DP-InstaHide: Data Augmentations Provably Enhance Guarantees Against Dataset Manipulations

Anonymous authors
Paper under double-blind review

Abstract

Data poisoning and backdoor attacks manipulate training data to induce security breaches in a victim model. These attacks can be provably deflected using differentially private (DP) training methods, although this comes with a sharp decrease in model performance. The InstaHide method has recently been proposed as an alternative to DP training that leverages supposed privacy properties of the mixup augmentation, although without rigorous guarantees. In this paper, we rigorously show that k-way mixup provably yields at least k times stronger DP guarantees than a naive DP mechanism, and we observe that this enhanced privacy guarantee is a strong foundation for building defenses against poisoning.

1 Introduction

As the capabilities of machine learning systems expand, so do their training data demands. To satisfy this massive data requirement, developers create automated web scrapers that download data without human supervision. The lack of human control over the machine learning pipeline may expose systems to *poisoned* training data that induces pathologies in models trained on it. Data poisoning and backdoor attacks may degrade accuracy or elicit incorrect predictions in the presence of a triggering visual feature (Shafahi et al., 2018; Chen et al., 2017).

To combat this threat model, a number of defenses against data poisoning have emerged. Certified defenses based on *differential privacy* (DP) provably desensitize models to small changes in their training data by adding noise to either the data or the gradients used by their optimizer (Ma et al., 2019). When a model is trained using sufficiently strong DP, it is not possible to infer whether a small collection of data points were present in the training set by observing model behaviors, and it is therefore not possible to significantly alter model behaviors by introducing a small number of poisoned samples. DP methods generally require adding large amounts of noise to data samples (or gradients), which can significantly degrade model performance.

As an alternative to DP, the InstaHide algorithm (Huang et al., 2020b) aims to create dataset privacy by averaging random image pairs and then multiplying the results by a random mask. Random averaging of images before training is known as mixup regularization (Zhang et al., 2017), and can improve model performance. By leaning on mixup rather than noise to create randomness, InstaHide seeks to avoid the drastic performance trade offs of differential privacy. Unfortunately, the privacy claims of InstaHide were not well founded, and the method was quickly broken (Carlini et al., 2020).

In this paper, we formally prove that mixup augmentation *enhances* the guarantees of classical DP methods, and the privacy benefits are highly advantageous for defending against dataset manipulation. Our proposed method, DP-InstaHide, applies k-way mixup before adding Laplacian noise, resulting in a factor of k improvement in ϵ-differential privacy bounds (*i.e.*, a k-fold reduction in ϵ) over the traditional Laplacian mechanism. When used to defend against poisoning and backdoor attacks, we find that mixup-based privacy yields a favorable robustness accuracy trade-off compared to other strong defenses. We benchmark the empirical performance of DP-InstaHide, in addition to several extensions based on other related augmentations (Yun et al., 2019; Zhang et al., 2017; Gong et al., 2020).

This paper is the full version of a preliminary short (4-page) paper in which we only consider data augmentations as an empirical defense without any formal guarantees [citation omitted]. We go beyond that preliminary work to characterize the privacy benefits of DP-InstaHide theoretically, and we greatly extend the empirical analysis of data augmentation defenses against data poisoning attacks.

1.1 Related Work

Broadly speaking, data poisoning attacks aim to compromise the performance of a network by maliciously modifying the data on which the network is trained. Data poisoning attacks vary in their goals, methods, and settings. In general, the goals of a data poisoning attack can be divided into *indiscriminate* attacks, which seek to degrade general test-time performance of a network, and *targeted* attacks, which aim to cause a specific example, or set of examples, to be misclassified (Barreno et al., 2010).

Early work on data poisoning often focused on indiscriminate attacks in simple settings, such as support vector machines, logistic regression models, principle component analysis, or clustering algorithms (Muñoz-González et al., 2017; Xiao et al., 2015; Biggio et al., 2012; Koh et al., 2018).

However, these early methods do not scale well to modern deep networks (Huang et al., 2020a). Many recent works instead focus on targeted attacks and backdoor attacks, which are easier to scale and can be more insidious since they do not lead to any noticeable degradation in validation accuracy, making them harder to detect (Geiping et al., 2020). Accordingly, in this work, we focus on defending against targeted and backdoor attacks. Within these attacks, however, there still exists a wide range of methods and settings. Below, we detail a few categories of attacks. A comprehensive enumeration of backdoor attacks, data poisoning attacks, and defense can be found in Goldblum et al. (2020).

A **feature collision** attack occurs when the attacker modifies training samples, so they collide with, or surround, a target test-time image. *Poison Frogs* (Shafahi et al., 2018) optimizes poisons to minimize the ℓ_2 distance in feature space between the poisoned and target features, while also including a regularization term on the size of the perturbations. Newer methods like *convex polytope* (Zhu et al., 2019) and *bullseye polytope* (Aghakhani et al., 2020) surround the target image in feature space to improve the stability of poisoning. All these methods work primarily in the transfer learning setting, where a known feature extractor is fixed and a classification layer is fine-tuned on the perturbed data.

From-scratch attacks modify training data to cause targeted misclassification of preselected test time images. Crucially, these attacks work in situations where a deep network is *a priori* trained on modified data, rather than being pretrained and subsequently fine-tuned on poisoned data. *MetaPoison* (Huang et al., 2020a) optimizes poisons by unrolling training iterations to solve a bi-level optimization problem. *Witches' Brew* (Geiping et al., 2020) approximately solve the bi-level optimization problem using a gradient alignment objective.

Backdoor attacks involve inserting a "trigger," often a fixed patch, into training data. Attackers can then add the same patch to data at test time to fool the network into misclassifying modified images as the target class. Some forms of backdoor attacks will patch a number of training images with a small pattern or even modify just a single pixel (Gu et al., 2017; Tran et al., 2018b). More complex attacks, like hidden-trigger backdoors (Saha et al., 2020), adaptively modify the training data to increase the success of the additive patch at test time.

Conversely, a variety of defenses against poisoning attacks have also been proposed. Many defenses to targeted poisoning attacks can broadly be classified as *filtering defenses*, which either remove or relabel poisoned data. These methods rely on the tendency of poisoned data to differ sufficiently from clean data in feature space. Intuitively, one could use a pretrained network as a feature extractor to sort out poison from the clean data. Once the poisoned data is found and isolated in feature space, it is removed from the dataset and the model is retrained from scratch. Conveniently, filtering defenses do not require any external source of trusted clean data and work even if the feature extractor is trained on poisoned data.

Among filtering defenses, *Spectral Signatures* (Tran et al., 2018b; Paudice et al., 2018) filter data based on which points have the highest correlation with the top right singular vector of the feature covariance matrix.

Activation Clustering (Chen et al., 2018) instead uses k-means clustering to separate feature space, relying on the heuristic that poisons tend to cluster in feature space. *DeepKNN* (Peri et al., 2019) relabels outlier data in feature space according to a k-nearest neighbors algorithm, hoping to diminish the effects of poison using the same heuristic that poisoned data are outliers in feature space.

Unfortunately, filtering defenses have proven weak against more advanced attacks, especially in the from-scratch setting (Geiping et al., 2020) and may be nullified by adaptive attacks that carefully circumvent detection (Koh et al., 2018).

Certified defenses avoid the possibility of breaking under adaptive attacks using robust mechanisms such as randomized smoothing or by partitioning the training data and individually training classifiers on each partition (Weber et al., 2020; Levine & Feizi, 2020).

Another class of principled defenses use differentially private SGD, where training gradients are clipped and noised thus diminishing the effects of poisoned gradient updates. However, these defenses have been shown to fail against advanced attacks, as they often lead to significant drops in clean validation accuracy (Geiping et al., 2020).

Outside of data poisoning, Lee et al. (2019) connect data augmentation and privacy by using tools for Rényi differential privacy for subsampling (Wang et al., 2019) to analyze Rényi bounds for image mixtures with Gaussian noise. While these bounds can readily be converted into differential privacy guarantees, they suffer from numeric instability and tend to be loose in the low privacy regime, where validation accuracy is maintained.

2 DP-INSTAHIDE: A MIXUP DEFENSE WITH PROVABLE DIFFERENTIAL PRIVACY ADVANTAGES

The original InstaHide method (Huang et al., 2020b) attempted to privatize data by first applying mixup, and then multiplying the results by random binary masks. While the idea that mixup enhances the privacy of a dataset is well founded, the original InstaHide scheme lies outside of the classical differential privacy framework and is now known to be insecure (Carlini et al., 2020). We propose a variant of the method, DP-InstaHide, which replaces the multiplicative random mask with additive Laplacian noise. The resulting method comes with a differential privacy guarantee that enables us to quantify and analyze the privacy benefits of mixup augmentation.

Differential privacy, developed by Dwork et al. (2014), aims to prevent the leakage of potentially compromising information about individuals present in released data sets. By utilizing noise and randomness, differentially private data release mechanisms are provably robust to any auxiliary information available to an adversary.

Formally, let $\mathcal{M} : \mathcal{D} \to \mathcal{R}$ be a random mechanism, mapping from the space of datasets to a co-domain containing potential outputs of the mechanism. We consider a special case where $\mathcal{R}$ is another space of datasets, so that $\mathcal{M}$ outputs a synthetic dataset. We say two datasets $D, D' \in \mathcal{D}$ are adjacent if they differ by at most one element, that is D' has one fewer, one more, or one element different from D.

Then, $\mathcal{M}$ is (ϵ, δ)-differentially private if it satisfies the following inequality for any $U \subseteq \mathcal{R}$:

$$\mathbb{P}[\mathcal{M}(D) \in U] \leq e^{\epsilon}\mathbb{P}[\mathcal{M}(D') \in U] + \delta. \tag{1}$$

Intuitively, the inequality and symmetry in the definition of dataset adjacency tells us that the probability of getting any outcome from $\mathcal{M}$ does not strongly depend on the inclusion of any individual in the dataset. In other words, given any outcome of the mechanism, a strong privacy guarantee implies one cannot distinguish whether D or D' was used to produce it. This sort of indistinguishability condition is what grants protection from linkage attacks such as those explored by Narayanan & Shmatikov (2006). The quantity ϵ describes the extent to which the probabilities differ for *most* outcomes, and δ represents the probability of observing an outcome which *breaks* the ϵ guarantee.

In the case where differentially private datasets are used to train neural networks, such indistinguishability also assures poisoned data will not have a large effect on the trained model. Ma et al. (2019) formalize this intuition by proving a lower bound for the defensive capabilities of differentially private learners against poisoning attacks.

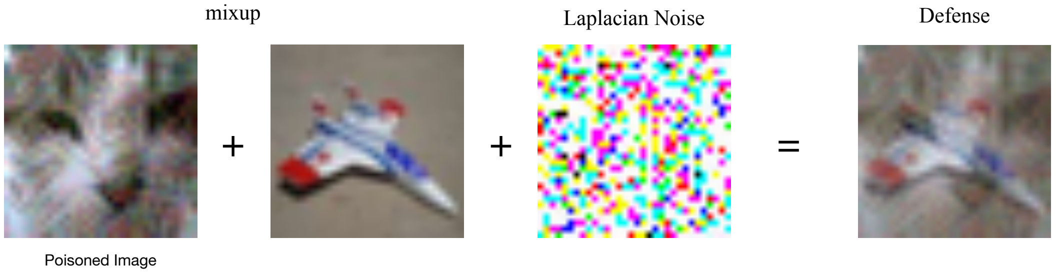


Figure 1: Illustration of the DP-InstaHide defense on two CIFAR-10 images, the first of which has been poisoned with $\varepsilon = 16$. Mixup is used to average two images, and then Laplacian noise is added,

We define the threat model as taken from Ma et al. (2019): The attacker aims to direct the trained model $\mathcal{M}(D')$ to reach some attack target by modifying at most l elements of the clean dataset D to produce the poisoned dataset D'. We measure the distance of $\mathcal{M}(D')$ from the attack target using a cost function C, which takes trained models as an input and outputs an element of $\mathbb{R}$. The attack problem is then to minimize the expectation of the cost of $\mathcal{M}(D')$.

$$\min_{D'} J(D') := \mathbb{E}[C(\mathcal{M}(D'))] \tag{2}$$

Theorem 1. *For an (ϵ, δ)-differentially private mechanism $\mathcal{M}$ and bounded cost function $|C| \leq B$, it follows that the attack cost $J(D')$ satisfies*

$$J(D') \geq \max\{e^{-l\epsilon}\left(J(D) + \frac{B\delta}{e^{\epsilon}-1}\right) - \frac{B\delta}{e^{\epsilon}-1}, 0\} \tag{3}$$

$$J(D') \geq \max\{e^{-l\epsilon}\left(J(D) + \frac{B\delta}{e^{\epsilon}-1}\right) + \frac{B\delta}{e^{\epsilon}-1}, -B\} \tag{4}$$

where the former bound holds for non-negative cost functions and the latter holds for non-positive cost functions.

Because DP is defined with *worst*-case scenarios, which do not always occur in practice, empirical studies show the defense offered by differential privacy mechanisms tends to be more effective than the theoretical limit.

We find that differential privacy achieved through DP-InstaHide, the combination of k-way mixup and additive Laplacian noise, is an example of such a defense, practically visualized in Fig.1. Because mixup augmentation concentrates training data near the center of the unit hypercube, less noise must be added to the mixed up data to render the noisy data indistinguishable from other points nearby in comparison to *solely* adding noise to the data points (Zhang et al., 2017). Additionally, mixup benefits from improved generalization due to its enforcement of linear interpolation between classes and has recently been shown to be robust to a variety of adversarial attacks, such as FGSM (Zhang et al., 2020). We use a combinatorial approach to achieve a formal differential privacy guarantee for mixup with Laplacian noise, which in tandem with the result from Ma et al. (2019) gives us a direct theoretical protection from data poisoning.

2.1 A Theoretical Guarantee for DP-InstaHide

Above, we discussed how strong data augmentations, such as mixup and random noise, provide an empirically strong defense against poisoning. We can explain the strength of this defense and provide a rigorous guarantee, by analyzing the privacy benefits of mixup within a differential privacy framework.

Let D be a dataset of size n and D' denote the same dataset with the point x_0 removed. Let d be the dimension of data points and assume the data lies in a set V of diameter one, i.e., $sup\{||D - D'||_1 : D, D' \in V\} \leq 1$. We sample a point of the form $z = \frac{1}{k}(x_1 + x_2 + \cdots + x_k) + \eta$, where the x_i are drawn at random from the relevant dataset P without replacement, and $\eta \sim Lap(\mathbf{0}, \sigma I)$

is the independent d-dimensional isotropic Laplacian additive noise vector with density function $\phi_\sigma(\eta) = \frac{1}{(2\sigma)^d} e^{\|\eta\|_1/\sigma}$.

The random variable representing the outcome of the sampling from dataset P is therefore a sum of random variables:

$$\mathcal{M}_P = \frac{1}{k}\sum_{i=1}^{k} X_i + H \tag{5}$$

We use p and q to denote the probability density functions of $\mathcal{M}_D$, and $\mathcal{M}_{D'}$ respectively.

Theorem 2. *Assume the data set D has ℓ_1-norm radius less than 1, and that mixup groups of mixture width k are sampled without replacement. The mixup plus Laplacian noise mechanism producing a data set of size N satisfies $(\epsilon, 0)$-differential privacy with*

$$\epsilon = N \max\{A, B\} \leq \frac{N}{k\sigma}$$

where

$$A = \log\left(1 - \frac{k}{n} + e^{\frac{1}{k\sigma}}\frac{k}{n}\right),\ B = \log\frac{n}{n - k + ke^{-\frac{1}{k\sigma}}}.$$

Proof. To prove differential privacy, we must bound the ratio of $\mathbb{P}[\mathcal{M}_D \in U]$ to $\mathbb{P}[\mathcal{M}_{D'} \in U]$ from above and below, where $U \subseteq V$ is arbitrary and measurable. For a fixed sampling combination $x = (x_1, \ldots, x_k) \in D^k$, the density for observing $z = \frac{1}{k}\sum_{i=1}^k x_i + N$ is given by $\phi_\sigma\left(z - \frac{1}{k}\sum_{i=1}^k x_i\right)$. Since there are $\binom{n}{k}$ possible values that x can take on, each of equal probability, we have

$$p(z) = \frac{k!(n-k)!}{n!}\sum_{x\in D^k}\phi_\sigma\left(z - \frac{1}{k}\sum_{i=1}^{k} x_i\right) \tag{6}$$

Let's now write a similar expression for $q(z)$. We have

$$q(z) = \frac{k!(n-k-1)!}{(n-1)!}\sum_{x\in D'^k}\phi_\sigma\left(z - \frac{1}{k}\sum_{i=1}^{k} x_i\right) \tag{7}$$

Now, we write the decomposition $p(z) = p_0(z) + p_1(z)$, where $p_0(z)$ is the probability of the ensemble not containing x_0 times the conditional density for observing z given this scenario, and $p_1(z)$ is the probability of having x_0 in the ensemble times the conditional density for observing z given this scenario.

Then, we have

$$p_0(z) = \left(1 - \frac{k}{n}\right)q(z) \text{ and } p_1(z) = \frac{k}{n}\frac{(k-1)!(n-k-2)!}{(n-1)!}\sum_{x\in D'^{k-1}}\phi_\sigma\left(z - \frac{x_0}{k} - \frac{1}{k}\sum_{i=1}^{k-1} x_i\right) \tag{8}$$

In the equation above, $\frac{k}{n}$ represents the probability of drawing an ensemble x that contains x_0, and the remainder of the expression is the probability of forming $z - x_0$ using the remaining $k-1$ data points in the ensemble.

We can simplify p_1 in equation 8 using a combinatorial trick. Rather than computing the sum over all tuples of size $k-1$, we compute the sum over all tuples of length k, but we discard the last entry of each tuple. We get

$$p_1(z) = \frac{k}{n}\frac{k!(n-k-1)!}{(n-1)!}\sum_{x\in D'^k}\phi_\sigma\left(z - \frac{x_0}{k} - \frac{1}{k}\sum_{i=1}^{k-1} x_i\right). \tag{9}$$

Now, from the definition of the Laplace density, we have that if $\|u - v\|_1 < \epsilon$ for any u, v then

$$e^{-\|u-v\|_1/\sigma}\phi_\sigma(v) \leq \phi_\sigma(u) \leq e^{\|u-v\|_1/\sigma}\phi_\sigma(v).$$

Let's apply this identity to equation 9 with $u = z - \frac{x_0}{k} - \frac{1}{k}\sum_{i=1}^{k-1} x_i$ and $v = z - \frac{1}{k}\sum_{i=1}^{k} x_i$. We get

$$e^{-\frac{1}{k\sigma}}\frac{k}{n}q(z) \leq p_1(z) \leq e^{\frac{1}{k\sigma}}\frac{k}{n}q(z),$$

where we have used the fact that the dataset D has unit diameter to obtain $\|u - v\|_1 \leq \frac{1}{k}$, and we used the definition equation 7 to simplify our expression.

Now, we add p_0 to this equation. We get

$$\left(1 - \frac{k}{n} + e^{-\frac{1}{k\sigma}}\frac{k}{n}\right)q(z) \leq p(z) \leq \left(1 - \frac{k}{n} + e^{\frac{1}{k\sigma}}\frac{k}{n}\right)q(z).$$

From this, we arrive at the conclusion

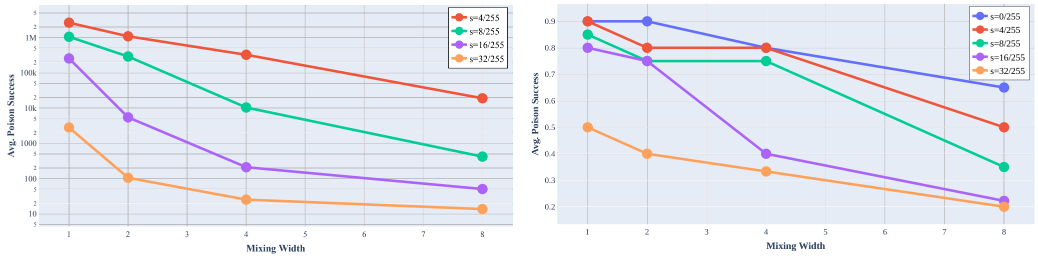
$$\frac{p(z)}{q(z)} \leq \left(1 - \frac{k}{n} + e^{\frac{1}{k\sigma}}\frac{k}{n}\right) \leq e^{\frac{1}{k\sigma}} \quad \text{and} \quad \frac{q(z)}{p(z)} \leq \frac{n}{n - k + ke^{-\frac{1}{k\sigma}}} \leq e^{\frac{1}{k\sigma}}. \tag{10}$$

The left-most upper bound in the above equation is achieved by replacing k with n wherever k appears outside of an exponent. We get the final result by taking the log of these bounds and using the composability property of differential privacy to account for the number N of points sampled.

□

Remark: A classical Laplacian mechanism for differentially private dataset release works by adding noise to each dataset vector separately and achieves privacy with $\epsilon = \frac{1}{\sigma}$. Theorem 2 recovers this bound in the case $k = 1$, however it also shows that k-way mixup enhances the privacy guarantee over the classical mechanism *by a factor of at least* k.

We investigate the practical implications of Theorem 2 in Figure 2, where we show the predicted theoretical privacy guarantees in Figure 2a and the direct practical application for defenses against data poisoning in Figure 2b. Figure 2b shows the average poison success for a strong, adaptive gradient matching attack against a ResNet-18 trained on CIFAR-10 (the setting considered in Geiping et al. (2020) with an improved adaptive attack). We find that the theoretical results predict the success of a defense by mixup with Laplacian noise surprisingly well.



(a) Theoretical privacy guarantees

(b) Empirical defense against poisoning attacks

Figure 2: Theoretical and empirical mixup. **Left:** Privacy guarantee ϵ as a function of mixture width k, computed for each implemented Laplacian noise level s. We use values $n = N = 50000$, corresponding to the CIFAR-10 dataset. **Right:** Poisoning success for a strong adaptive gradient matching attack for several mixture widths and noise levels.

3 Data Augmentation as an Empirical Defense against Dataset Manipulation

Now that we have established the provable benefits of mixup, we study the empirical effectiveness of data augmentations to prevent poisoning. We are mainly interested in data augmentations that mix data points; we consider the hypothesis that data poisoning attacks rely on the deleterious effects of a

Table 1: Validation accuracy and poison success for a baseline model, models trained with mixup and CutMix augmentations compared with Spectral Signature (Tran et al., 2018a) and Activation Clustering (Chen et al., 2018) defenses. The first two columns have 10% of one class poisoned, and the latter two have all poisoned (filter defenses are inapplicable here). The results are averaged across 20 runs.

	Acc. (10%)	Poison Success (10%)	Acc. (100%)	Poison Success (100%)
Baseline	94.3%	45.6%	85.0%	98.3%
CutMix	95.1%	**7.0%**	94.2%	**14.1%**
mixup	94.4%	23.9%	85.3%	99.8%
SS	92.3%	48.3%		
AC	89.4%	44.0%		

subset of modified samples, which can in turn be diluted and deactivated by mixing them with other, likely unmodified, samples.

In addition to mixup, which we analyzed in the previous section, we now consider several other augmentations. CutOut (DeVries & Taylor, 2017), which blacks out a randomly generated patch from an image, can be combined with mixup to form CutMix (Yun et al., 2019), another type of mixing augmentation. Specifically, the idea is to paste a randomly selected patch from one image onto a second image, with labels computed by taking a weighted average of the original labels. The weights of the labels correspond to the relative area of each image in the final augmented data point. MaxUp (Gong et al., 2020) can also be considered as a mixing data augmentation, which first generates augmented samples using various techniques and then selects the sample with the lowest associated loss value to train on. CutMix and mixup will be the central mixing augmentations that we consider in this work, which we contrast with MaxUp in select scenarios.

3.1 Backdoor Attacks

In contrast to recent targeted data poisoning attacks, *backdoor* attacks often involve inserting a simple preset trigger into training data to cause base images to be misclassified into the target class. For our experiments, we use small 4×4 randomly generated patches as triggers to poison the target class. To evaluate the baseline effectiveness of backdoor attacks, we poison a target class, train a ResNet-18 model on this poisoned data and use it to classify patched images from a victim test class. Only if a patched image from a victim class is labeled with the target class do we treat it as a successfully poisoned example. Our results show that backdoor attacks achieve 98.3% poison success when 100% of images from the target class are poisoned and 45.6% poison success when only 10% of target images are patched (see Table 1). In addition, when 100% of training images from the target class are patched, clean test accuracy of the model drops by almost 10% since the model is unable to learn meaningful features of the target class.

We then compare the baseline model to models trained with the mixup and CutMix augmentations. We find that although mixup helps when only part of the target class is poisoned, it is not efficient as a defense against backdoor attacks when all images in the target class are patched. In contrast, CutMix is an extremely effective defense against backdoor attacks in both scenarios and it reduces poison success from 98.3% to 14.1% in the most aggressive setting. Finally, models trained on poisoned data with CutMix data augmentation have a clean test accuracy similar to the accuracy of models trained on clean data. Intuitively, CutMix often produces patch-free mixtures of the target class with other classes, hence the model does not solely rely on the patch to categorize images of this class.

Table 2: Poison success rates (lower is better for the defender) for various data augmentations tested against the gradient matching attack of Geiping et al. (2020). All results are averaged over 20 trials. We report the success of both a non-adaptive and an adaptive attacker.

Augmentation	Non-Adaptive	Adaptive
2-way mixup	45.00%	72.73%
Cutout	60.00%	81.25%
CutMix	75.00%	60.00%
4-way mixup	5.00%	55.00%
MaxUp-Cutout	5.26%	20.00%

We extend this analysis to two more complex attacks, clean-label backdoor attacks (Turner et al., 2018), and hidden-Trigger backdoor attacks in Table 5.

3.2 Targeted Data Poisoning

We further evaluate data augmentations as a defense against targeted data poisoning attacks. We analyze the effectiveness of CutMix and mixup as a defense against feature collision attacks in Table 5. Applying these data augmentations as a defense against Poison Frogs (Shafahi et al., 2018) (FC) is exceedingly successful, as the poisoned data is crafted independently there, making it simple to disturb by data augmentations. The poisons crafted via Convex Polytope (CP) (Zhu et al., 2019) however, are more robust to data augmentations, due to the polytope of poisoned data created around the target. Nonetheless, the effectiveness of CP is diminished more by data augmentations than by other defenses.

We then evaluate the success of data augmentations against Witches' Brew, the gradient matching attack of Geiping et al. (2020) in Table 2. Against this attack, we evaluate a wide range of data augmentations, as the attack is relatively robust to basic mixup data augmentations which mix only two images. However, using a stronger augmentation that mixes four images still leads to a strong defense in the non-adaptive setting (where the attacker is unaware of the defense). As this attack can be adapted to specific defenses, we also consider such a scenario. Against the adaptive attack, we found MaxUp to be most effective, evaluating the worst-case loss for every image in a minibatch over four samples of data augmentation drawn from cutout. To control for the effects of the CIFAR-10 dataset that we consider for most experiments, we also evaluate defenses against an attack on the ImageNet dataset in Table 4, finding that the described effects transfer to other datasets.

Table 3: Poison success rates (lower is better for the defender) for competing defenses when tested against the gradient matching attack compared to mixup. For DP-SGD, we consider a noise level of $n = 0.01$. All results are averaged over 20 trials.

Defense	Poison Success
Spectral Signatures	95.00%
deepKNN	90.00%
Activation Clustering	30.00%
DP-SGD	86.25%
4-way mixup	5.00%

3.3 Comparison to Other Defenses

We compare our method to previous defenses referenced in Section 1.1. We show that our method outperforms filter defenses when evaluating backdoor attacks, such as in Table 1 and Table 5, as well as when evaluating targeted data poisoning attacks, as we show for Poison Frogs and Convex Polytope in Table 5 and for Witches' Brew in Table 4 and 3. We note that data augmentations do not require additional training compared to filter defenses in some settings and are consequently more computationally efficient.

Table 4: Success rate for selected data augmentation when tested against the gradient matching attack on the ImageNet dataset. All results are averaged over 10 trials.

Augmentation	Poison success
None	90%
2-way mixup	50.00%
4-way mixup	30.00%

In Figure 3, we plot the average poison success against the validation error for adaptive gradient matching attacks. We find that data augmentations exhibit a stronger security performance trade-off compared to other defenses.

3.4 Defenses with DP Augmentations in Practice

As a result of Theorem 2, we investigate the data augmentations previously considered in Section 3 with additional Laplacian noise, also in the setting of a gradient matching attack. Figure 3 shows that the benefits of Laplacian noise which we only prove for mixup also apply empirically to variants of mixing data augmentations. While we only prove formal guarantees for the mixup-based mechanism due to its mathematical simplicity, our intuitions also extend to other augmentations,

Table 5: On the left: poison success rate for Poison Frogs (Shafahi et al., 2018) and Convex Polytope (Zhu et al., 2019) attacks when tested with baseline settings and when tested with mixup and CutMix. On the right: success rate against backdoor attacks when tested with baseline settings and when tested with the mixup and CutMix. All results are averaged over 20 trials.

Attack	Baseline	SS	AC	mixup	CutMix
FC	80%	70%	45%	**5%**	**5%**
CP	95%	90%	75%	70%	**50%**

Attack	Baseline	SS	AC	mixup	CutMix
HTBD	60%	65%	55%	20%	**10%**
CLBD	65%	60%	45%	25%	**15%**

specifically CutMix and MaxUp, which mix together the features of two or more samples. For example, combining MaxUp with Laplacian noise of sufficient strength ($s = 16/255$) completely shuts down the data poisoning attack via adaptive gradient matching, significantly improving upon numbers reached by MaxUp alone.

4 Discussion

Strong data augmentations have previously been used to improve generalization in neural networks. In this work, we analyse these data augmentations theoretically through the lens of differential privacy, due to its connections to poisoning robustness. We prove that mixup augmentation enhances the defensive guarantees obtained by adding noise to inputs, improving standard guarantees at least linearly in mixture width. We then apply these findings practically, evaluating the effects of mixup data augmentations combined with Laplacian input noise. Finally, we show that such augmentations also yield a strong empirical defense against a range of data poisoning and backdoor attacks.

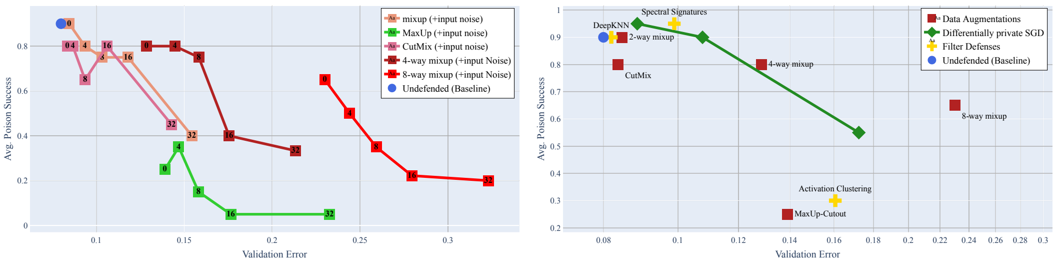


Figure 3: On the left: Enhancing various data augmentations with Laplacian noise. We visualize the security-performance trade-off when enhancing the data augmentations considered in Sec. 3 with Laplacian noise as predicted by Thm. 2. We visualize the development of these data augmentations when adding Laplacian noise with scales ($2/255$, $4/255$, $8/255$, $16/255$, $32/255$). On the right: Trade-off between average poison success and validation accuracy for various defenses against gradient matching (adaptive).

5 Ethics and Reproducibility

In our experiments, we limit analysis to a subset of existing poisoning attacks. It is important to recognize that though our empirical results likely extend to other attacks or new attacks to an extent, there is no strict guarantee of their effectiveness (with the exception of the lower bound provided by DP-InstaHide) and thus, practitioners should avoid harboring a false sense of security when applying our proposed defenses. Moreover, all experiments we conduct are on image datasets and data augmentations designed for computer vision. Real-world practitioners encounter a diverse array of learning problems and should be cautious in their expectations that our method will be highly effective in their own settings.

To reproduce our findings, please refer to the code available in the supplementary material. Specific methodologies and hyperparameters for producing the presented figures, and additional details for the proof of Theorem 2 can be found in the appendix section. For our work, we make use of the free, widely available CIFAR data set (Krizhevsky et al., 2009).

REFERENCES

Hojjat Aghakhani, Dongyu Meng, Yu-Xiang Wang, Christopher Kruegel, and Giovanni Vigna. Bullseye Polytope: A Scalable Clean-Label Poisoning Attack with Improved Transferability. *arXiv:2005.00191 [cs, stat]*, April 2020.

Marco Barreno, Blaine Nelson, Anthony D. Joseph, and J. D. Tygar. The security of machine learning. *Mach. Learn.*, 81(2):121–148, November 2010. ISSN 0885-6125. doi: 10.1007/s10994-010-5188-5.

Battista Biggio, Blaine Nelson, and Pavel Laskov. Poisoning attacks against support vector machines. *arXiv preprint arXiv:1206.6389*, 2012.

Nicholas Carlini, Samuel Deng, Sanjam Garg, Somesh Jha, Saeed Mahloujifar, Mohammad Mahmoody, Shuang Song, Abhradeep Thakurta, and Florian Tramer. An Attack on InstaHide: Is Private Learning Possible with Instance Encoding? *arXiv:2011.05315 [cs]*, November 2020.

Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, and Biplav Srivastava. Detecting backdoor attacks on deep neural networks by activation clustering. *arXiv preprint arXiv:1811.03728*, 2018.

Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, and Dawn Song. Targeted backdoor attacks on deep learning systems using data poisoning. *arXiv preprint arXiv:1712.05526*, 2017.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pp. 248–255. Ieee, 2009.

Terrance DeVries and Graham W Taylor. Improved regularization of convolutional neural networks with cutout. *arXiv preprint arXiv:1708.04552*, 2017.

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

Jonas Geiping, Liam Fowl, W Ronny Huang, Wojciech Czaja, Gavin Taylor, Michael Moeller, and Tom Goldstein. Witches' brew: Industrial scale data poisoning via gradient matching. *arXiv preprint arXiv:2009.02276*, 2020.

Micah Goldblum, Dimitris Tsipras, Chulin Xie, Xinyun Chen, Avi Schwarzschild, Dawn Song, Aleksander Madry, Bo Li, and Tom Goldstein. Dataset security for machine learning: Data poisoning, backdoor attacks, and defenses. *arXiv preprint arXiv:2012.10544*, 2020.

Chengyue Gong, Tongzheng Ren, Mao Ye, and Qiang Liu. Maxup: A simple way to improve generalization of neural network training. *arXiv preprint arXiv:2002.09024*, 2020.

Tianyu Gu, Brendan Dolan-Gavitt, and Siddharth Garg. Badnets: Identifying vulnerabilities in the machine learning model supply chain. *arXiv preprint arXiv:1708.06733*, 2017.

W Ronny Huang, Jonas Geiping, Liam Fowl, Gavin Taylor, and Tom Goldstein. Metapoison: Practical general-purpose clean-label data poisoning. *arXiv preprint arXiv:2004.00225*, 2020a.

Yangsibo Huang, Zhao Song, Kai Li, and Sanjeev Arora. InstaHide: Instance-hiding Schemes for Private Distributed Learning. In *International Conference on Machine Learning*, pp. 4507–4518. PMLR, November 2020b.

Pang Wei Koh, Jacob Steinhardt, and Percy Liang. Stronger data poisoning attacks break data sanitization defenses. *arXiv preprint arXiv:1811.00741*, 2018.

Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. University of Toronto, 2009. Master's thesis.

Kangwook Lee, Hoon Kim, Kyungmin Lee, Changho Suh, and Kannan Ramchandran. Synthesizing differentially private datasets using random mixing. In *2019 IEEE International Symposium on Information Theory (ISIT)*, pp. 542–546. IEEE, 2019.

Alexander Levine and Soheil Feizi. Deep partition aggregation: Provable defense against general poisoning attacks. *arXiv preprint arXiv:2006.14768*, 2020.

Yuzhe Ma, Xiaojin Zhu, and Justin Hsu. Data poisoning against differentially-private learners: Attacks and defenses. *arXiv preprint arXiv:1903.09860*, 2019.

Luis Muñoz-González, Battista Biggio, Ambra Demontis, Andrea Paudice, Vasin Wongrassamee, Emil C Lupu, and Fabio Roli. Towards poisoning of deep learning algorithms with back-gradient optimization. In *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*, pp. 27–38, 2017.

Arvind Narayanan and Vitaly Shmatikov. How to break anonymity of the netflix prize dataset. *arXiv preprint cs/0610105*, 2006.

Andrea Paudice, Luis Muñoz-González, and Emil C Lupu. Label sanitization against label flipping poisoning attacks. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 5–15. Springer, 2018.

Neehar Peri, Neal Gupta, W. Ronny Huang, Liam Fowl, Chen Zhu, Soheil Feizi, Tom Goldstein, and John P. Dickerson. Deep k-nn defense against clean-label data poisoning attacks, 2019.

Aniruddha Saha, Akshayvarun Subramanya, and Hamed Pirsiavash. Hidden trigger backdoor attacks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 11957–11965, 2020.

Avi Schwarzschild, Micah Goldblum, Arjun Gupta, John P Dickerson, and Tom Goldstein. Just how toxic is data poisoning? a unified benchmark for backdoor and data poisoning attacks. *arXiv preprint arXiv:2006.12557*, 2020.

Ali Shafahi, W Ronny Huang, Mahyar Najibi, Octavian Suciu, Christoph Studer, Tudor Dumitras, and Tom Goldstein. Poison frogs! targeted clean-label poisoning attacks on neural networks. *arXiv preprint arXiv:1804.00792*, 2018.

Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. *arXiv preprint arXiv:1811.00636*, 2018a.

Brandon Tran, Jerry Li, and Aleksander Madry. Spectral signatures in backdoor attacks. *arXiv preprint arXiv:1811.00636*, 2018b.

Alexander Turner, Dimitris Tsipras, and Aleksander Madry. Clean-label backdoor attacks. *arXiv*, 2018.

Yu-Xiang Wang, Borja Balle, and Shiva Prasad Kasiviswanathan. Subsampled rényi differential privacy and analytical moments accountant. In *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1226–1235. PMLR, 2019.

Maurice Weber, Xiaojun Xu, Bojan Karlas, Ce Zhang, and Bo Li. Rab: Provable robustness against backdoor attacks. *arXiv preprint arXiv:2003.08904*, 2020.

Huang Xiao, Battista Biggio, Gavin Brown, Giorgio Fumera, Claudia Eckert, and Fabio Roli. Is feature selection secure against training data poisoning? In *International Conference on Machine Learning*, pp. 1689–1698. PMLR, 2015.

Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 6023–6032, 2019.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Linjun Zhang, Zhun Deng, Kenji Kawaguchi, Amirata Ghorbani, and James Zou. How does mixup help with robustness and generalization? *arXiv preprint arXiv:2010.04819*, 2020.

Chen Zhu, W Ronny Huang, Hengduo Li, Gavin Taylor, Christoph Studer, and Tom Goldstein. Transferable clean-label poisoning attacks on deep neural nets. In *International Conference on Machine Learning*, pp. 7614–7623. PMLR, 2019.

A Appendix

A.1 Backdoor Attacks

For the patch attack, we insert patches of size 4×4 into CIFAR (Krizhevsky et al., 2009) train images from target class and test images from victim class. The patches are generated using a Bernoulli distribution and are normalized using the mean and standard deviation of CIFAR training data. The patch location for each image is chosen at random. To evaluate the effectiveness of the backdoor attack and our proposed defenses, we train a ResNet-18 model on poisoned data with cross-entropy loss. The model is trained for 80 epochs using SGD optimizer with a momentum of 0.9, a weight decay of 5e-4 and learning rate of 0.1 which we reduce by a factor of 10 at epochs 30, 50 and 70. A batch size of 128 is used during training.

We run our experiments for HTBD and CLBD in Table 5 by implementing mixup and CutMix in the publically available framework of Schwarzschild et al. (2020), and using this re-implementation for our comparison with the hyperparameters proposed there.

Standard error for our twenty trial experiments presented in 3 can be computed using the standard formula for binomial distributions

$$SE = \sqrt{\frac{p(p-1)}{n}},$$

where p is the success probability and $n = 20$ is the number of trials. The error bars were omitted from the body of our work in the interest of space and readability, but we present the same tables below with the full statistical information.

	Acc. (10%)	Poison Success (10%)	Acc. (100%)	Poison Success (100%)
Baseline	$94.3 \pm 5.2\%$	$45.6 \pm 11.13\%$	$85.0 \pm 7.98\%$	$98.3 \pm 2.89\%$
CutMix	$95.1 \pm 4.82\%$	$\mathbf{7.0 \pm 5.70\%}$	$94.2 \pm 5.22\%$	$\mathbf{14.1 \pm 7.78\%}$
mixup	$94.4 \pm 5.14\%$	$23.9 \pm 9.53\%$	$85.3 \pm 7.92\%$	$99.8 \pm 1.00\%$
SS	$92.3 \pm 5.96\%$	$48.3 \pm 11.17\%$		
AC	$89.4 \pm 6.88\%$	$44.0 \pm 11.10\%$		

Attack	Baseline	SS	AC	mixup	CutMix
HTBD	$60 \pm 10.95\%$	$65 \pm 10.67\%$	$55 \pm 11.12\%$	$20 \pm 8.94\%$	$\mathbf{10 \pm 6.71\%}$
CLBD	$65 \pm 10.67\%$	$60 \pm 10.95\%$	$45 \pm 11.12\%$	$25 \pm 9.68\%$	$\mathbf{15\% \pm 7.98\%}$

A.2 Targeted Data Poisoning

We run our experiments for feature collision attacks in Table 5 by likewise using the framework of Schwarzschild et al. (2020), running the defense with the same settings as proposed there and following the constraints considered in this benchmark. For gradient matching we likewise implement a number of data augmentations as well as input noise into the framework of Geiping et al. (2020). We run all gradient matching attacks within their proposed constraints, using a subset of 1% of the training data to be poisoned for gradient matching and an ℓ^∞ bound of 16/255. For all experiments concerning gradient matching we thus consider the same setup of a ResNet-18 trained on normalized CIFAR-10 with horizontal flips and random crops of size 4, trained by Nesterov SGD with 0.9

momentum and 5e-4 weight decay for 40 epochs for a batch size of 128. We drop the initial learning rate of 0.1 at epochs 14, 24 and 35 by a factor of 10. For the ImageNet (Deng et al., 2009) experiments we consider the same hyperparameters for an ImageNet-sized ResNet-18, albeit for a smaller budget of 0.01% as in the original work.

Comparing to poison detection algorithms, we re-implement *spectral signatures* (Tran et al., 2018b), *deep K-NN* (Peri et al., 2019) and *Activation Clustering* (Chen et al., 2018) with hyperparameters as proposed in their original implementations. For differentially private SGD, we implement Gaussian gradient noise and gradient clipping to a factor of 1 on the mini-batch level (otherwise the ResNet-18 architecture we consider would be inapplicable due to batch normalizations), and vary the amount of gradient noise with values $(0.0001, 0.001, 0.01)$ to produce the curve in Fig. 2.

To implement data augmentation defenses we generally these data augmentations straightforward as proposed in their original implementations, also keeping components such as the late start of Maxup after 5 epochs described in Gong et al. (2020) and the randomized activation of CutMix described in Zhang et al. (2017).

Standard error is computed in the same way as above, which produces the following tables:

ATTACK	BASELINE	SS	AC	MIXUP	CUTMIX
FC	$80.00 \pm 8.94\%$	$70.00 \pm 10.25\%$	$45.00 \pm 11.12\%$	$\mathbf{5.00 \pm 4.87\%}$	$\mathbf{5.00 \pm 4.87\%}$
CP	$95.00 \pm 4.87\%$	$90.00 \pm 6.71\%$	$75.00 \pm 9.68\%$	$70.00 \pm 10.25\%$	$\mathbf{50.00 \pm 11.18\%}$

AUGMENTATION	NON-ADAPTIVE	ADAPTIVE
2-WAY MIXUP	$45.00 \pm 11.12\%$	$72.73 \pm 9.96\%$
CUTOUT	$60.00 \pm 10.95\%$	$81.25 \pm 8.73\%$
CUTMIX	$75.00 \pm 9.68\%$	$60.00 \pm 10.95\%$
4-WAY MIXUP	$5.00 \pm 4.87\%$	$55.00 \pm 11.12\%$
MAXUP-CUTOUT	$5.26 \pm 4.99\%$	$20.00 \pm 8.94\%$

AUGMENTATION	POISON SUCCESS
NONE	$90.00 \pm 6.71\%$
2-WAY MIXUP	$50.00 \pm 11.18\%$
4-WAY MIXUP	$30.00 \pm 10.25\%$

DEFENSE	POISON SUCCESS
SPECTRAL SIGNATURES	$95.00 \pm 4.87\%$
DEEPKNN	$90.00 \pm 6.71\%$
ACTIVATION CLUSTERING	$30.00 \pm 10.25\%$
DP-SGD	$86.25 \pm 7.70\%$
4-WAY MIXUP	$5.00 \pm 4.87\%$

A.3 ADDITIONAL PROOF DETAILS

In the interest of preserving space and readability, we omit a few details from the proof of Theorem 2, which we present in full here. The expressions in (6) and (7) are derived using the identity $p(z) = \sum_x p(z|x)p(x)$ and $q(z) = \sum_x q(z|x)q(x)$ respectively.

The final inequality on the LHS of (10) can be rewritten as $e^{\frac{-1}{k\sigma}}\left(1 - \frac{k}{n}\right) + \frac{k}{n} \leq 1$ by making the observation

$$1 - \frac{k}{n} + e^{\frac{1}{k\sigma}}\frac{k}{n} = e^{\frac{1}{k\sigma}}\left[e^{\frac{-1}{k\sigma}}\left(1 - \frac{k}{n}\right) + \frac{k}{n}\right]$$

This can further be simplified to showing $e^{\frac{-1}{k\sigma}} \leq 1$, which is true because $\frac{-1}{k\sigma}$ is always negative. We can use the same trick of pulling out a factor of $e^{\frac{1}{k\sigma}}$ to show the final inequality on the RHS of (10) reduces to proving

$$\frac{n}{(n-k)e^{\frac{1}{k\sigma}} + k} \leq 1 \implies n - k \leq (n-k)e^{\frac{1}{k\sigma}}.$$

The above inequality holds because $k\sigma$ is necessarily positive, so $e^{\frac{-1}{k\sigma}} > 1$.

A.4 Compute Resources

All experiments were performed on Nvidia GeForce RTX 2080Ti GPUs. For CIFAR-10 experiments, the maximum total compute time utilized was 45 minutes per trial. ImageNet experiments required a maximum total compute time of 70 hours per trial. Summing over the 43 CIFAR-10 experiments and 3 ImageNet experiments each with 20 and 10 trials respectively, we arrive at approximately 2745 GPU hours or 16.3 GPU weeks used for the entire project.