社団法人　電子情報通信学会
THE INSTITUTE OF ELECTRONICS,
INFORMATION AND COMMUNICATION ENGINEERS

信学技報
TECHNICAL REPORT OF IEICE.

[サーベイ論文] Adversarial Training

足立　浩規†　平川　翼†　山下　隆義†　藤吉　弘亘†

† 中部大学　〒487–8501 愛知県春日井市松本町 1200
E-mail: †ha618@mprg.cs.chubu.ac.jp, ††{hirakawa,takayoshi,fujiyoshi}@isc.chubu.ac.jp

あらまし　Adversarial training (AT) は悪意のある摂動を付与したサンプル (AEs: Adversarial examples) を学習に使用して，攻撃に頑健なモデルの獲得を目的とした学習方法である．AT は AEs に対するモデルの頑健性能を向上させる一方で，通常のサンプルに対する分類精度を大幅に劣化させる性質がある．この問題を解消するために，様々な観点からアプローチした手法が数多く提案されている．本稿では AT についてサーベイし，AT の研究動向について体系的にまとめる．また，代表的な手法に関して，データセットやモデルなどを統一して分類精度の評価および比較をする．さらに，各手法を適用したモデルの低次元特徴空間を可視化しつつ，特徴空間の定量的評価指標を用いて比較をする．
キーワード　深層学習, 敵対的サンプル, 敵対的学習, サーベイ

Adversarial Training: A Survey

Hiroki ADACHI†, Tsubasa HIRAKAWA†, Takayoshi YAMASHITA†, and Hironobu FUJIYOSHI†

† Chubu University　1200 Matsumoto, Kasugai, Aichi, 487-8501 Japan
E-mail: †ha618@mprg.cs.chubu.ac.jp, ††{hirakawa,takayoshi,fujiyoshi}@isc.chubu.ac.jp

Abstract　Adversarial training (AT) is a training method that aims to obtain a robust model for defencing the adversarial attack by using adversarial examples (AEs). Although AT improves the robustness of the model to AEs, it significantly decreases the classification accuracy to natural samples. To overcome this problem, researchers proposed methods that approached from several perspectives. In this paper, we survey AT and systematically summarize about research trends of AT. Furthermore, we evaluate and compare the classification accuracy with the exact experimental details for the typical methods. Moreover, we visualize the low dimensional feature space of the model applied to each method and evaluate the feature representation using some quantitative evaluation indices.
Key words　Deep learning, Adversarial examples, Adversarial training, Survey

1. はじめに

コンピュータビジョン分野では，畳み込みニューラルネットワーク (CNN) の著しい発展によって優れた画像分類が可能である [1]．CNN は画像分類だけでなく，物体検出 [2] やセマンティックセグメンテーション [3]，画像生成 [4] などの様々な分野で活用されており，優れた性能を実現している．しかしながら，CNN は人が知覚困難な微小摂動を付与した画像 (AEs: Adversarial Examples) [5] に対して脆弱であることが知られている．CNN の脆弱性に対する攻撃は，一般に，Adversarial attack と呼ばれている．

Adversarial attack は，攻撃対象モデルの情報が全て既知である White-box attack と，入出力以外の情報が全て未知である Black-box attack の 2 つに大別できる．White-box attack に含まれる攻撃手法は，攻撃対象モデルの重みパラメータをもとに分類誤差が最大となる方向ベクトルを入力画像に対して求める．一方，Black-box attack は攻撃対象以外のモデルから求めた摂動を利用する方法や，攻撃対象の入出力の関係からモデルを近似的に求める方法などが利用されている．これらの攻撃によって求めた AEs は，モデルの予期せぬ誤分類を誘発することができる．また，AEs は画像分類や認識だけでなく，セマンティックセグメンテーション [6] や物体検出 [7]，強化学習 [8] など様々なタスクにおいて予測できない挙動を誘発することができる．そのため，Adversarial attack は深層学習をベースとしたアプリケーションのセキュリティの観点で問題視されており，攻撃リスクを緩和するための研究が盛んにされている．本稿では，画像分類及び認識に関するサーベイを対象とするため，画像分類以外のタスクに対する攻撃の詳細は [9] [10] を参照されたい．

AEs に対する防御手法は，Adversarial defense と呼ばれており，入力画像を操作するアプローチと AEs を学習に使用するアプローチの 2 つに大別できる．入力画像を操作する方法は，ノイズを除去した画像を CNN へ入力する Denoising [11] [12] と

AEs か否かを検出する Detection に分類される．Denoising 手法の多くは，画像生成モデル [13] [14] を用いて摂動の影響を緩和する．Detection では，AEs と判定されたサンプルは摂動の影響を緩和するような画像処理を適用した後に CNN によって分類する．従って，Detection は Denoising と組み合わせた手法と捉えることができる．一方，AEs を学習に使用する方法は，Adversarial training (AT) と呼ばれており，AEs を上手く分類できるような識別境界の獲得を目的としている．本稿は AT を対象としたサーベイであるため，その他のアプローチに関する議論はしないことに注意されたい．

AT は，AEs をベースに重みパラメータを更新することで，AEs による攻撃に頑健なモデルを獲得することができる一方，通常のサンプル（以降，誤解がない限り，本稿では Natural と呼称する）に対する分類性能が著しく劣化する．これはモデルが AEs に合わせた識別境界を学習したため，本来分類できる多くのサンプルに対して誤分類が生じることが原因である．この問題点に対処するために，理論的な証明や様々な実験をもとに，これまでに数多くの手法が提案されている．

AT は，学習中の摂動の求め方に着目した手法と，識別境界に着目した損失関数の設計や学習方法の 2 つに大別することができる．摂動の求め方を改善する方法の多くは，Madry らが提案した Projected Gradient Descent (PGD) [15] よりも強い摂動を学習することで，あらゆる攻撃に頑健なモデルの獲得を可能としている．一方，損失関数や学習方法を改善する方法では，識別境界からサンプルが離れるような損失関数の追加や，各サンプルの誤差に対して重み付けをすることで Adversarial attack に頑健にしつつ，Natural の性能低下を緩和している．従って，Adversarial attack のリスクを軽減するためには，サンプルと識別境界の関係が重要となる．

以上を踏まえて，本稿ではコンピュータビジョン分野の AT についてサーベイし，どのようなアプローチで頑健なモデルの獲得を実現しているかを理解する．本稿は 6 章の章立てで構成する．まず，2 章では代表的な Adversarial attack 手法について述べる．3 章では Adversarial defense の 1 つである AT をいくつかのカテゴリに分類し，様々な手法について述べる．4 章ではモデルの分類精度の算出方法や特徴空間の定量的評価方法について述べる．5 章では代表的な手法の分類性能を算出するだけでなく，特徴空間を定量的に評価する．最後に，6 章で本サーベイをまとめる．

2. Adversarial attack

Adversarial attack は攻撃対象モデルの情報が全て既知な White-box attack と，入出力以外のモデル情報が全て未知な Black-box attack に分類できる．特殊な場合を除いて，現実社会においてはモデル情報が開示されていないため，Black-box attack が現実に近い状況下での攻撃と捉えることができる．一方，White-box attack は CNN の挙動を深く理解するために重要な研究の 1 つとされている．本稿では，White-box attack の代表的な手法を中心にまとめる．

White-box の Adversarial attack は，1 ステップで摂動を求める方法とマルチステップで求める方法に分類できる．本章では，まず 2.1 節で，Adversarial attack の定義をすることによって，AEs をどのようなモチベーションで求めているかについて理解する．2.2 節以降では，それぞれのアプローチに分類して代表的な手法について述べる．

2.1 Adversarial attack の定義

Adversarial attack は，訓練データ集合を $\mathcal{D}$，クラス数を k，分類モデルを $f: \mathbb{R}^d \to \mathbb{R}^k$，分類モデル f の重みパラメータを $\boldsymbol{\theta}$，$\mathcal{D}$ からサンプリングした入力データと教師信号をそれぞれ $\boldsymbol{x}_i \in \mathbb{R}^d$，$\boldsymbol{y}_i \in \{0,1\}^k$ としたとき，以下の式を満たす摂動 $\boldsymbol{\delta}_i \in \mathbb{R}^d$ を導出することが目的である．

$$\boldsymbol{\delta}_i = \underset{\|\boldsymbol{\delta}_i\|_p \leq \epsilon}{\arg\max}\, L(\boldsymbol{x}_i + \boldsymbol{\delta}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) \ \text{ s.t. } f(\boldsymbol{x}_i) \neq f(\boldsymbol{x}_i + \boldsymbol{\delta}_i) \tag{1}$$

ここで，$L(\cdot)$，$\boldsymbol{y}_i$ はそれぞれ，損失関数，正解クラス以外が 0，正解クラスが 1 となるように one-hot 表現したベクトルを表している．損失関数は一般的に，Softmax 関数を $\sigma: \mathbb{R}^k \to (0,1)^k$ として，以下のクロスエントロピー誤差を利用することが多い．

$$L(\mathcal{D}) = -\frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathcal{D}} \boldsymbol{y}_i^\top \log \sigma(f(\boldsymbol{x}_i)) \ \text{ s.t. } \sum_{j=0}^{k-1} \sigma(f_j(\boldsymbol{x}_i)) = 1 \tag{2}$$

AEs は変化量の大きい摂動 $\boldsymbol{\delta}_i$ を使用することで，容易に誤分類を誘発することができるが，基データから大幅な変化をするため人が知覚可能になる．そのため，基データ $\boldsymbol{x}_i$ を中心とした半径 ϵ の l_p 空間で摂動を定義する．l_p-norm は l_2 または l_∞ が多く使用される．ここで，ϵ は極小の値であり，入力データが画像の場合は $[0, 255]$ または $[0, 1]$ の範囲で定義する．

2.2 1 ステップで摂動を求める手法

1 ステップで摂動を求める最も有名な方法として，Fast Gradient Sign Method (FGSM) [16] が提案されている．FGSM はニューラルネットワークが高次元空間で線形的な振る舞いをすることに着目した，モデルの勾配を利用した攻撃手法である．FGSM は式 (3) に示すように，入力データ $\boldsymbol{x}_i$ に関して損失関数を微分することで勾配を求め，l_∞ 空間内に収めるために勾配から符号を抜き出して ϵ を乗じて摂動を算出し，$\boldsymbol{x}_i$ に加算することで AEs を作成する．

$$\tilde{\boldsymbol{x}}_i = \boldsymbol{x}_i + \epsilon \cdot \mathrm{sign}\left(\nabla_{\boldsymbol{x}_i} L(\boldsymbol{x}_i + \boldsymbol{r}, \boldsymbol{y}_i; \boldsymbol{\theta})\right) \tag{3}$$

ここで，$\boldsymbol{r} \sim U[-\epsilon, \epsilon]$ であり，入力データ $\boldsymbol{x}_i$ をランダムな位置へ移動させて勾配計算をする．

FGSM を発展させた手法として，勾配情報を直接加算する Fast Gradient Value Method (FGVM) [17] が提案されている．FGSM で求めた摂動は分類対象以外の領域にも一定の変化が加わる一方，FGVM のように勾配情報を直接利用することで分類対象のみに変化を加えることが可能となる．

式 (3) は摂動を加算することで教師信号 $\boldsymbol{y}_i$ との誤差を最大化するため，どのクラスに誤分類するか一意に決定しない．一方，式 (4) のように，任意のクラス $\boldsymbol{t}_i$ との分類誤差が最小するように摂動を減算することで狙ったクラスへの誤分類を誘発できる．

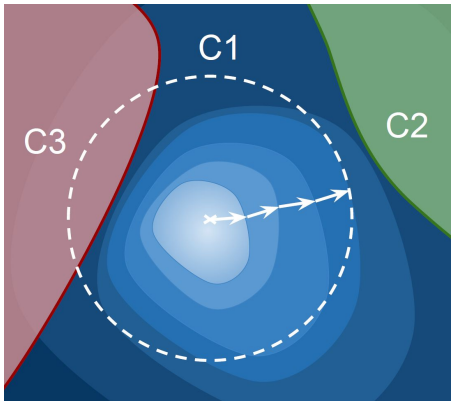


図 1　誤分類する摂動が上手く求めれない例. [22] から引用.

$$\tilde{\boldsymbol{x}}_i = \boldsymbol{x}_i - \epsilon \cdot \mathrm{sign}\left(\nabla_{\boldsymbol{x}_i} L(\boldsymbol{x}_i + \boldsymbol{r}, \boldsymbol{t}_i; \boldsymbol{\theta})\right) \quad \text{s.t. } \boldsymbol{y}_i \neq \boldsymbol{t}_i \tag{4}$$

2.3　マルチステップで摂動を求める手法

Basic Iterative Method (BIM) [18] は最初に提案されたマルチステップで摂動を求める手法である．BIM は式 (5) に示すように，ステップサイズ $\alpha \leq \epsilon$ を用いて，l_p 空間内の Global maxima に到達するように反復して探索する．

$$\tilde{\boldsymbol{x}}_i^{n+1} = \Pi_{\mathcal{B}[\boldsymbol{x}_i^0]}\left(\tilde{\boldsymbol{x}}_i{}^n + \alpha \cdot \mathrm{sign}(\nabla_{\tilde{\boldsymbol{x}}_i^n} L(\tilde{\boldsymbol{x}}_i^n, \boldsymbol{y}_i; \boldsymbol{\theta}))\right) \quad \text{s.t. } n \geq 0 \tag{5}$$

ここで, $\mathcal{X}$ をデータ集合とすると, $\mathcal{B}[\boldsymbol{x}] = \{\tilde{\boldsymbol{x}} \in \mathcal{X} \mid \|\boldsymbol{x} - \tilde{\boldsymbol{x}}\|_p \leq \epsilon\}$ である．従って，$\Pi_{\mathcal{B}[\boldsymbol{x}_i^0]}$ は $\boldsymbol{x}_i^0$ を中心とした l_p 空間外の値を空間内に投影する関数である．FGSM の時と同様に，$\boldsymbol{x}_i^0$ の時は乱数によって始点をランダムに移動させる．Projected Gradient Descent (PGD) [15] は BIM と同じ摂動の求め方だが，PGD で求めた AEs を AT に用いた最初の研究として有名である．

BIM や PGD を拡張することで，さらに強い摂動の導出を試みた研究として，慣性項を追加した Momentum Iterative FGSM (MI-FGSM) [19] や幾何変化を施した多様な画像によって摂動を求める Diverse Inputs I-FGSM [20] などが提案されている．特に，Calini と Wagner によって提案された攻撃 (CW) [21] は，摂動に関して式 (6) の最適化問題を解くことで，PGD よりも強い摂動を l_p 空間内で求めることを可能とした．

$$\min \|\boldsymbol{\delta}_i\|_p + c \cdot \phi(\boldsymbol{x}_i + \boldsymbol{\delta}_i) \tag{6}$$

ここで，c は定数である．損失関数 $\phi(\cdot)$ の設計については著者らが議論しているので，論文を参照されたい．

通常の Adversarial attack では，分類誤差を最大にすることを考えており，図 1 のように，l_p 空間内に異なるクラスが存在しても初期方向次第では騙されないことがある．この問題に対処した手法として，複数のターゲットを設定した Multiple targeted attack [23] が提案されているが，クラス数増加に伴って計算コストも増加する．Sriramanan らは，少ない計算コストで摂動の適切な初期方向を求める Guided Adversarial Margin Attack (GAMA) [22] を提案した．GAMA は，$p(\tilde{\boldsymbol{x}}_i) = \sigma(f(\tilde{\boldsymbol{x}}_i))$ として，以下に示す式を最大化する摂動を求める．

$$L_{\mathrm{GAMA}} = -p_y(\tilde{\boldsymbol{x}}_i) + \max_{j \neq y} p_j(\tilde{\boldsymbol{x}}_i) + \lambda \cdot \|f(\tilde{\boldsymbol{x}}_i) - f(\boldsymbol{x}_i)\|_2^2 \tag{7}$$

GAMA では，摂動導出終盤に向かって λ を小さくし，第 2 項の影響を少なくすることで Global maxima に到達することができる．

2.4　その他の攻撃手法

Croce と Hein は Auto-PGD，PGD Difference of Logits Ratio (PGD-DLR)，Fast Adaptive Boundary attack (FAB) [24]，Square attack [25] の攻撃結果をアンサンブルする AutoAttack [26] を提案した．Auto-PGD は PGD のステップサイズ α をダイナミックに変動させながら摂動の探索をする攻撃である．PGD-DLR は分類誤差を計算するためにクロスエントロピー誤差ではなく，式 (8) ロジット比の差を用いて摂動を求める．

$$\mathrm{DLR}(\boldsymbol{x}_i, \boldsymbol{y}_i) = -\frac{z_y - \max_{i \neq y} z_i}{z_{\pi_1} - z_{\pi_3}} \tag{8}$$

ここで，z，π は，ぞれぞれ，モデルが出力したロジット，z の要素を降順で並べたものを表している．

Adversarial attack は入力画像の各ピクセルに対する攻撃だけでなく，スーパーピクセルに対する攻撃をする Superpixel Attentional version I-FGM (SAI-FGM) [27] や任意の領域に対するパッチベースの攻撃 [28]，差分進化法を用いて画像中の 1 ピクセルのみを攻撃する One-pixel attack [29] など，様々な攻撃アプローチが提案されている．さらに，Unversal Adversarial Perturbation [30] は 1 つの摂動によってあらゆるサンプルの誤分類を誘発することを試みた研究の 1 つである．

3.　Adversarial training

Adversarial training (AT) [15] は AEs に対する防御手法の中で，最もシンプルで効果的な方法とされている．AT は，図 2 に示すように，(1)AEs の作成方法に着目したアプローチと，(2) 識別境界に着目したアプローチに大別することができる．1 つ目に分類される多くの手法は，AEs の導出過程や学習に使用する AEs そのものを直接的に操作する．2 つ目に分類される手法は，識別境界とサンプルの関係を考慮した学習をする．本章では，まず，3.1 節で AT の定義をする．3.2 節以降では，AT を分類しながら各手法について詳細に述べる．

3.1　Adversarial training の定義

AT の学習プロセスは図 3 に示すように，Inner-maximization と Outer-minimization の 2 つに分割して考えることができる．Inner-maximization では，$\mathcal{D}$ からサンプリングしたデータ $\boldsymbol{x}_i$ に対する摂動 $\boldsymbol{\delta}_i$ を教師信号 $\boldsymbol{y}_i$ を用いて PGD で求める．この時，モデルの重みパラメータは更新されないように固定する．一方，Outer-minimization では，算出した AEs を用いて教師信号 $\boldsymbol{y}_i$ との誤差が最小になるよう，重みパラメータを更新する．以上の処理は，$\tilde{\boldsymbol{x}}_i = \boldsymbol{x}_i + \boldsymbol{\delta}_i$ とすると，式 (9) で表すことができる．

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathcal{D}} \left[\max_{\|\boldsymbol{x}_i - \tilde{\boldsymbol{x}}_i\|_p \leq \epsilon} L(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) \right] \tag{9}$$

AT では，Inner-maximization によって，現在のモデルにおける worst-case な AEs を求め，Outer-minimization で誤分類しないように学習する．これを複数回繰り返すことによって頑健なモデルの獲得が可能となる．

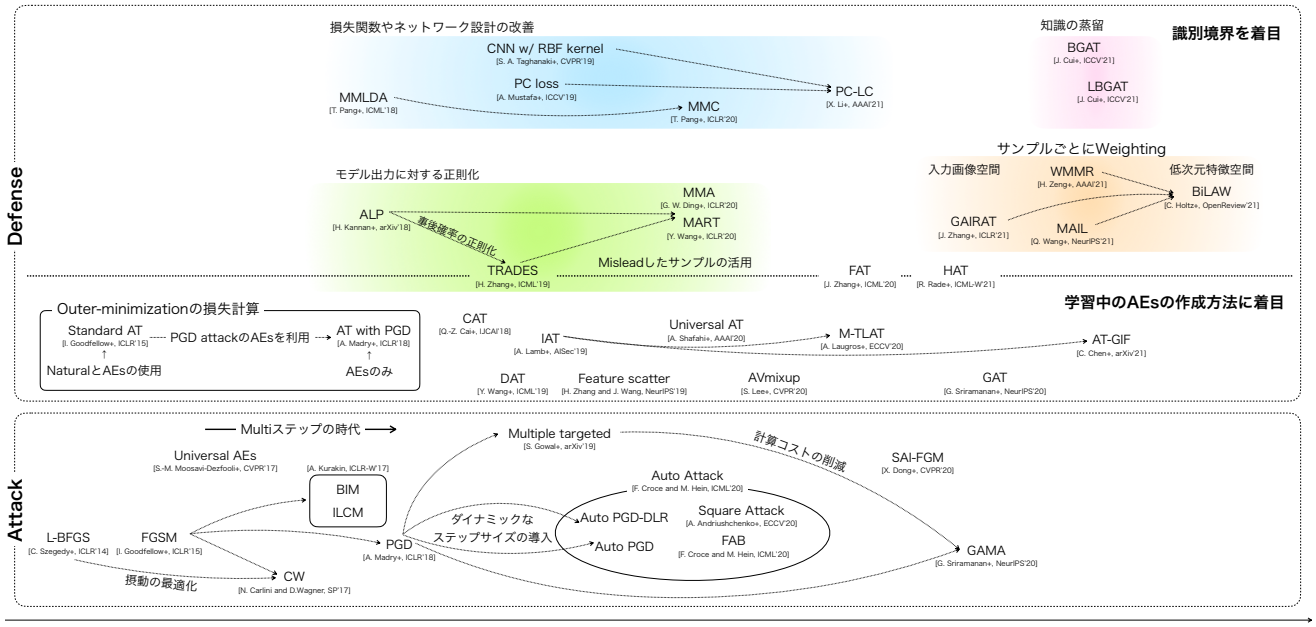


図 2　攻撃手法と Adversarial training の分類

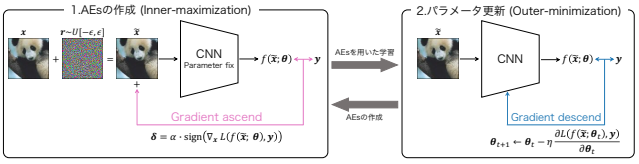


図 3　Adversarial training の概念図

3.2　摂動作成方法に着目した Adversarial training

一般的な AT では，PGD で求めた AEs を用いた最小化問題を解くことで，頑健なモデルを獲得する．しかしながら，Natural に対する分類性能が著しく劣化することが問題になっている．この問題に対処するために，Inner-maximization の改善や Outer-minimization に使用する摂動に改善を施した手法が提案されている．

3.2.1　Curriculum Adversarial Training

Cai ら [31] は，AT を適用したモデルにおいて，学習初期から強い AEs を学習することが Natural に対する性能劣化を生じさせる原因と考えた．そこで，著者らはカリキュラムラーニングを取り入れた Curriculum Adversarial Training (CAT) を提案した．CAT では，学習初期に弱い摂動を加算した AEs を学習し，攻撃に対して高い精度を達成したタイミングで加算する摂動を徐々に強くする．これによって，モデルの能力に応じた AEs を学習できるため，Natural に対する分類性能劣化を緩和すること可能とした．

3.2.2　Adversarial Vertex mixup

Adversarial Vertex mixup (AVmixup) [32] は PGD で求めた AEs を直接学習する代わりに，式 (10) に示すように，求めた摂動 $\boldsymbol{\delta}_i$ に定数 γ を乗じた Adversarial vertex を学習に用いる．

$$\boldsymbol{x}_i^{av} = \boldsymbol{x}_i + \gamma \cdot \boldsymbol{\delta}_i \tag{10}$$

さらに，$\boldsymbol{x}_i^{av}$ と Natural $\boldsymbol{x}_i$ を任意の比率で mixup [33] したサンプルを学習することで，定数倍していない AEs を学習サンプルの 1 つとして学習することができるため，Natural に対する分類性能を劣化させずに頑健なモデルを獲得することを実現した．Natural $\boldsymbol{x}_i$ と $\boldsymbol{x}_i^{av}$ の mixup と，それに対する教師信号を式 (11)(12) に示す．

$$\tilde{\boldsymbol{x}}_i = \alpha \cdot \boldsymbol{x}_i + (1-\alpha) \cdot \boldsymbol{x}_i^{av} \tag{11}$$

$$\tilde{\boldsymbol{y}}_i = \alpha \cdot \psi(\boldsymbol{y}_i, \lambda_1) + (1-\alpha) \cdot \psi(\boldsymbol{y}_i, \lambda_2) \tag{12}$$

ここで，α は Beta$(1,1)$ からサンプリングした確率である．また，λ_1，λ_2 はスムージングパラメータを表しており，スムーズング関数 $\psi : (0,1)^k \to (0,1)^k$ は $\psi(\boldsymbol{y}, \lambda) = \lambda \cdot \boldsymbol{y} + \frac{1-\lambda}{k-1} \cdot (1 - \boldsymbol{y})$ である．

3.2.3　Guided Adversarial Training

Guided Adversarial Training (GAT) [22] は，摂動の適切な初期方向に着目した AT であり，GAMA と同時に Sriramanan らによって提案された．GAT は GAMA の式 (7) の第 1 項を通常のクロスエントロピー誤差に変更して以下の式を用いた Outer-minimization をする．

$$L_{\mathrm{GAT}} = L(\boldsymbol{x}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) + \lambda \cdot \| f(\tilde{\boldsymbol{x}}_i) - f(\boldsymbol{x}_i) \|_2^2 \tag{13}$$

GAMA と異なり GAT では，1 ステップで摂動を求めて AEs を作成するため，基本的に λ の減衰はしない．GAT は 1 ステップの摂動導出であることから，従来の Adversarial training より短い学習時間で優れた頑健性能を獲得することができる．

3.3　識別境界に着目した Adversarial training

本節では，識別境界に着目した AT をさらに細かく分類して，各手法について述べる．

3.3.1　モデルの出力に対する正則化を追加した手法

従来の AT では，クロスエントロピー誤差を最小化することで $\boldsymbol{x}_i$ と $\tilde{\boldsymbol{x}}_i$ が同じクラスを出力できるように学習するため，$\boldsymbol{x}_i$ と $\tilde{\boldsymbol{x}}_i$ の出力値の関係性は考慮されていない．しかしながら，$\boldsymbol{x}_i$

と $\tilde{\boldsymbol{x}}_i$ の違いは摂動の有無であり，正しく分類できるときは捉える特徴量が類似するべきである．つまり，$\tilde{\boldsymbol{x}}_i$ の出力値は同じクラス内のどのサンプルよりも，ベースにした $\boldsymbol{x}_i$ と一致することが適切だと考えられる．

Adversarial Logit Pairing (ALP)：ALP [34] は Natural と，それに対応する AEs のロジットを平均二乗誤差によって一致させる手法である．ALP の損失関数は，以下にのように定義できる．

$$L_{\mathrm{ALP}} = L(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) + \frac{\lambda}{k} \sum_{j=0}^{k-1} \left(f_j(\boldsymbol{x}_i) - f_j(\tilde{\boldsymbol{x}}_i) \right)^2 \tag{14}$$

ALP では，通常の PGD によって求めた $\tilde{\boldsymbol{x}}_i$ に対するクロスエントロピー誤差を計算する．これにより，ベースとしたサンプル $\boldsymbol{x}_i$ のロジットを AEs の上限値として，AEs を正しく分類できるようにモデルを学習することが可能となる．

TRADES：TRADES [35] は，ALP と異なり，モデルが出力するロジットの代わりに，ロジットに Softmax 関数を適用した事後確率が一致するように KL ダイバージェンスによる分布間類似度計算を追加した手法である．また，TRADES では，AEs ではなく Natural に対してクロスエントロピー誤差を計算することで，Natural に対する性能劣化の緩和を実現した．TRADES の損失関数は，Natural に対する事後確率を $p(\boldsymbol{x}_i)$，AEs に対する事後確率を $p(\tilde{\boldsymbol{x}}_i)$ として，以下の式で定義できる．

$$\begin{aligned} L_{\mathrm{TRADES}} &= L(\boldsymbol{x}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) + \lambda \cdot D_{\mathrm{KL}} \left[p(\boldsymbol{x}_i) \,||\, p(\tilde{\boldsymbol{x}}_i) \right] \\ &\text{where } D_{\mathrm{KL}} \left[p(\boldsymbol{x}_i) \,||\, p(\tilde{\boldsymbol{x}}_i) \right] = \sum p(\boldsymbol{x}_i) \log \frac{p(\boldsymbol{x}_i)}{p(\tilde{\boldsymbol{x}}_i)} \end{aligned} \tag{15}$$

さらに，TRADES ではクロスエントロピー誤差を用いた Inner-maximization の代わりに，$p(\boldsymbol{x}_i)$ と $p(\tilde{\boldsymbol{x}}_i)$ が乖離するように KL ダイバージェンスを最大化することで摂動を求める．TRADES の Inner-maximization による摂動導出は，以下の式で表すことができる．

$$\boldsymbol{\delta}_i = \underset{\|\boldsymbol{\delta}_i\|_p \leq \epsilon}{\arg\max} D_{\mathrm{KL}} \left[p(\boldsymbol{x}_i) \,||\, p(\boldsymbol{x}_i + 0.0001 \cdot \boldsymbol{r}) \right] \quad \text{s.t. } \boldsymbol{r} \sim U[-1, 1] \tag{16}$$

3.3.2 誤分類したサンプルの効率的な利用

従来の AT では，Natural な状態の分類結果に関係なく，全サンプルに対して AEs を定義して誤分類が生じないようにクロスエントロピー誤差の最小化をする．しかしながら，Natural な状態で誤分類するサンプルは，摂動がない状態でも誤分類するため，AT に使用しても頑健なモデル獲得に寄与しないと考えられる．従って，誤分類が生じたサンプルをどのように扱うかが重要となる．

Max-Margin Adversarial Training (MMA)：MMA [36] は，Natural な状態で正しく分類できたサンプルと誤分類したサンプルで異なる損失計算をする手法である．正しく分類できたサンプルは，求めた AEs を用いてクロスエントロピー誤差の最小化をする．一方，誤分類したサンプルは，Natural な状態での正しい分類を促進するために，Natural を用いたクロスエントロピー誤差の最小化をする．MMA の損失関数は以下の式で定義できる．

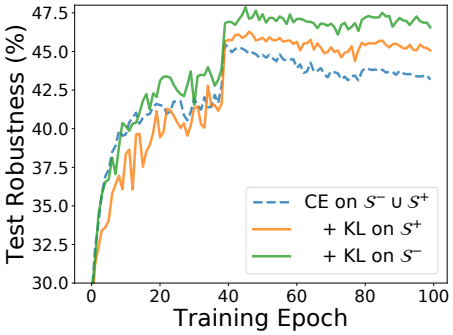


図 4　KL 正則化を適用するサンプルによる精度変化の例．KL は TRADES の KL 正則化，S^+ は正しく分類できたサンプル集合，S^- は誤分類したサンプル集合である．[37] から引用

$$L_{\mathrm{MMA}} = L(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) \cdot \mathbf{1}[y' = y] + L(\boldsymbol{x}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) \cdot \mathbf{1}[y' \neq y] \tag{17}$$

ここで，$y = \arg\max_i \boldsymbol{y}_i$，$y' = \arg\max_i p_i(\boldsymbol{x})$，$\mathbf{1}[\cdot]$ はカッコ内の事象が起こったときに 1 となるインジケータ関数である．

Misclassification Aware adveRsarial Training (MART)：Wang ら [37] は図 4 に示すように，TRADES で提案された KL 正則化を Natural な状態で誤分類しているサンプルに限定して適用することで著しく性能が向上することを実験によって発見し，この知見を参考に MART を提案した．MART の損失関数は，以下の式 (18) に示すように，$\boldsymbol{x}_i$ の正解クラスの確率が低いほど KL 正則化の影響が強くなる設計である．

$$L_{\mathrm{MART}} = \mathrm{BCE}(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i) + \lambda \cdot D_{\mathrm{KL}} \left[p(\boldsymbol{x}_i) \,||\, p(\tilde{\boldsymbol{x}}_i) \right] \cdot (1 - p_y(\boldsymbol{x}_i)) \tag{18}$$

また，MART ではクロスエントロピー誤差の代わりに，正解クラスと正解クラスを除いた中で確率が最大のクラスとのマージンが最大となるような損失計算をする Boosted Cross Entropy loss (BCE loss) を提案している．BCE loss は以下の式で表すことができる．

$$\mathrm{BCE}(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i) = -\log p_y(\tilde{\boldsymbol{x}}_i) - \log \left(1 - \max_{j \neq y} p_j(\tilde{\boldsymbol{x}}_i) \right) \tag{19}$$

3.3.3 損失関数やネットワークを改善した手法

通常の CNN が Adversarial attack に脆弱な原因の 1 つとして，特徴空間において識別境界と近いサンプルが数多く存在することが挙げられる．識別境界に近いサンプルは，微小な変動でも分類結果を容易に変化させることができるため，意図的に識別境界と引き離すような処理の適用やネットワーク設計をすることが重要となる．

カーネルトリックを用いた CNN：ニューラルネットワークでは Radial Basis Function (RBF) network を用いることで，高次元空間の非線形性が向上するため，Adversarial attack に頑健となる [16]．Taghanaki ら [38] はマハラノビス距離を用いた RBF カーネルを CNN に適用することで，特徴空間をコンパクトに表現して頑健性能向上を試みた．しかしながら，RBF network やマハラノビス距離は計算的に求めるパラメータが多く，隠れ層の状態がダイナミックに変化する CNN でそのまま使用するのは困難である．そこで，Taghanaki らは計算的に求めるパラ

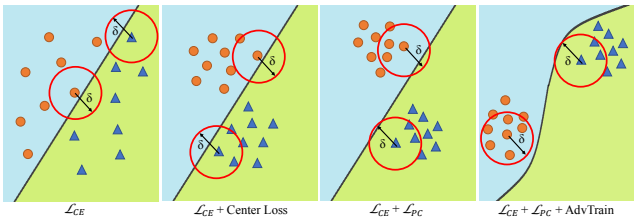


図 5　各損失関数を用いたときの特徴空間の概念図. [39] から引用.

メータを全て学習可能なパラメータとして定義し，CNN の重みパラメータ更新と同時に最適化をすることで問題点に対処した．Taghanaki らの提案した非線形カーネルは，以下の式で示すことができる．

$$g(f_l(\boldsymbol{x}_i)) = \exp\{-\beta_l \cdot D(f_l(\boldsymbol{x}_i), c_l)\} \tag{20}$$
$$\text{where}\quad D(f_l(\boldsymbol{x}_i), c_l) = (f_l(\boldsymbol{x}_i) - c_l)^\top (\Psi_l)^{-1} (f_l(\boldsymbol{x}_i) - c_l)$$

ここで，$f_l(\boldsymbol{x})$，c_l，β_l は，それぞれ，モデル f の l 層目の出力値，学習可能な l 層目の中心ベクトル，学習可能な l 層目の Gaussian の幅を表している．また，変換行列 Ψ を $\Psi = A^\top A$ に分解して，A を学習可能なパラメータとして定義することで，常に半正定値の制約を満たすことができる．

Prototype Conformity loss (PC loss)：PC loss [39] は Center loss [40] を拡張することで Adversarial attack に頑健なモデル獲得を実現した手法である．図 5 に示すように，Center loss は，各クラスの特徴量をクラス中心に集める特性があるが，識別境界から離すことは考えられていない．PC loss では，Center loss と同様で，学習可能な Prototype ベクトル $\boldsymbol{w}^c$ を定義して，クラス中心に集めつつ，各特徴量が識別境界から離れるように損失関数を設計する．PC loss で用いる損失関数は式 (21)~(23) で表すことができる．

$$L_{\text{prototype}} = L(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) + \lambda_1 \cdot L_{\text{Center}} - \lambda_2 \cdot L_{\text{PC}} \tag{21}$$
$$L_{\text{Center}} = \|f_i(\tilde{\boldsymbol{x}}_i) - \boldsymbol{w}^c_y\|_2 \tag{22}$$
$$L_{\text{PC}} = \frac{1}{k-1} \sum_{j \neq y_i} \left(\|f_i(\tilde{\boldsymbol{x}}_i) - \boldsymbol{w}^c_y\|_2 + \|\boldsymbol{w}^c_y - \boldsymbol{w}^c_j\|_2 \right) \tag{23}$$

ここで，λ_1，λ_2 は各損失の影響度を操作するハイパーパラメータである．式 (23) は特徴量を引き離す操作をするため，クラス間がワイドな特徴表現ができるモデルでは損失値が大きくなる．従って，λ_2 を小さい値で定義することで，損失が ∞ へ発散することを予防する．Prototype ベクトルは式 (21) を $\boldsymbol{w}^c$ に関して微分することで更新するが，λ_2 によって PC loss の影響が少ないため，学習率を η として，以下の式を用いた更新する．

$$\boldsymbol{w}^c_{t+1} = \boldsymbol{w}^c_t - \frac{\eta}{\lambda_2} \frac{\partial L_{\text{prototype}}}{\partial \boldsymbol{w}^c_t} \tag{24}$$

これにより，PC loss の損失値を直接用いたパラメータ更新が可能となる．

Probabilistically Compact with Logit Constraint (PC-LC)：Mustafa らが提案した PC loss は，特徴量を引き離したい層に Prototype ベクトルを設定し，学習する必要があるため，汎用性が乏しく計算コストが高い．PC-LC [41] では，損失関数の追加やネットワーク設計の変更をすることなく，損失関数の設計を変更するだけで [39] と同様の目的を達成した手法である．PC-LC の損失関数は以下に示す式で定義できる．

$$L_{\text{PC-LC}} = \frac{1}{m} \sum_{y' \neq y, i \in D} \max\left(0, p_{y'}(\tilde{\boldsymbol{x}}_i) + \xi - p_y(\tilde{\boldsymbol{x}}_i)\right) + \frac{\lambda}{m} \sum_{\boldsymbol{x}_i \in D} (d_{y,j} - C') \tag{25}$$

ここで，$d_{y,j} = \max(0, z_y(\tilde{\boldsymbol{x}}) - z_j(\tilde{\boldsymbol{x}}))$，$z(\tilde{\boldsymbol{x}})$ は $\tilde{\boldsymbol{x}}$ を入力した時のロジット，y_j は正解クラス以外で最もロジットが高いクラスを表している．PC-LC は正解クラス側に識別境界を ξ だけ移動させ，$[\xi, 1]$ の範囲外のサンプルは誤分類とみなすことでクラス内をコンパクトに表現する．また，$z_y(\tilde{\boldsymbol{x}}) - z_j(\tilde{\boldsymbol{x}}) > 0$ を要請することで，隣り合うクラスと交わることを防止している．これらの損失設計によって，クロスエントロピー誤差なしで正しく分類しつつ，頑健なモデルを獲得できる．

3.3.4　サンプルごとの Weighting を用いた手法

従来の AT では，全てのサンプルに対して等しく損失計算をして Outer-minimization をするため，AEs に対する過適合が生じやすいとされている．過適合が生じやすい要因として，攻撃リスクが低いサンプルと攻撃リスクが高いサンプルを等しく学習することが挙げられる．従って，識別境界付近のサンプルとクラス中心に近いサンプルで重要度が異なると捉えて学習する必要がある．

本項で述べる全ての手法は，各サンプルに対する重み $\omega(\boldsymbol{x}_i, \boldsymbol{y}_i)$ を用いて，以下の式を最適化することによって，あらゆる攻撃に頑健なモデルを獲得する．

$$\min_{\boldsymbol{\theta}} \frac{1}{|\mathcal{D}|} \sum_{(\boldsymbol{x}_i, \boldsymbol{y}_i) \in \mathcal{D}} \omega(\boldsymbol{x}_i, \boldsymbol{y}_i) L(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) \tag{26}$$

Geometry-Aware Instance-Reweighted Adversarial Training (GAIRAT)：GAIRAT [42] は入力画像空間において各サンプルが識別境界からどの程度離れているかを求めて，境界に近いほど大きな重み付けをして分類誤差を最小化する．著者らは PGD の更新回数を利用して識別境界と各サンプルのマージン (geometry value) の計算をした．少ない回数で誤分類が生じたサンプルは攻撃のリスクが高く，逆に，指定した PGD の更新で誤分類しないサンプルは攻撃のリスクが低いと捉えることができる．各サンプルに対する重み $\omega(\boldsymbol{x}_i, \boldsymbol{y}_i)$ の決定は，以下の式で定義できる．

$$\omega(\boldsymbol{x}_i, \boldsymbol{y}_i) = \frac{(1 - \tanh(\lambda + 5 \times (1 - 2 \times \kappa(\boldsymbol{x}_i, \boldsymbol{y}_i)/K)))}{2} \tag{27}$$

ここで，K，$\kappa(\boldsymbol{x}_i, \boldsymbol{y}_i)$，$\lambda$ はそれぞれ，PGD の最大更新回数，$\boldsymbol{x}_i$ に対する geometry value，任意の係数である．各サンプルに対する重みは $(0, 1)$ の範囲で決定される．GAIRAT では，各サンプルに対して geometry value を求め，式 (27) で重みの決定をするだけでなく，以下の式でバッチ内で重みの合計が 1 となるように正規化をする．

$$\omega(\boldsymbol{x}, \boldsymbol{y}) = \frac{\omega(\boldsymbol{x}_i, \boldsymbol{y}_i)}{\sum_{m=1}^{M} \omega(\boldsymbol{x}_m, \boldsymbol{y}_m)} \tag{28}$$

ここで，M はサンプル数を表している．従って，GAIRAT では以下の最適化式を解くことで，頑健なモデルを獲得する．GAIRAT を改良した手法として，局所的な重み付けをする Local Reweighting Adversarial Training (LRAT) [43] も提案されている．

Weighted MiniMax Risk (WMMR)：WMMR [44] は，GAIRAT と異なり，事後確率を参考に境界とサンプルのマージンを求めて，重みの決定をする．GAIRAT では，Natural な状態で誤分類しているサンプルを考慮した重み付けがされてないが，事後確率を用いることで誤分類したサンプルか否かも考慮した重みの決定を実現した．WMMR では，隣り合うクラス，つまり正解クラスを除いた中で最も高いクラスとのマージン利用した重み決定をする．WMMR は以下の式によって各サンプルに対する重みの決定をする．

$$\omega(\boldsymbol{x}_i, \boldsymbol{y}_i) = \exp\left(-\alpha \cdot \mathrm{margin}(\boldsymbol{x}_i, \boldsymbol{y}_i)\right) \tag{29}$$
$$\text{where } \mathrm{margin}(\boldsymbol{x}_i, \boldsymbol{y}_i) = p_y(\boldsymbol{x}) - \max_{j \neq y} p_j(\boldsymbol{x})$$

ここで，α は任意の係数である．境界とサンプルのマージンは，$\mathrm{margin}(\boldsymbol{x}_i, \boldsymbol{y}_i) > 0$ の時，正しく分類できるサンプルを表しており，$\mathrm{margin}(\boldsymbol{x}_i, \boldsymbol{y}_i) < 0$ の時，誤分類するサンプルを表している．また，$\mathrm{margin}(\boldsymbol{x}_i, \boldsymbol{y}_i) = 0$ の時は識別境界上のサンプルを表している．式 (29) は各サンプルに対して $(0, \infty)$ の範囲で重みを決定するため，負のマージンほど大きな重み決定がされるようになっている．WMMR では，求めた重みをそのまま分類誤差に乗算して最小化問題を解く．

Margin-Aware Instance reweighting Learning (MAIL)：MAIL [45] は WMMR と同じで隣り合うクラスとのマージン (PM: Probabilistic Margin) を考慮するために事後確率を用いて重みを決定する．WMMR では，AEs のみからマージンを計算する一方，MAIL では以下に示す 3 種類の計算方法が提案されている．

$$\mathrm{PM}_{\mathrm{nat}}(\boldsymbol{x}_i, \boldsymbol{y}_i) = p_y(\boldsymbol{x}_i) - \max_{j \neq y} p_j(\boldsymbol{x}_i) \tag{30}$$
$$\mathrm{PM}_{\mathrm{adv}}(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i) = p_y(\tilde{\boldsymbol{x}}) - \max_{j \neq y} p_j(\tilde{\boldsymbol{x}}) \tag{31}$$
$$\mathrm{PM}_{\mathrm{dif}}(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i) = p_y(\boldsymbol{x}_i) - p_y(\tilde{\boldsymbol{x}}_i) \tag{32}$$

$\mathrm{PM}_{\mathrm{nat}}$ は Natural に対するマージン，$\mathrm{PM}_{\mathrm{adv}}$ は AEs を用いたマージンの決定，$\mathrm{PM}_{\mathrm{dif}}$ は Natural と AEs を用いたマージンの決定を表している．MAIL において重みの決定式は，sigmoid 関数を用いて以下の式で定義できる．

$$\omega(\boldsymbol{x}_i, \boldsymbol{y}_i) = \mathrm{sigmoid}\left(-\gamma\left(\mathrm{PM}(\boldsymbol{x}_i, \boldsymbol{y}_i) - \beta\right)\right) \tag{33}$$

ここで，β と $\gamma \geq 0$ はそれぞれ，どれだけのデータが相対的に大きな重みを持つべきがを表すパラメータ，β 周辺の滑らかさを操作するパラメータである．MAIL も GAIRAT と同様で，式 (28) を用いてバッチ内で重みを正規化して分類誤差に乗算する．WMMR とは異なり，MAIL は sigmoid 空間で重み定義するため，極端に大きな重みが算出されることがない．

Bilevel Learnable Adversarial reWeighting：これまでの手法では，隣り合うクラスとのマージンのみを考慮した重み決定がされていた．しかしながら，Holtz ら [46] の実験によると，図 6 に示すように，強い摂動が加算された AEs ほど 2 番目以降のクラスと誤分類するサンプルが多いことが確認できる．この結果を基に，著者らはマルチクラスとのマージンを計算し，パラメトリックな関数を用いて重み決定をする Bilevel Learnable Adversarial reWeighting (BiLAW) を提案した．BiLAW では，マルチクラスのマージンを $\Delta: [0,1]^k \to [-1,1]^k$ を用いて以下の式で求める．

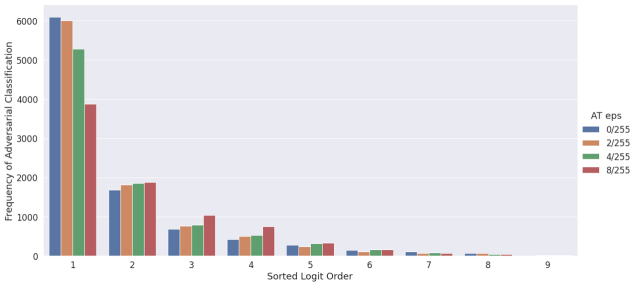


図 6　CIFAR-10 における誤分類するクラスの傾向. [46] から引用.

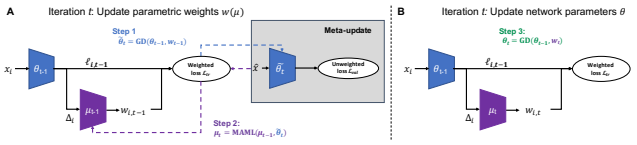


図 7　BiLAW の学習プロセス. [46] から引用.

$$\Delta^{(j)}(\boldsymbol{x}_i, \boldsymbol{y}_i) = p_y(\boldsymbol{x}_i) - p_j(\boldsymbol{x}_i) \tag{34}$$

式 (34) はクラス j とのマージンの計算を表しており，この計算を正解クラスを除いたクラス全てに対して行う．求めたマルチクラスのマージンは，パラメトリック関数 $s: \mathbb{R}^d \to \mathbb{R}^d$ に入力して重みを決定する．しかしながら，s に対する適した損失設計が困難なため，BiLAW ではメタ学習の一種である Model-Agnostic Meta-Learning (MAML) [47] を用いて学習する．BiLAW の学習は，図 7 に示すように 3 ステップから成る．まず，学習データと 1 時刻前の s が出力する重みを用いてクラス分類器 f を擬似的にアップデートする．次に，検証データセットと学習データの相関関係を用いて s の重みパラメータを更新する．最後に，学習データと現時刻の s を用いて分類器 f の重みパラメータ $\boldsymbol{\theta}$ を更新する．

3.3.5　Friendly Adversarial Training

通常の AT では，PGD を用いた Inner-maximization によって AEs を求めるため，図 8 上段に示すように，PGD の更新回数が増加するにつれて識別境界を大きく跨いでクラスが混同する．Zhang らはこの Inner-maximization が性能を劣化させる原因だと考えて，Friendly Adversarial Training (FAT) [48] を提案した．

FAT は従来の AT とは異なり，Inner-maximization を式 (35) に示すような Inner-minimization を利用することで性能劣化を予防した手法である．

$$\tilde{\boldsymbol{x}}_i = \underset{\tilde{\boldsymbol{x}}_i \in \mathcal{B}[\boldsymbol{x}_i]}{\arg\min}\ L(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i) \tag{35}$$
$$\text{s.t. } L(\tilde{\boldsymbol{x}}_i, \boldsymbol{y}_i; \boldsymbol{\theta}) - \min_{y \in \mathcal{Y}} L(\tilde{\boldsymbol{x}}_i, y; \boldsymbol{\theta}) \geq \rho$$

ここで，$\boldsymbol{y} \neq \arg\min_{y \in \mathcal{Y}} L(\tilde{\boldsymbol{x}}, y)$，$\mathcal{Y} = \{0, \cdots, k\}$, ρ は誤分類を

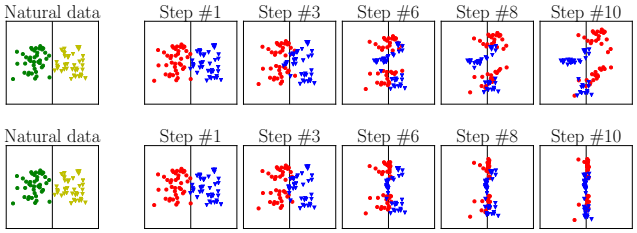


図 8　通常の Adversarial training と FAT における識別境界と AEs の関係. [48] から引用.

どの程度許容するかを表すパラメータである.

FAT を用いることで, 図 8 下段に示すように PGD の更新回数が増加しても識別境界付近の AEs を学習することができる. この Inner-minimization は, PGD を早期終了させる early-stopped PGD (PGD-K-τ) を用いることで実現ができる. PGD-K-τ は, 最大の更新回数を K として, 早期終了させるタイミングを τ で操作する. PGD-K-τ は $K = \tau$ の時に通常の PGD と捉えることができる.

PGD-K-τ では, 任意のサンプルに関して, 誤分類が生じてから何ステップ攻撃するかを τ によって操作する. 一般的な AT では, 誤差を最大化することのみに着目している一方, FAT では誤分類が生じたかどうかを考慮するため, Natural の性能劣化を防ぐことができる.

3.3.6　Learnable Boundary Guided Adversarial Training

従来の AT を含む多くの手法が AEs を直接学習することで頑健なモデルの獲得を可能とするが, 図 9(c) に示すように, 識別境界が AEs にフィットするため, Natural に対する性能が劣化する. そこで, Cui らは Natural の識別境界をガイドにしながら, 頑健なモデルを獲得する Boundary Guided Adversarial Training (BGAT) と Learnable BGAT (LBGAT) を提案した [49]. BGAT や LBGAT では, Natural のみで学習するモデル $\mathcal{M}_{nat}$ と AEs のみで学習するモデル $\mathcal{M}_{adv}$ の 2 つ使用する.

BGAT では, $\mathcal{M}_{nat}$ を事前に学習し, 識別境界を $\mathcal{M}_{adv}$ に蒸留することで Natural を意識した頑健なモデルの獲得が可能となる. BGAT は以下に示す最適化式を解くことで, 頑健なモデルを獲得する.

$$\min_{\boldsymbol{\theta}} \mathbb{E}_{(\boldsymbol{x}_i,\boldsymbol{y}_i)\in D}\left[\mathrm{MSE}(\mathcal{M}_{rob}(\tilde{\boldsymbol{x}_i}), \mathcal{M}_{nat}(\boldsymbol{x}_i))\right] \tag{36}$$

$$\text{where}\quad \mathrm{MSE}(A,B) = \frac{1}{k}\sum_{j=1}^{k}(A_j - B_j)^2$$

ここで, $\mathcal{M}_{rob}(\tilde{\boldsymbol{x}_i})$, $\mathcal{M}_{nat}(\boldsymbol{x}_i)$ はそれぞれ, 各モデルが出力するロジットを表しており, $\boldsymbol{\theta}$ は $\mathcal{M}_{rob}$ の重みパラメータを表している.

BGAT によって $\mathcal{M}_{rob}$ の頑健性能が獲得できるが, Natural のみでよく学習した時の識別境界が必ずしも頑健向上に適しているとは限らない. LBGAT は $\mathcal{M}_{rob}$ と $\mathcal{M}_{nat}$ を初期値から同時に共同学習することによって更なる頑健性能が期待できる手法として提案された. LBGAT は以下の式を最適化することにより, モデルの頑健性能を保証する.

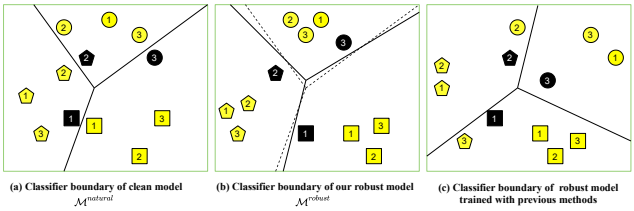


図 9　AEs と識別境界の関係性の例. 黄色が Natural, 黒色が AEs を表している. [49] から引用.

$$\min_{\boldsymbol{\theta},\boldsymbol{\theta}^*} \mathbb{E}_{(\boldsymbol{x}_i,\boldsymbol{y}_i)\in D}\left[\mathrm{MSE}(\mathcal{M}_{rob}(\tilde{\boldsymbol{x}_i}), \mathcal{M}_{nat}(\boldsymbol{x}_i)) + \beta L(\boldsymbol{x}_i,\boldsymbol{y}_i;\boldsymbol{\theta}^*)\right] \tag{37}$$

ここで, $\boldsymbol{\theta}^*$ は $\mathcal{M}_{nat}$ の重みパラメータを表している. LBGAT では, $\mathcal{M}_{nat}$ のみをクロスエントロピー誤差で分類誤差の最小化を行い, $\mathcal{M}_{rob}$ に関しては $\mathcal{M}_{nat}$ の識別境界を蒸留するのみであることに注意されたい.

従来の AT ではヒトの直感に反した誤分類が生じることが多いことが知られている. (例：犬と飛行機の誤分類) これは, AEs のみに着目して識別境界を決定することが原因と考えられる. 一方, BGAT や LBGAT では Natural の識別境界にフォーカスして AEs が分類できるように学習するため, このような誤分類を抑制することが可能となる.

4.　定量的評価指標

画像分類において, モデルの性能を証明する最も重要な指標として分類精度が使用されることが多い. 分類精度 Acc は, 推論データセットを $\hat{D}$ としてインジケータ関数 $\mathbf{1}[\cdot]$ を用いて, 以下の式によって表すことができる.

$$Acc = \frac{1}{|\hat{D}|}\sum_{(\boldsymbol{x}_i,\boldsymbol{y}_i)\in\hat{D}} \mathbf{1}\left[f(\boldsymbol{x}_i) = y\right] \tag{38}$$

しかしながら, 分類精度を計算するときはクラス確率が最大のインデックスを抜き出して, 教師信号と一致するか否かのみでしか評価していないため, 信頼度を考慮した性能評価ができていない. つまり, 特徴空間において優れた特徴表現かどうかの判断が困難である.

CNN の特徴空間を視覚的に表現する方法として t-SNE [50], UMAP [51] などが提案されているが, これらの手法は近似計算が含まれるため, 高次元な空間をそのまま 2 次元平面で表現することができない. 従って, 高次元空間を正確に評価するためには, 定量的な評価が必要となる.

特徴空間において, 優れたクラスタリングができているかを定量的に示す指標として, Calinski&Harabasz 指標 [52] や Silhouette スコア [53], Homogeneity, Completeness スコア [54] などが提案されている. 本章では, これらの指標をそれぞれ, 節に分割して詳細に述べる.

4.1　Calinski&Harabasz 指標

Calinski&Harabasz 指標 (Cal.) は, 図 10 に示すように, クラス内分散 W_k とクラス間分散 B_k から優れたクラスタリングができたかどうかを定量的に示すことが可能な指標である. Cal. では, 高次元な特徴ベクトルを次元削減することなく, そのま

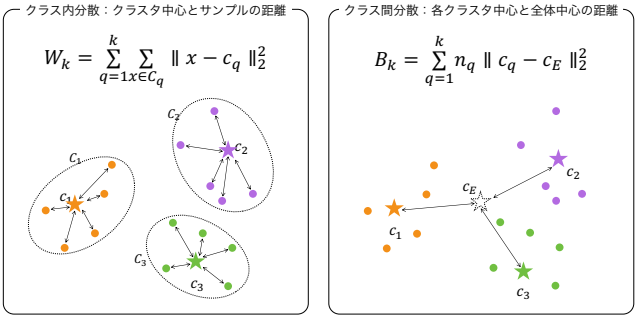


図 10　Calinski & Harabasz 指標の概念図

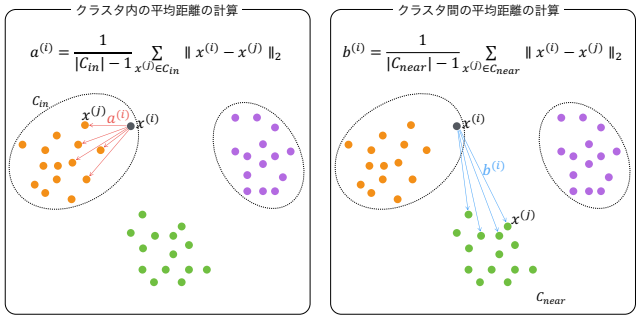


図 11　Silhouette スコアの概念図

ま評価に使用することができるため，正確な評価をすることができる．Cal. は以下に示す式を用いて求める．

$$\text{Cal.} = \frac{\text{tr}(B_k)}{\text{tr}(W_k)} \times \frac{(n_E - k)}{k - 1} \tag{39}$$

$$W_k = \sum_{q=1}^{k} \sum_{x \in C_q} \|x - c_q\|_2^2 \tag{40}$$

$$B_k = \sum_{q=1}^{k} n_q \|c_q - c_E\|_2^2 \tag{41}$$

ここで，k，C_q，c_q，c_E，n_q，n_E はそれぞれ，クラス数，クラス q のクラスタ，クラス q のクラス中心，データセット全体の中心，クラス q のサンプル数，データセット全体のサンプル数を表している．

Cal. はスコアの上限が定められていないため，高いほど優れた特徴表現ができていることを表している．逆に，Cal. のスコアが 0 に近いとき不適切なクラスタ分割，つまりクラスが混在していることを表している．

4.2　Silhouette スコア

Silhouette スコア (Sil.) は，任意のサンプルが適切なクラスタに属しているかを表す指標である．Sil. は，図 11 に示すように，任意のサンプル x_i が属するクラスタ C_{in} に含まれる全サンプルとの距離と，C_{in} から最も近いクラスタ C_{near} に含まれる全サンプルとの距離を計算することによって求める．Sil. も Cal. と同様で，特徴ベクトルのクラスタリング性能を評価するための指標であることから，次元削減をする必要がない．従って，次元削減を用いた視覚的な評価よりも正確な評価が期待できる指標の 1 つである．Sil. は以下に示す式を用いて求める．

$$\text{Sil.} = \sum_{i=1}^{n} \frac{b_i - a_i}{\max(a_i, b_i)} \tag{42}$$

$$a_i = \frac{1}{|C_{in}| - 1} \sum_{x_j \in C_{in}} \|x_i - x_j\|_2 \tag{43}$$

$$b_i = \frac{1}{|C_{near} - 1|} \sum_{x_j \in C_{in}} \|x_i - x_j\|_2 \tag{44}$$

ここで，n，x_j はそれぞれ，評価するサンプルの総数，C_{in} または C_{near} に含まれるサンプルを表している．

Sil. は $[-1, 1]$ の範囲で定義されるスコアであるため，Sil. ≈ 1 の時，x_i は正しいクラスタに属しており，コンパクトかつワイドな特徴表現ができていることを表している．逆に，Sil. ≈ -1 の時は，x_i が誤ったクラスタに属しているため，各クラスが混同していることを表している．

4.3　Homogeneity，Completeness スコア

Homogeneity スコアと Completeness スコアは，V-measure [54] に含まれる，事後確率を用いたクラスタリングに対する評価指標である．

Homogeneity スコア (Homo.)：Homo. はクラスタを基本概念として，事後確率と教師信号を用いてクラスタリング結果の定量的な評価をする．従って，図 12(a) のように，1 つのクラスタ内に 1 つのクラスのみが含まれる場合，Homo. は高いスコアとなる．一方，図 12(b) のように，1 つのクラスタ内に複数のクラスが含まれる場合，Homo. は低いスコアとなり，優れたクラスタリングができていないことを表している．Homo. は以下の示す式によって求めることができる．

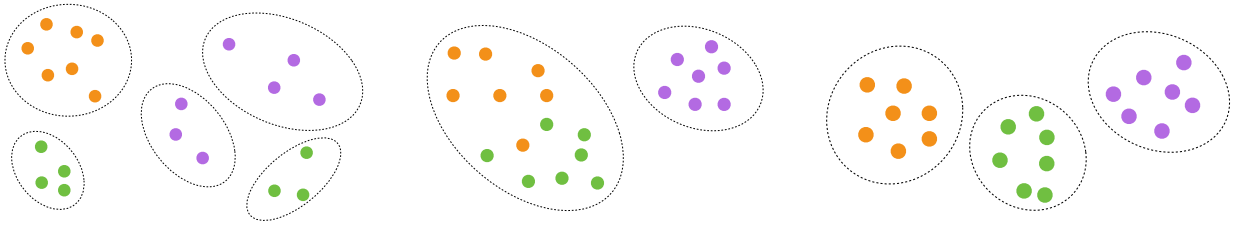
$$\text{Homo.} = \begin{cases} 1 & \text{if } H(C, K) = 0 \\ 1 - \frac{H(C|K)}{H(C)} & \text{otherwise} \end{cases} \tag{45}$$

$$H(C \mid K) = -\sum_{k=1}^{|K|} \sum_{c=1}^{|C|} \frac{a_{ck}}{N} \log \frac{a_{ck}}{\sum_{c=1}^{|C|} a_{ck}} \tag{46}$$

$$H(C) = -\sum_{c=1}^{|C|} \frac{\sum_{k=1}^{|K|} a_{ck}}{n} \log \frac{\sum_{k=1}^{|K|} a_{ck}}{n} \tag{47}$$

ここで，$C = \{c_i \mid i = 1, \ldots, n\}$，$K = \{k_i \mid i = 1, \ldots, m\}$，$A = \{a_{ij}\}$ は，それぞれ，クラス集合，クラスタ集合，クラス i でクラスタ j に含まれるデータ集合を表している．また，$H(C)$ は C に対するエントロピー，$D(C \mid K)$ は K を観測したときの C に対する条件付きエントロピーである．

Completeness スコア (Comp.)：Comp. は各サンプルに対する教師信号を基本概念として，事後確率と教師信号からクラスタリング結果の定量的結果を示すことが可能な計算方法である．Comp. は図 12(b) に示すように，任意のクラスのサンプル全てが 1 つのクラスタに含まれる時，Comp. のスコアが高くなる．この時，クラスタ内に複数のクラスが含まれていたとしても，同じクラスのサンプルが全て含まれればスコアが高くなる．一方，図 12(a) のように，同じクラスのサンプルが複数のクラスタに分かれている時スコアが低くなる．以上を踏まえると，Comp. の計算式は以下の式で定義することができる．



(a) Homo.が高く，Comp.が低い　(b) Homo.が低く，Comp.が高い　(c) Homo.とComp.が共に高い

図 12　Homogeneity スコアと Completeness スコアの関係性の例

$$\text{Comp.} = \begin{cases} 1 & \text{if } H(K,C) = 0 \\ 1 - \dfrac{H(K|C)}{H(K)} & \text{otherwise} \end{cases} \tag{48}$$

$$H(K \mid C) = -\sum_{c=1}^{|C|}\sum_{k=1}^{|K|} \frac{a_{ck}}{N} \log \frac{a_{ck}}{\sum_{k=1}^{|K|} a_{ck}} \tag{49}$$

$$H(K) = -\sum_{k=1}^{|K|} \frac{\sum_{c=1}^{|C|} a_{ck}}{m} \log \frac{\sum_{c=1}^{|C|} a_{ck}}{m} \tag{50}$$

Homo. と Comp. はベースとなる概念が入れ替わるだけなため，計算式に大幅な変更はない．

これまでに示したように，Homo. と Comp. は相反する結果を出力する．しかし，図 12(c) に示すように，同じクラスが 1 つのクラスタに含まれている場合は，Homo. と Comp. のスコアが共に高くなる傾向がある．つまり，正しくクラスタリングができた場合は 2 つのスコアが高くなる．従って，Homo. と Comp. は 2 つのスコアを併用して考察することで，クラスタリングが優れいるかどうかの判断ができるようになるといえる．また，Homo. と Comp. 共に次元削減なしで，事後確率から直接求めることができるため，正確な評価をすることができる．

5. 各手法の性能比較

本章では，4 章で述べた代表的な手法をデータセットとモデルを全て統一して分類性能を比較する[1]．また，特徴空間に対する定量的な評価も行うことで，優れた学習方法について議論する．

5.1 実験条件

本実験では，データセットとして CIFAR-10 を使用する．CIFAR-10 は，10 クラスの自然画像が含まれる，32 × 32 ピクセルの RGB 画像である．学習用と推論用データとして，それぞれ，50,000 サンプルと 10,000 サンプルずつ用意されている．学習に使用するモデルは，Adversarial attack や defense の研究において使用頻度が最も高い，34 層で Widen factor が 10 の WideResNet34-10 (WRN34-10) [55] で統一する．論文内に実験条件の詳細な記載がある場合，ハイパーパラメータや最適化関数などは論文の設定を使用する．論文に記載がない手法に関しては表 1 に示す設定で学習する．

モデルの性能は，Natural に対する分類精度と Adversarial attack による頑健性能を用いて評価する．Adversarial attack として，FGSM，反復数 10 回の PGD (PGD-10)，反復数 20 回の PGD (PGD-20)，CW の誤差を用いた PGD attack である CW_∞，AutoAttack の計 5 種類使用する．PGD attack で使用するパラメータは，それぞれ，$\alpha = 2/255$，$\epsilon = 8/255$ である．

表 1　論文に記載がない手法に対する実験設定

学習回数	100 epochs
最適化関数	momentum SGD
学習率	0.1
学習率の減衰	{75, 90} epochs で 1/10
momentum	0.9
weight decay	2×10^{-4}
データ増幅	Random Crop, Random Horizontal Flip
PGD の更新回数	10
ステップサイズ α	2/255
ϵ	8/255

特徴空間に対する定量的な評価は Calinski & Harabasz 指標 (Cal.)，Silhouette スコア (Sil.)，Homogeneity スコア (Homo.)，Completeness スコア (Comp.) を使用する．評価対象の特徴ベクトルは，WRN34-10 の全結合層に入力される特徴ベクトルとする．

5.2 各手法の性能比較

表 2 に各手法の分類精度と特徴空間の定量的評価結果を示す．表 2 より，AutoAttack を除いた AVmixup の結果が他手法と比較して最も優れていることが確認できる．AVmixup では，仮想的に定義した AEs との mixup をした画像を学習するため，データカバレッジが拡張されたことにより，Natural の精度を劣化させずに頑健なモデルを獲得できていると考える．しかしながら，AutoAttack の結果に着目すると，AVmixup の性能が著しく劣化していることが確認できる．このことから，AVmixup は学習時に使用した攻撃手法以外に対して脆弱であるといえる．

識別境界から意図的に引き離し，クラス内をコンパクトに表現する PC loss や PC-LC の結果は Standard な結果から劣化している．特徴空間の定量評価結果に着目すると，PC loss および PC-LC 共に Cal. のスコアが Standard よりも高いことが確認できる．特に，PC loss は倍近くのスコアであることから，Standard と比較して優れた特徴表現ができたといえる．一方，PC-LC は事後確率を用いて特徴空間をコンパクトに表現するため，Homo. と Comp. のスコアが PC loss よりも優れている．従って，PC-LC は出力空間において PC loss より優れた特徴表

（注1）：https://github.com/machine-perception-robotics-group/Adversarial-training

表 2　各手法の分類精度と特徴空間の定量的評価結果．表中の Standard は通常の AT [15] の結果を表している．また，太字が最も高い性能の手法，下線が次に高い性能の手法を示している．

	分類精度 [%]						特徴空間の定量評価			
	Natural	FGSM	PGD-10	PGD-20	CW_∞	AutoAttack	Cal. ↑	Sil. ↑	Homo. ↑	Comp. ↑
Standard	80.81	53.24	49.26	48.60	47.62	45.25	593.03	0.044	0.659	0.661
PC loss	76.57	49.45	45.22	44.46	43.11	40.63	1037.52	0.110	0.122	0.318
PC-LC	78.40	51.88	47.84	46.94	52.81	43.34	601.04	0.033	0.626	0.628
ALP	71.68	51.11	49.05	48.69	45.29	43.82	497.12	0.006	0.541	0.546
TRADES	84.45	60.45	55.30	54.44	43.81	51.48	876.27	0.100	0.706	0.707
MART	80.63	61.33	58.62	57.29	52.63	50.80	1329.52	0.048	0.660	0.665
GAIRAT	85.54	58.91	54.56	52.33	50.82	48.15	1020.54	0.108	0.725	0.726
WMMR	82.52	58.39	54.62	53.72	49.81	47.15	880.11	0.076	0.683	0.684
MAIL	84.60	59.54	55.96	54.12	52.85	50.21	1232.33	0.163	0.712	0.713
BGAT	88.93	61.05	52.64	50.83	52.26	48.81	841.92	0.120	0.651	0.655
LBGAT	86.73	62.22	56.09	54.50	54.06	**51.78**	**2925.62**	**0.333**	0.745	0.746
FAT	86.60	53.34	45.51	44.02	45.60	42.38	1228.33	0.204	**0.772**	0.774
AVmixup	**94.81**	**80.28**	**69.29**	**65.01**	**54.8**	16.75	1879.99	0.260	0.662	0.667
GAT	86.88	58.25	53.37	52.56	49.48	47.39	1403.48	0.134	0.739	**0.795**

現ができている．

BGAT と LBGAT を比較すると，BGAT は Natural のみの学習によって獲得した識別境界を頑健なモデルへ蒸留することから，LBGAT よりも Natural の性能が高い．しかしながら，頑健性能に着目すると LBGAT の結果が優れていることが確認できる．さらに，特徴空間においても LBGAT が全手法と比較して最も優れている．これは，通常のモデルと頑健なモデルを初期値から共同で学習することで，Natural な分類結果に影響されすぎない適切な学習ができる恩恵である．

サンプル毎ごとに重み付けをする手法は，意図的に特徴空間を操作していないにも関わらず，Standard や PC loss の結果より優れていることが確認できる．また，分類精度や頑健性能に関しても Standard な結果から大幅な性能向上が確認できる．

TRADES や MART，FAT は著しい性能向上が確認できないが，他手法と組み合わせることによって，頑健性能および分類性能の向上が多くの論文で報告されている．

5.3　特徴空間の可視化

一部の手法の特徴ベクトルを t-SNE で次元削減した特徴空間の例を図 13 に示す．定量的評価で最も特徴表現が優れていた LBGAT は，図 13(e) に示すように，クラスごとにクラスターが形成されており視覚的にも優れていることが確認できる．図 13(a)(b) に示すように，PC loss と Standard を比較すると視覚的に大きな違いが確認できない．しかしながら，定量評価結果に着目すると，PC loss は Cal. および Sil. が Standard より高スコアである．また，図 13(c)MART や (d)MAIL に関しても，視覚的に優れた特徴表現と捉えることが困難である．

このことから，次元削減して 2 次元平面にプロットした特徴空間と分類精度を用いた比較では，優れた分類精度のモデルを過大評価する 1 つの要因になる可能性が高いと言える．従って，分類精度と特徴空間の定量的評価指標を併用して評価することが非常に重要であり，次元削減した特徴空間のみで優れたモデルを判別することは困難であるため，定量的評価指標と合わせて判断することが重要である．

6.　おわりに

本稿では，代表的な Adversarial attack と Adversarial training の手法について述べた．まず，Adversarial Examples (AEs) の定義を簡潔にして，1 ステップ，マルチステップ，その他の摂動導出の 3 つに分類した．1 ステップで摂動を求める手法は FGSM を代表例として詳細な説明をした．マルチステップで摂動を求める手法は PGD や CW を代表例として詳細に述べた．

次に，Adversarial training を学習中の摂動の求め方に着目した手法と，識別境界に着目した手法の 2 つ分類した．摂動の求め方に着目した手法では，PGD より強い摂動を学習するだけでなく，適切な摂動の初期方向の定義や仮想的に定義した AEs との mixup を行う手法などについて述べた．

識別境界に着目した手法では，さらに詳細に分類してそれぞれの手法について述べた．1 つ目のモデルの出力に対する正則化を追加する手法では，AEs と AEs が基にした画像と出力が一致するような損失を定義することで，Natural に対する性能の劣化を予防している．2 つ目の誤分類したサンプルを活用する手法では，Natural な状態で誤分類するサンプルは頑健性能の向上に寄与しないため，Natural に対する分類結果に着目して損失計算することで頑健性能を向上を実現した．3 つ目の損失関数やネットワークを改善した手法では，意図的に特徴空間を非線形するようなネットワーク設計や，特徴量を意図的に識別境界から引き離すような損失によって優れた特徴表現の獲得を実現している．4 つ目のサンプルごとの重み付けを用いた手法では，識別境界とサンプルの近さを PGD の最小攻撃回数や，隣接クラスとの確率の差などを用いて特定し，境界に近いほど大きな重み付けがされるように関数設計をしている．その他，分類できなかったものとして，Natural の境界を蒸留する手法や，境界

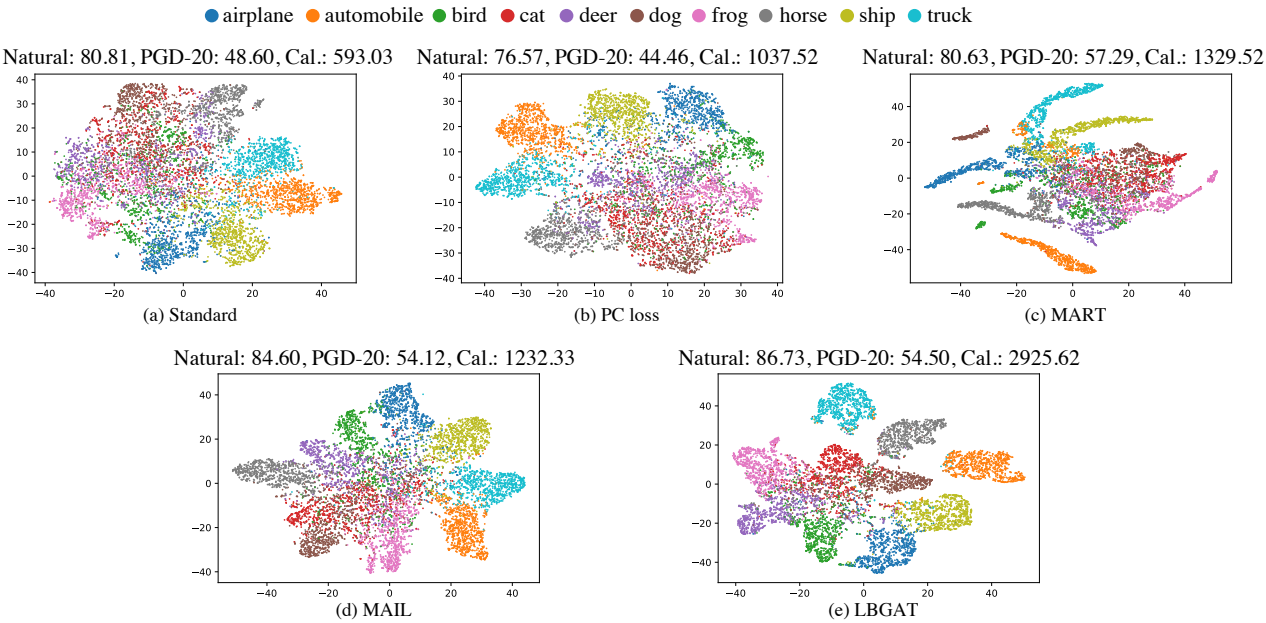


図 13　t-SNE を用いて次元削減した特徴空間の可視化例

付近の AEs を学習する手法などについて述べた．

4 章では，分類精度の計算方法を述べた後に，特徴空間に対する定量的評価指標について述べた．特徴空間の定量的評価指標は，評価対象の高次元空間の次元削減なしで，直接評価できるため視覚的な評価よりも正確な評価が可能である．

最後に，5 章で代表的な Adversarial training の性能を比較した．性能比較では，優れた分類性能を発揮しているモデルでも，特徴表現という観点では劣っている手法が散見された．逆に，分類性能や頑健性能の著しい向上がなくても，優れた特徴表現を可能とした手法も確認できた．従って，分類性能や頑健性能のみで手法の優劣を判断すると，精度の高いモデルの過大評価につながるため，特徴空間に対する定量的な評価を合わせて判断することが重要である．

文　　献

[1] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” IEEE Conference on Computer Vision and Pattern Recognition, pp.770–778, 2016.

[2] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, “You only look once: Unified, real-time object detection,” IEEE Conference on Computer Vision and Pattern Recognition, pp.779–788, 2016.

[3] J. Long, E. Shelhamer, and T. Darrell, “Fully convolutional networks for semantic segmentation,” IEEE Conference on Computer Vision and Pattern Recognition, pp.3431–3440, 2015.

[4] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, “Analyzing and improving the image quality of stylegan,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.8110–8119, 2020.

[5] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus, “Intriguing properties of neural networks,” arXiv preprint, 2013.

[6] C. Xie, J. Wang, Z. Zhang, Y. Zhou, L. Xie, and A. Yuille, “Adversarial examples for semantic segmentation and object detection,” IEEE International Conference on Computer Vision, pp.1369–1378, 2017.

[7] L. Huang, C. Gao, Y. Zhou, C. Xie, A. Yuille, C. Zou, and N. Liu, “Unversal physical camouflage attack on object detection,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.717–726, 2020.

[8] S. Huang, N. Papernot, I. Goodfellow, Y. Duan, and P. Abbeel, “Adversarial attack on neural network policies,” Workshop on International Conference on Learning Representations, pp.1–7, 2017.

[9] N. Akhtar and A. Mian, “Threat of adversarial attacks on deep learning in computer vision: A survey,” IEEE Access, vol.6, pp.14410–14430, 2018.

[10] N. Akhtar, A. Mian, N. Kardan, and M. Shah, “Advances in adversarial attacks and defenses in computer vision: A survey,” IEEE Access, vol.9, pp.155161–155196, 2021.

[11] P. Samangouei, M. Kabkab, and R. Chellappa, “Defense-gan: Protecting classifiers against adversarial attacks using generative models,” International Conference on Learning Representations, pp.1–17, 2018.

[12] Y. Bakhti, S.A. Fezza, W. Hamidouche, and O. Déforges, “Ddsa: A defense against adversarial attacks using deep denoising sparse autoencoder,” IEEE Access, vol.7, pp.160397–160407, 2019.

[13] D.P. Kingma and M. Welling, “Auto-encoding variational bayes,” International Conference on Learning Representations, pp.1–14, 2014.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, “Generative adversarial nets,” Advances in Neural Information Processing Systems, pp.••–••, 2014.

[15] A. Madry, A. Makelov, L. Schmidt, D. Tsipras, and A. Vladu, “Towards deep learning models resistant to adversarial attack,” International Conference on Learning Representations, pp.••–••, 2018.

[16] I. Goodfellow, J. Shlens, and C. Szegedy, “Explaining and harnessing adversarial examples,” International Conference on Learning Representations, pp.••–••, 2015.

[17] A. Rozsa, E.M. Rudd, and T.E. Boult, “Adversarial diversity and hard positive generation,” IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp.25–32, 2016.

[18] A. Kurakin, I. Goodfellow, and S. Bengio, “Adversarial examples in the physical world,” Workshop on International Conference on Learning Representations, pp.••–••, 2017.

[19] Y. Dong, F. Liao, T. Pang, H. Su, J.Z.X. Hu, and J. Li, “Boosting adversarial attacks with momentum,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.9185–9193, 2018.

[20] C. Xie, Z. Zhang, Y. Zhou, S. Bai, J. Wang, Z. Ren, and A. Yuille, “Improving transferability of adversarial examples with input diversity,” IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.2730–2739, 2019.

[21] N. Carlini and D. Wagner, "Towards evaluating the robustness of neural networks," IEEE Symposium on Security and Privacy, pp.39–57, 2017.

[22] G. Sriramanan, S. Addepalli, A. Baburaj, and V.B. R, "Guided adversarial attack for evaluating and enhancing adversarial defenses," Advances in Neural Information Processing Systems, vol.33, pp.••–••, 2020.

[23] S. Gowal, J. Uesato, C. Qin, P.-S. Huang, T. Mann, and P. Kohli, "An alternative surrogate loss for pgd-based adversarial testing," arXiv preprint, 2019.

[24] F. Croce and M. Hein, "Minimally distorted adversarial examples with a fast adaptive boundary attack," International Conference on Machine Learning, vol.119, pp.2196–2205, 2020.

[25] M. Andriushchenko, F. Croce, N. Flammarion, and M. Hein, "Square attack: a query-efficient black-box adversarial attack via random search," European Conference on Computer Vision, pp.••–••, 2020.

[26] F. Croce and M. Hein, "Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks," International Conference on Machine Learning, vol.119, pp.2206–2216, 2020.

[27] X. Dong, J. Han, D. Chen, J. Liu, H. Bian, Z. Ma, H. Li, X. Wang, W. Zhang, and N. Yu, "Robust superpixel-guided attentional adversarial attack," IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.12895–12904, 2020.

[28] Y. Qian, J. Wang, B. Wang, S. Zeng, Z. Gu, S. Ji, and W. Swaileh, "Visually imperceptible adversarial patch attacks on digital images," arXiv preprint, 2020.

[29] J. Su, D.V. Vargas, and S. Kouichi, "One pixel attack for fooling deep neural networks," IEEE Transactions on Evolutionary Computation, vol.23, no.5, pp.828–841, 2019.

[30] S.-M. Moosavi-Dezfooli, A. Fawzi, O. Fawzi, and P. Frossard, "Universal adversarial perturbations," IEEE Conference on Computer Vision and Pattern Recognition, pp.1765–1773, 2017.

[31] Q.-Z. Cai, M. Du, C. Liu, and D. Song, "Curriculum adversarial training," International Joint Conference on Artificial Intelligence, pp.3740–3747, 2018.

[32] S. Lee, H. Lee, and S. Yoon, "Adversarial vertex mixup: Toward better adversarially robust generalization," IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.269–278, 2020.

[33] H. Zhang, M. Cisse, Y.N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," International Conference on Learning Representations, pp.1–13, 2018.

[34] H. Kannan, A. Kurakin, and I. Goodfellow, "Adversarial logit pairing," arXiv preprint, 2018.

[35] H. Zhang, Y. Yu, J. Jiao, E. Xing, L.E. Ghaoui, and M. Jordan, "Theoretically principled trade-off between robustness and accuracy," International Conference on Machine Learning, vol.97, pp.7472–7482, 2019.

[36] G.W. Ding, Y. Sharma, K.Y.C. Lui, and R. Huang, "Mma training: Direct input space margin maximization through adversarial training," International Conference on Learning Representations, pp.1–28, 2020.

[37] Y. Wang, D. Zou, J. Yi, J. Bailey, X. Ma, and Q. Gu, "Improving adversarial robustness requires revisiting misclassified examples," International Conference on Learning Representations, pp.1–14, 2020.

[38] S.A. Taghanaki, K. Abhishek, S. Azizi, and G. Hamarneh, "A kernelized manifold mapping to diminish the effect of adversarial perturbations," IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp.11340–11349, 2019.

[39] A. defense by restricting the hidden space of deep neuralnetworks, "Aamir mustafa and salman khan and munawar hayat and roland goecke and jianbing shen and ling shao," IEEE International Conference on Computer Vision, pp.3384–3393, 2019.

[40] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," European Conference on Computer Vision, pp.499–515, 2016.

[41] X. Li, X. Li, D. Pan, and D. Zhu, "Improving adversarial robustness via probabilistically compact loss with logit constraints," AAAI Conference on Artificial Intelligence, vol.35, pp.8482–8490, 2021.

[42] J. Zhang, J. Zhu, G. Niu, B. Han, M. Sugiyama, and M. Kankanhalli, "Geometry-aware instance-reweighted adversarial training," International Conference on Learning Representations, pp.1–29, 2021.

[43] R. Gao, F. Liu, K. Zhou, G. Niu, B. Han, and J. Cheng, "Local reweighting for adversarial training," arXiv preprint, 2021.

[44] H. Zeng, C. Zhu, T. Goldstein, and F. Huang, "Are adversarial examples created equal? a learnable weighted minimax risk for robustness under non-uniform attacks," AAAI Conference on Artificial Intelligence, vol.35, pp.10815–10823, 2021.

[45] Q. Wang, F. Liu, B. Han, T. Liu, C. Gong, G. Niu, M. Zhou, and M. Sugiyama, "Probabilistic margins for instance reweighting in adversarial training," Advances in Neural Information Processing Systems, pp.1–12, 2021.

[46] C. Holtz, T.-W. Weng, and G. Mishne, "Learning sample reweighting for adversarial robustness," OpenReview, 2021.

[47] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," International Conference on Machine Learning, vol.70, pp.1126–1135, 2017.

[48] J. Zhang, X. Xu, B. Han, G. Niu, L. Cui, M. Sugiyama, and M. Kankanhalli, "Attacks which do not kill training make adversarial learning stronger," International Conference on Machine Learning, vol.119, pp.11278–11287, 2020.

[49] J. Cui, S. Liu, L. Wang, and J. Jia, "Learnable boundary guided adversarial training," IEEE/CVF International Conference on Computer Vision, pp.15721–15730, 2021.

[50] L. van derMaaten and G. Hinton, "Visualizing data using t-sne," Journal of Machine Learning Research, vol.9, no.86, pp.2579–2605, 2008.

[51] L. McInnes, J. Healy, and J. Melville, "Umap: Uniform manifold approximation and projection for dimension reduction," arXiv preprint, 2018.

[52] T. Caliński and J. Harabasz, "A dendrite method for cluster analysis," Communications in Statistics, vol.3, no.1, pp.1–27, 1974.

[53] P.J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," Journal of Computational and Applied Mathematics, vol.20, pp.53–65, 1987.

[54] A. Rosenberg and J. Hirschberg, "V-measure: A conditional entropy-based external cluster evaluation measure," Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning, pp.410–420, 2007.

[55] S. Zagoruyko and N. Komodakis, "Wide residual networks," British Machine Vision Conference, pp.87.1–87.12, 2016.