Resilient Linear Classification: An Approach to Deal with Attacks on Training Data

Sangdon Park
Department of Computer &
Information Science
University of Pennsylvania
sangdonp@cis.upenn.edu

James Weimer
Department of Computer &
Information Science
University of Pennsylvania
weimerj@cis.upenn.edu

Insup Lee
Department of Computer &
Information Science
University of Pennsylvania
lee@cis.upenn.edu

ABSTRACT

Data-driven techniques are used in cyber-physical systems (CPS) for controlling autonomous vehicles, handling demand responses for energy management, and modeling human physiology for medical devices. These data-driven techniques extract models from training data, where their performance is often analyzed with respect to random errors in the training data. However, if the training data is maliciously altered by attackers, the effect of these attacks on the learning algorithms underpinning data-driven CPS have yet to be considered. In this paper, we analyze the resilience of classification algorithms to training data attacks. Specifically, a generic metric is proposed that is tailored to measure resilience of classification algorithms with respect to worst-case tampering of the training data. Using the metric, we show that traditional linear classification algorithms are resilient under restricted conditions. To overcome these limitations, we propose a linear classification algorithm with a majority constraint and prove that it is strictly more resilient than the traditional algorithms. Evaluations on both synthetic data and a real-world retrospective arrhythmia medical case-study show that the traditional algorithms are vulnerable to tampered training data, whereas the proposed algorithm is more resilient (as measured by worst-case tampering).

CCS CONCEPTS

•**Computing methodologies** →**Supervised learning by classification; Batch learning;** •**Computer systems organization** →**Embedded and cyber-physical systems;** •**Security and privacy** →*Domain-specific security and privacy architectures;*

KEYWORDS

cyber-physical systems, linear classification, training data attacks

1 INTRODUCTION

The penetration of data-driven techniques (*e.g.,* machine learning) to monitor and control a broad range of cyber-physical systems has sharply increased. Autonomous cars rely on visual object detectors learned from image data for recognizing objects[12, 25, 30]. Building demand response can be effectively handled by data-driven modeling and prediction of the electric usage of buildings [33]. Smart insulin pumps can adapt to type 1 diabetic patients using data-driven modeling of user-specific eating and pump-using behavior [13]. While data-driven CPS offer remarkable capabilities for enhanced performance, they also introduce unprecedented security vulnerabilities with the risk of malicious attacks having catastrophic consequences. Specifically, the training data used for learning (be it online or offline), is vulnerable to malicious tampering that can result in data-driven CPS reacting incorrectly to safety-critical events.

The training data for data-driven CPS can be tampered in several ways, depending on the application. In modern automobiles, multiple vulnerabilities have been demonstrated where hackers obtain full control of automobiles by eavesdropping a Controller Area Network (CAN) and injecting CAN messages [11, 29], which provides possibilities to inject malicious data being used for online learning algorithms [12, 25]. Furthermore, automobiles and robots, which rely on sensor inputs from global positioning system (GPS), inertial measurement unit (IMU) or wheel speed sensors, can be susceptible on spoofing attacks [26, 43, 45]. This means attackers can tamper training data collected from sensors. Hacking incidents on medical devices and hospitals [1–3] suggest attackers can tamper both device-level and data center-level training data. Moreover, attackers with knowledge of the underlying machine learning techniques – *e.g.,* support vector machines (SVMs), principal component analysis, logistic regression, artificial neural network, and (ensemble) decision trees – can strategically alter the training data to minimize the accuracy of the algorithms [6, 7, 23, 27, 34, 46], to maliciously affect the performance of data-driven CPS [12, 13, 25, 33, 39, 42, 48].

Capabilities provided by traditional *cyber defenses* (*e.g.,* communication channel encryption and authentication), *fault tolerant techniques* (*e.g.,* data sanitization [16], robust loss functions [51, 53], and robust learning [14, 20]), and *adversarial learning* [9, 17, 19] are necessary to secure data-driven CPS, but they are not sufficient. Specifically, the *cyber defenses* are insufficient for defending against cyber-physical attacks (*e.g.,* GPS spoofing [26]) where a sensing environment can be maliciously altered such that correctly functioning sensors and systems can act erroneously. These challenges are compounded in dynamic applications (e.g., autonomous driving and closed-loop physiological control) where accurate physical models,

commonly required for *fault tolerant systems*, are challenging to obtain. Moreover, *adversarial learning* literature (*e.g.,* [9, 17, 19]) usually assumes a known attacker behavior and/or goal – which is likely unknown in complex CPS applications. Due to the shortcomings of traditional approaches for securing the training data of data-driven CPS, it is necessary to consider techniques for *resilient machine learning* that can defend against cyber-physical attacks and make minimal assumptions on environments and attackers.

Towards the ultimate goal of attack-resilient machine learning, we propose a resilience metric for the analysis and design of learning algorithms under cyber-physical attacks. The metric aims to quantify the resilience of learning algorithms for analysis, which in turn contributes to designing resilient learning algorithms. Specifically, this work considers binary linear classification algorithms in the presence of maliciously tampered training data. Binary linear classification represents a basic building block for more complex classification approaches, such as neural network, decision trees, and boosting; thus, developing attack resilient linear classifiers can lead to more advance resilient machine learning algorithms. To analyze binary classifiers in the presence of training data attacks, we introduce a generic measure of resilience for classification in terms of worst-case errors.

Based on the resilience metric, traditional linear classification algorithms are evaluated. First, we prove the maximal resilience of any linear classification algorithm, which provides an upper bound of a resilience condition that can be achievable. Then, we prove that convex loss linear classification algorithms, such as SVMs, and 0-1 loss linear classification algorithm can not achieve maximal resilience. Based on these results, we introduce a majority 0-1 loss linear classification algorithm that is strictly more resilient than the traditional approaches and achieves the maximal resilience condition.

Finally, we evaluate the different classification algorithms, in the presence of attacks, on a synthetic dataset and a medical case-study, introduced in [24], to design a detector for arrhythmia (*i.e.,* irregular heart beat). The evaluation on synthetic data illustrates conditions when the different algorithms are (and are not) resilient, while the arrhythmia dataset serves to illustrate resilient binary classification in a real-world data-driven medical CPS (described in Section 7).

In summary, the contributions of this work include: (i) introducing, to our knowledge, the first assessment metric for analyzing binary classifier resilience; (ii) providing an analysis of the resilience of traditional binary classification techniques illustrating their shortcomings; (iii) describing a resilient classification approach that provides maximal resilience; (iv) evaluating in a retrospective real-world arrhythmia classification case-study.

The following section describes the work most closely related to the resilient classification problem considered herein. In Section 3, we define attacker capabilities and a resilience metric. In Section 4, the resilient classification problem is formally defined while an analysis of traditional linear classification algorithms is provided in Section 5. In Section 6, a new resilient linear classification algorithm is proposed which achieves maximal resilience for the attacker's capabilities considered. Section 7 illustrates the theoretical results using case studies on synthetic and medical data. The final section provides conclusions with discussion about countermeasures and future work.

2 RELATED WORK

This section presents the related works for CPS security (Section 2.1) and traditional error/attack models in the machine learning literature (Section 2.2).

2.1 CPS security

Though the security of learning systems for data-driven CPS has been an afterthought, the security of CPS has seen much effort in the past decade. A mathematical framework considering attacks on CPS is proposed in [10, 41]. The necessary and sufficient conditions on CPS with a failure detector such that a stealthy attacker can destabilize the system are provided in [35]. State estimation for an electric power system is analyzed in [47] assuming attackers know a partial model of the true system. Resilient state estimators for CPS that tolerate a bounded number of sensors and/or actuators attacks are considered in [18, 38]. In mobility-as-a-Service systems (*e.g.,* ride-sharing services), it has been demonstrated that a fraction of cars are maliciously called by fake reservation for denial-of-service [52]. Surgical tele-operated robotic systems can be affected by denial-of-service attacks on communication channels [8]. Energy management systems, especially when connected to building networks, are vulnerable to cyber attacks that impact on the systems operation. This vulnerability can be attenuated by applying resilient policy when attacks are detected [39]. While there has been much recent work on CPS security, these approaches are (in general) not directly applicable to data-driven CPS.

2.2 Learning with Errors

In this subsection, we review the literature on learning in the presence of training data errors most closely related to our work, where a more complete survey of the entire literature can be found in [22, 36]. The error models can be categorized as either label errors or feature errors in Table 1, according to their classical definitions [22, 36]. Under each error model, the performance of a learning algorithm is analyzed against whether it achieves a desired classifier.

label errors	class-independent (CICE)	[4]
	class-dependent (CDCE)	[36]
feature errors	uniform random (URAE)	[44]
	product random (PRAE)	[22]
	malicious errors (ME)	[28]

Table 1: Taxonomy of training data errors in the literature.

When labels in training data are corrupted, the training data is said to have label errors, which can be divided into two subtypes: class-independent classification errors (CICE) [4] and class-dependent classification errors (CDCE) [36]. The class-independent classification error model assumes the error probability of positive and negative labels are same while the class-dependent classification error model allows the different error probability for positive and negative labels.

When features in the training data are corrupted, the training data is said to have feature errors, which can be divided into three subtypes: uniform random attribute errors [44], product random attribute errors [22], and malicious errors [28]. Both the uniform random attribute error (URAE) and the product random attribute error (PRAE) models assume errors on features (*i.e.,* columns of the feature matrix), where URAE assumes the same error probability for

all features and PRAE allows for variable error probabilities. From a CPS perspective, attacks on individual features require that each column of the feature matrix corresponds to a single attack surface (*e.g.*, a single sensor) – which restricts the use of multiple sensors in a single feature, as common in data-driven CPS [12, 25]. Different from URAE and PRAE, the malicious error (ME) model assumes arbitrary attacks on feature vectors (*i.e.*, rows of the feature matrix). However, the ME model assumes the probabilities of attacking the feature vectors corresponding to positive and negative labels are the same – a condition which may not be satisfied by savvy attackers. In contrast to this, our error (or attack) model assumes the probabilities can be different.

3 SETUP FOR RESILIENT BINARY CLASSIFICATION

This section introduces essential definitions that are the bases for describing resilient binary classification problem. In the following subsections, we present a traditional binary linear classification problem (Section 3.1), define our attacker assumptions (Section 3.2), and introduce a resilience metric (Section 3.3).

Notationally, we write $\mathbb{R}$, $\mathbb{R}_0^+$, $\mathbb{N}_0$ and $[a, b]$ to denote the set of real numbers, non-negative real numbers, non-negative integers, and integers from a to b, respectively. We write $\mathbb{1}$ as the ones vector of an appropriate size and $|\cdot|$ to denote the cardinality (*i.e.,* number of elements) of a finite set. The sign function is written as sign : $\mathbb{R} \rightarrow \{+1, -1\}$ and $\mathbb{1}\{\cdot\}$ corresponds to the indicator function that maps true and false to 1 and 0. Additionally, we write ℓ_{01} to denote a 0-1 loss function, such that $\ell_{01}(y_i, h(x_i)) = \mathbb{1}\{y_i \neq \text{sign}(h(x_i))\}$. Lastly, $\|\cdot\|_1$ and $\|\cdot\|_\infty$ denotes the 1-norm and the ∞-norm, respectively. See Table 2 for the glossary of mathematical notations in this paper.

3.1 Traditional Binary Classification

We begin by considering the traditional problem of binary classification in the absence of attacks (or errors). Namely, we consider un-attacked training data $\hat{D} = \{(x_i, y_i)\}_{i=1}^N \in \hat{\mathcal{D}}$, where N is the number of training data pairs, $\hat{\mathcal{D}} = \{D \subset \mathcal{X} \times \mathcal{Y} \mid |D| = N\}$ is a class of training data with N pairs, $\mathcal{X} \subseteq \mathbb{R}^p$ corresponds to a set of feature vectors (or attributes), $\mathcal{Y} = \{-1, +1\}$ denotes the set of labels (or classes), and each element of x_i is called a feature. In a traditional (binary) classification problem, such as [50], given training data, a designer specifies a set of (real-valued) classifiers $\mathcal{H} \subseteq \mathbb{R}^{\mathcal{X}}$, and a loss function $\ell : \mathcal{Y} \times \mathbb{R} \rightarrow \mathbb{R}_0^+$, to learn a (real-valued) classifier $\hat{h} \in \mathcal{H}$, according to

$$P_{\mathcal{H}, \ell}(\hat{D}) : \hat{h} = \arg \min_{h \in \mathcal{H}} \|W \cdot \hat{R}_\ell(h|\hat{D})\|_1, \qquad (1)$$

where $W \in \mathbb{R}^{2 \times 2}$ is the diagonal matrix with the positive risk weight w^+ and the negative risk weight w^- on the diagonal, and zeros elsewhere. $\hat{R}_\ell(h|\hat{D}) \in \mathbb{R}^2$ denotes the bi-dimensional vector of empirical risks corresponding to the positive and negative training data. Specifically, we write $\hat{R}_\ell(h|\hat{D}) = \begin{bmatrix} \hat{R}_\ell(h|\hat{D}^+) & \hat{R}_\ell(h|\hat{D}^-) \end{bmatrix}^\top$, where $\hat{R}_\ell(h|\hat{D}) = \frac{1}{|\hat{D}|} \sum_{i=1}^{|\hat{D}|} \ell(y_i, h(x_i))$ is the normalized empirical risk evaluated over the training data and $\hat{D}^+$ and $\hat{D}^-$ corresponds to the mutually exclusive sets of positive and negative training data pairs, respectively, such that $\hat{D} = \hat{D}^+ \cup \hat{D}^-$. We note that we use

symbol	description
$\hat{D}$	actual training data
$\hat{\mathcal{D}}$	class of training data
$\hat{D}^+$	positive training data
$\hat{D}^-$	negative training data
Θ	set of attacker capability parameters
α	attacker capability parameter where $\alpha \in \Theta$
$\hat{D}_\alpha$	tapered training data
$\hat{\mathcal{D}}_\alpha$	class of tampered training data
$\hat{D}_\alpha^+$	positive tampered training data
$\hat{D}_\alpha^-$	negative tampered training data
N	number of training data pairs (*i.e.,* $|\hat{D}|$)
$\boldsymbol{N}$	pair of $|\hat{D}^+|$ and $|\hat{D}^-|$
$\mathcal{F}$	set of classifiers
$\mathcal{H}$	subset of classifiers (*i.e.,* $\mathcal{H} \subseteq \mathcal{F}$)
$\mathcal{L}$	set of linear classifiers
$\mathcal{S}$	set of loss functions
ℓ	loss function in $\mathcal{S}$
ℓ_c	convex loss function in $\mathcal{S}$
ℓ_{01}	0-1 loss function in $\mathcal{S}$
P	classification algorithm
$\mathcal{P}_{\mathcal{F}, \mathcal{S}}$	class of classification algorithms
$P_{\mathcal{H}, \ell}$	classification algorithm
$\mathcal{P}_{\mathcal{L}, \mathcal{S}}$	class of linear classification algorithms
$P_{\mathcal{L}, \ell_c}$	convex loss linear classification algorithm
$P_{\mathcal{L}, \ell_{01}}$	0-1 loss linear classification algorithm
$P_{\mathcal{M}, \ell_{01}}$	majority 0-1 loss linear classification algorithm
g_P	resilience bound of a classification algorithm P
$\mathcal{G}$	set of resilience bounds
$\mathcal{A}_P$	resilience attack condition of a classification algorithm P where $\mathcal{A}_P \subseteq \Theta$
$\mathcal{B}_P$	perfectly attackable condition of a classification algorithm P where $\mathcal{B}_P \subseteq \Theta$

Table 2: The glossary of mathematical notations.

Equation (1) for distinguishing empirical risks over positive and negative training data, but it is equivalent to the standard notation [50] if $w^+ = |\hat{D}^+|$ and $w^- = |\hat{D}^-|$, and we assume the standard notion in this paper. Also, we call $\hat{h}$ a classifier (*i.e.,* $\hat{h} \in \mathcal{Y}^{\mathcal{X}}$) or a real-valued classifier (*i.e.,* $\hat{h}' \in \mathbb{R}^{\mathcal{X}}$), interchangeably, assuming the composition of a sign function and a real-valued classifier (*i.e.,* sign $\circ \hat{h}' \in \mathcal{Y}^{\mathcal{X}}$) is a classifier. Moreover, we say N is the number of training data pairs or $N = (|\hat{D}^+|, |\hat{D}^-|)$, interchangeably.

In this paper, we consider a set of classification algorithms $\mathcal{P}_{\mathcal{F}, \mathcal{S}}$, where $\mathcal{F}$ is a set of classifiers and $\mathcal{S}$ is the set of monotonically non-increasing functions that are lower-bounded by a 0-1 loss function. Specifically, the loss function $\ell(y, h(x))$ is represented as $\ell(y, h(x)) = \phi(t)$, where $t = yh(x)$, ϕ is lower-bounded by $\mathbb{1}\{t \leq 0\}$, $\phi(0) = 1$, ϕ is a monotonically non-increasing function, and $\lim_{t \to \infty} \phi(t) = c$ for some scalar $c < 1$. We note that these assumptions generalize a convex loss [5] to cover a non-convex loss.

Each algorithm in $\mathcal{P}_{\mathcal{F}, \mathcal{S}}$ is a map from a class of training data $\hat{\mathcal{D}}$ to a subset of $\mathcal{F}$ that uses a loss function in $\mathcal{S}$ (*i.e.,* $\mathcal{P}_{\mathcal{F}, \mathcal{S}} = \{P_{F, \ell} | F \subseteq \mathcal{F}, \ell \in \mathcal{S}\}$). Thus, empirical risk minimization (Equation (1)) for any hypothesis space $\mathcal{H} \subseteq \mathcal{F}$ and a loss function

$\ell \in \mathcal{S}$ is also a classification algorithm considered here (*i.e.,* $P_{\mathcal{H},\ell} \in \mathcal{P}_{\mathcal{F},\mathcal{S}}$).

3.2 Attacker Capabilities

In this work, we introduce a new class of an attack based on the number of training data elements the attacker can manipulate, referenced to as a *bounded feature attack* (BFA). Specifically, in this class of an attack, we assume the attacker has the following three capabilities; (i) The attacker knows the classification algorithm to be attacked, (ii) the attacker has unbounded computing power, (iii) the attacker knows all the training data (both before and after tampering), and (iv) the attacker can tamper the training data. However, the ability to tamper the training data is limited such that the *tampered training data* $\hat{D}_\alpha$ differs from the original training data $\hat{D}$ by a finite number of elements. We parameterize the tampered training data using an attacker capability parameter $\alpha = (\alpha^+, \alpha^-) \in \Theta = [0, |\hat{D}^+|] \times [0, |\hat{D}^-|]$ such that at most α^+ and α^- number of positive and negative feature vectors are maliciously manipulated, respectively. Formally, the α-bounded feature attack is defined as follows:

Definition 1 (bounded feature attack). Given $P_{\mathcal{H},\ell}$, $\hat{D}$, and α, then $\hat{D}_\alpha$ is a bounded feature attack (BFA) if $\hat{D}_\alpha \in \hat{\mathcal{D}}$ satisfies the following two conditions:

$$\text{(i) } |\hat{D}_\alpha^+ \backslash \hat{D}^+| \le \alpha^+, \text{ (ii) } |\hat{D}_\alpha^- \backslash \hat{D}^-| \le \alpha^-. \tag{2}$$

Additionally, let $\hat{\mathcal{D}}_\alpha$ be the set of all such $\hat{D}_\alpha$ (*i.e.,* $\hat{D}_\alpha \in \hat{\mathcal{D}}_\alpha \subseteq \hat{\mathcal{D}}$). We emphasize that Definition 1 only specifies what an attacker can do and which information can be used – but does not indicate *how* the attacker changes the data. This definition is consistent with the attacker capability definition used in the CPS security literature (*e.g.,* [18, 38]). Moreover, α is unknown in general, so algorithms considered in this paper do not assume anything on α.

The BFA represents a practical model of attacker capabilities. For example, assume several devices collect medical data and store it in the hospitals central data center. An attacker can exploit known vulnerabilities of the enterprise system of the data center to gain read access on all data (*i.e.,* knows all data), but can only alter data from specific devices having a certain vulnerabilities (*i.e.,* attacks some of the data). Here, we assume obtaining write access is more difficult than obtaining read access.

In comparison to other attack models discussed in Section 2, we emphasize that the proposed attacker capabilities are quite general; we only limit the number of tampered feature vectors. By definition, the BFA includes the ME; moreover, the BFA can represent attacks on (maliciously) manipulating labels in training data. This is achieved by manipulating a positive feature vector into one of the negative feature vectors, which effectively switches the label from positive to negative and suggests the BFA includes the CICE and CDCE models.

3.3 Resilience Metric

To evaluate a classification algorithm in the presence of a BFA, we aim to quantify the effect of a BFA on the learned classifier's *worst-case error metric* over all training data and all possible attacks. In traditional detection and classification theory, the true-positive and true-negative rates (or the corresponding false-positive and false-negative rates) are commonly used to evaluate the performance of

a classifier. We introduce a generic metric that utilizes the false-positive and false-negative rates such that it measures the worst-case weighted p-norm of the two error rates over all training data and all feasible attacks, defined as follows:

Definition 2 (resilience metric). Given N and α, the resilience of $P_{\mathcal{H},\ell}$ is quantified as the worst-case weighted p-norm of error rates over all $\hat{D} \in \hat{\mathcal{D}}$ and $\hat{D}_\alpha \in \hat{\mathcal{D}}_\alpha$, stated as

$$V_{W,p}(P_{\mathcal{H},\ell}|N,\alpha) = \max_{\hat{D},\hat{D}_\alpha} \left\| W \cdot \hat{R}_{\ell_{01}}(P_{\mathcal{H},\ell}(\hat{D}_\alpha)|\hat{D}) \right\|_p. \tag{3}$$

This resilience metric measures the performance of a classification algorithm (*i.e.,* $V_{W,p}(\cdot)$) in the presence of the worst-cast attack given the attacker capability parameter α.

In this work, we select $w^+ = w^- = 1$ and $p = \infty$. So, $V_{W,p}(\cdot)$ ranges from zero to one and equals one if $P_{\mathcal{H},\ell}$ outputs any classifier such that an attack could result in mis-classification of all the positive or negative feature vectors in the un-attacked training data $\hat{D}$. For notational simplicity, we denote $V_{W,p}(\cdot)$ as $V(\cdot)$. Our selection of $w^+ = w^- = 1$ means each label is equally important to model the unknown attacker's preference for each label. The choice of $p = \infty$ is motivated by the worst-case classification approach that minimizes the maximum of class-conditional error rates [31].

We note that other norm measures could have been chosen rather than the ∞-norm. For instance, selecting $p = 1$ results in evaluating the 1-norm of the false-positive and false-negative rates, where $V(\cdot) \ge 1$ implies that the classifier is at least as bad as a weighted coin-flip (*i.e.,* a trivial classifier) [49]. Additionally, selecting $p = 2$ specifies $V(\cdot)$ to be the Euclidean distance to the classifier error of zero. In general, the selection of p in Equation (3) can vary based upon the security concerns.

Applying the resilience metric in Equation (3), a binary classification algorithm $P_{\mathcal{H},\ell}$ can be evaluated for given N and α. Furthermore, the resilience metric can be upper bounded by a function in N and α, *i.e.,* $g(N,\alpha) : (\mathbb{N}_0 \times \mathbb{N}_0 \times \mathbb{N}_0) \to [0,1]$ where $\mathcal{G}$ is the set of all such g. Then, the upper bound characterizes the property of an algorithm over various attack parameters. In this case, the classification algorithm is called a $g(N,\alpha)$-resilient algorithm. Formally, we define the resilience property of a classification algorithm in the context of this work as follows.

Definition 3 ($g(N,\alpha)$-resilience). A classification algorithm P is $g(N,\alpha)$-resilient to a BFA if

$$V(P|N,\alpha) \le g(N,\alpha), \tag{4}$$

where $g \in \mathcal{G}$ denotes the worst-case resilience bound.

This worst-case resilience bound plays a key role in defining resilient binary classification problem, which is defined in the following section.

4 PROBLEM FORMULATION

This section formulates the problem of analyzing (and ultimately designing) resilient binary classification algorithms with respect to training data attacks. Specifically, given the number of positive and negative training data N, a set of classifiers $\mathcal{F}$, a set of loss functions $\mathcal{S}$, and a class of algorithms $\mathcal{P}_{\mathcal{F},\mathcal{S}}$, the goal of this paper is finding a classification algorithm P and a resilience bound g that

minimize the error of the resilience bound such that P is $g(N, \alpha)$-resilient to a BFA. Here, to measure the error of the resilience bound we use the number of α that makes the resilience bound maximum (*i.e.*, $\sum_{\alpha \in \Theta} \mathbb{1}\{g(N, \alpha) = 1\}$), but any other error measure can be used. In short, a resilient binary classification problem is defined as follows:

PROBLEM (BFA RESILIENT BINARY CLASSIFICATION PROBLEM). *Given N, $\mathcal{F}$, $\mathcal{S}$, and $\mathcal{P}_{\mathcal{F},\mathcal{S}}$, a BFA resilient binary classification problem is to find a classification algorithm $P \in \mathcal{P}_{\mathcal{F},\mathcal{S}}$ and a resilience bound $g \in \mathcal{G}$ according to*

$$(P, g) = \arg \min_{P,g} \sum_{\alpha \in \Theta} \mathbb{1}\{g(N, \alpha) = 1\} \qquad (5)$$

$$s.t. \ V(P|N, \alpha) \leq g(N, \alpha), \forall \alpha \in \Theta. \qquad (6)$$

We note several implications of the above problem. First, a feasible classification algorithm of this problem guarantees the worst-case performance characterized by g since the constraint of the problem enforces that the worst-case error (*i.e.*, $V(\cdot)$) is bounded by g for all possible attacks (*i.e.*, $\forall \alpha$). Next, the problem can consider the capabilities of classification algorithms by encoding prior knowledge on the class of classification algorithms $\mathcal{P}_{\mathcal{F},\mathcal{S}}$. Specifically, $\mathcal{P}_{\mathcal{F},\mathcal{S}}$ can be the class of classification algorithms that uses empirical risk minimization over $\mathcal{F}$ with convex loss functions. The resilient classification problem then finds a classification algorithm in the restricted class of $\mathcal{P}_{\mathcal{F},\mathcal{S}}$. We note that when choosing the restricted class of classification algorithms in this paper, we do not consider the attacker capability parameter α, implying we focus on finding an algorithm without assumptions on α. Then, the ultimate goal of the resilient binary classification problem is making $g(\cdot) \leq \epsilon$ for some N and for all α given conditions on $\mathcal{P}_{\mathcal{F},\mathcal{S}}$, where ϵ is a sufficiently small scalar. Finally, we note that the resilient binary classification problem is related to the problem of minimizing generalization error of a classifier considered in traditional classification (See Section 1 in [40]).

We note that in this paper a BFA resilient binary classification problem is simply called a resilient binary classification problem assuming a BFA as an attack model. g_P denotes the optimal g of the resilient binary classification problem to explicitly represent the dependency on P. Also, an algorithm A is *more resilient* than an algorithm B if $\sum_{\alpha \in \Theta} \mathbb{1}\{g_A(N, \alpha) = 1\} \leq \sum_{\alpha \in \Theta} \mathbb{1}\{g_B(N, \alpha) = 1\}$, and A, B, g_A, and g_B satisfy the constraint in the problem (Equation (6)). In the following section, we utilize the definition of the resilient binary classification problem to analyze traditional linear classification algorithms for resilience under a BFA.

5 RESILIENCE OF TRADITIONAL LINEAR CLASSIFICATION

Traditional classification algorithms (e.g. SVMs or 0-1 loss linear classification) rarely consider a learning environment that is partially controlled by attackers. Here, we focus on linear classification algorithms (*i.e.*, $\mathcal{F} = \mathcal{L}$, where $\mathcal{L}$ is the set of linear functions), which is a basic building block for more complex classification algorithms. In this section, we analyze whether traditional linear classification algorithms are resilient. First, linear classification algorithms with various convex loss functions are analyzed (Section 5.1). Next, a linear classification algorithm with a 0-1 loss function is analyzed (Section 5.2).

In the following, we strictly consider un-attacked training data $\hat{D}$ for which a perfect classifier exists – *i.e.*, for some $h \in \mathcal{H}$, $\|W \cdot \hat{R}_\ell(h|\hat{D})\|_1 = 0$ – such that only errors are introduced by attacks. We note that, in practice, the empirical risk over training data is rarely equal to zero due to errors from noise and an assumption on $\mathcal{H}$. However, by treating errors as attacks, the theoretical results in the following sections can be interpreted as assuming *worst-case errors* – *e.g.*, attacks.

The resilient binary classification problem finds a classification algorithm P and a resilience bound g_P, but the resilience bound may be trivial for some α, *i.e.*, $g(N, \alpha) = 1$. Thus, it is worthwhile to find a *resilience attack condition*, $\mathcal{A}_P \subseteq \Theta$, such that $g(N, \alpha)$ is non-trivial for all $\alpha \in \mathcal{A}_P$. In this case, we say that P is *resilient* w.r.t. $\mathcal{A}_P$.

Definition 4 (resilient w.r.t. $\mathcal{A}_P$). Given N, P, g_P, and $\mathcal{A}_P$, a classification algorithm P is resilient w.r.t $\mathcal{A}_P$ if the algorithm is $g(N, \alpha)$-resilient to a BFA and $g(N, \alpha) < 1$ for all $\alpha \in \mathcal{A}_P$.

Here, we emphasize that finding an attack condition on α that makes a classification algorithm 1-resilient to a BFA (*i.e.*, finding some set $\mathcal{B}_P$ such that $\mathcal{B}_P \subseteq \mathcal{A}_P^c$) is equally important to finding the resilience attack condition $\mathcal{A}_P$ since $\alpha \in \mathcal{B}_P$ can be a "breaking point" of the algorithm P. We refer to $\mathcal{B}_P$ as the *perfectly attackable condition* of P. Thus, we introduce a new notion, *perfectly attackable w.r.t $\mathcal{B}_P$*, which is formally described as follows:

Definition 5 (perfectly attackable w.r.t. $\mathcal{B}_P$). Given N, P, and $\mathcal{B}_P$, a classification algorithm P is perfectly attackable w.r.t $\mathcal{B}_P$ if the algorithm is 1-resilient to a BFA for all $\alpha \in \mathcal{B}_P$.

Next, we introduce a *maximal resilience attack condition* $\bar{\mathcal{A}} \subseteq \Theta$. It is a resilience attack condition of some linear classification algorithm or a combination of algorithms where the size of the condition is maximal. Formally,

Definition 6 (maximally resilient condition). $\bar{\mathcal{A}}$ is a maximal resilient condition if $\bar{\mathcal{A}} = \cup_{\ell \in \mathcal{S}} \mathcal{A}_{P_{\mathcal{L},\ell}}$.

We note that if the resilience attack condition $\mathcal{A}_P$ of a classification algorithm P is same as $\bar{\mathcal{A}}$, we say that P is *maximally resilient*. To find the maximal resilience attack condition $\bar{\mathcal{A}}$, we consider some superset of it (*i.e.*, $\bar{\mathcal{B}}^c$ such that $\bar{\mathcal{A}} \subseteq \bar{\mathcal{B}}^c$), which is a theoretical upper bound of the maximal resilience attack condition. We argue that there exists some classification algorithm that achieves the attack condition $\bar{\mathcal{B}}^c$. This then implies $\bar{\mathcal{B}}^c$ is the maximal resilience attack condition (See Theorem 2).

One example of $\bar{\mathcal{B}}$ can be some subset of $\cap_{\ell \in \mathcal{S}} \mathcal{B}_{P_{\mathcal{L},\ell}}$ due to $\bar{\mathcal{A}} = \cup_{\ell \in \mathcal{S}} \mathcal{A}_{P_{\mathcal{L},\ell}} \subseteq \cup_{\ell \in \mathcal{S}} \mathcal{B}_{P_{\mathcal{L},\ell}}^c \subseteq \bar{\mathcal{B}}^c$. The following theorems formally state $\bar{\mathcal{B}}$ and a condition when $\bar{\mathcal{B}}^c$ is the maximal resilience attack condition.

THEOREM 1. *Given $|\hat{D}^+|$ and $|\hat{D}^-|$, let $\bar{\mathcal{B}}$ be*

$$\left\{\alpha \Big| \alpha^+ \geq \frac{1}{2}|\hat{D}^+| \ or \ \alpha^- \geq \frac{1}{2}|\hat{D}^-|\right\}. \qquad (7)$$

For all $\ell \in \mathcal{S}$, $P_{\mathcal{L},\ell}$ is perfectly attackable w.r.t. $\bar{\mathcal{B}}$.

PROOF SKETCH. For all N, $\alpha \in \bar{\mathcal{B}}$, and $\ell \in \mathcal{S}$, we find some $\hat{D}$ and $\hat{D}_\alpha$ where $V(P_{\mathcal{L},\ell}(\hat{D}_\alpha)|N, \alpha) = 1$. See Section 2.2.1 in [40] for details. □

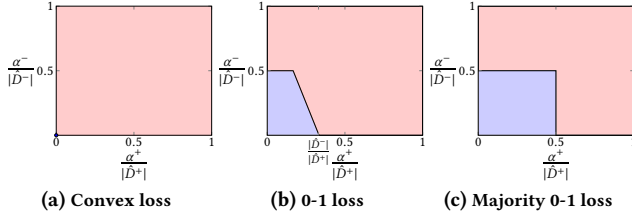**(a) Convex loss**   **(b) 0-1 loss**   **(c) Majority 0-1 loss**

Figure 1: Perfectly attackable conditions on α for each linear classification algorithm (colored in red). Assume $|\hat{D}^+| = 75$, and $|\hat{D}^-| = 25$.

THEOREM 2. *If there exists $P \in \mathcal{P}_{\mathcal{L},\mathcal{S}}$ such that $\mathcal{A}_P = \bar{\mathcal{B}}^c$, then $\bar{\mathcal{B}}^c$ is the maximal resilience attack condition.*

PROOF SKETCH. We use the following two set relations to prove $\bar{\mathcal{A}} = \bar{\mathcal{B}}^c$: (1) $\bar{\mathcal{A}} \subseteq \bar{\mathcal{B}}^c = \mathcal{A}_P$ and (2) $\mathcal{A}_P \subseteq \bar{\mathcal{A}}$. See Section 2.2.2 in [40] for details. □

The intuitive interpretation of $\bar{\mathcal{B}}$ is that if the number of tampered positive or negative feature vectors is greater than or equal to the half of $|\hat{D}^+|$ or $|\hat{D}^-|$, respectively, then any linear classification algorithm trained with this training data can be perfectly attackable w.r.t. $\bar{\mathcal{B}}$. We note that in Section 6 we show $\bar{\mathcal{B}}$ is actually the maximal resilience attack condition. Thus, we assume this from now on. In the following subsections, we show that two classical approaches do not achieve the maximal resilience: (i) convex loss linear classification; and (ii) 0-1 loss linear classification.

5.1 Convex Loss Linear Classification

In this section, the class of convex-loss linear classification algorithms is considered, where it is the collection of $P_{\mathcal{L},\ell_c} \in \mathcal{P}_{\mathcal{L},\mathcal{S}}$, where ℓ_c is any convex relaxation of a 0-1 loss function, such as a hinge loss function. SVMs and a maximum likelihood learning of logistic regression belong to this class. We prove that any algorithm in this class is perfectly attackable w.r.t. some attack condition where an attacker can tamper at least one feature vector. Let $\mathcal{B}_{P_{\mathcal{L},\ell_c}}$ be the attack condition for the convex-loss linear classification algorithms being perfectly attackable, and then the attack condition is formally stated as follows:

PROPOSITION 1. *Let $\mathcal{B}_{P_{\mathcal{L},\ell_c}}$ be the set of α that satisfies one of the following two conditions:*

$$(i)\ \alpha^+ > 0,\ (ii)\ \alpha^- > 0. \tag{8}$$

Then, $P_{\mathcal{L},\ell_c}$ is perfectly attackable w.r.t $\mathcal{B}_{P_{\mathcal{L},\ell_c}}$ and resilient w.r.t. $\mathcal{B}^c_{P_{\mathcal{L},\ell_c}}$.

PROOF SKETCH. The idea of "perfectly attackable" proof is that for all N and $\alpha \in \mathcal{B}_{P_{\mathcal{L},\ell_c}}$ we find some $\hat{D}$ and $\hat{D}_\alpha$ where $V(P_{\mathcal{L},\ell_c}(\hat{D}_\alpha)|N,\alpha) = 1$. The "resilient" proof is trivial. See Section 2.2.3 in [40] for details. □

This implies even though an attacker has weak ability to tamper training data, it can make the algorithm misclassify all positive or all negative feature vectors of un-attacked training data by tampering only one positive or negative feature vector (See Figure 1a for the visualization of the perfectly attackable condition on α). For example, data-driven CPS that use SVMs to train intrusion detectors [39] can be vulnerable if an attacker can tamper at least one feature vector. We note that convex-loss linear classification algorithms are not maximally resilient since $\mathcal{B}^c_{P_{\mathcal{L},\ell_c}} \subset \bar{\mathcal{B}}^c$.

5.2 0-1 Loss Linear Classification

A 0-1 loss linear classification algorithm is defined as $P_{\mathcal{L},\ell_{01}} \in \mathcal{P}_{\mathcal{L},\mathcal{S}}$, where $\ell_{01}(\cdot)$ is a 0-1 loss function. We prove that the 0-1 loss linear classification algorithm is perfectly attackable w.r.t. some attack condition where the number of tampered positive or negative feature vectors is greater than or equal to the half of $|\hat{D}^+|$ or $|\hat{D}^-|$, respectively, or the sum of the number of tampered positive feature vectors and the number of tampered negative feature vectors is greater than or equal to $|\hat{D}^-|$ or $|\hat{D}^+|$. Let $\mathcal{B}_{P_{\mathcal{L},\ell_{01}}}$ be the attack condition for the 0-1 loss linear classification algorithm being perfectly attackable, and then the attack condition is formally stated as follows:

PROPOSITION 2. *Given $|\hat{D}^+|$ and $|\hat{D}^-|$, let $\mathcal{B}_{P_{\mathcal{L},\ell_{01}}}$ be the set of α that satisfies one of the following four conditions:*

$$(i)\ \alpha^+ \geq \frac{1}{2}|\hat{D}^+|, \qquad (ii)\ \alpha^- \geq \frac{1}{2}|\hat{D}^-|,$$

$$(iii)\ \alpha^+ + \alpha^- \geq |\hat{D}^-|, \qquad (iv)\ \alpha^+ + \alpha^- \geq |\hat{D}^+|. \tag{9}$$

Then, $P_{\mathcal{L},\ell_{01}}$ is perfectly attackable w.r.t. $\mathcal{B}_{P_{\mathcal{L},\ell_{01}}}$.

PROOF SKETCH. For all N and $\alpha \in \mathcal{B}_{P_{\mathcal{L},\ell_{01}}}$ we find some $\hat{D}$ and $\hat{D}_\alpha$ where $V(P_{\mathcal{L},\ell_{01}}(\hat{D}_\alpha)|N,\alpha) = 1$. See Section 2.2.4 in [40] for details. □

This proposition implies the 0-1 loss linear classification is strictly more resilient than convex one (See Figure 1b for comparison). Thus, different to the convex case, tampering single feature vector is not critical for the 0-1 loss linear classification. This means any CPS using convex linear classification algorithms [12, 39, 42] can be converted into the 0-1 linear classification algorithm to defend against the single feature vector tampering; however, neither approach can provide maximal resilience due to $\mathcal{B}^c_{P_{\mathcal{L},\ell_{01}}} \subset \bar{\mathcal{B}}^c$.

6 RESILIENT LINEAR CLASSIFICATION

In this section, we propose a maximally resilient linear classification algorithm. A majority 0-1 loss linear classification is defined as $P_{\mathcal{M},\ell_{01}} \in \mathcal{P}_{\mathcal{L},\mathcal{S}}$, where $\mathcal{M}$ denotes a majority constraint that restricts a feasible set of classifiers by only allowing a classifier that correctly classifies at least half of positive and negative feature vectors, according to

$$\mathcal{M} = \left\{ h \in \mathcal{L} \middle| \hat{R}_{\ell_{01}}(h|\hat{D}_\alpha^+) < \frac{1}{2} \text{ and } \hat{R}_{\ell_{01}}(h|\hat{D}_\alpha^-) < \frac{1}{2} \right\}. \tag{10}$$

In the following subsections, the resilience proof and the worst-case resilience bound of the majority 0-1 classification are provided.

6.1 Resilience of Majority 0-1 Loss Linear Classification

The majority 0-1 loss linear classification is perfectly attackable w.r.t. some attack condition where an attacker can manipulate greater than or equal to the half of $|\hat{D}^+|$ or $|\hat{D}^-|$. Let $\mathcal{B}_{P_{\mathcal{M},\ell_{01}}}$ be the attack condition for the majority 0-1-loss linear classification algorithms being perfectly attackable, and then the attack condition is formally stated as follows:

THEOREM 3. *Given $|\hat{D}^+|$ and $|\hat{D}^-|$, let $\mathcal{B}_{P_{\mathcal{M},\ell_{01}}}$ be*

$$\left\{ \alpha \middle| \alpha^+ \geq \frac{1}{2}|\hat{D}^+| \text{ or } \alpha^- \geq \frac{1}{2}|\hat{D}^-| \right\}. \tag{11}$$

Then, $P_{\mathcal{M},\ell_{01}}$ is perfectly attackable w.r.t. $\mathcal{B}_{P_{\mathcal{M},\ell_{01}}}$ and resilient w.r.t. $\mathcal{B}^c_{P_{\mathcal{M},\ell_{01}}}$.

PROOF SKETCH. The ideal of "perfectly attackable" proof is that for all N and $\alpha \in \mathcal{B}_{P_{\mathcal{M},\ell_{01}}}$ we find some $\hat{D}$ and $\hat{D}_\alpha$ where $V(P_{\mathcal{M},\ell_{01}}(\hat{D}_\alpha)|N,\alpha)=1$. For the "resilient" proof, we exploit the property of the majority constraint. See Section 2.2.5 in [40] for details. □

This result shows that the majority 0-1 loss linear classification algorithm is more resilient than traditional linear classification algorithms, which is also illustrated in Figure 1c. Furthermore, it achieves the maximal resilience condition (Theorem 1) due to $\mathcal{B}_{P_{\mathcal{M},\ell_{01}}} = \bar{\mathcal{B}}$, showing this algorithm achieves the maximal resilience attack condition.

6.2 Robustness of Resilient Classification

If a classification algorithm is resilient, it is worth analyzing the degree of resilience. If $\alpha \in \mathcal{A}_{P_{\mathcal{M},\ell_{01}}}$, where $\mathcal{A}_{P_{\mathcal{M},\ell_{01}}} = \mathcal{B}^c_{P_{\mathcal{M},\ell_{01}}}$, then the worst-case resilience bound g of the majority 0-1 loss classification algorithm is nearly proportional to the tampering ability of an attacker, which is formally stated as follows:

THEOREM 4. *Given $|\hat{D}^+|$, $|\hat{D}^-|$, and $\alpha \in \mathcal{A}_{P_{\mathcal{M},\ell_{01}}}$, the resilience bound g of $P_{\mathcal{M},\ell_{01}}$ can be computed as follows:*

$$g(N,\alpha) = \max\left(\frac{\min\left(2\alpha^+ + \alpha^-, \alpha^+ + \frac{|\hat{D}^+|-1}{2}\right)}{|\hat{D}^+|}, \right.$$
$$\left. \frac{\min\left(\alpha^+ + 2\alpha^-, \alpha^- + \frac{|\hat{D}^-|-1}{2}\right)}{|\hat{D}^-|} \right). \qquad (12)$$

PROOF SKETCH. To prove $V(\cdot)$ is bounded by $g(\cdot)$ for all N and $\alpha \in \mathcal{A}_{P_{\mathcal{M},\ell_{01}}}$, we exploit the optimality condition of an optimal classifier $P_{\mathcal{M},\ell_{01}}(\hat{D}_\alpha)$ and the property of the majority constraint. To prove that the bound is tight for all N and $\alpha \in \mathcal{A}_{P_{\mathcal{M},\ell_{01}}}$, we find some $\hat{D}$ and $\hat{D}_\alpha$ where $V(\cdot) = g(\cdot)$. See Section 2.2.6 in [40] for details. □

This theorem shows that if $\alpha \in \mathcal{A}_{P_{\mathcal{M},\ell_{01}}}$, the resilience bound is non-trivial. Also, it shows that even if the attacker capability parameter α is restricted (i.e., $\alpha \in \mathcal{A}_{P_{\mathcal{M},\ell_{01}}}$) to ensure that the algorithm is resilient w.r.t. $\mathcal{A}_{P_{\mathcal{M},\ell_{01}}}$, the tampered portion of training data still affects on the accuracy of the algorithm. Finally, we note that the resilience bound g is tight.

7 CASE STUDY

In this section, we validate the proven resilience of algorithms experimentally. Qualitative results on synthetic data are presented in Figure 3 and results on a real-world retrospective arrhythmia data are shown in Table 3.

The majority 0-1 loss linear classification algorithm is formulated in the following mixed integer linear program (MILP).

$$\min_{h,e,z} \quad \mathbf{1}^\top z + \lambda \|h\|_2$$
$$\text{s.t.} \quad \forall i, e_i \geq 1 - y_i h^\top x_i, \ -\delta z \leq e \leq \delta z$$
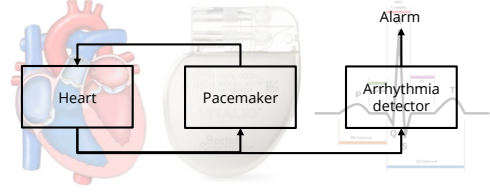$$\mathbf{1}_+^\top z \leq \frac{1}{2}(|\hat{D}_\alpha^+|-1), \ \mathbf{1}_-^\top z \leq \frac{1}{2}(|\hat{D}_\alpha^-|-1),$$

Figure 2: Pacemaker with an Arrhythmia detector.

where (x_i, y_i) is an ith training data pair, $h \in \mathbb{R}^p$ is a real-valued classifier, $e \in \mathbb{R}^{|\hat{D}_\alpha|}$ denotes a scaled classification error, $z \in \{0,1\}^{|\hat{D}_\alpha|}$ is a vector that indicates misclassification of each training data pair, λ is a regularization constant, set to zero, and δ is a sufficiently large positive constant, where $\delta = 10^3$. $\mathbf{1}_+$ and $\mathbf{1}_-$ represent vectors where a jth element is filled with one if $y_j = +1$ and $y_j = -1$, respectively, and zeros elsewhere. We note that the 0-1 loss linear classification algorithm is formulated in the same way to the above MILP except for the last two constraints (See Section 3 in [40]), related to the majority constraint, and we adopt a standard SVMs formulation [15] without a regularization term for fair comparison. Theoretically, the performance of the 0-1 loss linear classification algorithm is as good as that of the convex loss linear classification algorithms [5]. If there are no attack and no error, the 0-1 loss linear classification algorithm is same as the majority 0-1 loss linear classification algorithm since the last two constraints of MILP are not activated if there are no attacks and no error.

In experiments, we consider two types of attacks: a *point attack* and an *overlap attack*, which are concrete instances of a BFA. The point attack is an attack that manipulates a single feature vector to be located far from the training data as illustrated in Figure 3. The attacked single feature vector is chosen and tampered as follows. Let $\alpha^+ = 1$, and $\bar{x}^+$ and $\bar{x}^-$ be the mean of positive and negative feature vectors, respectively. Any positive feature vector is chosen and replaced to a scaled vector σx where the scaled vector is on the half-line from $\bar{x}^+$ to the direction of $\bar{x}^- - \bar{x}^+$, and the scale value σ is a sufficiently large scalar.

The overlap attack is an attack that manipulates positive and/or negative feature vectors to be overlapped negative and/or positive feature vectors, respectively, as illustrated in Figure 3. The overlap attack is briefly described as follows: when $\alpha = (\alpha^+, \alpha^-)$, α^+ and α^- number of positive and negative feature vectors are randomly chosen for tampering, respectively. The chosen positive and negative feature vectors are randomly overlapped to negative and positive un-attacked feature vectors, respectively. These steps are repeated until a target classification algorithm achieves a maximum desired resilience value $V(\cdot)$.

Synthetic data. In Figure 3, the classification results of each linear classification algorithm, such as SVMs, the 0-1 loss linear classification, and the majority 0-1 loss linear classification, are illustrated with different types of an attack. The original training data without attacks is randomly drawn from two Gaussian distributions, as illustrated in the first column and the first row, where $|\hat{D}^+| = 20$ and $|\hat{D}^-| = 80$. When there is no attack (the first row in Figure 3), all three algorithms correctly classify training data. If there is a point attack (the second row in Figure 3), only SVMs algorithm is affected by the attack, outputting a classifier

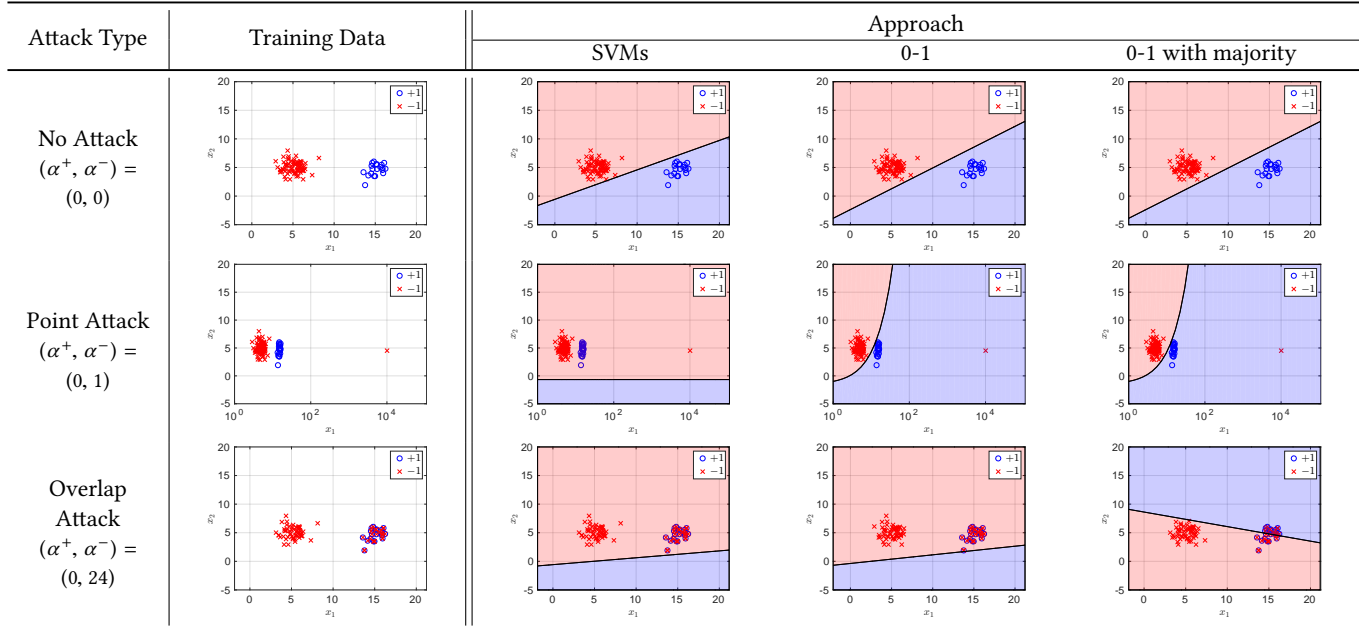| Attack Type	Training Data	Approach		
		SVMs	0-1	0-1 with majority
No Attack $(\alpha^+, \alpha^-) =$ (0, 0)				
Point Attack $(\alpha^+, \alpha^-) =$ (0, 1)				
Overlap Attack $(\alpha^+, \alpha^-) =$ (0, 24)				

Figure 3: The resilience of each linear classification algorithm under the specified attacks. The blue or red mark represents a positive or negative feature vector, respectively. The feature vector in the blue or red region is classified as positive or negative, respectively.

Attack Type	Approach		
	SVMs	0-1	0-1 with majority
No Attack $(\alpha^+, \alpha^-) = (0, 0)$	0.0	0.0	0.0
Point Attack $(\alpha^+, \alpha^-) = (0, 1)$	**1.0**	0.0270	0.0270
Overlap Attack $(\alpha^+, \alpha^-) = (16, 21)$	**1.0**	**1.0**	0.5946

Table 3: The resilience metric $V(\cdot)$ of each linear classification algorithm for Arrhythmia detection [24] under the specified attacks.
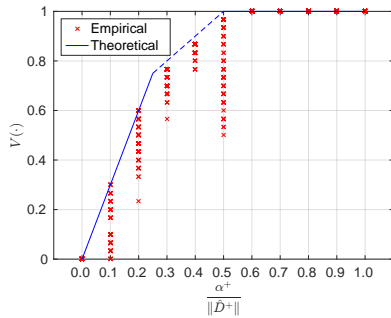

Figure 4: The degree of the resilience of $P_{\mathcal{M}, \ell_{01}}$ in the resilience metric $V(\cdot)$ with respect to the ability of an attacker. The blue solid and dashed line represents the theoretical robustness bound (Equation (12)) and the red cross means the empirically evaluated feasible resilience. Assume $|\hat{D}^+| = |\hat{D}^-|$ and $\alpha^+ = \alpha^-$.

that misclassifies all positive feature vectors of un-attacked training data. When an overlap attack (the third row in Figure 3) is applied, where $\alpha^- = 24$, both SVMs and the 0-1 loss linear classification output classifiers that misclassifies all positive feature vectors of un-attacked training data while the majority 0-1 loss classification algorithm still correctly classifies the portion of the positive feature vectors of un-attacked training data, showing that the majority 0-1 loss classification algorithm is more resilient than others.

Moreover, using the synthetic data, the theoretical worst-case resilience bound (Equation (12)) of the majority 0-1 loss linear classification is experimentally shown in Figure 4. The blue line represents the theoretical worst-case resilience bound. Red points are the resilience $V(\cdot)$ over the corresponding α. Specifically, 100 different $\hat{D}$ are randomly generated, where $\hat{D}^+$ and $\hat{D}^-$ are drawn from two Gaussian distributions of positive and negative labels, respectively. For each $\hat{D}$ and for each α^+, which ranges from 0 to the total number of positive feature vectors, an attacker moves α^+ number of positive feature vectors beyond the negative features in 100 different ways to obtain $\hat{D}_\alpha$ so that positive and negative feature vectors cannot be linearly separable. By taking the maximum of $V(\cdot)$ for 100 different $\hat{D}$ and 100 different $\hat{D}_\alpha$, the resilience $V(\cdot)$ is obtained for each α^+, which is represented in a red cross. In Figure 4, the red crosses do not excess the theoretical bound and the increasing trend follows the bound.

Medical data. We evaluated the resilience of traditional linear classification algorithms and the proposed algorithm using arrhythmia dataset. The arrhythmia, a.k.a irregular heartbeat, is a condition of the heart in which the heartbeat is irregular. An arrhythmia detector cooperated with logs from pacemaker can reduce stroke and death rate [21]. To design such a detector, electrocardiogram (ECG) training data can be collected from logs of the pacemaker (Figure 2)

whether ECG data is normal or abnormal (*e.g.,* atrial fibrillation or sinus tachycardia). But, if the pacemaker is vulnerable, the training data can be tampered to hinder to detect arrhythmia, possibly leading to death.

In Table 3, we have compared the resilience $V(\cdot)$ of each algorithm on real medical dataset. Arrhythmia dataset [24], which can be found at the UCI machine learning repository [32], is used for evaluating the resilience of each algorithm. The Arrhythmia dataset is preprocessed as follows. Due to the computational limitation to solve the MILP, we use 20 percent of training data (*i.e.,* $|\hat{D}^+|$ = 37 and $|\hat{D}^-|$ = 49) and select features from 40th and 99th for training classifiers. We've obtained the same results as illustrated with synthetic data. SVMs algorithm outputs a classifier that misclassifies all positive or negative feature vectors under both a point attack and an overlap attack. 0-1 loss linear classification does not affect on a point attack but outputs a classifier that misclassifies all positive or negative feature vectors when an overlap attack is applied by tampering 43.2 and 42.8 percent of positive and negative feature vector, respectively. However, the majority 0-1 loss linear classification still correctly classifies the portion of positive and negative feature vectors even though about 43 percent of training data were tampered. We emphasize that $V(\cdot)$ values in Table 3 are not prediction results, but they are evaluated over training data, making $V(\cdot) = 0$ possible. However, a higher $V(\cdot)$ value implies higher prediction error.

Comparison with [28]. Kearns and Li's paper [28] analyzes a binary classification problem under the malicious error (ME) model, but our paper analyzes a binary linear classification problem under a BFA, which is a general case of the ME model. Here, we compare each paper's result by providing an example. Assume a binary linear classification problem under the ME model, where $|\hat{D}^+| = |\hat{D}^-| = 50$ and $\alpha^+ = \alpha^- = 10$. Kearns and Li's paper states that if a designer wants to have the expected accuracy of 0.9, then $\frac{\alpha^+}{|\hat{D}^+|} = \frac{\alpha^-}{|\hat{D}^-|} < \frac{0.1}{1+0.1} \approx 0.091$ regardless of a classification algorithm. This means at most 0.091 percent of training data can be tampered to guarantee the expected accuracy. However, this does not state anything on the expected accuracy when $\frac{\alpha^+}{|\hat{D}^+|} \geq 0.091$. In comparison to this, our paper implies that, in the case of the majority 0-1 linear classification algorithm, $g(|\hat{D}|, (\alpha^+, \alpha^-)) = 0.6$. This means that a designer can expect the accuracy on training data that is at least $1 - 0.6 = 0.4$. This further implies the expected minimum accuracy can be approximately 0.4 when $\frac{\alpha^+}{|\hat{D}^+|} = 0.2 \geq 0.091$. We note that the connection between the accuracy on training data (*i.e.,* the performance measure of this paper) and the expected accuracy (*i.e.,* the performance measure of traditional classification) can be found in Section 1 in [40].

8 CONCLUSIONS

In particularly, the incorrect decisions on CPS directly affect on a physical environment, so learning techniques under training data attacks should be scrutinized. Toward the goal of resilient machine learning, we propose a resilience metric for the analysis and design of a resilient classification algorithm under training data attacks. Traditional algorithms, such as convex loss linear classification algorithms and the 0-1 loss linear classification algorithm, are proved to be resilient under restricted conditions. However, the proposed

0-1 loss linear classification with a majority constraint is more resilient than others, and it is the maximally resilient algorithm among linear classification algorithms. The worst-case resilience bound of the proposed algorithm is then provided, suggesting how resilient the algorithm is under training data attacks.

Countermeasures. The resilience analysis on different linear classification algorithms provides us clues for countermeasures on training data attacks. Here, we briefly discuss a possible direction for countermeasures and its challenges. In general, additional algorithms can be considered to eliminate the worst-case situations in the analysis of each classification algorithm. For example, to defend against the point attack on SVMs, it might be considered to add a preprocessing step that saturates large values in training data. Specifically, if a designer knows the minimum and maximum range of features, then range can saturate the large values that contribute to the point attack. However, this might not be an effective countermeasure since the range of features is not known in general and the point attack can be conducted after the preprocessing step, not before it.

Here, we emphasize that our analysis, which is purposely focused on a classification algorithm exclusively, helps to devise countermeasures: combining a classification algorithm with a preprocessing step or using a complex classification algorithm (*e.g.,* hierarchical approach and neural networks). We believe that the advanced algorithms work better under the training data attack in general and our analysis on the simple algorithms (*e.g.,* SVMs) can be a building block for analyzing and devising advanced algorithms.

Future works. As a future work, the following issues are worth being considered. A more practical mixed integer linear program can speed up computational time (*e.g.,* [37]) and it would be promising to design and analyze multiple algorithms in tandem (one to monitor the data, one to learn a classifier). It is also worth incorporating bounded noise error and designing error on $\mathcal{H}$ in analysis, and extending to non-linear and multiclass classification problem. Finally, to devise countermeasures, it would be promising to consider resilient algorithms that estimate attacker capabilities or model prior knowledge on attackers.

ACKNOWLEDGMENTS

REFERENCES

[1] 2016. FBI probing after hackers cripple computer systems at major hospital chain. (Mar. 2016). http://www.cbsnews.com/news/fbi-probing-after-hackers-cripple-computer-systems-at-major-hospital-chain-medstar-health

[2] 2016. R7-2016-07: Multiple Vulnerabilities in Animas OneTouch Ping Insulin Pump. (Oct. 2016).

[3] 2016. U.S. hospitals are getting hit by hackers. (Mar. 2016). http://money.cnn.com/2016/03/23/technology/hospital-ransomware/

[4] Dana Angluin and Philip Laird. 1988. Learning from noisy examples. *Machine Learning* 2, 4 (1988), 343–370.

[5] Peter L Bartlett, Michael I Jordan, and Jon D McAuliffe. 2006. Convexity, classification, and risk bounds. *J. Amer. Statist. Assoc.* 101, 473 (2006), 138–156.

[6] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Srndic, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. 2013. Evasion attacks against machine learning at test time. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer.

[7] Battista Biggio, Blaine Nelson, and Pavel Laskov. 2012. Poisoning Attacks against Support Vector Machines. In *Proceedings of the 29th International Conference on Machine Learning (ICML-12)*.

[8] Tamara Bonaci, Junjie Yan, Jeffrey Herron, Tadayoshi Kohno, and Howard Jay Chizeck. 2015. Experimental analysis of denial-of-service attacks on teleoperated robotic systems. In *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems*. ACM, 11–20.

[9] Michael Brückner and Tobias Scheffer. 2011. Stackelberg games for adversarial prediction problems. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 547–555.

[10] Alvaro A Cárdenas, Saurabh Amin, and Shankar Sastry. 2008. Research Challenges for the Security of Control Systems.. In *HotSec*.

[11] Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, Stefan Savage, Karl Koscher, Alexei Czeskis, Franziska Roesner, Tadayoshi Kohno, and others. 2011. Comprehensive Experimental Analyses of Automotive Attack Surfaces.. In *USENIX Security Symposium*. San Francisco.

[12] Chenyi Chen, Ari Seff, Alain Kornhauser, and Jianxiong Xiao. 2015. Deepdriving: Learning affordance for direct perception in autonomous driving. In *Proceedings of the IEEE International Conference on Computer Vision*. 2722–2730.

[13] Sanjian Chen, Lu Feng, Michael R Rickels, Amy Peleckis, Oleg Sokolsky, and Insup Lee. 2015. A Data-Driven Behavior Modeling and Analysis Framework for Diabetic Patients on Insulin Pumps. In *Healthcare Informatics (ICHI), 2015 International Conference on*. IEEE, 213–222.

[14] Yudong Chen, Constantine Caramanis, and Shie Mannor. 2013. Robust sparse regression under adversarial corruption. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*. 774–782.

[15] Corinna Cortes and Vladimir Vapnik. 1995. Support-vector networks. *Machine Learning* 20, 3 (1995), 273–297.

[16] Gabriela F Cretu, Angelos Stavrou, Michael E Locasto, Salvatore J Stolfo, and Angelos D Keromytis. 2008. Casting out demons: Sanitizing training data for anomaly sensors. In *2008 IEEE Symposium on Security and Privacy (sp 2008)*. IEEE, 81–95.

[17] Nilesh Dalvi, Pedro Domingos, Sumit Sanghai, Deepak Verma, and others. 2004. Adversarial classification. In *Proceedings of the tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, 99–108.

[18] Hamza Fawzi, Paulo Tabuada, and Suhas Diggavi. 2014. Secure estimation and control for cyber-physical systems under adversarial attacks. *IEEE Trans. Automat. Control* 59, 6 (2014), 1454–1467.

[19] Uriel Feige, Yishay Mansour, and Robert Schapire. 2015. Learning and inference in the presence of corrupted inputs. In *Proceedings of The 28th Conference on Learning Theory*.

[20] Jiashi Feng, Huan Xu, Shie Mannor, and Shuicheng Yan. 2014. Robust logistic regression and classification. In *Advances in Neural Information Processing Systems*. 253–261.

[21] Taya V Glotzer, Anne S Hellkamp, John Zimmerman, Michael O Sweeney, Raymond Yee, Roger Marinchak, James Cook, Alexander Paraschos, John Love, Glauco Radoslovich, and others. 2003. Atrial high rate episodes detected by pacemaker diagnostics predict death and stroke report of the atrial diagnostics ancillary study of the MOde Selection Trial (MOST). *Circulation* 107, 12 (2003), 1614–1619.

[22] Sally A. Goldman and Robert H. Sloan. 1995. Can pac learning algorithms tolerate random attribute noise? *Algorithmica* 14, 1 (1995), 70–84.

[23] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. 2014. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572* (2014).

[24] H Altay Guvenir, Burak Acar, Gulsen Demiroz, and Ayhan Cekin. 1997. A supervised machine learning algorithm for arrhythmia analysis. In *Computers in Cardiology 1997*. IEEE, 433–436.

[25] Raia Hadsell, Pierre Sermanet, Jan Ben, Ayse Erkan, Marco Scoffier, Koray Kavukcuoglu, Urs Muller, and Yann LeCun. 2009. Learning long-range vision for autonomous off-road driving. *Journal of Field Robotics* 26, 2 (2009), 120–144.

[26] Todd E Humphreys, Brent M Ledvina, Mark L Psiaki, Brady W OHanlon, and Paul M Kintner Jr. 2008. Assessing the spoofing threat: Development of a portable GPS civilian spoofer. In *Proceedings of the ION GNSS International Technical Meeting of The Satellite Division*.

[27] Alex Kantchelian, JD Tygar, and Anthony D Joseph. 2015. Evasion and Hardening of Tree Ensemble Classifiers. *arXiv preprint arXiv:1509.07892* (2015).

[28] Michael Kearns and Ming Li. 1993. Learning in the presence of malicious errors. *SIAM J. Comput.* 22, 4 (1993), 807–837.

[29] Karl Koscher, Alexei Czeskis, Franziska Roesner, Shwetak Patel, Tadayoshi Kohno, Stephen Checkoway, Damon McCoy, Brian Kantor, Danny Anderson, Hovav Shacham, and others. 2010. Experimental security analysis of a modern automobile. In *2010 IEEE Symposium on Security and Privacy*. IEEE, 447–462.

[30] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems*. 1097–1105.

[31] Gert RG Lanckriet, Laurent El Ghaoui, Chiranjib Bhattacharyya, and Michael I Jordan. 2002. A robust minimax approach to classification. *Journal of Machine Learning Research* 3, Dec (2002), 555–582.

[32] M. Lichman. 2013. UCI Machine Learning Repository. (2013). http://archive.ics.uci.edu/ml

[33] Achin Jain Madhur Behl and Rahul Mangharam. 2016. Data-Driven Modeling, Control and Tools for Cyber-Physical Energy Systems. *ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS)* (apr 2016).

[34] Shike Mei and Xiaojin Zhu. 2015. Using Machine Teaching to Identify Optimal Training-Set Attacks on Machine Learners.. In *AAAI*. 2871–2877.

[35] Yilin Mo and Bruno Sinopoli. 2010. False data injection attacks in control systems. In *Preprints of the 1st workshop on Secure Control Systems*. 1–6.

[36] Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. 2013. Learning with noisy labels. In *Advances in Neural Information Processing Systems*. 1196–1204.

[37] Tan Nguyen and Scott Sanner. 2013. Algorithms for Direct 0–1 Loss Optimization in Binary Classification. In *Proceedings of the 30th International Conference on Machine Learning (ICML-13)*. 1085–1093.

[38] Miroslav Pajic, James Weimer, Nicola Bezzo, Paulo Tabuada, Oleg Sokolsky, Insup Lee, and George J Pappas. 2014. Robustness of attack-resilient state estimators. In *ICCPS'14: ACM/IEEE 5th International Conference on Cyber-Physical Systems (with CPS Week 2014)*. IEEE Computer Society, 163–174.

[39] Kaveh Paridari, Alie El-Din Mady, Silvio La Porta, Rohan Chabukswar, Jacobo Blanco, André Teixeira, Henrik Sandberg, and Menouer Boubekeur. 2016. Cyber-Physical-Security Framework for Building Energy Management System. In *2016 ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS)*. 1–9.

[40] Sangdon Park, James Weimer, and Insup Lee. 2017. *Resilient Linear Classification: An Approach to Deal with Attacks on Training Data Supplementary Material*. Technical Report MS-CIS-17-01. University of Pennsylvania.

[41] Fabio Pasqualetti, Florian Dörfler, and Francesco Bullo. 2013. Attack detection and identification in cyber-physical systems. *IEEE Trans. Automat. Control* 58, 11 (2013), 2715–2729.

[42] Young-Woo Seo, Junsung Kim, and Ragunathan Rajkumar. 2014. Predicting dynamic computational workload of a self-driving car. In *2014 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 3030–3035.

[43] Yasser Shoukry, Paul Martin, Paulo Tabuada, and Mani Srivastava. 2013. Non-invasive spoofing attacks for anti-lock braking systems. In *International Workshop on Cryptographic Hardware and Embedded Systems*. Springer, 55–72.

[44] Robert Sloan. 1988. Types of noise in data for concept learning. In *Proceedings of the first annual Workshop on Computational Learning Theory*. Morgan Kaufmann Publishers Inc., 91–96.

[45] Yunmok Son, Hocheol Shin, Dongkwan Kim, Youngseok Park, Juhwan Noh, Kibum Choi, Jungwoo Choi, and Yongdae Kim. 2015. Rocking drones with intentional sound noise on gyroscopic sensors. In *24th USENIX Security Symposium (USENIX Security 15)*. 881–896.

[46] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2014. Intriguing properties of neural networks. In *International Conference on Learning Representations*. http://arxiv.org/abs/1312.6199

[47] André Teixeira, Saurabh Amin, Henrik Sandberg, Karl H Johansson, and Shankar S Sastry. 2010. Cyber security analysis of state estimators in electric power systems. In *49th IEEE conference on decision and control (CDC)*. IEEE, 5991–5998.

[48] Jorge Valenzuela, Jianhui Wang, and Nancy Bissinger. 2013. Real-time intrusion detection in power system operations. *IEEE Transactions on Power Systems* 28, 2 (2013), 1052–1062.

[49] Harry L Van Trees. 2004. *Detection, estimation, and modulation theory*. John Wiley & Sons.

[50] Vladimir N Vapnik. 1999. An overview of statistical learning theory. *IEEE Transactions on Neural Networks* 10, 5 (1999), 988–999.

[51] Yichao Wu and Yufeng Liu. 2012. Robust truncated hinge loss support vector machines. *J. Amer. Statist. Assoc.* (2012).

[52] Chenyang Yuan, Jérôme Thai, and Alexandre M Bayen. 2016. ZUbers against ZLyfts Apocalypse: An Analysis Framework for DoS Attacks on Mobility-as-a-Service Systems. In *2016 ACM/IEEE 7th International Conference on Cyber-Physical Systems (ICCPS)*. IEEE, 1–10.

[53] Tong Zhang. 2004. Solving large scale linear prediction problems using stochastic gradient descent algorithms. In *Proceedings of the 21st International Conference on Machine Learning*. ACM, 116.