相関がある x_1 と x_3 の MAE が 0.44，0.41 と低く，そうでない x_1 と x_4 の MAE は 0.58 と 0.49 であった．また，f と ψ がどちらも線形モデルのとき，正確に入力特徴を推論出来た．

補題. f を線形回帰による説明モデルとする．任意の $i \in N$，$S \subseteq N \setminus \{i\}$，参照ベクトル $(x_1^0, x_2^0, \ldots, x_n^0)$ について，

$$f(\boldsymbol{x}_{[S\cup\{i\}]}) - f(\boldsymbol{x}_{[S]}) = \beta_i(x_i - x_i^0) \tag{3}$$

である．

証明)　説明モデル f を $f(\boldsymbol{x}) = \beta_0 + \beta_1 x_1 + \cdots + \beta_n x_n$ と表すと，線形モデルのため，

$$\begin{aligned} f(\boldsymbol{x}_{[S\cup\{i\}]}) - f(\boldsymbol{x}_{[S]}) &= \beta_0 + \sum_{k\in S\cup\{i\}} \beta_k x_k + \sum_{k\in N\setminus(S\cup\{i\})} \beta_k x_k^0 \\ &\quad - \beta_0 + \sum_{k\in S} \beta_k x_k + \sum_{k\in N\setminus S} \beta_k x_k^0 \\ &= \beta_i(x_i - x_i^0) \end{aligned} \tag{4}$$

□

命題. f を線形モデルによる説明モデル，ψ を線形モデルによる推定アルゴリズムとする．$n < |\mathcal{X}_{aux}|$ のとき，ψ による推定の MAE = 0 である．

証明)　補題より，s_i について

$$\begin{aligned} s_i &= \frac{1}{n!} \sum_{S\subseteq N\setminus\{i\}} |S|!(n-|S|-1)! f(\boldsymbol{x}_{[S\cup\{i\}]}) - f(\boldsymbol{x}_{[S]}) \\ &= \frac{1}{n!} \sum_{S\subseteq N\setminus\{i\}} |S|!(n-|S|-1)! \beta_i(x_i - x_i^0) \\ &= \lambda_i(x_i - x_i^0) \end{aligned} \tag{5}$$

ただし，ここで $\lambda_i = \frac{1}{n!}\sum_{S\subseteq N\setminus\{i\}} |S|!(n-|S|-1)!\beta_i$ をまとめた項である．したがって，ψ による推定モデルは，

$$\begin{aligned} \hat{x}_i &= \alpha_0 + \alpha_1 s_1 + \cdots + \alpha_n s_n \\ &= \alpha_0 + \alpha_1(\lambda_1(x_1 - x_1^0)) + \cdots + \alpha_n(\lambda_n(x_n - x_n^0)) \\ &= \alpha_0 - \sum_{k=1}^{n} \alpha_k \lambda_k x_k^0 + \alpha_1\lambda_1 x_1 + \cdots + \alpha_n \lambda_n x_n \\ &= \gamma_0 + \gamma_1 x_1 + \cdots + \gamma_n x_n \end{aligned} \tag{6}$$

と，$x_1, \ldots, x_n$ の線形式で与えられる．ただし，ここで $\gamma_i = \alpha_i\lambda_i$，$\gamma_0 = \alpha_0 - \sum_{k=1}^{n} \alpha_k\lambda_k x_k^0$ をまとめた項である．

$\mathcal{X}_{aux}$ が十分に大きく，$n+1$ 以上の行数があるならば，最小二乗法により，誤差なく $\gamma_1, \ldots, \gamma_n$ が算出される．□

結果のところ，線形式によるモデルの説明データを提供すると，属性推定リスクが上がることを意味している．

4. 提案方式

想定する攻撃は先行研究と同様に，図 1 とする．実験に用いるデータセットを表 6 に示す．

表 6　使用データセット

データセット	レコード数	クラス	特徴量
Adult[10]	48842	2	14

この設定下において，説明モデル f や攻撃者 1 が採用す

る最適化アルゴリズム，各説明変数と目的変数間の相関などに対する属性推定リスクを調べる．

4.1　評価指標

属性推定リスクの評価に用いる指標は，MAE と攻撃成功率 SR の 2 つである．

4.1.1　MAE

MAE（Mean Absolute Error），すなわち $\ell_1$loss は誤差の絶対値の平均を取る．m 行 n 列のデータセット $\boldsymbol{x}$ に対する推定データ $\hat{\boldsymbol{x}}$ の MAE は

$$\ell_1(\hat{\boldsymbol{x}}, \boldsymbol{x}) = \frac{1}{mn}\sum_{j=1}^{m}\sum_{i=1}^{n}|\hat{x}_i^j - x_i^j| \tag{7}$$

で与えられる．

4.1.2　攻撃成功率 SR

SR（Success Rate）は攻撃によって正しく推定された入力特徴量の割合を表す．質的変数に対しては推定カテゴリが一致しているかどうか，量的変数に対しては推定値と真の値との誤差の絶対値がある閾値以下であるかどうかで成功の判定をしている．$\boldsymbol{x}$ と推定 $\hat{\boldsymbol{x}}$ の SR は

$$SR(\hat{\boldsymbol{x}}, \boldsymbol{x}) = \frac{success(\hat{\boldsymbol{x}}, \boldsymbol{x})}{mn} \tag{8}$$

で与えられる．ここで，推定に成功した入力特徴の個数を $success(\hat{\boldsymbol{x}}, \boldsymbol{x})$ とする．

4.2　実験方法

4.2.1　実験 1

説明モデル f に対する属性推定リスクを調べる．モデル f はニューラルネットワーク（NN），ランダムフォレスト（RF），勾配ブースティング木（GBDT）の 3 種類で実験を行う．NN は Pytorch で実装し，n 次元の入力層と c 次元の出力層を持ち，ニューロン数 $2n$ の隠れ層を 2 つ持つ．活性化関数は出力層のみ softmax でそれ以外は ReLU を用いる．RF，GBDT は sklearn で実装した．RF と GBDT の木の数と最大の深さはそれぞれ $(100, 5)$，$(100, 3)$ とする．その他のパラメータは全てデフォルトの値とする．RG-E は $\mathcal{X}_{aux}$ に基づく経験分布からランダムに予測したときの結果であり，RG-U と RG-N はそれぞれ $U(0, 1)$ の一様分布に従う乱数と $N(0.5, 0.25^2)$ の正規分布に従う乱数を予測としたときの結果である．

それぞれの攻撃者について，攻撃者が持つデータセット $\mathcal{X}_{aux}$，$\mathcal{X}_{rand}$ の行数を変化させたときに，どのように MAE と SR が変化するかを調べる．攻撃者 1 の結果を図 2 と図 3，攻撃者 2 の結果を図 4 と図 5 にそれぞれ示す．

4.2.2　実験 2

線形モデル f に対する攻撃が成功することを確認する実験を行う．Adult データセット [10] から f をフィッティングし，それに対して攻撃者 1 のアルゴリズムを用いて

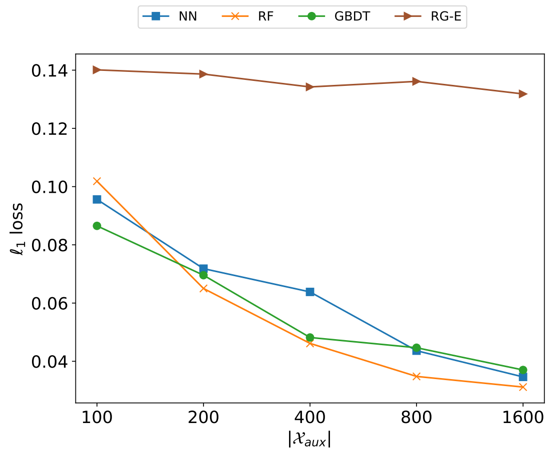

図 2　各モデル f とデータセットの大きさ $|\mathcal{X}_{aux}|$ についての攻撃者 1 の MAE の分布

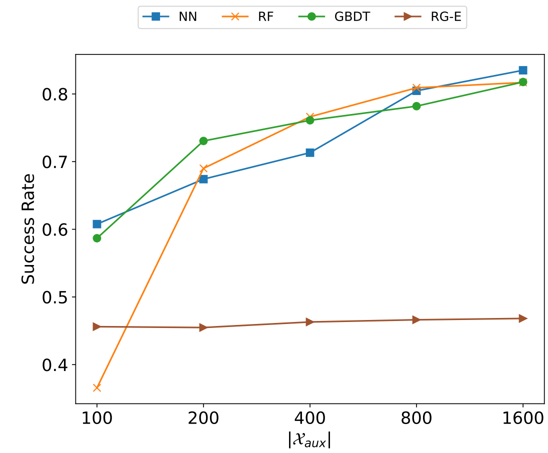

図 3　各モデル f とデータセットの大きさ $|\mathcal{X}_{aux}|$ についての攻撃者 1 の SR の分布

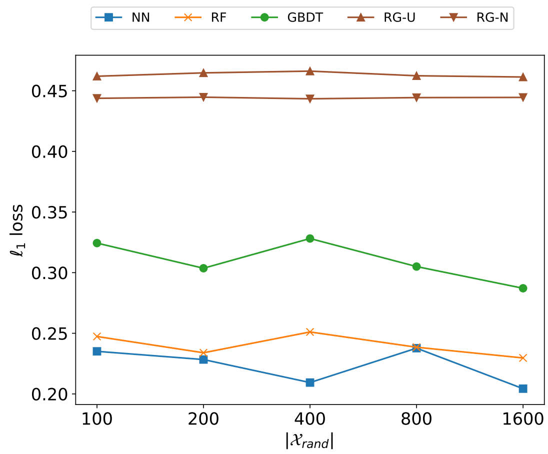

図 4　各モデル f とデータセットの大きさ $|\mathcal{X}_{rand}|$ についての攻撃者 2 の MAE の分布

属性推論を行う線形回帰モデル ψ を学習する．その結果として，$\mathcal{X}_{test}$ の各列に対する SR を表 7 に示す．ただし，$|\mathcal{X}_{aux}| = 1600$ とする．

4.2.3　実験 3

攻撃者 1 が採用する最適化アルゴリズムを変化させたと

表 7　$\mathcal{X}_{test}$ の各列に対する SR

列	1	2	3	4	5	6	7	8	9	10	11	12	13	14
SR	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00

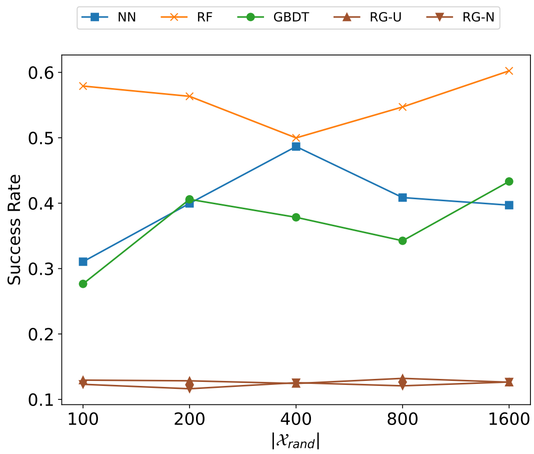

図 5　各モデル f とデータセットの大きさ $|\mathcal{X}_{rand}|$ についての攻撃者 2 の SR の分布

きの属性推定リスクを調査する．攻撃者 1 の採用する推定アルゴリズムとは，攻撃者 1 が Algorithm 1 において攻撃モデル ψ のパラメータ θ_ψ を更新する式 (9)

$$\theta_\psi \leftarrow \theta_\psi - \alpha \nabla_{\theta_\psi} loss \tag{9}$$

の変種を意味する．本研究では，勾配降下法ベースの最適化アルゴリズムとして，SGD[12]，Momentum[13]，RMSprop[14]，Adam[15] の 4 種類を調べる．推定モデル ψ は先行研究と同じく，特徴量の数 n に対して隠れ層のニューロン数 $4n$，出力層のニューロン数 n のニューラルネットワークとし，活性化関数は全てで sigmoid 関数を用いる．実装は Pytorch で行い，SGD の学習率 $\eta = 0.01$，Momentum（すなわち SGD のうち $momentum \neq 0$ のもの）の学習率 $\eta = 0.01$，$momentum = 0.9$ と指定したもの以外は全てデフォルトのパラメータを用いる．

実験の結果を図 6 に示す．

4.2.4　実験 4

データセットの各説明変数と目的変数との間の相関係数に対して，属性推定を行ったときの MAE の関係を明らかにする．相関係数の計算は，目的変数が質的変数であるため，説明変数が量的変数のときは相関比，質的変数のときは Cramer の連関係数 [11] で与える．説明モデル f には NN を，攻撃者 1 の属性推論アルゴリズム ψ には RMSprop を用いる．各属性ごとの MAE を補助データセットやランダムデータセットの大きさを変化させて計算し，それらの平均を取ったものをその属性に対する MAE を評価する．実験の結果を，攻撃者 1 について図 7 に，攻撃者 2 について図 8 に示す．

4.3　実験結果と考察

4.3.1　結果 1

図 2 について，データセットの大きさ $|\mathcal{X}_{aux}|$ が増えるにしたがって全てのモデルで MAE が小さくなった．最も MAE の小さいモデルは，$|\mathcal{X}_{aux}| = 100$ のとき GBDT，それ以外のときは RF であった．

図 3 について，$|\mathcal{X}_{aux}|$ が増えるにしたがって全てのモデルで SR が大きくなった．ただし，$|\mathcal{X}_{aux}| = 100$ のときのみ RF に対する SR はランダムな予測 RG-E の SR を下回った．最も SR の大きいモデルは，$|\mathcal{X}_{aux}| = 100, 1600$ のときは NN，$|\mathcal{X}_{aux}| = 200$ のときは GBDT，それ以外のときは RF であった．

図 4 について，データセットの大きさ $|\mathcal{X}_{rand}|$ に対して全てのモデルで MAE は減少しなかった．また，全ての $|\mathcal{X}_{rand}|$ について，最も MAE の大きいモデルは GBDT，最も MAE の小さいモデルは NN であった．

図 5 について，$|\mathcal{X}_{rand}|$ に対して全てのモデルで SR は増加しなかった．最も SR の大きいモデルは RF であり，最も SR の小さいモデルは $|\mathcal{X}_{rand}| = 200, 1600$ のときは NN，それ以外のときは GBDT であった．

$|\mathcal{X}_{aux}| = 100$ のときの RF を除いて，全てのモデルとデータセットの大きさについて属性推定の精度はランダムな予測を上回った．

4.3.2　結果 2

全ての入力属性について，SR が 1.00 となった．したがって，$|\mathcal{X}_{rand}|$ が十分に大きいとき，説明モデル f と属性推定モデル ψ が線形モデルであれば攻撃が出来る．

4.3.3　結果 3

MAE と SR の双方において，SGD が最も属性推定の精度が低く，RMSprop が最も属性推定の精度が高い．全ての最適化アルゴリズムで，$|\mathcal{X}_{aux}|$ が増加するにつれて MAE は減少した．また，Adam と RMSprop は MAE が小さく，SGD と Momentum は MAE が大きい傾向が見られた．同様に，$|\mathcal{X}_{rand}|$ が増加するにつれて SR も増加した．最も SR の高いものから順に RMSprop，Adam，Momentum，SGD となった．

4.3.4　結果 4

攻撃者 1 に関して，相関が弱いときには MAE が大きいものも小さいものも存在していたが，相関が強いときには MAE が小さくなった．最も MAE が大きかったのは hours-per-week の列であり，その相関係数は 0.052 であった．一方で，攻撃者 2 に関しては，攻撃者 1 と同じような傾向が見られなかった．

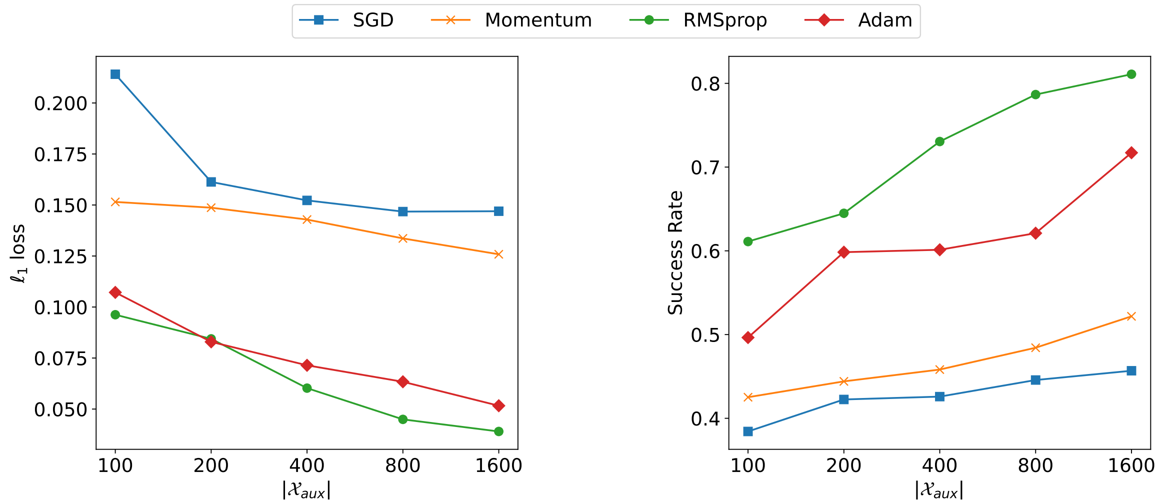

図 6　攻撃者 1 が採用する最適化アルゴリズムに対する，補助データセットの大きさを変化させたときの MAE と SR の変化

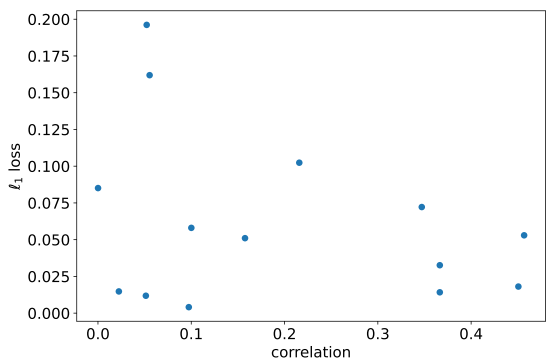

図 7　各説明変数と目的変数間の相関係数に対する攻撃者 1 の MAE の分布

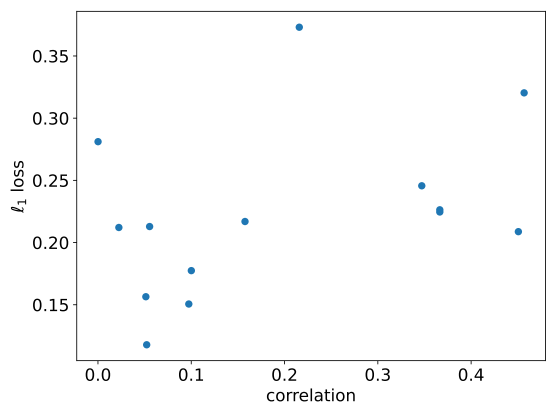

図 8　各説明変数と目的変数間の相関係数に対する攻撃者 2 の MAE の分布

4.3.5　質的変数に対するエンコーディング手法の影響

本研究では，データセット内の質的変数を全て One-hot エンコーディングして数値に変換している．これにより，モデル f への入力は本来の 14 次元から 119 次元に変換されている．この変換を行う写像 ϵ は単射であるが全射でない．そのため，ψ によって推定される 119 次元のベクトルは ϵ の値域に収まらない．これは，One-hot エンコーディングされた列は本来 0 か 1 のどちらかであるが，ψ によって推定された結果は各列ごとに sigmoid 関数を通しているため，取る値は $(0, 1)$ であることで説明される．したがって，元の次元数よりも情報量の多いベクトルを推定するため，本来の属性推定リスクとは異なる評価が得られているはずである．

5.　おわりに

Luo ら [1] の手法に基づき，2 種類の攻撃アルゴリズムについて属性推定リスクを調べた．結果として，全ての説明モデル f に対して，ランダムな予測よりも高い精度で属性推定された．特に，f が SVM であるときに最も属性推定リスクがある．また，f と ψ が線形モデルのとき，Shapley 値から正確にプライベートな入力特徴量の推定が可能である．

属性推定のリスクを抑えるために，公開する Shapley 値にノイズを加えることを提案する．また，Bozorgpanah ら [2] はデータそのものを匿名加工や差分プライバシーによって保護しても，ある程度であれば Shapley 値の有用性を損なわないことを報告している．そのため，データに対する加工と Shapley 値に対する加工によって属性推定リスクを下げられることが期待される．

今後の課題として，Shapley 値にノイズを加えたときの属性推定リスクの調査や Shapley 値以外の説明可能性技術に対する属性推定リスクの調査が挙げられる．

参考文献

[1] Xinjian Luo, Yangfan Jiang, and Xiaokui Xiao. 2022. Feature Inference Attack on Shapley Values.

In *Proceedings of the 2022 ACM SIGSAC Conference on Computer and Communications Security (CCS '22), November 7-11, 2022, Los Angeles, CA, USA*. ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3548606.3560573

[2] Bozorgpanah, A., Torra, V., and Aliahmadipour, L. 2022. Privacy and Explainability: The Effects of Data Protection on Shapley Values. Technologies 10, 6, 125. https://doi.org/10.3390/technologies10060125

[3] Vijay Arya, Rachel K. E. Bellamy, Pin-Yu Chen, Amit Dhurandhar, Michael Hind,Samuel C. Hoffman, Stephanie Houde, Q. Vera Liao, Ronny Luss, Aleksandra Mo-jsilovic, Sami Mourad, Pablo Pedemonte, Ramya Raghavendra, John T. Richards,Prasanna Sattigeri, Karthikeyan Shanmugam, Moninder Singh, Kush R. Varshney,Dennis Wei, and Yunfeng Zhang. 2019. One Explanation Does Not Fit All: AToolkit and Taxonomy of AI Explainability Techniques. CoRRabs/1909.03012(2019). https://arxiv.org/abs/1909.03012

[4] Cynthia Rudin. 2019. Stop explaining black box machine learning models forhigh stakes decisions and use interpretable models instead. *Nature Machine Intelligence* 1, 5 (2019), 206-215.

[5] Jianbo Chen, Le Song, Martin J. Wainwright, and Michael I. Jordan. 2018. Learning to Explain: An Information-Theoretic Perspective on Model Interpretation. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, Vol. 80. PMLR, 882-891.

[6] Amazon Web Services, Inc. 2023. Amazon SageMaker Clarify Model Explainability. https://docs.aws.amazon.com/sagemaker/latest/dg/clarify-model-explainability.html

[7] Google Cloud. 2023. Introducion to AI Explanations for AI Platform. https://cloud.google.com/ai-platform/prediction/docs/ai-explanations/overview?hl=en

[8] Microsoft. 2023. Model Interpretability. https://learn.microsoft.com/en-us/azure/machine-learning/how-to-machine-learning-interpretability?view=azureml-api-2

[9] Lloyd S Shapley. 1953. A value for n-person games. Vol. 2. Princeton UniversityPress, 303-317.

[10] Dheeru Dua and Casey Graff. 2017. UCI Machine Learning Repository. https://archive.ics.uci.edu/dataset/2/adult

[11] Cramér, Harald. 1946. Mathematical Methods of Statistics. Princeton: Princeton University Press, page 282

[12] Bottou, Léon. 1998. Online Algorithms and Stochastic Approximations. Cambridge University Press. ISBN 978-0-521-65263-6.

[13] Ilya Sutskever, James Martens, George Dahl, Geoffrey Hinton. 2013. On the importance of initialization and momentum in deep learning. In Proceedings of the 30th international conference on machine learning (ICML-13). Vol. 28. Atlanta, GA. pp. 1139-1147. Retrieved 14 January 2016.

[14] Geoffrey Hinton. 2018. Coursera Neural Networks for Machine Learning Lecture 6. https://www.cs.toronto.edu/ tijmen/csc321/slides/lecture_slides_lec6.pdf

[15] Diederik P. Kingma, Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. ICLR 2015. https://arxiv.org/abs/1412.6980