Generative Adversarial Networksを用いた深層学習モデルに対するConcept Extraction攻撃

草野 光亮[1]　佐久間 淳[1,2]

概要：機械学習を利用したサービスにおいて，オフライン環境で予測を行う場合など，予測モデルを外部に公開しなければならない場合がある．しかし，訓練データやその生成分布が秘密情報である場合，公開された予測モデルから訓練データ生成分布が第三者に推定される (Concept Extraction 攻撃) 可能性がある．本稿では，この攻撃を定式化し，Generative Adversarial Network を用いた攻撃アルゴリズムの提案を行う．実験により，攻撃者が攻撃対象であるラベルのサンプルを知識として持っていなくとも，補助データを活用し，攻撃者が訓練データ生成分布を推定することができることを示した．

キーワード：PWS, プライバシー, Concept Extraction 攻撃, モデル公開, Generative Adversarial Networks

1. はじめに

昨今の深層学習の急速な発展により，深層学習技術を利用し予測を行うサービスやアプリケーションに対し注目が集まっている [1]．このようなサービスでは，訓練データを入力として与えると予測を出力する予測モデルを学習させ利用する．例として，顔画像 $\mathbf{x}$ から個人 t を識別するようなサービスでは，$\mathbf{x}$ が対象の個人と識別される確率を出力する分類モデル $f(\mathbf{x}) \simeq \Pr[T = t|X = \mathbf{x}]$ を利用すると考えられる．

深層学習技術を利用したアプリケーションにおいて，サービス上の制約により予測モデルを公開情報として扱わなければならない場合が存在する．例をいくつか取り上げる．

クラウド予測サービス: サービス運営者が予測を行う際にクラウドの計算資源を利用する場合，サービス運営者の持つ予測モデルをクラウドに対し公開する必要がある．

個別化医療: 個別化医療など遺伝情報から何らかの予測を行う例を考える．ユーザーが遺伝情報を保持しており，サービス運営者が予測モデルを保持しているとする．遺伝情報は一般に秘密情報として扱われ，ユーザーは遺伝情報をサービス運営者に公開せず予測を行いたいとする．このときユーザーのローカル環境で予測を行うとすると，サービス運営者は予測モデルをユーザーに公開する必要がある．

オフライン環境での予測: 深層学習モデルで物体認識を行う自動運転車や顔画像でログイン認証を行うラップトップが挙げられる．双方ともに移動体という性質上オフライン環境下で予測を行うことを想定する必要がある．このとき，ローカル環境で予測モデルにアクセスできる必要があり，予測モデルを配布していると考えられ公開情報となる．

本稿では訓練データ $\mathcal{D}_{\mathrm{cls}}$ とその生成分布 $d_{X|T_{\mathrm{cls}}}$ が秘密情報である場合に，予測モデル f から訓練データ生成分布 $d_{X|T_{\mathrm{cls}}}$ が推定されるリスクについて議論する．訓練データ生成分布が推定された場合，プライバシー上のリスク，リバースエンジニアリングのリスク，訓練データの機密性が脅かされるリスクが存在する．各リスクについて議論する．

プライバシー上のリスク: Alice の顔画像を用いて学習した予測モデルの公開を考える．このとき訓練データ生成分布は Alice の顔画像の分布である．攻撃者が予測モデルから Alice の顔画像の分布を推定できた場合，任意の Alice の顔画像，様々な角度や様々な表情の Alice の顔画像を攻撃者は得ることができ，プライバシー上のリスクとなりうる．

リバースエンジニアリングのリスク: 自動運転で使われるような標識などの画像を入力し運転に関する意思決定を行う予測モデルを考える．この予測モデルの訓練データ生成分布を推定すること自体が，予測モデルに対するリバースエンジニアリングである．推定された訓練データ生成分布を活用することにより，予測モデルを誤識別させるサンプルを探すなど予測モデルの脆弱性を露呈させることが可能になる恐れがある．

訓練データの機密性のリスク:化合物から生理活性を予測する予測モデルを考える．生理活性とはその化合物が人体

[1] 筑波大学 大学院
University of Tsukuba
[2] 理化学研究所 革新知能統合研究センター
RIKEN Center for AIP

© 2017 Information Processing Society of Japan

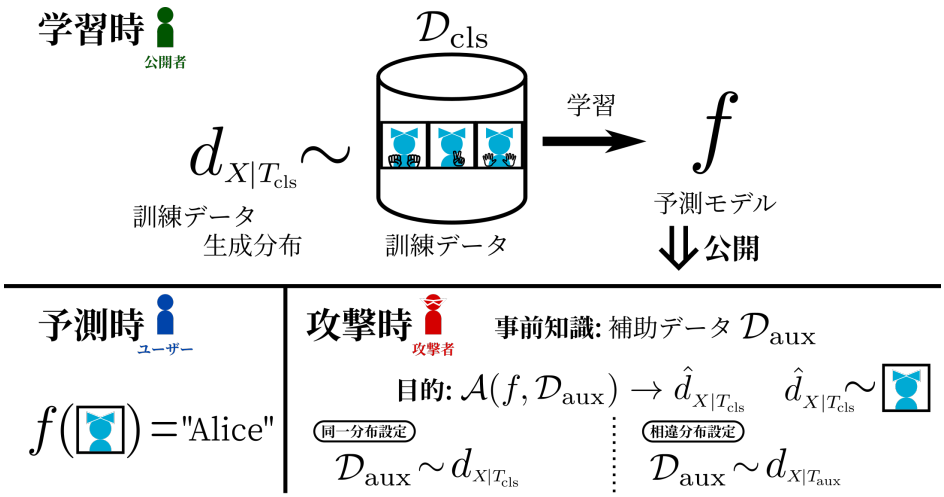

図 1 Concept Extraction 攻撃の概要. 学習は公開者 (サービス) が訓練データ生成分布 $d_{X|T_{\mathrm{cls}}}$ に従う訓練データ $\mathcal{D}_{\mathrm{cls}}$ から予測モデル f(alice の顔を識別するモデル) を学習する．予測時は，入力 $\mathbf{x}$(alice の顔画像) を与え予測値 (alice のラベル) を得る．攻撃者は，f と補助データ $\mathcal{D}_{\mathrm{aux}}$ から $d_{X|T_{\mathrm{cls}}}$ の推定値 $\hat{d}_{X|T_{\mathrm{cls}}}$ を得ることを目的とする．もし $\hat{d}_{X|T_{\mathrm{cls}}}$ が正確に推定できた場合，攻撃者は任意のサンプル (alice の任意の顔画像) を生成できる．

に対し及ぼす効果であり創薬などで議論される．訓練データである化合物活性のデータは，製薬会社などが新薬の開発のため試作した化合物などが含まれるため機密であるとする．予測モデルが公開もしくは漏洩し，予測モデルから訓練データ生成分布が推定できた場合，訓練データで利用した試作化合物の特徴が推定でき，そこから新薬の情報が推定できるなど機密性のリスクが存在する．

訓練データ生成分布 $d_{X|T_{\mathrm{cls}}}$ は概念的特徴 (Concept) と考えることができる．Alice の顔画像の生成分布が推定できれば，様々な角度や様々な表情の Alice の顔画像が推定できるため，その生成分布は Alice の顔画像の概念的特徴であるといえる (図 1)．化合物の例では，新薬の開発のため試作した化合物は，新薬の着眼点に関連する共通の特徴を持つと考えられ，その化合物の生成分布は新薬の着眼点の概念的特徴を表しているといえる．

我々は訓練データ生成分布を概念的特徴 (Concept) と考え，予測モデルを得た第三者 (攻撃者) による訓練データ生成分布の推定攻撃を Concept Extraction 攻撃と命名する．Concept Extraction 攻撃のリスクを評価することは，前述のようなリスクが存在するかを評価することにあたり，公開可能かの判断するために重要である．

Concept Extraction 攻撃は困難であると考えられてきた．理由は大きく 2 つある．一つ目は，高次元のデータの生成分布を推定する事自体がそもそも困難である点である．特に画像や音声のような高次元なデータの生成分布を正確に推定することは未だに難しく，盛んに研究されている [2]．二つ目は，攻撃者は訓練データを保有しておらず，予測モデルから訓練データ生成分布を推定しなければならないことである．一般に，予測モデルは訓練データより情報量が少ないと考えられる．これは予測モデルは識別などのタスクを解くことを目的としており，タスクを解くために必要な特徴のみに注目し学習しているためである．データから生成分布の推定自体が困難であるにも関わらず，訓練データより情報量の少ない予測モデルから訓練データ生成分布を推定することはより困難であると考えられる．

我々は，Concept Extraction 攻撃手法として Generative Adversarial Networks(GANs) に注目する．GANs は近年の深層学習の分野にて提案されたデータから生成分布を推定する設計手法である．これまで推定の難しかった画像や動画など高次元のデータであっても生成分布の推定ができると言われている [2]．GANs の特徴的な性質として，訓練データに含まれないサンプルを作り出すことができる．具体的には，サンプル間の補間[*1]や，特徴の演算[*2]ができる．

しかしながら，GANs を用いても，予測モデルのみを用いて Concept Extraction 攻撃を行うことは難しい．そこで，攻撃者が予測モデルの入力に対する事前知識として補助データ $\mathcal{D}_{\mathrm{aux}}$ を保有していると想定する．$\mathcal{D}_{\mathrm{aux}}$ が $d_{X|T_{\mathrm{cls}}}$ から得られた場合 (同一分布設定) と，$\mathcal{D}_{\mathrm{aux}}$ が $d_{X|T_{\mathrm{cls}}}$ と異なる分布 $d_{X|T_{\mathrm{aux}}}$ から得られた場合 (相違分布設定) が考えられる．前者は強い攻撃者を想定しており，後者は弱い攻撃者を想定している．各設定について，例を用いて議論する．

同一分布設定の例として，ある研究所のメンバーの顔を識別する予測モデルにおける Concept Extraction 攻撃を考える．攻撃者は研究室のメンバーを識別する予測モデルで

[*1] 2 つの顔画像の中間の顔画像を生成する [3] など
[*2] メガネを掛けた男の画像から男の潜在ベクトルを減算し女の潜在ベクトルを加算するとメガネを掛けた女の画像が得られる [3]

© 2017 Information Processing Society of Japan

あることを事前知識として知っており，加えて各メンバーの顔写真などを撮り補助データを作成すると仮定する．このとき，得られた補助データは $d_{X|T_{\mathrm{cls}}}$ から得られたといえる．攻撃者が攻撃対象のサンプルを得ることができるという設定であるため，現実的な設定であるとは言い難い．

より現実的な設定として，相違分布設定を考える．例として，同様に研究所のメンバーの顔識別する予測モデルにおける Concept Extraction 攻撃を考える．攻撃者は，各メンバーの顔写真などを撮るなどはできないが予測モデルが顔を識別することは知っていると仮定し，攻撃者は研究所メンバーの含まれていない一般公開されている顔画像データなどを補助データとして用いると仮定する．このとき，得られた $\mathcal{D}_{\mathrm{aux}}$ は $d_{X|T_{\mathrm{cls}}}$ と異なる分布 $d_{X|T_{\mathrm{aux}}}$ から得られたと考えられる．このような一般に利用可能なデータを用いた攻撃はより現実的な設定である．本研究では，同一分布設定と相違分布設定の両方について議論を行う．

1.1 本研究の貢献

本研究の貢献は以下のとおりである．

- まず，予測モデルと補助データから訓練データ生成分布を推定する攻撃 (Concept Extraction 攻撃) の定式化を行う．この際に，攻撃者の用いる補助データが訓練データ生成分布と同じ分布から得られる場合と得られない場合双方を想定できるようにしている．(4 章)

- Concept Extraction 攻撃のアルゴリズムとして，Generative Adversarial Network を利用した攻撃手法として PreImageGAN を提案する．この提案手法は攻撃者の用いる補助データが訓練データと同じ分布から得られなくとも，予測モデルの情報を活用し，推定することができるよう設計されている．(5 章)

- 以上の提案手法に対して，EMNIST と MNIST を利用した評価実験を行い，攻撃者が攻撃対象のラベルのサンプルを知識として全く持っていない場合であっても，つまり攻撃者の用いる補助データが訓練データと異なる分布から得られたとしても，攻撃対象のラベルのサンプルの生成分布を推定できることを示す．(6 章)

2. 関連研究

本節では，関連する攻撃として Model Inversion 攻撃を紹介する．加えて，Concept Extraction 攻撃に関連する解析として深層学習モデルの内部表現の可視化の研究を紹介する．しかしながら，Concept Extraction 攻撃のような，学習済み予測モデルと補助データから訓練データ生成分布を推定する攻撃について未だ研究されていない．

2.1 Model Inversion 攻撃

Fredrikson らは，予測モデルと予測値から秘匿情報である入力値の推定を行う Model Inversion 攻撃 (MI 攻撃) を議論した [4], [5]．彼らは遺伝情報から血液凝固剤の投与量を推定するケースに注目し，予測値である投与量と予測モデルから秘匿情報である遺伝情報が推定できる危険性を示した [4]．また，彼らは決定木などにおいても予測値から入力値が推定できる危険性のあることを示した [5]．MI 攻撃では，入力 $\mathbf{x}$ が秘密情報であり，攻撃者は入力の分布 $\Pr[X]$ と予測モデル f を事前知識として知っている．攻撃者の目的は予測値 $y = f(\mathbf{x})$ を与えた場合の入力の条件付き分布 $\Pr[X|Y = y]$ を推定し尤らしい $\mathbf{x}$ を得ることである．

MI 攻撃と Concept Extraction 攻撃との違いとして，攻撃者の事前知識の違いが挙げられる．MI 攻撃では攻撃者が $\Pr[X]$ を知識として持っている．これは，彼らの注目した遺伝情報は一般に低次元であり $\Pr[X]$ が推定可能であり，特に遺伝情報は統計値として $\Pr[X]$ が公開されているためである．しかしながら，画像など入力ドメインは非常に高次元である場合，攻撃者が $\Pr[X]$ が知っているとは考えにくい．Alice の顔の例を用いると，攻撃者が顔画像の分布 $\Pr[X]$ を既知とすると，任意の人間の顔画像を生成することができ，Alice の顔画像も生成可能という設定となり現実的ではない．相違分布設定における Concept Extraction 攻撃では，攻撃者は $\Pr[X]$ は未知とし，代わりに訓練データと異なる分布から得られた補助データを活用する．これは Alice の含まれていないデータから Alice の顔画像の分布を推定することに対応する．画像など入力が高次元である場合，Concept Extraction 攻撃がより適切な問題設定であるといえる．以上より MI 攻撃は Concept Extraction 攻撃と関連はあるが異なる問題設定である．

2.2 深層学習モデルの学習する特徴の解析

学習された深層学習モデルがどのような特徴を予測に用いているのかを可視化する研究が，深層学習の分野において行われている．これらの研究は，深層学習モデルの予測が内部的にどのように行われるかの解析を目的とし，深層学習モデルの獲得している表現や畳み込み層の各フィルターの役割などの解析を行なっている [6], [7]．彼らの研究は，予測モデルが訓練データのどのような特徴を掴んでいるかを知るという観点では関連はあるが，予測モデルから未知の訓練データ生成分布を推定するなどセキュリティやプライバシーの文脈ではない．

3. Generative Adversarial Networks

Generative Adversarial Networks(GANs) とは，Goodfellow らが提案した生成モデルの設計手法であり，与えたデータの生成分布を推定し，この分布に従ったサンプルを生成することができる [8]．GANs は画像生成タスクのよ

© 2017 Information Processing Society of Japan

うな高次元空間におけるデータの生成分布推定において優れた性能を発揮し，人間の視覚において本物のサンプルと見分けのつかないようなサンプルを生成することができる．本節では，Goodfellow らの提案した GAN （VanillaGAN），分布間の距離として Wasserstein Distance に注目した Wasserstein GAN （WGAN），条件付きなどの事前知識を盛り込める conditional GAN （cGAN），cGAN と WGAN の自然な拡張である cWGAN の 4 つを紹介する．

3.1 Generator と Discriminator

GANs の学習は，Generator と Discriminator の 2 人のプレイヤーからなる minimax ゲームにより定式化される．このゲームはよく偽造コインの例に例え説明される [8]. Generator は偽造コインを作るプレイヤーであり，Discriminator は偽造コインと本物のコインを識別するプレイヤーである．Generator は Discriminator が本物のコインと誤認する偽造コインを作成するよう訓練され，Discriminator は 2 つを識別するように訓練される．このように学習された Generator は本物のコインと見た目の違いのわからない偽造コインを作るようになる．

定式化すると以下のようになる．Generator G は，一様分布などから得た乱数 $\mathbf{z} \sim d_{\mathbf{z}}$ を用い，偽物のサンプル $G(\mathbf{z})$ を生成する．Discriminator D は，データ生成分布 $d_{\mathbf{x}}$ から得られた本物のサンプル $\mathbf{x} \sim d_{\mathbf{x}}$ であるならば 1 を出力するように，偽物のサンプル $G(\mathbf{z})$ であるならば 0 もしくは-1 を出力をするように学習を行う．このように学習することより，任意の $\mathbf{z}$ について Generator の生成するサンプル $G(\mathbf{z})$ が本物のサンプル $\mathbf{x} \sim d_{\mathbf{x}}$ と区別ができなくなるよう G が学習される．$d_{\mathbf{z}}$ における確率変数を Z とすると，$G(Z)$ は Generator の生成するサンプルの分布とみなすことができる．GANs における学習は，Generator の獲得した分布 $G(Z)$ がデータ生成分布 $d_{\mathbf{x}}$ と近くなるよう G を最適化していることと一致することが知られている [8].

3.2 GANs の目的関数とその亜種

Goodfellow らの提案した GAN(VanillaGAN) は，式 1 を最適化する [8]．これは $G(Z)$ と $d_{\mathbf{x}}$ の Jensen Shannon(JS) Divergence を最小化していることが示されている [8].

$$\min_G \max_D - \mathbb{E}_{\mathbf{x}\sim d_{\mathbf{x}}}[\log(1 - D(\mathbf{x}))] - \mathbb{E}_{z\sim d_{\mathbf{z}}}[\log D(G(z))] \tag{1}$$

式 1 では，勾配消失問題や mode collapse と言われる問題が起こりやすい [8], [9]．この解決のため，$G(Z)$ と $d_{\mathbf{x}}$ の Wasserstein Distance を最小化する Wasserstein-GAN(WGAN) が提案された (式 2)[9], [10]．式中の $\|D\|_{\mathrm{L}} \leq 1$ は D が 1-Lipschitz を満たすことを意味する．

$$\min_G \max_{\|D\|_{\mathrm{L}}\leq 1} \mathbb{E}_{\mathbf{x}\sim d_{\mathbf{x}}}[D(\mathbf{x})] - \mathbb{E}_{\mathbf{z}\sim d_{\mathbf{z}}}[D(G(\mathbf{z}))] \tag{2}$$

サンプル $\mathbf{x}$ に関連情報 $\mathbf{c}$ のついたデータ $\{(\mathbf{x}, \mathbf{c}), \cdots\}$ から，$\mathbf{c}$ を条件付きとしたデータ生成分布 $\Pr[X|C = \mathbf{c}]$ を推定するために，Generator と Discriminator に条件ベクトル $\mathbf{c}$ を追加した conditional GAN(cGAN) が提案された [8], [11]．cGAN の活用例として，年齢や性別を $\mathbf{c}$ とし顔画像を $\mathbf{x}$ とし，年齢や性別に対応する顔画像を生成することができるようになる [11], [12].

$$\min_G \max_D - \mathbb{E}_{(\mathbf{x},\mathbf{c})\sim d_{\mathbf{x},\mathbf{c}}}[\log(1 - D(\mathbf{x}, \mathbf{c}))] - \mathbb{E}_{\mathbf{z}\sim d_{\mathbf{z}},\mathbf{c}\sim d_{\mathbf{c}}}[\log D(G(\mathbf{z}, \mathbf{c}), \mathbf{c})] \tag{3}$$

cGAN と WGAN の自然な拡張として，cGAN の損失関数に Wasserstein distance を利用した conditional Wasserstein GAN(cWGAN) を考えることができる [13].

$$\min_G \max_{\|D\|_{\mathrm{L}}\leq 1} \mathbb{E}_{(\mathbf{x},\mathbf{c})\sim d_{\mathbf{x},\mathbf{c}}}[D(\mathbf{x}, \mathbf{c})] - \mathbb{E}_{\mathbf{z}\sim d_{\mathbf{z}},\mathbf{c}\sim d_{\mathbf{c}}}[D(G(\mathbf{z}, \mathbf{c}), \mathbf{c})] \tag{4}$$

本研究では，cWGAN を応用し Concept Extraction 攻撃を行う PreImageGAN を提案する．そのため，次節で Concept Extraction 攻撃の定式化を行う．

4. 問題設定

本研究で扱う Concept Extraction 攻撃を定義する．ラベル集合を $\mathbb{T}$ とし，予測モデルの扱う出力ラベルの集合を $\mathbf{T}_{\mathrm{cls}} \subset \mathbb{T}$ と補助データのラベルの集合を $\mathbf{T}_{\mathrm{aux}} \subset \mathbb{T}$ とする．$X, Y, T_{\mathrm{cls}}, T_{\mathrm{aux}}$ はそれぞれ $\mathbb{X}, \mathbb{Y}, \mathbf{T}_{\mathrm{cls}}, \mathbf{T}_{\mathrm{aux}}$ における確率変数とする．予測モデルのドメイン $\mathbb{F}$ を $\mathbb{X} \to \mathbb{Y}$ とする．$\mathbb{X}$ は予測モデルの入力ドメインであり d 次元の実数ベクトル $\mathbb{R}^d$ とし，$\mathbb{Y}$ は $|\mathbf{T}_{\mathrm{cls}}|$ 次元の確率ベクトル $\Delta^{|\mathbf{T}_{\mathrm{cls}}|}$ とする．

訓練データ生成分布を $d_{X|T_{\mathrm{cls}}} = \Pr[X|T_{\mathrm{cls}}]$ とする．予測モデルの訓練データ $\mathcal{D}_{\mathrm{cls}} = \{(\mathbf{x}, t)|\mathbf{x} \in \mathbb{X}, t \in \mathbf{T}_{\mathrm{cls}}\}$ はラベル付きであり，$d_{X|T_{\mathrm{cls}}}$ から得られるとする．予測モデル f は学習アルゴリズム $\mathcal{T}$ により $f = \mathcal{T}(\mathcal{D}_{\mathrm{cls}})$ により得られる．このとき，f は $\Pr[T_{\mathrm{cls}}|X]$ を学習したと考える．

Concept Extraction 攻撃において，公開者と攻撃者の 2 つのステークホルダーが存在する．

公開者は，$\mathcal{D}_{\mathrm{cls}}$ から $\mathcal{T}$ を用い $f = \mathcal{T}(\mathcal{D}_{\mathrm{cls}})$ を学習し，$f, \mathcal{T}, \mathbf{T}_{\mathrm{cls}}$ を公開する．秘密情報は $\mathcal{D}_{\mathrm{cls}}$ と $d_{X|T_{\mathrm{cls}}}$ である．

攻撃者は，攻撃対象ラベル $t^* \in \mathbf{T}_{\mathrm{cls}}$ が与えられ，f,$\mathcal{T}$,$\mathcal{D}_{\mathrm{aux}}$,$\mathbf{T}_{\mathrm{cls}}$ を知識として持つ．補助データはラベルなしデータセット $\mathcal{D}_{\mathrm{aux}} = \{\mathbf{x}|\mathbf{x} \in \mathbb{X}\}$ とし，$d_{X|T_{\mathrm{aux}}} = \Pr[X|T_{\mathrm{aux}}]$ から得られる．$\mathcal{D}_{\mathrm{aux}}$ はラベルを含まないデータであり，攻撃者はラベルと入力のタプルを直接的な知識として持っていない．加えて $\mathcal{D}_{\mathrm{aux}}$ は $\mathcal{D}_{\mathrm{cls}}$ と独立である．

攻撃者の目的を，t^* に対応する条件付き確率分布 $d_{X|T_{\mathrm{cls}}=t^*} = \Pr[X|T_{\mathrm{cls}} = t^*]$ を推定することとする．具体的には，攻撃者は条件付き確率分布 $d_{X|T_{\mathrm{cls}}=t^*}$ に従うサンプル $\hat{\mathbf{x}}^*$ を生成するアルゴリズム $\mathcal{A}$ の設計を行う．

© 2017 Information Processing Society of Japan

$$\hat{\mathbf{x}}^* \sim \mathcal{A}(f, \mathcal{T}, \mathcal{D}_{\text{aux}})$$

攻撃者が持つ $\mathcal{D}_{\text{aux}} = \{\mathbf{x}|\mathbf{x} \in \mathbb{X}\}$ はラベルを含まないデータであるため，攻撃者は直接入力 $\mathbf{x}$ とラベル $t \in \mathbf{T}_{\text{cls}}$ の対応関係を知識として持っていない．しかし，攻撃者は $\Pr[T_{\text{cls}}|X]$ を学習した f を利用することができる．攻撃者が f を利用し $d_{X|T_{\text{cls}}=t^*}$ が推定できたのであれば，f の学習した t^* に対応するサンプルの分布 (concept) を f から抽出することと捉えることができる．このため，この攻撃を Concept Extraction 攻撃と命名する．

4.1 攻撃の評価

Concept Extraction 攻撃において，攻撃者が攻撃を成功させたかの評価するのは難しい．攻撃の成功度合いの一つの指標として，推定された生成分布 $d_{X|T_{\text{cls}}=t^*}$ と真の生成分布 $d_{X|T_{\text{cls}}=t^*}$ の距離が挙げられる．しかし実データにおいて生成分布 $d_{X|T_{\text{cls}}=t^*}$ は未知であり，距離を評価することができない．この問題は GANs の性能評価と関連する問題である．現在，GANs の性能評価は見た目による評価で行なっており，評価指標の開発は open problem である [2].

Concept Extraction 攻撃のリスクの評価という観点では，攻撃者が得た特徴がリスクとなりうる特徴であるか否かを評価することもできる．よってこのリスクを評価するには，実際に $d_{X|T_{\text{cls}}=t^*}$ を推定しこの分布に従うサンプル $\hat{\mathbf{x}}^*$ を生成し，攻撃者がどのような特徴が推定可能かを人間の目により判断するほかない．

4.2 Concept Extraction 攻撃が容易な場合

Concept Extraction 攻撃の困難さは $\mathcal{A}, f, \mathbf{T}_{\text{cls}}, \mathbf{T}_{\text{aux}}$ に依存している．特に f が多くの特徴を利用している場合，$\mathbf{T}_{\text{aux}}$ のサンプルから $\mathbf{T}_{\text{cls}}$ のサンプルの特徴が推定しやすい場合，攻撃が容易であると考えられる．

f が予測を行う際に活用する特徴によって，Concept Extraction 攻撃により得られるサンプル $\hat{\mathbf{x}}^*$ が大きく異なる．例えば，物体認識モデル f が人間・車・飛行機を識別するモデルとしたとする．このとき肌色の車や飛行機は少ないため，f は入力画像が肌色であるならば人間と識別するように学習されることがありうる．この f を攻撃者が人間を攻撃対象ラベルとして Concept Extraction 攻撃しても，攻撃者は人間が肌色であることしかわからず，手足や口を持つことなど他の特徴はわからない．これは，攻撃者が t^* に対応する特徴を f からのみ得られるため，f が一部の特徴しか捉えていないときその特徴以外の特徴を攻撃者が知ることはできないためである．

逆に，Concept Extraction 攻撃が比較的容易な場合として，f が多くの特徴を予測に利用している場合が考えられる．具体的には，予測ラベルが似ている場合，予測ラベルが膨大である場合，予測の正確さ (confidence) も学習している場合が挙げられる．予測ラベルが似ている場合として，犬・猫・山羊の識別モデルを考えた時，f は顔の配置・毛並みなど細部の特徴を使わなければ正しい予測ができず，f に細部の特徴が予測に利用される．同様に，予測ラベルが膨大である場合も細部の特徴を見なければ正しい予測ができないため，f に細部の特徴が学習されると考えられる．予測の正確さ (confidence) も学習する場合として，予測ラベルに該当なし (not applicable) を含めることを考える．このとき，人間・車・飛行機の例における肌色の特徴のような部分的な特徴のみを学習していては該当なしを予測することができないため，f はより多くの特徴を学習し攻撃が容易になると考えられる．

$\mathbf{T}_{\text{aux}}$ のサンプルから $\mathbf{T}_{\text{cls}}$ のサンプルの特徴が推定しやすい場合，Concept Extraction 攻撃は容易であると考えられる．Concept Extraction 攻撃では，補助データ $\mathcal{D}_{\text{aux}}$ のサンプルをモンタージュ写真のように合成し，攻撃対象ラベル t^* の特徴を合成し，$d_{X|T_{\text{cls}}=t^*}$ を推定することを考えている．このため，攻撃対象ラベル t^* の特徴と似たデータの含まれる $\mathcal{D}_{\text{aux}}$ からは，t^* の特徴を合成することができ，$d_{X|T_{\text{cls}}=t^*}$ が容易に推定できると考えられる．例えば，手書き数字識別モデル f を考え，数字を攻撃対象ラベルとし Concept Extraction 攻撃を行うことを考える，このとき $\mathbf{T}_{\text{cls}}$ は数字である．攻撃者が補助データとして，英字を持っている場合 ($\mathbf{T}_{\text{aux}}$ が英字)，漢字を持っている場合 ($\mathbf{T}_{\text{aux}}$ が漢字) を考える．このとき，補助データが英字であるほうが，Concept Extraction 攻撃が容易であると考えられる．これは，英字のほうが数字の持つ曲線や形状と似たサンプル (0 と O，1 と l，2 と z など) が含まれており，漢字には含まれていないからである．

5. 提案手法: PreImageGAN

攻撃者が $d_{X|T_{\text{cls}}=t^*}$ を推定する手法として，cGAN と WGAN を応用した PreImageGAN を提案する．

PreImageGAN はノイズ $\mathbf{z}$ と予測値 $\mathbf{y}$ と引数にとり，条件付き確率分布 $\Pr[X|Y]$ に従ったサンプルを生成するように訓練する．具体的には，任意の $\mathbf{z}, \mathbf{y}$ を与えたとき，生成されたサンプル $\hat{\mathbf{x}}_y = G(\mathbf{z}, \mathbf{y})$ は $\mathbf{y}$ において f の逆像の一つとなるようにつまり $\mathbf{y} \simeq f(\hat{\mathbf{x}}_y)$ となるように訓練される．$\Pr[X|Y]$ が推定できれば，容易に $\Pr[X|T = t^*]$ を得ることもできる．なぜならば，t^* に対応する要素が 1 である確率ベクトル $\mathbf{y}_{t^*} \in \mathbb{Y}$ を考えれば，$\Pr[X|Y = \mathbf{y}_{t^*}]$ は $\Pr[X|T = t^*]$ と一致するからである．

PreImageGAN の目的関数は式 5 である．

$$\begin{aligned}\min_G \max_{\|D\|_{\text{L}} \le 1} & \mathbb{E}_{\mathbf{x} \sim d_{\mathbf{x}}} [D(\mathbf{x}, f(\mathbf{x}))] \\ & - \mathbb{E}_{z \sim d_{\mathbf{z}}, \mathbf{y} \sim d_{\mathbf{y}}} [D(G(z, \mathbf{y}), f(G(z, \mathbf{y})))] \\ & + \lambda \mathbb{E}_{z \sim d_{\mathbf{z}}, \mathbf{y} \sim d_{\mathbf{y}}} [l(f(G(z, \mathbf{y})), \mathbf{y})]\end{aligned} \quad (5)$$

© 2017 Information Processing Society of Japan

最初の 2 項は conditional Wasserstein GAN の項である．これらの項は $\mathbf{x}$ が与えられ $f(\mathbf{x})$ を関連情報 $\mathbf{y}$ としたとき，$\mathbf{y}$ を条件として $\mathbf{x}$ が本物か偽物かを判別するよう disriminator に学習させる．$\mathbf{y}$ が全く異なる場合は，本物か偽物かの判別も異なるように学習される．generator も同様に $\mathbf{y}$ を条件として与えるため，異なる $\mathbf{y}$ が与えられた時生成するサンプルも異なるように学習される．

第 3 項は，generator が $\mathbf{y}$ を与えられた時生成されたサンプルの予測結果 $f(G(\mathbf{z},\mathbf{y}))$ が近くなるようにするための項である．本稿では，負の cosine similarity を用いる．λ はこの項の強さを調節するパラメータである．

6. 実験

提案手法が学習済み予測モデルから補助データを用い特徴を抽出できていることを実験的に示す．具体的には，手書き数字英字データである MNIST と EMNIST を用いて以下のことを示す．

- 攻撃者は，ラベルを直接的に持っておらず，予測モデル f とラベルなしデータ $\mathcal{D}_{\text{aux}} = \{\mathbf{x}|\mathbf{x} \in \mathbb{X}\}$ という間接的なラベルの情報しか持っていない．予測モデルの獲得したラベルに関する情報を用い，ラベルに対応するサンプルの生成分布を推定できるか？ (実験 1)

- 攻撃者は，予測ラベル $\mathbf{T}_{\text{cls}}$(大小英字数字) の部分的なラベル (大小英字) の補助データしか持っていないとする．攻撃者は補助データに含まれていないラベル (数字) の生成分布を推定できるか？ (実験 2)

- 攻撃者は，予測ラベル $\mathbf{T}_{\text{cls}}$(数字) と共通のラベルが存在しないラベル (大小英字) の補助データしかもっていない ($\mathbf{T}_{\text{cls}} \cap \mathbf{T}_{\text{aux}} = \emptyset$)．このとき，$\mathbf{T}_{\text{cls}}$ に含まれるラベル (数字) の生成分布を推定できるか？ (実験 3)

6.1 実験設定

本実験では，表 1 のように $\mathbf{T}_{\text{cls}}, \mathbf{T}_{\text{aux}}$ の設定する．4.2 節で議論したように，$\mathbf{T}_{\text{cls}}, \mathbf{T}_{\text{aux}}$ の設定により問題の困難性が大きく異なる．実験 1 では $\mathbf{T}_{\text{cls}} = \mathbf{T}_{\text{aux}}$ であるため実験の中では最も攻撃が容易であると考えられ，次に実験 2 は $\mathbf{T}_{\text{aux}} = \mathbf{T}_{\text{cls}} \setminus$ 数字 であり次に容易，実験 3 は共通のラベルが存在しない ($\mathbf{T}_{\text{cls}} \cap \mathbf{T}_{\text{aux}} = \emptyset$) ため最も攻撃が難しいと考えられる．加えて，$\mathcal{D}_{\text{cls}}$ と $\mathcal{D}_{\text{aux}}$ に同一のサンプルが含まれないようにする．サンプル数 $|\mathcal{D}_{\text{cls}}|$ は実験 1,2 では 116,323，実験 3 では 10,000 である．同様に $|\mathcal{D}_{\text{aux}}|$ は実験 1 では 697,932，実験 2,3 では 352,897 である．

本実験において，$d_{\mathbf{z}}$ は 128 次元の-1 から 1 の一様分布とする．$d_{\mathbf{y}}$ は $\{f(\mathbf{x})|\mathbf{x} \in \mathcal{D}_{\text{aux}}\}$ においてカーネル密度推定 (バンド幅:0.01, カーネル関数:ガウシアン) した分布とする．パラメータ λ は 1 に設定し，最適化には Adam を利用した．ネットワーク構造は表 2 のようなニューラルネットワークモデルを利用した．f, G の最終層以外の層の活性化関数は ReLU を使用し活性化関数の前に batch normalization を行なっている．D の最終層以外の層の活性化関数は LeakyReLU を使用し活性化関数の前に batch normalization を行なっている．f の最終層の活性化関数は softmax，G は sigmoid，D は線形関数を利用している．

実験 1,2 で使用した f は $\mathcal{D}_{\text{cls}}$ により学習され，EMNIST においてテスト精度 (test accuracy) は 0.8780 であった．実験 3 の f は MNIST において精度は 0.9853 であった．

表 1 実験設定

	予測ラベル: $\mathbf{T}_{\text{cls}}$	$\mathcal{D}_{\text{aux}}$ に含まれるサンプルのラベル: $\mathbf{T}_{\text{aux}}$
実験 1	大小英字, 数字	大小英字, 数字
実験 2	大小英字, 数字	大小英字のみ
実験 3	数字	大小英字のみ

表 2 実験で使用したネットワークアーキテクチャ．Conv$N$/Deconv$N$ は N 個のフィルターを持ち，kernel size は 5x5, stride は 2 である畳み込み層/デコンボリューション層を意味する．FCN は出力次元 N の全結合層を示し，Reshape は 6272 次元のベクトルを $7 \times 7 \times 128$ のテンソルに変形する層である

	f	G		D	
入力	$\mathbf{x}$	$\mathbf{z}$	$\mathbf{y}$	$\mathbf{x}$	$\mathbf{y}$
1 層目	Conv64	Concat		Conv64	FC196|$\mathbf{T}_{\text{cls}}$|
2 層目	Conv128	FC1024		Concat	
3 層目	Dropout0.5	FC6272		Conv128	
4 層目	FC1024	Reshape		Conv256	
5 層目	Dropout0.5	Deconv128		Conv1024	
6 層目	FC|$\mathbf{T}_{\text{cls}}$|	Deconv64		FC256	
7 層目		Deconv1		FC1	

6.2 実験結果

実験結果を図 2 に示す．図上段 (補助データ $\mathcal{D}_{\text{aux}}$) は，攻撃者が利用した補助データである．図中段 (生成結果 $\mathbf{y}: 0, \cdots, 9$) は，$\mathbf{x}$ はランダムに，$\mathbf{y}$ は対応するラベルの値を 1 とし他を 0 と設定し生成したサンプル $G(\mathbf{x},\mathbf{y})$ である．図下段 (生成結果 $\mathbf{y}$: ランダム) は，それぞれ $d_{\mathbf{x}}, d_{\mathbf{y}}$ からランダムに $\mathbf{x},\mathbf{y}$ を設定し生成したサンプル $G(\mathbf{x},\mathbf{y})$ である．ランダム生成 (生成結果 $\mathbf{y}$: ランダム) においては数字英字が混じったものが生成されているのが確認できる．これは $\mathbf{T}_{\text{aux}}$ が大小英字数字を含むため，大小英字数字の分布を PreImageGAN が学習したため妥当な結果である．$\mathbf{y}$ を各ラベル t^* に設定した場合 ($\mathbf{y}: 0, \cdots, 9$)，ほぼそれぞれのラベルに対応した数字が生成されているのがわかる．

実験 2,3 共に，ランダム生成 (y : ランダム) においては，英字が生成されているのが確認できる．これは双方 $\mathbf{T}_{\text{aux}}$ が英字のみであるため，PreImageGAN は英字の分布を推定したため妥当な結果である．$\mathbf{y}$ を各ラベルに設定した場

© 2017 Information Processing Society of Japan

合，それぞれの数字に似た文字が生成されている．実験 2 では，数字に近いサンプルが生成され，数字の生成分布が推定できたと考えられる．このとき，攻撃者は補助データに数字のサンプルを持っておらず，英字のデータしか持っていない．よって，英字の補助データを利用し，攻撃者は未知の Concept を f から抽出できたといえる．実験 3 では，数字というより英字に近いサンプルが生成されている．英字に見えるサンプルにおいても f はほぼ確率 1 で対象ラベル t^* と予測されたため，英字に見えるサンプルと数字に見えるサンプルを f が区別できていない，4.2 節で議論した f が部分的な特徴を捉えている場合であると考えられる．しかしながら，いくつか数字と酷似したサンプルが生成されている．これらのサンプルは，英字から数字の特徴を推定し生成されたサンプルであり，攻撃者が訓練データ (数字) と似ているが全く異なるデータ (英字) からでも，最悪ケースにおいては数字という概念的特徴が推定できることを示唆している．

実験 2,3 の比較として，実験 2 は実験 3 よりも予測ラベル $\mathbf{T}_{\text{cls}}$ の個数を多く，$\mathbf{T}_{\text{cls}}$ 以外の条件は同一である．このとき実験 2 のサンプルはより数字に見える．これは，4.2 節で議論したように，予測ラベルが多くなると予測モデルが多くの特徴を獲得するため，攻撃が容易になるためと考えられ，4.2 節の議論の一部が実験的に示せたといえる．

以上のように，PreImageGAN を用いることで英字のデータから数字の特徴を予測モデル f から抽出する Concept Extraction 攻撃は可能である．加えて，4.2 節で議論したように，予測モデルや攻撃者の事前知識により Concept Extraction 攻撃の難度が異なることを実験的に示した．

7. 結論

本稿では，学習済み予測モデルから特徴を抽出する Concept Extraction 攻撃を定義した．Concept Extraction 攻撃は，訓練データ生成分布 (concept) を秘密情報としており，この分布に従う訓練データで学習した予測モデルから，補助データを用い訓練データ生成分布を推定する攻撃である．我々は GANs と呼ばれる深層学習アーキテクチャに注目し，Concept Extraction 攻撃を行う PreImageGAN を提案した．PreImageGAN に対して，EMNIST と MNIST を利用した評価実験を行い，攻撃者が予測ラベルのサンプルを知識として全く持っていない場合であっても，補助データを活用することにより，知識として持たないラベルのサンプルの生成分布を推定することができることを示した．

今後の発展として，より実際のサービスで使われるデータに近いケースを想定し実験を行おうと考えている．

謝辞 本研究は科学研究費 16H02864 の助成を受けました．加えて，実験のための計算資源として株式会社ドワンゴの GPU サーバーである紅莉栖を利用させて頂けたことに感謝致します．

参考文献

[1] Goodfellow, I., Bengio, Y. and Courville, A.: Deep Learning, MIT Press (2016). http://www.deeplearningbook.org.

[2] Goodfellow, I. J.: NIPS 2016 Tutorial: Generative Adversarial Networks, CoRR, Vol. abs/1701.00160 (online), available from ⟨http://arxiv.org/abs/1701.00160⟩ (2017).

[3] Radford, A., Metz, L. and Chintala, S.: Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks, CoRR, Vol. abs/1511.06434 (online), available from ⟨http://arxiv.org/abs/1511.06434⟩ (2015).

[4] Fredrikson, M., Lantz, E., Jha, S., Lin, S., Page, D. and Ristenpart, T.: Privacy in Pharmacogenetics: An End-to-End Case Study of Personalized Warfarin Dosing, 23rd USENIX Security Symposium (USENIX Security 14), San Diego, CA, USENIX Association, pp. 17–32 (2014).

[5] Fredrikson, M., Jha, S. and Ristenpart, T.: Model Inversion Attacks That Exploit Confidence Information and Basic Countermeasures, Proceedings of the 22Nd ACM SIGSAC Conference on Computer and Communications Security, CCS '15, New York, NY, USA, ACM, pp. 1322–1333 (online), DOI: 10.1145/2810103.2813677 (2015).

[6] Mahendran, A. and Vedaldi, A.: Understanding deep image representations by inverting them, IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015, pp. 5188–5196 (online), DOI: 10.1109/CVPR.2015.7299155 (2015).

[7] Mahendran, A. and Vedaldi, A.: Visualizing Deep Convolutional Neural Networks Using Natural Pre-images, International Journal of Computer Vision, Vol. 120, No. 3, pp. 233–255 (online), DOI: 10.1007/s11263-016-0911-8 (2016).

[8] Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y.: Generative Adversarial Nets, Advances in Neural Information Processing Systems 27 (Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N. D. and Weinberger, K. Q., eds.), Curran Associates, Inc., pp. 2672–2680 (online), available from ⟨http://papers.nips.cc/paper/5423-generative-adversarial-nets.pdf⟩ (2014).

[9] Arjovsky, M. and Bottou, L.: Towards Principled Methods for Training Generative Adversarial Networks, ArXiv e-prints (2017).

[10] Arjovsky, M., Chintala, S. and Bottou, L.: Wasserstein GAN, ArXiv e-prints (2017).

[11] Mirza, M. and Osindero, S.: Conditional Generative Adversarial Nets, CoRR, Vol. abs/1411.1784 (online), available from ⟨http://arxiv.org/abs/1411.1784⟩ (2014).

[12] Gauthier, J.: Conditional generative adversarial nets for convolutional face generation, Class Project for Stanford CS231N: Convolutional Neural Networks for Visual Recognition, Winter semester (2014).

[13] fairytale0011: fairytale0011/Conditional-WassersteinGAN: Tensorflow implementaion of a conditional Wasserstein GAN, https://github.com/fairytale0011/Conditional-WassersteinGAN. Accessed: 2017-08-24.

© 2017 Information Processing Society of Japan

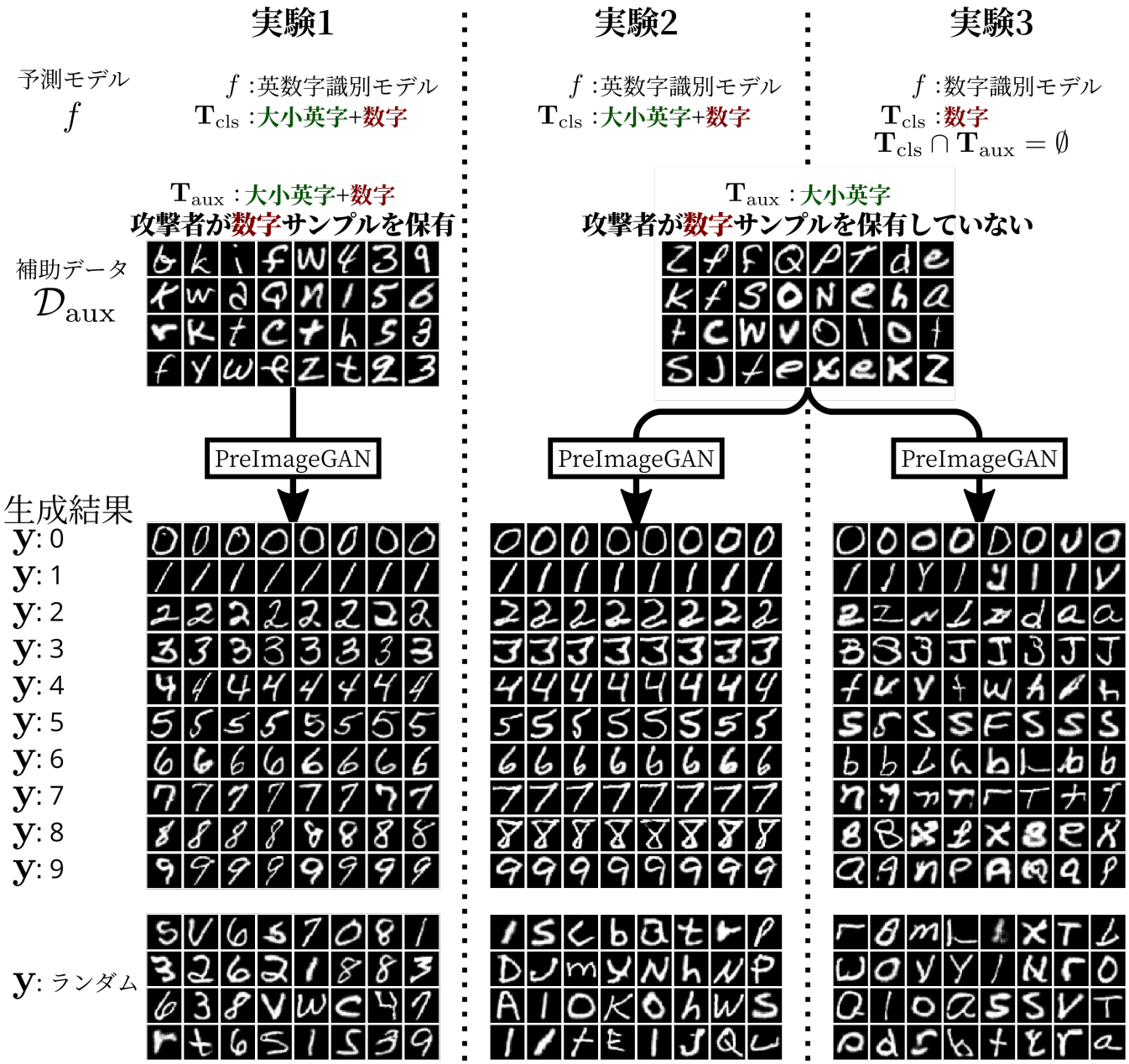

図 2 **実験結果**．数字を攻撃対象ラベルとし，攻撃者の事前知識である補助データ $\mathcal{D}_{\mathrm{aux}}$ と予測モデル f を各実験で変化させ，PreImageGAN で攻撃対象ラベルの生成分布の推定を行なった．$\mathbf{y}$ は PreImageGAN に与えるベクトルである．$\mathbf{y}:0$ は，予測モデルが 0 と識別するようサンプルを生成したことを示す．実験 1 は同一分布設定であり，実験 2,3 は相違分布設定である．

実験 1,2 において，$\mathbf{y}$ で指定したラベルに対応する数字に見えるサンプルが生成されている．このことから，実験 1 からは，攻撃者が直接ラベルを保有していなくとも f があれば，攻撃対象ラベルの生成分布が推定できることがわかる．実験 2 では，攻撃者が攻撃対象ラベル (数字) のサンプルを保有しておらずとも，攻撃対象ラベルの生成分布が推定できていることがわかる．実験 3 では，攻撃対象ラベル (数字) に見えるサンプルと英字に見えるサンプルが混在している．このとき，ほぼすべてのサンプルにおいて f は対応するラベルと識別しており，この f は数字のようなサンプルと英字のようなサンプルが区別できていない．これは，実験 1,2 より少ないラベルを識別する予測モデルを用いており，f が十分な特徴を捉えていないため，加えて攻撃者の補助データのラベルと予測ラベルが互いに素 ($\mathbf{T}_{\mathrm{cls}} \cap \mathbf{T}_{\mathrm{aux}} = \emptyset$) であったため，攻撃が難しかったためと考えられる．実験 3 のような攻撃が難しい場合においても，攻撃対象ラベル (数字) と似たサンプルが生成されており，最悪ケースを考えれば数字という概念的特徴が推定可能であるといえる．

© 2017 Information Processing Society of Japan