データセットの分布の違いを表現する指標の検討と予測結果の関係性分析

東 結香[1]　津田 侑[2]

概要：教師あり機械学習では学習時と評価時で対象データが同じ分布である前提で，研究や実装が進められている．しかし，実社会においてはその前提に合致しないことがある．例えばマルウェアを対象とした場合 Dataset shift が発生することが知られている．そのような環境下では精度の高いモデルの構築を可能とするとされるアルゴリズムであっても，想定する性能を発揮できるとは限らない．こういった課題を解決するために，対象とするデータの分布が異なる前提での評価の重要性が高まると同時に異なる分布を持つデータセットを表現する指標が必要となる．本論文では，分布の比較を行う検定を用い，データセットの分布の違いを表現する指標の検討を行う．そして，その指標を用い分布の異なるデータセットを用いる評価のメリットについて，EMBER データセットによるユースケースを提示する．

キーワード：データセット，マルウェア，機械学習

A Study of Indices for the Difference in Distribution of Datasets and Relationship Analysis of Prediction

Yuka Higashi[1]　Yu Tsuda[2]

Abstract: In supervised machine learning, researchers conduct machine learning studies based on the assumption that the target data has the same distribution between learning and evaluation. However, in the real world, it may not meet the premise. For example, Dataset Shift is known to occur in the case of malware datasets. In such the environment, even a state-of-the-art algorithm does not always exhibit the expected performance. In order to solve these problems, the importance of evaluation on the assumption that the distribution of the target data is different is increased, and at the same time, an index that expresses a dataset having a different distribution is required. In this paper, we use statistical testing to compare distributions and validate an index that represents the difference in the distribution of datasets. And we show the merit of the evaluation with datasets of different distribution by using EMBER dataset.

Keywords: Dataset, Malware, Machine learning

1. はじめに

昨今，サイバー攻撃による脅威はとどまるところを知らず，日々新たなマルウェアや攻撃パケットが生成されている．それに伴いサイバー攻撃に関連するデータが蓄積され，機械学習を用い蓄積したデータを有効活用しマルウェア検出やネットワーク上の攻撃検知といった様々な課題解決が試みられている．

研究分野として成熟する一方で，論文では非常に良い結果が出ているにもかかわらず，必ずしも実環境下で想定している精度とならないという場合が存在する [1]．そういった場合，過剰適合や使用したデータセットに過剰に適合した特徴抽出が行われている可能性がある．もちろん機械学習分野において，汎化性能を向上させるために様々な手法

[1] トレンドマイクロ株式会社
Trend Micro Incorporated
[2] 国立研究開発法人 情報通信研究機構
National Institute of Information and Communications Technology

が研究提案されている．具体的にはクロスバリデーションやパラメータチューニング，特徴エンジニアリングなどである．

しかし，これらの手法はあくまで手元にあるデータセット内での汎化性能向上である．一般に解決したい問題に関連するデータは無限母集団であることが多く，手元にあるデータセットで全ての事象を網羅しているという状況は稀だと考えられる．そのため，研究と実環境下での使用時の性能にギャップが生じてしまう．

そのギャップを埋めるために，我々は特徴抽出手法や学習アルゴリズムを 2 つの観点から評価する必要がある考える．1 つは特徴抽出やアルゴリズムが対象とする事象に対して適切であるかという観点である．同一対象であるが “異なる” データセット A，B を用意し，A，B それぞれを学習データ・テストデータに分割し評価を行う．これにより，解決対象の事象に対して特徴抽出やアルゴリズムが有効であることが言える．次に，対象とする事象が変化した場合やドメインが異なる場合に適切であるかという観点である．前者と同様に異なるデータセット A，B を用意し，A を学習データ，B をテストデータとして使用し評価を行う．これにより，ドメインが異なっていたり変化した場合にも提案の特徴抽出や学習アルゴリズムが有効であるといえる．後者は転移学習における元ドメイン（学習データ）と目標ドメイン（テストデータ）の分布が異なる場合のシナリオである．

セキュリティ分野，特にマルウェアについては Concept Drift の発生が確認されており，提案手法が上記の前者・後者のどちらのシナリオで評価されているかは実環境下での利用を考える際非常に重要となる [2].

上述の評価を行うためには最低でも異なる 2 種類以上のデータセットが必要となる．大量のデータを集めたところで，性質の異なる複数のデータセットを構築することは難しい．旧来より，標本の分布を表す値として記述統計量が用いられている．しかし，これらの値で使用するデータセットの性質を表現することは困難である．Concept Drift の発生を想定し収集時期をずらしたデータセットにより，異なるデータセットを用いた評価を試みている研究も存在する [3]．しかし，収集日や Firstseen が不明なデータの存在やどの程度収集期間をずらすべきかについて明確な答えは現状存在しない．ゆえに，2 つのデータセットが異なることを表現する新たな指標が必要である．

そこで，本論文ではデータセット違いを表す指標について検討する．具体的には，性質が異なるデータセットを分布が異なるデータセットと定義し，分布を調べる仮説検定を用い 2 つのデータセットの違いを数値として表現することが可能であるかを検証する．本論文の構成は次のとおりである．2 章にて機械学習におけるデータセットを取り巻く現状及び関連研究について述べる．3 章では仮説検定を用いた指標の提案と検証方法，4 章にて EMBER データセット [4] の概要及び検証実験について述べ，5 章にて考察を与える．

本論文の貢献は以下のとおりである．

(1) データセットの分布の差を表す指標の提案

(2) ユースケースを用いた分布の異なるデータセットを用いるメリットの提示

2. 背景と関連研究

なぜ機械学習アルゴリズムにおいて異なるデータセットが必要となるかについて関連研究を交え説明する．まずデータセットを取り巻く現状について 2.1 節で述べる．2.2 節でデータセットの分布に関する先行研究として Dataset Shift について整理する．

2.1 データセットを取り巻く現状

本節では，教師あり機械学習（以下，本稿では単純に「機械学習」と呼ぶ）を用いたセキュリティ研究におけるデータセットに起因する課題として次の 2 点を取り上げる．

(1) 実社会に対する有効性の低さ

(2) ベンチマークデータセット不足による再現性の低さ

ここでの広義の “再現性” は Reproducibility と Replication を含むものとする．Reproducibility とは同一のデータ・手法を利用し検証することを示す．GitHub 等でデータ・実験に使ったソースコードを公開し検証する場合が当てはまる．Repication とは独立したデータセットを用い，異なる研究者が検証を行うことを指す.

(1) 実社会に対する有効性の低さ

一般的に学習時とテスト時で同じ分布のデータが利用されることが前提となっている． しかし実社会では，機械学習を用いたシステムを構築した時期と運用を開始した時期とでデータに乖離があり，想定している本来の性能を発揮できない場合がある．このように，入力と出力のデータが学習時と運用時で異なることは Dataset Shift や Concept Drift とよばれる．コンピュータセキュリティ分野では攻撃手法の移り変わりもあり Dataset Shift や Concept Drift が発生しやすく，実環境下に近い環境で想定した性能が発揮できない事例もある [1]．実際，Pendlebury らは Android アプリケーションの分類問題において高い性能を発揮するとされる手法を実環境下に近い環境・設定で試したところ想定していた精度を出すことが難しいということを示した．つまり，現状のデータセットでの評価では実社会において高い有効性を出すことは困難である．

(2) ベンチマークデータセット不足による再現性の低さ

コンピュータセキュリティ分野では，画像認識分野のような機械学習の利用が盛んな分野と違い，ベンチマークデータセットが少ない．具体例として，マル

ウェア分類器の用途，特に Winodws マルウェアについては Microsoft [5] や EMBER，Android マルウェアについては Avast [6] や AndroZoo [7,8] などがある．AndroZoo のように正式な手順を踏むことによりバイナリを入手することができるものもあるが，多くのセキュリティ用のデータセットには何らかの加工が施されているため，研究者自身でデータセットを作成する場合が多い [9]．しかし，研究用に作成されたデータセットが公開されることも非常にまれで他者による実験の再現が難しい [10,11]．

これらの課題を解消するために，実社会に即したデータセットの作成及びその公開が必要である．これにより，先述したように提案アルゴリズムが高い有効性を示せる状況を明らかにでき，他者の Replication による再現性の確保にもつながる．しかし，データセットの作成および公開の実現は非常に困難である．この理由には，セキュリティ分野においては Dataset Shift が発生しやすくデータセットのメンテナンス負荷が高くなることや，データセットにはマルウェアや攻撃パケットを含むことになり悪用される可能性があることが挙げられる．

異なる複数のデータセットを用いて評価を行えば，どのような状況下で提案アルゴリズムが有効であるか示すことができ，最適な運用の検討が可能となる．また，異なる研究者が行うという要件が満たせないので完全ではないが，Replication を行うとみなすことができる．そのため，再現性の向上にも寄与する．以上より，異なるデータセット表す指標が確立することは，提案アルゴリズムの有効性と再現性の向上に貢献できると考える．

2.2 Dataset Shift

前節で述べた通り，Dataset Shift はデータセットの分布を体系化した分野であり，Covariate Shift，Prior Probability Shift，Concept Drift に大別される [12]．本論文では，データの分布について述べるため，本節では先行研究としてそれぞれの概要をまとめる．

2.2.1 Covariate Shift

Covariate Shift とは学習時とテスト時でインプットの分布のみ変化するシフトである [13]．入力を x，出力を y としたときモデル $P(y|x)$ において，$P_{train}(x) \neq P_{test}(x)$ かつ $P_{train}(y|x) = P_{test}(y|x)$ と表すことができる．

機械学習においては，外挿またはサンプリング時に発生する可能性がある．サンプリングでは標本選択バイアスにより，標本で母集合の分布を表現できない場合，対象の問題に対する精度の高いモデルを生成することは難しい．Covariate Shift においては，発生すること自体は問題ではない．むしろ Covariate Shift の発生を考慮し，学習データ分布及びテストデータの分布を用いた重要度重み付けによりモデルの性能を向上させることも可能である．

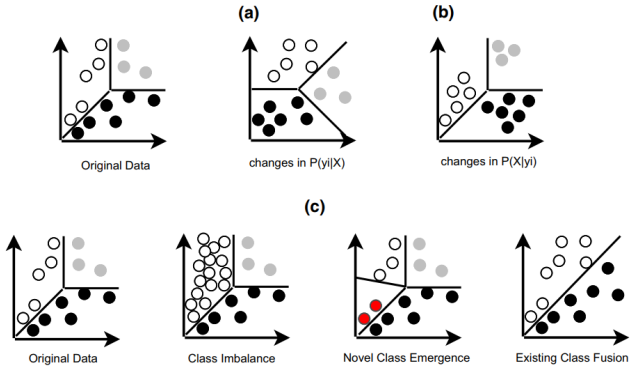

図 1 Concept Drift の種類 [16]

2.2.2 Prior Probability Shift

クラスの事前確率 $P(y)$ が変化するが学習時とテスト時で $P(x|y)$ は変化しない場合を Prior Probability Shift という．例えばベイズによるスパムメールの分類にを考える．学習データではノーマルとスパム 5 : 5 だが，実際の環境（テスト）ではノーマルとスパムが 1 : 9 である場合である．ベイズの定理を正しく適用するためには $P(y)$ を更新しなければならない．しかし，実環境下で正しくラベルの分布を把握するのは不可能である．一方，スパムメールを考えた際，スパムメールに含まれやすい単語，厳密には $P(x|y)$ が大きく変わるわけではない．そのため，分かっている情報から比率を推定する研究がおこなわれている [14]．

2.2.3 Concept Drift

時間の経過によって入力と出力の関係が変化するものを Concept Shift または Concept Drift という．Concept Drift には大きく 2 つのタイプがある．

Real Concept Drift

時間とともに $P(y|x)$ が変わる．つまり分類時の境界面が変化する．

Virtual Concept Drift

時間とともに $P(x)$ が変化し $P(x|y)$ が変わる．同一クラス内の分布は変化するがクラスの決定する境界は変化しない [15]．

最近は Prior Probability Shift により引き起こされる Class Prior Concept Drift というものも提唱されている [16]．

3. 調査手法

前述のとおり，データセットの性質を表す指標は今後のセキュリティにおける機械学習を用いた研究において必要不可欠である．しかし，実環境下では母集団を完全に知ることは困難であるため，絶対値として数値化することは難しい．1 章や 2.1 節で述べた課題を解決するために，分布が異なるかどうかの判断を行う場合に使われる検定手法を用い，2 つデータセットの分布が異なることを示す指標としての利用を提案する．3.1 節では調査手法の概要を述べ，

表 1　パターン

	比率	Benign	Malicious	name
Train	9 : 1	180,000	20,000	mal10
Train	5 : 5	100,000	100,000	mal50
Train	1 : 9	20,000	180,000	mal90
Test	9 : 1	18,000	2,000	tm10
Test	5 : 5	10,000	10,000	tm50
Test	1 : 9	2,000	18,000	tm90

3.2 節以降，具体的な手法について述べる．

3.1 概要

本論文では仮説検定がデータセットの違いを表す手法として使用できるのか，すなわち仮説検定を用い分布の異なるデータセットを数値として表現できるのかを検証する．

仮説検定は，2 標本コルモゴロフ-スミルノフ検定を用いる（以下，本稿では KS 検定と呼ぶ）．KS 検定は 2 つの標本が同一の母集団から発生するというかを調べる検定，つまり 2 つの標本の分布が異なるかどうかを調べるための仮説検定である．

検証のために以下の 3 種の実験を行う．

- ラベルが不均衡なデータを用いた分類器の性能比較
- KS 検定の P 値の評価
- 異なるアルゴリズムでの性能比較

3.2 ラベルが不均衡データを用いた分類器の性能比較

本論文では Endgame 社の公開している EMBER データセット [4] を使用する．Anderson らによると Malicious と Benign が同数含まれている EMBER データセットを LightGBM [17] を用いてデフォルト設定で学習する[*1]と，False Positive Rate が 0.1% 未満の場合 92% 以上の検出率で Malicious なサンプルを検出することが可能であると示した．つまり，EMBER データセットとその特徴量は Malicious と Benign の違いを表現できていると考えられる．このデータセットより極端にラベルが偏ったデータセット生成すると，分布の異なるデータセットを生成することが可能であると考える．ラベルの偏りが異なるデータセットを複数生成し，それらを学習およびテストデータとして学習および評価を行う．学習データとテストデータの各特徴毎に KS 検定を行い，P 値の中央値および棄却された次元数によってラベルの偏りを表すことができるのか，そしてモデルの精度とどのような関係性があるかを調べる．

実験データセットの生成は以下のとおりである．まずテストデータとして EMBER データセットから 2 万個サンプリングする．学習データは母集団からテストサンプルを除外したものから 3 つのパターンでサンプリングする (表 1)．

テストデータセットを抽出後，各比率で学習データセットを 100 回サンプリングし，それぞれの Accuracy，F1 値，KS 検定の P 値を算出しする．EMBER のベンチマークテストと同様，本研究でもデフォルト設定の LightGBM [17] を用いる．

3.3 KS 検定の P 値の評価

機械学習では学習データ数とテストデータ数に差が存在する場合がある．本論文でも 3.2 節では学習データがテストデータの 10 倍のサンプル数であった．KS 検定の定義上，算出される P 値は比較するデータセットの要素数に影響を受ける．比較するサンプル数を変化させることにより，P 値がどのように変化するのかを調べる．学習データが合計 2 万サンプルとなるようにサンプリングを行い，3.2 節の結果のうちテストデータの比率が 5 : 5 の場合と比較を行う．本実験も 3.2 節と同様，テストサンプルを選択後，学習サンプルのサンプリングを 100 回繰り返した．加えて，学習データもテストデータもともに大きい場合の値を確認するために 3.2 節で用いた比率の異なる学習データセット（各 20 万サンプル）の値も比較する．

3.4 異なるアルゴリズムでの性能比較

異なるアルゴリズムでも，同様の関連性が見られるかを調べるためにランダムフォレストを用いた実験を行う．3.2 節及び 3.3 節ではテストサンプルを固定して学習データをサンプリングする実験を行っていたが，本実験ではテストサンプル選択自体 10 回繰り返し，各テストサンプル選択後さらに学習サンプルのサンプリングを 10 回繰り返した．テストサンプルの比率は 5 : 5 とした．ランダムフォレストはデフォルト設定 scikit-learn[*2] version 0.20 を用いた．

4. 実験

本章では，3 章で述べた実験で使用するデータセット及び結果を述べる．

4.1 データセット

本節では使用する EMBER データセットについて説明する．110 万サンプル存在するデータセットであるが，本論文では Benign または Malicious のラベルが付与されている 80 万サンプルを学習データとして使用する (表 2)．また，EMBER データセットとしては学習サンプルとテストサンプルを固定して提供しているが，本論文ではランダムサンプリングにより学習サンプルとして扱うかテストサンプルとして扱うかを決定した．データサンプリングは pandas[*3]を用いランダムサンプリングを行う．データセットは JSON 形式で提供されており，数値データだけでなく

[*1] https://github.com/endgameinc/ember/blob/master/scripts/classify_binaries.py

[*2] https://scikit-learn.org/

[*3] https://pandas.pydata.org/

表 2　EMBER データセットのサンプル数

	Benign	Malicious	Unknown	計
Train	300,000	300,000	300,000	900,000
Test	100,000	100,000	0	200,000
計	400,000	400,000	300,000	1,100,000

表 3　次元数

Category	Dims
Bytehistogram	256
ByteEntropyHistogram	256
StringExtractor	104
GeneralInfo	10
HeaderfileInfo	62
SectionIndo	255
ImportInfo	1280
ExportInfo	128

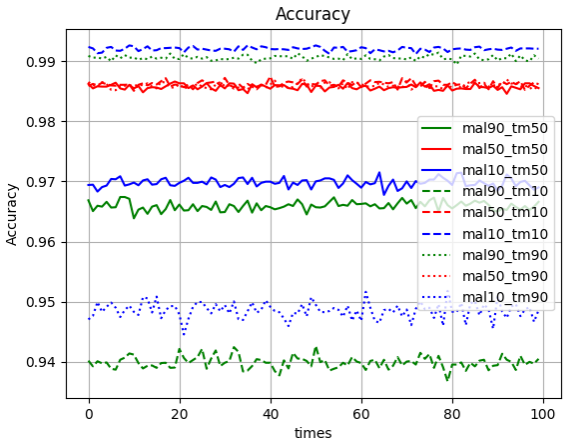

図 2　Accuracy

文字列も含まれる．文字列は scikit-learn の特徴抽出用のモジュール FeatureHasher を用いベクトル化している．ベクトル化した後の特徴は 2,351 次元であり，各データの次元数は表 3 のとおりである．

4.2　実験結果

本節では実験の結果について述べる．凡例の “mal 数字” はサンプリングした学習データにおけるマルウェアの比率を表し，“tm 数字” はテストデータでのマルウェアの比率を表す．“test” は “tm50” と同義である．

4.2.1　ラベルが不均衡なデータを用いた分類器の性能比較

学習データ 3 パターン×テストデータ 3 パターンの合計 9 パターンの Accuracy および F1 値を算出した．また，サンプリングした学習データセットとテストデータセットに対し KS 検定を行い各次元の P 値の中央値および有意水準を 1% とし棄却されたものの次元数を図示した (図 2，図 3，図 4，図 5)．

4.2.2　KS 検定の P 値の評価

4.2.1 節と同様に，Accuracy および各次元の KS 検定の

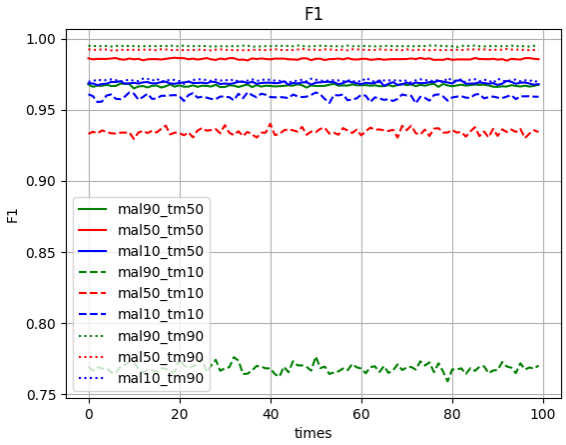

図 3　F1 値

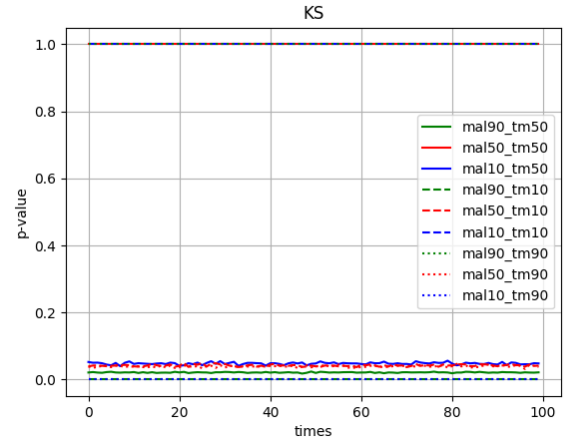

図 4　KS 検定の P 値の中央値

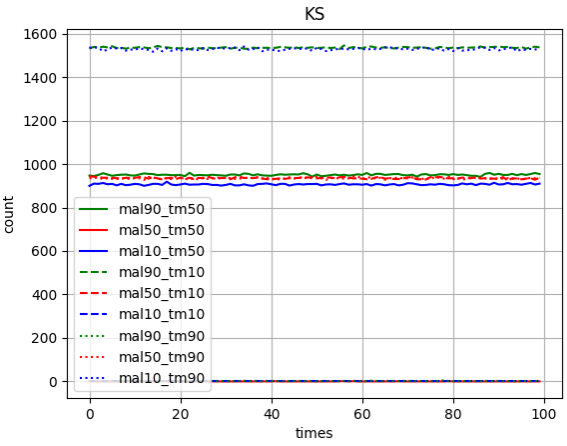

図 5　KS 検定で棄却された次元数

P 値の中央値および棄却次元数をマッピングした (図 6，図 7，図 8)．また，サンプル数と KS 検定の P 値関係をみるために，4.2.1 節の学習データ間での KS 検定の P 値の中央値および棄却次元数も算出した (図 9，図 10)．

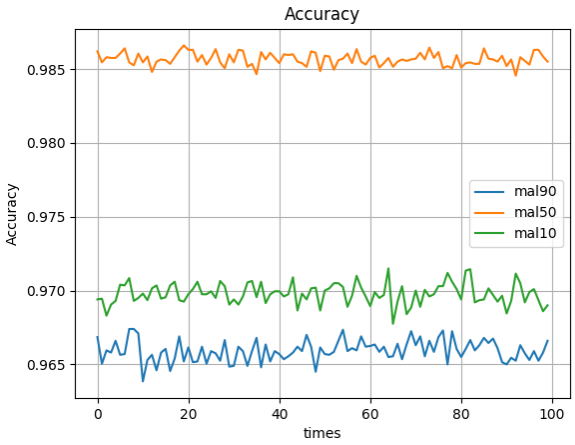

図 6 学習データ 2 万サンプルでの Accuracy

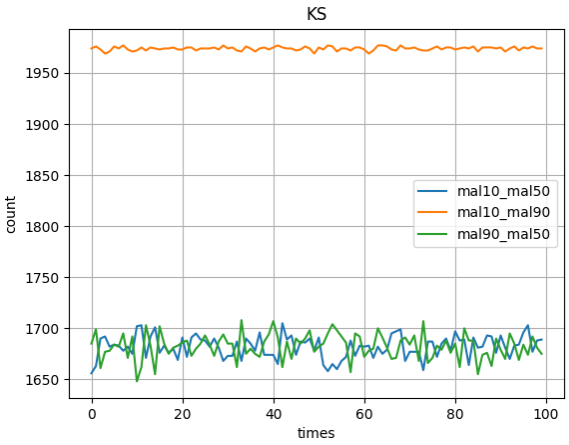

図 9 各 20 万サンプルのデータセット間で棄却された次元数

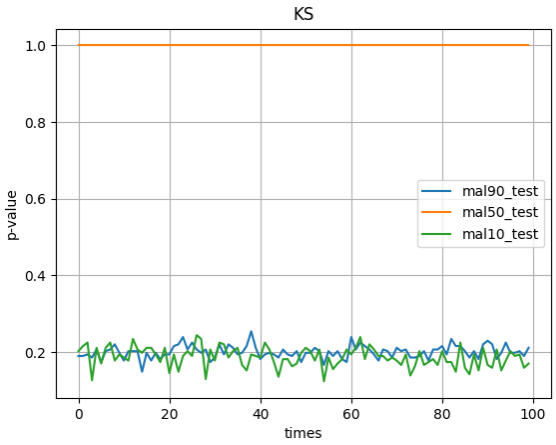

図 7 学習データ 2 万サンプルでの P 値の中央値

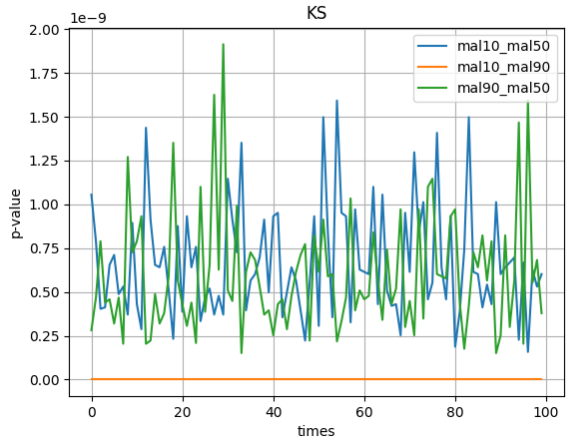

図 10 各 20 万サンプルのデータセット間での P 値の中央値

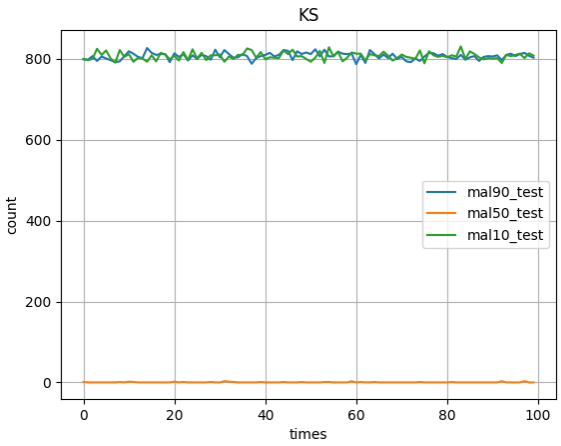

図 8 学習データ 2 万サンプルでの棄却された次元数

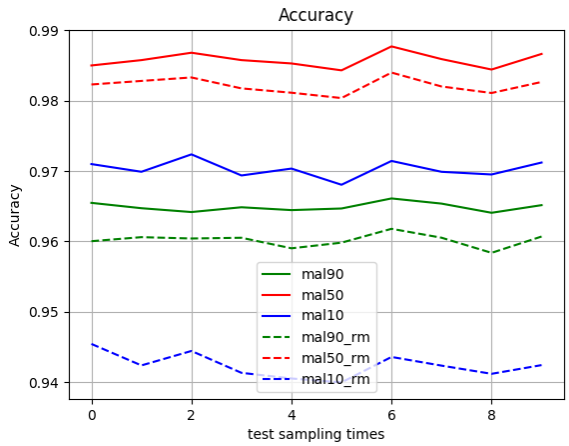

図 11 LightGBM とランダムフォレストの Accuracy の平均

4.3 異なるアルゴリズムでの性能比較

サンプリングしたテストサンプルごとに学習サンプルを 10 回サンプリングしている．可読性向上のため，テストサンプリングごとの Accuracy，F1 値，KS 検定の中央値及び棄却数の平均を示す (図 11，図 12，図 13，図 14)．凡例の“mal 数字” は LightGBM での結果，“rm” が入っているものはランダムフォレストの結果を表す．

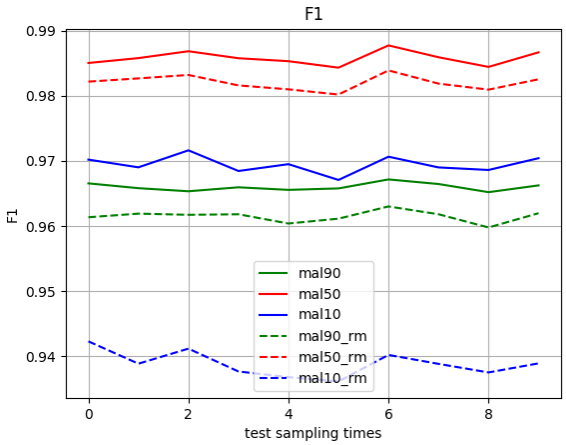

図 12 LightGBM とランダムフォレストの F1 の平均

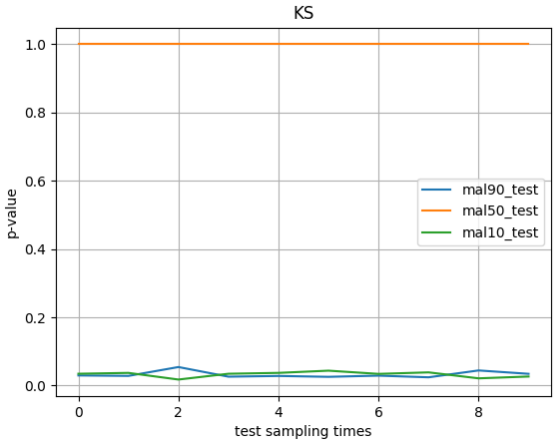

図 13 KS 検定の P 値の中央値の平均

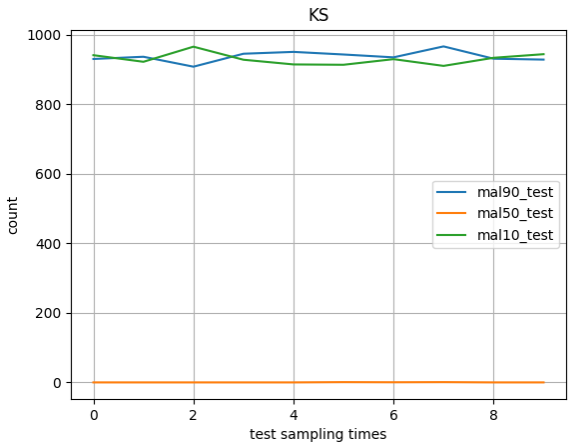

図 14 KS 検定で棄却された次元数の平均

5. 考察

5.1 KS 検定の P 値によるデータセットの表現

図 4 及び図 5 より，学習とテストで比率が同じデータセットは KS 検定の P 値の中央値が 1 に近く，ほとんど棄却されなかった．比率が異なるものは割合によって p 値の中央値及び棄却数に違いが表れた．一方，比較するデータセットのサンプルサイズ比に影響を受けることも明らかとなった．図 7，図 8，図 9，図 10 より，比率が異なるデータセットを比較した場合，学習・テストともに 2 万サンプルの場合，P 値の中央値は 0.2 付近であり，学習テストともに 20 万サンプルの場合は 1e-9 である．また，．テストサンプルを固定して学習データを 100 回ランダムサンプリングした図 4 では常に $mal10\_tm50 > mal90\_tm50$ であったが図 7 及び図 8，図 13 及び図 14 において，サンプリングしたテストデータによっては，$mal90\_tm50$ 及び $mal10\_tm50$ の KS 検定の P 値が逆転しているものがあった．分布の異なるデータセット同士を比較する場合，サンプル数の少ないデータの分布の影響を強く受けるため，mal90 と mal10 のどちらが mal50 に類似しているかという指標での利用はできない．

以上より，KS 検定は棄却できなからといって分布が類似しているとは言えないものの，比率が同じ場合と異なる場合で優位に差が存在するため，分布が異なるかどうかを表現する指標としては利用できる．しかし，異なるもの同士でどちらが近いかという指標としては利用できない可能性が高いこと言える．

5.2 学習結果とデータセットの分布

LightGBM では同比率で学習したモデルが最も高い精度となった．その場合 KS 検定の P 値の中央値は最も高く，棄却された次元は最も少なかった．マルウェアの比率が 90% のものに比べ 10% のものは 0.5% 程度精度が低くなり，KS 検定の P 値の中央値は低く，棄却された次元は多かった．また，テストサンプルと学習サンプルの比率が同じ場合，Accuracy が高くなるが，比率が異なる場合でも mal50 で学習した場合安定した精度が出ている．傾向としては Malicious の比率が高いより低いほうが高い精度になっていた．F1 値についても mal10 のデータセットでテストした場合が全てのモデルにおいて最も低かった．LightGBM では Malicious より Benign のほうが特徴反映しづらい可能性が考えられたため，Benign ファイルを増やして，mal40，mal30 の比率でも実験を行った (図 15) が，精度の向上は見られなかった．つまり本研究で使用した特徴，データセットに対し LightGBM で学習を行う場合，実環境下での Malicious と Benign の比率が不明である場合は mal50 で学習させることがバランスが良いといえる．

また，図 11 や図 12 は学習データを 10 回サンプリングした LightGBM とランダムフォレストで学習した結果の平均値だが，LightGBM とランダムフォレストですべての試行において mal90 と mal10 を学習した際の Accuracy 及び F1 値が逆転していた．今回行った実験においてランダムフォレストより LightGBM の性能が高かったが，実環境

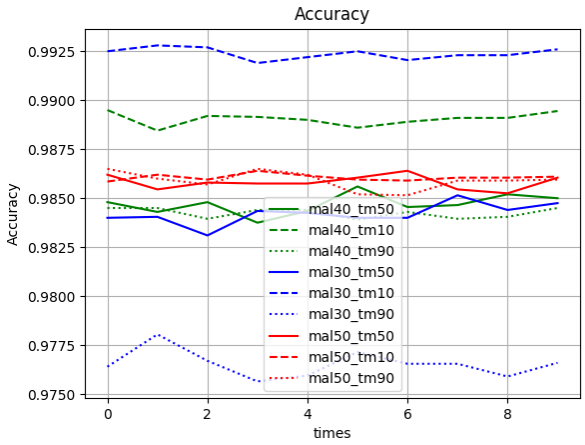

図 15 mal50-mal30 での Accuracy

下で発生する可能性がある分布の異なるデータに対しても LightGBM がランダムフォレストより優れている可能性が高いといえる．

このように，異なるデータセットを用いた評価を行うことにより，そのアルゴリズムのの特性や実環境下での有効性を示すことに役立つといえる．

6. おわりに

仮説検定の 1 つである KS 検定の P 値を用い，複数次元を持つ機械学習で利用されるデータセットの違いを表現できるかについて検証を行った．分布が異なるかどうかについては表現可能であるが，差を数値で表現することは難しいことが明らかになった．また，分布の異なる複数のデータセットを用いることは，学習アルゴリズムの有効性を示す評価として活用できることが分かった．

一方，今回は限られたデータやアルゴリズムでの検証にとどまっている．今後，同様の実験をほかのデータセット及びアルゴリズムでも行い，さまざまなデータセットで利用可能であることを示す必要がある．また，今回は KS 検定を用いたが統計学には様々な指標が存在している．KS 検定で調べることができる分布が異なるかどうかという点で利用はできるが，異なるもの同時の差を表現することは検定では難しいため，引き続きデータセットを表現するのに適切な指標を検討する．

参考文献

[1] Feargus Pendlebury, Fabio Pierazzi, Roberto Jordaney, Johannes Kinder, Lorenzo Cavallaro, and College London. TESSERACT : Eliminating Experimental Bias in Malware Classification across Space and Time. In *Proceedings of 28th USENIX Security Symposium*, 2019.

[2] Richard Harang and Felipe Ducau. Measuring the speed of the Red Queen's Race Who we are. In *BlackHat USA 2018*, 2018.

[3] Nedim Srndic and Pavel Laskov. Detection of Malicious PDF Files Based on Hierarchical Document Structure. In *The Network and Distributed System Security Symposium (NDSS) 2013*, 2013.

[4] Hyrum S Anderson and Phil Roth. EMBER: An Open Dataset for Training Static PE Malware Machine Learning Models. In *NeurIPS 2018 Workshop on Security in Machine Learning*, 2018.

[5] Microsoft Malware Classification Challenge (BIG 2015). `https://www.kaggle.com/c/malware-classification/`.

[6] Petr Gronát, Javier Aldana-Iuit, and Martin Bálek. MaxNet: Neural Network Architecture for Continuous Detection of Malicious Activity. In *2nd Deep Learning and Security Workshop*, 2018.

[7] K. Allix, T. F. Bissyand, J. Klein, and Y. L. Traon. AndroZoo: Collecting Millions of Android Apps for the Research Community. In *2016 IEEE/ACM 13th Working Conference on Mining Software Repositories (MSR)*, pp. 468–471, May 2016.

[8] Li Li, Jun Gao, Médéric Hurier, Pingfan Kong, Tegawendé F. Bissyandé, Alexandre Bartel, Jacques Klein, and Yves Le Traon. AndroZoo++: Collecting Millions of Android Apps and Their Metadata for the Research Community. *CoRR*, Vol. abs/1709.05281, , 2017.

[9] 東結香, 津田侑. マルウェアデータセットに関する調査. コンピュータセキュリティシンポジウム 2018, 2018.

[10] C. Rossow, C. J. Dietrich, C. Grier, C. Kreibich, V. Paxson, N. Pohlmann, H. Bos, and M. v. Steen. Prudent Practices for Designing Malware Experiments: Status Quo and Outlook. In *2012 IEEE Symposium on Security and Privacy*, pp. 65–79, May 2012.

[11] Daniele Ucci, Leonardo Aniello, and Roberto Baldoni. Survey of Machine Learning Techniques for Malware Analysis. October 2017.

[12] Jose G Moreno-Torres, Troy Raeder, Rocío Alaiz-Rodríguez, Nitesh V Chawla, and Francisco Herrera. A Unifying View on Dataset Shift in Classification. *Pattern Recognit.*, Vol. 45, No. 1, pp. 521–530, January 2012.

[13] Hidetoshi Shimodaira. Improving Predictive Inference under Covariate Shift by Weighting the Log-likelihood Function. *J. Stat. Plan. Inference*, Vol. 90, No. 2, pp. 227–244, October 2000.

[14] Afonso Fernandes Vaz, Rafael Izbicki, and Rafael Bassi Stern. Quantification Under Prior Probability Shift: the Ratio Estimator and its Extensions. *J. Mach. Learn. Res.*, Vol. 20, No. 79, pp. 1–33, 2019.

[15] João Gama, Indrė Žliobaitė, Albert Bifet, Mykola Pechenizkiy, and Abdelhamid Bouchachia. A Survey on Concept Drift Adaptation. *ACM Comput. Surv.*, Vol. 46, No. 4, pp. 44:1–44:37, March 2014.

[16] Imen Khamassi, Moamar Sayed-Mouchaweh, Moez Hammami, and Khaled Ghédira. Self-Adaptive Windowing Approach for Handling Complex Concept Drift. *Cognit. Comput.*, June 2015.

[17] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in Neural Information Processing Systems 30 (NIPS 2017)*, pp. 3146–3154, 2017.