機械学習品質評価・向上技術に関する報告書

機械学習品質マネジメントガイドライン付属文書

第 1 版

2021 年 7 月 5 日

国立研究開発法人産業技術総合研究所

デジタルアーキテクチャ研究センター
テクニカルレポート DigiARC-TR-2021-02

サイバーフィジカルセキュリティ研究センター
テクニカルレポート CPSEC-TR-2021002

人工知能研究センター
テクニカルレポート

まえがき

国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）受託業務（JPNP 20006）「機械学習システムの品質評価指標・測定テストベッドの研究開発」では、機械学習品質を説明可能にするために機械学習品質マネジメントガイドライン[1]を開発しており、このガイドライン開発と並行して、機械学習品質の評価・向上技術の調査・研究開発も行っている。本調査・研究開発は現在も継続中であるが、機械学習品質マネジメントガイドラインに記載されている品質評価に関する技術的な知見も得られているため、本稿では、これまでの（2019～2020 年度の）本調査・研究開発の内容と結果について報告する。

目次

1　はじめに

統計的機械学習を利用した各種産業製品の品質を明確に説明可能にするために、機械学習品質マネジメントガイドラインが開発されている（第 1 版[1]）。このガイドライン第 2 版では、機械学習システムに対する内部品質の 9 つの特性（学習モデルの安定性やプログラムの健全性等）に着目しているが、これら内部品質特性の評価や向上の技術は必ずしもまだ確立していない。本稿は、ガイドラインの開発と並行して行われている内部品質特性の評価・向上技術の調査・研究開発に関する報告書である。

1.1　本調査・研究開発の概要

図 1.1 に、2019~2020 年度に調査・研究開発した機械学習品質評価・向上技術（図 1.1 中の中央黄色四角、左端数字は本稿で説明する章番号）と機械学習モデルのライフサイクルの各フェーズ、9 つの内部品質特性との関係を示す。以下、各技術については 2 章以降で詳しく説明するが、ここで簡単に説明しておく。

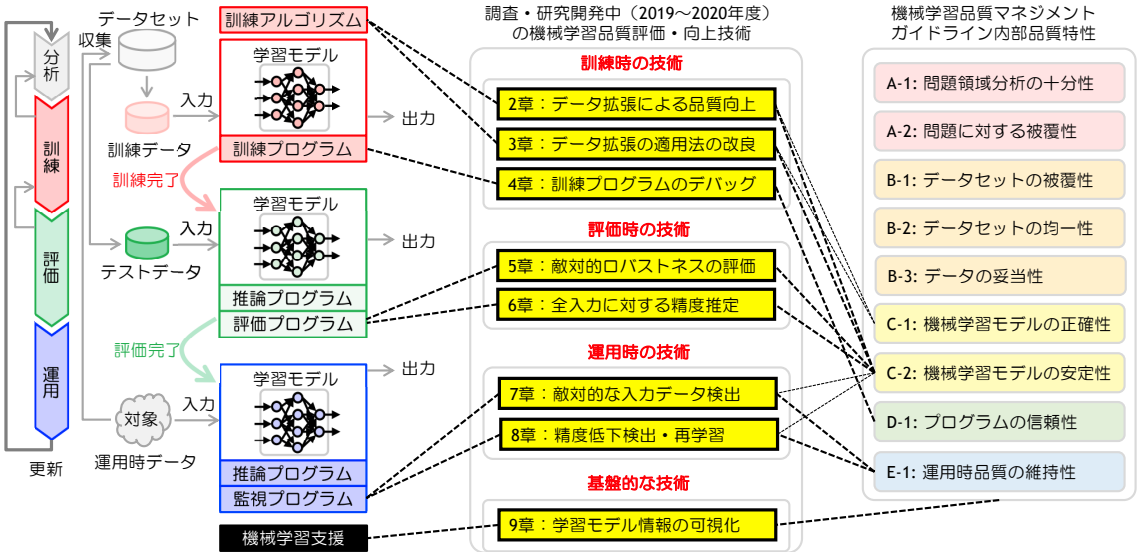


図 1.1　機械学習品質評価・向上技術（2019～2020 年度調査・研究開発）

- **2 章　データ拡張による品質向上：**
  学習データを加工してデータを増強するデータ拡張手法が画像識別の品質評価に与える影響とその品質改善方法について検討を行った。様々なデータ拡張手法（とその組み合わせ）に対して実験を行い、識別精度の平均値だけでなく分散も計測することが、品質を評価するために重要であるとの結論を得た。

- **3 章　深層学習におけるデータ拡張の適用法の改良による品質改善：**
  データ拡張によって得られる多様性をさらに向上させ、深層学習モデルの正確性や安

定性を向上させるための訓練手法として、FC-mixup 法や Latent DA 法等を提案し、画像識別の問題においてネットワークモデルの品質向上に寄与することを確認した[7][8]。

- **4 章　深層 NN ソフトウェアのデバッグ・テスティング：**
  深層学習型の機械学習モデルの不具合の原因を推論時の直接原因（予測・推論プログラムによる）と訓練時の根本原因（訓練・学習プログラム、学習モデル、学習データセットによる）に分けて整理し、訓練・学習プログラムのバグの有無を、推論時の機械学習モデルの内部情報（ニューロンカバレッジ）によって推定するための検査指標と分析手法を提案し、実験によってその検査指標の有効性を確認した[2]-[6]。

- **5 章　ロバストネスの評価・向上技術：**
  敵対的データを含む入力ノイズに対するロバストネスの指標の一つに最大安全半径（誤判断を生じないことを保証できるノイズの最大値）を計測する技術と、その半径を増加させる技術について調査した。最大安全半径の評価の計算コストは非常に高いが、近年、その近似値を効率よく見積もる手法が提案されてきている。

- **6 章　汎化誤差上界の見積り技術：**
  機械学習モデルの汎化性能を評価するため、全入力に対する不正解率（汎化誤差）の上界の見積り方法を調査した。従来の汎化誤差上界の見積り方法の精度は低く、汎化性能の評価指標として適用することは現時点（2020 年）ではまだ難しいが、最近、見積り精度は改善されてきており、将来的には汎化性能評価技術として期待できる。

- **7 章　敵対的データ検出技術：**
  入力画像が敵対的データであるかを判別する方法を実用的に確立するために、最先端の敵対的データ検出手法を調査し、4 つの主要なカテゴリに分類した。それらの敵対的データ検出手法を評価するため、実際に代表的な手法に対して追実験を行い、4 つの中で NIC 法が最も高い検出率を示すことを確認した。

- **8 章　運用時における AI 品質管理技術：**
  運用時に発生するデータの変化や未知のデータの到来に対応するために、データ分布変化に対する検知・適応技術等を幅広く調査した。これまでに開発されている手法のほとんどが適応時に運用データの正解ラベルを用いる教師あり手法であるが、適用可能性の面や運用コストの面では教師なし/半教師あり手法が有望であり、その視点から整理し検討した調査結果をサーベイとしてまとめた。

- **9 章　学習モデル情報の可視化：**
  複数の学習モデル間の差分・比較結果の可視化や各モデルに反映されている作業者（アノテータ、モデル設計者）の感性の可視化を主な目的として、機械学習支援のための可視化手法について調査し、学習モデルの差分可視化ツールを試作した。一方、作業者の感性可視化手法についても検討を進めている。

1.2 著者リスト

各章の著者は以下のとおり：

- 1 章：磯部 祥尚 （産業技術総合研究所）
- 2 章：大西 正輝 （産業技術総合研究所）
- 3 章：高瀬 朝海 （産業技術総合研究所）
- 4 章：中島 震 （国立情報学研究所）
- 5 章：磯部 祥尚 （産業技術総合研究所）
- 6 章：磯部 祥尚 （産業技術総合研究所）
- 7 章：中島 裕生 （テクマトリックス株式会社）
- 8 章：大川 佳寛、小林 健一 （富士通株式会社）
- 9 章：宮城 優里 （産業技術総合研究所）

1.3 謝辞

本調査・研究開発は、国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）受託業務（JPNP20006）「機械学習システムの品質評価指標・測定テストベッドの研究開発」として行っています。

2　データ拡張による品質向上

深層学習を用いた画像認識の中でも物体の識別の品質評価に焦点をあてて研究を進める。学習データを加工することでデータを増強する Data Augmentation（データ拡張）が画像識別の品質評価にどのように影響を与えるのか、品質を改善するためにはどうすればいいのかについての検討を行った。

特に議論の中で「品質評価に関して論文では state-of-the-art を競うことが多いが、産業応用を考えた場合には精度が 98%という場合にはデータセットを変えた場合にでも 98%の精度が出ることが望ましい。それが保証されるなら 98%ではなく 80%でも構わない」という意見があった。これまでに研究会や国際会議で発表されている手法においてもデータセットに過学習することで精度が上がっているように見えるという問題は散見している。

2.1　学習データ数と識別率の関係（予備実験）

最初に予備実験として以下の 2 点を検証した。

予備実験① 学習データを増加していくことで識別率はどのように変化していくのか
予備実験② 学習データを変えることで識別率はどのように変化するのか

両予備実験において深層学習のモデルとして WideResNet28-10 を用い、データセットは CIFAR10 を用いる。CIFAR10 は 10 クラスの画像で構成され、各クラス毎に 6000 枚のデータ（合計 6 万枚）が用意されている。これらの画像を左右反転に加え、上下左右にずらすことでデータを 10 倍に増やし 60 万枚とし、この 60 万枚の世界がデータによって構成される全ての世界であると仮定する。予備実験①では 60 万枚の全ての世界の中から N 枚が観測されたとして学習を行い、全ての世界の 60 万枚で評価を行い識別率を求める。つまり N = 600000 で学習する際には世界の全てが観測できたと仮定して識別機を作成し、識別率を求めることとなる。これは十分に識別性能が発揮されるネットワークモデルにおいては識別率が 100% になることが報告されているが、実際にそうなるかを確かめる実験でもある。識別結果を図 2.1 に示す。横軸はデータ数の対数グラフ、縦軸は識別率である。

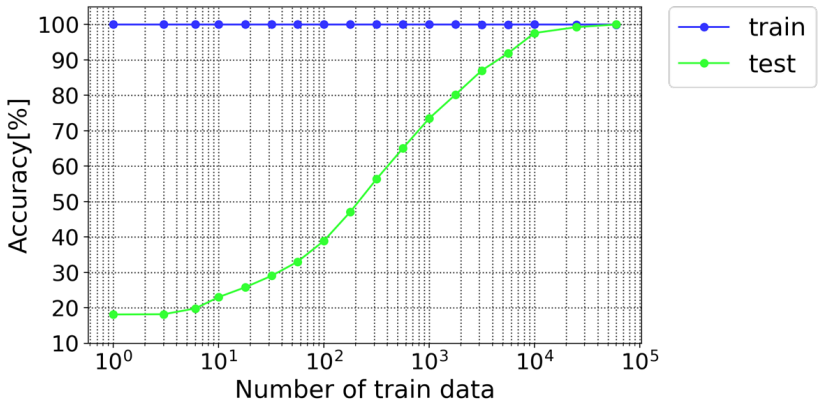


図 2.1　学習データ数と識別率の関係

これまでの研究で報告されているように 60 万画像という大量のデータであっても世界の全ての画像を学習データに利用することが可能であれば識別率は 100%を実現できることが分かる。また、各クラス 100 枚の学習データで 60 万枚の画像を識別すると 40%程度の識別精度であることが分かる。そこで予備実験②を行うための学習データ数は 10 クラス×100 枚の N = 1000 枚とした。オリジナルの各クラス 6000 枚の画像を 100 枚×60 セットに分割し、60 セットの学習データを用いてニューラルネットワークを学習し、60 万枚の画像で評価を行った。それぞれの学習データに対する識別精度のヒストグラムを図 2.2 に示す。

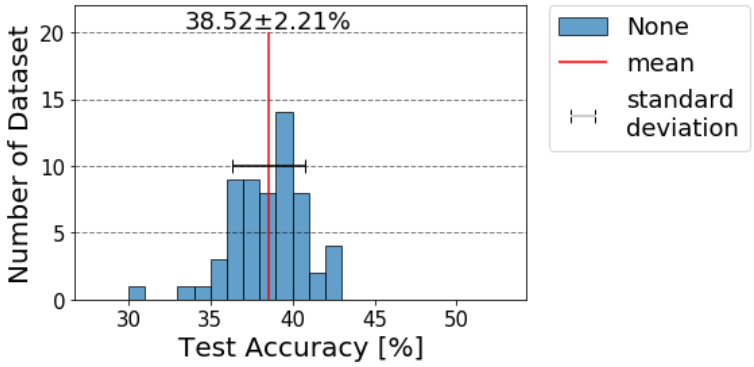


図 2.2 学習データの違いによる識別精度のヒストグラム

学習データがある組み合わせの場合には 42%の識別率が得られる一方で、学習データがある組み合わせの場合には識別率は 30%しか得られないことが分かり、これまでの識別精度だけの評価指標では品質の評価という観点では適していないことが明らかになった。なお、平均は 38.52%であり標準偏差は 2.21 である。この予備実験の結果から品質を評価する際には従来のような識別率だけで評価するのではなく、品質の安定性を評価するためには分散や標準偏差を考慮した評価が必要であることが分かる。

2.2 識別率の平均値と標準偏差の評価

これまでに提案されている様々なデータ拡張について以下の実験③を行うことで各データ拡張手法を品質の観点で評価した。

実験③ 様々なデータ拡張に対して予備実験②の方法で平均識別率と標準偏差を求める

基本的な実験設定は予備実験②と同じである。データ拡張としては変形や回転などの幾何学変換として Skew, Scale Augmentation, Shear, Rotate, Rotate Zoom の 5 種類、ノイズ付加として PCA Color Augmentation, Gaussian Noise, Patch Gaussian, Salt Pepper Noise の 4 種類、色情報の非線形変換として Gamma Transform, Contrast Transform の 2 種類、ぼかし処理として Gaussian Filter と Smoothing Filter の 2 種類、マスク処理として Cutout, Random Erasing, Cut Mix の 3 種類、データ合成として Manifold-Mixup を実装し、各評価を行った。評価の結果を図 2.3 に示す。

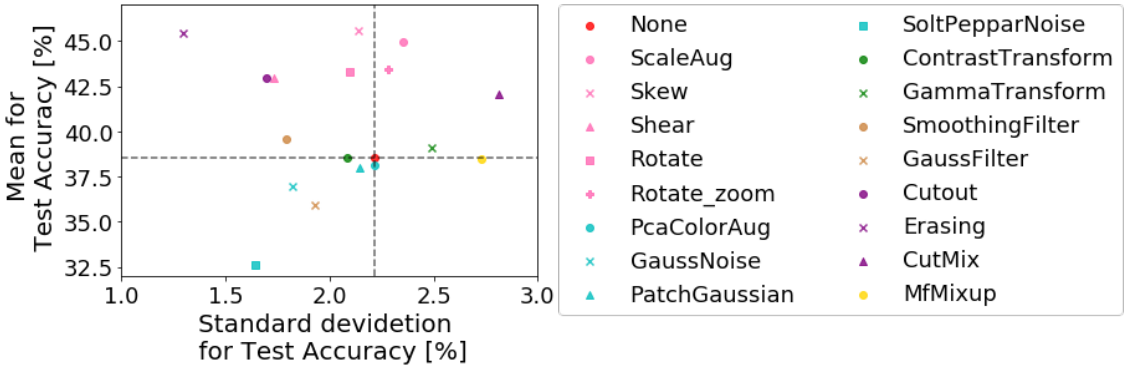


図 2.3　データ拡張による品質評価

それぞれのデータ拡張手法について縦軸に識別精度、横軸に標準偏差をプロットした散布図となっている。データ拡張を何もしていない None（赤点）を基準点として、基準点の左上の第二象限は精度を向上させながら標準偏差が小さくなっているため識別精度を上げながらもばらつきを抑えるデータ拡張であると言うことができ、品質を高める手法であると結論付けることができる。一方で基準点の右上の第一象限は識別精度の向上は認められるものの標準偏差が大きくなっているため、データによっては効果があるが、そのばらつきは大きいと考えられる。基準点の左下の第三象限は認識精度は下がるもののばらつきは小さくなっているため、識別精度は必ずしも上がらないが、品質を保証するという意味では貢献できるデータ拡張手法であるといえる。基準点の右下の第四象限は識別精度を下げつつばらつきを大きくするというデータ拡張に適していない処理がプロットされるエリアであるが、このようなデータ拡張手法は今回の適用例には見られなかった。

従来の state-of-the-art な評価という観点では上に行くほど良いデータ拡張手法であると言えるが、標準偏差を考慮した品質という点で考えると左上に行くほど良いデータ拡張手法であると言える。傾向としては幾何学変換のデータ拡張手法（図中のピンクのプロット）は上へと評価が上がる傾向にあり、ノイズ付加（水色のプロット）は左下へと下がる傾向にある。一方でマスク処理（紫のプロット）は左上へと上がる傾向にあり、データ拡張の中でも特に優れている手法であると評価することができる。

2.3　データ拡張手法の組み合わせの効果

次にデータ拡張としてばらつきを下げる効果のあった幾何学変換とノイズ付加、マスク処理の 3 つについてそれぞれのデータ拡張を全て行う場合とランダムに行う場合に対して一定の確率でデータ拡張する場合と徐々にデータ拡張をする割合を増やした場合を組み合わせて、識別率がどのように変化するかを実験④で検証した。実験④の検証項目は以下の通りである。

実験④ 品質を向上させるデータ拡張手法の組み合わせを明らかにする

幾何学変換、ノイズ付加、マスク処理の 3 つのデータ拡張について、それぞれのデータ拡

張を全て行う場合（All)、ランダムに行う場合（Random)、一定の割合で行う場合（const)、行う割合を線形で増やす場合（Linear）で実験を行い散布図を作成した。図 2.4 にその結果を示す。

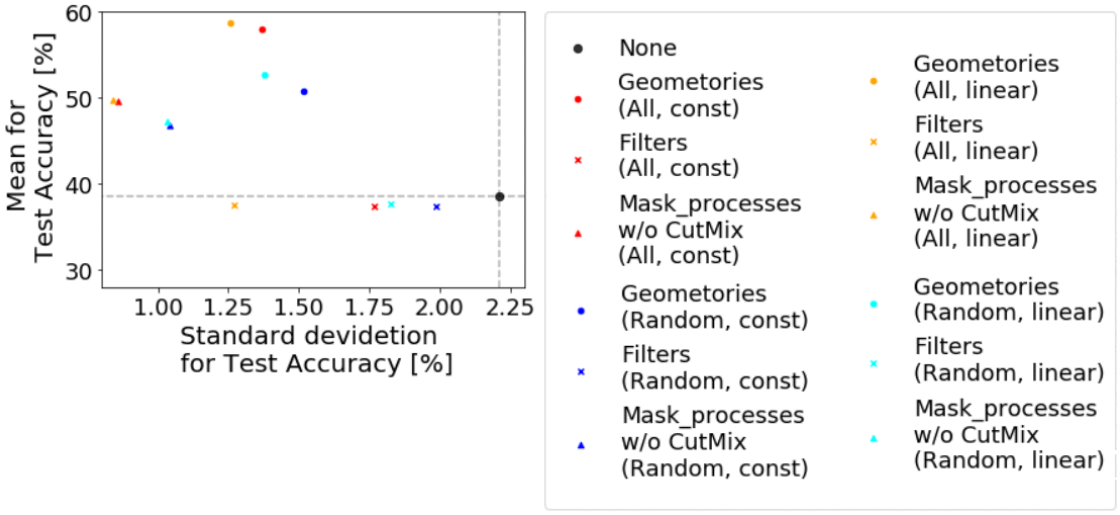


図 2.4　データ拡張手法の組み合わせ実験による品質評価

実験の結果から幾何学変換とマスク処理のデータ拡張においてはランダムに行うよりも全てを行う方が品質がよくなり、一定の確率で行うよりも徐々に行う確率を増やしていった方が識別精度を高く、かつ標準偏差を小さくできることが分かった。一方でノイズ付加に関しては識別精度を上げることはできなかったが、大きく下げることなく標準偏差を小さくする効果があることは分かった。

2.4　まとめ

これまでの多くの論文で行われている識別精度だけで機械学習システムの品質を評価するのではなく、分散や標準偏差も考慮することで品質を評価する必要があることが明らかになり、品質を向上させるためのデータ拡張としてはマスク処理と幾何学変換の組み合わせが有効であることが分かった。

3 深層学習におけるデータ拡張の適用法の改良による品質改善

本研究では、基礎・応用を問わず、深層学習が用いられる場面において広く利用されているデータ拡張について、新しい拡張法の開発を行い、多クラス分類のベンチマークデータセットを用いた実験を通して、深層学習の品質への影響を検証した。

3.1 研究目的

データ拡張は、データに変形を加えることでデータ数を増やす技術であり、訓練データ数が少ないときに性能が落ちてしまうという性質をもつ深層学習において、高い効果を発揮する。一方で、データ拡張の有効性は用いるデータに強く依存するため、データ拡張手法の選択や各手法がもつパラメータを適切に設定しなければならない。しかし、データ拡張の理論解析は難しく、汎用的な使用法が確立していないという現状があり、経験的（あるいは慣例的）な使用がなされるケースが多い。これは意図せず不適切な使用をすることにつながり、学習の品質を損ねてしまう。

そこで、データ拡張の経験的な使用からの脱却を促進するために、本研究はデータの多様性に焦点を当てた。多様性を高めることはデータ拡張の本質的な目的であり、それによる影響を調査することは、データ拡張に対する理解を深め、その発展に貢献することが期待される。一般的なデータ拡張によるデータの多様性の増加が汎化性能の向上に大きく影響を及ぼすことは、[9]の研究において実証されている。本研究では、データの多様性を高める拡張手法を新たに提案し、その効果を検証する。近年、複数のデータ拡張操作からランダムに選ばれた操作を学習中に動的に適用する RandAugment[10]という手法が注目されているが、これは多様性を大きく向上させる半面、調整が必要なパラメータも多く、効果的に利用するのは難しい。そこで、本研究では、簡易なアルゴリズムをもつ 2 つの手法（3.2 節の FC-mixup および 3.3 節の Latent DA）を考案した。[9]の研究ではデータの多様性を評価する指標も提案されているが、それを利用した検証は今後の課題とし、本研究の実験では汎化性能に与える影響のみを調査した。

また、データ拡張は、学習前に入力データに対して適用することで訓練データ数を増加させるという使い方をすることがあるが、深層学習においては、学習中、サンプルをモデルに入力するたびに、その都度データ拡張を適用することができるため、適用法について様々な工夫を施すことが可能である。3.2 節および 3.3 節の手法では、この性質をうまく利用し、ニューラルネットワークの中間層で得られるデータに対して動的にデータ拡張を適用している。

3.2 Feature Combination Mixup（FC-mixup）

画像データだけでなく表形式データにも適用できる汎用的なデータ拡張手法として Mixup[11]がある。これは、2 つのサンプルの線形補間によって新たなサンプルを生成する手法であり、次式に示されるように、入力値およびラベルのそれぞれについて同じ比率で線形補間をとる。

$$\begin{cases} \tilde{x} = \lambda x_i + (1-\lambda)x_j \\ \tilde{y} = \lambda y_i + (1-\lambda)y_j \end{cases}$$

ここで、(x_i, y_i)および(x_j, y_j)はi番目およびj番目のサンプルの入力値を表し、λはベータ分布からサンプルした混合率を表す。これをニューラルネットワークの中間層でも行えるように改良したものは Manifold mixup[12]と呼ばれるが、いずれの手法も、2 点間の線分上というデータ分布上の局所的な範囲にしかサンプルが生成されず、またその線分上の点の性質が非線形に変化する分布をもつデータセットに対しては不適切であるという欠点をもつ。

本研究で提案する Feature Combination Mixup（FC-mixup）は、従来の Mixup とは異なる方法でサンプルを混ぜ合わせる手法であり、その概要を図 3.1 に示す。同じバッチ内に含まれる 2 つのサンプル A と B が、ランダムに選ばれた層において、Z_AとZ_Bという特徴量を出力するとする。dをその層の特徴量の総数とすると、FC-mixup は、Z_Aから$d\lambda$個、Z_Bから$d(1-\lambda)$個の特徴量をランダムに抽出し組み合わせて新たなサンプルZ_Xを生成する。その組み合わせの数は、一つのλの値について多数考えられるため、乱数に応じて異なったデータが生成され、したがってデータ分布上の広い範囲にサンプルを生成することができる。FC-mixup は次式のように表現されるので、この式が満たされるようにZ_AとZ_Bを混合する。

$$|Z_A \cap Z_X| = d\lambda$$

このように 2 つのデータがもつ各パーツの組み合わせにより新たなデータを生成する技術は、Puzzle Mix[13]においてもみられるが、これは対象が入力画像に限定される。また、Adversarial mixup resysthesis[14]において類似した手法が利用されているが、オートエンコーダでの使用に限られており、FC-mixup はより汎用的な使用を想定して設計されている。本研究では、生成データの多様性をさらに高めるために、FC-mixup と Manifold mixup[12]を融合した Hybrid1 および Hybrid2 を提案した。Hybrid1 はバッチごとに確率 0.5 でどちらかの mixup を利用する手法であり、Hybrid2 は両手法を同時に実現する手法である。

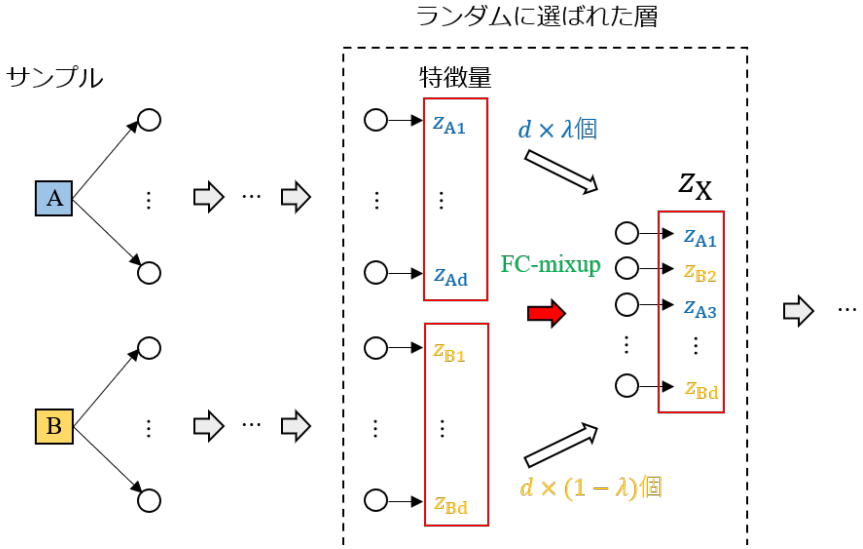


図 3.1　FC-mixup の概要

FC-mixup の性能を調べるために、複数の多クラス分類データセットを用い、従来手法（データ拡張なし、入力層での mixup[11], Manifold Mixup[12]）と提案手法（FC-mixup,Hybrid1, Hybrid2）とで、テストデータの識別精度を比較した。モデルには、WideResNet28-10、ResNet50、多層パーセプトロン（MLP)、小さいサイズの畳み込みニューラルネットワーク（Small CNN）を用いた。フルサイズのデータに加えて、1,000 サンプルをランダムに抽出した Reduced データでも実験を行った。表 3.1 の結果から、多くの場合で提案手法は最も高い精度を示していることがわかる。データセットによっては Manifold mixup より低い精度を示すこともある（ImageNet、Reduced CIFAR-10）が、FC-mixup を試してみることで品質の改善につながる可能性があることが期待される結果となった。

表 3.1 多クラス分類データにおけるテストデータ精度（下線は提案手法）

	CIFAR-10 WRN-28-10	CIFAR-100 WRN-28-10	Core SVHN WRN-28-10	Car Evaluation MLP	Epileptic Seizure MLP	Letter Recognition MLP
Default	96.65	81.15	96.73	92.42	39.18	89.98
Input	96.97	83.29	97.10	92.79	46.19	91.01
Manifold	97.16	83.90	97.36	92.98	45.17	91.23
FC	96.81	**84.15**	97.36	**93.87**	**46.71**	91.16
Hybrid1	**97.19**	83.85	97.49	93.42	45.44	91.18
Hybrid2	96.94	83.88	**97.66**	93.57	46.45	**91.42**
	Full SVHN WRN-28-10	**Tiny ImageNet ResNet50**	**ImageNet ResNet50**	**Reduced MNIST small CNN**	**Reduced CIFAR-10 small CNN**	**Reduced SVHN small CNN**
Default	98.39	62.16	76.54	97.37	43.88	77.87
Input	98.59	65.94	77.14	97.47	44.45	73.51
Manifold	98.60	67.21	**77.26**	97.91	**46.93**	75.55
FC	98.36	66.76	76.81	97.92	45.64	**78.60**
Hybrid1	**98.61**	65.97	76.46	97.93	46.05	77.25
Hybrid2	98.59	**67.55**	77.06	**97.99**	45.59	77.57

3.3 特徴マップへのデータ拡張の適用（Latent DA）

データ拡張は入力データに適用するのが一般的であるが、ニューラルネットワークでは中間層で出力された特徴量を取り出し、データ拡張を適用することが可能である。Manifold mixup[12]はまさにこれを行うが、mixup[11]に特化しており、他の操作は扱えない。本研究では、画像データに対して用いられるアフィン変換やマスク処理といった汎用的なデータ拡張の操作を、中間層で適用することを考えた。CNN では、出力層に近づくにつれ複雑な特徴になるよう階層的に特徴が抽出されるため、バッチごとにランダムに選ばれた様々な層でデータ拡張を行うことで、その効果も異なり、多様なサンプルが生成される。入力画像への適用と同じように、特徴マップに対してデータ拡張を適用することができるので、実装も容易である。

実際に入力画像および特徴マップに対して、マスク処理と平行移動を適用した例を図 3.2 に示す。ここでは、学習中のモデルにサンプルを入力し、同じパラメータ（マスク位置、移動量）に設定したデータ拡張を異なる層で適用した直後の画像を、サイズを揃えて上段に表示している。そのサンプルの最終層における特徴マップを下段に示しているが、それらはデ

ータ拡張を適用した層によって異なる画像となっている。この結果から、様々な層でデータ拡張を行うことは、生成データの多様性の向上につながり、入力データのみにデータ拡張を適用する場合とは異なった学習が行われることがわかる。

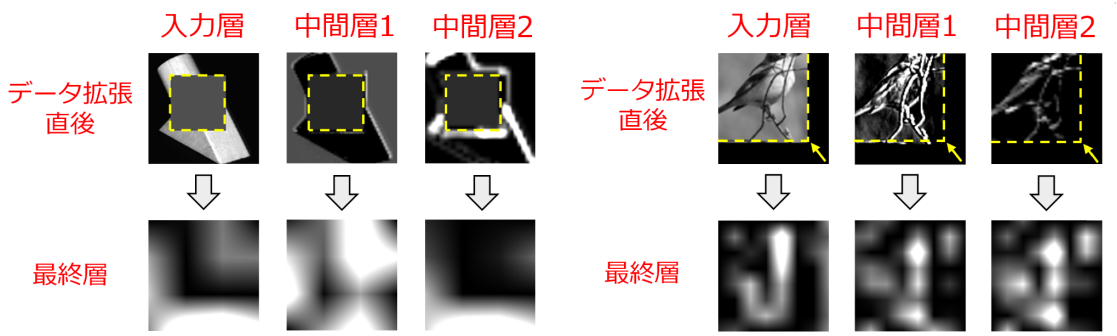
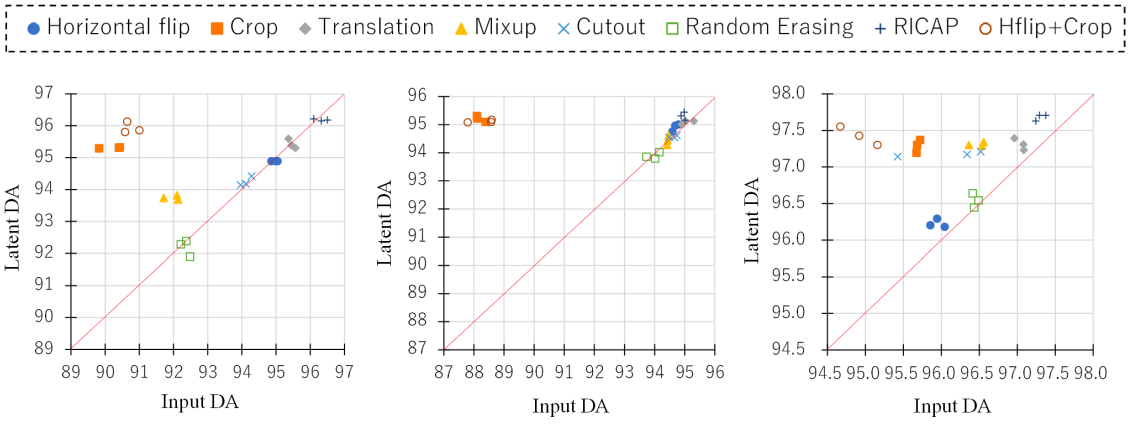


図 3.2　入力画像および中間層で得られる特徴マップにデータ拡張を適用した例

入力層におけるデータ拡張と特徴マップへのデータ拡張の性能を比較するために、様々なデータ拡張を用い、教師ありで学習したモデルのテスト精度を求めた。ここでは、CIFAR-10, Fashion-MNIST, SVHN（補助データを含まない）データセットを用いて、WideResNet28-10 を 200 エポックの間学習した。結果を図 3.3 に示している。各図において、横軸は従来手法（Input DA）、縦軸は提案手法（Latent DA）の精度[%]を表している。これらの結果からわかるように、従来手法よりも提案手法の方が高い精度を示す傾向があり、Crop を用いた結果のように、従来手法では精度が低くなる場合においても、提案手法は高い精度を与えた。この結果から、ランダムな層へのデータ拡張の適用により生成された多様なサンプルは、性能の向上に効果的であることがわかった。



(a) CIFAR-10　(b) Fashion-MNIST　(c) SVHN (core)

図 3.3　Input DA と Latent DA によるテスト精度の比較

4 深層 NN ソフトウェアのデバッグ・テスティング

深層 NN ソフトウェア開発の初期段階では 3 つの観点（実現機能の具体化，学習に用いるデータセットの整備，深層 NN 学習モデルの選択）から試行錯誤的な繰り返しを通して，要求機能ならびに予測性能が達成可能かを確認する。この試行錯誤過程は従来のプログラム開発のデバッグ作業に対応するが，深層 NN ソフトウェア（DNN ソフトウェア）の場合，デバッグ・テスティングの入力データセット生成，訓練学習進行状況の監視と評価，要求実現を阻害する原因の特定と除去といった作業になる。以下，令和 2 年度に実施したデバッグ・テスティングの方法を報告し，得られた実験結果の考察と今後の計画を整理する。

4.1 不具合の直接原因

教師あり DNN 学習の標準的な方法では，訓練・学習と予測・推論の 2 種類のプログラムが関わる。学習データを与えられて学習タスクが決まった時，目的とする DNN ソフトウェアの実現に必要な学習モデルを選び，また，訓練学習過程で用いる方式を決める。利用可能な OSS の学習フレームワーク提供機能を用いる場合，フレームワークのパラメータを決めれば良い。次に，学習データから訓練データセットを構築する。そして，学習モデル・訓練データセットを入力として訓練・学習プログラム（学習フレームワーク提供）を作動させ，その結果，訓練済み学習モデルを導出する。より詳細には，訓練・学習プログラムが求めるのは，訓練済み学習モデルを定義する重みパラメータ値の集まりである。この訓練済み学習モデルが予測・推論プログラムの振舞いを規定する。

利用者からみると，DNN ソフトウェアの実体は予測・推論プログラムである。たとえば分類学習タスクの場合，入力データに対する分類の確からしさを求めるプログラムである。そして，この出力結果を調べることで，構築した DNN ソフトウェアが意図通りに作動しているかを判断する。期待する結果が得られず不具合があるとみなす時，訓練・学習プログラムの実行以前にもどって，欠陥の在り処を調べて除去する。つまり，デバッグ作業を行う。

不具合が生じる時，訓練・学習プログラム実行過程で用いる情報の何処かに欠陥があり，訓練データセット・学習モデル・学習機構のいずれか，あるいは，これらの複数が原因となる。一方，予測・推論結果に不具合をもたらす直接原因は，訓練済み学習モデルあるいは重みパラメータ値の集まりである。訓練データセット・学習モデル・学習機構の欠陥が不具合の根本原因である一方で，直接原因は重みパラメータ値にある。つまり，根本原因となった欠陥は重みパラメータ値の不具合として顕在化し，その不具合が示す訓練済み学習モデルの歪みが利用者からみた不具合の直接原因となる欠陥である[15]。この歪みを計測する方法が必要である。

本章では，重みパラメータ値を測定する内部指標を導入することで，DNN ソフトウェアの不具合を検知できるか否かを調べる。重みパラメータ値は訓練・学習プログラムの出力であるが，その出力値が妥当かを調べる直接の方法はない。その理由は，出力として期待する重みパラメータ値を予め知ることができないことによる。このような期待する重みパラメータ値が既知であれば，訓練・学習は不要になる。その既知の値を使えばよい。

4.2 内部指標

ニューロン・カバレッジ（NC）の考え方を紹介する。学習モデルをニューロンのネットワークとみなす。閾値を決めた時，出力値が閾値を超えるニューロンは活性状態にあるという。学習モデルを構成するニューロン数を N とし，活性状態のニューロン数を A とする時，活性ニューロンの比（NC = A/N）をニューロン・カバレッジと定義する。文献[16]は，NC を検査網羅性の基準と仮定し，評価用入力データの選び方が NC 値，すなわち，訓練済み学習モデルの検査網羅性に影響することを調べた。

図 4.1 訓練済み学習モデル

本章では，NC を計算する対象ニューロンの選び方を工夫することで，不具合の有無を調べる内部指標[17]として用いる。図 4.1 に訓練済み学習モデルの模式的な図を示した。中間層の最終段階（網掛けした層）を対象とするニューロンに対して NC を定義し内部指標とする。

一般に，機械学習の技術では，此の Penultimate Layer を特別の観測対象とすることが多い。たとえば，画像分類タスクの場合，その前段までが画像認識などの具体的なアルゴリズムの役割を果たす相関分析（ピクセル値のパターンの分析）の処理であり，その計算結果が，此の層に集約されることが理由である。そして，本章では，想定される欠陥原因が，この内部状態として顕在化すると仮定する。さらに，この内部状態をもとに，さまざまな統計指標を導出することができる。調べたい不具合によって，何が適切な導出指標かを実験によって調べる。

4.3 実験の方法と結果

いくつかの実験結果を示し，先に定義した内部指標あるいは導出指標の有用性を考察する。最初に，訓練・学習プログラム（学習フレームワーク）に欠陥がある時の比較実験結果を示す。以下，BI は PC に欠陥挿入した訓練・学習プログラムである。

図 4.2 は学習モデルとして古典的な全結合ネットワークを用い，中間層のニューロン数を変化させて，試験データセットの正解率をプロットした。十分な数のニューロンを持つ時（横軸で 50），PC と BI で正解率に大きな差がないことがわかる。つまり，正解率を調べても，PC と BI を区別することが困難であり，その結果，欠陥の有無がわからない。これは，これまでに得られた知見を再確認するものである。以下，この知見（図 4.2）に加えて，さらに状況を系統的に調べる実験の結果（図 4.3）を示す。

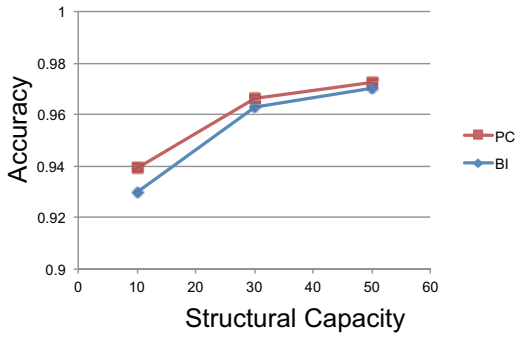


図 4.2 中間層の異なる学習モデル

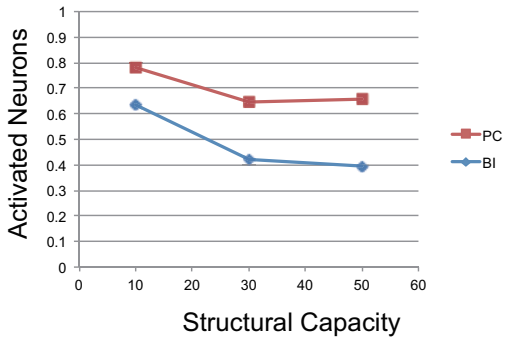


図 4.3 内部指標との関係

図 4.3 は，縦軸に内部指標（本章でのニューロン・カバレッジ）をプロットした。この指標の絶対値を参照すると，たとえば、BI の 10 と PC の 30 とは，共に 0.7 程度であって，BI と PC の区別がつかない。そこで, 内部指標からの導出指標に適切なものがあるかを調べる。今，試験データセットに対するニューロン・カバレッジの集まりを得て，この平均値 μ と分散 σ^2 を求めて，さらに，σ / μ を計算する。横軸にこの導出指標 σ / μ を用いる場合を図 4.4 に示した。縦軸の値から図 4.3 を参照することで，どの学習モデルがどの値かを知ることができる。

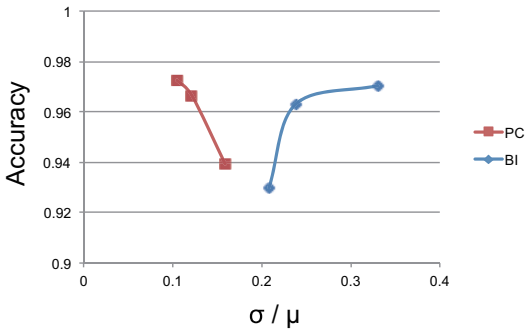


図 4.4 導出指標

図 4.4 によると，PC と BI を区別できることがわかる。つまり，内部指標では，キャパシ

ティ（中間層のニューロン数）が異なる PC と BI を区別できない（図 4.3）が，導出指標を工夫して σ/μ によって比較すると，ニューロン・カバレッジが有用な情報を与えることがわかる。

次に，欠損データを評価用に用いて，PC と BI が個々のデータに対して出力する分類の確からしさを散布図（図 4.5）で表す。

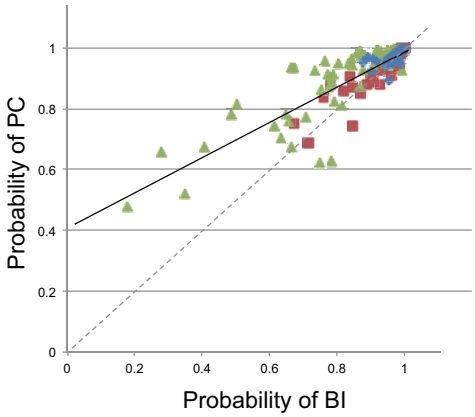


図 4.5 分類の確からしさ

図 4.5 で△が欠損データに対する出力値を表す。PC と BI で同等の値を出力すると仮定すると，原点を通る点線上に分布する筈である。実際，試験データセットから選んだ□は概ね此の線上にのることがわかる。一方，欠損データ（△）は実線上に分布し，PC がより良い分類の確からしさであること示す。つまり，欠陥混入した BI は、正解率は変わらない（図 4.2 を参照）が，頑健性に劣るといえる。

次の実験は，内部指標を用いることで頑健性の違いを検出できることを確認するものである。

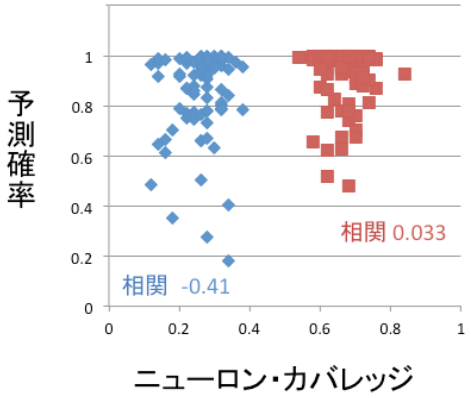


図 4.6 内部指標の違い

図 4.6 は，前記の欠損データを入力し，先に定義した内部指標を横軸にプロットした。右側に分布する□は PC，左側に分布する◇は BI による結果を示す。これによると，(1) PC の内部指標の値が大きいこと，(2) 内部指標と予測確率（分類の確からしさ）の相関が弱いこと，がわかる。次に，σ/μ を計算すると，PC は 0.0876 であり，BI は 0.2183 となった。図

4.6 は頑健性に影響する欠損データを用いる実験であり，σ/μ の値が頑健性と強く関係すると考えられる。

次に，訓練データセットに系統的な歪みを与えて，訓練済み学習モデルを導出する実験を行った。系統的な歪みが生成できることが，これまでの実験からわかっている。この実験は，訓練データセットの違いが，内部指標に影響を与えるかを調べることに相当する。同一の試験データセットに対する正解率をプロットした。

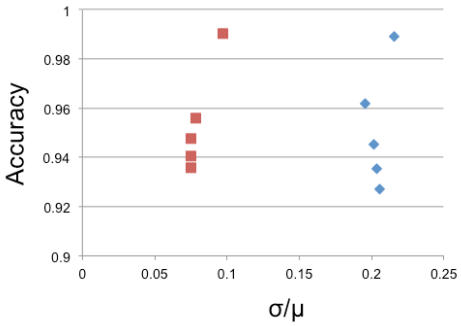


図 4.7　訓練データセットの違い

図 4.7 の上から下へ（正解率の良い方から悪い方へ），大きい歪みの訓練データセットを用いた場合に対応する。つまり，試験データセットのデータ・シフトが訓練時に用いたデータセットから相対的に大きくなるので正解率が低下することを確認できる。一方で，横軸の値（σ/μ）は，PC（□）と BI（◇）で明らかに異なる。この実験の状況は，訓練データセット・シフトの一例と考えて良い。図 4.7 は PC と BI で独立した 2 系列を示す。正解率が示す正確性と σ/μ 値が示唆する頑健性とが独立な観点であることを確認できる。

以上から，訓練データセットの歪みは，内部指標では区別がつきにくいが，正解率に基づく方法で調べることができる。実務で行われているように，訓練データセットの良し悪しの検討に際しては，正解率を調べる方法が有用であると云えよう。一方，訓練・学習プログラムの欠陥など他の要因が関わる可能性がある（多重欠陥が想定される）場合，内部指標や導出指標（σ/μ）の値を同時に調べることが望ましい。

4.4　関連研究

ニューロン・カバレッジは DeepXplore [16]で導入されたメトリックスである。従来のソフトウェア・テスティングで用いられてきたテスト網羅性基準を参考に提案された。従来法は，実行文をノードとする制御フロー・グラフ（Control Flow Graph, CFG）でプログラムを表現する時，あるテスト入力データによって実行される文を基本単位として，テスト網羅性を定義する。最も簡単には，CFG のノードが入力テストによって実行される経路に含まれるか否か，つまり文が実行されるか否かを基準とする。これは，文網羅基準あるいは C0 基準と呼ばれる。DNN 学習モデルはネットワークとして表現されることから，CFG 上で定義された C0 基準との対比により，ノードに位置するニューロンが活性化（出力値が閾値を超えること）されるか否かによって，ニューロン・カバレッジ（NC）を定義した。そして，NC 値を増加させるように新しいテスト入力データを生成する方法を論じた。

ニューロン・カバレッジは従来のテスト網羅性基準から類推できる素直な考え方だろう。その後，NC を向上させる入力データの作成が難しくないという経験から，複数ニューロンの相関や異なる層の間での相関を考慮したメトリックスが提案された[18]。また，NC が増加するように，NC 値をガイドとして，データ補完の方法で有用なテスト入力データを系統的に生成する方法[19]がある。さらに，NC をガイドに用いる方法と GAN をベースとするテスト入力自動生成[20]とを組み合わせる方法が論じられている[21]。データ生成をガイドする指標として，NC が有用なことが確認されたと言ってよい。

テスティングの例として，文献[19]および[20]は回帰問題 DNN モデルの予測結果をもとに計算したステアリング角度というアプリケーションの機能を検査の観点とした。文献[22]は NC 値を大きくするテスト入力が欠陥発見に役立つかを調べた。何を欠陥とするかで NC が有用か否かの判断が異なることを論じている。逆に，この研究[22]は，正確性を中心とする外部指標と NC の間の相関が弱いことを述べている。本章では，両者に相関が弱いという観察から，NC に基づく内部指標を検査に用いた。文献[22]と矛盾しないどころか，同じ方向の議論を展開しているといえる。なお，網羅性は検査終了の判断基準であり，一方，欠陥発見はテスト入力データがコーナーケースを実行するかに依存する。両者は異なる側面を論じているともいえる。実際、従来のソフトウェア・テスティングにおいても，網羅性の向上が必ずしも欠陥発見の効率向上に結びつかないことが報告されている。

本章の方法は，文献[15][17]で論じられているように，NC 値を簡易的な検査指標に用いるというものである。従来の研究が NC を検査の網羅性基準に用いるのに対して，DNN モデルの欠陥が NC 値として現れる，という見方を採用した。実験では，この考え方に基づく具体的な検査として，訓練・学習プログラムの信頼性および訓練済み学習モデルの頑健性を調べることができた。

4.5 おわりに

本章では，Penultimate Layer でのニューロン・カバレッジに基づく内部指標を用いた。これはスカラーであることから，計測ならびに導出指標の定義が容易であり，検査指標として利用しやすい。一方，NC はニューロン個々の値に関する情報を捨象しており，有用な情報が欠落する。実際，文献[23]では，ニューロン値の分布を推定し，これをもとにテスト入力データが妥当かを論じる方法を提案している。これを応用すると，ニューロン値の分布は訓練データセットの歪みを，より詳細に表すと考えられるだろう。今後，この分布を利用する考え方を応用することで，訓練データセットをデバッグする方法を検討する。

5　ロバストネスの評価・向上技術

本章におけるロバストネスは、入力ノイズ（敵対的データも含む）に対する機械学習モデルの耐性である。例えば、どの程度ノイズを付加しても推論結果を維持できるかを評価する。そのロバストネスの指標の一つに最大安全半径がある。本章では、順伝播型ニューラルネットワークを用いた分類器を対象として、敵対的データと最大安全半径について説明した後、最大安全半径を計測する技術と増加させる技術についての調査結果について報告する。

5.1　ロバストネスの指標（最大安全半径）

機械学習ソフトウェアにおいては、訓練済み学習モデル（正確には、学習モデルにもとづく推論プログラム）は、わずかにノイズを付加された入力データによっても誤推論する問題が知られている。そのような誤推論させる入力データは敵対的データ（adversarial example）[24]と呼ばれており、近年、敵対的データに対する研究が活発に行われている。入力データ$x \in \mathbb{R}^n$（$\mathbb{R}$は実数の集合）のδ近傍（半径$\delta \in \mathbb{R}$の球の内側）に含まれる全ての敵対的データの集合$Adv_\delta(x)$は次のように定義できる。

$$Adv_\delta(x) = \{x' \mid \| x - x' \| \leq \delta \ \wedge \ f(x) \neq f(x')\}$$

ここで、$f(x)$は入力xに対する機械学習モデルの出力（分類結果）を表す関数、$\|x - x'\|$は 2 つのデータx, x'の距離（差）を表す。距離の定義にはpノルムが使われることが多い。

図 5.1 を用いて敵対的データについて説明する。図 5.1 の左側はニューラルネットワークへの入力空間、右側がニューラルネットワークからの出力空間を表す。入力空間の赤い球の中心がパンダ画像（元データ）であり、その球（半径δ）の内側が大きさδ未満の微小ノイズを付加した画像の集合（δ近傍）である。このδ近傍の入力画像の集合に対するニューラルネットワークの出力の集合が、右側の出力空間の赤い領域である。ここで、出力側の赤い領域の右下の決定境界を超えて猿の領域に入っている部分が誤分類を表しており、この部分にマップされる入力が敵対的データである。

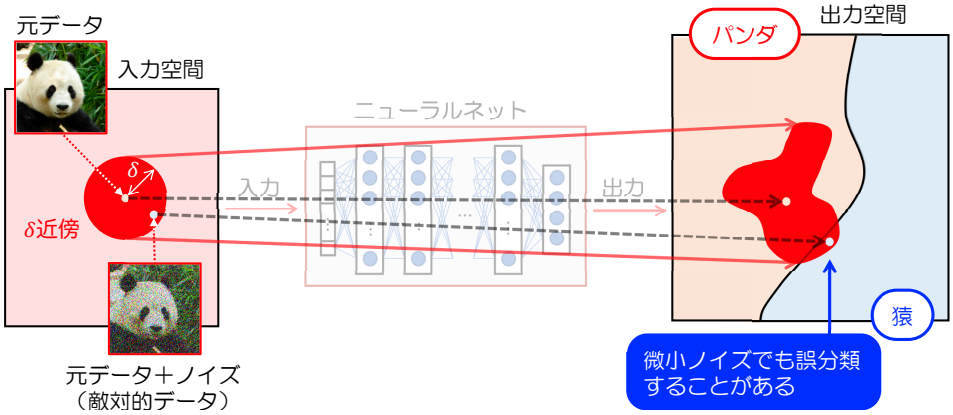


図 5.1　敵対的データ（ノイズ付加パンダ画像を猿に誤分類）

入力データxのδ近傍（xを中心とする半径δの球の内側）に敵対的データが存在しないと

き、δをxの安全半径という。特にxの安全半径の中で最大の半径（最大安全半径, Maximum Safe Radius）$MSR(x)$は次のように定義される。

$$MSR(x) = \max\{\delta \mid Adv_\delta(x) = \emptyset\}$$

この最大安全半径が大きいほど、敵対的データによる攻撃は難しくなるため、機械学習モデルのロバストネス（敵対的データを含む入力ノイズに対する耐性）の指標のひとつとして最大安全半径を使うことができる。

図 5.1 のδ近傍の入力画像の一部は猿に誤分類されるため、このδは安全半径ではない。一方、図 5.2 のδ近傍の入力画像は誤分類されることはないため、このδは安全半径であり、これ以上δを大きくすると誤分類される可能性がある（敵対的データを含む）ため、図 5.2 のδは最大安全半径である。

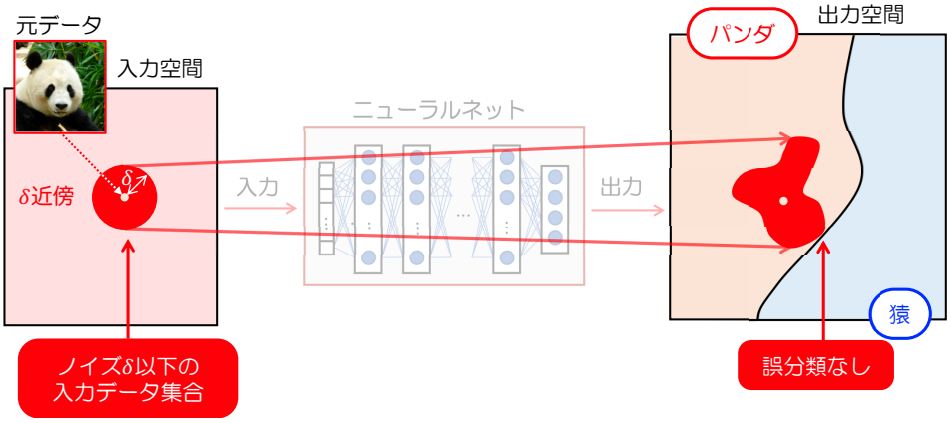


図 5.2　最大安全半径 δ

5.2　ロバストネスの評価・向上技術調査結果

ロバストネスの評価と向上に関する研究論文について調査した結果を表 5.1 に示す。ロバストネスについては多くの研究論文が発表されているが、比較的良い結果が得られている最近の論文を代表として示している。表 5.1 の各論文の下には、その論文で提案されている手法を適用できるスケールの参考情報として、その手法の評価実験に使用されたニューラルネットワークの情報を記入している。表 5.1 は次の観点から整理している。

- 横方向（技術の用途）：
  - ロバストネスの評価：最大安全半径の見積り
  - ロバストネスの向上：指定された最大安全半径をもつデータ数の増加
- 縦方向（評価の信頼度・精度）：
  - 保証あり：δ近傍に敵対的データが存在しないことを保証できる
    - 厳密：最大安全半径の厳密な見積り
    - 近似：最大安全半径よりも小さめ（安全半径のひとつ）の見積り
      - 決定的：δ近傍に敵対的データは存在しない（100%安全）
      - 確率的：δ近傍に敵対的データが存在しない確率はp%
  - 保証なし：δ近傍に敵対的データが存在しないことを保証できない

表 5.1 ロバストネス（最大安全半径）の評価と向上に関する技術

<table>
<tr><th colspan="3"></th><th>ロバストネスの評価</th><th>ロバストネスの向上</th></tr>
<tr><td rowspan="4">保証あり</td><td rowspan="2">厳密</td><td></td><td>最大安全半径の厳密な見積り
Katz et al. 2017 (Reluplex) [25]
ACAS-XU-DNN, 300 ReLU nodes
6 hidden layers,
数百ノード程度が上限

Tjeng et al. 2019 [26]
CIFAR-10, ResNet, 9-CNN, 2-layer,
107,496 ReLU units,
Reluplex より 100～1,000 倍程度高速</td><td></td></tr>
<tr><td rowspan="2">近似</td><td>決定的</td><td>最大安全半径の小さめの見積り
Weng et al. 2018 (Fast-Lin) [27]
CIFAR, 6-layer, 12,288 ReLU units
Reluplex より 10,000 倍程度高速

Boopathy et al. 2019 (CNN-Cert)[28]
CIFAR-10 (32x32x3), 5-layer,
10 filters, 29,360 hidden nodes,
Fast-Lin より高速</td><td>近傍に敵対的データがないように訓練
Wong and Kolter 2018 [32]
SVHN (32x32x3), 2-conv, 32-ch,
100, 10 hidden units, ReLU,
ImageNet への適用は困難</td></tr>
<tr><td>確率的</td><td>確率的最大安全半径の小さめの見積り
Weng et al. 2019 (PROVEN) [29]
CIFAR, 5-layer, CNN, ReLU
CNN-Cert と同程度</td><td>訓練後に保証付ランダムスムージング
Lecuyer at el. 2019 [33]
ImageNet (299x299x3),
Inception-v3 + auto-encoder

Cohen at el. 2019 [34]
ImageNet (299x299x3),
ResNet-50 (50-layer)
Lecuyer [33]より精度向上</td></tr>
<tr><td>保証なし</td><td colspan="2"></td><td>最大安全半径の大きめの見積り
Carlini and Wagner 2017 [30]
ImageNet (299x299x3),
Inception-v3

最大安全半径のおおよその見積り
Weng et al. 2018 (CLEVER) [31]
ImageNet (299x299x3),
ResNet-50 (50-layer)</td><td>近傍の敵対的データを探索しながら訓練
Madry et al. 2018 [35]
CIFAR (32x32x3),
28-10 wide ResNet</td></tr>
</table>

以降、小節 5.2.1～5.2.7 で表 5.1 の各手法について簡単に説明する。

5.2.1　ロバストネス評価、保証あり（厳密）

Katz 等は、与えられた性質を機械学習モデルが満たすことを判定する方法 Reluplex を提案した[25]。その方法を実装した実証用ツールも公開されている。性質は入出力関係についての制約であり、入力データの δ 近傍に敵対的データが存在しないことを網羅的に厳密に（健全かつ完全に）判定できる。すなわち、二分探索などで半径 δ を変えながら敵対的データの有無を判定することによって、最大安全半径を見積もることができる。Reluplex は Simplex 法（線形計画問題の解法のひとつ）に ReLU 関数用の規則を追加した解法であり、実数を扱える充足可能性判定ツール（SMT-Solver）によって実装されている。ロバストネス以外の性質も判定できる強力なツールであるが、計算コストが高く、扱えるニューロン数が ReLU 数百個程度という制約がある。

Tjeng 等は、最大安全半径を効率よく計算する方法を提案し、その方法を混合整数線形計画法ソルバ（MILP）上に実装して、10 万個の ReLU 型ニューロンをもつネットワークの最大安全半径を厳密に求められることを示した[26]。まだ実用的な機械学習モデルに適用するには十分とは言えないが、このようにスケーラビリティ改善の研究は進んでいる。

5.2.2　ロバストネス評価、保証あり（近似、決定的）

Weng 等は、ReLU 型ニューラルネットの最大安全半径を近似的に見積る方法 Fast-Lin を提案した[27]。Fast-Lin では、図 5.3 に示すように、出力領域を多角形で線形近似し、最大安全半径よりも少し小さめの近似値δを見積もる。この近似値δは最大安全半径を超えないため（健全）、δ近傍に敵対的データが存在しないことを保証できる。すなわち、δは安全半径のひとつであり、最大安全半径の下界である（$\delta \leq MSR(x)$）。多角形で線型近似することによって、厳密な方法（Reluplex）よりも 1 万倍以上速い結果が得られたことが報告されている。

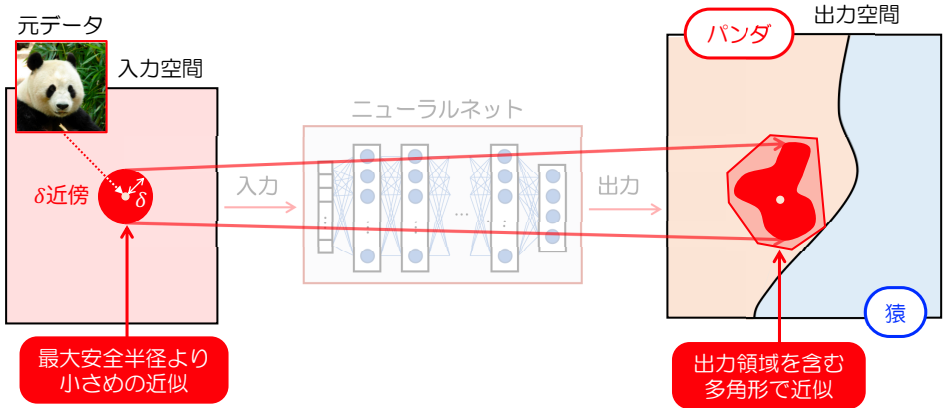


図 5.3　最大安全半径の近似値（少し小さめ）の見積り

Boopathy 等は Fast-Lin を改良した手法 CNN-Cert を提案した[28]。CNN-Cert では、ReLU 以外の活性化関数（sigmoid, tanh, arctan）を含む Convolutional ネットワークにも対応し、Fast-Lin よりも近似精度と計算速度を向上した。

5.2.3　ロバストネス評価、保証あり（近似、確率的）

Weng 等は、確率的な最大安全半径を近似的に見積もる方法 PROVEN を提案した[29]。安全確率ρの最大安全半径δは、図 5.4 に示すように、δ近傍に敵対的データが存在しない確率がρである、すなわち、$(1-\rho)$の確率で敵対的データが存在することを許容する最大の半径である（決定的な最大安全半径は安全確率1の最大安全半径に相当する）。PROVEN は、CNN-Cert をベースに開発されおり、計算量は CNN-Cert から大きくは増えていない。

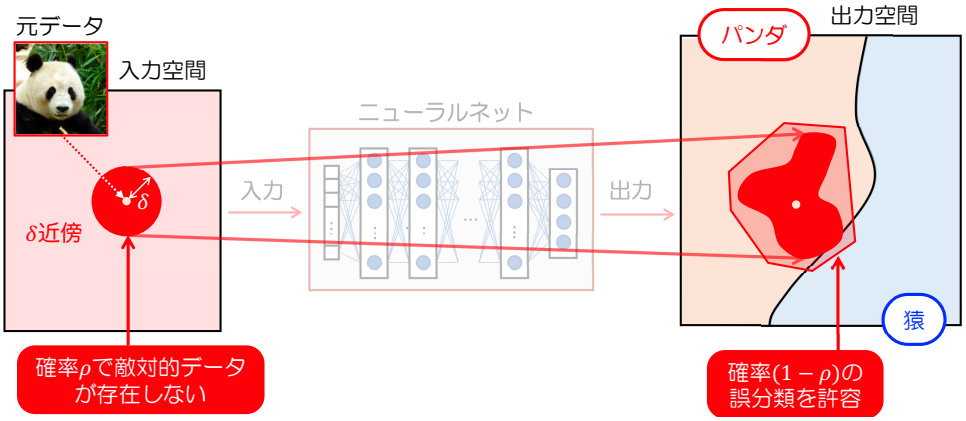


図 5.4　安全確率ρの最大安全半径の近似値の見積り

5.2.4　ロバストネス評価、保証なし

Carlini と Wagner は、既存の最適化ツール（Adam）を用いて、入力データxに最も近い敵対的データを探索し、その距離δを見積もる方法を提案した[30]。ただし、この方法で得られる距離δが実際に敵対的データまでの最短距離である保証はなく、その距離よりも近いところに敵対的データが存在する可能性はある。すなわち、最大安全半径の上界である（$msr(x) \leq \delta$）。この方法で得られる距離δが安全半径である保証はないが、最大安全半径の目安として、最近のロバストネスの論文ではしばしば評価に使われている。

Weng 等は、攻撃方法に依存しないロバストネスの評価値として、おおよその最大安全半径を求める方法 CLEVER を提案した[31]。比較的大きなネットワークにも適用可能な方法であり、画像認識モデル Inception-v3 を 10 秒程度で評価できたと報告している。入力のわずかな変化が出力に与える影響の最大値を極値理論によって推定し、最大安全半径に近い値dを見積もっている。図 5.5 に示すように、見積もった値δは実際の最大安全半径より大きいこともあるため、δ近傍内に敵対的データが存在する可能性はある（安全半径である保証はない）。

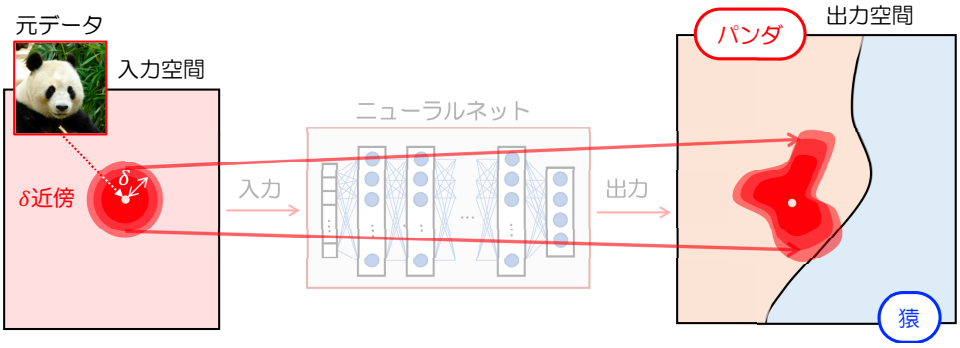


図 5.5　最大安全半径の近似値の見積り（保証なし）

5.2.5　ロバストネス向上、保証あり（近似、決定的）

Wong 等は、訓練データセットの各データの最大安全半径がδ（指定値）になるように訓練する方法（ロバスト訓練）を提案した[32]。この方法は訓練後に全ての訓練データに対して最大安全半径δを保証するものではないが、各入力データの最大安全半径の近似値（安全半径）を見積もる方法を与えている。このロバスト訓練では、各訓練データのδ近傍で最も誤推論する可能性のあるデータに対して正しい推論をするように訓練する。

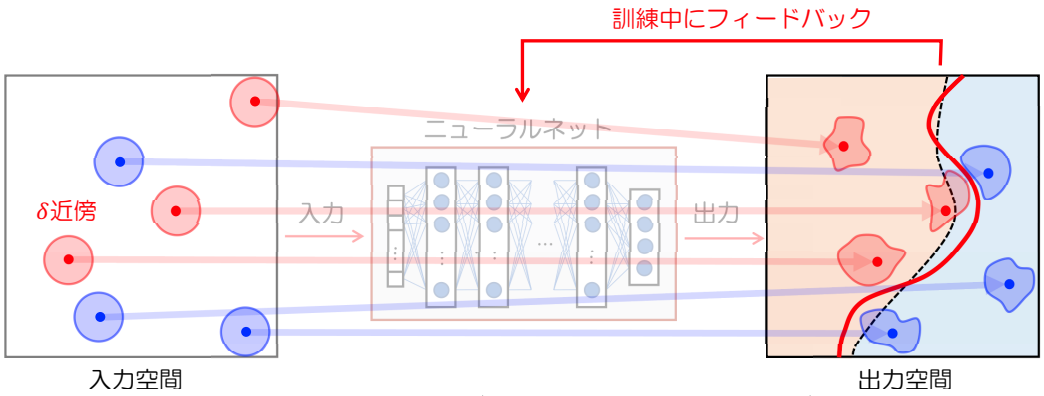


図 5.6　ロバスト訓練（入力のδ近傍も含めた訓練）

ロバスト訓練の様子を図 5.6 に示す。図 5.6 の出力空間の点線は通常の訓練によって学習した決定境界、赤色線はロバスト訓練によって学習した決定境界を表す。入力空間の 6 個の訓練データについては両方の境界線によって正しく分類されているが、各訓練データのδ近傍のデータについては点線の境界線（通常訓練）では誤分類が生じている。一方、ロバスト訓練では赤色の境界線のように、δ近傍のデータについても正しい分類になるように訓練を行う。Wong 等のロバスト訓練は、ロバストな学習モデルを保証付で訓練する方法であるが、スケーラビリティが低く、訓練可能なネットーワークのサイズを大きくできない問題がある。この論文は、MNIST（28×28）と SVHN（32×32）のデータセットに対して有効性を示しているが、ImageNet（256×256）には適用できなかったと報告している。

5.2.6　ロバストネス向上、保証あり（近似、確率的）

Lecuyer 等はランダムスムージングによって確率的に保証可能な最大安全半径を見積もる方法を提案した[33]。ランダムスムージングとは、入力データに防御用のノイズを付加した推論を繰り返し、その複数の出力の平均値を最終出力とする方法である。

ランダムスムージングの様子を図 5.7 に示す。図 5.7 の出力空間の点線は防御用ノイズを付加しない場合の決定境界、赤色線は防御用ノイズを付加した場合の決定境界を表す。ランダムスムージングは決定境界を滑らかにすることによってロバストネスを向上させる技術であり、ImageNet（299×299×3）のような大きな入力データに対する機械学習モデルのロバストネスの保証にも成功している。付加するノイズの分散を増加させると保証可能な最大安全半径も増加するが、一方で正解率等の推論精度は低下する。この論文では、差分プライバシの技術（類似した 2 つの入力に対する出力をノイズ等によって統計的に区別できなくする技術）を適用し、保証可能な最大安全半径、防御ノイズ付推論回数、許容される敵対的データの存在確率等の関係を明確にした。

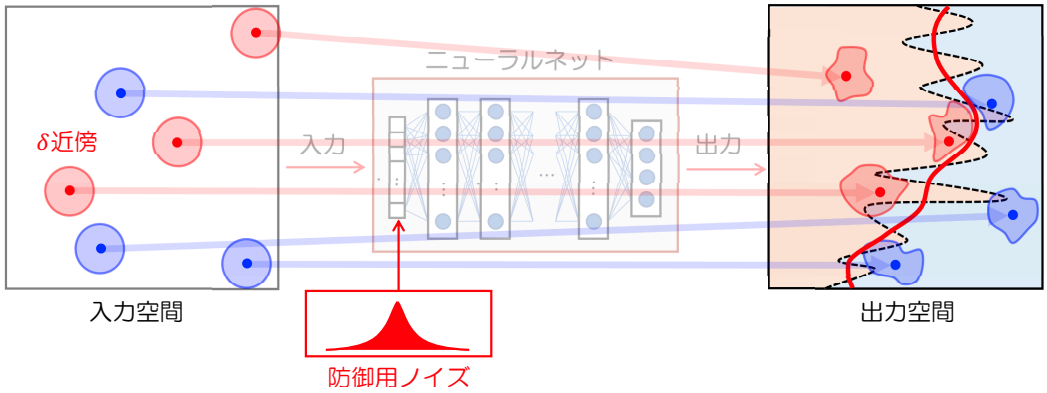


図 5.7　ランダムスムージングによるロバストネスの向上

Cohen 等は、ランダムスムージングによって確率的に保証可能な最大安全半径を、Lecuyer 等の方法[33]よりも精度良く（より大きく）見積もる方法を提案した[34]。

ランダムスムージングは 1 回の推論を行うために内部で複数回（実験的には数十～数百回程度）推論する必要はあるが、Lecuyer や Cohen 等の研究によって、大規模なネットワークに対しても、ロバストネスを確率的に保証することが可能になる。

5.2.7　ロバストネス向上、保証なし

Madry 等は訓練データセットの各データの最大安全半径がδ（指定値）になるように訓練する方法（敵対的訓練）を提案した[35]。この方法では、訓練中に各訓練データのδ近傍で敵対的データになる可能性のあるデータを探索し、そのデータに対しても正しい推論を行えるように訓練する。Wong 等の保証付のロバスト訓練[32]と比較して、ロバストネスを保証することはできないが、ロバスト訓練よりも大きなネットワークに適用可能である。ランダムスムージングのように推論時に複数回の推論を行う必要がなく、ロバストネス向上のための候補技術になりうる。

5.3 まとめ

一般に、ロバストネスを向上させると正解率が低下する傾向にあり、現在は正解率などの評価指標の方が重視されることが多い。しかし、ロバストネスを考慮しない場合、わずかなノイズでも正解率が低下する可能性があるため、誤判断によるリスクが高い場合はロバストネスによる評価も重要である。今回調査したロバストネスに関する技術は最近（2019 年頃）の論文で提案されたものであり、まだこれらの技術を容易に利用できる評価環境は整備されていないが、技術的には実用的なニューラルネットワークにも適用可能になりつつある。今後、そのような評価環境が整備されれば、ロバストネスもニューラルネットワークの一般的な評価指標のひとつになりうると考える。

6　汎化誤差上界の見積り技術

本章では、未知の入力データに対する機械学習の振舞い保証を目的として、機械学習モデル、特に分類器の汎化誤差上界の見積り技術の調査結果について報告する。汎化誤差とは全ての入力に対する分類器の出力の不正解率の期待値である。

6.1　汎化誤差上界の見積り方法の概要

本章では、図 6.1 に示すような順伝播型ニューラルネットワークを用いた分類器 f_w（重み行列 w の学習モデル）を対象とする。また、各分類対象データ x に対して正解の分類クラス y が存在し、(x, y)は分布$\mathbb{P}$にしたがうと仮定する。学習モデル f_w の出力層では各クラスにひとつのニューロンが割り当てられており、入力 x に対するクラス y の出力ニューロンの値を $f_w(x)[y]$と書く。入力xの分類結果 $f_w(x)$ は、$f_w(x)[y]$が最大となるクラス y である。

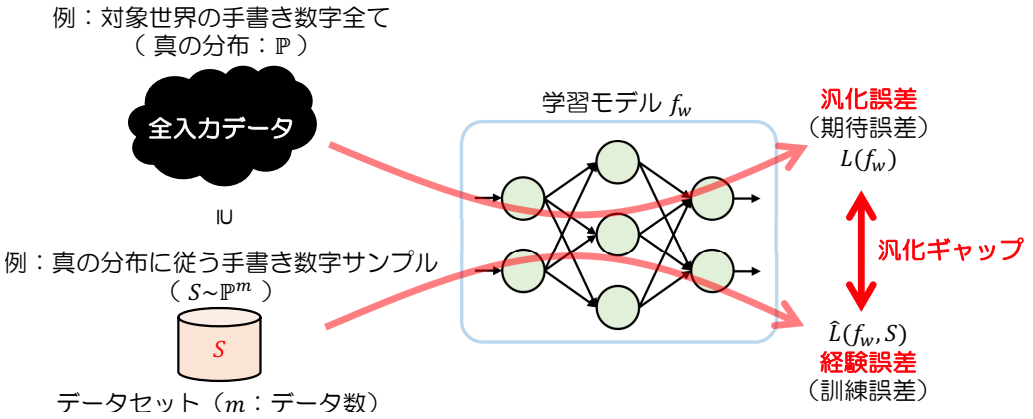


図 6.1　汎化誤差と経験誤差

このとき、汎化誤差とは、分布$\mathbb{P}$にしたがう全入力データに対する学習モデル f_w の不正解率の期待値 $L(f_w)$ であり、次式によって定義される。

$$(f_w) = \mathbb{E}_{(x,y)\sim\mathbb{P}}\left[\mathbb{I}\left(f_w(x)[y] \leq \max_{y'\neq y} f_w(x)[y']\right)\right]$$

ここで、$\mathbb{I}(b)$ は、$b = True$ ならば 0 、$b = False$ ならば 1 を返す指示関数である。

一方、経験誤差とは、分布$\mathbb{P}$にしたがうm 個の入力データサンプルの集合 $S\sim \mathbb{P}^m$ に対する学習モデル f_w の不正解率 $\hat{L}(f_w, S)$ であり、次式によって定義される。

$$\hat{L}(f_w, S) = \frac{1}{m}\sum_{(x,y)\in S}\left[\mathbb{I}\left(f_w(x)[y] \leq \max_{y'\neq y} f_w(x)[y']\right)\right]$$

一般に入力空間には無数の入力データが存在するため、汎化誤差を正確に計算することは困難である。そこで、汎化誤差の上界を確率的に見積もる方法が研究されている。例えば「この学習モデル f_w の汎化誤差は○%以下であることを 99% の確率で保証する」というようなことが可能になる。例えば、そのような汎化誤差上界を見積もるための基本的な方法として次の 3 つの方法が知られている[36]。

- **構造による汎化誤差上界の見積り方法**：学習モデルの構造（層数、ニューロン数など）によって汎化誤差上界を見積もる方法である。
- **出力マージンによる汎化誤差上界の見積り方法**：出力層の正解と不正解のニューロンの出力の差（出力マージン）によって汎化誤差上界を見積もる方法である。
- **ロバストネによる汎化誤差上界の見積り方法**：学習モデルの重み行列にノイズを付加した場合の出力のロバストネスによって汎化誤差上界を見積もる方法である。

以降、6.2～6.4 節でこれら 3 つの基本的な見積り方法について順番に説明し、6.5 節で汎化誤差上界の見積り精度の向上について述べる。

6.2 構造による汎化誤差上界の見積り方法

よく知られている汎化誤差上界の見積り方法のひとつに VC 次元に基づく方法がある。VC 次元はニューラルネットワークの構造に対して与えられる値であり、その構造で分類可能なデータ数の最大値を表す。学習モデル（順伝播型ニューラルネットワーク）fの VC 次元 $VC(f)$の上界は次式により与えられることが示されている[37]。

$$VC(f) \leq d + \left(\sum_{i=1}^{d} (d - i + 1)\omega_i \right) \log_2 \left(8e \sum_{i=1}^{d} i\alpha_i \log_2 \left(4e \sum_{j=1}^{d} j\alpha_j \right) \right)$$

ここで、d は層数、α_i は第i層のユニットの個数、ω_i は第i層への重みの個数であり、eはネイピア数（$e = 2.7182\cdots$）である。

学習モデル f の汎化誤差 $L(f)$ の上界は、VC 次元を用いて次の不等式によって見積もることができる。この不等式は確率 $(1 - \delta)$ で成り立つことが示されている[36]。

$$L(f) \leq \hat{L}(f) + 144K \sqrt{\frac{VC(f)}{m}} + \sqrt{\frac{\ln \frac{1}{\delta}}{m}}$$

ここで、m は訓練データセットのサイズ、K は分類クラス数である。パラメータδ を 0 に近づけるとこの不等式の成り立つ確率は 100% に近づくが、そのとき右辺は無限に大きくなるため、δ は適度な値に設定する必要がある。例えば、$\delta = 0.01$ にすれば、汎化誤差は上式の右辺の値以下であることを 99% の確率で保証できる。

このような構造（VC 次元）による汎化誤差上界では、任意の訓練データセットと任意の訓練アルゴリズムで得られる任意の学習モデルの汎化誤差の上界を見積もることになる。すなわち、汎化誤差上界の最悪値を見積もるため、その値は 100% を大きく超えることが多く、学習モデルの汎化性能の評価に利用することは難しい。また、上記の不等式は、ニューラルネットワークの構造の各パラメータと汎化誤差上界の関係を定式化しているが、訓練パラメータ数とともに汎化誤差上界が増加することを示しており（経験的には十分な訓練パラメータがあるときは増加しない傾向にある）、必ずしも構造と汎化誤差の関係を表していないところもある。

6.3 出力マージンによる汎化誤差上界の見積り

出力マージンとは、出力層の正解クラスのニューロンの出力値とそれ以外のニューロンの出力値の最大値との差である。例えば、手書き数字の分類器に 7 の画像を入力したときの出力層の各ニューロンの出力値が図 6.2 のようになった場合（正解クラスのニューロンの出力値は0.8、それ以外のニューロンの出力値の最大値は0.6）、その出力マージンは0.2である。

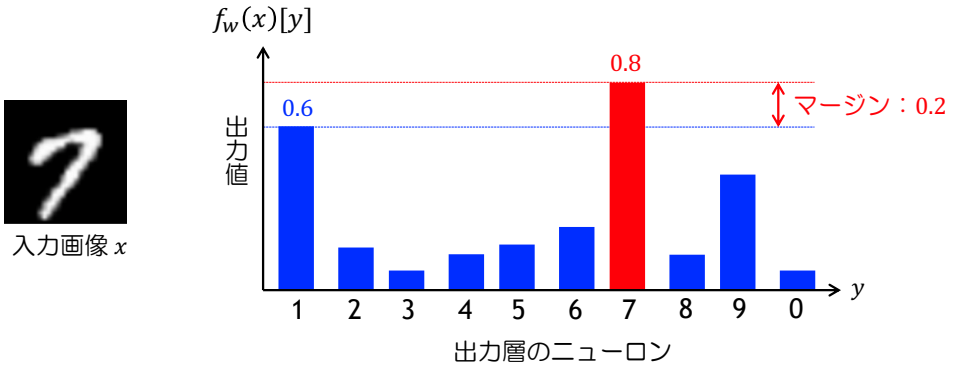


図 6.2 手書き数字分類器の出力マージンの例

2 つの学習モデルが、あるデータセット に対して同じ経験誤差（不正解率）をもつ場合でも、出力マージンの大きい学習モデルの方がノイズなどに対して耐性があり、一般に安定した推論を行うことができる。そのような出力マージンを考慮した経験誤差は次式により定義される。

$$\hat{L}_\gamma(f_w, S) = \frac{1}{m}\sum_{(x,y)\in S}\left[\mathbb{I}\left(f_w(x)[y] \le \gamma + \max_{y'\neq y} f_w(x)[y']\right)\right]$$

この式は、正解クラスのニューロンの出力値が他のニューロンの出力値の最大値よりも大きくても、その差（出力マージン）が閾値 γ 以下ならば不正解としてカウントされることを意味している。

訓練済み学習モデル f_w の汎化誤差 $L(f_w)$ の上界は、出力マージンを用いて次の不等式によって見積もることができる。この不等式は確率 $(1-\delta)$ で成り立つことが示されている[36][38]。

$$L(f_w) \le \hat{L}_\gamma(f_w, S) + \sqrt{\frac{\left(42\sum_{i=1}^{d}\left(2\sqrt{\omega_i} + \sqrt{2\ln(2d)}\right)\right)^2 \prod_{i=1}^{d}\|w_i\|_2^2 \times \sum_{i=1}^{d}\frac{\|w_i\|_F^2}{\|w_i\|_2^2} + \ln\left(\frac{m}{\delta}\right)}{\gamma^2 m}}$$

ここで、d は層数、w_i は第i層への重み行列、ω_i はw_i の要素数、m は訓練データセットのサイズである。なお、$\|w\|_2$ は行列 w のスペクトルノルム、$\|w\|_F$ は w のフロベニウスノルムである。

この不等式の右辺の第 2 項は出力マージンの閾値 γ を大きくすると汎化ギャップ（汎化誤差と経験誤差の差）が小さくなることを表している。一方で、右辺の第 1 項 $\hat{L}_\gamma(f_w, S)$ は γ とともに増加する。これらのことは、出力マージンの大きい学習モデルの汎化誤差は小さ

くなることを意味しており、経験的な傾向とも一致している。一般に、この不等式による上界の見積りも 100% を超えることは多く、汎化性能の絶対的な評価への適用は難しいが、閾値 γ は汎化性能を相対的に評価する指標（汎化尺度）として有効である[36]。

6.4　ロバストネスによる汎化誤差上界の見積り

ロバストネスによる汎化誤差上界の見積り方法は PAC-Bayesian と呼ばれる分析方法を基礎にしている。この分析方法によって、訓練データセットに依存しない重みの分布 P（訓練前の分布）と依存する分布 Q（訓練後の分布）について、次の不等式が確率 $(1-\delta)$ で成り立つことが示されている[39]。

$$\mathbb{E}_{w\sim Q}[L(f_w)] \leq \mathbb{E}_{w\sim Q}\left[\hat{L}(f_w, S)\right] + 4\sqrt{\frac{1}{m}\left(\mathrm{KL}(Q||P) + \ln\frac{2m}{\delta}\right)}$$

ここで、$\mathrm{KL}(Q||P)$は KL ダイバージェンスと呼ばれる情報量であり、2 つの分布Q, Pの異なる度合い（一致するときは0）を表す。例えば、P が平均 0，標準偏差 σ の正規分布、Q が平均 w，標準偏差 σ の正規分布ならば、分布Q, Pの KL ダイバージェンスは次式により与えられる。

$$\mathrm{KL}(Q||P) = \frac{\|w\|_2^2}{2\sigma^2}$$

正規分布に対する KL ダイバージェンスの式を PAC-Bayesian に基づく不等式に代入して、訓練済み学習モデル f_w の重みに正規分布のノイズを付加したときの汎化誤差 $L(f_w)$ の期待値 $\mathbb{E}_{u\sim N(0,\sigma)^\omega}[L(f_{w+u})]$ の上界は、次の不等式によって見積もることができる。この不等式は確率 $(1-\delta)$ で成り立つことが示されている[36]。

$$\mathbb{E}_{u\sim N(0,\sigma)^\omega}[L(f_{w+u})] \leq \mathbb{E}_{u\sim N(0,\sigma)^\omega}\left[\hat{L}(f_{w+u}, S)\right] + 4\sqrt{\frac{1}{m}\left(\frac{\|w\|_2^2}{2\sigma^2} + \ln\frac{2m}{\delta}\right)}$$

ここで、w は重み行列、m は訓練データセットのサイズである。この不等式の右辺の第 1 項 $\mathbb{E}_{u\sim N(0,\sigma)^\omega}\left[\hat{L}(f_{w+u}, S)\right]$ は、図 6.3 に示すように、訓練済み学習モデル f_w の重みに平均 0，標準偏差 σ の正規分布ノイズを付加した場合の経験誤差 $\hat{L}(f_{w+u}, S)$ を複数回測定した平均値によって近似できる。

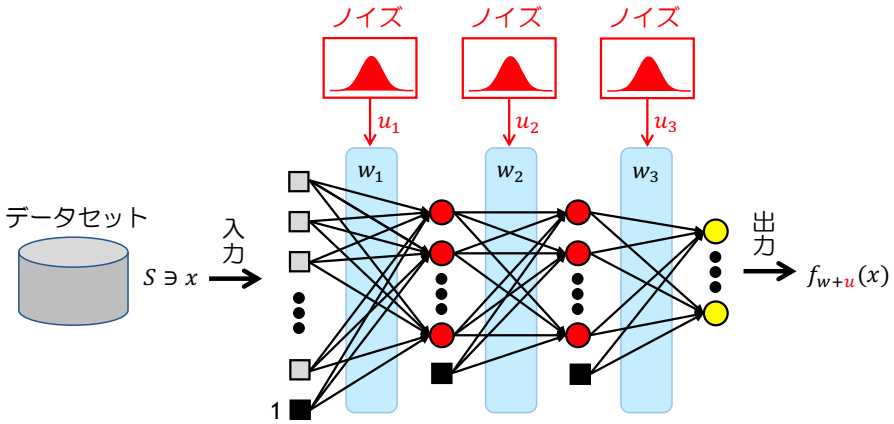


図 6.3　各重み（訓練パラメータ）へのノイズ付加

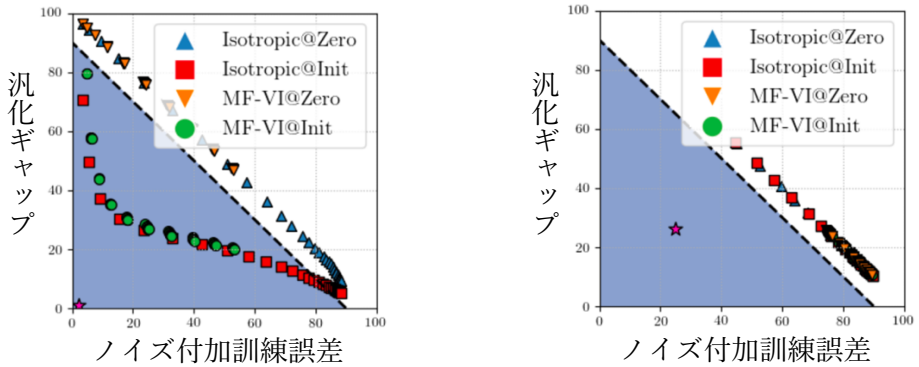
一般に、このロバストネスによる汎化上界の見積り方法は、6.2 節の構造や 6.3 節の出力マージンによる見積り方法よりは小さな上界を見積もることができるが、それでも 100%を超えることも多い。なお、この不等式の右辺の第 2 項はノイズの標準偏差 σ を大きくすることによって汎化ギャップを小さくできることを示している。一方で、右辺の第 1 項はノイズ σ とともに増加する。これらのことは、ロバストネスの高い（ノイズに対して耐性のある）学習モデルの汎化誤差は小さくなることを意味しており、経験的な傾向とも一致している。6.3 節の出力マージンの閾値 γ と同様に、σ は汎化性能を相対的に評価する指標（汎化尺度）として有効である[36]。

6.5　汎化誤差上界の見積り精度

6.2～6.4 節で紹介した基本的な見積り方法では、汎化誤差上界の計算結果が 100% を超える無意味な値になることが多い。これは、汎化誤差の研究の多くが、「訓練パラメータ数が訓練データ数より非常に多くても深層学習では汎化性能が得られる理由」の理論的な解明等を目標にしており、汎化性能の評価を目標にはしていなかったためである。その一方で、2019 年頃から汎化誤差上界を 100% 未満に抑えることを目標にした論文も発表されるようになってきた。以下、 100% 未満の無意味でない（non-vacuous）汎化誤差上界の見積りに関する最近の論文を 3 件紹介する。

6.5.1　汎化誤差上界見積り結果の分析方法

Pitas 等は PAC-Bayesian に基づく（ロバストネスによる）汎化誤差上界の見積り方法によって無意味でない上界が得られるかを分析する方法を提案した[39]。この分析方法では、図 6.4 に示すように、横軸に訓練誤差（経験誤差）、縦軸に汎化ギャップをとり、その平面上に訓練済みの学習モデルをプロットした散布図を作成する。図 6.4 には、4 種類（初期値：Zero, Init、最適化：Isotropic, MF-VI）の学習モデルを MNIST と CIFAR-10 で訓練したときの汎化ギャップの計算結果がプロットされている。各図の左下の灰色の領域は汎化誤差（ノイズ付加訓練誤差と汎化ギャップの和）が90%未満の無意味でない領域を表している（クラス数は 10 のため、ランダムの分類器でも汎化誤差 90% は得られる）。



(a) MNIST データセット　　(b) CIFAR-10 データセット

図 6.4　ノイズ付加による訓練誤差と汎化ギャップの計算例（論文[39]の図 2）

6.4 節のロバストネスによる汎化誤差上界の見積り方法で説明したように、重みに付加するノイズは小さくても大きくても見積もられる汎化誤差上界は大きくなる。この分析方法はその最適なノイズの大きさを評価するために有効である。図 6.4 の例では、MNIST で訓練した学習モデルの汎化誤差は 50% 程度となっているが、CIFAR-10 では 100% 未満の汎化誤差上界は得られていない。

6.5.2 学習モデル圧縮による汎化誤差上界見積り

Zhou 等は、PAC-Bayesian に基づく汎化誤差上界の見積り方法と学習モデルの圧縮法を組み合わせて、ImageNet を例に、実用的なデータセットで訓練した学習モデルでも無意味でない100%未満の汎化誤差上界が得られることを示した[40]。ImageNet は 120 万枚の画像と 1000 の分類クラスをもつデータセットであり、Zhou 等の論文以前は100%未満の汎化誤差上界は得られていなかった。Zhou 等の見積り方法は、枝刈りと量子化の方法で圧縮した学習モデルの汎化誤差上界の見積り方法を提案しており、これによってより小さな（タイトな）汎化誤差上界の見積りを可能にしている。この実験では、ImageNet で訓練した学習モデルの圧縮後の汎化誤差は 96.5% 以下であることを 95% の確率で保証できたことが報告されている。96.5% は 100% に近い値ではあるが、100% 未満にできたことが重要である。

6.5.3 周辺尤度 PAC-Bayesian による汎化誤差上界見積り

Pérez 等は、既存の汎化誤差上界の見積り方法をサーベイし、その適用範囲（課される仮定の強さ）によって分類した[41]（2020 年 12 月公開）。この論文の詳細な内容については現在も調査中であるが、既存技術の中では、周辺尤度を用いた PAC-Bayesian による汎化誤差上界見積りの精度が高いことが示されている。

6.6 まとめ

本調査の結果として、既存手法による汎化誤差上界の計算結果は 100% を超える無意味な値になることが多く、汎化性能の評価指標として適用することは、現時点では難しいことがみえてきた。その一方で、2019 年頃から汎化誤差上界を 100% 未満に抑えることを目標にした論文も発表されるようになってきており、その汎化誤差上界の見積り精度は改善されつつある。数年後には、汎化誤差上界も機械学習モデルの汎化性能の評価指標のひとつになりうると考える。今後は、最新の汎化誤差上界の見積り方法の調査を継続するとともに、実際に汎化誤差上界を計算するための実験を行い、その有効性を確認する。

7 敵対的データ検出技術

7.1 研究概要

与えられた入力画像が敵対的データ（Adversarial Example）であるかを判別する方法を実用的に確立することを目標として、敵対的データを生成する攻撃と検出手法について、下記の点に着目して代表的な技術の調査を実施している。

- 敵対的データ検出プログラムコードの裏付けと計算実験による確認
- 敵対的データ検出手法の論文の実験結果の再現

敵対的データ検出（Adversarial Examples Detection）とは、与えられた入力の中から敵対的データを検出することであり、既存の最先端の敵対的データ検出方法は次の 4 つの主要なカテゴリに分類できる。

① メトリックベースアプローチ（例. [42]）
② ディノイザーアプローチ（例. [43]）
③ 予測不整合ベースアプローチ（例. [44]）
④ ニューラルネットワーク不変式チェック（NIC）アプローチ（例. [45]）

本章では、これら①～④の各アプローチに基づく敵対的データ検出手法を比較・評価するために追試実験を行った結果について報告する。論文[45]に報告されているように、④のアプローチ（NIC：Neural Network Invariant Checking）が①～④の中で最も高い検出率を示すことを確認できた。この追試実験において、①～③については公開されている実装コードを使用したが、④については実装コードが公開されていなかったため、論文[45]にしたがって NIC を実装して計算機実験を行った。そのため、本章では主に④の NIC について説明する。

以降、4 つのアプローチの概要を説明したのち、NIC による敵対的データの検出方法を説明して、その実装方法について述べる。最後に、各アプローチの追試実験と NIC による実験の結果について述べる。

7.2 敵対的データ検出アプローチの概要

以下、最先端の敵対的データ検出のための 4 つのアプローチの概要について説明する。

7.2.1 メトリックベースアプローチ

入力（および各ニューロンの出力）の統計的測定を実行して、敵対的データを検出する方法であり、Ma 等は、最近、Local Intrinsic Dimensionality（LID）と呼ばれる測定の使用を提案した[42]。この方法では、サンプルの距離分布と個々のレイヤーの近隣の数を計算することによって、サンプルを囲む領域の空間充填能力を評価する LID 値を推定し、敵対的データ

が大きな LID 値を持つ傾向がある性質を用いて、敵対的データを検出している。LID は、敵対的データの検出に対して、従来のカーネル密度推定（KD）やベイジアン不確実性（BU）よりも優れており、現在この種の検出器の最先端の技術となっている。

7.2.2 ディノイザーアプローチ（Denoiser、ノイズ除去）

各入力に対して前処理ステップでノイズを除去することによって敵対的データを検出する方法である。この方法では、学習モデル内の主要なコンポーネントを強調できるように、学習モデルまたはノイズ除去器（エンコーダーおよびデコーダー）をトレーニングして画像をフィルター処理する。このフィルターを用いて、攻撃者が敵対的データを生成するために追加したノイズを除去し、誤分類を修正することができる。MagNet[43]は、検出器とリフォーマー（トレーニング済みの自動エンコーダーと自動デコーダー）を使用して、敵対的データを検出する方法である。

7.2.3 予測不整合ベースのアプローチ（Prediction inconsistency based approach）

元のニューラルネットワークと人間の知覚可能な属性で強化されたニューラルネットワークとの間の不一致を測定して、敵対的データを検出する方法である。この方法の最先端の検出技術であるフィーチャスクイージング（Feature Squeezing）[44]は、さまざまな攻撃に対して非常に高い検出率を実現することができる。フィーチャースクイージングは、ディープニューラルネットワーク DNN の不必要に大きな入力特徴空間によって攻撃者が敵対的データを生成できることに着目しており、勾配ベースの攻撃の検出に焦点を当てている。フィーチャースクイージングによる敵対的データの検出手順を以下に示す。

1. 元の入力画像にスクイージング技術（画像の色深度を減らし、画像を平滑化する技術）を適用して複数のスクイーズ画像を生成する。
2. 元の入力画像と複数のスクイーズ画像をディープニューラルネットワークに入力し、入力画像の推論結果（予測ベクトル）と各スクイーズ画像の推論結果との距離を測定する。
3. 元の入力画像とスクイーズ画像の差（距離）の一つがしきい値を超える場合に、元の入力画像を敵対的データとして検出する。

7.2.4 ニューラルネットワーク不変性チェック（NIC）アプローチ

NIC[45]では、ニューラルネットワーク内部の値の不変量（VI: Value Invariants）と来歴不変量（PI: Provenance Invariants）に着目する。値の不変量 VI は各層の可能なニューロン値の分布であり、来歴不変量 PI は 2 つの連続した層の可能なニューロン値パターン（2 層にわたるフィーチャ間の相関の要約）である。ある入力がこれらの不変量に違反している場合に、その入力は敵対的データとして検出される。それらの不変量 VI と PI を良性の入力データで訓練し、敵対的データを検出する 1 クラス分類（OCC）問題としてモデル化する。上で説明した①～③に基づく手法よりも高い検出率が報告されている[45]。以降、NIC のシステム設計概要と実装について、各々7.3 節と 7.4 節で詳しく説明する。

7.3 NIC のシステム設計概要

NIC の検出器の構築と検出の手順（ステップ A～C：訓練時、D～E：実行時）を図 7.1 を用いて説明する[45]。この不変量 VI,PI の訓練では、敵対的ではない良性のデータのみを使用する。

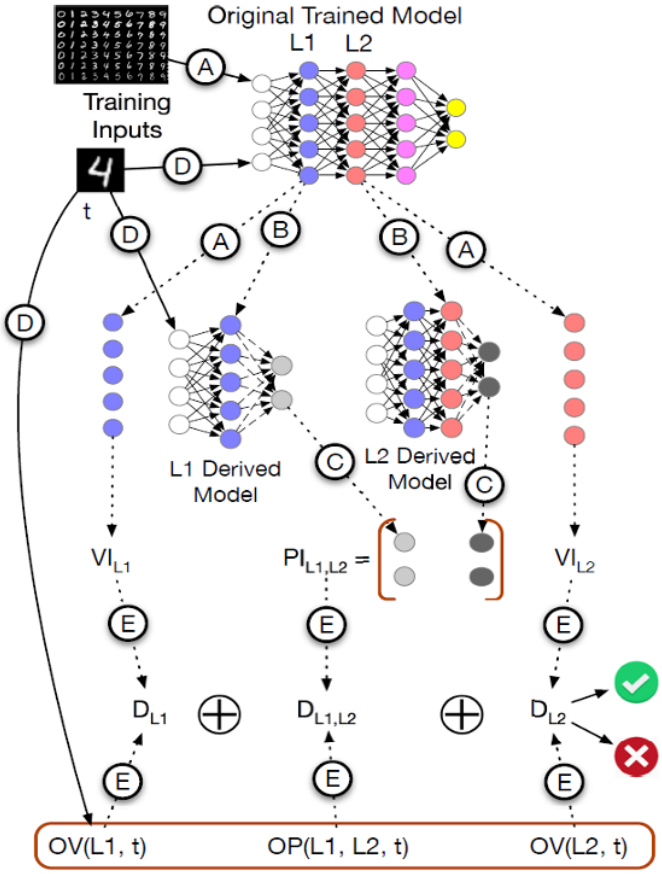


図 7.1 システム設計概要（論文[45]の図 8）

- **ステップ A**：各訓練データ入力の各層で各ニューロンの出力値を収集する。
- **ステップ B**：各層 k（例えば、L1、L2）に対して、入力層から k 層までのサブモデルを抽出し、元のモデルと同じ出力ラベルをもつ新しい softmax 層を追加して、派生モデル（図 7.1 中の DerivedModel）を作成する
- **ステップ C**：すべての派生モデルに対して各良性訓練データを入力し、これらのモデルの最終出力（すなわち、個々のクラスの出力確率値）を収集する。連続する層の組ごとに、この派生モデルの分類結果の分布を用いて訓練する。この訓練された分布が、これらの 2 つの層の PI となる。
- **ステップ D**：各テストデータ t（例えば、図 7.1 中の“4”の画像）を、元のモデルの他に、すべての派生モデルにも入力して、元のモデルの各層の活性化値を観測値 OV（例えば、図 7.1 中の OV(L1,t)）と、連続した層の派生モデルの分類結果（組）を収集す

る。この分類結果から観測された出処 OP（例えば、OP(L1,L2,t)など）を得る。

- **ステップ E**：OV と OP が対応する VI と PI の分布に適合する確率 D を計算する。入力 t が敵対的である可能性を、これらの D 値をすべて集約して同時予測する。

7.4 NIC のシステム実装

NIC に基づいて敵対的データを検出するために、PI と VI から直和空間（ベクトル）を構成し、このベクトルを分類するための OSVM（One Class Support Vector Machine）を構築する。訓練済みの DNN（Deep Neural Network）のモデル（以降、これを M と記述する）の層 l に対する入力を x_l とするとき、層 l の出力 f_l は次式により与えられる。

$$f_l = \sigma(x_l \cdot w_l^T + b_l)$$

ここで、σ は層 l の活性化関数、w_l^T は重み行列、b_l はバイアスである。このとき、VI と PI、および OSVM で分類する直和空間は次のように求められる。

- **VI の計算**：モデル M の各層 l の VI は以下の最適化問題を解いて決定する。

$$VI_l = \min\left[\sum_{x \in X_b} J(f_l \circ f_{l-1} \circ \cdots \circ f_1(x) \cdots w^T - 1)\right]$$

ここで、J はエラー評価関数、X_b は M を作成する際に使用したバッチである。また、$\circ$はモノイドであり、この場合では、f_k をベクトル化したものである。

- **PI の計算**：$PI_{l,l+1}(x)$は層 l および層$l+1$ の派生モデルの分類出力に基づいて、x が良性である（敵対的でない）確率は、次の最適化問題を解いて推測する。

$$PI_{l,l+1}(x) = \min\left[\sum_{x \in X_b} J(concat(D_l(x), D_{l+1}(x)) \cdots w^T - 1)\right]$$

ここで、層 l の派生モデルD_l は、層 l の後に softmax 層を追加してのように定義される。

$$D_l = \text{softmax} \circ f_l \circ f_{l-1} \circ \cdots \circ f_1$$

- **PI と VI の直和空間**：上記の最適化により求めた VI と PI から、モデル M の学習データのバッチごとに以下の直和空間（ベクトル）を作成する。

$$VI_1 \oplus PI_{1,2} \oplus VI_2 \oplus PI_{2,3} \cdots VI_B \oplus PI_{B-1,B} \oplus VI_B$$

このベクトルは $L \times 3$ 次元（L は M の層数）であり、これは個数B のベクトル空間（直和空間）になる。NIC ではこの空間に対して OSVM を行う。

7.5 計算機実験

敵対的データ検出技術（NIC)の効果を確認するため、下記の実験環境で、論文[45]の実験の追試を行なった。

- ハードウェア環境：産総研 ABCI[46]
- データセット：画像分類の実験には、MNIST[47]、CIFAR-10[48]の 2 つの一般的な画像データセットを用いた。MNIST は手書き数字認識に使用されるグレースケール画像データセットであり、CIFAR-10 はオブジェクト認識に使用されるカラー画像データセットである。なお、NIC に対しては、LFW（顔画像）[49]についても実験を行った。
- 攻撃：敵対的データの生成には、非標的型攻撃（FGSM L^2, L^∞）、標的型攻撃 JSMA、勾配ベースの攻撃（CW L^2）の方法を使用した。FGSM と JSMA の実装には、Cleverhans ライブラリ[50]を使用した。

最初に、①～③の各アプローチに基づく敵対的データ検出手法を評価するため、LID[42]、MagNet[43]、フィーチャスクイージング[44]の公開されている実装コードを用いて、各論文の追試実験を行った。その結果、各論文に報告されている検出率を確認でき、この 3 つの中では、フィーチャスクイージングが最も高い検出率を示していた。

次に、④のアプローチに基づく敵対的データ検出手法を評価するため、7.4 節で実装した NIC のコードを用いて実験を行った。表 7.1～表 7.3 に、各々、MNIST、CIFAR-10、LFW のデータセットに対する敵対的データ検出計算実験の結果示す。ここで、正答率は、7.4 節で説明した分類器（OSVM）に敵対的データを入力し、敵対的データであると判定された割合である。なお、実験に使用した CNN モデルは LeNet5、OSVM の Kernel は RBF（MNIST：$\gamma = 0.1\sim0.27$, CIFAR-10：$\gamma = 0.11\sim0.2$, LFW：$\gamma = 0.005\sim0.90$）である。本実験結果では、論文[45]で報告されていたデータセットや攻撃方法だけでなく、報告されていないデータセット LFW、攻撃方法（FGSM L^∞）についても高い検出性能を確認することができた。

表 7.1　MNIST データセットに対する敵対的データ検出計算実験結果

Data Set	Attack	Invariant	正答率	データ件数	論文[45]正答率
MNIST	FGSM L2	VI	97%	2800	100%
		PI	98%		84%
		NIC	97%		100%
	FGSM L∞	VI	98%	2800	—
		PI	98%		—
		NIC	98%		—
	JSMA	VI	100%	280	83%
		PI	100%		100%
		NIC	100%		100%
	CW2	VI	100%	280	95%
		PI	100%		96%
		NIC	100%		100%
	Trojan	VI	100%	3200	100%
		PI	100%		100%
		NIC	100%		100%

表 7.2 CIFAR-10 データセットに対する敵対的データ検出計算実験結果

Data Set	Attack	Invariant	正答率	データ件数	論文[45]正答率
CIFAR-10	FGSM L2	VI	99%	6400	100%
		PI	99%		52%
		NIC	99%		100%
	FGSM L∞	VI	100%	6400	—
		PI	100%		—
		NIC	100%		—
	JSMA	VI	97%	320	62%
		PI	95%		100%
		NIC	96%		100%
	CW2	VI	98%	320	88%
		PI	95%		89%
		NIC	96%		100%
	Trojan	VI	100%	3200	100%
		PI	100%		100%
		NIC	100%		100%

表 7.3 LFW データセットに対する敵対的データ検出計算実験結果

Data Set	Attack	Invariant	正答率	データ件数	論文[45]正答率
LFW	FGSM L2	VI	98%	28222	—
		PI	98%		—
		NIC	98%		—
	FGSM L∞	VI	100%	2822	—
		PI	100%		—
		NIC	100%		—
	JSMA	VI	100%	280	—
		PI	100%		—
		NIC	100%		—
	CW2	VI	100%	840	—
		PI	100%		—
		NIC	100%		—
	Trojan	VI	100%	3200	—
		PI	100%		—
		NIC	100%		—

8 運用時における AI 品質管理技術

本章では、運用時における AI 品質管理技術として、コンセプトドリフトと呼ばれる時間経過に伴うデータ分布の変化に対し、その分布変化の検知と、変化後の分布に機械学習モデルを適応させる最新技術の調査結果について報告する。

コンセプトドリフトは、運用中の AI システム内で稼働する機械学習モデルの性能低下を引き起こす主な原因の 1 つである。そのため、システムの運用開始時点で充足されていた品質を、運用期間を通じて維持するためには、ドリフトが生じているか否かを継続的に監視することに加え、必要に応じてシステム内の機械学習モデルを最新のデータを用いて再学習することで、ドリフト後の分布にシステムを適応させることが必要である。特に近年の機械学習技術の利用拡大に伴い、今後の AI システムの運用場面では、これまで扱われなかった種類のデータを含め、正解ラベル付けされていない大量のデータを短期間で処理することが求められる。

そこで、2019～2020 年度において、運用中の機械学習モデルの性能維持を目的として、上記のコンセプトドリフトの検知および適応を行う最新技術に関する調査を行った。その結果、これまでに開発されている手法の多くが、検知および適応時に運用中に新たに取得した入力データの正解ラベルを用いる教師あり手法であった。しかしながら、正解ラベルは必ずしも入手できるとは限らず、入手できたとしてもコストがかかる場合が多い。そのため、適用可能性を広げるため、または運用コストを削減するためには、それらの正解ラベルを用いない「教師なし手法」や、少数の正解ラベルのみ限定的に用いる「半教師あり手法」が有望であることがわかった。そこで、その視点から整理し検討した調査結果をサーベイとしてまとめた。各サーベイに関する詳細については、検知手法に関しては機械学習品質マネジメントガイドライン[1] 7.8 節を、適応手法に関しては文献[51]をそれぞれ参照されたい。

9 学習モデル情報の可視化

ブラックボックスである学習モデルの構造や挙動についての分析を支援する手法として、情報可視化技術の導入が進んでいる。このような学習モデル可視化に関し、以下の 2 点を目的として新たな手法の研究を開始した。

- 複数のモデル間の差分・比較結果の可視化
  （人間が解釈・理解しやすい表現を用いた可視化の実装）
- モデルに反映されている作業者（アノテータ、モデル設計者）の感性の可視化
  （品質評価に用いることができる新たな要素の提案）

本章では、まず近年の学習モデル可視化技術の調査結果について解説し、その後、2020 年度に試作した、モデル比較のための可視化ツールの実行例、今後の実装方針について報告する。

9.1 機械学習支援のための可視化手法についての調査

機械学習を対象とした可視化手法の基本的な目的はモデルの解釈可能性の向上であり、近年注目されている XAI（Explainable AI）と大きな関連がある。XAI の定義や評価方法について確定的なものは存在しないが、XAI の分類を試みる論文等は多数発表されているため、これらに沿って可視化の目的や手法を考案することができる。[52]では解釈可能性を上げるためのアプローチについて以下の 4 通りに分類している。この中では特に(2)(4)について可視化が貢献できる余地が高く、研究事例も多いとみられる。

(1) 大局的な説明（簡易なモデルによる複雑なモデル構造の近似）
(2) 局所的な説明（モデルの出力結果に関する判断根拠の説明）
(3) 説明可能なモデルの設計（設計段階での可読性の高いモデルの作成）
(4) 深層学習モデルの説明（画像データ内でモデルが認識している部分のハイライトなど）

このような機械学習の可視化手法は継続的に研究されており、用途や対象事例が多様であるためサーベイ論文も増加している。例えば[53]では、深層学習可視化手法について 5W1H の要素に沿って解説、分類している。深層学習可視化分野の全体的な方向性や課題についても多数提示しており、特に品質評価のための可視化手法の開発を目指す本研究と関わりの深いものとして「モデル評価のためのインタラクションの改良」「モデルに対する人の積極的な関与による解釈可能性の向上」などが挙げられる。

機械学習可視化の研究が進み、実社会でも利用される機会が増加するにつれて、1 つの可視化画面で複合的な分析が行われる傾向が強まっている。従来は単体のモデルの詳細な分析や、データ（[54]）かモデル構造（[55]）のいずれかに特化した可視化などが多かった。しかし近年はデータとモデル構造の複合的な可視化手法や、複数のモデルを比較することを目的とした可視化手法についても研究が進んでいる。機械学習モデルを構成する要素は

膨大なものとなるため、例えばモデル単体の可視化結果をモデル個数分作成し、並べて比較するには大きな手間がかかる。また、比較しようとするモデル間の構造や精度の差が小さく、モデルの特徴などを見落とす可能性も考えられる。そのため、差分を強調する表現を用い、限られた画面内で効率的に差分を発見できるようにする可視化手法の必要性が高い。（例えば[56]では、10 種類以上のモデルについてデータの入力から出力までのパイプライン、ハイパパラメータの値などを一画面で比較できる。）

ここまでで、モデル自体の性質や精度に関する可視化手法についての傾向と事例を紹介した。これと並行して、モデル作成に関わる作業者（アノテータ、設計者）とモデルとの関わり方についても調査を進めた。画像認識などの分野では人の認識能力を超える精度を持つモデルが開発される一方で、「モデルの精度を向上させるためには積極的な人の介入が望ましい」という提言は分野を問わず根強い。AI と人との関係性やモデル作成過程での効果的な介入方法に関し、以下のような項目について議論した論文が多く存在する。

- 学習過程での、モデルの精度を向上させるための操作（調整や評価）の導入方法
- 操作性が良く、作業者のモチベーションを維持しやすいインタフェースの設計方法
- 認知科学、心理学のような関連分野との連携

一例として[57]では、モデルの評価・改善のためのフィードバックを課された作業者の心理状態について検証しており、「モデルに正しい処理手順を直接指示できることを好む」「自身の行動によってモデルの精度が改善されていることが可視化されると、モチベーションの向上やより積極的なフィードバックを見込める」などの傾向を紹介している。このような作業者自身の情報や、各作業者がモデルに与えた影響についての可視化事例は少ないという印象だが、以下のように品質保証の根拠として採用できると考えられる。

- 分析対象の事例、あるいは機械学習の専門家がモデル作成に参加していた場合、彼らの知識が十分にモデルの挙動に反映されていることを示す
- どの作業者の行動がモデルに強く反映されているのかを示し、調整するべき要素（データ、パラメータなど）を特定するための手がかりとする

以上のような調査結果から、本研究では学習モデル可視化手法のうち特に「複数モデルの比較可視化」「作業者の感性に関する可視化」を重要視し、これらの性質を併せ持つような可視化ツールの設計を進めた。

9.2 モデルの差分可視化ツールの試作

上述した調査結果をもとに複数モデルの比較に特化した可視化ツールの検討、試作を進めた。本ツールの主なユーザをモデルの設計者と想定し、可視化に不慣れなユーザが含まれる可能性を考慮して基本的な可視化手法（折れ線グラフ、棒グラフなど）を組み合わせ、それらの連携（関連部分のハイライトなど）を積極的に行うという方針で実装した。図 9.1 は試作した可視化ビューの一覧であり、2 つの簡易なモデルについて MNIST の学習・出力結果を可視化したものである。

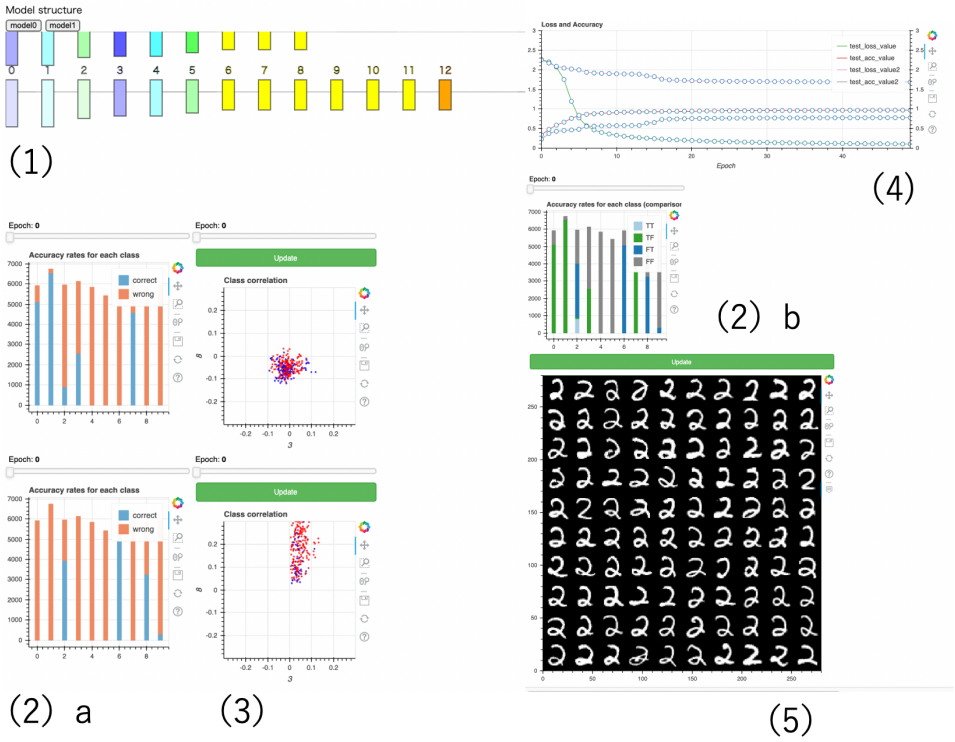


図 9.1　実装した可視化ビューの一覧

JupyterLab 上で主に機械学習ライブラリ PyTorch、可視化ライブラリ Bokeh などを用いて以下のように作成し、2 つのモデルの特徴を比較できるようにした。

(1) 各モデル構造のネットワーク
(2) クラスごとの出力結果の棒グラフ
　a) モデルごとの可視化
　b) 2 つのモデル間の差分の可視化
(3) モデルごとの、選択した 2 クラス間の出力結果相関の散布図
(4) 精度の折れ線グラフ
(5) 特に高い（低い）確信度で分類されたデータのサムネイル一覧

図 9.2 は、2 つのモデルについて MNIST に対する出力を分類した結果の例である。横軸は 0〜9 のクラス、縦軸はデータ数を表す。それぞれの棒の色分けは、2 つのモデルについての正解（T）・不正解（F）の組み合わせを表したものである。TF（FT）は「モデル 1（2）のみ正しく分類できたデータ」を意味する。学習開始直後（図 9.2 左）は、モデル 1 がクラス 0、1、7、モデル 2 がクラス 2、6、8 での正解率が高く、それぞれのモデルが異なる強みを持っていたことがわかる。学習が進んだ段階（図 9.2 右）ではどちらのモデルも多くのクラスで正解率が高くなっている。特にモデル 1 はモデル 2 が不得意とするクラス 3、4、5、7 を含めて正答率が高く、この段階ではモデル 1 の学習の方がモデル 2 よりも進んでいることがわかる。

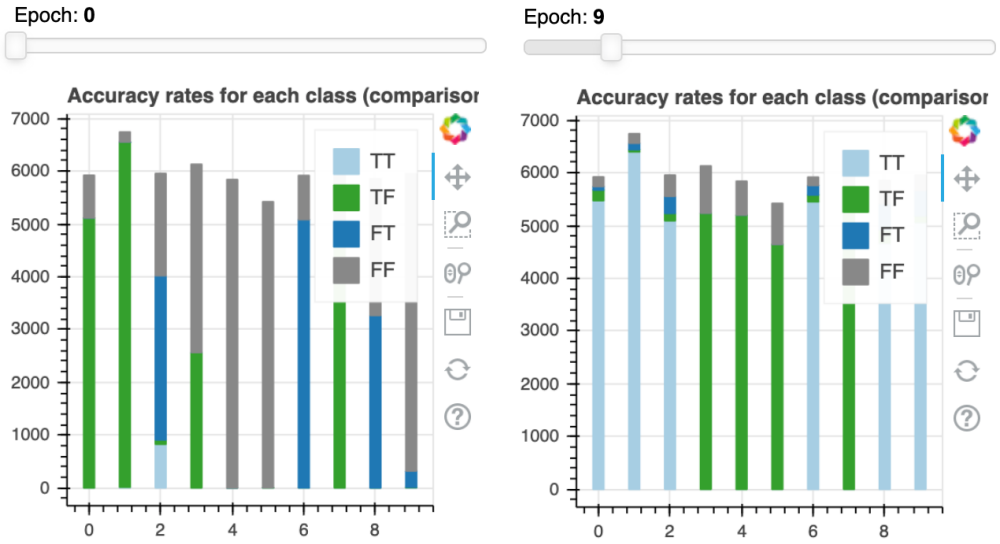


図 9.2　2 つのモデルの出力結果比較の例

9.3　今後の方針

図 9.1 のビューに加えて、作業者の感性可視化のためのビューを追加する。具体的には (1) 学習過程での作業内容と精度変化の時系列可視化　(2) モデルに対するアノテータとモデル設計者の感性反映度の可視化、の実装を進める。これらの可視化で必要となる作業者の情報について、当面は学習や調整作業のログから得られる情報を元に可視化を試み、ユーザの負担が大きくなるインタラクティブな操作での入力が過剰にならないようにする。また、追加予定の 2 種を含めた各可視化ビューについて、連携機能の強化や重要な差分に注目させる表現の実装を進めたい。

10 参考文献リスト

（1 章の参考文献）

[1] 大岩 寛（産総研）他, 機械学習品質マネジメントガイドライン,
第 1 版：CPSEC-TR-2020001, 2020 年 6 月, 第 2 版：CPSEC-TR-2021001, 2021 年 6 月.
https://www.cpsec.aist.go.jp/achievements/aiqm/

[2] 中島 震（情報研）, 敵対的なセマンティック・ノイズの実行時検知, 情報処理学会・ソフトウェア工学研究会, 2020 年 7 月.

[3] 中島 震（情報研）, 統計的な部分オラクルによるテスティング方法, 日本ソフトウェア科学会大会, 2020 年 9 月.

[4] 中島 震（情報研）, ニューラルネットワーク・ソフトウェアの頑健性検査, 情報処理学会・ソフトウェア工学研究会, 2020 年 11 月.

[5] Shin Nakajima (NII), Software Testing with Statistical Partial Oracles, 10th SOFL+MSVL, 2021 年 3 月.

[6] 中島 震（情報研）, 訓練済み機械学習モデル歪みの定量指標, 電子情報通信学会・ソフトウェアサイエンス研究会, 2021 年 3 月.

[7] 高瀬朝海（産総研）, 星野貴行（慶大）, 畳み込みニューラルネットワークの特徴マップへの Data Augmentation 適用, 第 23 回画像の認識・理解シンポジウム,2020 年 8 月.

[8] Tomoumi Takase（AIST）, Dynamic batch size tuning based on stopping criterion for neural network training, Neurocomputing, Volume 429, pp.1-11, 2021 年 3 月.

（3 章の参考文献）

[9] Gontijo-Lopes, R., Smullin, S. J., Cubuk, E. D., and Dyer, E., Affinity and Diversity: Quantifying Mechanisms of Data Augmentation. arXiv preprint arXiv:2002.08973, 2020.

[10] Cubuk, E. D., Zoph, B., Shlens, J., and Le, Q., RandAugment: Practical Automated Data Augmentation with a Reduced Search Space. In Neural Information Processing Systems, 33, 2020.

[11] Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D., Mixup: Beyond Empirical Risk Minimization. In International Conference on Learning Representations, 2018.

[12] Verma, V., Lamb, A., Beckham, C., Najafi, A., Mitliagkas, I., Lopez-Paz, D., and Bengio, Y., Manifold Mixup: Better Representations by Interpolating Hidden States. In International Conference on Machine Learning, pp. 6438–6447, PMLR, 2019.

[13] Kim, J-H., Choo, W., and Song, H. O., Puzzle mix: Exploiting saliency and local statistics for optimal mixup. In International Conference on Machine Learning, 2020.

[14] Beckham, C., Honari, S., Verma, V., Lamb, A., Ghadiri, F., Hjelm, R. D., Bengio, Y., and Pal, C. On adversarial mixup resynthesis. In Neural Information Processing Systems, 2019.

（4 章の参考文献）

[15] 中島 震，ソフトウェア工学から学ぶ機械学習の品質問題，丸善出版 2020.
[16] Pei, K., et al., DeepXplore: Automated Whitebox Testing of Deep Learning Systems, In Proc. 26th SOSP, 2017, pp.1-18.
[17] Nakajima, S., Distortion and Faults in Machine Learning Software, In Post-Proc. 9th SOFL+MSVL, 2020, pp.29-41.
[18] Ma, L., et al., DeepGauge: Multi-Granularity Testing Criteria for Deep Learning Systems, In Proc. ASE, 2018, pp.120-131.
[19] Tian, Y., et al., DeepTest: Automated Testing of Deep-Neural-Network-driven Autonomous Cars, In Proc. 40th ICSE, 2018, pp.303-314.
[20] Zhang, M., et al., DeepRoad: GAN-Based Metamorphic Testing and Input Validation Framework for Autonomous Driving Systems, In Proc. ASE, 2018, pp.132-142.
[21] Zhang, P, et al., CAGFuzz: Coverage-Guided Adversarial Generative Fussing Testing of Deep Learning Systems, arXiv:1911.07931, 2019.
[22] Harel-Canada, F., et al., Is Neuron Coverage a Meaningful Measure for Testing Deep Neural Networks? In ESEC/FSE, 2020, pp.851-862.
[23] Kim, J. et al., Guiding Deep Learning System Testing Using Surprise Adequacy, In Proc. 41st ICSE, 2019, pp.1039-1049.

（5 章の参考文献）

[24] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus, Intriguing properties of neural networks, The International Conference on Learning Representations (ICLR 2014), pp.1-10, 2014. https://arxiv.org/abs/1312.6199
[25] Guy Katz, Clark Barrett, David Dill, Kyle Julian, and Mykel Kochenderfer, Reluplex: An Efficient SMT Solver for Verifying Deep Neural Networks, International Conference on Computer-Aided Verification (CAV), 2017. https://arxiv.org/abs/1702.01135
[26] Vincent Tjeng, Kai Xiao, and Russ Tedrake, Evaluating robustness of neural networks with mixed integer programming, International Conference on Learning Representations (ICLR), 2019. https://arxiv.org/abs/1711.07356
[27] Tsui-Wei Weng, Huan Zhang, Hongge Chen, Zhao Song, Cho-Jui Hsieh, Duane Boning, Inderjit S. Dhillon, and Luca Daniel, Towards Fast Computation of Certified Robustness for ReLU Networks, International Conference on Machine Learning, PMLR 80, pp.5276-5285, 2018. https://arxiv.org/abs/1804.09699
[28] Akhilan Boopathy, Tsui-Wei Weng, Pin-Yu Chen, Sijia Liu, and Luca Daniel, CNN-Cert: An Efficient Framework for Certifying Robustness of Convolutional Neural Networks, The Thirty-Third AAAI Conference on Artificial Intelligence (AAAI 2019), pp.3240-3247, 2019. https://arxiv.org/abs/1811.12395
[29] Tsui-Wei Weng, Pin-Yu Chen, Lam Nguyen, Mark Squillante, Akhilan Boopathy, Ivan

Oseledets, and Luca Daniel, PROVEN: Verifying Robustness of Neural Networks with a Probabilistic Approach, International Conference on Machine Learning (ICML 2019), PMLR vol. 97, pp.6727-6736, 2019. http://proceedings.mlr.press/v97/weng19a.html

[30] Nicholas Carlini and David Wagner, Towards Evaluating the Robustness of Neural Networks, IEEE Symposium on Security and Privacy (SP), pp.39-57, 2017. https://arxiv.org/abs/1608.04644

[31] Tsui-Wei Weng, Huan Zhang, Pin-Yu Chen, Jinfeng Yi, Dong Su, Yupeng Gao, Cho-Jui Hsieh, and Luca Daniel, Evaluating the Robustness of Neural Networks: An Extreme Value Theory Approach, International Conference on Learning Representations (ICLR 2018), 2018. https://arxiv.org/abs/1801.10578

[32] Eric Wong and J. Zico Kolter, Provable defenses against adversarial examples via the convex outer adversarial polytope, International Conference on Machine Learning (ICML 2018), PMLR vol. 80, pp.5283-5292, 2018. https://arxiv.org/abs/1711.00851

[33] Mathias Lecuyer, Vaggelis Atlidakis, Roxana Geambasu, Daniel Hsu, and Suman Jana, Certified Robustness to Adversarial Examples with Differential Privacy, The IEEE Symposium on Security and Privacy (SP), 2019. https://arxiv.org/abs/1802.03471

[34] Jeremy M Cohen, Elan Rosenfeld, and J. Zico Kolter, Certified Adversarial Robustness via Randomized Smoothing, The 36th International Conference on Machine Learning (ICML 2019), 2019. https://arxiv.org/abs/1902.02918

[35] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu, Towards Deep Learning Models Resistant to Adversarial Attacks, The Sixth International Conference on Learning Representations (ICLR 2018), 2018. https://arxiv.org/abs/1706.06083

（6 章の参考文献）

[36] Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio, Fantastic Generalization Measures and Where to Find Them, International Conference on Learning Representations (ICLR 2020). https://arxiv.org/abs/1912.02178

[37] Peter L. Bartlett, Nick Harvey, Christopher Liaw, and Abbas Mehrabian, Nearly-tight VC-dimension and Pseudodimension Bounds for Piecewise Linear Neural Networks, Journal of Machine Learning Research, Vol.20, No.63, pp.1－17, 2019. https://jmlr.org/papers/v20/17-612.html

[38] Konstantinos Pitas, Mike Davies, and Pierre Vandergheynst, PAC-Bayesian Margin Bounds for Convolutional Neural Networks, arXiv:1801.00171, 2018. https://arxiv.org/abs/1801.00171

[39] Konstantinos Pitas, Dissecting Non-Vacuous Generalization Bounds based on the Mean-Field Approximation, ICML 2020. arXiv:1909.03009, 2020. https://arxiv.org/abs/1909.03009

[40] Wenda Zhou, Victor Veitch, Morgane Austern, Ryan P. Adams, and Peter Orbanz, Non-vacuous Generalization Bounds at the ImageNet Scale: a PAC-Bayesian Compression

Approach, ICLR 2019. https://arxiv.org/abs/1804.05862
[41] Guillermo Valle-Pérez and Ard A. Louis, Generalization bounds for deep learning, arXiv:2012.04115v2, 2020. https://arxiv.org/abs/2012.04115

（7 章の参考文献）

[42] X. Ma, Characterizing adversarial subspaces using Local Intrinsic Dimensionality, 2018.
[43] D. Meng , Magnet: a two-pronged defense against adversarial examples, in Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security. ACM, 2017.
[44] W. Xu, Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks, in Proceedings of the 2018 Network and Distributed Systems Security Symposium (NDSS), 2018.
[45] Shiqing Ma, NIC: Detecting Adversarial Samples with Neural Network Invariant Checking, Network and Distributed Systems Security Symposium (NDSS), NDSS 2019.
[46] 産業技術総合研究所, AI Bridging Cloud Infrastructure, https://abci.ai/ja/
[47] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proceedings of the IEEE, vol. 86, no. 11, pp.2278–2324, 1998. [Online]. Available: http://yann.lecun.com/exdb/mnist/
[48] A. Krizhevsky and G. Hinton, Learning multiple layers of features from tiny images, 2009.
[49] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, Labeled faces in the wild: A database for studying face recognition in unconstrained environments, University of Massachusetts, Amherst, Tech. Rep. 07-49, October 2007.
[50] Nicolas Papernot, Ian Goodfellow, Ryan Sheatsley, Reuben Feinman, and Patrick McDaniel. cleverhans v1.0.0: an adversarial machine learning library. arXiv preprint arXiv:1610.00768, 2016.

（8 章の参考文献）

[51] 大川 佳寛, 小林 健一, ラベルなし運用データに対するコンセプトドリフト適応技術に関するサーベイ, 第 35 回 人工知能学会全国大会, 2021 年 6 月.

（9 章の参考文献）

[52] 原 聡, 私のブックマーク「機械学習における解釈性」, 人工知能, vol. 33, no. 3, pp. 366-369, 2018.
[53] Fred Hohman, Minsuk Kahng, Robert Pienta, Duen Horng Chau, Visual Analytics in Deep Learning: An Interrogative Survey for the Next Frontiers, IEEE Transactions on Visualization and Computer Graphics, vol. 25, no. 8, pp. 2674-2693, 2018.
[54] Bilal Alsallakh, Amin Jourabloo, Mao Ye, Xiaoming Liu, Liu Ren, Do Convolutional Neural Networks Learn Class Hierarchy?, IEEE Transactions on Visualization and Computer Graphics, vol. 24, no. 1, pp. 152-162, 2018.

[55] Mengchen Liu, Jiaxin Shi, Kelei Cao, Jun Zhu, Shixia Liu, Analyzing the Training Processes of Deep Generative Models, IEEE Transactions on Visualization and Computer Graphics, vol.24, no.1, pp.77-87, 2018.

[56] Jorge Piazentin Ono, Sonia Castelo, Roque Lopez, Enrico Bertini, Juliana Freire, Claudio Silva, PipelineProfiler: A Visual Analytics Tool for the Exploration of AutoML Pipelines, IEEE Transactions on Visualization and Computer Graphics, vol.27, no.2, pp.390-400, 2021.

[57] Saleema Amershi, Maya Cakmak, W. Bradley Knox, Todd Kulesza, Power to the People: The Role of Humans in Interactive Machine Learning. AI Magazine, vol.35, no.4, pp.105-120, 2014.