# A Cautionary Tale: On the Role of Reference Data in Empirical Privacy Defenses

Caelin G. Kaplan
SAP Labs France, Inria, Université Côte d'Azur
caelin.kaplan@sap.com

Chuan Xu
Univ. Côte d'Azur, Inria, CNRS, I3S
chuan.xu@inria.fr

Othmane Marfoq
Inria, Université Côte d'Azur, Accenture Labs
othmane.marfoq@inria.fr

Giovanni Neglia
Inria, Univ. Côte d'Azur
giovanni.neglia@inria.fr

Anderson Santana de Oliveira
SAP Labs France
anderson.santana.de.oliveira@sap.com

## ABSTRACT

Within the realm of privacy-preserving machine learning, empirical privacy defenses have been proposed as a solution to achieve satisfactory levels of training data privacy without a significant drop in model utility. Most existing defenses against membership inference attacks assume access to reference data, defined as an additional dataset coming from the same (or a similar) underlying distribution as training data. Despite the common use of reference data, previous works are notably reticent about defining and evaluating reference data privacy. As gains in model utility and/or training data privacy may come at the expense of reference data privacy, it is essential that all three aspects are duly considered. In this paper, we conduct the first comprehensive analysis of empirical privacy defenses. First, we examine the availability of reference data and its privacy treatment in previous works and demonstrate its necessity for fairly comparing defenses. Second, we propose a baseline defense that enables the utility-privacy tradeoff with respect to both training and reference data to be easily understood. Our method is formulated as an empirical risk minimization with a constraint on the generalization error, which, in practice, can be evaluated as a weighted empirical risk minimization (WERM) over the training and reference datasets. Although we conceived of WERM as a simple baseline, our experiments show that, surprisingly, it outperforms the most well-studied and current state-of-the-art empirical privacy defenses using reference data for nearly all relative privacy levels of reference and training data. Our investigation also reveals that these existing methods are unable to trade off reference data privacy for model utility and/or training data privacy, and thus fail to operate outside of the high reference data privacy case. Overall, our work highlights the need for a proper evaluation of the triad "model utility / training data privacy / reference data privacy" when comparing privacy defenses.

## KEYWORDS

privacy-preserving machine learning, empirical privacy defenses, statistical learning

## 1 INTRODUCTION

Data-driven applications, often using machine learning models, are proliferating throughout industry and society. Consequently, concerns about the use of data relating to individual persons has led to to a growing body of legislation, most notably the European Union's General Data Protection Regulation (GDPR) [36]. According to the GDPR principle of data minimization, it is necessary to reduce the degree to which data can be connected to individuals, even when that data is used for the purposes of training a statistical model [37]. It has therefore become important to ensure that a machine learning model is not leaking private information about its training data.

Membership inference attacks (MIAs), which seek to discern whether or not a given data point has been used during training, have emerged as a key evaluation tool for empirically measuring a machine learning model's privacy leakage [41]. Indeed, inferring training dataset membership can be thought of as the most fundamental privacy violation. Although other attacks exist, such as model inversion [13], property inference [15], dataset reconstruction [38], and model extraction [18, 22, 46], they all require a stronger adversary than is necessary for MIAs.

Many methods have been proposed to defend against MIAs. The use of differential privacy [11] (DP) has emerged as a leading candidate for two reasons. First, it provides mathematically rigorous guarantees that upper-bound the influence a given data point can exert on the final machine learning model. Second, it is straightforward to integrate DP into a machine learning model's training procedure with algorithms such as differentially private gradient descent (DP-SGD) [1] or PATE [33]. Despite the many advantages associated with DP, there are several key drawbacks that include: the significant degradation of model utility when using DP during training [45], even more severe for underrepresented groups [2, 10, 14, 47], and the difficulty of translating DP's theoretical privacy guarantees to real-world privacy leakage [4, 5, 32, 50].

To address these issues, empirical privacy defenses (i.e., without theoretical privacy guarantees) have been developed to protect the privacy of training data against MIAs. Existing empirical privacy defenses can be categorized by their method of protecting the training data (e.g., regularization [26, 30], confidence-vector masking [20, 49], knowledge distillation [44]). Alternatively, one can group defenses by whether they use only the private training data [44] or require access to reference data [20, 26, 30, 40, 48, 49], defined as additional data from the same (or a similar) underlying

*Proceedings on Privacy Enhancing Technologies 2024(1), 525–548*
© 2024 Copyright held by the owner/author(s).
https://doi.org/10.56553/popets-2024-0031

distribution [30]. The two most prominent differentially private defenses can also be distinguished according to this distinction, where PATE [33] requires access to (unlabeled) reference data but DP-SGD [1] does not.

There are several problems with the current evaluation strategy of empirical privacy defenses. First, today's best practice is to produce a utility-privacy curve that compares a model's classification accuracy with its training data privacy for different values of a given defense parameter (e.g., different regularization term values). Although this approach appears valid in the general case, assuming access to reference data makes the situation more complicated. This additional dataset may have its own privacy requirements [26, 40, 49], which we discuss in detail in Section 2.4. As gains in model utility and/or training data privacy usually come at the expense of reference data privacy, it is only possible to meaningfully compare defenses when the *relative* level of privacy considerations between these two datasets is made explicit. To demonstrate this issue, we present a concrete example in Figure 1, where "AdvReg" corresponds to adversarial regularization [30], the most well-studied empirical privacy defense, and "AdvReg-RT" corresponds to an alternative version of the defense that we propose (defined in Section 4.3.2). Looking only at the utility-privacy curves[1] with respect to training data, it seems that AdvReg-RT is strictly better than AdvReg: for any given value value of test accuracy, AdvReg-RT is able to produce a model that yields a lower MIA accuracy on the training data. However, when the utility-privacy curves are examined with respect to both training and reference data, one cannot determine the better method without knowing their relative privacy considerations.

A second problem with the current evaluation methodology is the lack of a well-understood and simple baseline. The literature contains several examples where proposed empirical privacy defenses have been later shown to leak significantly more training data privacy than originally reported and sometimes to even perform worse than simpler defenses [8, 26, 42]. A well-established baseline could have provided more accurate expectations about the ability of these defenses.

*Thus, there is a strong need for the development of a baseline designed to operate in the same assumption setting as the vast majority of existing empirical privacy defenses and for an evaluation that takes reference data privacy into account.*

**Contributions.** We introduce the notion of a training-reference data privacy tradeoff and conduct the first comprehensive investigation into how empirical privacy defenses perform with respect to all three relevant metrics: model utility, training data privacy leakage, and reference data privacy leakage. Given this evaluation setting, we propose a well-motivated baseline that introduces the privacy requirement as a constraint on the generalization capability [39] of the learned model. Our formulation leads to a convenient weighted empirical risk minimization (WERM), where the training and reference data can be weighted according to the relative privacy level of the two datasets. We prove that WERM enjoys theoretical guarantees both on the resulting model utility and the relative privacy level of training and reference data.

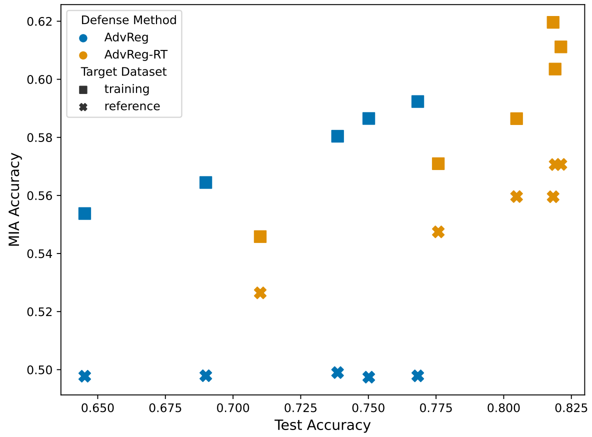

**Figure 1: Tradeoff between a defended classifier's prediction accuracy on test data (i.e., its model utility), MIA accuracy on training data (i.e., training data privacy leakage), MIA accuracy on reference data (i.e., reference data privacy leakage) for Purchase100 dataset. The key takeaway is that one cannot solely look at training data privacy leakage when evaluating the utility-privacy tradeoff of a given defense method.**

Our experimental results show that, surprisingly, WERM outperforms state-of-the-art empirical privacy defenses using reference data in nearly all training and reference data relative privacy regimes, including the case of public reference data. Additionally, we demonstrate that existing methods are only capable of extracting limited information from reference data during training and thus fail to effectively trade off reference data privacy for model utility and/or training data privacy. In particular, the mechanisms provided by these defenses to control the utility-privacy tradeoff with respect to the three aforementioned factors do not function as expected, since they are only able to operate in the case where reference data privacy is highly valued. By contrast, WERM is interpretable, straightforward to train, and highly effective. These traits enable it to serve as a baseline for evaluating future empirical privacy defenses using reference data. Importantly, comparing against our method requires selecting relative weights for the loss on the training data and the reference data, which makes explicit the underlying assumption about their relative privacy.

The remainder of the paper is organized as follows. In Section 2, we provide the background knowledge necessary to understand the domain of empirical privacy defenses. In Section 3, we present WERM and analyze its theoretical properties. In Section 4, we conduct a comprehensive set of experiments to evaluate our baseline in comparison to existing state-of-the-art methods. In Section 6, we conclude our paper and discuss future work.

# 2 BACKGROUND

## 2.1 Machine Learning Notation

In standard classification tasks, the goal is to learn a function $f_\theta$ that maps a set of input examples $x \in \mathcal{X}$ to a k-class probability distribution over a set of classes $\mathcal{Y} = \{1, 2, \ldots, k\}$. The function's

[1] For the AdvReg and AdvReg-RT, the curves are obtained by changing the relative importance of the classification loss and the attacker loss [30], i.e., the value of the parameter $\lambda$ in (8).

output, $f_\theta(x)$, is a vector, known as the confidence-vector, where each entry, $f_\theta(x)_y$, represents the model's confidence about input $x$ belonging to class $y$. The model training entity has access to a training dataset of $n$ examples, $D_T = \{(x_1, y_1), \dots, (x_{N_T}, y_{N_T})\}$, which have been drawn from an unknown underlying distribution $\mathbb{D}$.

Although we only have access to $D_T$, for a learned function to make useful predictions, it must perform well on unseen data also coming from $\mathbb{D}$ (i.e., test data). More formally, the task of training a model entails finding the vector of parameters, $\theta \in \Theta$, that minimize the expected risk (expected loss) $L_{\mathbb{D}}$:

$$\min_{\theta\in\Theta} L_{\mathbb{D}}(f_\theta) = \min_{\theta\in\Theta} \mathop{\mathbb{E}}_{(x,y)\sim\mathbb{D}} [\ell(f_\theta, (x, y))] \tag{1}$$

where $\ell(\cdot,\cdot)$ is a loss function. In supervised classification tasks, the loss function is often chosen to be the cross-entropy loss, $\ell(f_\theta, (x, y)) = -\sum_{y'\in\mathcal{Y}} \mathbb{1}_{y'=y} \log(f_\theta(x)_y)$. As we do not have access to $\mathbb{D}$, we cannot directly minimize the expected risk. Therefore, we instead minimize the loss over our training data, $D_T$, which we define as the empirical risk (training loss) $L_{D_T}$:

$$\min_{\theta\in\Theta} L_{D_T}(f_\theta) = \min_{\theta\in\Theta} \frac{1}{N_T} \sum_{i=1}^{N_T} \ell(f_\theta, (x_i, y_i)) \, . \tag{2}$$

The empirical risk minimization (ERM) in (2) is often solved using gradient descent methods [25]. Given a satisfactory set of model parameters, $\theta_s$, the generalization error, also referred to as the generalization gap, is defined as:

$$\text{Generalization Error} = L_{\mathbb{D}}(f_{\theta_s}) - L_{D_T}(f_{\theta_s}). \tag{3}$$

The generalization error serves to quantify the difference between the training loss and expected loss. The framework of statistical learning theory [39] enables the derivation of theoretical bounds for the generalization error, which we use to provide guarantees for our proposed method.

## 2.2 Membership Inference Attacks

*2.2.1 Attack Setting.* In the most generic case, a MIA operates in a setting where there exists:

- A training dataset, $D_T$, (drawn from the distribution $\mathbb{D}$), whose privacy should be protected
- A machine learning model, $f_\theta$, which will be referred to as the target model, that is trained on $D_T$ and possibly additional data sources (e.g., reference data)
- An adversary, $\mathcal{A}$, who seeks to infer whether a target data point in a set $D^{\text{adv}}$ belongs to $D_T$

*2.2.2 Evaluation Setting.* The dataset $D^{\text{adv}}$, used for evaluating the performance of most previous attacks [30, 41, 42, 52], is constructed such that it contains half of the training data, denoted as $D^{\text{adv}}_T$, and an equal size sample of non-training data from the same underlying distribution, denoted as $D^{\text{adv}}_{\overline{T}}$. Accuracy is the standard metric used for evaluation, although recent work by Carlini et al. [5] proposes an alternative.

We use the notation $\mathcal{A}(x, y)$ to define the binary output of a generic MIA, which codes members as 1 and non-members as 0. The accuracy of an attack against $D^{\text{adv}}$ can thus be calculated as:

$$\frac{\sum_{(x_i,y_i)\in D^{\text{adv}}_T} \mathcal{A}(x_i, y_i) + \sum_{(x_i,y_i)\in D^{\text{adv}}_{\overline{T}}} (1 - \mathcal{A}(x_i, y_i))}{\left|D^{\text{adv}}_T\right| + \left|D^{\text{adv}}_{\overline{T}}\right|} \tag{4}$$

*2.2.3 Threat Model.* The potential for adversaries to perform effective membership inference increases with every additional piece of information they can access. Therefore, it is important to clearly articulate the assumptions underlying each potential attack. To the best of our knowledge, all known attacks proposed in the literature rely on at least one of the following four fundamental assumptions about the adversary's knowledge:

1. Knowledge of the ground-truth label for a target data point.
2. Access to either the largest confidence value or the entire confidence-vector when evaluated on a target data point, as opposed to merely the predicted label.
3. Access to a dataset drawn from the same distribution as the training data (often referred to as population data [50]).[2]
4. Access to either a portion or all of the ground-truth training data, excluding the target data point whose membership the adversary wants to infer.

Adversaries possessing access to either population data (Assumption 3) and/or ground-truth training data (Assumption 4) are positioned to launch significantly more sophisticated and potent attacks. In Table 5 (in Appendix D.1), we present the different adversary assumptions for some of the most well-known MIAs.

*2.2.4 Existing Membership Inference Attacks.* MIAs can be levied against discriminative [41] and generative [7] machine learning models. One key distinction among MIAs is whether the adversary has access to the inner-workings of the target model, such as weights, gradients, etc. (white-box), or only access to the target model's output (black-box). When evaluating our proposed baseline against state-of-the-art defenses, we follow previous work [20, 30, 44] and assume that the adversary has black-box access to the target model. Therefore, from now on we focus on black-box attacks, and refer the reader to [31] for a comprehensive review of white-box attacks.

The simplest attack is the gap attack [51], which predicts any correctly classified data point as a member and any misclassified data point as a non-member:

$$\mathcal{A}_{\text{gap}}(x, y) = \mathbb{1}\{\operatorname*{argmax}_i f_\theta(x)_i = y\}. \tag{5}$$

The name is derived from the fact that the attack directly exploits the generalization error (gap) described in (3). This attack only requires the assumption that an adversary has access to the ground-truth label.

When the adversary has access to more fine-grained information (e.g., the confidence value associated to the predicted class or the entire confidence-vector), one can conduct a threshold-based attack [42, 51]. Using the confidence value associated to the predicted class as an example, we have:

$$\mathcal{A}_{\text{conf}}(x, y) = \mathbb{1}\{f_\theta(x)_y > \tau\}, \tag{6}$$

[2] Note that the attacker's population data plays a similar role to reference data for the defender, but for clarity we avoid using the same name.

where $\tau$ is a class-independent threshold. Song and Mittal [42] demonstrated that threshold-based MIAs are the most effective among those that do not require access to training/non-training data. Further details regarding the design of the gap attack and extensions of threshold-based MIAs can be found in Appendix D.2.

In Section 4, following the methodology laid out in [42], we assess our proposed defense, Weighted Empirical Risk Minimization (WERM), against a variety of threshold-based MIAs. Additionally, we consider a neural network-based MIA [41], which could be employed by a stronger adversary.

## 2.3 Empirical Privacy Defenses

Among empirical privacy defenses using reference data, the methods based on regularization techniques are the best performing [26]. As WERM belongs to this group, we provide background for this type of defense and refer the reader to [44] for background on defenses using knowledge distillation. The idea of regularization defenses is to achieve a model that has good generalization, such that the distribution of model outputs on training data is similar to the output on unseen test data. Standard approaches to improve regularization, such as early-stopping [6], weight decay [23], and dropout [43], have been observed to improve a model's robustness against a variety of MIAs [41, 42]. Additionally, some regularization terms have been proposed that seek to explicitly protect against attacks, such as adversarial regularization [30] and MMD-based regularization [26].

All empirical privacy defenses using reference data assume that the model training entity has access to training data, $D_T = \{(x_1, y_1), \ldots, (x_{N_T}, y_{N_T})\}$, and reference data, $D_R = \{(x'_1, y'_1), \ldots, (x'_{N_R}, y'_{N_R})\}$, which come from the same (or a similar) underlying distribution and are of size $N_T = |D_T|$ and $N_R = |D_R|$, respectively. The defenses aim to make model predictions on training and reference data sufficiently similar, such that it will be hard for an attacker to distinguish a model's output on training and non-training data. The closer the distributions of training data and reference data, the easier the task for the defense and the smaller the model utility loss.

*2.3.1 Adversarial Regularization.* Adversarial regularization (AdvReg) [30] is a model training framework that is formulated as a min max game, where a classifier, $f_\theta$, is trained to be optimally protected against a MIA model, $h_\phi$. The first component is the loss of the classifier, $f_\theta$, over the training data, i.e., $L_{D_T}(f_\theta)$ as described in (2). The second component is the gain of the attack model:

$$\begin{aligned} G_{D_T,D_R}(f_\theta, h_\phi) = \frac{1}{N_T} \sum_{i=1}^{N_T} \log[h_\phi(x_i, y_i, f_\theta(x_i))] + \\ \frac{1}{N_R} \sum_{i=1}^{N_R} \log[1 - h_\phi(x'_i, y'_i, f_\theta(x'_i))], \end{aligned} \tag{7}$$

where $h_\phi(x, y, f_\theta(x))$ outputs the probability that a given target data point is a member of the training data. The attack model's gain quantifies its ability to predict the training data as members and the reference data as non-members.

The whole optimization problem can be formulated as:

$$\min_{\theta\in\Theta} \max_{\phi\in\Phi} \; L_{D_T}(f_\theta) + \lambda \, G_{D_T,D_R}(f_\theta, h_\phi), \tag{8}$$

where $\lambda$ is the regularization term's weight and serves to trade utility for privacy (a larger $\lambda$ should result in the trained model having greater privacy protection at the cost of decreased utility). The minmax problem described in (8) is solved by alternating some gradient method steps for the minimization and the maximization problem.

*2.3.2 MMD-based Regularization.* Alternatively, in MMD-based regularization (MMD) as proposed in [26], the regularization term may be the Maximum Mean Discrepancy (MMD), leading to the following problem:

$$\min_{\theta\in\Theta} L_{D_T}(f_\theta) + \lambda \cdot \left\| \frac{1}{N_T} \sum_{i=1}^{N_T} \psi(f_\theta(x_i)) - \frac{1}{N_R} \sum_{i=1}^{N_R} \psi(f_\theta(x'_i)) \right\|_{\mathcal{H}} \tag{9}$$

where $\mathcal{H}$ is a universal Reproducing Kernel Hilbert Space (RKHS) and $\psi$ is a function mapping model's outputs to points in $\mathcal{H}$. By solving the problem in (9), the resulting model seeks to simultaneously minimize the empirical risk of the training data and the difference in output of the model on training and reference data in the space $\mathcal{H}$. Traditionally, to calculate the MMD one would find $\psi$ such that it maximizes the distance in $\mathcal{H}$. Instead, to simplify the training process, the authors of [26] select $\psi$ to be a given Gaussian kernel.

## 2.4 Reference Data Overview and Threat Model

The vast majority of empirical privacy defenses in the literature [20, 26, 30, 33, 34, 48, 49] require access to reference data, which is assumed to come from the same (or a similar) underlying distribution as training data. In Section 2.4.1, we discuss the availability of reference data and its level of privacy. In Section 2.4.2, we examine how existing empirical privacy defenses have dealt with the privacy of reference data.

*2.4.1 Reference Data Availability and Privacy.* Although not always called "reference data," the notion of having access to a distinct dataset coming from the same (or a similar) underlying distribution as training data is common throughout many domains of machine learning literature (e.g., the design of MIAs as mentioned in Section 2.2.3). We can divide the examples into cases where reference data is public and cases where reference data is private. In the public reference data setting, large publicly available datasets are routinely employed to pre-train a model which will later be fine-tuned using a private and smaller training dataset [3, 8] or a public dataset can be used for knowledge transfer across heterogeneous models trained on private local datasets in a federated learning scenario [27, 28]. When reference data is public, empirical privacy defenses can use it to augment the privacy of training data, while disregarding concerns about the privacy of the reference data itself.

In the private reference data setting, the availability of reference data may result from model training entities having private datasets that contain certain records with distinct privacy requirements. The "pay-for-privacy" business model enables companies to acquire data from users or consumers at various privacy levels [12]. For example, ISPs are known to provide discounts to their users in exchange for the possibility of exploiting their data for targeted advertisement (possibly powered by a machine learning model) [9], and some mobile phone applications offer a free and a paid version that provides

**Table 1: Comparison of empirical privacy defenses by reference data treatment. In the third column, "relative level unspecified" means the target level of relative privacy requirements between training and reference data is not stated. In the fourth column, "single privacy level" means the reference data privacy leakage is evaluated at a single point on the utility-privacy curve. We use a dashed line (—) to convey that the defense either does not use reference data or does not need to evaluate reference data privacy leakage.**

| Defense | Category | Reference Data Privacy Setting | Reference Data Privacy Evaluation |
|---|---|---|---|
| Adversarial Regularization [30] | regularization | not mentioned | no evaluation |
| MemGuard [20] | confidence-vector masking | not mentioned | no evaluation |
| Model Pruning [48] | knowledge distillation | not mentioned | no evaluation |
| MMD-based Regularization [26] | regularization | private (relative level unspecified) | yes (single privacy level) |
| Distillation for Membership Privacy [40] | knowledge distillation | private (relative level unspecified) | yes (single privacy level) |
| Prediction Purification [49] | confidence-vector masking | private (relative level unspecified) | yes (single privacy level) |
| WERM (this paper) | regularization | all possible settings discussed | yes (all privacy levels) |

better privacy protection to users of the paid service [16]. Training and reference data can then correspond to data from users with a different pricing scheme. Different privacy levels may also be due to past data leaks, e.g., due to malicious security breaches or human errors. As will become apparent, in this scenario where a single dataset has two segments with distinct privacy considerations, one can use either the more or less private data segment as reference data to better protect the privacy of the remaining segment (training data). Even in standard machine learning training, such considerations may be a leading factor in choosing how to split the available data into training and validation segments, as they have been shown to each leak different amounts of private information [35]. Finally, we observe that heterogeneity in privacy levels is also implicitly assumed in fog learning [19], where federated learning clients share a part of their local datasets to bring their respective distributions closer to facilitate the training of a common model.

*2.4.2 Reference Data in Empirical Privacy Defenses.* In Table 1, we present seven empirical privacy defenses using reference data: the first six are existing defenses and the seventh is our proposed method, WERM. The existing defenses can be subdivided into two categories based on reference data privacy treatment: private [26, 40, 49] and "not mentioned" [20, 30, 48]. We use the label "not mentioned" to represent works where reference data privacy is neither discussed nor evaluated.[3] Moreover, each of the three works that consider reference data to be private evaluate its privacy leakage at only a single point on the utility-privacy curve and show it to be much smaller than the training data privacy leakage. These results reveal an implicit choice by the authors: reference data privacy is valued more highly than training data privacy.

We do not take a particular stance on the relative privacy of training and reference data, i.e., if the reference data in empirical privacy defenses should be considered more or less private than training data—as shown in Section 4, we evaluate WERM in all possible reference data privacy settings and show that it outperforms state-of-the-art defenses across almost the entire spectrum. Yet, we argue that, without quantifying the relative importance assigned to the three key objectives (model utility, training data privacy, and reference data privacy), we cannot adequately compare the performance of these defenses. For example, in the papers that consider reference data more private than training data, the proposed defenses are still allowing for some reference data privacy leakage to achieve a high model utility and training data privacy protection. Is this the right amount of privacy leakage? Perhaps, one should instead seek to trade much more reference data privacy to improve the other two metrics. Alternatively, if reference data privacy is of the utmost importance, the current leakage may already be unacceptable. Similar considerations hold for the public reference data case: given that reference data privacy is not a concern, are the proposed methods achieving the best possible tradeoff between model utility and training data privacy?

The next section will introduce our method and show how its utility-privacy tradeoffs are amenable to analysis.

## 3 WEIGHTED EMPIRICAL RISK MINIMIZATION

In this section, we introduce our proposed baseline, WERM, and analyze its theoretical properties related to generalization and privacy protection. WERM's design is rooted in the fundamental principles of statistical learning, particularly in the generalization error (3). WERM utilizes a weight term, $w$, which simultaneously regulates the tradeoff between the privacy of training data and reference data, as well as the tradeoff between utility and privacy. Employing tools from differential privacy (DP) [11] and statistical learning theory [39], we derive theoretical bounds that enable us to understand how $w$ and the size of the two datasets impact the relative privacy leakage and the model's utility. Following all related work [20, 26, 30, 40, 48, 49], we consider the two distributions from which $D_T$ and $D_R$ are drawn to be identical. The relative privacy results in Theorem 3.1 do not depend on $D_T$ and $D_R$ coming from the same underlying distribution and the generalization bound in Theorem 3.2 can be extended to the case where the distributions are only similar.

### 3.1 Motivation

Drawing any conclusion about the quality of a defense can only come after comparing it to an interpretable and well-performing

[3] We note that the omission may suggest they implicitly consider the reference data to be public.

baseline. Therefore, our goal is to propose a baseline that makes the training-reference data privacy tradeoff explicit and can operate across the entire range of possible privacy settings. Our method's design originates from the understanding that all black-box MIAs share a common design feature, which is exploiting the difference between a model's output on training and non-training data. What they consider as a model's output may differ (e.g., predicted label, loss, confidence-vector), but the distinguishability of output distributions is the prerequisite for a membership inference vulnerability to exist in the black-box setting. Thus, employing an ideal membership inference defense will result in a defended model that behaves identically when queried with training or non-training data from the same distribution. The design of a defense based on regularization requires a decision about how to define equivalence of output. AdvReg (Section 2.3.1) introduces a regularization term that constrains the difference between a classifier's confidence-vector output on training and reference data based on a learned neural network; MMD (Section 2.3.2) constrains this difference using a Gaussian kernel. Our proposed baseline is motivated by the fact that a smaller generalization error implies that the empirical loss is closer to the expected loss and, subsequently, the loss observed on any future sample drawn from the same distribution, making it difficult for the adversary to conclude which samples were part of the training data. Thus, WERM addresses the fundamental challenge common to all regularization defenses: learning a classifier whose outputs are indistinguishable between training and reference data. However, its design, rooted in statistical learning principles, results in a unique algorithm. WERM not only exhibits superior performance (Section 4.4) but also provides enhanced interpretability (Section 3.2), simpler configuration (Section 5.1), and reduced computational costs (Section 5.2).

## 3.2 Method

We propose to train a standard ERM using both training and reference data, while constraining the generalization error with respect to each of the datasets. Our problem can be formulated as:

$$\begin{array}{rl} \text{Input:} & D_T \sim \mathbb{D}^{N_T}, D_R \sim \mathbb{D}^{N_R},\ c_T, c_R \in \mathbb{R}^+ \\ \min\limits_{\theta \in \Theta} & L_D(f_\theta) \\ \text{s.t.} & L_{\mathbb{D}}(f_\theta) - L_{D_T}(f_\theta) \leq c_T \\ & L_{\mathbb{D}}(f_\theta) - L_{D_R}(f_\theta) \leq c_R \end{array} \tag{10}$$

where $D = D_T \cup D_R$, $N_T$ and $N_R$ are the respective sizes of training and reference data, $L_D(f_\theta) = \frac{N_T}{N} L_{D_T}(f_\theta) + \frac{N_R}{N} L_{D_R}(f_\theta)$, with $N = N_T + N_R$, and the constants $c_T$ and $c_R$ constrain the generalization error on the training data and on the reference data, respectively. On the basis of our discussion in Section 3.1, smaller values of $c_T$ ($c_R$) correspond to greater privacy protection for training (reference) data. For the purpose of readability, in the rest of this section, we write $L_D(f_\theta)$ as $L_D$ (i.e., the loss over a given dataset is implied to be evaluated for $f_\theta$). Moreover, for simplicity, we consider the case where $N_T = N_R$.

Studying the Lagrangian of problem 10 and introducing the optimal multipliers $\lambda^*$ and $\mu^*$, as detailed in Appendix A.1, we can show that (10) becomes equivalent to the following two problems:

$$\min_{\theta \in \Theta} \left[\frac{1}{2} + \mu^*\right] L_{D_T} + \left[\frac{1}{2} - \mu^*\right] L_{D_R}, \tag{11}$$

$$\min_{\theta \in \Theta} \left[\frac{1}{2} - \lambda^*\right] L_{D_T} + \left[\frac{1}{2} + \lambda^*\right] L_{D_R}, \tag{12}$$

where (11) corresponds to the case when reference data privacy is a stricter constraint (i.e., $c_R < c_T$) and (12) corresponds to the case when training data privacy is a stricter constant (i.e., $c_T < c_R$). In both cases, we obtain a weighted sum of the two empirical risks with a larger weight (i.e., $> 1/2$) given to the dataset with looser privacy constraints. Using equal weights corresponds to equal privacy constraints.

Motivated by this reasoning, we propose the following weighted empirical risk minimization (WERM) as a baseline for privacy defenses using reference data:

$$\min_{\theta \in \Theta} L_D^w(f_\theta) = (1 - w) L_{D_T}(f_\theta) + w L_{D_R}(f_\theta), \text{ for some } w \in [0, 1]. \tag{13}$$

This formulation allows us to simply trade reference data privacy for training data privacy by changing the parameter $w$. Higher (lower) values of $w$ lead to greater privacy protection for training (reference) data. In particular, the privacy of training data and reference data is perfectly protected for $w = 1$ and $w = 0$, respectively, which is the case where the corresponding dataset is not used to compute the defended model. Another benefit of WERM's formulation in (13) is its ability to accommodate multiple datasets, each with a distinct privacy level (up to the limit case where every point is a separate dataset with its own privacy considerations). It is unclear how AdvReg [30] or MMD [26] could be adapted to this scenario. We prove Theorem 3.1 on the relative privacy leakage (Appendix A.3) and Theorem 3.2 on the generalization bound (Appendix A.4) for this generalized case.

Along with its high interpretability, WERM is also a lightweight defense, as its computational cost is equivalent to training an undefended model by minimizing the empirical risk over $N$ samples. This is less computationally expensive than solving AdvReg's minmax problem in (8) and MMD's additional requirement of comparing the distance for unique classes in a batch (see implementation details in Appendix B.2). A detailed comparison of the training time for these defenses (Section 5.2) confirms this intuition.

In the remainder of this section, we provide theoretical guarantees for WERM's relative training-reference data privacy (Section 3.3) and WERM's model utility (Section 3.4).

## 3.3 WERM's Privacy

Our analysis in the previous section led us to qualitatively conclude that increasing (decreasing) the reference data weight, $w$, in WERM results in increased privacy protection for the training (reference) data. Particularly, when the two datasets are the same size and have the same privacy requirements, one should select $w = 1/2$. In this section, we derive more formal privacy guarantees and configuration rules for $w$ considering general dataset sizes.

The formulation of WERM in (13) is not intrinsically differentially private. However, using DP-SGD [1] as the optimization algorithm to solve (13) enables WERM to become a differentially private method. For the purpose of our analysis, we assume this situation in order to employ tools from DP [11] to measure the relative privacy tradeoff between training data and reference data. Consequently, the $\epsilon$ values presented are simply a convenient way to achieve our primary goal of quantifying how the weight term, w, and the size of the two datasets impact WERM's relative privacy level. We emphasize that, while possible, we are not proposing to train WERM with DP-SGD to achieve $\epsilon$-DP privacy guarantees.

DP-SGD works by clipping the gradient values below a certain threshold and adding Gaussian noise to each of them with scale $\sigma$. If properly configured, DP-SGD enjoys $(\epsilon, \delta)$-DP guarantees, i.e., when a single sample of the dataset is changed, the probability of any possible event observable by an attacker changes at most by a multiplicative factor $\exp(\epsilon)$ and by an additive term $\delta$. The larger noise scale $\sigma$, the smaller $\epsilon \geq 0$ and $\delta \in [0, 1)$, and the stronger the privacy guarantees.

Fundamentally, an empirical privacy defense that has access to reference data must make a choice regarding how much of the reference data's privacy should be sacrificed to protect the privacy of the training data. We rely on the $\epsilon$ parameter from DP to quantify the relative privacy of the two datasets. As we will argue after stating our result, in practice, we can consider that the conclusions about the relative privacy hold even if DP-SGD is not used during training.

Theorem 3.1 (Privacy Leakage). *For some overall number of training steps, K, WERM minimized with DP-SGD is:*

$$\Big(O(\epsilon_T), \delta\Big) - DP \text{ w.r.t. the training dataset } (D_T) \tag{14}$$

$$\Big(O(\epsilon_R), \delta\Big) - DP \text{ w.r.t. the reference dataset } (D_R) \tag{15}$$

*where:*

$$\epsilon_T = \epsilon_0 \frac{1-w}{N_T}, \epsilon_R = \epsilon_0 \frac{w}{N_R}$$

$$0 < \epsilon_0 < \min\Big(\frac{N_T}{1-w}, \frac{N_R}{w}\Big), \tag{16}$$

$$\sigma = \alpha\sqrt{K}\sqrt{2\log\frac{1.25}{\delta}}\frac{C}{\epsilon_0}, \tag{17}$$

*and $\sigma$, C, and $\alpha$ are the noise scale, gradient norm bound, and sampling ratio in DP-SGD, respectively.*

The proof of Theorem 3.1 and a detailed description of how we adapt the analysis of DP-SGD from Abadi et al. [1] to be compatible with WERM can be found in Appendix A.3.

It is important to note that the relative privacy of the two datasets, as quantified by the ratio $\epsilon_T/\epsilon_R$ is completely governed by $w$ and the size of the two datasets and independent of $\epsilon_0$. In particular, the training data will be more private if and only if $\frac{1-w}{N_T} < \frac{w}{N_R}$. Specifically, setting the weight of each empirical loss in (13) proportional to the size of its corresponding dataset leads to the same privacy guarantees for samples in both datasets. In the case where $N_T = N_R$, we recover the result we were able to conclude qualitatively in the previous section, i.e., that setting $w = 1/2$ will result in equivalent privacy guarantees for the training and reference data.

The independence of the ratio $\epsilon_T/\epsilon_R$ on $\epsilon_0$ implies that the same value for the relative privacy of the two datasets is achieved if we set $\epsilon_0$ to a very large value (on the order of the dataset size, see (16)) and then use DP-SGD with a negligible noise ($\sigma \approx 0$ in (17)). These considerations justify our experimental results in Section 4.4, where WERM, trained with the usual gradient descent method (i.e., without clipping or adding noise) provides relative privacy guarantees—as measured by the success of MIAs—qualitatively aligned with the conclusions of Theorem 3.1.

## 3.4 WERM's Model Utility

We provide a bound for the expected loss of the model learned through WERM ($f_{\theta_{\text{WERM}}}$) with respect to the smallest possible loss $\min_{\theta\in\Theta} L_{\mathbb{D}}(f_\theta)$.

Theorem 3.2 (Generalization bound). *Under the assumption that the loss function is bounded in the range [0, 1], it follows that:*

$$L_{\mathbb{D}}(f_{\theta_{WERM}}) \leq \min_{\theta\in\Theta} L_{\mathbb{D}}(f_\theta) + 2\sqrt{\frac{VCdim(\Theta)}{N_{eff}}} \cdot \sqrt{\gamma_2 + \log\left(\frac{N}{VCdim(\Theta)}\right)} + \sqrt{\frac{2\ln 2/\delta}{N_{eff}}} \tag{18}$$

*with probability $\geq 1 - \delta$, where:*

$$f_{\theta_{WERM}} = \operatorname*{argmin}_{\theta\in\Theta} L^w_D(f_\theta),$$

$$L^w_D(f_\theta) = (1-w)L_{D_T}(f_\theta) + wL_{D_R}(f_\theta),$$

$$\gamma_2 = \max\Big\{\frac{4}{VCdim(\Theta)}, 1\Big\}, N_{eff} = \left[\frac{(1-w)^2}{N_T} + \frac{w^2}{N_R}\right]^{-1},$$

*$D_T \sim \mathbb{D}^{N_T}$, $D_R \sim \mathbb{D}^{N_R}$, $D = D_T \cup D_R$, $N = |D|$, and VCdim($\Theta$) is the VC-dimension of hypothesis class $F_\Theta = \{f_\theta : \theta \in \Theta\}$.*

The proof of Theorem 3.2 can be found in Appendix A.4. For our purposes, it is important to keep in mind that a smaller bound in (31) implies that the performance of a model learned by WERM will be closer to the performance of the best model in the hypothesis class $F_\Theta$, resulting in higher model utility.

Theorem 3.2 makes clear how the classifier's utility depends directly on $w$. Given a fixed total dataset size and an already selected model class, $N$ and VCdim($\Theta$) in (31) are held constant. Consequently, the only term that influences the generalization bound is the *effective number of samples* $N_{\text{eff}}$: the larger $N_{\text{eff}}$, the higher the model's utility. It is easy to show that the effective dataset size is always upper-bounded by the total dataset size (i.e., $N_{\text{eff}} \leq N$) and is maximized for $w^* = N_R/N$. This choice results in the same weight given to every sample independently of whether it belongs to the training or reference data, i.e., $L^w_D = \frac{1}{N}\sum_{(x,y)\in D_T\cup D_R} \ell(f_\theta, (x, y))$. When $w = w^*$, training and reference data have the same privacy guarantees ($\epsilon_T = \epsilon_R$, see Section 3.3). Alternatively, when the privacy considerations are unequal, $N_{\text{eff}}$ degrades quadratically with respect to the difference between $w$ and $w^*$. We can thus conclude that heterogeneous privacy requirements for training and reference data (i.e., $\epsilon_T \neq \epsilon_R$) lead to samples being weighted differently in the two datasets, which causes an increase in the privacy of a selected dataset at the expense of overall model utility.

## 3.5 Theoretical Utility-Privacy Tradeoff

By combining Theorem 3.1 and Theorem 3.2, we can study WERM's utility-privacy tradeoff for different dataset sizes and different values of the weight $w$. In Figure 2, we plot $\epsilon$ privacy values against effective dataset sizes ($N_{\text{eff}}$) for different proportions of training

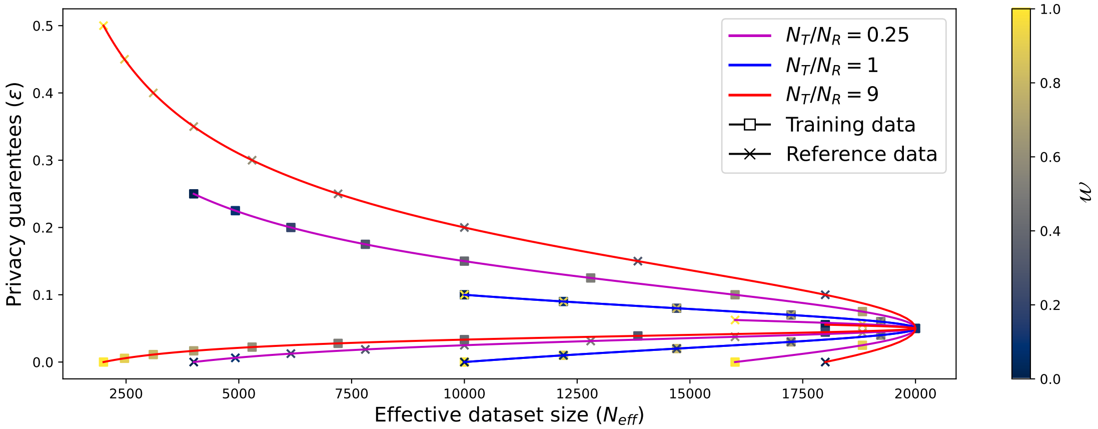

**Figure 2: Theoretical utility-privacy tradeoff for WERM trained with DP-SGD as derived by the bounds in Theorem 3.1 and Theorem 3.2, $\epsilon_0$ = 1000 and N = $N_T + N_R$ = 20, 000. The curves show the effective dataset size ($N_{eff}$, a proxy for model utility), training data privacy guarantees ($\epsilon_T$), and reference data privacy guarantees ($\epsilon_R$) are influenced by reference data weight values (w) and dataset size proportion ($N_T/N_R$). In Section 4.4, we show that the theoretical results presented here are aligned with the empirical results presented in Figure 3.**

data and reference data ($\frac{N_T}{N_R}$) and weight values in the interval [0.0, 1.0] using a fixed total dataset size (N) of 20000. We select values of $\frac{N_T}{N_R}$ equal to 0.25, 1, and 9 to represent each possible distinct data setting ($N_T > N_R, N_T = N_R, N_T < N_R$) without leading to overlapping curves.[4]

For a given dataset size proportion, by varying the reference data weight, $w$, WERM is capable of achieving a wide spectrum of tradeoffs for model utility, training data privacy, and reference data privacy. When $w = 0$, indicated by the darkest colored points in Figure 2, the reference data is fully protected (marked with "x'), while the training data is most exposed (marked with "o"). As $w$ increases, indicated by the color of the points becoming lighter, sacrificing reference data privacy (i.e., $\epsilon_R$ increasing) leads to greater training data privacy protection (i.e., $\epsilon_T$ decreasing).

The interaction between $w$ and model utility is particularly interesting. We observe that increasing the weight causes the model utility to first increase and then decrease. The maximum utility ($N_{\text{eff}} = N$) is obtained for $w^* = N_R/N$, which coincides with the setting of equal privacy guarantees for the two datasets ($\epsilon_T = \epsilon_R = \frac{\epsilon_0}{N}$). This result is independent of the relative size of the two datasets, and indeed we can observe that all curves share the point $(N, \epsilon_0/N)$. When $\epsilon_0$ increases, the $y$-scale in Figure 2 increases accordingly, but the shape of the curves does not change. Overall, Figure 2 confirms that WERM's utility-privacy tradeoff is easy to interpret, which is highly desirable for its role as a baseline defense.

## 4 EXPERIMENTS

In this section, we outline our training strategy and evaluation setting, describe in detail our process for training each empirical privacy defense, and conduct a systematic evaluation of our WERM baseline in a variety of utility-privacy settings. Ultimately, we will demonstrate that WERM's empirical utility-privacy tradeoff (Figure 3) is qualitatively similar to what is predicted by the theoretical analysis (Figure 2), which confirms our intuition that WERM is an interpretable baseline in both theory and practice.

### 4.1 Datasets

We chose to conduct our experiments on the Purchase100, Texas100, and CIFAR100 datasets because they have been widely used for assessing empirical privacy defenses and MIAs [20, 26, 30, 31, 40–42, 49, 51]. A detailed description of each dataset is provided in Appendix C.

### 4.2 Methodology

*4.2.1 Training.* Conducting a fair comparison of empirical privacy defenses requires using a standardized approach for dataset pre-processing (e.g., equivalent training/reference/test data size proportions) and model architecture choices for all methods. As AdvReg [30] is the first proposed empirical privacy defense and most well-studied method, its experimental setting has consequently become the de-facto standard for comparing defenses on the Purchase100 and Texas100 dataset [8, 26, 42]. In this setting, one applies the defense mechanism to a 4-layer fully connected neural network classifier with layer sizes [1024, 512, 256, 100], and uses 10% of Purchase100 (≈ 20,000 samples) and 15% of Texas100 (≈ 10,000 samples) as training data. For the CIFAR100 dataset, we use 20,000 samples for training data and align our study with more recent evaluations that consider a ResNet-18 [17] as the classification model [26, 48]. We assume that each defense has access to reference data that is the same size as the training data. Following the strategy of the original AdvReg experiments, all classification models are trained using an Adam optimizer [21] with a learning rate equal to 0.001. For reasons that are described in Appendix B.2, MMD requires using a batch size equal to 512 or greater. This contrasts with the

[4] We observe that the training (resp. reference) data curve for a given ratio $N_T/N_R = a$ coincides with the reference (resp. training) data curve for the reciprocal $N_T/N_R = 1/a$. Thus, it is also possible to observe the behavior for $N_T/N_R = 4$ and $N_TN_R = 1/9$ in Figure 2. We show an example for $N_T/N_R = 0.25$ in Figure 4 (Appendix E).

original AdvReg experiments that use a batch size equal to 128. To ensure a fair comparison, we train all evaluated defenses using both batch sizes and select the best version for each method. For a given defense, the reported results are mean values over 10 training runs for different seeds of a random number generator. Following the same training strategy as previous works [8, 30, 42], we train each defense for a specific number of epochs that ensures the model converges without severely overfitting.[5] Additionally, the regularization values we select for training each defense are explicitly chosen to demonstrate all possible relative privacy levels that a given method can achieve.[6]

*4.2.2 Evaluation.* We use the same methodology and released code[7] as Song and Mittal [42], where an empirical privacy defense is evaluated against three threshold-based MIAs and the gap attack [51]. Additionally, we evaluate against a neural network-based attack [41] that could be executed by a stronger adversary with access to training/non-training samples, and the results, shown in Figure 6 (Appendix E), are qualitatively similar to those in Figure 3.

Following the standard evaluation methodology [8, 20, 26, 30, 42], a distinct test dataset from the same underlying distribution as the training data is used to evaluate the final accuracy of the trained model and as the "non-training" data to evaluate (together with part of the training data) the accuracy of the MIA according to (4). Across all datasets, the two most effective attacks were threshold-based and used either the confidence value or modified-entropy. To quantify the privacy leakage in our experimental results, we report the MIA accuracy of the attack using confidence values because it requires less assumptions and performs equivalently well.

In our evaluation, we explicitly measure the capabilities of empirical privacy defenses in a variety of model utility, training data privacy, and reference data privacy settings to determine the most effective methods in each case. We define the notion of a *model instance* as a defended classifier obtained by training with a certain regularization or weight value. This terminology will be used as we select model instances that most closely adhere to a specific privacy setting (e.g., WERM trained with a weight value $w = 0.5$ is a model instance that is coherent with equal training and reference data privacy requirements, as the two datasets have the same size).

## 4.3 Evaluated Defenses

As WERM is designed to be a baseline for empirical privacy defenses using reference data, we only compare against methods in this category, which excludes the recently proposed Self-Distillation [44]. Specifically, we evaluate AdvReg [30], which is the most well-studied, and MMD [26], which is the current state-of-the-art. We do not consider confidence-vector masking defenses [20, 49] because they have been shown to be ineffective against label-based attacks such as the simple gap attack (5) [8, 42].

*4.3.1 WERM.* Using the classification models described in Section 4.2.1 for each dataset, we train WERM using weight values equal to 0.0, 0.03, 0.1, 0.3, and 0.5, as well as values of 0.98 and 0.02 (Purchase100), 0.999 and 0.001 (Texas100), and 0.9975 and 0.005 (CIFAR100) that are chosen specifically to achieve the constraints for "public reference data" and "high reference data privacy" as outlined in Section 4.4.[8] These weights were chosen to reflect the full range of utility-privacy tradeoffs that the method can achieve. To train WERM, for all reference data weight configurations, we fix the number of training epochs at 20, 4, and 25 for the Purchase100, Texas100, and CIFAR100 datasets, respectively. The number of training epochs for WERM, as well as for the other empirical privacy defenses we evaluate, are selected based on the standard methodology discussed in Section 4.2.1. Additionally, in all our experiments, we use the standard version of gradient descent, and the resulting models, therefore, have no formal DP guarantees. Nevertheless, we show that WERM's relative privacy guarantees—as measured by MIA accuracy—qualitatively align with the conclusions of Theorem 3.1, a result that is justified by our discussion at the end of Section 3.3.

While we analyze the generalization bound of WERM in Section 3.4 for the setting where the empirical loss is minimized, in practice it is possible to end training before convergence. This simple technique, known as early stopping [6], has been observed to protect privacy [42]. As WERM can potentially benefit from early stopping without incurring a loss of interpretability, we evaluate a version of our baseline, henceforth referred to as WERM-ES, that uses this approach. To train WERM-ES, for all reference data weight configurations, we fix the number of training epochs at 7, 1, and 6 for the Purchase100, Texas100, and CIFAR100 datasets, respectively.

*4.3.2 Adversarial Regularization.* Our AdvReg implementation relies on the officially released code[9] with a few changes to solve several problems we discuss in Appendix B.1.

We also evaluate a variant of AdvReg that can be obtained by modifying the gradient update in [30]. Although the declared objective is to solve problem (8), when taking the gradient of (8) with respect to $\theta$, Nasr et al. [30] only consider the terms evaluated on training data:

$$\nabla_\theta \frac{1}{m} \sum_{i=1}^{m} \ell(f_\theta, (x_i, y_i)) + \lambda \, \log[h_\phi(x_i, y_i, f_\theta(x_i))] \tag{19}$$

However, the gradient of (8) with respect to $\theta$ contains an additional term that is evaluated on the reference data:

$$\frac{\lambda}{m'} \sum_{i=1}^{m'} \log[1 - h_\phi(x'_i, y'_i, f_\theta(x'_i))] \tag{20}$$

We refer to this variant using the reference data term as AdvReg-RT. As was observed in Figure 1, AdvReg-RT achieves a distinct set

[5] It is also possible to use validation data to find an opportune epoch to end training. However, using a validation dataset introduces questions regarding the validation data's degree of privacy leakage [35]. As we are already evaluating the relative training and reference data privacy leakage, introducing another dataset will add further complexity to our analysis, which will make it more difficult to interpret the results. In Figure 5 (Appendix E), we present the utility-privacy curves using validation data to determine the number of training epochs. The difference is negligible compared to our results using a predetermined number of epochs.

[6] As discussed in Section 2.3, higher values of the regularization value should lead to higher privacy protection for training data, potentially leaking more information about reference data.

[7] https://github.com/inspire-group/membership-inference-evaluation

[8] Due to the symmetric role of training and reference data in WERM, privacy evaluation for training data for a given value $w$ corresponds to privacy evaluation for reference data for $1 - w$. In practice, the reported results therefore allow for evaluating a larger range of values including 0.7, 0.9, 0.97, and 1.0 for all datasets and 0.98, 0.999, and 0.995 for Purchase100, Texas100, and CIFAR100, respectively.

[9] https://github.com/SPIN-UMass/ML-Privacy-Regulization

of model utility, training data privacy, and reference data privacy tradeoffs compared to AdvReg. We therefore choose to compare both formulations with our WERM baseline.

Using the classification models described in Section 4.2.1 for each dataset, we train both versions of AdvReg using regularization values equal to 1, 2, 3, 6, 10, and 20 for Purchase100 and Texas100 and 1e-6, 1e-3, 1e-1, and 1 for CIFAR100. These values were selected on a per dataset basis to best represent the utility-privacy tradeoff that each formulation is capable of achieving. The number of training epochs is fixed at 10, 10, and 25 when training AdvReg and 35, 20, and 25 when training AdvReg-RT, for the Purchase100, Texas100, and CIFAR100 datasets, respectively.

*4.3.3 MMD-based Regularization.* Using the classification models described in Section 4.2.1 for each dataset, we train MMD using regularization values equal to 0.1, 0.2, 0.35, 0.7, and 1.5 that demonstrate the total achievable utility-privacy curve. As the released code implementation of AdvReg [30] benefits from training the classifier for a few warm-up steps without regularization, we also train MMD with and without a warm-up, reporting only the best results for each dataset. The number of training epochs is fixed at 25, 8, and 15 when training without warm-up steps and 20, 8, and 8 when training with warm-up steps, for the Purchase100, Texas100, and CIFAR100 datasets, respectively. Details about the implementation can be found in Appendix B.2.

## 4.4 Empirical Results

In Figure 3, we show the empirical utility-privacy tradeoffs obtained by AdvReg [30], MMD [26], AdvReg-RT, and WERM for the Purchase100, Texas100, and CIFAR100 datasets. In these plots, we show the exact points that make up the curve, as well as some qualitative lines to highlight the trends and improve readability. The curves derived from theoretical bounds in Figure 2 and from experimental results in Figure 3 both show utility vs. privacy. However, Figure 2 evaluates the utility through the effective number of samples, $N_{\text{eff}}$, and privacy leakage through the DP parameters, $\epsilon_T$ and $\epsilon_R$, whereas Figure 3 uses the test accuracy and MIA accuracy. Table 2 focuses specifically on our three key privacy settings: public reference data, equal training-reference data privacy and high reference data privacy.

*4.4.1 Utility-Privacy Curve Analysis.* The three objectives of model utility, training data privacy leakage, and reference data privacy leakage are inherently in conflict with one another. Ideally, one would like to see that an empirical privacy defense can produce a landscape of model instances that spans a vast range of utility-privacy regimes. In theory, each of the methods we evaluate should have this capability, as they all have a mechanism for controlling the amount of regularization that is applied during training. Examining the utility-privacy curves in Figure 3 allows us to understand the tradeoffs that the various defenses can achieve in practice. As noted in Section 4.2.2, we quantify privacy leakage using a threshold-based MIA on a classifier's confidence values.[10]

WERM is the only defense that can clearly tradeoff between the three objectives. For Purchase100, over the range of privacy settings from equal privacy ($w = 0.5$) to high reference data privacy ($w = 0$), we see that WERM achieves values of 87% / 54.7% / 54.9% and 81.8% / 57.6% / 50%, for test accuracy (model utility), MIA accuracy on training data, and MIA accuracy on reference data, respectively. Between these edge cases, we see that WERM can produce model instances capable of trading off reference data privacy for both model utility and training data privacy. The same trend can be observed for WERM on the Texas100 and CIFAR100 datasets. Regarding WERM-ES, as shown in Figure 7 (Appendix E), the defense exhibits equivalent behavior to WERM, but, as expected, it achieves higher overall privacy protection at the cost of lower model utility. Figure 7 also contains the utility-privacy curves for early stopping (EarlyStop) using only the training data.

Looking at the state-of-the-art defenses we evaluate (AdvReg-RT, AdvReg, and MMD) reveals two situations. First, we can see that AdvReg-RT is able to sacrifice model utility for overall better privacy protection. However, it does not have the ability to trade off between training data privacy and reference data privacy, as changing the regularization value $\lambda$ results in the training and reference data privacy leakage increasing/decreasing together. Due to this limited functionality, AdvReg-RT is never able to reach the setting where training data privacy protection is equal to reference data privacy protection. Second, the utility-privacy curves for AdvReg and MMD demonstrate that these defenses are completely unable to trade reference data privacy for either model utility or training data privacy. While training data privacy can be sacrificed for better model utility, attack accuracy on reference data never materially changes, remaining below 51% for both defenses at all meaningful test accuracy values. Overall, we observe that for the entire curve of possible utility-privacy tradeoffs, excluding the high reference data privacy setting, WERM/WERM-ES is unequivocally the best-performing method, and in any regime where training data privacy is valued absolutely equal to or greater than reference data privacy (including the public reference data case), WERM/WERM-ES is, in fact, the only viable defense.

In addition to WERM being a baseline defense with good utility, we also want its output to align with the desired relative privacy level that is encoded in a given choice of $w$. Comparing WERM's empirical utility-privacy tradeoffs in Figure 3 with the theoretical tradeoffs in Figure 2, we see that they exhibit the same trend where a gradual transition occurs from the setting of high reference data privacy to that of equal privacy over the weight value interval of [0.0, 0.5]. The fact that our theoretical bounds are qualitatively aligned with our experimental results helps to demonstrate that WERM is indeed an interpretable baseline. We conduct a quantitative comparison in Section 5.1.

*4.4.2 Public Reference Data.* First, we examine the case where reference data is public. In this setting, the privacy of reference data is of no concern. Therefore, an optimal defense should utilize the reference data to the furthest extent possible to decrease training data privacy leakage and increase test accuracy. For a given empirical privacy defense, we select model instances using the following procedure: 1) Identify all model instances with MIA accuracy on training data less than or equal to 51%, 2) Among the

[10] A random guesser would get an average expected accuracy of 0.5 but its average accuracy on a finite dataset can either exceed or fall short of 0.5. It should then not be surprising that some attacks have an accuracy marginally below 0.5. (e.g., 0.498), as can be seen in Figure 3.

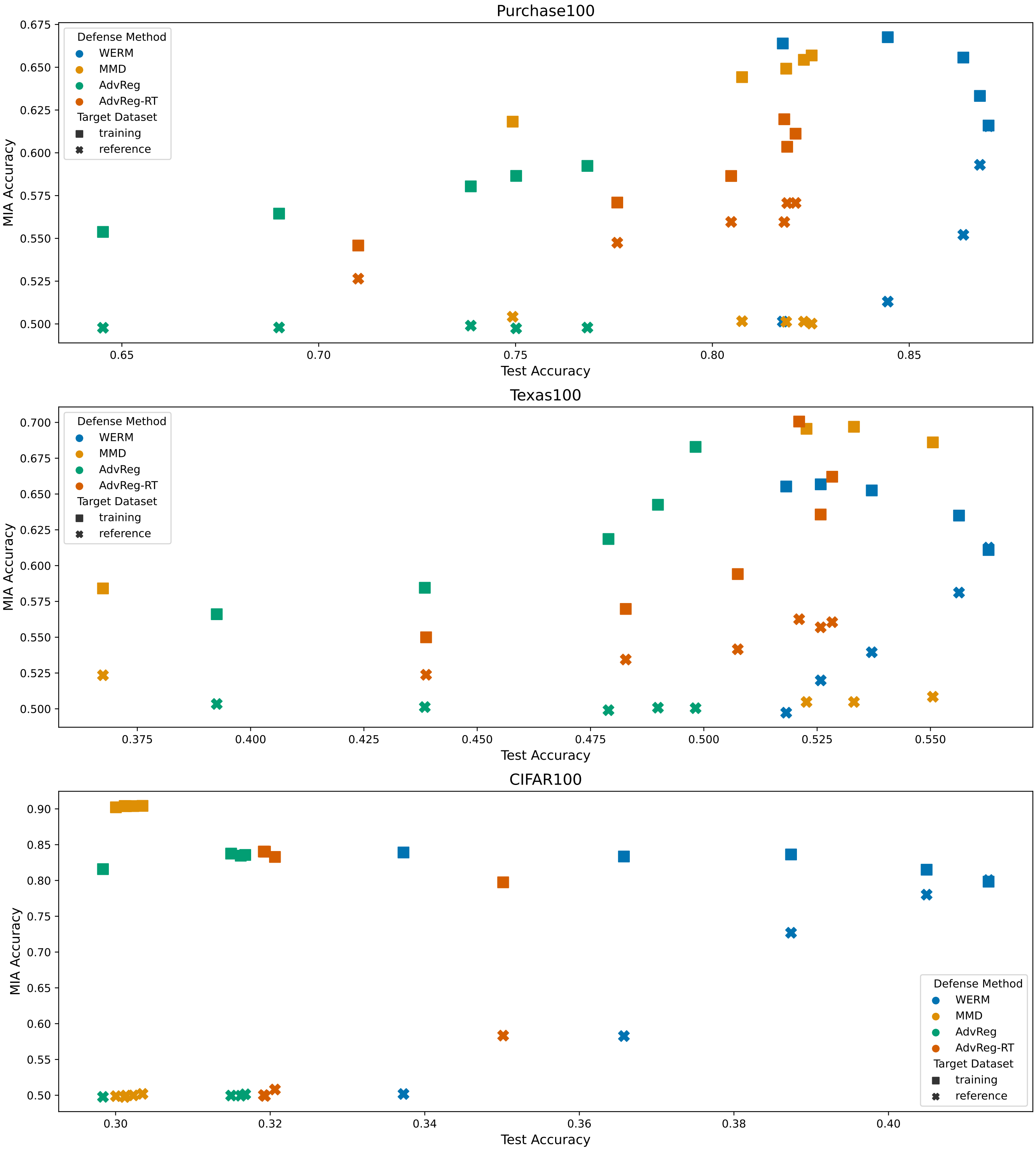

**Figure 3: Utility-privacy tradeoffs obtained by various empirical privacy defenses for the Purchase100, Texas100, and CIFAR100 datasets. The test accuracy of a defended classifier is measured using unseen test data and the MIA accuracy on training and reference data is evaluated with a threshold-based using confidence values (Eq. 6). Each point on the curve represents the evaluation of a model instance using a distinct regularization value (for AdvReg, AdvReg-RT, and MMD) and reference data weight value (for WERM). We highlight some qualitative trends to help demonstrate that the empirical curves coincide with the theoretical curves in Figure 2.**

**Table 2: Comparison of test accuracy and MIA accuracy for AdvReg, AdvReg-RT, MMD, WERM, and WERM-ES under the settings of public reference data, equal privacy considerations, and high reference data privacy. A dashed line (—) means that the defense produced no model instances that met the criteria. The values under "MIA Train" and "MIA Ref" represent the membership inference attack accuracy on training and reference data, respectively.**

| Dataset | Defense | Public Reference Data | | | Equal Privacy Considerations | | | High Reference Data Privacy | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | | Test Acc. | MIA Train | MIA Ref | Test Acc. | MIA Train | MIA Ref | Test Acc. | MIA Train | MIA Ref |
| Purchase100 | AdvReg | — | — | — | — | — | — | 76.8% | 59.2% | 50.0% |
| | AdvReg-RT | — | — | — | 82.1% | 61.1% | 57.1% | — | — | — |
| | MMD | — | — | — | — | — | — | 82.5% | 65.7% | 50.0% |
| | WERM | 83.8% | 51.0% | 66.7% | 87.0% | 61.5% | 61.5% | 84.2% | 68.0% | 50.9% |
| | WERM-ES | 77.9% | 50.1% | 57.4% | 83.6% | 54.7% | 54.9% | 78.4% | 57.8% | 50.0% |
| Texas100 | AdvReg | — | — | — | — | — | — | 49.8% | 68.3% | 50.0% |
| | AdvReg-RT | — | — | — | 48.3% | 57.0% | 53.4% | — | — | — |
| | MMD | — | — | — | — | — | — | 55.1% | 68.6% | 50.8% |
| | WERM | 54.2% | 50.9% | 65.6% | 56.3% | 61.1% | 61.3% | 52.0% | 65.7% | 50.9% |
| | WERM-ES | 43.7% | 50.3% | 55.9% | 49.8% | 54.3% | 54.0% | 44.4% | 56.3% | 50.5% |
| CIFAR100 | AdvReg | — | — | — | — | — | — | 31.7% | 83.6% | 50.0% |
| | AdvReg-RT | — | — | — | — | — | — | 31.1% | 83.3% | 50.8% |
| | MMD | — | — | — | — | — | — | 30.2% | 90.4% | 50.0% |
| | WERM | 34.2% | 50.5% | 84.0% | 41.3% | 79.8% | 80.1% | 33.8% | 83.5% | 50.9% |
| | WERM-ES | 32.9% | 50.1% | 63.3% | 40.1% | 60.2% | 60.2% | 33.0% | 63.6% | 50.0% |

model instances meeting this criterion, select the one with best test accuracy. As can be observed in Table 2, on Purchase100, Texas100, and CIFAR100, only WERM or WERM-ES are able to produce suitable model instances in this setting. Although MMD and AdvReg include a regularization term that is intended to tradeoff privacy against test accuracy, the methods are simply not capable of maximally exploiting reference data. Alternatively, WERM can sacrifice reference data privacy to achieve a high test accuracy and strict training data privacy protection.

*4.4.3 Equal Privacy Requirements.* Second, we examine the setting where training and reference data have equal privacy requirements. For a given empirical privacy defense, we select the model instance using the following procedure: 1) Identify all model instances where the difference between the attack accuracy on training data and reference data is less than or equal to 4%, 2) Among the model instances meeting this condition, select the one with best test accuracy. We use 4% as the threshold to define "equal" privacy considerations because at lower thresholds AdvReg-RT is not able to achieve a model instance with satisfactory utility to be relevant for comparison. Table 2 shows that only WERM/WERM-ES and AdvReg-RT are able to operate in this privacy regime; MMD and AdvReg fail to produce any viable model instances.

On Purchase100 and Texas100, for the selected model instances, WERM-ES outperforms AdvReg-RT on all three objectives. Compared to AdvReg-RT and WERM-ES, WERM achieves significantly higher model utility, at the expense of worse training and reference data privacy. On CIFAR100, AdvReg-RT is unable to yield a model instance that meets the conditions, making WERM/WERM-ES the only working defense.

*4.4.4 High Reference Data Privacy.* Lastly, we examine the case where reference data is considered highly private and its privacy can therefore only be minimally sacrificed. For a given empirical privacy defense, we select model instances using the following procedure: 1) Identify all model instances with MIA accuracy on reference data less than or equal to 51%, 2) Among the model instances meeting this criterion, select the one with best test accuracy. In Table 2, it can be seen that on Purchase100 and Texas100, AdvReg, MMD, WERM, and WERM-ES are all able to produce a model instance that leaks only minimal reference data privacy according to our selection method, whereas AdvReg-RT is unable to yield a suitable model instance. However, on CIFAR100, all five defenses achieve a valid result.

By setting very strict privacy requirements for reference data, we aim to remove one objective from the evaluation such that we can make a comparison based solely on the model utility and training data privacy leakage. Nonetheless, it is still possible that two model instances satisfy the reference data privacy requirement but cannot be definitively compared because one can achieve better utility and the other better training data privacy protection. On Purchase100, for example, WERM-ES outperforms AdvReg, but it is not clear which method is preferable between WERM, WERM-ES, and MMD without deciding the relative importance assigned to model utility and training data privacy. WERM can achieve higher test accuracy, WERM-ES can achieve higher privacy protection on training data, and MMD can achieve a utility-privacy tradeoff that is a middle ground between these two methods. On Texas100, WERM and MMD both outperform AdvReg, but a comparison between the two also requires making explicit the relative importance of model utility and training data privacy. On CIFAR100, however, WERM/WERM-ES is clearly superior to the other three defenses.

**Table 3: Comparison of the Pearson Correlation Coefficient between the training-reference data desired privacy ratio (as determined by the choice of $w$ or $\lambda$) and the empirical privacy ratio (as measured by a MIA) for WERM, AdvReg, and MMD. The coefficient is computed across the Purchase100, Texas100, and CIFAR100 datasets.**

| Defense | WERM | AdvReg | MMD |
|---|---|---|---|
| Pearson Correlation Coefficient | 0.84 | 0.07 | 0.48 |

# 5 DISCUSSION

## 5.1 Selection of Defense Parameters

Even when a model training entity has clearly defined its desired relative privacy level between training and reference data, realizing a classifier with this exact degree of relative privacy protection still requires selecting the corresponding parameters of the empirical privacy defense: the reference data weight term ($w$) for WERM and regularization weight term ($\lambda$) for AdvReg or MMD. As an illustration, if the two datasets are of equal size, and the reference data needs to be twice as private, what should be the chosen values of $w$ for WERM or $\lambda$ for AdvReg or MMD? We argue that an empirical privacy defense becomes more practical if there exists an intelligible (e.g., linear) mapping to guide a machine learning practitioner in the selection of a defense parameter. AdvReg and MMD do not provide a practical guideline except for the general intuition that a larger value of $\lambda$ should provide higher training data privacy and lower reference data privacy. For WERM, the parameter $w$ can be adjusted to ensure a specified theoretical level of relative privacy, as dictated by the equations in Theorem 3.2 ($\epsilon_T/\epsilon_R = (1-w)/w \times N_R/N_T$). However, translating DP-like theoretical privacy guarantees into practical privacy guarantees e.g., in terms of MIA accuracy, is a highly complex and still unresolved issue in the field of privacy-preserving machine learning [4, 5, 32, 50]. The effectiveness of such a configuration rule therefore needs to be evaluated in terms of the empirical privacy leakage.

In Table 3 we report the Pearson correlation coefficient (PCC) between WERM's theoretical relative privacy, as defined by the value $\epsilon_T/\epsilon_R$, and its empirical relative privacy, as measured by the ratio between the MIA accuracy on training data and on reference data. The coefficient gauges the linear correlation between the two quantities. For AdvReg and MMD, without clear configuration guidelines, we report the PCC between the reciprocal of the regularization parameter $1/\lambda$ and the empirical relative privacy. WERM displays the largest PCC at 0.84, which stands in stark contrast to the 0.07 for AdvReg and 0.48 for MMD. These results underscore WERM as the sole method offering a practical configuration rule to achieve a target relative privacy.

## 5.2 Computational Cost Comparison

In addition to comparing the utility-privacy tradeoff and practical usability of empirical privacy defenses, it is also important to consider their computational cost. Table 4 shows, for a fixed batch size, the number of seconds it takes to train each defense for a single epoch and the overall training time considering the total number of epochs used in our experiments. We calculate the overall training time as the per epoch training time multiplied by the total number of epochs. Each of these experiments are run using a single GPU on an NVIDIA DGX system. Analyzing Table 4 confirms that WERM is indeed significantly less computationally expensive than AdvReg and MMD, as discussed in Section 3.2. On Purchase100, Texas100, and CIFAR100, WERM is 19x, 7x, and 19x faster to train on a per epoch basis, compared to the second fastest method.

**Table 4: Comparison of per epoch and overall training time, in seconds (s), for each empirical privacy defense on Purchase100, Texas100, and CIFAR100.**

| Dataset | Defense | Per Epoch (s) | Overall (s) |
|---|---|---|---|
| Purchase100 | WERM | 0.5 | 10 |
| | AdvReg | 9.5 | 95 |
| | MMD | 16.4 | 328 |
| Texas100 | WERM | 0.9 | 3.6 |
| | AdvReg | 6.4 | 64 |
| | MMD | 6.8 | 54.4 |
| CIFAR100 | WERM | 4.1 | 102.5 |
| | AdvReg | 78.8 | 1970 |
| | MMD | 94.7 | 757.6 |

# 6 CONCLUSION AND FUTURE WORK

In this work, we have analyzed the role of reference data in empirical privacy defenses and identified the issue that reference data privacy leakage must be explicitly considered to conduct a meaningful evaluation. We advanced the current state-of-the-art by proposing a generalization error constrained ERM, which can in practice be evaluated as a weighted ERM over the training and reference datasets. As WERM is intended to function as a baseline, we derive theoretical guarantees about its utility and privacy to ensure that its results will be well-understood in all utility-privacy settings. We present experimental results showing that our principled baseline outperforms the most well-studied and current state-of-the-art empirical privacy defenses in nearly all privacy regimes (i.e., independent of the nature of reference data and its level of privacy). Our experiments also reveal that existing methods are unable to trade off reference data privacy for model utility and/or training data privacy, and thus cannot operate outside of the highly private reference data case.

Regarding ethical concerns, our proposed baseline operates on the defense side of machine learning privacy; no novel attack has been proposed. Nevertheless, our experiments have analyzed the average privacy leakage over the whole dataset, but privacy protection is not always fair across groups in a dataset [10, 24]. Future work can evaluate then the fairness of various defense mechanisms using reference data or propose the creation of privacy defenses intended to operate in use-case dependent settings. We hope that our work will continue to motivate the development of a robust evaluation framework for privacy defenses.

## ACKNOWLEDGMENTS

This research was supported in part by ANRT in the framework of a CIFRE PhD (2021/0073) and by the Horizon Europe project dAIEDGE.

## REFERENCES

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. 2016. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*. 308–318.

[2] Eugene Bagdasaryan, Omid Poursaeed, and Vitaly Shmatikov. 2019. Differential privacy has disparate impact on model accuracy. *Advances in neural information processing systems* 32 (2019).

[3] Yoshua Bengio. 2012. Deep learning of representations for unsupervised and transfer learning. In *Proceedings of ICML workshop on unsupervised and transfer learning*. JMLR Workshop and Conference Proceedings, 17–36.

[4] Daniel Bernau, Philip-William Grassal, Jonas Robl, and Florian Kerschbaum. 2019. Assessing differentially private deep learning with membership inference. *arXiv preprint arXiv:1912.11328* (2019).

[5] Nicholas Carlini, Steve Chien, Milad Nasr, Shuang Song, Andreas Terzis, and Florian Tramer. 2022. Membership inference attacks from first principles. In *2022 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1897–1914.

[6] Rich Caruana, Steve Lawrence, and C Giles. 2000. Overfitting in neural nets: Backpropagation, conjugate gradient, and early stopping. *Advances in neural information processing systems* 13 (2000).

[7] Dingfan Chen, Ning Yu, Yang Zhang, and Mario Fritz. 2020. Gan-leaks: A taxonomy of membership inference attacks against generative models. In *Proceedings of the 2020 ACM SIGSAC conference on computer and communications security*. 343–362.

[8] Christopher A Choquette-Choo, Florian Tramer, Nicholas Carlini, and Nicolas Papernot. 2021. Label-only membership inference attacks. In *International conference on machine learning*. PMLR, 1964–1974.

[9] Federal Communications Commission et al. 2016. Protecting the Privacy of Customers of Broadband and Other Telecommunications Service (2016). (2016).

[10] Anderson Santana de Oliveira, Caelin Kaplan, Khawla Mallat, and Tanmay Chakraborty. 2023. An Empirical Analysis of Fairness Notions under Differential Privacy. *arXiv preprint arXiv:2302.02910* (2023).

[11] Cynthia Dwork, Aaron Roth, et al. 2014. The algorithmic foundations of differential privacy. *Foundations and Trends® in Theoretical Computer Science* 9, 3–4 (2014), 211–407.

[12] Stacy-Ann Elvy. 2017. Paying for privacy and the personal data economy. *Colum. L. Rev.* 117 (2017), 1369.

[13] Matt Fredrikson, Somesh Jha, and Thomas Ristenpart. 2015. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*. 1322–1333.

[14] Georgi Ganev, Bristena Oprisanu, and Emiliano De Cristofaro. 2022. Robin Hood and Matthew Effects: Differential Privacy Has Disparate Impact on Synthetic Data. In *International Conference on Machine Learning*. PMLR, 6944–6959.

[15] Karan Ganju, Qi Wang, Wei Yang, Carl A Gunter, and Nikita Borisov. 2018. Property inference attacks on fully connected neural networks using permutation invariant representations. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*. 619–633.

[16] Catherine Han, Irwin Reyes, Álvaro Feal, Joel Reardon, Primal Wijesekera, Narseo Vallina-Rodriguez, Amit Elazar, Kenneth A Bamberger, and Serge Egelman. 2020. The price is (not) right: Comparing privacy in free and paid apps. *Proceedings on Privacy Enhancing Technologies* 2020, 3 (2020).

[17] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[18] Xinlei He, Jinyuan Jia, Michael Backes, Neil Zhenqiang Gong, and Yang Zhang. 2021. Stealing links from graph neural networks. In *30th USENIX Security Symposium (USENIX Security 21)*. 2669–2686.

[19] Seyyedali Hosseinalipour, Christopher G Brinton, Vaneet Aggarwal, Huaiyu Dai, and Mung Chiang. 2020. From federated to fog learning: Distributed machine learning over heterogeneous wireless networks. *IEEE Communications Magazine* 58, 12 (2020), 41–47.

[20] Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, and Neil Zhenqiang Gong. 2019. Memguard: Defending against black-box membership inference attacks via adversarial examples. In *Proceedings of the 2019 ACM SIGSAC conference on computer and communications security*. 259–274.

[21] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[22] Kalpesh Krishna, Gaurav Singh Tomar, Ankur P Parikh, Nicolas Papernot, and Mohit Iyyer. 2019. Thieves on sesame street! model extraction of bert-based apis. *arXiv preprint arXiv:1910.12366* (2019).

[23] Anders Krogh and John Hertz. 1991. A simple weight decay can improve generalization. *Advances in neural information processing systems* 4 (1991).

[24] Bogdan Kulynych, Mohammad Yaghini, Giovanni Cherubin, Michael Veale, and Carmela Troncoso. 2019. Disparate vulnerability to membership inference attacks. *arXiv preprint arXiv:1906.00389* (2019).

[25] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proc. IEEE* 86, 11 (1998), 2278–2324.

[26] Jiacheng Li, Ninghui Li, and Bruno Ribeiro. 2021. Membership inference attacks and defenses in classification models. In *Proceedings of the Eleventh ACM Conference on Data and Application Security and Privacy*. 5–16.

[27] Qinbin Li, Bingsheng He, and Dawn Song. 2021. Practical One-Shot Federated Learning for Cross-Silo Setting. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, Zhi-Hua Zhou (Ed.). International Joint Conferences on Artificial Intelligence Organization, 1484–1490. https://doi.org/10.24963/ijcai.2021/205 Main Track.

[28] Tao Lin, Lingjing Kong, Sebastian U Stich, and Martin Jaggi. 2020. Ensemble distillation for robust model fusion in federated learning. *Advances in Neural Information Processing Systems* 33 (2020), 2351–2363.

[29] Othmane Marfoq, Giovanni Neglia, Richard Vidal, and Laetitia Kameni. 2022. Personalized federated learning through local memorization. In *International Conference on Machine Learning*. PMLR, 15070–15092.

[30] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2018. Machine learning with membership privacy using adversarial regularization. In *Proceedings of the 2018 ACM SIGSAC conference on computer and communications security*. 634–646.

[31] Milad Nasr, Reza Shokri, and Amir Houmansadr. 2019. Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning. In *2019 IEEE symposium on security and privacy (SP)*. IEEE, 739–753.

[32] Milad Nasr, Shuang Songi, Abhradeep Thakurta, Nicolas Papemoti, and Nicholas Carlin. 2021. Adversary instantiation: Lower bounds for differentially private machine learning. In *2021 IEEE Symposium on Security and Privacy (SP)*. IEEE, 866–882.

[33] Nicolas Papernot, Martín Abadi, Ulfar Erlingsson, Ian Goodfellow, and Kunal Talwar. 2016. Semi-supervised knowledge transfer for deep learning from private training data. *arXiv preprint arXiv:1610.05755* (2016).

[34] Nicolas Papernot, Shuang Song, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Úlfar Erlingsson. 2018. Scalable private learning with pate. *arXiv preprint arXiv:1802.08908* (2018).

[35] Nicolas Papernot and Thomas Steinke. 2021. Hyperparameter tuning with renyi differential privacy. *arXiv preprint arXiv:2110.03620* (2021).

[36] European Parliament. 2016. *General Data Protection Regulation (GDPR)*. European Parliament. https://gdpr-info.eu/

[37] European Parliament. 2020. *The impact of the General Data Protection Regulation (GDPR) on artificial intelligence*. European Parliamentary Research Service. https://www.europarl.europa.eu/RegData/etudes/STUD/2020/641530/EPRS_STU(2020)641530_EN.pdf

[38] Ahmed Salem, Apratim Bhattacharya, Michael Backes, Mario Fritz, and Yang Zhang. 2020. {Updates-Leak}: Data Set Inference and Reconstruction Attacks in Online Learning. In *29th USENIX security symposium (USENIX Security 20)*. 1291–1308.

[39] Shai Shalev-Shwartz and Shai Ben-David. 2014. *Understanding machine learning: From theory to algorithms*. Cambridge university press.

[40] Virat Shejwalkar and Amir Houmansadr. 2021. Membership privacy for machine learning models through knowledge transfer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 35. 9549–9557.

[41] Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. 2017. Membership inference attacks against machine learning models. In *2017 IEEE symposium on security and privacy (SP)*. IEEE, 3–18.

[42] Liwei Song and Prateek Mittal. 2021. Systematic evaluation of privacy risks of machine learning models. In *30th USENIX Security Symposium (USENIX Security 21)*. 2615–2632.

[43] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* 15, 1 (2014), 1929–1958.

[44] Xinyu Tang, Saeed Mahloujifar, Liwei Song, Virat Shejwalkar, Milad Nasr, Amir Houmansadr, and Prateek Mittal. 2021. Mitigating membership inference attacks by self-distillation through a novel ensemble architecture. *arXiv preprint arXiv:2110.08324* (2021).

[45] Florian Tramer and Dan Boneh. 2020. Differentially private learning needs better features (or much more data). *arXiv preprint arXiv:2011.11660* (2020).

[46] Florian Tramèr, Fan Zhang, Ari Juels, Michael K Reiter, and Thomas Ristenpart. 2016. Stealing machine learning models via prediction {APIs}. In *25th USENIX security symposium (USENIX Security 16)*. 601–618.

[47] Archit Uniyal, Rakshit Naidu, Sasikanth Kotti, Sahib Singh, Patrik Joslin Kenfack, Fatemehsadat Mireshghallah, and Andrew Trask. 2021. DP-SGD vs PATE: Which

Has Less Disparate Impact on Model Accuracy? *arXiv preprint arXiv:2106.12576* (2021).

[48] Yijue Wang, Chenghong Wang, Zigeng Wang, Shanglin Zhou, Hang Liu, Jinbo Bi, Caiwen Ding, and Sanguthevar Rajasekaran. 2020. Against membership inference attack: Pruning is all you need. *arXiv preprint arXiv:2008.13578* (2020).

[49] Ziqi Yang, Bin Shao, Bohan Xuan, Ee-Chien Chang, and Fan Zhang. 2020. Defending model inversion and membership inference attacks via prediction purification. *arXiv preprint arXiv:2005.03915* (2020).

[50] Jiayuan Ye, Aadyaa Maddi, Sasi Kumar Murakonda, Vincent Bindschaedler, and Reza Shokri. 2021. Enhanced membership inference attacks against machine learning models. *arXiv preprint arXiv:2111.09679* (2021).

[51] Samuel Yeom, Irene Giacomelli, Matt Fredrikson, and Somesh Jha. 2018. Privacy risk in machine learning: Analyzing the connection to overfitting. In *2018 IEEE 31st computer security foundations symposium (CSF)*. IEEE, 268–282.

[52] Samuel Yeom, Irene Giacomelli, Alan Menaged, Matt Fredrikson, and Somesh Jha. 2020. Overfitting, robustness, and malicious algorithms: A study of potential causes of privacy risk in machine learning. *Journal of Computer Security* 28, 1 (2020), 35–70.

# A WERM THEORETICAL ANALYSIS

## A.1 Details on Lagrangian

The Lagrangian of our constrained minimization problem has the following form:

$$\min_{\theta,\lambda,\mu} L_D + \lambda\Big[L_{\mathbb{D}} - L_{D_T} - c_T\Big] + \mu\Big[L_{\mathbb{D}} - L_{D_R} - c_R\Big]. \tag{21}$$

Assuming we know the optimal multipliers, $\lambda^*$ and $\mu^*$, we can find $\theta$ by minimizing the following equation:

$$\min_{\theta} L_D + \lambda^*\Big[L_{\mathbb{D}} - L_{D_T} - c_T\Big] + \mu^*\Big[L_{\mathbb{D}} - L_{D_R} - c_R\Big] \tag{22}$$

or equivalently, as $c_T$ and $c_R$ are constants:

$$\min_{\theta} L_D + \lambda^*\Big[L_{\mathbb{D}} - L_{D_T}\Big] + \mu^*\Big[L_{\mathbb{D}} - L_{D_R}\Big]. \tag{23}$$

In the ideal case, one would have access to the underlying distribution, $\mathbb{D}$, and compute all terms in (23). Since $\mathbb{D}$ is unknown, we must instead use an approximation. Given that training and reference data are both drawn from $\mathbb{D}$, we estimate $L_{\mathbb{D}}$ as $L_{D_R}$ in the second term of (23) and $L_{\mathbb{D}}$ as $L_{D_T}$ in the third term of (23). Thus, our formulation becomes:

$$\begin{aligned}\min_{\theta} L_D + \lambda^*\Big[L_{D_R} - L_{D_T}\Big] + \mu^*\Big[L_{D_T} - L_{D_R}\Big] \qquad (24)\\ = \Big[\frac{1}{2} - \lambda^* + \mu^*\Big]L_{D_T} + \Big[\frac{1}{2} + \lambda^* - \mu^*\Big]L_{D_R}.\end{aligned}$$

Although both $\lambda^*$ and $\mu^*$ appear in (24), at the optimum $(\theta^*, \lambda^*, \mu^*)$ we expect that only the stricter of the two privacy constraints in (10) will be active (i.e., satisfied with equality) and then with a positive Lagrange multiplier, while the other will be inactive (i.e., it will be a strict inequality) with a null Lagrange multiplier. Since $D_T$ and $D_R$ are drawn from $\mathbb{D}$ and have the same size, $L_{D_T}$ and $L_{D_R}$ are two random variables with the same distribution. As a result, the stricter constraint is determined by the smaller of the two constants ($c_T$ and $c_R$), and we will have either $\lambda^* = 0$ and $\mu^* > 0$ (if $c_R < c_T$) or $\lambda^* > 0$ and $\mu^* = 0$ (if $c_T < c_R$).

## A.2 Extension to Multiple Datasets

In this paper, we have considered the case when there are two datasets (training data and reference data) with different levels of privacy. We will prove Theorem 3.1 and Theorem 3.2 under the more general case, when there are multiple datasets coming from the same distribution, and each individual dataset can have its own distinct weighting according to its privacy level (up to the limit case where every point is a separate dataset with distinct privacy).

In this case $L_D^w$ consists of any possible weighted combination of losses evaluated over multiple datasets, i.e., $L_D^w = \sum_{m=1}^{M} w_m L_{D_m}$, and $\sum_{m=1}^{M} w_m = 1$. A given dataset, $D_m$, is comprised of $|D_m| = N_m$ data points that are drawn i.i.d. from $\mathbb{D}$, such that $D_m = \{z_m^{(i)}, i \in [N_m]\}$. Accordingly, a given aggregated dataset, $D$, is comprised of $N = |D|$ data points where $N = \sum_{i=1}^{M} N_m$.

## A.3 Proof of Theorem 3.1

**Algorithm.** In the DP-SGD algorithm [1], differential privacy is added to the training procedure by clipping a model's gradients below a certain threshold and adding noise to the sum using the following equations (slightly modified from their original presentation in Abadi et al. [1]):

$$\bar{g}_i = \frac{g_i}{\max\Big\{1, \frac{\|g_i\|_2}{C}\Big\}} \tag{25}$$

$$\tilde{g} = \frac{1}{L}\left(\sum_{i}^{L} \bar{g}_i + \mathcal{N}\left(0, \sigma^2 C^2 \mathbf{I}\right)\right) \tag{26}$$

where $g_i$ is the gradient-vector of an arbitrary data point, C is the gradient norm bound, $\sigma$ is the noise scale, and L is the number of data points considered during a given step (i.e., batch size). By applying these operations and choosing $\sigma$ to be $\sqrt{2\log\frac{1.25}{\delta}}/\epsilon$, according to standard arguments [11], each step of DP-SGD is $(\epsilon, \delta)$-differentially private with respect to the batch. Moreover, for a training dataset of size $N$, sampling ratio equal to $\alpha = L/N$, and some overall number of steps $K$, using the moments account [1] the algorithm is $(O(\epsilon\alpha\sqrt{K}), \delta)$-differentially private with respect to the training dataset for an appropriate choice of noise scale and gradient norm bound.

Equation 26 considers that each data point comes from the same dataset and has the same weight. In presence of multiple datasets (see A.2) we extend DP-SGD as follows

$$\tilde{g} = \sum_{m=1}^{M} \frac{w_m}{L_m} \sum_{i=1}^{L_m} \bar{g}_{i,m} + \mathcal{N}\left(0, \sigma^2 C^2 \mathbf{I}\right), \tag{27}$$

where $L_m$ is the batch size for dataset $D_m$, $\bar{g}_{i,m}$ is the gradient-vector of a data point in $L_m$ after the clipping operation (25) has been applied, and all other terms are the same as in (26). In our analysis, we consider the case where $L_m = \alpha N_m$ and then $\alpha = \frac{L}{\sum_{i=1}^{m} N_m}$ is the sampling ratio.

***Theorem 3.1.*** *For some overall number of training steps, K, WERM minimized with DP-SGD (Eq. 27) is:*

$$\Big(O(\epsilon_T), \delta\Big) - DP \text{ w.r.t. the training dataset } (D_T)$$

$$\Big(O(\epsilon_R), \delta\Big) - DP \text{ w.r.t. the reference dataset } (D_R)$$

*where:*

$$\epsilon_T = \epsilon_0 \frac{1-w}{N_T}$$
$$\epsilon_R = \epsilon_0 \frac{w}{N_R}$$
$$0 < \epsilon_0 < \min\left(\frac{N_T}{1-w}, \frac{N_R}{w}\right),$$
$$\sigma \geq \alpha\sqrt{K}\sqrt{2\log\frac{1.25}{\delta}}\frac{C}{\epsilon_0},$$

*$w$ is the reference data weight in* (13), *$N_T$ is the size of training data, $N_R$ is size of the reference data, $K$ is the number of training steps, and $\sigma$, $C$, and $\alpha$ are the noise scale, gradient norm bound, and sampling ratio in DP-SGD, respectively.*

Proof. We define $\tilde{g}_m = \frac{w_m}{L_m}\sum_{i=1}^{L_m}\bar{g}_{i,m}$ and start studying the case $\alpha = 1$. The $\ell_2$ sensitivity of $\tilde{g}_m$ w.r.t. a point in the dataset is $\Delta\tilde{g}_m = 2\frac{w_m}{N_m}C$.

Let $\epsilon_1 = \epsilon_0\frac{w_1}{N_1}, \epsilon_2 = \epsilon_0\frac{w_2}{N_2}, \ldots, \epsilon_M = \epsilon_0\frac{w_M}{N_M}$ for some $\epsilon_0 > 0$ such that $\max\{\epsilon_1, \epsilon_2, \ldots, \epsilon_M\} < 1$. We observe that with this choice $\frac{\Delta\tilde{g}_1}{\epsilon_1} = \frac{\Delta\tilde{g}_2}{\epsilon_2} = \ldots = \frac{\Delta\tilde{g}_M}{\epsilon_M} = \frac{C}{\epsilon_0}$.

Reasoning as in the proof of Theorem 3.22 in Dwork and Roth [11], we conclude that if $\sigma \geq \sqrt{2\log\frac{1.25}{\delta}}\frac{\Delta\tilde{g}_m}{\epsilon_m} = \sqrt{2\log\frac{1.25}{\delta}}\frac{C}{\epsilon_0}$ then a step of the algorithm is:

$$(\epsilon_m, \delta) - \text{DP w.r.t. the training dataset } D_m. \tag{28}$$

When $\alpha < 1$, i.e., the batch size is smaller than the dataset size, we can invoke the privacy amplification theorem and each step of the algorithm becomes:

$$\left(O\left(\epsilon_m\alpha\right), \delta\alpha\right) - \text{DP w.r.t. } D_m. \tag{29}$$

Applying the moments account technique in [1] we see that over the whole training procedure for $K$ steps, the algorithm is:

$$\left(O\left(\epsilon_m\alpha\sqrt{K}\right), \delta\right) - \text{DP w.r.t. } D_m. \tag{30}$$

□

## A.4 Proof of Theorem 3.2

As a matter of notation, we write $L_D(f_\theta)$ as $L_D(\theta)$ to mean that the loss is evaluated over a model parameterized by $\theta$. Also for this proof we consider the more general scenario with multiple datasets introduced in A.2. We will refer to additional auxiliary datasets $\hat{D}_m = \{\hat{z}_m^{(i)}, i \in [N_m]\}$ and $\hat{D} = \bigcup_{m=1}^{M}\hat{D}_m$.

The proof of Theorem 3.2 relies on the following lemma, which we prove at the end of this section.

***Lemma 1*** *Under the assumption that the loss function is bounded, using our weighted ERM with training and reference data, it follows that:*

$$\mathbb{E}_{D\sim\mathbb{D}^N}\left[\sup_{\theta\in\Theta}\left|L_{\mathbb{D}}(\theta) - L_D^w(\theta)\right|\right]$$
$$\leq 2\sqrt{\frac{VCdim(\Theta)}{N_{eff}}} \cdot \sqrt{\gamma_2 + \log\left(\frac{N}{VCdim(\Theta)}\right)}$$

*where:*

$$L_D^w(\theta) = (1-w)L_{D_T}(\theta) + wL_{D_R}(\theta),$$
$$\gamma_2 = \max\left\{\frac{4}{VCdim(\Theta)}, 1\right\},$$
$$N_{eff} = \left[\frac{(1-w)^2}{N_T} + \frac{w^2}{N_R}\right]^{-1},$$

*$D_T, D_R \sim \mathbb{D}$, $N_T = |D_T|$, $N_R = |D_R|$, $D = D_T \cup D_R$, $N = |D|$, $\theta$ is a hypothesis in model class $\Theta$, and VCdim($\Theta$) is VCdim($\Theta$) is the VC-dimension of hypothesis class $F_\Theta = \{f_\theta : \theta \in \Theta\}$, as defined in Shalev-Shwartz and Ben-David [39].*

***Theorem 3.2*** *Under the assumption that the loss function is bounded in the range [0, 1], it follows that:*

$$L_{\mathbb{D}}(\theta_{WERM}) \leq \min_{\theta\in\Theta} L_{\mathbb{D}}(\theta)$$
$$+ 2\sqrt{\frac{VCdim(\Theta)}{N_{eff}}} \cdot \sqrt{\gamma_2 + \log\left(\frac{N}{VCdim(\Theta)}\right)} \tag{31}$$
$$+ \sqrt{\frac{2\ln 2/\delta}{N_{eff}}}$$

*with probability $\geq 1 - \delta$, where:*

$$\theta_{WERM} = \operatorname*{argmin}_{\theta\in\Theta} L_D^w(\theta)$$
$$L_D^w(\theta) = (1-w)L_{D_T}(\theta) + wL_{D_R}(\theta),$$
$$\gamma_2 = \max\left\{\frac{4}{VCdim(\Theta)}, 1\right\},$$
$$N_{eff} = \left[\frac{(1-w)^2}{N_T} + \frac{w^2}{N_R}\right]^{-1},$$

*$D_T \sim \mathbb{D}^{N_T}$, $D_R \sim \mathbb{D}^{N_R}$, $D = D_T \cup D_R$, $N = |D|$, and VCdim($\Theta$) is the VC-dimension of hypothesis class $F_\Theta = \{f_\theta : \theta \in \Theta\}$.*

Proof. We denote:

$$A(N_{\text{eff}}) = 2\sqrt{\frac{\text{VCdim}(\Theta)}{N_{\text{eff}}}} \cdot \sqrt{\gamma_2 + \log\left(\frac{N}{\text{VCdim}(\Theta)}\right)}$$

Lemma 1 proves the following:

$$\mathbb{E}_{D\sim\mathbb{D}^N}\left[\sup_{\theta\in\Theta}\left|L_{\mathbb{D}}(f_\theta) - L_D^w(f_\theta)\right|\right] \leq A(N_{\text{eff}})$$

In this proof, we will use the weaker bound:

$$\mathbb{E}_{D\sim\mathbb{D}^N}\left[\sup_{\theta\in\Theta}\left(L_{\mathbb{D}}(f_\theta) - L_D^w(f_\theta)\right)\right] \leq A(N_{\text{eff}}). \tag{32}$$

We observe that changing a point in dataset $D_m$ leads $L_{\mathbb{D}}(\theta) - L_D^w(\theta)$ to change by at most $\frac{w_m}{N_m}$ in absolute value. In fact, call $\tilde{D}$ the dataset where a single point $z_m^{(i)}$ in $D$ has been replaced by point $\tilde{z}_m^{(i)}$:

$$\left|\sup_{\theta\in\Theta} L_{\mathbb{D}}(\theta) - L^w_D(\theta) - \left(\sup_{\theta\in\Theta} L_{\mathbb{D}}(\theta') - L^w_{\tilde{D}}(\theta')\right)\right| \tag{33}$$

$$\leq \left|\sup_{\theta\in\Theta}\left(L_{\mathbb{D}}(\theta) - L^w_D(\theta) - L_{\mathbb{D}}(\theta) + L^w_{\tilde{D}}(\theta)\right)\right| \tag{34}$$

$$= \left|\sup_{\theta\in\Theta}\left(L^w_{\tilde{D}}(\theta) - L^w_D(\theta)\right)\right| \tag{35}$$

$$= \left|\sup_{\theta\in\Theta}\frac{w_m}{N_m}\left(\ell(\theta,\tilde{z}_m^{(i)}) - \ell(\theta,z_m^{(i)})\right)\right| \tag{36}$$

$$= \sup_{\theta\in\Theta}\left|\frac{w_m}{N_m}\left(\ell(\theta,\tilde{z}_m^{(i)}) - \ell(\theta,z_m^{(i)})\right)\right| \tag{37}$$

$$\leq \frac{w_m}{N_m} \tag{38}$$

We can then apply McDiarmid's inequality [39] and we obtain:

$$\begin{aligned}
&\text{Prob}\Big(\sup_{\theta\in\Theta}\left(L_{\mathbb{D}}(\theta) - L^w_D(\theta)\right) \\
&\quad \leq \mathbb{E}\Big[\sup_{\theta\in\Theta} L_{\mathbb{D}}(\theta) - L^w_D(\theta)\Big] + \epsilon\Big) \\
&\quad \geq 1 - \exp\left(\frac{-2\epsilon^2}{\sum_{m=1}^{M}\sum_{i=1}^{N_m}\left(\frac{w_m}{N_m}\right)^2}\right) \\
&\quad = 1 - \exp\left(-2\epsilon^2 N_{\text{eff}}\right)
\end{aligned} \tag{39}$$

If we let $\delta = \exp\left(-2\epsilon^2 N_{\text{eff}}\right)$ we obtain:

$$\begin{aligned}
&\text{Prob}\Big(\sup_{\theta\in\Theta}\left(L_{\mathbb{D}}(\theta) - L^w_D(\theta)\right) \\
&\quad < \mathbb{E}\Big[\sup_{\theta\in\Theta} L_{\mathbb{D}}(\theta) - L^w_D(\theta)\Big] \\
&\quad + \sqrt{\frac{1}{2N_{\text{eff}}}\ln\frac{1}{\delta}}\Big) \geq 1 - \delta
\end{aligned} \tag{40}$$

and using (32):

$$\begin{aligned}
&\text{Prob}\Big(\sup_{\theta\in\Theta}\left(L_{\mathbb{D}}(\theta) - L^w_D(\theta)\right) \leq \\
&\quad A(N_{\text{eff}}) + \sqrt{\frac{1}{2N_{\text{eff}}}\ln\frac{1}{\delta}}\Big) \geq 1 - \delta
\end{aligned} \tag{41}$$

As this is true for the $\sup_{\theta\in\Theta} L_{\mathbb{D}}(\theta) - L^w_D(\theta)$ it is true in particular for $\theta_{\text{WERM}}$:

$$\text{Prob}\left(L_{\mathbb{D}}(\theta_{\text{WERM}}) - L^w_D(\theta_{\text{WERM}}) \leq A(N_{\text{eff}}) + \sqrt{\frac{1}{2N_{\text{eff}}}\ln\frac{1}{\delta}}\right) \geq 1-\delta. \tag{42}$$

Consider $\theta^* \in \text{argmin}_{\theta\in\Theta} L_{\mathbb{D}}(\theta)$:

$$L^w_D(\theta^*) = \sum_{m=1}^{M}\sum_{i=1}^{N_m}\frac{w_m}{N_m}\ell(\theta^*, z_m^{(i)}),$$

then it is a sum of independent random variables $\gamma_m^{(i)} = \frac{w_m}{N_m}\ell(\theta^*, z_m^i) \in \left[0, \frac{w_m}{N_m}\right]$. By applying Hoeffding's inequality:

$$\text{Prob}\Big(L^w_D(\theta^*) - L_{\mathbb{D}}(\theta^*) \leq \epsilon\Big) \tag{43}$$

$$\geq 1 - \exp\left(-\frac{2\epsilon^2}{\sum_{m=1}^{M}\sum_{i=1}^{N_m}\left(\frac{w_m}{N_m}\right)^2}\right) \tag{44}$$

$$= 1 - \exp(-2\epsilon^2 N_{\text{eff}}) \tag{45}$$

Similarly to above, we conclude:

$$L^w_D(\theta^*) - L_{\mathbb{D}}(\theta^*) \leq \sqrt{\frac{1}{2N_{\text{eff}}}\ln\frac{1}{\delta}} \text{ w.p.} \geq 1 - \delta \tag{46}$$

Both (42) and (46) hold w.p. $1 - 2\delta$ ($P(A \cap B) = P(A) + P(B) - P(A \cup B) \geq P(A) + P(B) - 1$), thus:

$$\begin{aligned}
&L_{\mathbb{D}}(\theta_{\text{WERM}}) - L_{\mathbb{D}}(\theta^*) \\
&= L_{\mathbb{D}}(\theta_{\text{WERM}}) - L^w_D(\theta_{\text{WERM}}) + L^w_D(\theta_{\text{WERM}}) \\
&\qquad - L^w_D(\theta^*) + L^w_D(\theta^*) - L_{\mathbb{D}}(\theta^*) \\
&\leq L_{\mathbb{D}}(\theta_{\text{WERM}}) - L^w_D(\theta_{\text{WERM}}) + L^w_D(\theta^*) - L_{\mathbb{D}}(\theta^*) \\
&\leq A(N_{\text{eff}}) + 2\sqrt{\frac{1}{2N_{\text{eff}}}\ln\frac{1}{\delta}} \\
&\text{w.p.} \geq 1 - 2\delta
\end{aligned}$$

Replacing $\delta$ by $\frac{\delta}{2}$, we obtain the theorem. □

### A.4.1 Proof of Lemma 1.

Proof. Substituting the expectation over $L_{\hat{D}}(\theta)$ for $L_{\mathbb{D}}(\theta)$ and using Jensen's inequality, it follows that:

$$\mathbb{E}_D\left[\sup_{\theta\in\Theta}\left|L_{\mathbb{D}}(\theta) - L^w_D(\theta)\right|\right] \tag{47}$$

$$\leq \mathbb{E}_{D,\hat{D}}\left[\sup_{\theta\in\Theta}\left|L_{\hat{D}}(\theta) - L^w_D(\theta)\right|\right] \tag{48}$$

$$= \mathbb{E}_{D,\hat{D}}\left[\sup_{\theta\in\Theta}\left|\sum_{m=1}^{M}\sum_{i=1}^{N_m} w_m\left(\ell(\theta; z_m^{(i)}) - \ell(\theta; \hat{z}_m^{(i)})\right)\right|\right] \tag{49}$$

$$= \mathbb{E}_{D,\hat{D}}\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta}\left|\sum_{m=1}^{M}\sum_{i=1}^{N_m}\sigma_m^{(i)}\cdot w_m\left(\ell(\theta; z_m^{(i)}) - \ell(\theta; \hat{z}_m^{(i)})\right)\right|\right] \tag{50}$$

where $\sigma_m^{(i)}$ is a random variable drawn from the uniform distribution over $\{\pm 1\}$ that is uniquely sampled for each $m \in [M]$ and $i \in [N_m]$. Next, we fix $D$ and $\hat{D}$ and let C be the instances appearing in the two datasets. As defined in Definition 6.2 from Shalev-Shwartz and Ben-David [39], we assign $\Theta_C$ to be the restriction of $\Theta$ to C. Thus:

$$\begin{aligned}
&\mathbb{E}_D\left[\sup_{\theta\in\Theta}\left|L_{\mathbb{D}}(\theta) - L^w_D(\theta)\right|\right] \\
&\quad \leq \mathbb{E}_{D,\hat{D}}\mathbb{E}_\sigma\left[\sup_{\theta\in\Theta_C}\left|\sum_{m=1}^{M}\sum_{i=1}^{N_m}\sigma_m^{(i)}\cdot w_m\left(\ell(\theta; z_m^{(i)}) - \ell(\theta; \hat{z}_m^{(i)})\right)\right|\right]
\end{aligned} \tag{51}$$

We fix some $\theta \in \Theta_C$ and denote $\gamma_m^{(i)} = \sigma_m^{(i)} \cdot w_m \left( \ell(\theta; z_m^{(i)}) - \ell(\theta; z_m'^{(i)}) \right)$ for $m \in [M]$ and $i \in [N_m]$. Without a loss of generality, by assuming the bound on the loss is between 0 and 1, we have that $\mathbb{E}[\gamma_m^{(i)}] = 0$ and $\gamma_m^{(i)} \in [-w_m, w_m]$. Given that each $\gamma_m^{(i)}$ is an independent random variable, we invoke Hoeffding's inequality to say that:

$$\mathbb{P}\left[ \left| \sum_{m=1}^{M} \sum_{i=1}^{N_m} \sigma_m^{(i)} \cdot w_m \left( \ell(\theta; z_m^{(i)}) - \ell(\theta; \hat{z}_m^{(i)}) \right) \right| \geq \rho \right] \leq 2 \exp(-2 N_{\text{eff}} \rho^2) \tag{52}$$

where $N_{\text{eff}} = \left( \sum_{m=1}^{M} \sum_{i=1}^{N_m} w_m^2 \right)^{-1}$. Taking the the union bound over $\theta \in \Theta_C$, invoking Lemma A.4 in Shalev-Shwartz and Ben-David [39], we have:

$$\begin{aligned} \mathbb{E}\left[ \sup_{\theta \in \Theta_C} \left| \sum_{m=1}^{M} \sum_{i=1}^{N_m} \sigma_m^{(i)} \cdot w_m \left( \ell(\theta; z_m^{(i)}) - \ell(\theta; \hat{z}_m^{(i)}) \right) \right| \right] &\leq \frac{4 + \sqrt{\log(\Theta_C)}}{\sqrt{2N_{\text{eff}}}} \\ &\leq \frac{4 + \sqrt{\log(\tau_\Theta(N))}}{\sqrt{2N_{\text{eff}}}} \end{aligned} \tag{53}$$

where $\tau_\Theta$ is the growth function of $\Theta$ as defined by Definition 6.9 in Shalev-Shwartz and Ben-David [39]. Applying Lemma 6.10 in Shalev-Shwartz and Ben-David [39] to upper bound $\tau_\Theta$ with respect to the VC-dimension and using the same steps as the proof of Lemma A.1 in Marfoq et al. [29], we arrive at our generalization bound:

$$\begin{aligned} &\mathbb{E}_D \left[ \sup_{\theta \in \Theta} \left| L_{\mathbb{D}}(\theta) - L_D^w(\theta) \right| \right] \\ &\leq 2 \sqrt{\frac{\text{VCdim}(\Theta)}{N_{\text{eff}}}} \cdot \sqrt{\gamma_2 + \log\left( \frac{N}{\text{VCdim}(\Theta)} \right)} \end{aligned} \tag{54}$$

where $\gamma_2 = \max\left\{ \frac{4}{\text{VCdim}(\Theta)}, 1 \right\}$. The result we present in the paper is for the case where $M = 2$, which represents having access to training and reference data.

□

# B IMPLEMENTATION DETAILS

## B.1 Adversarial Regularization

The issue most relevant for our work is that the two players are not really alternating their optimization steps to solve the minmax problem, as described in the algorithm from Nasr et al. [30]. Instead, the code version updates the attack model for 52 batches, then the classifier for two batches, repeating this process 76 times, over the course of 20 rounds. By frequently alternating between the classifier and attack model, the training procedure in the code corresponds more closely to taking the entire gradient over the objective function and updating both models simultaneously. After evaluating different variants that utilize some aspects from the general description and other aspects from the code implementation, the best results are obtained by updating each model with a higher frequency.

Additionally, the released code contains a bug where not all batches are observed an equal number of times. To address this issue and remain faithful to the goal of alternating model updates with a higher frequency, we use the following procedure:

(1) Define the ratio that the attack model should be trained compared to the classifier
(2) For each batch in an epoch, sample from a Bernoulli distribution using the defined training ratio
(3) If a 0 is drawn, train the attack model; if a 1 is drawn, train the classifier
(4) Repeat steps 1-3 until the conclusion of the epoch

Taking into account the bug, we measured the effective attack model to classifier training ratio to be 22:1. We produce the same results using the above training procedure with a ratio of 20:1.

As a final point, in the original formulation, the attack model inputs the raw feature vectors during training. Alternatively, in the released code version, the attack model only has access to the ground-truth labels and confidence-vector outputs. We follow the code implementation in our experiments.

## B.2 MMD-based Regularization

As there is no released code associated with MMD [26], we implement the method ourselves based on the paper's description. It is written that "In the implementation of our MMD loss, we reduce the difference between the probability vector distributions of members and non-members for the same class. That is, a batch of training samples and a batch of validation samples in the same class are used together to compute the MMD score." Typically, during training one calculates the gradients of the objective function with respect to a mini-batch that is sampled uniformly from the training data. Selecting a mini-batch the only has data with the same label results in a model training procedure that stagnates and fails to learn. How does MMD ensure that the regularization term only compares data with the same label? We reached out to the authors to ask for their exact training strategy but did not receive a reply. Accordingly, we propose the following procedure that is faithful to the description provided in the text:

(1) Sample a mini-batch uniformly from the training data using a sufficiently large batch size to include multiple instances of each label
(2) Sample an equally sized mini-batch uniforming of reference data
(3) Identify all unique labels in the training data mini-batch
(4) For each distinct label in the training data mini-batch, calculate the MMD between instances with this label in the training and reference mini-batches
(5) Use the average MMD across all distinct labels as the regularization term

Additionally, as the value of the variance for the Gaussian kernel used in (9) is not specified, we experimented with several values before deciding to use 1.0.

# C DATASET DESCRIPTIONS

## C.1 Purchase100

This dataset, derived from the "acquire valued shopper" challenge on Kaggle, consists of shopping records for several thousand individuals. [11] Using this dataset, participants aim to find discounts that can attract shoppers to buy new products. We use the same pre-processed version of the dataset as in Nasr et al. [30]. In total, there are 197,324 data points, where each entry contains 600 binary features that indicate whether the shopper has purchased a certain item. Based on these binary features vectors, the data is clustered into 100 classes that represent distinct categories of shoppers. The prediction task is to determine the class associated to each shopper.

## C.2 Texas100

This dataset, released by the Texas Department of State Health Services, consists of information regarding inpatient stays at several health facilities. Each record encodes the external cause of injury (e.g., suicide, drug misuse), the diagnosis (e.g., schizophrenia, depression), the procedures underwent by the patient (e.g., X-ray, surgery), the length of stay, personal information relating to the patient (e.g., gender, age, race), and hospital-specific identifiers (e.g., hospital ID). We use the same pre-processed version of the dataset as in Nasr et al. [30]. In total, there are 67,330 data points, where each entry contains 6,170 binary features that indicate whether a patient underwent any of the 100 most common medical procedures. Based on these binary features vectors, the data is clustered into 100 classes that represent distinct categories of patients. The prediction task is to determine the class associated to each patient.

## C.3 CIFAR100

This dataset is a major benchmark generally used to in image recognition. It contains 60,000 images and 100 classes. Each image is composed of 32 x 32 color pixels.

# D MEMBERSHIP INFERENCE ATTACK DETAILS

## D.1 Assumptions in Membership Inference Attacks

In Table 5, we present a list of the most well-known MIAs categorized by their assumption settings.

## D.2 Design Details

### D.2.1 Gap Attack.

In the standard case where an attack is evaluated using two equally sized datasets, $D_T^{\text{adv}}$ and $D_{\overline{T}}^{\text{adv}}$, the overall attack accuracy for he gap attack (5) can be computed as follows:

$$\text{acc}_{\text{gap attack}} = \frac{1}{2} + \frac{\text{acc}_T - \text{acc}_{\overline{T}}}{2} \tag{55}$$

where $\text{acc}_T$ and $\text{acc}_{\overline{T}}$ correspond to the target model's prediction accuracy computed over $D_T^{\text{adv}}$ and $D_{\overline{T}}^{\text{adv}}$, respectively [8]. As a model's loss is negatively correlated with its accuracy, a model that has poor generalization (i.e., relatively low train loss compared to test loss) will be be vulnerable to the gap attack. Thanks to its simplicity and demonstrated effectiveness [42], the gap attack has been suggested as a baseline attack against which all proposed defenses should be evaluated [8].

### D.2.2 Threshold-based Attacks.

When the adversary has access to a model's entire confidence-vector output, various other metrics, such as the entropy of the output distribution, can be utilized instead of the confidence value for threshold-based attacks [41]. Estimating a class-dependent threshold or class-specific metric (e.g., modified-entropy [42]) is possible when the adversary knows the target data points' ground-truth labels. It is sometimes assumed that the adversary has access to additional data from the underlying distribution in the form of known member and non-member data, which allows for the estimation of more precise thresholds [41, 42, 51].

Threshold-based attacks exploit the phenomenon that target model's outputs for training data are usually distinguishable from target model's outputs for non-training data (e.g., confidence values can be more skewed in the first case). Moreover, all variations of threshold-based attacks, as well as other types of MIAs, can be improved by querying the target model with transformed or augmented versions of target data points and predicting membership based on the aggregated output [5, 8].

# E ADDITIONAL FIGURES/TABLES

In Figure 4, we show the behavior for $N_T/N_R = 0.25$ and its reciprocal $N_T/N_R = 4$. In Figure 5, we present the utility-privacy curves where the model instances are selected on the basis of validation data. In Figure 6, we present the utility-privacy curves using a neural network attack to evaluate the MIA accuracy. We follow the same methodology as [30] in designing the neural network attack. The results do not seem better than the threshold-based MIAs we evaluate against. However, one must consider that our threshold-based MIAs optimize the threshold with the exact same information that the neural network attack learns from. In Figure 7, we present the utility-privacy curves including the curve for WERM-ES, without highlighting trends, and without removing low test accuracy outliers for AdvReg and MMD. In Table 6, we show a comparison of the Pearson correlation coefficient calculated for each dataset individually. We can see that WERM outperforms both AdvReg and MMD on all three datasets. The correlation difference is particularly significant on CIFAR100.

[11] https://www.kaggle.com/competitions/acquire-valued-shoppers-challenge/overview

| attack name | ground-truth training data | population data | vector of attack | ground-truth label |
|---|---|---|---|---|
| Gap Attack [51] | no | no | predicted label | yes |
| Confidence Attack [51] | no | no | largest confidence value | no |
| Entropy Attack [41, 42] | no | no | confidence-vector | no |
| Modified-entropy Attack [42] | no | no | confidence-vector | yes |
| Neural Network Attack [41] | yes | yes | confidence-vector | yes |
| Distillation Attack [50] | no | yes | model loss | yes |
| Likelihood Ratio Attack (LiRA) [5] | maybe | yes | model loss | yes |
| Leave-one-out Attack [50] | yes | yes | model loss | yes |

**Table 5: Analyzing black-box membership inference attacks by assumptions. The term "maybe" means that using additional knowledge derived from this assumption can be directly integrated into the attack to improve performance.**

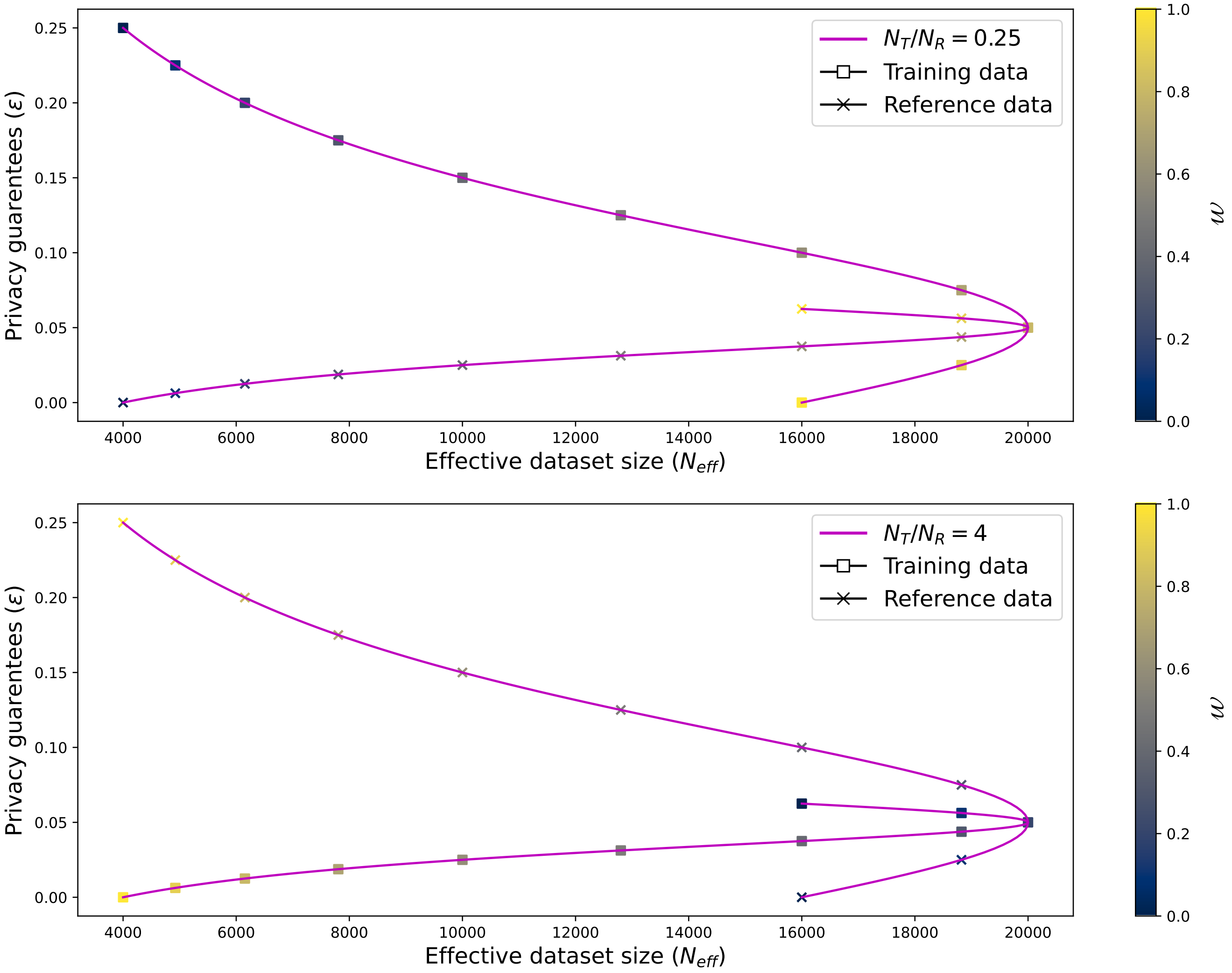

**Figure 4: Theoretical bounds for** $\frac{N_T}{N_R} = 0.25$ **and its inverse** $\frac{N_T}{N_R} = 4$.

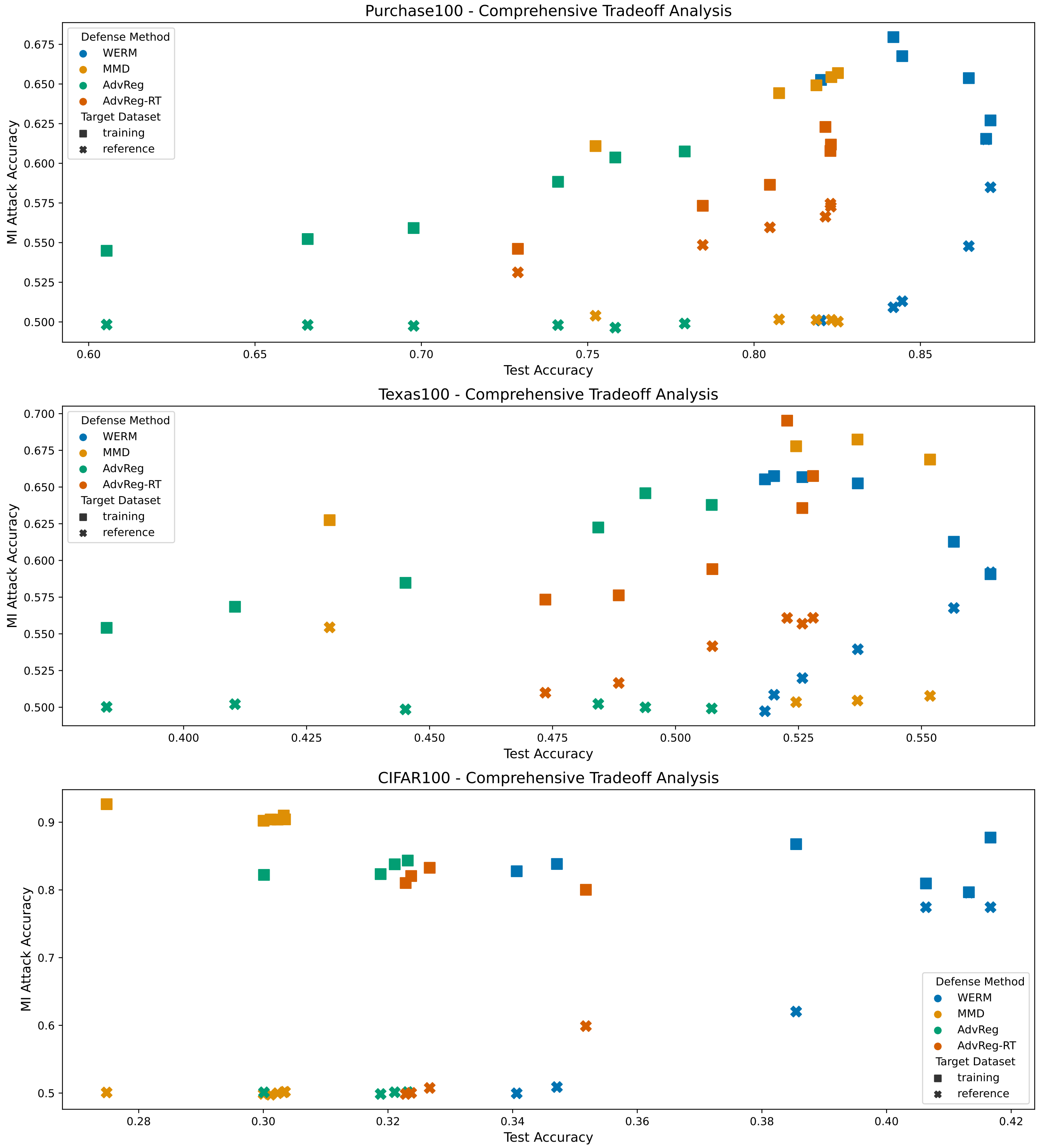

**Figure 5: Utility-Privacy tradeoffs obtained by various empirical privacy defenses for the Purchase100, Texas100, and CIFAR100 datasets. The test accuracy of a defended classifier is measured using unseen test data and the MIA accuracy on training and reference data is evaluated with a threshold-based using confidence values (Eq. 6). Each point on the curve represents the evaluation of a model instance using a distinct regularization value (for AdvReg, AdvReg-RT, and MMD) and reference data weight value (for WERM). In these curves, the model instance is selected on the basis of validation data.**

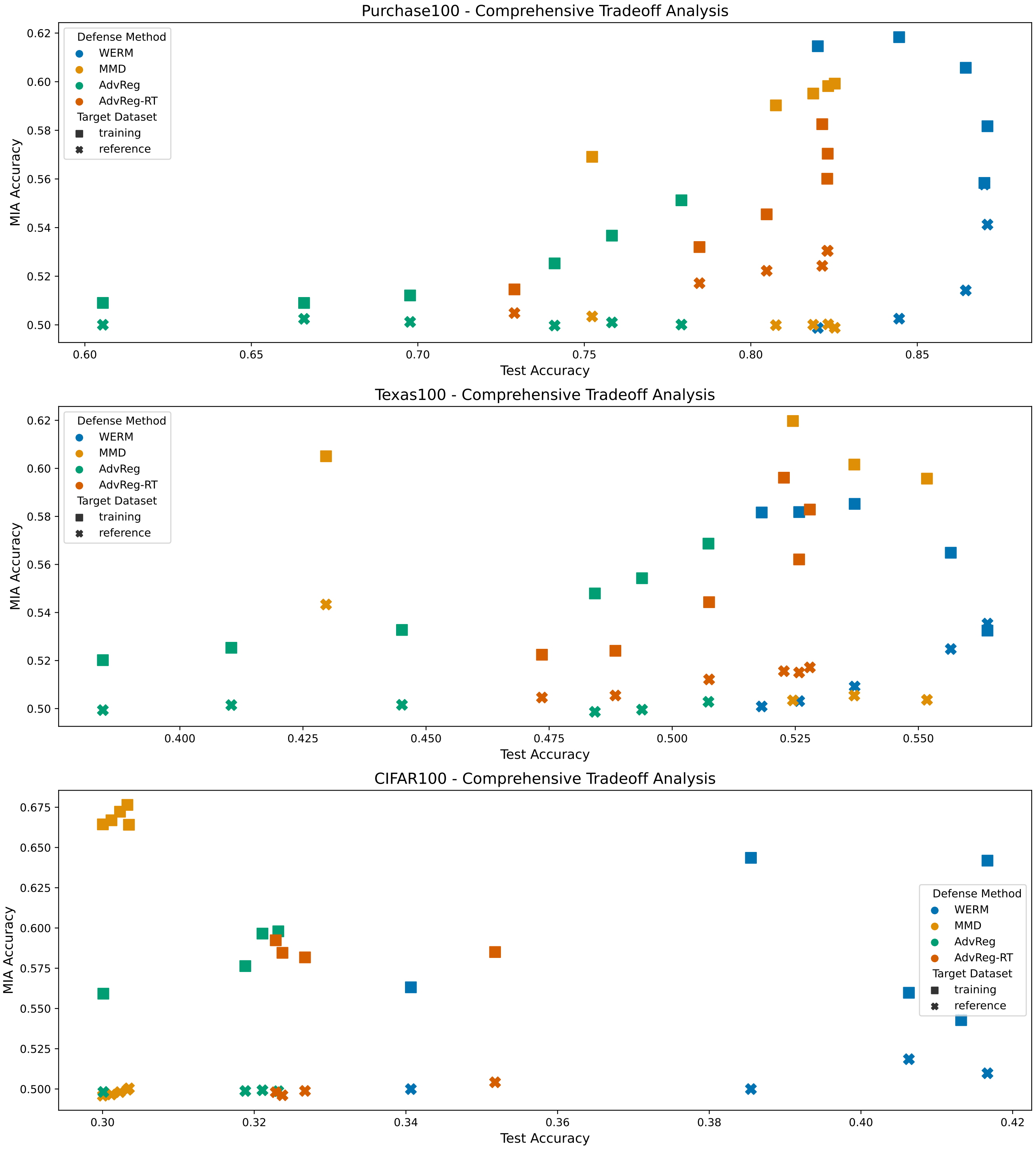

**Figure 6: Utility-Privacy tradeoffs obtained by various empirical privacy defenses for the Purchase100, Texas100, and CIFAR100 datasets. The test accuracy of a defended classifier is measured using unseen test data and the MIA accuracy on training and reference data is evaluated with a neural network attack [30, 41]. Each point on the curve represents the evaluation of a model instance using a distinct regularization value (for AdvReg, AdvReg-RT, and MMD) and reference data weight value (for WERM). In these curves, the model instance is selected on the basis of validation data.**

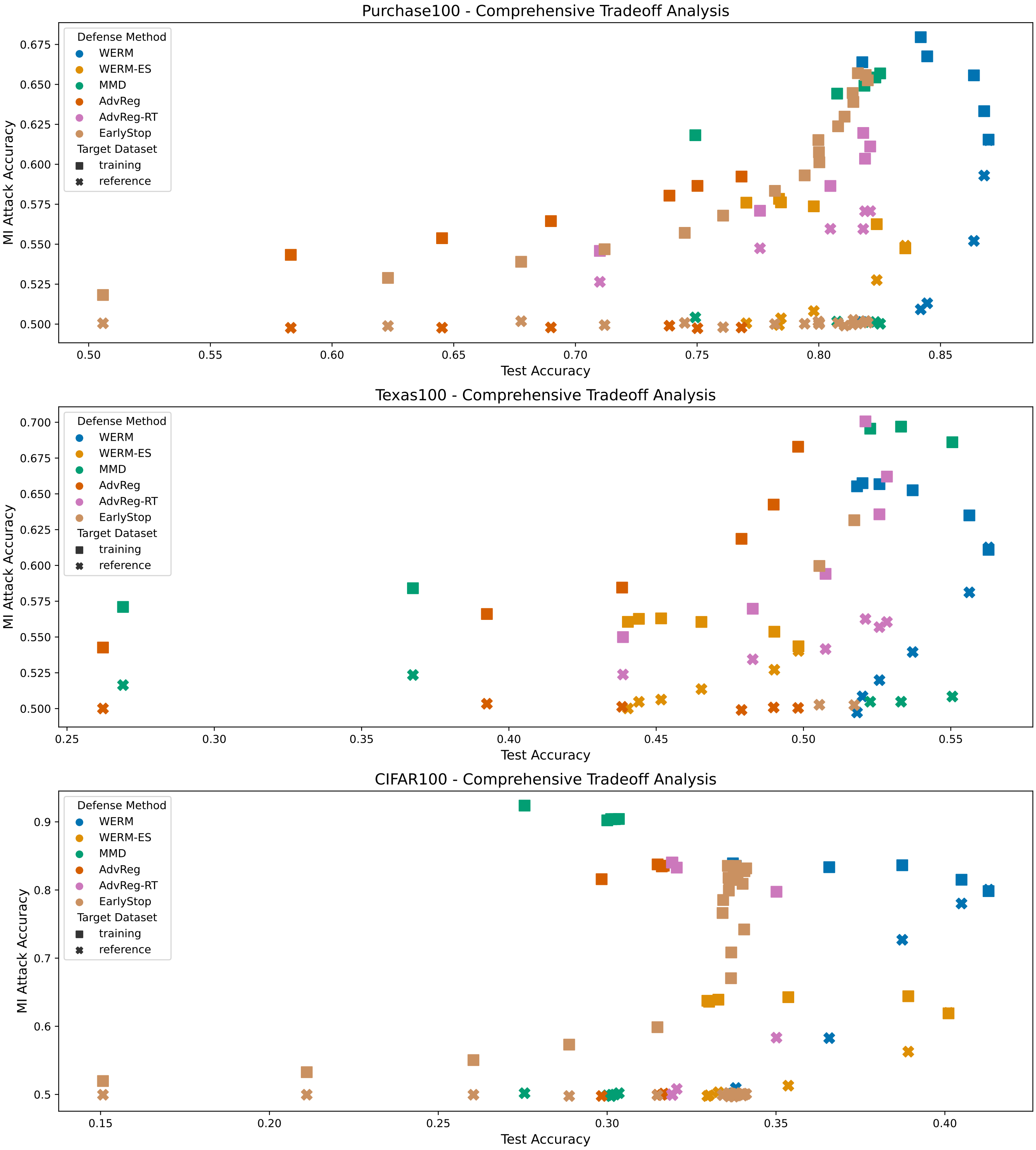

**Figure 7: This figure examines the utility-privacy curves produced by all versions of empirical privacy defenses for the Purchase100, Texas100, and CIFAR100 datasets. The test accuracy of a defended classifier is measured using unseen test data and the MIA accuracy on training and reference data is evaluated with a threshold-based using confidence values (Eq. 6). Each point on the curve represents the evaluation of a model instance using a distinct regularization value.**

**Table 6: Comparison of the Pearson Correlation Coefficient (PCC) between the training-reference data desired privacy ratio (as determined by the choice of $w$ or $\lambda$) and the empirical privacy ratio (as measured by a MIA) for WERM, AdvReg, and MMD. The coefficient is computed for the Purchase100, Texas100, and CIFAR100 datasets individually and across all datasets (i.e., the theoretical and empirical relative privacy values are aggregated before calculating the PCC).**

| defense | dataset | Pearson Correlation Coefficient |
|---|---|---|
| WERM | Purchase100 | 1.0 |
| | Texas100 | 0.99 |
| | CIFAR100 | 0.99 |
| | Overall | 0.84 |
| AdvReg | Purchase100 | 0.87 |
| | Texas100 | 0.09 |
| | CIFAR100 | -0.97 |
| | Overall | 0.07 |
| MMD | Purchase100 | 0.93 |
| | Texas100 | 0.86 |
| | CIFAR100 | 0.49 |
| | Overall | 0.48 |