Enhancing the Antidote: Improved Pointwise Certifications against Poisoning Attacks

Shijie Liu[1], **Andrew C. Cullen**[1], **Paul Montague**[2], **Sarah M. Erfani**[1], **Benjamin I. P. Rubinstein**[1]

[1]School of Computing and Information Systems, University of Melbourne, Melbourne, Australia
[2]Defence Science and Technology Group, Adelaide, Australia
shijiel2@student.unimelb.edu.au, andrew.cullen@unimelb.edu.au, paul.montague@dst.defence.gov.au,
sarah.erfani@unimelb.edu.au, benjamin.rubinstein@unimelb.edu.au

Abstract

Poisoning attacks can disproportionately influence model behaviour by making small changes to the training corpus. While defences against specific poisoning attacks do exist, they in general do not provide any guarantees, leaving them potentially countered by novel attacks. In contrast, by examining worst-case behaviours Certified Defences make it possible to provide guarantees of the robustness of a sample against adversarial attacks modifying a finite number of training samples, known as pointwise certification. We achieve this by exploiting both Differential Privacy and the Sampled Gaussian Mechanism to ensure the invariance of prediction for each testing instance against finite numbers of poisoned examples. In doing so, our model provides guarantees of adversarial robustness that are more than twice as large as those provided by prior certifications.

Introduction

Despite the impressive performance, many modern machine learning models have been shown to be vulnerable to adversarial data perturbations (Biggio, Nelson, and Laskov 2013; Chen et al. 2017; Chakraborty et al. 2018). This adversarial sensitivity is a significant concern now that machine learning models are increasingly being deployed in sensitive applications. Of particular concern are data poisoning attacks, where an adversary manipulates the training set to change the decision boundary of learned models. The risk of such attacks is heightened by the prevalence of large, user-generated datasets that are constructed without vetting. The fact that these attacks can render a model useless further underscores the need for robust defence mechanisms. Some examples of models that are vulnerable to data poisoning attacks include email spam filters and malware classifiers. These models have been shown to be susceptible to attacks that either render the model ineffective (Biggio, Nelson, and Laskov 2013), or that produce targeted misclassifications (Chen et al. 2017).

The defences intrinsically counter specific poisoning attacks means that even state-of-the-art defences (Carnerero-Cano et al. 2020; Paudice et al. 2018) can be vulnerable to new attacks. To circumvent this inherent dependency of defences on attacks, recent work has begun to consider the construction of guarantees of predictive invariance against bounded numbers of poisoned training examples. This is known as the *certified robustness*, which is commonly achieved through the addition of calibrated noise through *randomised smoothing* (Lecuyer et al. 2019). While these certifications have been successfully applied to poisoning attacks on labels and/or input features (Rosenfeld et al. 2020; Wang et al. 2020), their applicability has been limited to attacks that modify training examples, rather than the more general insertion/deletion operations. On the other hand, classifiers trained with *differential privacy* (DP) can be shown to be certifiably robust against poisoning attacks even against insertion/deletion operations (Ma, Zhu, and Hsu 2019; Hong et al. 2020). However, to date, such certifications do not provide *pointwise guarantees* which ensures the robustness for individual samples against a finite number of poisoned training examples. This omission still leaves a vulnerability that can be exploited by a motivated adversary to compel the model to misclassify a particular testing sample. Recent works (Jia, Cao, and Gong 2020; Levine and Feizi 2021) leveraging bagging have achieved pointwise guarantees against poisoning attacks that allow insertion/deletion. However, some of these methods are specialized to particular learning approaches.

In this work, we establish a general framework for deriving pointwise-certifiably robust guarantees against data poisoning attacks that can influence both the label and feature sets. Such guarantees ensure that the predicted class of an individual sample are invariant to a finite number of changes to the training dataset. Prior works have leveraged DP to improve statistical properties of certification against data poisoning across a dataset. In contrast, we are the first to extend DP to certify individual samples. By producing an *improved group privacy for the Sampled Gaussian Mechanism*, our new approach even yields certifications that hold for more changes to the training dataset than what had been identified by prior approaches (Ma, Zhu, and Hsu 2019; Jia, Cao, and Gong 2020; Levine and Feizi 2021). Our specific achievements can be summarized as follows:

- A general framework providing *pointwise-certified robustness* guarantees for models that are trained with differentially-private learners.
- The framework provides a *general poisoning attack defence* against insertion, deletion, and modification attacks on both the label and feature sets. The defence improves the existing differential privacy based approaches, and its

efficiency is enhanced through optimised group privacy in the Sampled Gaussian Mechanism and sub-sampled training.

- Our defence method achieves more than double the number of poisoned examples compared to existing certified approaches as demonstrated by experiments on MNIST, Fashion-MNIST and CIFAR-10.

Data Poisoning Attacks and Defences

Training-time or data poisoning attacks (Barreno et al. 2006; Biggio, Fumera, and Roli 2014) enable malicious users to manipulate training data and modify the learned model. The expressiveness of machine learning model families makes modern machine learning particularly susceptible to such attacks (Chakraborty et al. 2018; Goldblum et al. 2021). These attacks can be taxonomically classified as either label attacks, which only modify dataset labels (Xiao, Xiao, and Eckert 2012); features attacks, in which the training features are modified (Shafahi et al. 2018); or example attacks, such as the backdoor, which seek to influence both labels and features of the training corpus (Shafahi et al. 2018). Defending against any of these attacks is inherently complex, as their existence implies that the attacker has access to both the training architecture and dataset. Although previous works have examined attackers who solely modify the training data, our threat model assumes a more comprehensive scenario, whereby attackers have the freedom to introduce or remove samples from the training dataset, as outlined in Table 1. However, this freedom is subject to certain constraints that aim to reduce the probability of detection.

Threat Model. We consider supervised multi-class classification on a training dataset $\mathcal{D}_1 = \{(\mathbf{x}_i, y_i)\}_{i=1}^n$, where each example comprises an input instance $\mathbf{x} \in \mathbb{R}^m$ and label $y_i \in \mathcal{L} = \{1, \ldots, L\}$. Consider a (possibly randomised) learner M taking $\mathcal{D}_1$ to parameters $\boldsymbol{\theta} \in \Theta$. We refer to learned parameters and *model* interchangeably.

In this paper we consider alternate forms of inferred scores per class for randomised learners on a given input instance $\mathbf{x} \in \mathbb{R}^m$ as $I_l(\mathbf{x}, \boldsymbol{\theta})$, such that $\sum_{l\in\mathcal{L}} I_l(\mathbf{x}, \boldsymbol{\theta}) = 1$ and $I_l(\mathbf{x}, \boldsymbol{\theta}) \in [0, 1]$, necessitating alternate choices of $I_l(\cdot)$. Let's consider a function $\mathbf{y}(\mathbf{x}, \boldsymbol{\theta})$ that returns a deterministic vector of predicted class scores in $\mathbb{R}^L$, with the ith component denoted $y_i(\mathbf{x}, \boldsymbol{\theta}) \in \mathbb{R}$. For example, the softmax layer of a deep network outputs a score per class, these y_i sit in $[0, 1]$ and sum to unity.

Definition 1 (Inference by multinomial label)**.** Define the *multinomial label* inference function as

$$I_l(\mathbf{x}, M(\mathcal{D}_1)) = \Pr[\arg\max_i y_i(\mathbf{x}, M(\mathcal{D}_1)) = l] \ .$$

Conditioned on a deterministic model θ, we may make a prediction on $\mathbf{x}$ as the highest y_i score; however, with these predictions as random induced by M, we make inferences given a training dataset as the most likely prediction.

Definition 2 (Inference by probability scores)**.** Define the *probability scores* inference function as

$$I_l(\mathbf{x}, M(\mathcal{D}_1)) = \mathbb{E}[y_l(\mathbf{x}, M(\mathcal{D}_1))] \ .$$

In other words, we consider the $\mathbf{y}$ scores as random variables (due to the randomness in learner M) conditional on training dataset $\mathcal{D}_1$ and input instance $\mathbf{x}$. We infer the class l with the largest expected score y_l.

These two inference rules capture alternate approaches to de-randomising class predictions and offer different relative advantages in terms of robustness. We discuss this further in the "Outcomes-Guaranteed Certifications" section.

The attacker is assumed to have perfect knowledge of both the dataset $\mathcal{D}_1$, learner M, and inference rule I (*i.e.*, a white-box attacker) with unbounded computational capabilities. However, in order to minimise the likelihood of an attack being detected, it assumed that a finite number $r \in \mathbb{N}$—known as the radius—of changes to the dataset. To reflect the assumed level of access of the attacker, these changes can take the form of additions, deletions, or modifications. We consider the attacker as attempting to achieve

$$\arg\max_{l\in\mathcal{L}} I_l(\mathbf{x}, M(\mathcal{D}_2)) \neq \arg\max_{l\in\mathcal{L}} I_l(\mathbf{x}, M(\mathcal{D}_1)) \ , \quad (1)$$

subject to the bound

$$\mathcal{B}(\mathcal{D}_1, r) := \{\mathcal{D}_2 : |\mathcal{D}_1 \ominus \mathcal{D}_2| \leq r\} \ . \quad (2)$$

Here $|\mathcal{D}_1 \ominus \mathcal{D}_2|$ measures the *size* of the symmetric difference between datasets $\mathcal{D}_1$ and $\mathcal{D}_2$, or in other words, the minimum number of operations required to map $\mathcal{D}_1$ to $\mathcal{D}_2$. The objective of the defence is to achieve *pointwise-certified robustness* for an individual sample $\mathbf{x}$ when passed through $I \circ M$. While such a threat model can be applied to any model, henceforth we will limit our consideration of randomised learners incorporating certified defences, of the form that will be described within the remainder of the paper.

Certified Defences. The concept of pointwise-certified robustness has been widely used as a testing-time defence (Cohen, Rosenfeld, and Kolter 2019; Lecuyer et al. 2019), and has recently been extended to training-time Rosenfeld et al. (2020). Pointwise robustness is advantageous over statistical guarantees on bounds of the objective function (Ma, Zhu, and Hsu 2019; Hong et al. 2020), as it ensures that the attacked learner will remain unchanged for finite, bounded perturbations. The nature of these perturbations, and the certification radius r are intrinsically linked to the underlying threat model.

Definition 3 (Pointwise-Certified Robustness)**.** A learner is said to be r-*pointwise-certified robust* poisoning attacks, at input instance $\mathbf{x}$, if there exists no $\mathcal{D}_2 \in \mathcal{B}(\mathcal{D}_1, r)$ such that Equation (1) is true. A learner M is said to be (η, r)-*pointwise-certified robust* if it is r-pointwise-certified robust with probability at least $1 - \eta$ in the randomness of M.

In other words, the prediction of the poisoned model remains the same (or the same w.h.p.), as the poisoned dataset does not alter the probabilities of labels sufficiently to change the predicted classification.

One approach for achieving pointwise certification is randomised smoothing (Rosenfeld et al. 2020; Weber et al. 2021), in which a new classifier is created such that its prediction is defined as the most probable class returned by the original classifier under calibrated input noise. While this noise is often applied directly to the input samples, it has been shown that model bagging can also generate output randomness in a fashion that allows for certifications against data poisoning attacks (Jia, Cao, and Gong 2020; Chen et al. 2020; Levine and Feizi 2021).

Outcomes Guarantee

By exploiting both DP and the Sampled Gaussian Mechanism (SGM), our certification framework empirically improves pointwise certifications against data poisoning. Such certificates can be used to quantify the confidence in a sample's prediction, in the face of potential dataset manipulation. To support our enhancements, we will first define some key properties of DP, then propose the outcomes guarantee that generalises to most DP mechanisms, and finally introduce the SGM with improved group privacy.

Differential Privacy. A framework (Dwork et al. 2006; Abadi et al. 2016; Friedman and Schuster 2010) quantifies the privacy loss due to releasing aggregate statistics or trained models on sensitive data. As a versatile notation of smoothness to input perturbations, DP has successfully been used as the basis of several certification regimes.

Definition 4 (Approximate-DP)**.** A randomised function M is said to be (ϵ, δ)-approximate DP (ADP) if for all datasets $\mathcal{D}_1$ and $\mathcal{D}_2$ for which $\mathcal{D}_2 \in \mathcal{B}(\mathcal{D}_1, 1)$, and for all measurable output sets $\mathcal{S} \subseteq \mathrm{Range}(M)$:

$$\Pr[M(\mathcal{D}_1) \in \mathcal{S}] \leq \exp(\epsilon) \Pr[M(\mathcal{D}_2) \in \mathcal{S}] + \delta \ , \tag{3}$$

where $\epsilon > 0$ and $\delta \in [0, 1)$ are chosen parameters.

Smaller values of the privacy budget ϵ tighten the (multiplicative) influence of a participant joining dataset $\mathcal{D}_2$ to form $\mathcal{D}_1$, bounding the probability of any downstream privacy release. The confidence parameter δ then relaxes this guarantee, such that no bound on the privacy loss is provided for low-probability events.

To bound the residual risk from ADP, Rényi-DP was introduced by Mironov (2017). Rényi-DP quantifies privacy through sequences of function composition, as required when iteratively training a deep net on sensitive data, for example. As we shall see in this paper, this tighter analysis leads to improved certifications in practice.

Definition 5 (Rényi divergence)**.** Let P and Q be two distributions on $\mathcal{X}$ defined over the same probability space, and let p and q be their respective densities. The Rényi divergence of finite order $\alpha \neq 1$ between P and Q is defined as

$$\mathrm{D}_\alpha(P \| Q) \triangleq \frac{1}{\alpha - 1} \ln \int_{\mathcal{X}} q(x) \left(\frac{p(x)}{q(x)} \right)^\alpha \mathrm{d}x \ . \tag{4}$$

Definition 6 (Rényi Differential Privacy)**.** A randomised function M preserves (α, ϵ)-Rényi-DP, with $\alpha > 1, \epsilon > 0$, if for all datasets $\mathcal{D}_1$ and $\mathcal{D}_2 \in \mathcal{B}(\mathcal{D}_1, 1)$:

$$\mathrm{D}_\alpha \left(M(\mathcal{D}_1) \| M\left(\mathcal{D}_2\right) \right) \leq \varepsilon \ . \tag{5}$$

Definition 7 (Outcomes guarantee)**.** Let $\mathcal{K}$ be a set of strictly monotonic functions on the reals, and r a natural number. A randomised function M is said to preserve a $(\mathcal{K}, r)$-*outcomes guarantee* if for any $K \in \mathcal{K}$ such that for all datasets $\mathcal{D}_1$ and $\mathcal{D}_2 \in \mathcal{B}(\mathcal{D}_1, r)$,

$$\Pr[M(\mathcal{D}_1) \in \mathcal{S}] \leq \mathrm{K}(\Pr[M(\mathcal{D}_2) \in \mathcal{S}]) \ . \tag{6}$$

Both ADP and RDP are generalised as specific cases of the outcome guarantee with, respectively,

$$\mathcal{K}_{\epsilon,\delta}(x) = \exp(\epsilon)x + \delta \tag{7}$$

$$\mathcal{K}_{\epsilon,\alpha}(x) = (\exp(\epsilon)x)^{\frac{\alpha-1}{\alpha}} \ . \tag{8}$$

The RDP's family is obtained by applying Hölder's inequality to the integral of the density function in the Rényi divergence (Mironov 2017).

This definition formalises a discussion on "bad-outcomes guarantee" due to Mironov (2017). With this definition, we are able to generalise our certification framework to the essential structure across variations of differential privacy. Note this definition incorporates *group privacy* (Dwork et al. 2006): extending DP to pairs of datasets that differ in up to r datapoints $\mathcal{D}_1 \in \mathcal{B}(\mathcal{D}_2, r)$.

Our framework also relies upon the *post-processing* property (Dwork et al. 2006) of DP: any computation applied to the output of a DP algorithm retains the same DP guarantee. This property, which simplifies the DP analysis in multi-layered models acting on a DP-preserving input, holds for any ADP, RDP, or indeed outcome-guaranteed mechanism.

Sampled Gaussian Mechanism with Improved Group Privacy. While many DP mechanisms have been proposed and widely studied for machine learning (Abadi et al. 2016; Mironov, Talwar, and Zhang 2019), they typically rely upon the addition of noise to training samples. In contrast, the Sampled Gaussian Mechanism (SGM) (Mironov, Talwar, and Zhang 2019) adds randomness both through the injection of noise and the sub-sampling process. Each element of the training batch is sampled without replacement with uniform probability q from the training dataset, while each weight update step also introduces additional additive gaussian noise to the gradients. When applied to a model M, the SGM has been shown (Mironov, Talwar, and Zhang 2019) to preserve (α, ϵ)-Renyi-DP, where ϵ is determined by the parameters (α, M, q, σ). We denote the computation steps of ϵ in SGM as function SG such that $\epsilon = \mathrm{SG}(\alpha, M, q, \sigma)$.

However, this guarantee fails to exploit the advantages given by Rényi-DP group privacy under the SGM. As such, by constructing our group privacy in a manner specific to the SGM, we are able to produce *tighter bounds* than prior works (Mironov 2017), producing the following pointwise guarantee of certification.

Theorem 8 (Improved Rényi-DP group privacy under the SGM)**.** *If a randomised function M obtained by SGM with sample ratio q and noise level σ achieves $(\alpha, \mathrm{SG}(\alpha, M, q, \sigma))$-Rényi-DP for all datasets $\mathcal{D}_1$ and $\mathcal{D}_2 \in \mathcal{B}(\mathcal{D}_1, 1)$, then for all datasets $\mathcal{D}_3 \in \mathcal{B}(\mathcal{D}_1, r)$*

$$\mathrm{D}_\alpha \left(M(\mathcal{D}_1) \| M\left(\mathcal{D}_3\right) \right) \leq \mathrm{SG}(\alpha, M, 1 - (1-q)^r, \sigma) \ . \tag{9}$$

Proof. (Here we provide a proof sketch, the detailed proof is available in Appendix A.2.) In the work (Mironov, Talwar, and Zhang 2019), they proposed calculating the amount of

	Training-time threat model		*Testing-time certification*	
	Modification	Addition/ Deletion	Statistical certification	Pointwise certification
Statistical DP (Ma, Zhu, and Hsu 2019)	✓	✓	✓	✗
Randomized smoothing (Rosenfeld et al. 2020; Weber et al. 2021)	✓	✗	✓	✓
Bagging (Jia, Cao, and Gong 2020; Levine and Feizi 2021)	✓	✓	✓	✓
This Paper	✓	✓	✓	✓

Table 1: A summary of different approaches of certified defence against data poisoning attacks. We investigate them from two perspectives. The training-time threat model: whether it permits the more general addition/deletion of training samples or only modification. The testing-time certification: whether it provides the more strict pointwise certification for each test sample or only statistical certification over all test samples.

Rényi-DP obtained from SGM in their Theorem 4. We extend it from "adjacent datasets" to "datasets that differ in up to r examples". We consider the datasets S and $S' = S \cup \{x_1, x_2, ..., x_r\}$, and calculate the mixing of distributions of taking a random subset of S' by the SGM $\mathcal{M}$ where each element of S' is independently placed with probability q as

$$\mathcal{M}\left(S'\right) = \sum_{T} p_T \left(\sum_{k=0}^{r} \binom{r}{k} q^k (1-q)^{r-k} \mathcal{N}\left(\mu, \sigma^2 \mathbb{I}^d\right) \right)$$
$$V \subseteq \{x_1, x_2, ..., x_r\} \qquad k = \|V\| \ .$$

We can complete the proof by replacing the original $\mathcal{M}(S')$ by the above distribution, and by following the remainder of the paper. □

Outcomes-Guaranteed Certifications

While pointwise-certified robustness guarantees can be applied to the output of any model, within this work we highlight both multinomial outputs and scored outputs.

Lemma 9 (Pointwise outcomes guarantee)**.** *Consider a randomised learner M that preserves a $(\mathcal{K}, r)$-outcome guarantee, and an arbitrary (possibly randomised) inference function I mapping learned parameters and the input instance to an inferred score. Then for any $K \in \mathcal{K}$ such that, for any input instance $\mathbf{x}$, label $l \in \mathcal{L}$, training datasets $\mathcal{D}_1$ and $\mathcal{D}_2 \in \mathcal{B}(\mathcal{D}_1, r)$,*

$$I_l\left(\mathbf{x}, M\left(\mathcal{D}_1\right)\right) \leq \mathrm{K}\left(I_l\left(\mathbf{x}, M\left(\mathcal{D}_2\right)\right)\right)$$
$$I_l\left(\mathbf{x}, M\left(\mathcal{D}_1\right)\right) \geq \mathrm{K}^{-1}\left(I_l\left(\mathbf{x}, M\left(\mathcal{D}_2\right)\right)\right) \ .$$

Proof. In the case of multinomial outputs, the first inequality follows from the post-processing property: the composition $I \circ M$ preserves the same outcome guarantee. The second inequality follows by symmetry in the roles of $\mathcal{D}_1, \mathcal{D}_2$ and by K being strictly increasing. To admit scored outputs, the probabilities $\Pr[M(\mathcal{D}) \in \mathcal{S}]$ in $(\mathcal{K}, r)$-outcome guarantee need to be converted into expected values $\mathbb{E}[M(\mathcal{D})]$. To that end, the integral over the right-tail distribution function of the probabilities in Definition 7 are taken. The expected value $(\mathcal{K}, r)$-outcome guarantee of (ϵ, δ)-ADP and (α, ϵ)-Rényi-DP can be shown to take the same form of Equation (7) and Equation (8) by Lecuyer et al. (2019) and Hölder's inequality (as detailed in Appendix A.3) respectively. □

The main result of this section establishes conditions under which a DP learner provides pointwise-certified robustness against general poisoning attacks up to size r.

Theorem 10 (Pointwise-certified robustness by outcomes guarantee)**.** *Consider a training dataset $\mathcal{D}$, an input instance $\mathbf{x}$, and a randomised learner M that preserves a $(\mathcal{K}, r)$-outcomes guarantee. Let*

$$l_1 = \arg\max_{l \in \mathcal{L}} I_l(\mathbf{x}, M(\mathcal{D}))$$

denote the label predicted on $\mathbf{x}, \mathcal{D}$ under multinomial interpretation of Definition 1. If there exist $\mathrm{K}_{upper}, \mathrm{K}_{lower} \in \mathcal{K}$ such that

$$\mathrm{K}_{lower}^{-1}(I_{l_1}(\mathbf{x}, M(\mathcal{D}))) > \max_{l \in \mathcal{L} \setminus \{l_1\}} \mathrm{K}_{upper}(I_l(\mathbf{x}, M(\mathcal{D}))) \quad ,$$

then $I \circ M$ is pointwise-certified robust to radius r about dataset $\mathcal{D}$ at input $\mathbf{x}$ (see Definition 3).

The proof can be found in Appendix A.1.

Algorithmic Implementation

The very nature of data poisoning attacks intrinsically requires pointwise certifications to incorporate modifications to both the training and testing processes. The remainder of this section illustrates the steps required to produce a prediction and certification pair (l, r) for a test time sample $\mathbf{x}$ in the testing dataset $\mathcal{D}_e$, the details of which are further elaborated over the Algorithm 1.

Training. Any certification using the aforementioned DP based processes inherently requires the model M_{DP} to be randomised. The SGM achieves the requisite randomness for DP via sub-sampling and injecting noise. The randomised model M_{DP} is instanced such that $(\hat{M}_{DP_1}, \hat{M}_{DP_2}, ..., \hat{M}_{DP_P})$. As each instance is a model with an identical training process, the training of such is an embarrassingly parallel process, a fact that can be leveraged to improve training efficiency. Further efficiencies for larger datasets can be found by incorporating training over subset $D_{sub} \subseteq D$. Under the SGM, the total privacy cost with regards to $\mathcal{D}$ is calculated by accumulating the privacy cost of each update with a subsample from $\mathcal{D}$. Therefore, we can analogously compute the privacy cost of using a sub-training dataset $D_{sub} \subseteq D$ with regards to the entire training dataset $\mathcal{D}$ by reducing the number of updates under the SGM.

Rather than exploiting the SGM, an alternate approach is to construct sub-training datasets across a set of model instances through bagging. Taking such an approach allows the model instance to work on a subset solely without knowing the entire training dataset. The privacy gains can be quantified by way of DP amplification (Balle, Barthe, and Gaboardi 2018). However, both SGM and bagging yield a level of privacy can then be translated into certifications of size r by Theorem 8 by deriving the set of outcomes guarantee bound functions $\mathcal{K}$ through Algorithm 1.

Certification. In general, the certification involves estimating the upper and lower bounds of inferred scores for each label and searching for the maximum radius that satisfies Theorem 10. The *multinomial label* and *probability scores* require similar but slightly different treatments. For the former, each testing sample $\mathbf{x}_i$ is passed through the set of model instances $(\hat{M}_{DP_1}, \hat{M}_{DP_2}, ..., \hat{M}_{DP_P})$. From this, the top-2 most frequent labels are selected and labelled as l_{1i} and l_{2i}. Uncertainties from sampling are then quantified through the lower and upper confidence bounds of $\Pr[M_{DP}(\mathbf{x}_i, \mathcal{D}) = l_{1i}]$ and $\Pr[M_{DP}(\mathbf{x}_i, \mathcal{D}) = l_{2i}]$, which are constructed to a confidence level $1 - \eta$ by the SIMUEM method of Jia, Cao, and Gong (2020), yielding $\underline{p_{l_{1i}}}$ and $\overline{p_{l_{2i}}}$ respectively.

Algorithm 1 demonstrates that a binary search can then be used to identify the maximum certified radius r_i of the optimisation problem in Theorem 10, subject to

$$\mathrm{K}_{lower}^{-1}(\Pr[M_{DP}(\mathbf{x}_i, \mathcal{D}) = l_{1i}]) = \mathrm{K}_{lower}^{-1}(\underline{p_{l_{1i}}})$$
$$\max_{l_{ji} \in \mathcal{L} \setminus \{l_{1i}\}} \mathrm{K}_{upper}(\Pr[M_{DP}(\mathbf{x}_i, \mathcal{D}) = l_{ji}]) = \mathrm{K}_{upper}(\overline{p_{l_{2i}}}) \tag{10}$$

Here the bound functions $\mathrm{K}_{upper}, \mathrm{K}_{lower} \in \mathcal{K}$ being derived by Theorem 9. The outputs for a testing sample $\mathbf{x}_i$ are the predicted label l_{1i} with certified radius r_i.

The process for the *probability scores* case is similar but involves collecting the probability scores from each model instance and computing the confidence interval for the expected values $\mathbb{E}[y_{l_i}(\mathbf{x}_i, M_{DP}(\mathcal{D})]$ via Hoeffding's inequality (Hoeffding 1963) or empirical Bernstein bounds (Maurer and Pontil 2009).

Experiments

To verify the effectiveness of our proposed pointwise-certified defence, we conducted experiments across MNIST, Fashion-MNIST, and CIFAR-10 for varying levels of added noise σ. For MNIST and Fashion-MNIST, training occurred using the LeNet-5 architecture (Lecun et al. 1998), with class probabilities/expectations estimated based upon 1000 model instances trained on the entire dataset. In contrast, CIFAR-10 was trained upon the example model from Opacus tutorial (Yousefpour et al. 2021) (Opa-tut) with rather simple architecture, and more complex ResNet-18 (He et al. 2015) for comprehensive evaluation. Both were estimated based upon 500 instances trained on sub-datasets of size 10000.

Across all experiments adjust the sample ratio q to have a batch size of 128, with training conducted using ADAM with a learning rate of 0.01 optimising the Cross-Entropy loss. The clip size C is fine-tuned for each experiment (around 1.0 on MNIST, 25.0 on CIFAR-10). In each case, uncertainties were estimated for a confidence interval suitable for $\eta = 0.001$. All experiments were conducted in Pytorch using a single NVIDIA RTX 2080 Ti GPU with 11 GB of GPU RAM.

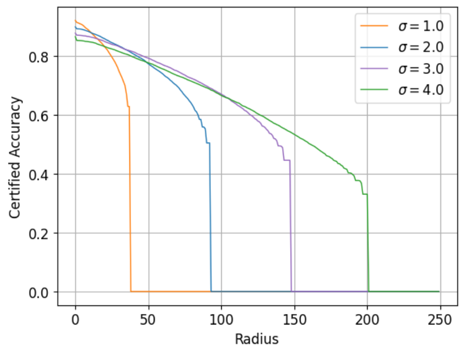
To quantify performance the proportion of samples correctly predicted with a certification of at least r was used, henceforth known as the *certified accuracy*. This quantity takes the form

$$CA_r = \frac{\sum_{\mathbf{x}_i \in \mathcal{D}_e} \mathbb{I}(l_i = y_i) \cdot \mathbb{I}(r_i \geq r)}{|\mathcal{D}_e|} \quad , \tag{11}$$

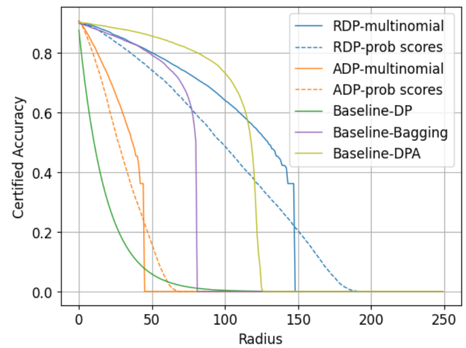
where $\mathbf{x}_i$ and y_i are the input instances and corresponding ground truth labels for a testing sample, and l_i, and r_i are the predicted label and corresponding certified radius returned by the defence model. We also investigate the median and maximum value of certification achieved among all samples.

We further divide our experiments into four different frameworks. These are ADP with either multinomial labels (ADP-multinomial) or probability scores (ADP-prob-scores) output, and then Rényi-DP with either multinomial labels (RDP-multinomial) or probability scores (RDP-prob-scores) output. In each case, Theorem 10 is employed to generate a guaranteed certificate of defence to data poisoning attacks.

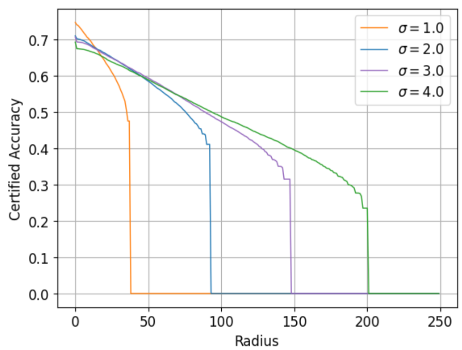
To validate the efficacy of our technique, these results are considered against prior works, specifically the DP-based defence method of Ma, Zhu, and Hsu (2019) (Baseline-DP), the bagging-based defence of Jia, Cao, and Gong (2020); Chen et al. (2020) (Baseline-Bagging) and deterministic Deep Partition Aggregation (DPA) method of (Levine and Feizi 2021) (Baseline-DPA). Of these, conceptual similarities between our work and DP-baseline allow both techniques to be compared while utilising the same trained models. However, it must be noted that Ma, Zhu, and Hsu (2019) bound the DP-baseline in terms of statistically certified accuracy which is calculated as the lower bound of expected accuracy with confidence level $1 - \eta$ among obtained model instances. As for Bagging-baseline, it provides the same pointwise-certified defence as we do. Hence, by letting the number of base classifiers equal the number of model instances and adjusting the size of sub-training datasets, we force the Bagging-baseline to have the same certified accuracy at radius $r = 0$. The DPA method has significant differences between their underlying assumptions and ours. The DPA only applies to the models that are *deterministic*, which means for a given training dataset the parameters in the resulting model should always be the same. This approach requires specific model architectures and a deterministic training process while our method applies to more general situations. Compared with standard training approaches, the extra step involved in incorporating



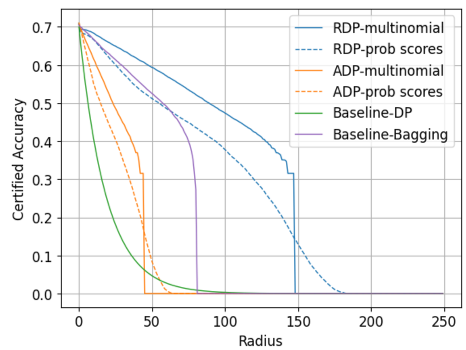
(a) MNIST (LeNet-5), performance against different σ.



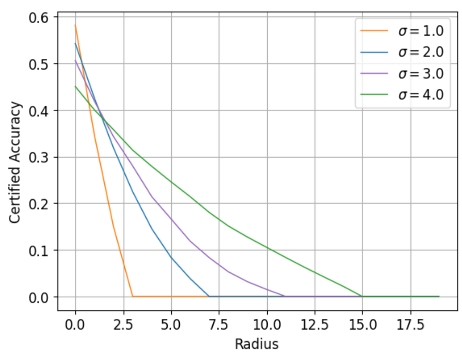
(b) MNIST (LeNet-5), comparative performance at $\sigma = 3.0$.



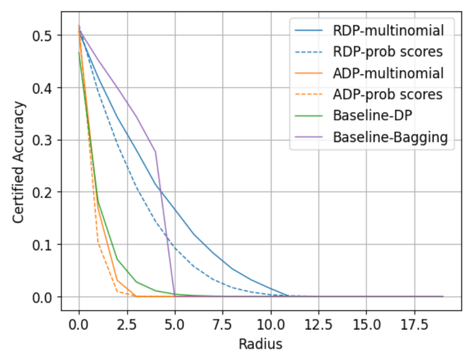
(c) Fashion-MNIST (LeNet-5), performance against different σ.



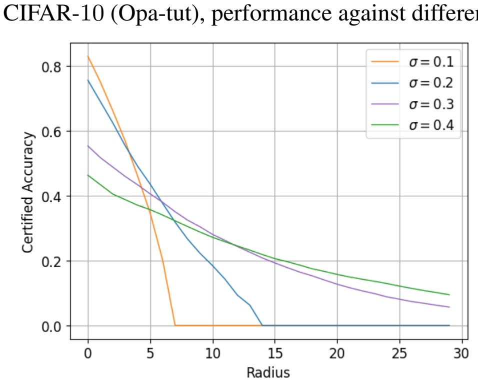
(d) Fashion-MNIST (LeNet-5), comparative performance at $\sigma = 3.0$.



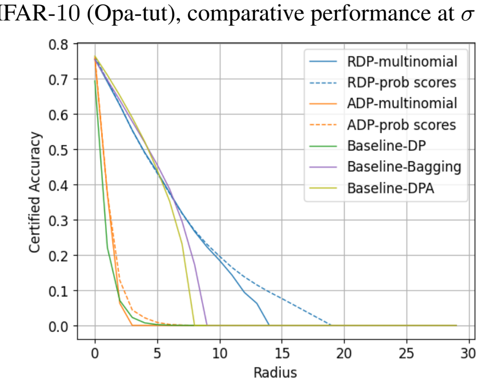
(e) CIFAR-10 (Opa-tut), performance against different σ.



(f) CIFAR-10 (Opa-tut), comparative performance at $\sigma = 3.0$.



(g) CIFAR-10 (ResNet-18), performance against different σ.



(h) CIFAR-10 (ResNet-18), comparative performance at $\sigma = 0.2$.

Figure 1: The left column contains certified accuracy plots for the method RDP-multinomial against different noise levels (σ); the right column contains certified accuracy plots for comparisons against variants and baselines. In the plots, the X-axis is radius r (symmetric difference) while the Y-axis is the corresponding certified accuracy CA_r at radius r.

Algorithm 1: Certifiably Robust Differentially Private Defence Algorithm

Input: model M_{DP}, training dataset $\mathcal{D}$, testing dataset $\mathcal{D}_e$, number of instances P, confidence interval $1-\eta$
$(\hat{M}_{DP_1}, \hat{M}_{DP_2}, ..., \hat{M}_{DP_P}) \leftarrow$ DifferentialPrivateTrain(M, P) {Train DP model instances}
for each $x_i \in \mathcal{D}_e$ **do**
  counts$[j] \leftarrow \sum_{o=1}^{P} \mathbb{I}\left(\hat{M}_{DP_o}(\mathbf{x}_i) = j\right), j \in \mathcal{L}$ {Count votes for each label}
  $l_{1i}, l_{2i} \leftarrow$ top two indices in counts
  $\underline{p_{l_{1i}}}, \overline{p_{l_{2i}}} \leftarrow$ SimuEM(counts, $1-\eta$) {Calculate the corresponding lower and upper bounds}
  if $\underline{p_{l_{1i}}} \geq \overline{p_{l_{2i}}}$ **then**
    $r_i \leftarrow$ BinarySearchForCertifiedRadius$(\underline{p_{l_{1i}}}, \overline{p_{l_{2i}}}, \|D\|)$ {Calculate certified radius if possible}
  else
    $r_i \leftarrow$ ABSTAIN {Unable to certify}
  end if
end for
Output: $l_{11}, l_{12}, .., l_{1e}$ and $r_1, r_2, .., r_e$

SGM introduces a negligible difference in training time. Note the change in the relative performance of Baseline-Bagging and Baseline-DPA from the original papers are the product of different model architectures. We ensure all methods apply the same model architecture for fair comparisons (Appendix A.4).

Figure 1 demonstrates that our method consistently provides a more robust certified defence, across the full suite of experiments. In the case of MNIST and Fashion-MNIST, for a given radius, RDP-multinomial is capable of providing the highest certified accuracy in most cases, which means more testing samples are certified to be correctly predicted within this radius. For example, in the experiments on Fashion-MNIST, RDP-multinomial achieves 52.21% certified accuracy at radius $r = 80$, whereas the other baselines only achieve at most 27.23% certified accuracy. Additionally, our method can generate the largest certification as shown, which provides a better defence for the confident testing samples. As illustrated in the experiments on CIFAR-10 for both Opatut and ResNet-18 models, RDP-prob-scores outperform the other baselines with regard to the largest certified radius by doubling the size. Based upon these results, when considering Fashion-MNIST our method achieves a 56% and 130% improvement in the median and maximum value respectively when compared to Baseline-Bagging (further details of this can be found in Appendix A.6).

As the bound functions are the same in both multinomial and probability scores methods, the difference between them can be directly attributed to the differences in how these techniques construct their upper and lower bounds. As indicated in Theorem 10, the larger the gap between the lower and upper bounds, the larger radius it can certify. Intuitively, if the defence model is confident with the predicted label of an easy testing sample, then this sample should be more resilient to poisoned examples in the training dataset. In the multinomial method, the uncertainty within each model instance is ignored by selecting a single label, while the uncertainty remains in the probability scores method. As a consequence of this, the multinomial method provides a higher radius for moderately confident examples but the probability scores method is able to certify a larger radius for the very confident ones. Further improvements can be found in the application of Renyi-DP, relative to Approximate-DP, due to the former providing a more precise accounting of model privacy. This in turn allows tighter bounds to be constructed, with performance further enhanced by way of Theorem 8.

The influence of the magnitude of injected noise σ is shown in the left-hand column of Figure 1. These results broadly align with previous works, in that adding more noise can produce larger robustness guarantees (larger certified radius), at the cost of decreased predictive performance upon un-attacked datasets ($r = 0$). The increase of semantic-complexity of the dataset also limits the tolerance of the noise. It is also important to note that the sample rate ($q \in (0, 1]$) and robustness are negatively correlated, as increasing the sample rate requires that more training examples are utilised in constructing the output, which provides weaker privacy guarantees. Therefore, a grid search is usually required to find the best combination of parameters (σ, q, clip size).

Limitations and Future Directions The nature of the SGM inherently requires a significant allocation of computational resources, due to the need to train multiple models from scratch in parallel. While improvements in these resource demands may be possible, at this stage any direct application of this work would likely be restricted to systems that are considered particularly sensitive to adversarial behaviours. We also note that while this work improves upon the achievable bounds for certification by exploiting RDP in the context of the SGM, further gains may be possible by extending these proofs to Approximate DP via the conversion from RDP to ADP (Balle et al. 2019).

Conclusion

By carefully exploiting both DP, SGM, and bagging, this work presents a mechanism for tightening guarantees of pointwise-certified robustness relative to prior implementations. This is made possible by calculating group privacy directly from the SGM. When compared to the current state-of-the-art, our technique can produce a more than 50% improvement in the median certification.

Acknowledgements

This research was undertaken using the LIEF HPC-GPGPU Facility hosted at the University of Melbourne. This Facility was established with the assistance of LIEF Grant LE170100200. This work was also supported in part by the Australian Department of Defence Next Generation Technologies Fund, as part of the CSIRO/Data61 CRP AMLC project. Sarah Erfani is in part supported by the Australian Research Council (ARC) Discovery Early Career Researcher Award (DECRA) DE220100680.

References

Abadi, M.; Chu, A.; Goodfellow, I.; McMahan, H. B.; Mironov, I.; Talwar, K.; and Zhang, L. 2016. Deep Learning with Differential Privacy. *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, 308–318. ArXiv: 1607.00133.

Balle, B.; Barthe, G.; and Gaboardi, M. 2018. Privacy Amplification by Subsampling: Tight Analyses via Couplings and Divergences. *arXiv:1807.01647 [cs, stat]*. ArXiv: 1807.01647.

Balle, B.; Barthe, G.; Gaboardi, M.; Hsu, J.; and Sato, T. 2019. Hypothesis Testing Interpretations and Renyi Differential Privacy. *arXiv:1905.09982 [cs, stat]*. ArXiv: 1905.09982.

Barreno, M.; Nelson, B.; Sears, R.; Joseph, A. D.; and Tygar, J. D. 2006. Can Machine Learning Be Secure? In *Proceedings of the 2006 ACM Symposium on Information, Computer and Communications Security*, 16–25.

Biggio, B.; Fumera, G.; and Roli, F. 2014. Security Evaluation of Pattern Classifiers under Attack. *IEEE Transactions on Knowledge and Data Engineering*, 26(4): 984–996. Conference Name: IEEE Transactions on Knowledge and Data Engineering.

Biggio, B.; Nelson, B.; and Laskov, P. 2013. Poisoning Attacks against Support Vector Machines. *arXiv:1206.6389 [cs, stat]*. ArXiv: 1206.6389.

Carnerero-Cano, J.; Muñoz-González, L.; Spencer, P.; and Lupu, E. C. 2020. Regularisation Can Mitigate Poisoning Attacks: A Novel Analysis Based on Multiobjective Bilevel Optimisation. *arXiv:2003.00040 [cs, stat]*. ArXiv: 2003.00040.

Chakraborty, A.; Alam, M.; Dey, V.; Chattopadhyay, A.; and Mukhopadhyay, D. 2018. Adversarial Attacks and Defences: A Survey. *arXiv:1810.00069 [cs, stat]*. ArXiv: 1810.00069.

Chen, R.; Li, J.; Wu, C.; Sheng, B.; and Li, P. 2020. A Framework of Randomized Selection Based Certified Defenses Against Data Poisoning Attacks. *arXiv:2009.08739 [cs, stat]*. ArXiv: 2009.08739.

Chen, X.; Liu, C.; Li, B.; Lu, K.; and Song, D. 2017. Targeted Backdoor Attacks on Deep Learning Systems Using Data Poisoning. *arXiv:1712.05526 [cs]*. ArXiv: 1712.05526.

Cohen, J. M.; Rosenfeld, E.; and Kolter, J. Z. 2019. Certified Adversarial Robustness via Randomized Smoothing. *arXiv:1902.02918 [cs, stat]*. ArXiv: 1902.02918.

Dwork, C.; McSherry, F.; Nissim, K.; and Smith, A. 2006. Calibrating Noise to Sensitivity in Private Data Analysis. In *Theory of cryptography conference*, 265–284. Springer.

Friedman, A.; and Schuster, A. 2010. Data Mining with Differential Privacy. In *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*, 493–502.

Goldblum, M.; Tsipras, D.; Xie, C.; Chen, X.; Schwarzschild, A.; Song, D.; Madry, A.; Li, B.; and Goldstein, T. 2021. Dataset Security for Machine Learning: Data Poisoning, Backdoor Attacks, and Defenses. *arXiv:2012.10544 [cs]*. ArXiv: 2012.10544.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2015. Deep Residual Learning for Image Recognition. *arXiv:1512.03385 [cs]*.

Hoeffding, W. 1963. Probability Inequalities for Sums of Bounded Random Variables. *Journal of the American Statistical Association*, 58(301): 13–30. Publisher: [American Statistical Association, Taylor & Francis, Ltd.].

Hong, S.; Chandrasekaran, V.; Kaya, Y.; Dumitraş, T.; and Papernot, N. 2020. On the Effectiveness of Mitigating Data Poisoning Attacks with Gradient Shaping. *arXiv:2002.11497 [cs]*. ArXiv: 2002.11497.

Jia, J.; Cao, X.; and Gong, N. Z. 2020. Intrinsic Certified Robustness of Bagging against Data Poisoning Attacks. *arXiv:2008.04495 [cs]*. ArXiv: 2008.04495.

Lecun, Y.; Bottou, L.; Bengio, Y.; and Haffner, P. 1998. Gradient-based Learning Applied to Document Recognition. *Proceedings of the IEEE*, 86(11): 2278–2324.

Lecuyer, M.; Atlidakis, V.; Geambasu, R.; Hsu, D.; and Jana, S. 2019. Certified Robustness to Adversarial Examples with Differential Privacy. *arXiv:1802.03471 [cs, stat]*. ArXiv: 1802.03471.

Levine, A.; and Feizi, S. 2021. Deep Partition Aggregation: Provable Defense against General Poisoning Attacks. *arXiv:2006.14768 [cs, stat]*. ArXiv: 2006.14768.

Ma, Y.; Zhu, X.; and Hsu, J. 2019. Data Poisoning against Differentially-Private Learners: Attacks and Defenses. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, 4732–4738. Macao, China: International Joint Conferences on Artificial Intelligence Organization. ISBN 978-0-9992411-4-1.

Maurer, A.; and Pontil, M. 2009. Empirical Bernstein Bounds and Sample Variance Penalization. *arXiv:0907.3740 [stat]*. ArXiv: 0907.3740.

Mironov, I. 2017. Renyi Differential Privacy. *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, 263–275. ArXiv: 1702.07476.

Mironov, I.; Talwar, K.; and Zhang, L. 2019. Rényi Differential Privacy of the Sampled Gaussian Mechanism. *arXiv:1908.10530 [cs, stat]*.

Paudice, A.; Muñoz-González, L.; Gyorgy, A.; and Lupu, E. C. 2018. Detection of Adversarial Training Examples in Poisoning Attacks through Anomaly Detection. *arXiv:1802.03041 [cs, stat]*. ArXiv: 1802.03041.

Rosenfeld, E.; Winston, E.; Ravikumar, P.; and Kolter, Z. 2020. Certified Robustness to Label-Flipping Attacks via Randomized Smoothing. In *Proceedings of the 37th International Conference on Machine Learning*, 8230–8241. PMLR. ISSN: 2640-3498.

Shafahi, A.; Huang, W. R.; Najibi, M.; Suciu, O.; Studer, C.; Dumitras, T.; and Goldstein, T. 2018. Poison Frogs! Targeted Clean-Label Poisoning Attacks on Neural Networks. *arXiv:1804.00792 [cs, stat]*. ArXiv: 1804.00792.

Wang, B.; Cao, X.; jia, J.; and Gong, N. Z. 2020. On Certifying Robustness against Backdoor Attacks via Randomized Smoothing. *arXiv:2002.11750 [cs]*. ArXiv: 2002.11750.

Weber, M.; Xu, X.; Karlaš, B.; Zhang, C.; and Li, B. 2021. RAB: Provable Robustness Against Backdoor Attacks. *arXiv:2003.08904 [cs, stat]*. ArXiv: 2003.08904.

Xiao, H.; Xiao, H.; and Eckert, C. 2012. Adversarial Label Flips Attack on Support Vector Machines. *ECAI 2012*, 870–875. Publisher: IOS Press.

Yousefpour, A.; Shilov, I.; Sablayrolles, A.; Testuggine, D.; Prasad, K.; Malek, M.; Nguyen, J.; Ghosh, S.; Bharadwaj, A.; Zhao, J.; Cormode, G.; and Mironov, I. 2021. Opacus: User-Friendly Differential Privacy Library in PyTorch. *arXiv preprint arXiv:2109.12298*.