

USENIX

THE ADVANCED COMPUTING SYSTEMS ASSOCIATION

Sliced Rényi Pufferfish Privacy: Tractable Privatization Mechanism and Private Learning with Gradient Clipping

Tao Zhang and Yevgeniy Vorobeychik, *Washington University in St. Louis*

https://www.usenix.org/conference/usenixsecurity26/presentation/zhang-tao-pufferfish

This paper is included in the Proceedings of the 35th USENIX Security Symposium.

August 12–14, 2026 • Baltimore, MD, USA

ISBN 978-1-939133-58-8

Open access to the Proceedings of the 35th USENIX Security Symposium is sponsored by



جامعة الملك عبدالله للعلوم والتقنية

King Abdullah University of Science and Technology

ARTIFACT EVALUATED
USENIX
AVAILABLE

Sliced Rényi Pufferfish Privacy: Tractable Privatization Mechanism and Private Learning with Gradient Clipping

Tao Zhang
Washington University in St. Louis

Yevgeniy Vorobeychik
Washington University in St. Louis

Abstract

We study the design of a privatization mechanism and privacy accounting in the Pufferfish Privacy (PP) family. Specifically, motivated by the curse of dimensionality and lack of practical composition tools for iterative learning in the recent Rényi Pufferfish Privacy (RPP) framework, we propose *Sliced Rényi Pufferfish Privacy (SRPP)*. SRPP preserves PP/RPP semantics (customizable secrets with probability-aware secret–dataset relationships) while replacing high-dimensional Rényi divergence with projection-based quantification via two sliced measures, *Average SRPP* and *Joint SRPP*. We develop *sliced Wasserstein mechanisms*, yielding sound SRPP certificates and closed-form Gaussian noise calibration. For iterative learning systems, we introduce an SRPP-SGD scheme with gradient clipping and new accountants based on *History-Uniform Caps (HUC)* and a *subsampling-aware* variant (sa-HUC), enabling decompose-then-compose privatization and additive composition under a common slicing geometry. Experiments on static and iterative privatization show that the proposed framework exhibits favorable privacy–utility trade-offs, as well as practical scalability.

1 Introduction

Differential Privacy (DP) [12] is the de facto standard for measuring data privacy risk. Under DP semantics, the secret is an individual record's presence or absence, and an adversary with arbitrary side information cannot draw substantially different conclusions from the released output about that inclusion. DP quantifies risk via ε-DP or (ε,δ)-DP, i.e., worst-case indistinguishability between neighboring datasets, equivalently expressed through bounds on max divergence and hockey-stick divergence, respectively. Rényi DP (RDP) [27, 32] preserves DP semantics but measures privacy using (α,ε)-Rényi divergence.

Pufferfish Privacy (PP) [21] generalizes DP semantics by allowing customizable secrets and modeling probabilistic secret–dataset relationships through *priors* that may represent subjective adversarial beliefs or an objective secret-data-generation process [46]. ε-PP and (ε,δ)-PP [49] adopt the same indistinguishability measures as DP, while Rényi Pufferfish Privacy (RPP) [34] shares PP semantics but uses (α,ε)-Rényi divergence.

Beyond semantics and a quantification measure, a useful privacy framework should provide implementable *privatization mechanisms*. For DP, Laplace [12] and Gaussian [11] mechanisms calibrate noise to query sensitivity. However, computing sensitivity is generally NP-hard [43]. DP and RDP composition properties underpin DP-SGD with gradient clipping [1], a *decompose-then-compose* approach for deep learning without sensitivity computation: each iteration enforces bounded sensitivity via gradient clipping, adds subsampled Gaussian noise, and a moments accountant converts per-iteration RDP costs into an overall (ε,δ)-DP guarantee.

The key technical difference between DP and PP is that PP involves a *probabilistic secret-dataset relationship*. Because a query only takes the realized dataset as input, PP analysis must account for the induced mapping from secret to output through the prior distributions. Consequently, practical PP calibration depends on prior-driven distributional quantities and is typically carried out from finite samples and estimators, so error-aware, assumption-dependent guarantees are often unavoidable for useful privacy-utility trade-offs.

For Pufferfish privatization, Song et al. [38] provide the first general-purpose privatization mechanism, the *Wasserstein mechanism*, which uses the ∞-Wasserstein distance as an analogue of sensitivity and generalizes the Laplace mechanism. Ding [10] proposes a Kantorovich-based exponential (and Gaussian) additive-noise mechanism for PP calibrated to 1-Wasserstein sensitivity. RPP [34] introduces the General Wasserstein Mechanism (GWM), which uses the shift-reduction lemma [14] to calibrate additive noise via ∞-Wasserstein distance. Extensions of GWM include a distribution-aware variant (DAGWM) using p-Wasserstein distances to capture average-case data geometry. While these Wasserstein mechanisms significantly advance the applicability of the PP family frameworks, they inherit the curse

of dimensionality from optimal transport (OT): estimating Wasserstein distances and sensitivities remains statistically and computationally expensive in high dimensions, limiting practical applicability. The tractable general mechanism design principles for PP family remain largely open [18].

Moreover, PP/RPP lack practical, general-purpose composition theorems for privacy accounting in stochastic optimization algorithms such as SGD. Existing results rely on restrictive structure (e.g., universally composable evolution scenarios [21] or Bayesian networks via Markov quilts [38]). Recent work [46] shows the obstacle is fundamental: probabilistic secret–dataset relationships induce inter-mechanism dependencies that break standard composition; their inverse composition requires solving an infinite-dimensional convex program, making integration with iterative learning nontrivial. RPP's Privacy Amplification by Iteration (PABI) [34] enables privacy guarantees for iterative convex optimization under contractivity, but still depends on Wasserstein distance computation, reintroducing high-dimensional OT costs.

Therefore, making PP-family guarantees operational in practice requires resolving two obstacles, both rooted in the probabilistic secret–dataset relationships encoded by priors: (i) a *curse of dimensionality in mechanism design*, namely calibrating noise without intractable high-dimensional Wasserstein computations; and (ii) a *lack of graceful composition for iterative optimization and learning*, namely enabling decompose-then-compose privatization for stochastic methods such as SGD under Pufferfish semantics.

Contributions. We address these obstacles by introducing *Sliced Rényi Pufferfish Privacy (SRPP)* (Sec. 4), a new Pufferfish-family privacy notion, together with two privatization routes with provable guarantees:

(i) the *Sliced Wasserstein Mechanism (SWM)* (Sec. 4.3), a direct mechanism calibrated by an efficiently computable *sliced Wasserstein sensitivity* (*SW-sensitivity*) (Sec. 3) that circumvents the curse of dimensionality of OT;

(ii) a decompose-then-compose scheme for iterative learning, based on *History-Uniform Caps (HUC)* and their *subsampling-aware* variant (*sa-HUC*) under gradient clipping, with matching SRPP/sa-SRPP accountants that avoid Wasserstein sensitivity computation altogether.

Sliced quantification. Calibrating a Wasserstein mechanism through full-dimensional Wasserstein sensitivity, as in RPP's General Wasserstein Mechanism, is bottlenecked by high-dimensional OT. Building on the sliced Wasserstein distance [3, 29, 35], we replace this sensitivity with SW-sensitivity, computed from one-dimensional (1-D) projections, and design SWM around it (Sec. 4.3). To align the privacy quantification with this sensitivity, we introduce two SRPP notions: *Average SRPP* (Sec. 4.1) and *Joint SRPP* (Sec. 4.2). Both share the Pufferfish semantics of PP/RPP but quantify privacy using *Average* and *Joint Sliced Rényi Divergence* (Ave-SRD and Joint-SRD), respectively.

Iterative learning. We develop an SRPP privatization route for iterative learning via HUC and sa-HUC, enabling DP-SGD-style privatization with gradient clipping *without* requiring contractivity of the update map. A central element of the contribution is that HUC/sa-HUC isolate a single scenario-dependent quantity—the *discrepancy cap* (in deterministic form K_t and subsampling-aware form $\overline{K}_t^2$)—that governs the per-iteration secret-induced shift. This identification (i) avoids the utility-draining worst-case "group-privacy" [34, 38] bound (see App. A.4 for a formal discussion), and (ii) enables a decompose-then-compose accountant for SGD with gradient clipping, without computing (sliced) Wasserstein sensitivities.

Composition of the SRPP-SGD mechanisms. For the HUC/sa-HUC-based SRPP-SGD mechanisms developed in this paper, we prove additive composition under a shared slice profile: when several such mechanisms act on the same dataset under a common slicing geometry, the total (sa-)Ave-/Joint-SRPP cost is bounded by the sum of the per-mechanism costs (Theorem 5.8). This gives modular accounting for pipelines and cascades built from these mechanisms.

Experiments on static query release and iterative learning demonstrate favorable privacy–utility trade-offs and practical scalability. A complete appendix, including formal proofs and detailed analysis, appears in the online extended version of this paper [48].

1.1 Related Work

Privacy Beyond Individual Data Records. One-sided DP [23] protects a designated sensitive subset of the dataset while allowing minimal or no protection for the remaining records. Fernandes et al. [15] use generalized DP with Earth Mover's Distance on word embeddings to obfuscate authorship. Hardt and Roth [16] study private singular vector computation under entry-level privacy, where each matrix entry is a secret. PAC Privacy [39, 41] and Residual PAC Privacy [47] bound the information-theoretic hardness of inference for general secrets, enabling simulation-based measurement and control. Pufferfish Privacy (PP) [21, 38] and Rényi Pufferfish Privacy (RPP) [34] generalize DP and RDP to arbitrary secrets within the same indistinguishability framework. Related distribution-level notions protect properties of the data distribution rather than individual records [4].

Instantiations of Pufferfish Privacy. PP has been instantiated for correlated data [38], attribute privacy [49], and confounding settings where secrets are data-derived [46]. Related lines include Blowfish, which uses policy constraints to flexibly specify what is protected and what is known [17], and mutual-information PP, an information-theoretic formulation based on conditional mutual information with improved composition [31]. Applied deployments include smart-meter time-series release and local energy markets [20] and temporally correlated time-series trading (HORAE) [30].

2 Preliminaries

In this section, we review Pufferfish-type privacy, with a main focus on Rényi Pufferfish Privacy (RPP) [34] and Wasserstein mechanisms, laying the necessary background for Sliced Rényi Pufferfish Privacy (SRPP). Formal introductions of DP and DP-SGD are deferred to App. A.1.

A distinguishability-based privacy framework can be viewed through three layers. The first is a *semantic interpretation*, specifying the protected secrets, the secret pairs to be made indistinguishable, and the assumptions under which the guarantee is required. The second is a *quantification measure*, specifying how distinguishability between the corresponding mechanism-output distributions is bounded. Together, these two layers characterize the privacy notion. The third layer consists of compatible *privatization mechanisms*, which provide practical ways to implement the guarantee. Guided by this perspective, Sec. 2.4 clarifies the status and interpretation of SRPP by separating its Pufferfish semantic goal from its sliced Rényi divergence quantification. It also explains how SRPP relates to DP, PP, and RPP. Sec. 2.5 then provides concrete intuition and real-world examples illustrating what types of information are, and are not, protected under these definitions.

2.1 Semantic Interpretation

DP has a clear semantic interpretation [11, 19]: *regardless of auxiliary knowledge, an adversary who observes the mechanism output should not reach substantially different conclusions depending on whether an individual record was included in the input database.*

A *Pufferfish scenario* $(\mathcal{S}, Q, \Theta)$ specifies the semantic layer. $\mathcal{S}$ is the set of protected secrets, and $Q \subseteq \mathcal{S} \times \mathcal{S}$ specifies which pairs of secrets should be made indistinguishable. Unlike DP, whose neighboring relation is defined directly over realized datasets, the Pufferfish-type framework models X as a random variable drawn from a distribution $\theta \in \Theta$, where each θ is a *prior*, i.e., a probabilistic model capturing either adversarial beliefs about the data or an admissible data-generating scenario, including secret–data relationships. Crucially, the Pufferfish family is not a prior-dependent relaxation of DP: privacy is defined uniformly over an admissible family of priors Θ and over secret pairs in Q. A Pufferfish-type guarantee therefore requires specifying Θ; practical mechanisms may additionally require estimating the relevant prior-driven sensitivity or output-distribution quantities.

Given $(\mathcal{S}, Q, \Theta)$, we can state a Pufferfish semantic interpretation (Pufferfish semantics) in the spirit of DP: *for every admissible prior $\theta \in \Theta$, an adversary observing the mechanism output should reach nearly indistinguishable conclusions about whether s_i or s_j holds, for every $(s_i, s_j) \in Q$.*

This Pufferfish semantics includes DP as a special case. In particular, DP can be represented through a Pufferfish scenario $(\mathcal{S}_{\mathrm{DP}}, Q_{\mathrm{DP}}, \Theta_{\mathrm{DP}})$ where $\mathcal{S}_{\mathrm{DP}}$ contains secrets about individual records, Q_{DP} pairs alternatives that differ only in one individual record, and Θ_{DP} restricts records to be mutually independent. Under this independence structure, conditioning on one individual's secret does not change the distribution of the remaining records, so the Pufferfish comparison reduces to the usual DP comparison between neighboring datasets. Thus, DP semantics can be viewed as the record-level, distribution-free special case of the broader Pufferfish semantics.

2.2 Quantification Measure and Definition

Any formal distinguishability-based privacy definition requires a quantification measure that specifies how indistinguishability is mathematically bounded. Although DP and PP have different semantics and scopes, classic PP [21, 49] uses the same DP-style classes of indistinguishability measures, primarily *max divergence* and *hockey-stick divergence*, but applies them to secret-conditioned output distributions induced by a Pufferfish scenario. RPP retains the Pufferfish semantics of PP while using *Rényi divergence* as the quantification measure for privacy loss, in direct analogy with how Rényi Differential Privacy (RDP) [27] uses Rényi divergence under the DP semantics.

For each $\theta \in \Theta$, let $P_\theta(S, X)$ denote the joint distribution over secrets S and datasets X, with marginals P_θ^S and P_θ^X, and conditional $P_\theta(x \mid s) = P_\theta(s, x)/P_\theta^S(s)$. We write $\Pr[\cdot \mid s, \theta]$ to denote the probability of an event under the conditional distribution $P_\theta(\cdot \mid s)$.

Definition 2.1 (Rényi Pufferfish Privacy (RPP) [34])**.** *Let $\alpha > 1$ and $\varepsilon \geq 0$. A mechanism $\mathcal{M}$ is (α, ε)-RPP in $(\mathcal{S}, Q, \Theta)$ if for all $\theta \in \Theta$ and $(s_i, s_j) \in Q$ with $P_\theta^S(s_i), P_\theta^S(s_j) > 0$,*

$$\mathtt{D}_\alpha\big(\Pr[\mathcal{M}(X) \mid S = s_i, \theta] \,\big\|\, \Pr[\mathcal{M}(X) \mid S = s_j, \theta]\big) \;\leq\; \varepsilon,$$

where $\mathtt{D}_\alpha$ is the Rényi divergence of order α:

$$\mathtt{D}_\alpha(\mathrm{P} \,\|\, \mathrm{Q}) := \tfrac{1}{\alpha - 1} \log \mathbb{E}_{Z \sim \mathrm{Q}}\left[\left(\tfrac{\mathrm{P}(Z)}{\mathrm{Q}(Z)}\right)^{\alpha}\right], \tag{1}$$

for distributions P *and* Q *on a common measurable space.*

RPP enjoys post-processing immunity. However, like classic PP, it does not provide the graceful composition behavior of the DP family [34]. This limits the direct use of RPP or PP to privatize iterative learning in a DP-SGD style.

We recall the PP definition here to highlight that PP and RPP share the same Pufferfish semantics but use different quantification measures. A mechanism $\mathcal{M}$ satisfies $(\hat{\varepsilon}, \delta)$-PP if for all $\theta \in \Theta$, all $(s_i, s_j) \in Q$ such that $P_\theta^S(s_i) > 0$ and $P_\theta^S(s_j) > 0$, and all measurable output events $\mathcal{T} \subseteq \mathcal{Y}$,

$$\Pr\big[\mathcal{M}(X) \in \mathcal{T} \mid S = s_i, \theta\big] \leq e^{\hat{\varepsilon}} \Pr\big[\mathcal{M}(X) \in \mathcal{T} \mid S = s_j, \theta\big] + \delta,$$

where $\hat{\varepsilon} \geq 0$ and $\delta \in [0, 1]$. This condition is mathematically equivalent to bounding the hockey-stick divergence. When $\delta = 0$, the mechanism satisfies pure $\hat{\varepsilon}$-PP, and the underlying quantification measure simplifies to the max divergence.

2.3 Wasserstein Mechanisms

Wasserstein mechanisms form a general class of privatization mechanisms for PP [38] and RPP [34]. In the three-layer perspective above, they belong to the mechanism layer: they do not change the protected secrets or the divergence used to quantify privacy, but provide a way to calibrate the additive noise to a Wasserstein-distance-based analogue of DP sensitivity, which we refer to as the *Wasserstein sensitivity*. We first recall the optimal transport (OT) notions: couplings and the ∞-Wasserstein distance.

Definition 2.2 (Coupling)**.** *Let μ and ν be probability measures on $(\mathbb{R}^d, \mathcal{B}(\mathbb{R}^d))$. A coupling of (μ,ν) is any probability measure π on $(\mathbb{R}^d \times \mathbb{R}^d, \mathcal{B}(\mathbb{R}^d) \otimes \mathcal{B}(\mathbb{R}^d))$ whose marginals are μ and ν. We denote the set of all couplings by $\Pi(\mu,\nu)$.*

Definition 2.3 (∞-Wasserstein Distance)**.** *Fix a norm $\|\cdot\|$ on $\mathbb{R}^d$. For probability measures μ and ν on $\mathbb{R}^d$, the ∞-Wasserstein distance (∞-WD) is $W_\infty(\mu,\nu) = \inf_{\pi\in\Pi(\mu,\nu)} \sup_{(x,y)\in\mathrm{supp}(\pi)} \|x-y\|$.*

For a *query function* $f : \mathcal{X} \to \mathbb{R}^d$, the *$\infty$-Wasserstein sensitivity* is then defined as

$$\Delta_\infty := \max_{(s_i,s_j)\in Q,\ \theta\in\Theta} W_\infty\big(\Pr[f(X) \mid s_i,\theta], \Pr[f(X) \mid s_j,\theta]\big). \quad (2)$$

Let ζ be the noise distribution. Pierquin et al. [34] provide the General Wasserstein Mechanism (GWM), a general formulation describing how Δ_∞, ζ, and α together determine an upper bound on the Rényi divergence of $\mathcal{M}(X) = f(X) + N$, with $N \sim \zeta$, based on the *shift reduction lemma* [14]. When $\zeta = \mathcal{N}\left(0, \frac{\alpha\Delta_\infty^2}{2\varepsilon} I_d\right)$, with Δ_∞ computed with respect to the ℓ_2 norm, $\mathcal{M}(X)$ satisfies (α,ε)-RPP. The main limitation of Wasserstein mechanisms is the scalability bottleneck of optimal transport (OT), which is exacerbated in high dimensions. In particular, computing W_∞, or more generally W_p (see Definition A.2 in App. A.3) between empirical distributions requires solving large-scale OT problems whose complexity grows superlinearly with the sample size. This motivates replacing Δ_∞ by a sliced Wasserstein sensitivity, introduced in Sec. 3.

2.4 SRPP: Overview

We introduce Sliced Rényi Pufferfish Privacy (SRPP). To clarify its status relative to existing frameworks, SRPP retains the broad Pufferfish semantics shared by classic PP and RPP, while replacing the full-dimensional Rényi divergence used by RPP with *Sliced Rényi Divergence* (SRD). We develop SRD under two variants: Average SRD (Definition 4.1) and Joint SRD (Definition 4.3). Under the three-layer perspective above, DP/RDP differ from Pufferfish-family notions primarily in semantic scope: DP/RDP protect record-level neighboring-dataset secrets, whereas PP/RPP/SRPP protect the secret pairs specified by a Pufferfish scenario. Within the Pufferfish semantics, PP, RPP, and SRPP notions differ in the quantification measure used to bound distinguishability.

To establish its status as a formal privacy notion, SRPP should be understood as a change in the quantification layer, while keeping the semantic privacy goal of PP/RPP intact. Semantically, because the dataset X is the input to a query or training algorithm, leakage about a protected secret is reflected in the difference between the secret-conditioned output distributions, under the admissible priors in Θ. Bounding this distinguishability is a meaningful privacy principle: if the output distributions for two protected secrets are hard to distinguish, then an adversary observing the release has limited ability to infer which secret holds. This shared semantic objective unites PP, RPP, and SRPP; they differ in how the output-distribution distinguishability is quantified.

To quantify this distinguishability, SRPP uses SRD. The use of sliced quantification is motivated by the sliced Wasserstein distance (SWD) literature: SWD compares high-dimensional distributions by aggregating discrepancies of 1-D projections. Analogously, SRD evaluates Rényi divergence after 1-D projections and aggregates these per-slice divergences according to a slice profile. Thus, SRPP is not an approximation of RPP obtained by estimating the full-dimensional Rényi divergence. Rather, SRPP is a distinct Pufferfish-family guarantee: it keeps the Pufferfish semantics but uses SRD as its quantification measure. The slice profile is part of this quantification measure, not an assumption that the adversary only observes projected outputs.

The primary motivation for introducing SRPP is operational tractability. As noted in Sec. 2.3, existing Wasserstein mechanisms are calibrated through the full-dimensional ∞-Wasserstein sensitivity Δ_∞, or related p-Wasserstein quantities [34]. Computing these quantities requires large-scale OT computations and suffers from the curse of dimensionality, making direct calibration difficult for high-dimensional functions. By moving to a sliced quantification measure, SRPP aligns the privacy definition with *sliced Wasserstein sensitivity* (Sec. 3). This replaces full-dimensional OT sensitivity computation by 1-D projected sensitivity computations. The change from RPP to SRPP is therefore a change in the quantification layer and the resulting calibration formula, not a change in the basic perturbation format: the mechanism may still release $f(X)+N$ with full-dimensional Gaussian noise. For iterative learning, we develop SRPP-SGD with gradient clipping and HUC/sa-HUC accountants, providing a DP-SGD-style route to tractable per-iteration calibration and additive accounting under Pufferfish semantics.

Threat Model. We consider a standard trusted-curator setting: the data holder runs a randomized mechanism $\mathcal{M}$ on the dataset and releases its output (e.g., a query answer or trained model). The adversary observes the release and may know both $\mathcal{M}$ and the Pufferfish scenario $(\mathcal{S}, Q, \Theta)$. Side information about the data-generating process or secret–data relationships is modeled through the admissible prior set Θ.

Privacy is required for all $(s_i, s_j) \in Q$ uniformly over $\theta \in \Theta$. In the SGD setting, the adversary observes the trained model or updates released by our algorithm, but not internal randomness beyond what the release reveals. The guarantee is therefore relative to the specified Pufferfish scenario: if Θ omits plausible secret-data relationships, the guarantee need not cover those omitted relationships. In this paper, we take $(\mathcal{S}, Q, \Theta)$ as the intrinsic modeling setting and focus on the resulting mechanistic and computational challenges of privatization mechanism design.

2.5 Examples

We give two examples to illustrate the semantic coverage of Pufferfish-family guarantees. In both cases, the protected fact is not simply the inclusion or value of one independent record, but a secret whose truth changes the distribution of the data through temporal correlation or aggregate structure.

Example 1: Temporal correlation in time-series [38]. Consider releasing an activity histogram over a long time series $X = \{X_1, \ldots, X_T\}$ while protecting the activity at a specific time t ($X_t = a \in \mathcal{A}$). A Pufferfish scenario can take $\mathcal{S} = \{s_a^t : s_a^t = \{X_t = a\}, a \in \mathcal{A}, t \in [T]\}$, $Q = \{(s_a^t, s_b^t) : a \neq b, t \in [T]\}$, and Θ as a class of time-homogeneous Markov chains capturing temporal correlations in human behavior [38]. Here, the probabilistic secret–dataset relationship is explicit: conditioning on the secret $s_a^t = \{X_t = a\}$ changes the conditional distribution of nearby time points through the Markov transition model in Θ. This protects activity-at-time-t secrets under the models in Θ. In this setting, standard record-level DP does not directly apply because the time series belongs to a single subject; record-level DP does not erase evidence of activities that persist across correlated time points, whereas the group-privacy baseline used in [38] adds noise that scales with T and can destroy utility. Pufferfish-family notions instead use Θ to calibrate noise to the modeled secret-induced influence across the time series.

Example 2: Macro-level attribute privacy [49]. Consider releasing aggregate statistics, such as average height and weight, over a clinical cohort while protecting a dataset-level attribute, such as the cohort's gender distribution. A Pufferfish scenario can specify aggregate secrets such as $s_1 = \{\text{fraction of female patients} \geq 0.7\}$ and $s_2 = \{\text{fraction of female patients} \leq 0.3\}$, include (s_1, s_2) in Q, and let Θ model correlations among biometric features and the protected aggregate attribute, e.g., via a Gaussian model [49]. This protects a macro-level property of the dataset. Record-level DP does not directly target this goal, whereas Pufferfish-family notions calibrate noise to reduce distinguishability between the specified aggregate secrets under Θ.

These examples illustrate that the Pufferfish family is useful when the protected fact is not merely the inclusion or value of one independent record, but a secret for which conditioning on different secret values induces different data distributions under the specified scenario. In such settings, the scenario $(\mathcal{S}, Q, \Theta)$ explicitly specifies the protected secrets, the alternatives to be made indistinguishable, and the admissible secret–data relationships. DP can be recovered as a special record-level case under independence assumptions, but Pufferfish-type notions allow the privacy statement to target temporally correlated secrets or dataset-level aggregate attributes directly.

What is not directly certified. The coverage of any Pufferfish-family guarantee is limited to the specified scenario $(\mathcal{S}, Q, \Theta)$. For example, an SRPP guarantee targeting a global demographic attribute does not automatically protect unrelated temporal patterns or individual-level secrets unless those secrets and relationships are included in the scenario. Thus, the scenario must be chosen to match the real-world privacy risks that the deployment intends to protect.

3 Sliced Wasserstein Sensitivity

In this section, we introduce the *Sliced Wasserstein sensitivity (SW-sensitivity)*, obtained by slicing the ∞-WD, to mitigate the computational bottlenecks of Wasserstein mechanisms in high dimensions.

We first introduce notation. Let $f : \mathcal{X} \to \mathbb{R}^d$ be a *query* function. For each $\theta \in \Theta$ and $s \in \mathcal{S}$, let $P_\theta^{f,s}$ denote the conditional distribution of the (non-perturbed) query output $f(X)$ given $S = s$, i.e., $P_\theta^{f,s} := P_\theta(f(X) \mid s)$. For a measurable map $g : \mathcal{W} \to \mathcal{C}$ and a Borel measure μ on $\mathcal{W}$, the *pushforward* $g_\# \mu$ on $\mathcal{C}$ is defined by

$$g_\#\mu(A) \;=\; \mu\big(g^{-1}(A)\big), \qquad \forall A \in \mathcal{B}(\mathcal{C}), \tag{3}$$

where $\mathcal{B}(\mathcal{C})$ denotes the Borel σ-algebra on $\mathcal{C}$. Let $\mathbb{S}^{d-1} = \{u \in \mathbb{R}^d : \|u\|_2 = 1\}$ be the unit sphere and let ω be a probability measure on $\mathbb{S}^{d-1}$. For each *direction* $u \in \mathbb{S}^{d-1}$, define the projection $\Psi^u : \mathbb{R}^d \to \mathbb{R}$ by $\Psi^u(a) = \langle a, u \rangle$. Then, for any Borel probability measure P on $\mathbb{R}^d$, the pushforward $\Psi^u_\# \mathrm{P}$ is the 1-D distribution of $\langle \hat{X}, u \rangle$ when $\hat{X} \sim \mathrm{P}$. For a direction u, we refer to the induced 1-D projection Ψ^u as a *slice*.

Next, we define the *sliced Wasserstein distance*, which defines a metric for comparing probability measures on $\mathbb{R}^d$ by integrating WD over all 1-D projections.

Definition 3.1 (Sliced Wasserstein Distance (SWD) [35])**.** *Let $\mathcal{P}_p(\mathbb{R}^d)$ denote the set of all Borel probability measures on $\mathbb{R}^d$ with finite p-th moment, i.e., $\int_{\mathbb{R}^d} \|x\|^p \, \mathrm{d}\mu(x) < \infty$ for all $\mu \in \mathcal{P}_p(\mathbb{R}^d)$. For $p \geq 1$ and $\mu, \nu \in \mathcal{P}_p(\mathbb{R}^d)$, the p-SWD is defined by*

$$\mathrm{SW}_p^p(\mu, \nu) \;:=\; \int_{\mathbb{S}^{d-1}} W_p^p(\Psi^u_\#\mu,\, \Psi^u_\#\nu)\, \mathrm{d}\lambda(u), \tag{4}$$

where λ is the uniform probability measure on $\mathbb{S}^{d-1}$.

The SWD is appealing because it replaces a high-dimensional OT comparison by an expectation over 1-D projections, where W_p admits a closed form (including $p = \infty$) in

terms of quantile functions. By the Cramér-Wold theorem [6], a probability measure on $\mathbb{R}^d$ is uniquely determined by the totality of its 1-D projections across all directions $u \in \mathbb{S}^{d-1}$. Consequently, the SWD is a proper metric.

In practice, SW_p is typically estimated by Monte Carlo sampling of directions and sorting the projected samples, reducing the computation to projections and 1-D Wasserstein evaluations. This yields *finite* slicing (or empirical slice distribution) and thus an approximate SWD estimator (See Sec. 4.3.2 for formal finite-sample analyses). The resulting finite-slice quantity remains a pseudometric (it is nonnegative, symmetric, and satisfies the triangle inequality).

In one dimension, $W_p(\hat{\mu},\hat{\nu})$ admits the representation [33, Remark 2.30]:

$$W_p^p(\hat{\mu},\hat{\nu}) = \int_0^1 \left|\mathsf{F}_{\hat{\mu}}^{-1}(u) - \mathsf{F}_{\hat{\nu}}^{-1}(u)\right|^p \mathrm{d}u, \tag{5}$$

where $\mathsf{F}_{\hat{\mu}}^{-1}$ and $\mathsf{F}_{\hat{\nu}}^{-1}$ denote the quantile functions of $\hat{\mu}$ and $\hat{\nu}$. The map $\hat{\mu} \mapsto \mathsf{F}_{\hat{\mu}}^{-1}$ (the same for $\hat{\nu} \mapsto \mathsf{F}_{\hat{\nu}}^{-1}$) gives an isometric embedding of $(\mathcal{P}_p(\mathbb{R}), W_p)$ into $L_p([0,1])$. For $p=2$ the metric is induced by the L_2 inner product on this embedded subset, yielding a simple, essentially linear geometry compared to the much richer geometry in higher dimensions.

SW_∞ can be viewed as the $p \to \infty$ endpoint of the sliced Wasserstein family SW_p.[1] At the slice level, SW_∞ provides a natural analogue of the W_∞ (Definition 2.3) for sensitivity analysis in PP/RPP: it captures the largest possible shift in the projected query output induced by changing the secret.

Given a Pufferfish scenario $(\mathcal{S}, Q, \Theta)$, for each direction $u \in \mathbb{S}^{d-1}$, we define the *per-slice ∞-Wasserstein sensitivity (SW-sensitivity)* as

$$\Delta_\infty^u := \max_{(s_i,s_j)\in Q,\ \theta\in\Theta} W_\infty\Big(\Psi_\#^u P_\theta^{f,s_i},\ \Psi_\#^u P_\theta^{f,s_j}\Big), \tag{6}$$

which is the worst-case ∞-WD between the 1-D projections of $P_\theta^{s_i}$ and $P_\theta^{s_j}$ along direction u. Equivalently, it quantifies the maximal shift in the projected query output when the secret changes from s_i to s_j under any $\theta \in \Theta$.

Conceptually, we might hope to replace the direction-dependent sensitivity Δ_∞^u in (6) by a single scalar derived from SW_p. However, SW_p is an *average* over directions of 1-D transport costs and therefore does not provide a uniform, worst-case bound across all $u \in \mathbb{S}^{d-1}$. In contrast, our per-slice Rényi envelope (9) (Sec. 4.3) requires controlling each slice, and within each slice taking the worst case over admissible secret pairs and priors. In particular, an average can be small even when a subset of directions has large sensitivity.

Moreover, an adversary observes the d-dimensional query output, not a 1-D projection. Thus, calibrating noise using a sliced quantity (e.g., an average of Δ_∞^u over u) while retaining the original RPP definition introduces a mismatch between the geometry used for calibration and the divergence geometry in the privacy definition, and can therefore lead to unsound privacy bounds. In particular, for any $u \in \mathbb{S}^{d-1}$, the projection $\Psi^u(x) = \langle x, u\rangle$ is 1-Lipschitz with respect to the Euclidean norm, i.e., $|\Psi^u(x) - \Psi^u(y)| \le \|x-y\|_2$. Hence, for any probability measures μ,ν on $\mathbb{R}^d$, $W_\infty\big(\Psi_\#^u\mu,\ \Psi_\#^u\nu\big) \le W_\infty(\mu,\nu)$, where W_∞ on $\mathbb{R}^d$ is computed with the ℓ_2 norm. Indeed, for any coupling $\gamma \in \Pi(\mu,\nu)$ we have $|\Psi^u(x) - \Psi^u(y)| \le \|x-y\|_2$ γ-almost surely. Taking the essential supremum over $(x,y) \in \mathrm{supp}(\gamma)$ and then the infimum over γ yields the inequality. Applying this bound to $\mu = P_\theta^{s_i}$ and $\nu = P_\theta^{s_j}$, and then taking the maximum over $(s_i,s_j) \in Q$ and $\theta \in \Theta$, gives $\Delta_\infty^u \le \Delta_\infty$, where Δ_∞ is defined in (2).

This raises a natural question: which privacy notion is compatible with sliced Wasserstein sensitivity? We answer it by defining *Sliced Rényi Pufferfish Privacy* in the next section.

4 Sliced Rényi Pufferfish Privacy

In this section, we introduce *Sliced Rényi Pufferfish Privacy* (SRPP), a directional refinement of RPP defined relative to a *slice profile* $(\mathcal{U},\omega)$, where $\mathcal{U} \subseteq \mathbb{S}^{d-1}$ is a set of directions and ω is a probability measure on $\mathbb{S}^{d-1}$ supported on $\mathcal{U}$. In the finite case, $\mathcal{U} = u_1,\dots,u_m$ and ω is discrete with weights $(\omega_1,\dots,\omega_m)$, so that $\int_{\mathbb{S}^{d-1}} f(u)\,\mathrm{d}\omega(u) = \sum_{i=1}^m \omega_i f(u_i)$.

We define two variants: Average SRPP (Ave-SRPP; Sec. 4.1) and Joint SRPP (Joint-SRPP; Sec. 4.2). Ave-SRPP bounds the ω-average of per-slice Rényi divergences. Joint-SRPP aggregates per-slice divergences through a joint log-moment bound, making it more sensitive to rare high-risk directions. Sec. 4.3 formally introduce *Sliced Wasserstein Mechanisms* using SW-sensitivity defined by Sec. 3. We defer additional discussions of SRPP, including the properties such as post-processing immunity and the conversion to *sliced PP*, and the difference between Ave- and Joint-SRPP in App. B.

4.1 Average SRPP

Our Ave-SRPP aggregates per-slice Rényi divergences by averaging, where the quantification measure is *Average Sliced Rényi Divergence.*

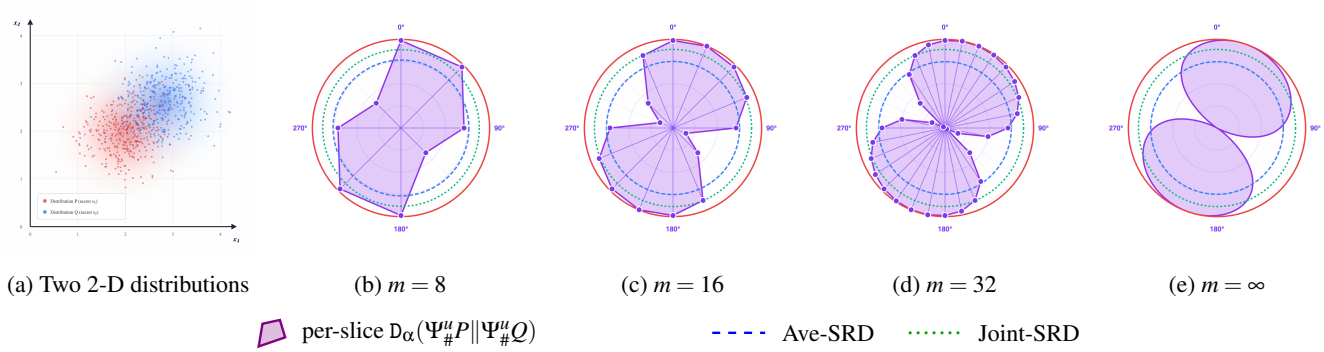
Definition 4.1 (Average Sliced Rényi Divergence (Ave-SRD))**.** *For two probability distributions* P *and* Q *on* $\mathbb{R}^d$*, and a slice profile* $(\mathcal{U},\omega)$*, the Ave-SRD of order* $\alpha > 1$ *with respect to* ω *is*

$$\mathtt{AveSD}_\alpha^\omega(\mathrm{P} \parallel \mathrm{Q}) := \int_{\mathbb{S}^{d-1}} \mathtt{D}_\alpha(\Psi_\#^u\mathrm{P} \parallel \Psi_\#^u\mathrm{Q})\,\mathrm{d}\omega(u), \tag{7}$$

where $\mathtt{D}_\alpha$ *is the standard Rényi divergence.*

Ave-SRD averages the 1-D Rényi divergences between the projected measures $\Psi_\#^u\mathrm{P}$ and $\Psi_\#^u\mathrm{Q}$ over directions $u \sim \omega$. When $d=1$, $\mathtt{AveSD}_\alpha^\omega$ reduces to the standard $\mathtt{D}_\alpha(\mathrm{P}\|\mathrm{Q})$.

[1] Formally, this corresponds to taking the $p \to \infty$ limit of the per-slice L_p aggregation of 1-D Wasserstein distances.



(a) Two 2-D distributions (b) $m = 8$ (c) $m = 16$ (d) $m = 32$ (e) $m = \infty$

per-slice $\mathtt{D}_\alpha(\Psi^u_\# P \| \Psi^u_\# Q)$ Ave-SRD Joint-SRD

Figure 1: SRD profiles in $\mathbb{R}^2$ at order $\alpha = 4$ under a uniform slice distribution ω. (a) Two 2-D Gaussian distributions P (red) and Q (blue). (b)–(d) Discrete slice profiles with $m \in \{8, 16, 32\}$ directions: the filled purple polygon traces the per-direction divergences $\mathtt{D}_\alpha(\Psi^u_\# P \| \Psi^u_\# Q)$. The blue dashed circle is Ave-SRD (directional mean), and the green dotted circle is Joint-SRD (log-moment aggregation, emphasizing large divergences). (e) The continuous-direction limit $m = \infty$.

Definition 4.2 ($(\alpha, \varepsilon, \omega)$-Ave-SRPP)**.** *Let $\alpha > 1$ and $\varepsilon \geq 0$. Given a slice profile $(\mathcal{U}, \omega)$, a mechanism $\mathcal{M}$ satisfies $(\alpha, \varepsilon, \omega)$-Ave-SRPP in $(\mathcal{S}, Q, \Theta)$ if for all $\theta \in \Theta$ and all $(s_i, s_j) \in Q$ with $P^S_\theta(s_i), P^S_\theta(s_j) > 0$,*

$$\mathtt{AveSD}^\omega_\alpha\big(\Pr[\mathcal{M}(X) \mid s_i, \theta] \,\big\|\, \Pr[\mathcal{M}(X) \mid s_j, \theta]\big) \leq \varepsilon.$$

Thus, Ave-SRPP bounds the ω-average of per-slice Rényi divergences between the output distributions for any admissible secret pair (uniformly over $\theta \in \Theta$).

4.2 Joint SRPP

Joint-SRPP aggregates per-slice divergences through a joint log-moment (Rényi) bound. Its quantification measure is the *Joint Sliced Rényi Divergence*.

Definition 4.3 (Joint Sliced Rényi Divergence (Joint-SRD))**.** *For two probability distributions* P *and* Q *on $\mathbb{R}^d$ and a slice profile $(\mathcal{U}, \omega)$, the Joint-SRD of order $\alpha > 1$ is*

$$\mathtt{JSD}^\omega_\alpha(\mathrm{P} \,\|\, \mathrm{Q}) := \tfrac{1}{\alpha - 1} \log \int_{\mathbb{S}^{d-1}} \exp\Big((\alpha - 1)\mathtt{D}_\alpha(\Psi^u_\# \mathrm{P} \,\|\, \Psi^u_\# \mathrm{Q})\Big)\, \mathrm{d}\omega(u). \tag{8}$$

Joint-SRD aggregates per-slice divergences via a log-moment transform, making it more sensitive than Ave-SRD to rare directions with large 1-D divergence (see Figure 1). When $d = 1$, $\mathtt{JSD}^\omega_\alpha(\mathrm{P} \| \mathrm{Q})$ reduces to the standard $\mathtt{D}_\alpha(\mathrm{P} \| \mathrm{Q})$.

Definition 4.4 ($(\alpha, \varepsilon, \omega)$-Joint-SRPP)**.** *Let $\alpha > 1$ and $\varepsilon \geq 0$. Given a slice profile $(\mathcal{U}, \omega)$, a mechanism $\mathcal{M}$ satisfies $(\alpha, \varepsilon, \omega)$-Joint-SRPP in $(\mathcal{S}, Q, \Theta)$ if for all $\theta \in \Theta$ and all $(s_i, s_j) \in Q$ with $P^S_\theta(s_i), P^S_\theta(s_j) > 0$,*

$$\mathtt{JSD}^\omega_\alpha\big(\Pr(\mathcal{M}(X) \mid s_i, \theta) \,\big\|\, \Pr(\mathcal{M}(X) \mid s_j, \theta)\big) \leq \varepsilon.$$

4.3 Sliced Wasserstein Mechanism

In this section, we present the *Sliced Wasserstein Mechanism (SWM)* to obtain SRPP guarantee by leveraging the computational tractability afforded by SW-sensitivity of Sec. 3.

Let $N \sim \zeta$ denote additive noise with distribution ζ on $\mathbb{R}^d$. For $a \in \mathbb{R}^d$, write ζ_{-a} for the shifted distribution (i.e., the distribution of $U - a$ for $U \sim \zeta$), where $\zeta_{-a}(w) = \zeta(w - a)$. For any $\pi : \mathbb{S}^{d-1} \to \mathbb{R}_+$ and direction $u \in \mathbb{S}^{d-1}$, we define the *per-slice Rényi envelope* in as

$$R_\alpha(\zeta, \pi(u)) = \sup_{\|a\| < \pi(u)} \mathtt{D}_\alpha\big(\zeta_{-a}, \zeta\big). \tag{9}$$

Recall $P^{f,s}_\theta := P_\theta(f(X) \mid s)$ for a query function f. When $\mathcal{M}(X) = f(X) + N$ with $N \sim \zeta$ independent of $f(X)$, let $\mathcal{M}^\theta_s = P^{f,s}_\theta * \zeta$ be the distribution of $\mathcal{M}(X)$ where $*$ is the convolution operator.

Lemma 4.1. *Fix a Pufferfish scenario $(\mathcal{S}, Q, \Theta)$ and a slice profile $(\mathcal{U}, \omega)$. Let $\mathcal{M}(X) = f(X) + N$ with $N \sim \zeta$ independent of X. For any $\theta \in \Theta$ and any $(s_i, s_j) \in Q$ with $P^S_\theta(s_i) > 0$ and $P^S_\theta(s_j) > 0$, we have for all $\alpha > 1$ and all $u \in \mathcal{U}$,*

$$\mathtt{D}_\alpha\Big(\Psi^u_\# \mathcal{M}^\theta_{s_i} \,\Big\|\, \Psi^u_\# \mathcal{M}^\theta_{s_j}\Big) \;\leq\; R_\alpha\big(\zeta, z^u_\infty(\theta, s_i, s_j)\big),$$

where $z^u_\infty(\theta, s_i, s_j) := W_\infty\Big(\Psi^u_\# P^{f,s_i}_\theta,\ \Psi^u_\# P^{f,s_j}_\theta\Big)$.

Lemma 4.1 states that, along each direction u, the per-slice Rényi envelope evaluated at the shift radius z^u_∞ upper-bounds the actual Rényi divergence between the projected output distributions under secrets s_i and s_j.

We then define the *averaged Rényi envelope* as

$$\mathtt{AR}^\infty_{\alpha,\omega}\big(\mathcal{M}^\theta_{s_i} \,\big\|\, \mathcal{M}^\theta_{s_j}\big) := \int_{\mathbb{S}^{d-1}} R_\alpha\big(\zeta, z^u_\infty\big)\, \mathrm{d}\omega(u). \tag{10}$$

This quantity aggregates the per-slice Rényi envelope bounds over slice $u \sim \omega$ and will serve as our sliced Rényi envelope

surrogate for the true averaged sliced Rényi divergence. Similarly, we define the *joint sliced Rényi envelope* by

$$\begin{aligned}&\mathtt{JR}^{\infty}_{\alpha,\omega}\big(\mathcal{M}^{\theta}_{s_i}\big\|\mathcal{M}^{\theta}_{s_j}\big)\\&:=\tfrac{1}{\alpha-1}\log\left(\int_{\mathbb{S}^{d-1}}\exp\big((\alpha-1)R_\alpha(\zeta,z^u_\infty)\big)d\omega(u)\right).\end{aligned}\tag{11}$$

Here, instead of averaging the per-slice envelope values linearly, we aggregate them via an exponential moment over $V\sim\omega$, followed by a log and $1/(\alpha-1)$ rescaling. This construction mirrors the passage from an average of Rényi divergences to a joint Rényi divergence for the pair $(V,\Psi^V(\mathcal{M}(X)))$. As a result, $\mathtt{JR}^{\infty}_{\alpha,\omega}$ is more sensitive to rare directions with large envelope values than $\mathtt{AR}^{\infty}_{\alpha,\omega}$.

Corollary 4.1. *Fix a continuous or finite slice profile $\{\mathcal{U},\omega\}$. Let $\alpha>1$ and $\varepsilon\geq 0$. For a query function f, let $\mathcal{M}(X)=f(X)+N$ with $N\sim\zeta$. Then, (i) $\mathcal{M}(X)$ satisfies $(\alpha,\varepsilon,\omega)$-Ave-SRPP if $\mathtt{AR}^{\infty}_{\alpha,\omega}\big(\mathcal{M}^{\theta}_{s_i}\,\big\|\,\mathcal{M}^{\theta}_{s_j}\big)\leq\varepsilon$; and (ii) $\mathcal{M}(X)$ satisfies $(\alpha,\varepsilon,\omega)$-Joint-SRPP if $\mathtt{JR}^{\infty}_{\alpha,\omega}\big(\mathcal{M}^{\theta}_{s_i}\,\big\|\,\mathcal{M}^{\theta}_{s_j}\big)\leq\varepsilon$.*

Corollary 4.1 follows immediately from Lemma 4.1. In particular, an ε-bounded Rényi envelope under (α,ω) is sufficient to guarantee $(\alpha,\varepsilon,\omega)$-SRPP, for both the average and joint variants. The envelopes $\mathtt{AR}^{\infty}_{\alpha,\omega}$ and $\mathtt{JR}^{\infty}_{\alpha,\omega}$ are especially useful operational targets: our Sliced Wasserstein Mechanism (SWM), calibrated via SW-sensitivity, provides a direct and tractable way to enforce $\mathtt{AR}^{\infty}_{\alpha,\omega}\big(\mathcal{M}^{\theta}_{s_i}\,\|\,\mathcal{M}^{\theta}_{s_j}\big)\leq\varepsilon$ and $\mathtt{JR}^{\infty}_{\alpha,\omega}\big(\mathcal{M}^{\theta}_{s_i}\,\|\,\mathcal{M}^{\theta}_{s_j}\big)\leq\varepsilon$ as we detail next.

4.3.1 Gaussian Sliced Wasserstein Mechanism

We specialize to Gaussian noise, $N\sim\mathcal{N}(0,\sigma^2 I_d)$. Given a slice profile $(\mathcal{U},\omega)$ (continuous or finite) and the per-slice SW-sensitivity Δ^u_∞ in (6), define

$$\overline{\Delta}^2:=\int_{\mathbb{S}^{d-1}}\big(\Delta^u_\infty\big)^2\,\mathrm{d}\omega(u),\qquad \Delta^2_\star:=\sup_{u\in\mathcal{U}}\big(\Delta^u_\infty\big)^2.$$

Here, $\overline{\Delta}^2$ is the second moment of SW-sensitivity and $\Delta^2_\star$ is the largest squared directional sensitivity over $u\in\mathcal{U}$.

Theorem 4.5. *Fix a continuous or finite slice profile $\{\mathcal{U},\omega\}$. Let $\alpha>1$ and $\varepsilon\geq 0$. A mechanism $\mathcal{M}(X)=f(X)+N$ with $N\sim\zeta$ satisfies $(\alpha,\varepsilon,\omega)$-Ave-SRPP in $(\mathcal{S},\mathcal{Q},\Theta)$ if $\zeta=\mathcal{N}(0,\frac{\alpha\overline{\Delta}^2}{2\varepsilon}I_d)$.*

Theorem 4.5 provides a closed-form Gaussian noise calibration for Ave-SRPP under a fixed slice profile. Specifically, setting the isotropic variance to $\sigma^2=\frac{\alpha\overline{\Delta}^2}{2\varepsilon}$ ensures that the averaged Rényi envelope $\mathtt{AR}^{\infty}_{\alpha,\omega}$ is bounded by ε uniformly over $\theta\in\Theta$ and $(s_i,s_j)\in\mathcal{Q}$. By Corollary 4.1, this envelop bound is sufficient to guarantee $(\alpha,\varepsilon,\omega)$-Ave-SRPP. Moreover, the per-slice Rényi envelope is attained by the Gaussian SWM.

Theorem 4.6. *Fix a continuous or finite slice profile $\{\mathcal{U},\omega\}$. Let $\alpha>1$ and $\varepsilon\geq 0$. A mechanism $\mathcal{M}(X)=f(X)+N$ with $N\sim\zeta$ satisfies $(\alpha,\varepsilon,\omega)$-Joint-SRPP in $(\mathcal{S},\mathcal{Q},\Theta)$ if $\zeta=\mathcal{N}(0,\frac{\alpha\Delta^2_\star}{2\varepsilon}I_d)$.*

Theorem 4.6 gives the analogous closed-form noise calibration for Joint-SRPP. Taking $\sigma^2=\frac{\alpha\Delta^2_\star}{2\varepsilon}$ ensures that the joint Rényi envelope $\mathtt{JR}^{\infty}_{\alpha,\omega}$ is at most ε uniformly over $\theta\in\Theta$ and $(s_i,s_j)\in\mathcal{Q}$, and hence $\mathcal{M}$ satisfies $(\alpha,\varepsilon,\omega)$-Joint-SRPP by Corollary 4.1. Compared to the Gaussian SWM for the Ave-SRPP, this choice depends on the worst directional sensitivity $\Delta^2_\star$, rather than the profile average $\overline{\Delta}^2$, making it more conservative and typically requiring larger noise to protect against rare high-sensitivity directions.

4.3.2 Computational Properties of SWM

The SRPP guarantees of the SWM in Theorems 4.5 and 4.6 hold for both *continuous* slice profiles ω on $\mathbb{S}^{d-1}$ and *finite* profiles. In practice, even when ω is conceptually continuous, SWM is instantiated by sampling finitely many directions and the conditional distributions $P^{f,s}_\theta$ are typically accessed only through samples. This induces (i) a *finite-slice* (Monte Carlo) approximation of the profile aggregates and (ii) a *finite-sample* approximation of the per-slice sensitivities Δ^u_∞.

Notations. Fix a query function $f:\mathcal{X}\to\mathbb{R}^d$ and a Pufferfish scenario $(\mathcal{S},\mathcal{Q},\Theta)$. Fix finite instantiated families $\widehat{\Theta}\subseteq\Theta$ and $\widehat{\mathcal{Q}}\subseteq\mathcal{Q}$. For each $(s,\theta)\in\mathcal{S}\times\widehat{\Theta}$, draw n i.i.d. samples $Z^{(\theta,s)}_{1:n}\sim P^{f,s}_\theta$. For a direction u, write $V^{\theta,s}_k(u):=\langle Z^{(\theta,s)}_k,u\rangle$, and let $\widehat{F}^u_{\theta,s}$ denote the empirical CDF of $\{V^{\theta,s}_k(u)\}_{k=1}^n$, with generalized inverse $(\widehat{F})^{-1}(t):=\inf\{x:\widehat{F}(x)\geq t\}$. Let ω be a continuous slice profile on $\mathbb{S}^{d-1}$ and draw i.i.d. directions $u_1,\dots,u_m\sim\omega$. Define the empirical profile $\omega_m:=\frac{1}{m}\sum_{\ell=1}^m\delta_{u_\ell}$ and the profile aggregates $\overline{\Delta}^2(\omega):=\mathbb{E}_{U\sim\omega}\big[(\Delta^U_\infty)^2\big]$ and $\overline{\Delta}^2(\omega_m)=\frac{1}{m}\sum_{\ell=1}^m(\Delta^{u_\ell}_\infty)^2$.

For $\widetilde{\rho}\in(0,1)$, define $\mathsf{e}_n(\widetilde{\rho}):=\sqrt{\frac{\log(2/\widetilde{\rho})}{2n}}$ [2]. For $u\in\mathbb{S}^{d-1}$ and $(\theta,(s_i,s_j))\in\widehat{\Theta}\times\widehat{\mathcal{Q}}$, define

$$\begin{aligned}\widehat{W}^{\theta,s_i,s_j}_\infty(u,\widetilde{\rho}):=\sup_{t\in[\mathsf{e}_n(\widetilde{\rho}/2),\,1-\mathsf{e}_n(\widetilde{\rho}/2)]}\Big|&\big(\widehat{F}^u_{\theta,s_i}\big)^{-1}\big(t+\mathsf{e}_n(\widetilde{\rho}/2)\big)\\&-\big(\widehat{F}^u_{\theta,s_j}\big)^{-1}\big(t-\mathsf{e}_n(\widetilde{\rho}/2)\big)\Big|.\end{aligned}$$

For each sampled direction u_ℓ, define $\widehat{\Delta}_\infty(u_\ell):=\max_{(\theta,(s_i,s_j))\in\widehat{\Theta}\times\widehat{\mathcal{Q}}}\widehat{W}^{\theta,s_i,s_j}_\infty\big(u_\ell,\widetilde{\rho}\big)$.

Theorems 4.7 and 4.8 show that SWM admits PAC-style calibrations under *both* Monte Carlo slicing and finite-sample estimation of $P^{f,s}_\theta$, yielding $(\alpha,\varepsilon,\omega)$-Ave/Joint-SRPP guarantee with probability at least $1-\gamma$.

[2] $\sup_x|\widehat{F}(x)-F(x)|\leq\mathsf{e}_n(\widetilde{\rho})$ holds with probability at least $1-\widetilde{\rho}$ due to the Dvoretzky–Kiefer–Wolfowitz inequality.

Theorem 4.7. *Fix a Pufferfish scenario $(\mathcal{S}, Q, \Theta)$, a query $f : \mathcal{X} \to \mathbb{R}^d$, and a continuous slice distribution ω on $\mathbb{S}^{d-1}$. Draw i.i.d. directions $U_{1:m} \sim \omega$. Assume $0 \le \Delta_\infty^u \le \Delta_0$ for all u. Fix $\alpha > 1$, $\varepsilon \ge 0$, and $\gamma \in (0,1)$. Let $\widehat{\Delta}_\infty(U_\ell)$ be random quantities satisfying $\Pr\left(\Delta_\infty^{U_\ell} \le \widehat{\Delta}_\infty(U_\ell) \text{ for all } \ell \in [m]\right) \ge 1-\gamma/2$, where the probability is over the conditional sampling used to form $\widehat{\Delta}_\infty(U_\ell)$. If Gaussian SWM uses $N \sim \mathcal{N}(0, \sigma^2 I_d)$ with*

$$\sigma^2 \;\ge\; \frac{\alpha}{2\varepsilon}\left(\frac{1}{m}\sum_{\ell=1}^{m}\left(\widehat{\Delta}_\infty(U_\ell)\right)^2 + \Delta_0^2\sqrt{\frac{\log(4/\gamma)}{2m}}\right),$$

then with probability at least $1-\gamma$ (over both the conditional sampling and the draw of $U_{1:m}$), $\mathcal{M}(X) = f(X) + N$ satisfies $(\alpha, \varepsilon, \omega)$-Ave-SRPP.

Theorem 4.8. *Fix a Pufferfish scenario $(\mathcal{S}, Q, \Theta)$, a query $f : \mathcal{X} \to \mathbb{R}^d$, and a continuous slice distribution ω on $\mathbb{S}^{d-1}$. Draw i.i.d. directions $U_{1:m} \sim \omega$. Assume $0 \le \Delta_\infty^u \le \Delta_0$ for all u. Fix $\alpha > 1$, $\varepsilon \ge 0$, and $\gamma \in (0,1)$. Let $\widehat{\Delta}_\infty(U_\ell)$ be random quantities satisfying $\Pr\left(\Delta_\infty^{U_\ell} \le \widehat{\Delta}_\infty(U_\ell) \text{ for all } \ell \in [m]\right) \ge 1-\gamma/2$, where the probability is over the conditional sampling used to form $\widehat{\Delta}_\infty(U_\ell)$. For $\sigma^2 > 0$, define $c(\sigma) := \frac{\alpha(\alpha-1)}{2\sigma^2}$, $\widehat{\mu}_m(\sigma) := \frac{1}{m}\sum_{\ell=1}^{m}\exp\left(c(\sigma)\left(\widehat{\Delta}_\infty(U_\ell)\right)^2\right)$, and $b(\sigma) := \exp\left(c(\sigma)\Delta_0^2\right)$. Gaussian SWM uses $N \sim \mathcal{N}(0, \sigma^2 I_d)$ with σ^2 satisfying*

$$\frac{1}{\alpha-1}\log\left(\widehat{\mu}_m(\sigma) + \left(b(\sigma)-1\right)\sqrt{\frac{\log(4/\gamma)}{2m}}\right) \le \varepsilon, \qquad (12)$$

then with probability at least $1-\gamma$ (over both the conditional sampling and the draw of $U_{1:m}$), $\mathcal{M}(X) = f(X) + \mathcal{N}(0, \sigma^2 I_d)$ satisfies $(\alpha, \varepsilon, \omega)$-Joint-SRPP.

Since $c(\sigma) = \frac{\alpha(\alpha-1)}{(2\sigma^2)}$ is decreasing in σ, both $\widehat{\mu}_m(\sigma)$ and $b(\sigma)$ are decreasing in σ, hence the left-hand side of (12) is decreasing. Therefore the Joint-SRPP calibration condition $\Phi(\sigma) \le \varepsilon$ can be enforced by a 1-D bisection search over σ.

Finite ω. If ω is discrete on $\mathcal{U} = \{u_1, \ldots, u_m\}$ with weights $\{\omega_\ell\}$ (no Monte Carlo directions), for Ave-SRPP, we can replace $\frac{1}{m}\sum_{\ell=1}^{m}(\cdot)$ by $\sum_{\ell=1}^{m}\omega_\ell(\cdot)$ and drop the Hoeffding term $\Delta_0^2\sqrt{\log(4/\gamma)/(2m)}$. For Joint-SRPP, the same statement applies with $U_\ell = u_\ell$. Thus, only the finite-sample (distribution-estimation) layer remains.

Proposition 4.1 (Computational Advantage of SMD, Informal)**.** *With n samples per conditional distribution $P_\theta^{f,s}$, query dimension d, $\mathsf{M} := |\widehat{\Theta}|\,|\widehat{Q}|$ instances, and m slice directions, the runtime of computing SW-sensitivity is $O\big(\mathsf{M}\,m\,(dn + n\log n)\big)$. When the unsliced WM is computed by the standard discrete OT (Kantorovich) formulation on empirical measures, the runtime is $O\big(\mathsf{M}\,(dn^2 + n^3\log n)\big)$.*

Proposition 4.1 shows that slicing converts d-dimensional OT into m 1-D computations, reducing runtime from quadratic/cubic in n to nearly linear (up to sorting) (see App. P for the formal version). Beyond SWD, another mainstream scalable OT approximation is entropic OT via the Sinkhorn algorithm [5, 7, 24, 40]. App. A.5 describes a Sinkhorn–Wasserstein mechanism for RPP and summarizes why it is less suitable for privacy calibration and privatization.

4.4 Discussion on Slice Profile

In the SRPP framework, the privacy notion is parameterized by a slice profile $(\mathcal{U}, \omega)$, where $\mathcal{U} \subseteq \mathbb{S}^{d-1}$ is a set of projection directions and ω is a probability measure over those directions. To clarify the status of the profile and address how finite slicing affects the privacy guarantee, it is necessary to separate the slice profile's role in the *definition of the privacy notion* from its role in the *implementation of the privatization mechanism*. Depending on the operational context, a finite set of directions used during SWM calibration can be interpreted through two distinct modes.

Mode-I: Exact Finite-Profile SRPP. In this mode, the target guarantee is SRPP with a specified finite slice profile $(\mathcal{U}, \omega)$. The mechanism, e.g., SWM, uses the same $(\mathcal{U}, \omega)$. Hence, the resulting guarantee is an exact SRPP guarantee for the discrete distribution ω over $\mathcal{U}$, not an approximation to another slice profile. In particular, there is no finite-slicing approximation error relative to this finite-profile SRPP guarantee. Because the specified slice profile is part of the SRPP quantification, changing the finite profile changes the finite-profile SRPP guarantee being stated: it changes how the sliced Rényi divergence is evaluated and aggregated, not the mathematical form of the per-slice Rényi bound. Therefore, two finite-profile SRPP guarantees should not be interpreted as providing the same protection merely because their native numerical (α, ε) values are equal. Their protection strength should be compared through a common operational metric, such as the induced Bayes-optimal attack accuracy (e.g., Experiment 2 of Sec. 4.5). The same logic applies when the target guarantee uses a continuous slice profile and the corresponding SW-sensitivity can be computed exactly.

Mode-II: Continuous-Profile SRPP via Finite Estimation. In this mode, the target guarantee is SRPP with a continuous slice profile, such as uniform ω on $\mathbb{S}^{d-1}$. The finite directions used in the mechanism are not the slice profile in the SRPP definition. Instead, they are used to estimate the continuous-profile quantities needed for Gaussian SWM calibration. For Ave-SRPP, this quantity is the second moment $\overline{\Delta}^2(\omega) = \mathbb{E}_{U\sim\omega}[(\Delta_\infty^U)^2]$. For Joint-SRPP, it is the continuous-profile log-moment envelope appearing in the Joint-SRPP calibration, not the worst-direction quantity $\Delta_\star^2$ used in the closed-form fixed-profile bound. Therefore, the resulting guarantee must account for both Monte Carlo slicing error and finite-sample error in the estimated per-slice SW-sensitivities. These errors are handled by Theorems 4.7–4.8, which give high-probability SRPP guarantees for the target continuous slice

profile. In this mode, different finite slice sets are not different finite-profile SRPP guarantees; they are different finite-slice and finite-sample estimates of the same continuous-profile target. A smaller number of sampled directions typically yields looser PAC-style calibration bounds, so the sufficient Gaussian noise scale in these bounds can increase in order to obtain the target SRPP guarantee with high probability.

This two-mode distinction highlights that the same computational object, a finite set of projection directions, has different roles in different settings. In **Mode-I**, the finite directions are the slice profile used in the SRPP definition. In **Mode-II**, by contrast, the finite directions are only used to estimate the quantities needed for a continuous-profile SRPP guarantee. This is analogous to how different choices of the protected secret-pair set Q in a Pufferfish scenario specify different privacy requirements: changing Q changes the Pufferfish semantic requirement, whereas changing the slice profile changes the quantification of the SRPP guarantee, not the underlying Pufferfish semantics.

The Effect of Finite Slicing. In **Mode-I**, a finite slice profile affects the directional resolution of the SRPP guarantee, not its validity for that profile. Both continuous-profile and finite-profile SRPP mechanisms perturb the full output vector using isotropic Gaussian noise; the slice profile is used to compute the noise variance and to specify the projections over which the sliced Rényi divergence is evaluated. It does not leave any output coordinate unperturbed. Under the standard finite-sensitivity assumption (common in DP), for any direction $u \in \mathbb{S}^{d-1}$, whether or not $u \in \mathcal{U}$, the per-slice SW-sensitivity Δ_∞^u is finite. Once the finite-profile SRPP mechanism fixes the Gaussian noise variance using the chosen profile $(\mathcal{U}, \omega)$, it defines a single full-dimensional randomized release $\mathcal{M}(X) = f(X) + N$, regardless of which projection direction is later used for analysis. Hence, directions outside $\mathcal{U}$ do not create unperturbed or infinite-leakage channels; they are simply not included in the finite-profile SRPP aggregation. Thus, compared with continuous profiles, a finite slice profile yields lower directional resolution in the SRPP quantification: the guarantee evaluates and aggregates the per-direction Rényi envelope bounds only over the chosen slice profile $(\mathcal{U}, \omega)$.

Approximation and estimation in Pufferfish-family mechanisms. The finite-slice and finite-sample bounds above should be understood in the broader context of practical Pufferfish-family privatization. Mechanisms for PP/RPP also depend on prior-driven, distribution-dependent sensitivity quantities, which are rarely available in closed form outside simple cases. Although PP/RPP does not use slicing, estimating its Wasserstein sensitivity in high dimensions can suffer from severe sample complexity and the curse of dimensionality. SRPP does not eliminate these scenario-dependent estimation requirements; rather, it moves the relevant calibration problem into sliced, 1-D forms for which finite-sample and finite-slice errors can be explicitly controlled.

4.4.1 Guidance on Slice Profile Selection.

The choice of slice profile determines how sliced Rényi divergence is evaluated and aggregated in the SRPP guarantee.

Uniform Default. A uniform profile over the sphere is a natural default when there is no prior reason to weight some projection directions more than others in the SRPP quantification. It treats directions isotropically and aligns with common practice in the sliced Wasserstein distance (SWD) literature. Finite profiles drawn uniformly from the sphere provide a practical and scalable implementation of this principle.

Application-Specific Profiles. SRPP does not require the slice profile to be uniform. An application-specific profile means that the SRPP guarantee places more weight on projection directions that are independently identified as relevant for the released output. This does not change the protected secrets or the Pufferfish scenario; it changes how the sliced Rényi divergence is evaluated and aggregated. For example, domain knowledge or sensitivity analysis may indicate that certain coordinates, features, or linear combinations of outputs carry more secret-relevant variation, motivating a structured profile.

This is consistent with the SWD literature: although standard SWD uses uniform directions, non-uniform variants optimize, weight, or adapt projection distributions to problem geometry [8, 28]. In SRPP, the analogous choice is the slice profile used in the sliced Rényi quantification. This is also conceptually related to geometry-aware DP mechanisms, such as anisotropic Gaussian mechanisms [42], but the roles differ: DP geometry shapes the noise covariance, whereas SRPP geometry shapes the sliced divergence being evaluated. Such geometry must be justified independently of the privacy definition. A structured-profile SRPP guarantee remains valid SRPP, just as a properly calibrated anisotropic Gaussian mechanism remains valid DP. Choosing such a profile is therefore an application-dependent design choice, not a search for a universally best approximation to PP/RPP. It should balance the computational cost of evaluating the selected directions with the intended resolution of the SRPP quantification for the deployment. Developing adaptive or application-aware methods for slice-profile selection is therefore a natural and important direction for future work.

4.5 Example: SRPP vs. RPP

In this section, we provide a tractable numerical experiment to illustrate the difference between full-dimensional RPP and finite-profile SRPP under **Mode-I** (Sec. 4.4). We quantify the gap between their native certificates (i.e., the numerical values of privacy parameters) and their operational security (via Bayes-optimal attack accuracy). See App. W for details.

Setup. Consider a Pufferfish scenario $(\mathcal{S}, Q, \Theta)$, with $\mathcal{S} = \{s_1, s_2, s_3\}$, $Q = \{(s_i, s_j) : i \neq j\}$, and $\Theta = \{\theta^A, \theta^B\}$. The data domain contains tuples $x = (i, b, z)$, where $i \in \{1,2,3\}$, $b \in \{A, B\}$, and $z \in \mathcal{Z} := \{(0,0,0), (0.1, 0.1, 0.1)\}$. Define a query

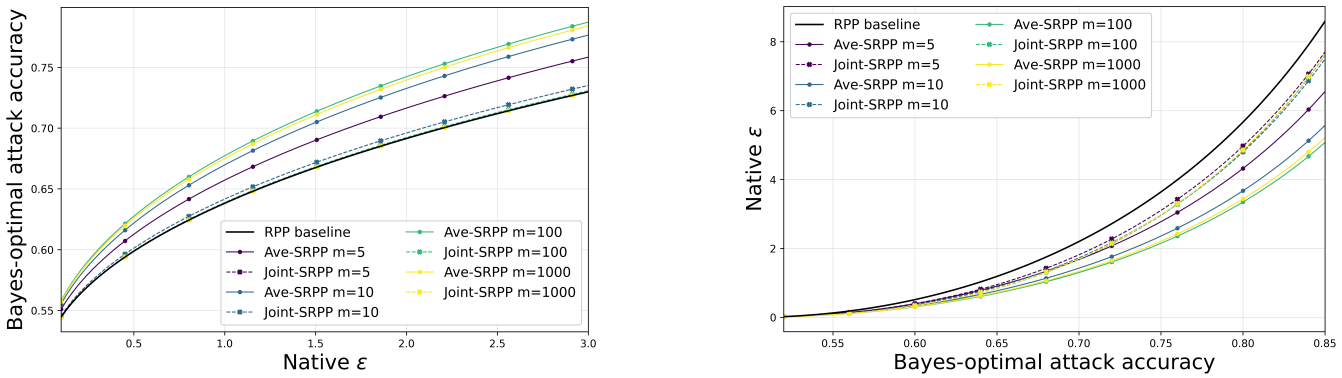


Figure 2: Comparison of RPP, Ave-SRPP, and Joint-SRPP under finite slice profiles: (Left) induced Bayes-optimal attack accuracy at the same native ε; (Right) native certificate values at the same Bayes-optimal attack accuracy.

function $f(x) := \mu_i^b + z \in \mathbb{R}^3$, where $\mu_1^A = e_1$, $\mu_2^A = e_2$, $\mu_3^A = e_3$, $\mu_1^B = e_1$, $\mu_2^B = \frac{1}{2}(e_1 + e_2)$, and $\mu_3^B = e_3$. Under prior θ^b, conditioning on secret s_i means that the data has the form $X = (i, b, Z)$, where Z is uniform on $\mathcal{Z}$. Hence, $P_{\theta^b}^{f,s_i}$ is the distribution of $\mu_i^b + Z$. We apply the Gaussian mechanism $\mathcal{M}(X) = f(X) + N$ with $N \sim \mathcal{N}(0, \sigma^2 I_3)$ and set $\alpha = 4$.

Sensitivities. The exact full-dimensional Wasserstein sensitivity for RPP is $\Delta_\infty^2 = \max_{b\in\{A,B\},(s_i,s_j)\in Q} \|\mu_i^b - \mu_j^b\|_2^2 = 2$. For SRPP, we evaluate nested finite slice profiles $\mathcal{U}_m \subset \mathbb{S}^2$ with uniform weights for $m \in \{5, 10, 100, 1000\}$, where $\mathcal{U}_m$ consists of the first m directions from a single size-1000 sample. For each direction u_ℓ, the per-slice SW-sensitivity is $\Delta_\infty^{u_\ell} = \max_{b\in\{A,B\},(s_i,s_j)\in Q} |\langle u_\ell, \mu_i^b - \mu_j^b\rangle|$. The Ave-SRPP and Joint-SRPP sensitivity quantities are $\overline{\Delta}^2(\mathcal{U}_m) = \frac{1}{m}\sum_{\ell=1}^m (\Delta_\infty^{u_\ell})^2$ and $\Delta_\star^2(\mathcal{U}_m) = \max_{\ell\in[m]} (\Delta_\infty^{u_\ell})^2$, respectively.

We conduct two reciprocal comparisons between RPP and SRPP:

Experiment 1: fixed certificate → protection strength. First, we fix the native privacy budget ε value across RPP, Ave-SRPP, and Joint-SRPP, calibrate their respective σ^2 values, using the closed-form worst-direction calibration for Joint-SRPP, and report the worst-pair Bayes-optimal attack accuracy, $\mathrm{Acc}^\star(\sigma^2) = \Phi(\sqrt{2}/(2\sigma))$. As shown in the left panel of Fig. 2, equal native numerical ε values at the same fixed α do not imply equal operational protection across RPP and finite-profile SRPP, because the certificates are defined using different quantification measures and, for SRPP, different slice profiles. Thus, finite-profile SRPP certificates should be interpreted relative to their specified slice profiles rather than compared to RPP solely by matching ε values.

Experiment 2: fixed protection strength → certificate gap. Second, we fix the target Bayes-optimal attack accuracy a, compute the physically required common noise variance $\sigma_a^2 = [2(\Phi^{-1}(a))^2]^{-1}$, and calculate the native certificate values assigned by each framework to this identical physical mechanism. Letting $t(a) = (\Phi^{-1}(a))^2$, the assigned certificates are: $\varepsilon_{\mathrm{RPP}}(a) = 8t(a)$, $\varepsilon_{\mathrm{Ave}}(a; \mathcal{U}_m) = 4\overline{\Delta}^2(\mathcal{U}_m)t(a)$, and $\varepsilon_{\mathrm{Joint}}(a; \mathcal{U}_m) = \frac{1}{3}\log\left[\frac{1}{m}\sum_{\ell=1}^m \exp\left(12(\Delta_\infty^{u_\ell})^2 t(a)\right)\right]$. The Joint-SRPP value $\varepsilon_{\mathrm{Joint}}(a; \mathcal{U}_m)$ is the tight log-mean-exp envelope; the closed-form Joint bound replaces it by the worst-direction upper bound (See App. W). As shown in the right panel of Fig. 2, once protection is fixed on a common operational scale, RPP and finite-profile SRPP assign different native certificate values to the same Gaussian mechanism. This illustrates that "weaker" or "stronger" protection is not determined by the native ε value alone when the underlying quantification measures differ; such comparisons require a common operational metric, such as Bayes-optimal attack accuracy.

5 SRPP-Learning with Gradient Clipping

For stochastic iterative algorithms such as SGD, directly bounding the end-to-end sensitivity of an entire training run is typically intractable under both DP and Pufferfish scenarios. Following the decompose–then–compose paradigm of DP-SGD with gradient clipping (Sec. A.2), we extend SRPP to stochastic learning by introducing SRPP-SGD with gradient clipping, an SRPP analogue of DP-SGD in the Pufferfish setting. We begin by fixing notation.

Dataset. Let $x = (x_1, \ldots, x_n) \in \mathcal{X} = \bar{\mathcal{X}}^n$ be the finite samples of data points, where each $x_i = (a_i, b_i) \in \bar{\mathcal{X}}$ consists of features a_i and label b_i. In this section, we take P_θ^X as the empirical (marginal) distribution of the data samples $x = (x_1, \ldots, x_n) \in \mathcal{X} = \bar{\mathcal{X}}^n$ for all $\theta \in \Theta$. At iteration t, we draw a random mini-batch $\mathsf{I}_t \subseteq [n]$ using a subsampling scheme $\{\eta, \rho\}$ characterized by a rate η ($\eta = \eta_t$ may vary with t) and a scheme type ρ, where $\rho \in \{\mathsf{WR}, \mathsf{WOR}, \mathsf{Poisson}\}$ denotes sampling with-replacement (WR), without-replacement (WOR), or via a Poisson process (Poisson), respectively. Let $X^t = (X_i)_{i\in\mathsf{I}_t}$ denote the random mini-batch subsampled from x, with x^t representing a particular realization.

Sampling randomness. Given a subsampling scheme $\{\eta, \rho\}$, let $(\mathcal{R}, \mathscr{R}, \mathbb{P}_{\eta,\rho})$ denote its probability space, with

Algorithm 1 SRPP-SGD with gradient clipping

Require: Data $x_1,\dots,x_n$; loss $\ell(\xi;x)$; subsampling scheme $\{\eta,\delta\}$; clip norm C; update maps $(T_t)_{t=1}^T$; noise covariances $(\Sigma_t)_{t=1}^T$.

1: Initialize parameters ξ_0; set history $y_{<1} \leftarrow \xi_0$
2: **for** $t = 1,\dots,T$ **do**
3: Sample minibatch index set $\mathsf{I}_t \subseteq [n]$ via scheme $\{\eta,\delta\}$; let $B_t \leftarrow |\mathsf{I}_t|$
4: For each $i \in \mathsf{I}_t$:

$$g_t(x_i) \leftarrow \nabla_\xi \ell(\xi_{t-1};x_i),$$
$$\tilde{g}_t(x_i) \leftarrow g_t(x_i) \cdot \min\left\{1,\ \frac{C}{\|g_t(x_i)\|_2}\right\}.$$

5: $\bar{g}_t \leftarrow \frac{1}{B_t}\sum_{i\in\mathsf{I}_t} \tilde{g}_t(x_i)$
6: Draw $N_t \sim \mathcal{N}(0,\Sigma_t)$ and set $\hat{g}_t \leftarrow \bar{g}_t + N_t$
7: $\xi_t \leftarrow T_t(\hat{g}_t; y_{<t})$
8: Update history $y_{<t+1}$
9: **end for**
10: **return** ξ_T; compute SRPP parameters from (Σ_t) via HUC/sa-HUC accountant.

$\mathbb{P}_{\eta,\rho}$ the corresponding probability measure. For a sampling draw $r \sim \mathbb{P}_{\eta,\rho}$, let $\mathsf{I}_t(r) \subseteq [n]$ denote the (multi)set of selected indices at iteration t and $B_t(r) := |\mathsf{I}_t(r)|$ its size. For $\rho \in \{\mathsf{WR},\mathsf{WOR}\}$, we have $B_t(r) \equiv B$ (deterministic). For Poisson subsampling ($\rho = \mathsf{Poisson}$; see App. E for details), $B_t(r)$ is random. We write $\mathcal{B} : \bar{\mathcal{X}}^n \times \mathcal{R} \to \bar{\mathcal{X}}^{B_t(r)}$ and $x^t = \mathcal{B}(x;r) := (x_i)_{i\in\mathsf{I}_t(r)}$ for the (mini-batch) subsampling operator.

Empirical Risk Minimization. In supervised learning, each example is $x_i = (a_i, b_i)$ with features a_i and label b_i. Let $\ell(\xi;x_i)$ denote the per-sample loss of a model with parameter ξ. The empirical risk minimization (ERM) problem is $\min_\xi F(\xi) := \frac{1}{n}\sum_{i=1}^n \ell(\xi;x_i)$.

(Clipped) SGD as a history-dependent query. At iteration t, a subsampling scheme $\{\eta,\rho\}$ draws randomness $r_t \sim \mathbb{P}_{\eta,\rho}$ and selects a minibatch $\mathcal{B}_t(x;r_t) = (x_i)_{i\in\mathsf{I}_t}$ of size B. Let $y_{<t} \in \mathcal{Y}_{<t}$ denote the history available at iteration t; for standard SGD, $y_{<t} = \xi_{t-1}$. Define the per-example gradients

$$g_t(x_i;\xi_{t-1}) := \nabla_\xi \ell(\xi_{t-1};x_i),$$

and the clipped batch gradient $\bar{g}_t(r_t) := \frac{1}{B}\sum_{i\in\mathsf{I}_t} \tilde{g}_t(x_i;\xi_{t-1})$, where $\tilde{g}_t(x_i)$ is the per-example clipped gradient (Line 4 of Algorithm 1). The pre-noise update (query) with step size $\kappa_t > 0$ is

$$f_t(x, y_{<t}; r_t) := \xi_{t-1} - \kappa_t\, \bar{g}_t(r_t),$$

and we suppress r_t when unambiguous.

In DP-SGD, composition is key to the effectiveness of the DP privatization [1]. But PP/RPP generally lack graceful composition [21, 34, 38, 46] because conditioning on secrets via the prior can induce dependencies between the outputs of mechanisms that are otherwise independent [46]. The slicing process of SRPP does not circumvent such induced independence, and thus SRPP inherits the RPP's lack of proper composition properties. To address this and make SRPP applicable to stochastic learning, we introduce the *History-Uniform Cap*.

5.1 History-Uniform Cap

We define HUC for a fixed Pufferfish scenario $(\mathcal{S},\mathcal{Q},\Theta)$ and a slice profile $(\mathcal{U},\omega)$. For any $(s_i,s_j) \in \mathcal{Q}$ and $\theta \in \Theta$, let $\Pi(\mu^\theta_{s_i},\mu^\theta_{s_j})$ denote the set of couplings between $\mu^\theta_{s_i}$ and $\mu^\theta_{s_j}$ (Definition 2.2), where $\mu^\theta_{s_i}$ and $\mu^\theta_{s_j}$ are the distributions of X given $(S = s_i,\theta)$ and of X' given $(S = s_j,\theta)$, respectively.

Definition 5.1 (History-Uniform Cap (HUC))**.** *A vector $h_t = (h_{t,i})_{i=1}^m \in \mathbb{R}_+^m$ is a HUC for $\mathcal{U} = \{u_i\}_{i=1}^m$ at iteration t if, for all $\theta \in \Theta$, $(s_i,s_j) \in \mathcal{Q}$, history $y_{<t}$ in the support, and every coupling $\gamma \in \Pi(\mu^\theta_{s_i},\mu^\theta_{s_j})$, we have $\gamma \times \mathbb{P}_{\eta,\rho}$-a.s. in $((X,X'),R_t)$,*

$$\left|\langle f_t(X,y_{<t};R_t) - f_t(X',y_{<t};R_t), u_i\rangle\right| \le \sqrt{h_{t,i}}, \forall i \in [m]. \quad (13)$$

Thus, a HUC $h_t = (h_{t,i})_{i=1}^m$ provides a *pathwise* bound that is uniform over histories: for every admissible coupling and every history $y_{<t}$ in support, the directional displacement along u_i is almost surely bounded by $\sqrt{h_{t,i}}$, and the constants $h_{t,i}$ do not depend on the particular value of $y_{<t}$. In matrix form, we may represent these caps by a PSD *HUC matrix* $H_t \succeq 0$ such that, $\gamma \times \mathbb{P}_{\eta,\rho}$-a.s. in $((X,X'),R_t)$,

$$\left|\left\langle f_t(X,y_{<t};R_t) - f_t(X',y_{<t};R_t), u\right\rangle\right|^2 \le u^\top H_t u, \quad \forall u \in \mathcal{U},$$

and in particular $h_{t,i} = u_i^\top H_t u_i$ for all $i \in [m]$. We view H_t as a deterministic, history-uniform sensitivity ellipsoid at iteration t, while h_t records its values on the slice set $\mathcal{U}$.

We next show that a finite HUC always exists. We start by introducing necessary notions.

Discrepancy cap. A general (probabilistic) secret–dataset relation may alter attributes, individual data points, or global properties of datasets input to algorithms as the secret varies. Under minibatch subsampling from a fixed finite dataset $x = (x_1,\dots,x_n)$, its effect on the iteration-t update is fully captured by *how many sampled points differ* between two coupled datasets.[3] For a sampling draw $r \sim \mathbb{P}_{\eta,\rho}$ with index (multi)set $\mathsf{I}_t(r)$ and size $B_t = |\mathsf{I}_t(r)|$[4], define the *discrepancy*

$$K_t(x,x';r) := \sum\nolimits_{j\in\mathsf{I}_t(r)} \mathbf{1}\{x_j \neq x'_j\}. \quad (14)$$

Given a coupling $\gamma \in \Pi(\mu^\theta_{s_i},\mu^\theta_{s_j})$, define the *iteration-$t$ discrepancy cap* as

$$K_t(\gamma) := \operatorname*{ess\,sup}_{((X,X'),R_t)\sim\gamma\times\mathbb{P}_{\eta,\rho}} K_t(X,X';R_t). \quad (15)$$

[3] App. G clarifies how this coupling-based viewpoint differs from *group* DP's worst-case k-neighbor model.

[4] We replace B_t by $B_t^{\max} := \operatorname{ess\,sup}_{R_t\sim\mathbb{P}_{\eta,\rho}} |\mathsf{I}_t(R_t)|$ if the size is random.

If a pair $(s_i, s_j) \in Q$ guarantees that X and X' differ in at most $\overline{K}$ coordinates γ-a.s., then $K_t(\gamma) \le \min\{\overline{K}, B_t\}$ at iteration t for the draw r. We say that a value K_t is a *feasible discrepancy cap* at iteration t (for the realized r) if $K_t \in [K_t(\gamma), B_t]$ for all $\theta \in \Theta$ and $(s_i, s_j) \in Q$. Intuitively, a feasible discrepancy cap provides a uniform upper bound on the number of differing sampled points that can arise at round t across all priors and admissible secret pairs. Operationally, estimating the discrepancy cap reduces to a counting problem: under a coupling of the two secret-conditioned data distributions, count how many minibatch records can differ when the protected secret is switched. The HUC cap controls this discrepancy count in a worst-case or tail sense. The subsampling-aware variant introduced below in Sec. 5.2 instead controls the corresponding second moment under minibatch sampling. App. D gives Monte Carlo and analytic estimators with finite-sample guarantees.

Update map and L_t-Lipschitz. At iteration t, we write the pre-perturbation update as a map $T_t : \mathbb{R}^d \times \mathcal{Y}_{<t} \to \mathbb{R}^d$ such that, for any realized sampling draw r,

$$f_t(x, y_{<t}; r) = T_t\big(\bar{g}_t(\mathcal{B}(x; r)); y_{<t}\big),$$

where $\bar{g}_t(\mathcal{B}(x;r)) \in \mathbb{R}^d$ is the clipped and averaged minibatch gradient (Line 5 of Algorithm 1). We say that T_t is *L_t-Lipschitz* in its first argument if there exists $L_t < \infty$ such that, for all $z, z' \in \mathbb{R}^d$ and all histories $y_{<t}$ in the support, $\big\|T_t(z; y_{<t}) - T_t(z'; y_{<t})\big\|_2 \le L_t \|z - z'\|_2$. App. C gives two common examples of L_t-Lipschitz update maps.

Assumption 5.2 (Slice-wise Lipschitz updates)**.** *For each iteration t and each direction $u_i \in \mathcal{U}$, there exists a finite constant $L_{t,i} < \infty$ such that, for all $z, z' \in \mathbb{R}^d$ and all histories $y_{<t}$ in the support, $\big|u_i^\top \big(T_t(z; y_{<t}) - T_t(z'; y_{<t})\big)\big| \le L_{t,i} \big|u_i^\top (z - z')\big|$.*

Assumption 5.2 enforces a directional non-expansiveness property: for each $u_i \in \mathcal{U}$, the u_i-component of the update cannot amplify the u_i-component of a gradient perturbation by more than $L_{t,i}$. This holds for standard SGD and for preconditioned updates with bounded linear preconditioners. Assumption 5.2 is a convenient sufficient condition for a fixed finite direction set $\mathcal{U}$. One may alternatively assume global ℓ_2-Lipschitzness with constant L_t of T_t, which implies Assumption 5.2 with $L_{t,i} \le L_t$ for all i, or impose other geometric constraints. App. F discusses structured (e.g., per-layer) clipping/perturbation variants that yield analogous slice-wise bounds with modified constants.

Proposition 5.1. *Fix a slice profile $(\mathcal{U}, \omega)$. Suppose per-example gradients are ℓ_2-clipped at threshold $C > 0$. Let K_t be a feasible discrepancy cap at iteration t, and let $B_t \ge 1$ denote the (deterministic) minibatch size. Suppose Assumption 5.2 holds. Then, the following is a valid HUC for each $u_i \in \mathcal{U}$,*

$$h_{t,i} := \Big(\frac{2K_t L_{t,i} C}{B_t}\Big)^2. \tag{16}$$

Proposition 5.1 provides an explicit worst-case, history-uniform bound on the directional displacement of the iteration-t update. If at most K_t sampled examples differ in a minibatch of size B_t, then the averaged clipped gradient can change by at most $\frac{2K_t C}{B_t}$ in any direction. Assumption 5.2 further implies that the u_i-component of the update can expand this change by at most a factor $L_{t,i}$. Thus, the directional shift along u_i is bounded by $\frac{2K_t L_{t,i} C}{B_t}$, and hence the squared shift is bounded by $(\frac{2K_t L_{t,i} C}{B_t})^2$, which matches the cap $h_{t,i}$ in (16). App. H characterizes HUC's *minimality* and *attainability*.

5.2 Subsampling-Aware HUC

Any feasible discrepancy cap K_t used in $h_{t,i}$ is a uniform worst-case upper bound on the random discrepancy $K_t(X, X'; R_t)$ under $\gamma \times \mathbb{P}_{\eta,\rho}$. While this worst-case viewpoint is sound, it can be overly conservative in SGD, where the subsampling randomness η is independent of the prior and has a genuine averaging effect. To better reflect this structure and improve utility, we introduce a *subsampling-aware HUC* (sa-HUC), which uniformly bounds the *expected* squared directional shifts over subsampling, while retaining worst-case uniformity over priors, secrets, and couplings.

Definition 5.3 (Subsampling-Aware History-Uniform Cap (sa-HUC))**.** *Fix t and a slicing set $\mathcal{U} = \{u_i\}_{i=1}^m \subset \mathbb{S}^{d-1}$. A vector $h_t^{\mathsf{sa}} = (h_{t,i}^{\mathsf{sa}})_{i=1}^m \in \mathbb{R}_+^m$ is a sa-HUC for $\mathcal{U}$ at iteration t if, for all $\theta \in \Theta$, all $(s_i, s_j) \in Q$, all histories $y_{<t}$ in the support, and every coupling $\gamma \in \Pi(\mu_{s_i}^\theta, \mu_{s_j}^\theta)$, we have, for γ-almost every (X, X'), all $i = 1, \dots, m$,*

$$\mathbb{E}_{\eta_t}\Big[\big|\langle f_t(X, y_{<t}; R_t) - f_t(X', y_{<t}; R_t), u_i\rangle\big|^2\Big] \le h_{t,i}^{\mathsf{sa}},$$

where the expectation is taken only over the subsampling randomness $R_t \sim \mathbb{P}_{\eta,\rho}$.

Let $\overline{K}_t^2$ be any scalar such that, for all $\theta \in \Theta$, $(s_i, s_j) \in Q$, all histories $y_{<t}$ in the support, and all couplings $\gamma \in \Pi(\mu_{s_i}^\theta, \mu_{s_j}^\theta)$, we have, for γ-almost every (X, X'),

$$\mathbb{E}_{\eta_t}\big[K_t(X, X'; R_t)^2\big] \le \overline{K}_t^2. \tag{17}$$

Proposition 5.2. *Fix an iteration t and a slicing set $\mathcal{U} = \{u_i\}_{i=1}^m \subset \mathbb{S}^{d-1}$. Assume:*

- *Per-example gradients are ℓ_2-clipped at radius $C > 0$.*
- *The minibatch size $B_t \ge 1$ is deterministic.*
- *Assumption 5.2 holds with constants $\{L_{t,i}\}_{i=1}^m$.*

For each $\theta \in \Theta$, $(s_i, s_j) \in Q$, history $y_{<t}$ in the support, and $\gamma \in \Pi(\mu_{s_i}^\theta, \mu_{s_j}^\theta)$, let $\overline{K}_t^2$ satisfy (17). Then the vector $h_t^{\mathsf{sa}} = (h_{t,i}^{\mathsf{sa}})_{i=1}^m$ with

$$h_{t,i}^{\mathsf{sa}} := \Big(\frac{2L_{t,i} C}{B_t}\Big)^2 \overline{K}_t^2, \qquad i = 1, \dots, m, \tag{18}$$

is a valid sa-HUC.

Proposition 5.2 shows that, under slice-wise Lipschitz updates and ℓ_2-clipping, the subsampling-aware discrepancy $\overline{K}_t^2$ directly yields a closed-form sa-HUC. In particular, if $\overline{K}_t^2$ uniformly bounds the expected squared discrepancy K_t^2 over subsampling, then the subsampling-aware directional shift along each slice u_i is controlled by $h_{t,i}^{\mathsf{sa}} = \left(\frac{2L_{t,i}C}{B_t}\right)^2 \overline{K}_t^2$. Thus h_t^{sa} is a valid sa-HUC.

To calibrate privacy using $\mathbb{E}_{\eta_t}[K_t^2]$ instead of a worst-case K_t, we now introduce subsampling-aware variants of Ave- and Joint-SRPP that match the sa-HUC notion. Let η denote all subsampling randomness used in SRPP-SGD, and for each fixed η, let $\mathcal{M}_s^{\theta,\eta}$ be the distribution of the mechanism's output under prior θ and secret s, conditional on η.

Definition 5.4 (Subsampling-Aware Average SRPP (sa-Ave-SRPP))**.** *Fix $\alpha > 1$ and a slice profile $(\mathcal{U}, \omega)$. A mechanism $\mathcal{M}$ with subsampling randomness η satisfies $(\alpha, \varepsilon, \omega)$-sa-Ave-SRPP if, for all $\theta \in \Theta$ and all $(s_i, s_j) \in \mathcal{Q}$,*

$$\mathbb{E}_\eta\left[\mathtt{ASD}_\alpha^\omega\left(\mathcal{M}_{s_i}^{\theta,\eta} \,\middle\|\, \mathcal{M}_{s_j}^{\theta,\eta}\right)\right] \le \varepsilon.$$

Here the expectation is taken only over η.

Compared to standard Ave-SRPP, sa-Ave-SRPP keeps the same sliced Rényi divergence and the same uniformity over priors and secrets, but relaxes the guarantee from a worst-case bound for each subsampling pattern η to an *on-average* bound over the subsampling randomness.

Definition 5.5 (Subsampling-Aware Joint SRPP (sa-Joint-SRPP))**.** *Under the same setup as Definition 5.4, a mechanism $\mathcal{M}$ satisfies $(\alpha, \varepsilon, \omega)$-sa-Joint-SRPP if, for all $\theta \in \Theta$ and all $(s_i, s_j) \in \mathcal{Q}$,*

$$\mathbb{E}_\eta\left[\mathtt{JSD}_\alpha^\omega\left(\mathcal{M}_{s_i}^{\theta,\eta} \,\middle\|\, \mathcal{M}_{s_j}^{\theta,\eta}\right)\right] \le \varepsilon.$$

Here, sa-Joint-SRPP is the subsampling-aware analogue of Joint-SRPP: it still treats the slice index and projection jointly, but now requires that the joint sliced Rényi divergence be small *on average over subsampling* rather than for every fixed realization of η. As with sa-Ave-SRPP, the only change from the standard notion is replacing a worst-case guarantee in η by an expectation over η.

5.3 HUC Accountant for Gaussian Noise

Similar to DP-SGD, an important issue of SRPP-SGD is computing the aggregated privacy cost of the entire training process, i.e., the *composition part* of SRPP-SGD as a decompose-then-compose privatization scheme. In this section, we set out to show how HUC can circumvent the RPP/SRPP's lack of graceful composition properties and enable a privacy *accountant* process to compute the privacy costs along the SGD-based training process. We refer to it as *HUC accountant*.

We start by introducing the Gaussian slice Rényi Envelope, which gives a per-iteration, per-slice RPP cost after noise perturbation.

Lemma 5.1. *Fix t and a slicing direction $u_i \in \mathbb{S}^{d-1}$. Let $N_t \sim \mathcal{N}(0, \Sigma_t)$ be independent of $((X, X'), R_t)$ and set $Y_t = f_t(X, y_{<t}; R_t) + N_t$. Define $v_{t,i} := u_i^\top \Sigma_t u_i$, and let $h_t = (h_{t,1}, \ldots, h_{t,m})$ be any HUC vector. Assume that $v_{t,i} > 0$. Then, for any $\alpha > 1$, we have*

$$\mathtt{D}_\alpha\left(\Pr\left(\langle Y_t, u_i\rangle \mid s_i, \theta\right) \middle\| \Pr\left(\langle Y_t, u_i\rangle \mid s_j, \theta\right)\right) \le \frac{\alpha}{2}\frac{h_{t,i}}{v_{t,i}}. \quad (19)$$

For ease of exposition, we absorb any step-size κ and preconditioning into the covariance Σ_t of the Gaussian noise. Lemma 5.1 gives a per-slice Rényi cost at iteration t directly in terms of the HUC component $h_{t,i}$ and the variance $v_{t,i}$ of the projected noise.

Theorem 5.6. *Fix a slicing profile $\{\mathcal{U}, \omega\}$. Let $h = \{h_1, \ldots, h_T\}$ with each $h_t = (h_{t,1}, \ldots, h_{t,m})$ as a* valid HUC *vector. Let $N_t \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2 I_d)$. Then, for any $\alpha > 1$, we have:*

(i) **Ave-SRPP:** *For any $\sigma^2 \ge \frac{\alpha}{2\varepsilon}\sum_{t=1}^{T}\sum_{\ell=1}^{m}\omega_\ell h_{t,\ell}$, Algorithm 1 is $(\alpha, \varepsilon, \omega)$-Ave-SRPP.*

(ii) **Joint-SRPP:** *For any $\sigma^2 \ge \frac{\alpha}{2\varepsilon}\sum_{t=1}^{T}\max_\ell h_{t,\ell}$, Algorithm 1 is $(\alpha, \varepsilon, \omega)$-Joint-SRPP.*

Theorem 5.6 shows that once a valid sequence of $\{h_t\}_{t=1}^T$ has been constructed for a fixed slicing set $\mathcal{U}$, the overall SRPP guarantee for the entire SGD trajectory reduces to explicit, closed-form calibration conditions on the Gaussian noise. Both Ave-SRPP and Joint-SRPP are ensured by choosing a single variance parameter σ^2 large enough to offset the cumulative per-step HUC contributions, either in the ω-weighted average form (part (i)) or under the worst-slice aggregation (part (ii)).

Theorem 5.7. *Fix a slicing profile $\{\mathcal{U}, \omega\}$ with $\mathcal{U} = \{u_\ell\}_{\ell=1}^m$ and $\omega \in \Delta(\mathcal{U})$. Let $h^{\mathsf{sa}} = \{h_1^{\mathsf{sa}}, \ldots, h_T^{\mathsf{sa}}\}$ with each $h_t^{\mathsf{sa}} = (h_{t,1}^{\mathsf{sa}}, \ldots, h_{t,m}^{\mathsf{sa}})$ a* valid sa-HUC *vector. Let $N_t \overset{i.i.d.}{\sim} \mathcal{N}(0, \sigma^2 I_d)$. Then, for any $\alpha > 1$, we have:*

(i) **sa-Ave-SRPP:** *If $\sigma^2 \ge \frac{\alpha}{2\varepsilon}\sum_{t=1}^{T}\sum_{\ell=1}^{m}\omega_\ell h_{t,\ell}^{\mathsf{sa}}$, then lgorithm 1 satisfies $(\alpha, \varepsilon, \omega)$-sa-Ave-SRPP.*

(ii) **sa-Joint-SRPP:** *If $\sigma^2 \ge \frac{\alpha}{2\varepsilon}\sum_{t=1}^{T}\max_\ell h_{t,\ell}^{\mathsf{sa}}$, then Algorithm 1 satisfies $(\alpha, \varepsilon, \omega)$-sa-Joint-SRPP.*

Theorem 5.7 is the subsampling-aware analogue of Theorem 5.6. Once we have a sequence of sa-HUC caps h_t^{sa} that bound the *average* (over subsampling) squared directional shifts at each iteration, the overall sa-SRPP guarantee for the full SGD trajectory reduces to the same kind of closed-form calibration on a single noise level σ^2. Part (i) ensures an $(\alpha, \varepsilon, \omega)$ sa-Ave-SRPP bound by matching the ω-weighted sum of sa-HUCs, while part (ii) does the same for

sa-Joint-SRPP using the worst-slice aggregation. Compared to the standard HUC case, the structure of the accountant is unchanged; the only difference is that the guarantee is now on-average over subsampling randomness. In App. F, we characterize SGD with per-layer gradient clipping [26], where we impose an assumption of layer-wise Lipschitz updates instead of Assumption 5.2.

Proposition 5.3 (Utility of sa-SRPP, informal)**.** *Under mild conditions, subsampling-aware discrepancy caps yield more utility than worst-case caps at the same true SRPP of the mechanism.* (see Proposition J.1 in App. J for formal results)

Proposition 5.3 confirms that averaging over the subsampling randomness improves utility at the same true SRPP. App. J provides formal and detailed analysis.

5.4 Composition of SRPP-SGD Mechanisms

In this section, we show that the HUC- and sa-HUC-based SRPP/sa-SRPP-SGD mechanisms developed above admit additive composition under a shared slice profile. This result should not be interpreted as a general composition theorem for arbitrary SRPP mechanisms. As in PP/RPP, arbitrary mechanisms may fail to compose gracefully because conditioning on Pufferfish secrets can induce dependencies across releases. The result below applies to the structured SRPP-SGD mechanisms considered in this section, where the per-iteration secret-induced shifts are controlled by HUC or sa-HUC sequences and the mechanisms are calibrated under the same slice profile. For clarity, we present the setup and results in terms of HUC and SRPP; the same arguments apply to sa-HUC and sa-SRPP after replacing the corresponding notions.

We consider J SRPP-SGD mechanisms $\mathcal{M}_1, \ldots, \mathcal{M}_J$ acting on the same dataset X. Without loss of generality, we take the primitive output of each mechanism to be a vector in a common parameter space $\mathbb{R}^d$ (e.g., a final clipped update or parameter vector), and we fix a shared slice profile $(\mathcal{U}, \omega)$ with $\mathcal{U} = \{u_\ell\}_{\ell=1}^m \subset \mathbb{S}^{d-1}$ and $\omega \in \Delta(\mathcal{U})$. In this section, we restrict to the following mechanisms: for each $j \in \{1, \ldots, J\}$, the run $\mathcal{M}_j : \mathcal{X} \to \mathbb{R}^d$ is an SRPP-SGD mechanism with iterates $\theta_t^{(j)} \in \mathbb{R}^d$ $(t = 0, \ldots, T_j)$, additive Gaussian noises $N_t^{(j)} \sim \mathcal{N}(0, \Sigma_t^{(j)})$, and an associated HUC (or sa-HUC) sequence $h^{(j)} = \{h_t^{(j)}\}_{t=1}^{T_j}$ that satisfies the conditions of Theorem 5.6 (resp. Theorem 5.7). The composition is

$$\vec{\mathcal{M}}(X) := \big(\mathcal{M}_1(X), \ldots, \mathcal{M}_J(X)\big) \in (\mathbb{R}^d)^J. \tag{20}$$

Theorem 5.8. *Fix a slice profile $\{\mathcal{U}, \omega\}$ and a Pufferfish scenario $\{\mathcal{S}, \mathcal{Q}, \Theta\}$. Let $\vec{\mathcal{M}}$ be the composition in (20). Fix any $\Xi \in \{\text{Ave}, \text{Joint}, \text{sa-Ave}, \text{sa-Joint}\}$. Suppose that, for each $j \in \{1, \ldots, J\}$, $\mathcal{M}_j$ is an HUC-based or sa-HUC-based SRPP-SGD mechanism satisfying the corresponding assumptions of Theorem 5.6 or Theorem 5.7, and suppose that $\mathcal{M}_j$ is $(\alpha, \varepsilon_j, \omega)$-$\Xi$-SRPP under the shared slice profile $\{\mathcal{U}, \omega\}$. Then the composed mechanism $\vec{\mathcal{M}}$ is $(\alpha, \sum_{j=1}^J \varepsilon_j, \omega)$-$\Xi$-SRPP.*

In other words, for any fixed SRPP notion (Ave, Joint, or their subsampling-aware counterparts), the privacy costs of J SRPP-SGD mechanisms acting on the same data and slice profile add up linearly: the composed mechanism has the same α and ω, with total privacy cost $\sum_j \varepsilon_j$.

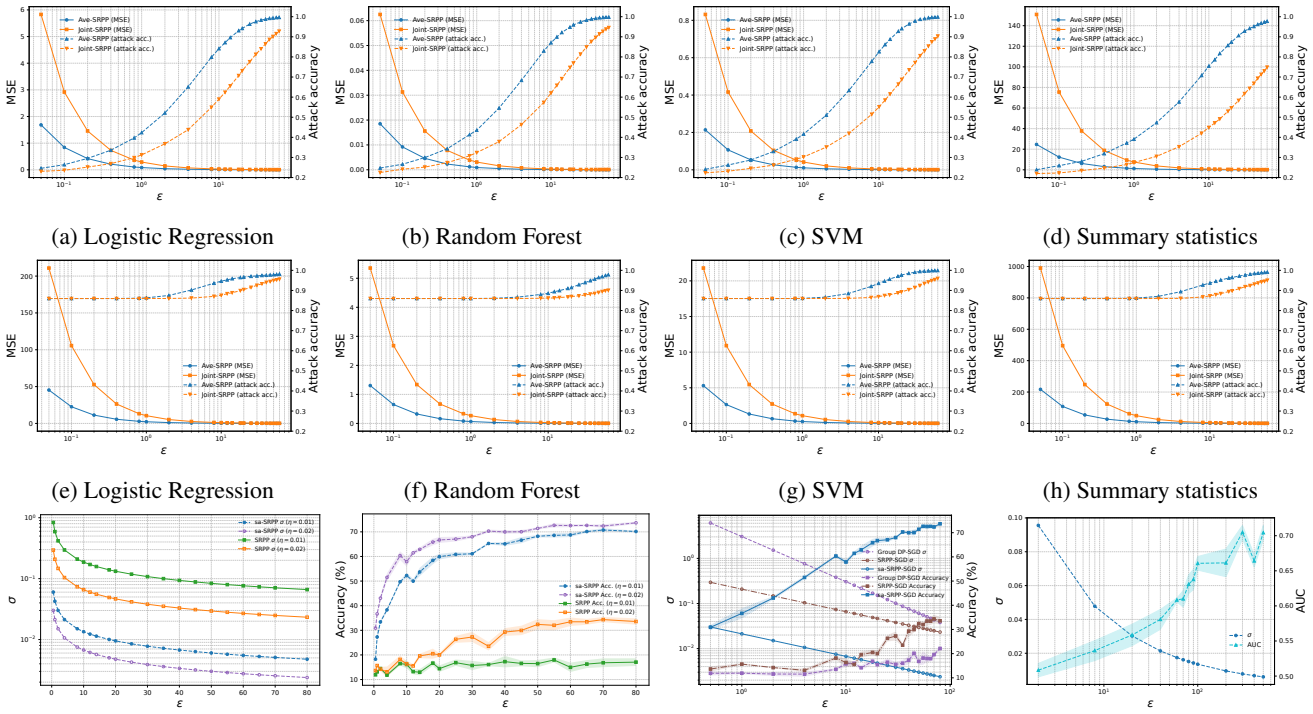
6 Experiments

We empirically evaluate our Gaussian sliced Wasserstein mechanisms in both the static and stochastic-learning settings. For static queries, we instantiate SRPP guarantees (Section 4), and for iterative training we use SRPP/sa-SRPP together with gradient clipping to privatize SGD (Section 5). Full experimental details and additional plots are in App. X.

6.1 Gaussian Sliced Wasserstein Mechanisms

Dataset and Pufferfish scenario. We evaluate our sliced Wasserstein mechanisms on three standard tabular benchmarks in an attribute-privacy setting [49]: Adult Census [2] (race as secret, $k = 5$), Cleveland Heart Disease [9], and Student Performance [36], with Pufferfish query set $Q = \{(s_i, s_j) : s_i \neq s_j\}$. We present results for Adult and App. X shows results for the other datasets. For Adult, we consider both the natural imbalanced race prior (majority race probability 0.86) and a balanced, stratified subsample with a uniform 0.20 prior over the five race categories. We study two query families: (i) low-dimensional summary statistics (means, standard deviations, and rates of selected features) conditioned on each secret value; and (ii) machine learning (logistic regression, random forest, and SVM) queries on Adult, trained separately per secret group. In all experiments, we use Gaussian mechanisms $Y = \mathcal{M}(X) + N$, where $N \sim \mathcal{N}(0, I_d \sigma)$ with σ calibrated via our sliced SRPP. Utility is measured by mean squared error and related norms between $\mathcal{M}(X)$ and Y. We also use a prior-aware Gaussian MAP attack on the secret attribute to perform empirical privacy auditing, reporting accuracy and advantage over the prior baseline (i.e., 0.86 for imbalanced Adult, 0.20 for balanced Adult).

Figures 3a-3d and 3e-3h show utility-privacy trade-off as a function of the SRPP budget ε on balanced and imbalanced Adult datasets for four query types. Across all four queries, the behavior is consistent for both balanced and imbalanced settings. For very strong privacy (small ε), both Ave- and Joint-SRPP force high noise and the MAP attack accuracy stays close to the prior baselines (i.e., 0.2 and 0.86), indicating that the mechanism reveals little additional information about the race. As ε increases, MSE decreases toward zero while attack accuracy rises monotonically, reflecting the expected utility-privacy trade-off. In addition, at any fixed ε, Joint-SRPP is more conservative than Ave-SRPP: it induces larger MSE but



(a) Logistic Regression (b) Random Forest (c) SVM (d) Summary statistics

(e) Logistic Regression (f) Random Forest (g) SVM (h) Summary statistics

(i) ResNet: σ (j) ResNet: Accuracy (k) SRPP vs. Group DP (l) Empirical Attack Auditing

Figure 3: Privacy–utility tradeoffs with the Gaussian sliced Wasserstein mechanism. **(a)–(d)** ($\alpha = 4$) Ave-/Joint-SRPP on balanced Adult (uniform 0.2 prior) for four query families: logistic regression, SVM, random forest feature importances, and summary statistics; each panel plots utility and attack accuracy vs. the SRPP budget ε. **(e)–(h)** ($\alpha = 4$) the same on imbalanced Adult (0.86 prior). **(i)–(l)** use $\alpha = 16$ on CIFAR-10. **(i)** calibrated per-iteration noise σ vs. ε for SRPP-SGD and sa-SRPP-SGD at $\eta \in 0.01, 0.02$. **(j)** corresponding test accuracy vs. ε. **(k)** σ and test accuracy for group-DP-SGD, SRPP-SGD, and sa-SRPP-SGD under matched clipping, subsampling, and model. **(l)** overfitted SRPP-SGD ($\eta = 0.02$): empirical MIA attack AUC vs. the sa-SRPP budget ε.

yields uniformly lower attacker accuracy. This gap is most visible in the moderate privacy regime, where the Ave-SRPP curves already allow the attacker to reach substantially higher accuracy, while Joint-SRPP keeps the attack much closer to the prior baseline.

6.2 SRPP-SGD Using Gaussian Noise

We evaluate our SRPP-SGD and sa-SRPP-SGD mechanisms on image classification with Gaussian noise added to clipped gradients in a DP-SGD-style training pipeline. In our implementation, the HUC envelope is instantiated using a *global* ℓ_2-Lipschitz bound on the (clipped, averaged) gradient update map. Specifically, for each iteration t we bound the change in the update under neighboring worlds by a scalar constant L_t, so the global sensitivity of the update in Euclidean norm is at most L_t. This global envelope automatically upper-bounds all 1-D projections and therefore induces valid SRPP guarantees for any slicing profile $(\mathcal{U}, \omega)$ via the linear-update setting. Hence, Ave- and Joint-SRPP-SGD would have the same HUC. App. X.2 provides more details.

Dataset and Pufferfish scenario. We use CIFAR-10 (50,000 training and 10,000 test images). We create a two-world Pufferfish scenario, where `cat` is designated as the secret class, and construct two worlds s_0 and s_1 that differ only in the prevalence of this class: in the prior, the `cat` frequency shifts from $p_{\text{low}} = 0.10000$ to $p_{\text{high}} = 0.10004$. The realized world s_0 is then fixed and encoded by overwriting the CIFAR-10 training labels with the realized label vector; all training procedures (private and non-private) operate only on this realized world.

Models. Our main model is a ResNet-22 similar to [42], implemented with ELU activations, GroupNorm-based layers, a 3×3 adaptive average pooling, and a linear classifier on a standardized $64 \times 3 \times 3$ feature vector. The architecture is made differentially private–compatible (ready for Opacus [45]) by replacing non–DP-friendly components (such as non-picklable shortcuts) with equivalent modules.

SRPP-SGD and sa-SRPP-SGD training. Training follows the standard DP-SGD template: at each iteration, per-sample gradients are clipped to an ℓ_2 bound C and then perturbed by isotropic Gaussian noise. We use a worst-case dis-

crepancy cap K_{cap} for SRPP-SGD and use $\mathbb{E}_\eta[K_t^2]$ for sa-SRPP-SGD. We pre-compute, for each target SRPP and sa-SRPP budget ε at a fixed Rényi order $\alpha = 16$, a per-step noise scale $\sigma(\varepsilon)$ that ensures the resulting Gaussian mechanisms satisfies the SRPP- and sa-SRPP-SGD under the given Pufferfish scenario. In addition, we include a diagnostic group-DP-SGD curve under the same experimental setup, differing only in the noise calibration. For this curve, we follow the standard DP-SGD calibration [1] and convert it to a group-DP-style calibration using an effective group-size proxy derived from the sa-HUC mean-square discrepancy cap $\overline{K}_t^2$, namely $k_{\text{eff}} = \lceil \sqrt{\overline{K}_t^2} \rceil$; see App. G for details.

Overfitted regime and empirical privacy attacker. In addition to the above accuracy-focused experiments on the full training set, we run a separate overfitted regime tailored to empirical privacy evaluation. We restrict the training data to a small random subset of CIFAR-10 (on the order of 10^3 images), substantially increase the number of epochs, and remove or reduce weight decay, so that the resulting private SRPP-SGD models exhibit markedly higher training accuracy than test accuracy. On these overfitted models, we apply the loss-threshold membership inference attack of Yeom et al. [44]: for each trained model, we draw disjoint sets of member and non-member points, compute per-example cross-entropy losses, and use the negative loss as a scalar membership score (lower loss $\Rightarrow$ more likely member). We report the AUC of this score as a function of ε and compare against a non-private overfitted baseline trained on the same subset.

As shown in Fig. 3i–3j, across all ε and for both step sizes, sa-SRPP consistently requires much smaller Gaussian noise than worst-case SRPP: the sa-SRPP σ curves lie well below their SRPP counterparts, often by a factor of several and approaching an order of magnitude around $\varepsilon \approx 20$–40. This tighter calibration directly translates into higher utility. For small budgets both methods yield low accuracy, but as ε increases the sa-SRPP models continue to improve, reaching roughly 60%–70% test accuracy at the largest budgets, whereas SRPP saturates around 20%–35%. The same pattern holds for both $\eta = 0.01$ and $\eta = 0.02$: sa-SRPP dominates SRPP in terms of the accuracy–privacy tradeoff, demonstrating that replacing a worst-case discrepancy cap K_{cap} by the subsampling-aware cap $\mathbb{E}_\eta[K_t^2]$ yields substantially milder noise for the same ε.

Although group-DP, SRPP, and sa-SRPP are different privacy notions with different semantic goals, Fig. 3k reports a diagnostic group-DP-style calibration applied to the same intended Pufferfish secret. Specifically, we run DP-SGD with an effective group-size proxy derived from the sa-HUC mean-square discrepancy cap $\overline{K}_t^2$; see App. G for the calibration and for the distinction between this surrogate and the SRPP/sa-SRPP. This curve should not be read as a direct baseline for SRPP-SGD or sa-SRPP-SGD, since it corresponds to a different formal privacy guarantee. Rather, it is a diagnostic surrogate showing what happens when the Pufferfish secret-level dependency is translated into an effective group-size parameter for DP-SGD. In this setting, that translation yields conservative noise levels across the range of ε. The SRPP/sa-SRPP calibration developed in Sec. 5 instead keeps the calibration within the Pufferfish framework by using HUC/sa-HUC discrepancy caps rather than a group-size surrogate.

In the overfitted setting, the calibrated noise scale σ exhibits the same privacy–utility behavior as in Fig. 3i: tighter sa-SRPP budgets (smaller ε) require larger σ. Fig. 3l shows the corresponding stress-test of the empirical membership-inference attack. Tighter budgets (larger noise) yield weaker membership inference, while looser budgets reduce the noise at the cost of stronger membership-inference signals.

7 Conclusion

We introduced *Sliced Rényi Pufferfish Privacy* (SRPP), a PP/RPP-semantics-preserving framework that replaces high-dimensional Rényi divergence with projection-based quantification via *Average*- and *Joint*-SRPP. To mitigate the curse of dimensionality in PP-family mechanism design, we developed *sliced Wasserstein mechanisms* calibrated by a computable sliced Wasserstein sensitivity, yielding provable SRPP guarantees and closed-form Gaussian noise calibration without incurring the computational burden of full optimal transport. We further provided finite-sample characterizations, including conditions under which Gaussian noise achieves SRPP with high confidence.

To enable practical privacy accounting for iterative learning under probabilistic secret–dataset relationships, we proposed SRPP-SGD with gradient clipping and new accountants based on *History-Uniform Caps* (HUC) and a subsampling-aware variant (sa-HUC) that improves utility. This supports a decompose-then-compose privatization strategy without requiring Wasserstein sensitivity computation or contractivity assumptions. Under a shared slicing profile, we proved graceful additive composition for Ave/Joint SRPP and their subsampling-aware counterparts, enabling modular accounting for multi-stage pipelines and cascades. Empirically, our experiments on both static privatization and SRPP-SGD demonstrate favorable privacy–utility trade-offs and scalability.

Ethical Considerations

We propose Sliced Rényi Pufferfish Privacy (SRPP) together with practical sliced mechanisms (e.g., Ave-/Joint-SRPP and SRPP-SGD) to make Pufferfish-style, scenario-based privacy more tractable to certify and deploy in high dimensions while preserving the intended semantics. Our contributions are methodological: we do not collect new data, do not scrape data, do not interact with production systems or APIs, and do not discover or disclose vulnerabilities. We organize ethical

considerations along the generic pipeline

Data $\rightarrow$ **Mechanisms** $\rightarrow$ **Downstream Decision-Making**,

and focus on the benefits and risks of releasing privatized query answers or trained models in a trusted-curator setting.

Stakeholders

Data subjects and curators. Although our experiments use public benchmarks, SRPP-like mechanisms may be applied to sensitive real-world datasets (e.g., health, education, finance). Relevant stakeholders include data subjects, dataset curators, and data stewards responsible for governance and consent.
Practitioners and auditors. Data scientists, ML engineers, privacy/security teams, and external auditors may rely on SRPP guarantees to decide whether (and how) to release statistics or models under a specified Pufferfish scenario $(\mathcal{S}, Q, \Theta)$.
Affected individuals and oversight bodies. People impacted by downstream decisions (e.g., lending, admissions, clinical risk scoring) and regulators/standards bodies that interpret privacy claims and assess compliance.

Potential Harms and Mitigations

Misinterpretation and privacy-washing. SRPP is scenario-based: its protection depends on the declared secrets $\mathcal{S}$, discriminative pairs Q, and admissible worlds Θ. A key harm is misrepresenting a scenario-specific guarantee as universal privacy, or obscuring the assumptions behind the guarantee (e.g., which secret pairs are protected). We mitigate this by clearly stating the threat model and emphasizing that deployments must document the chosen scenario, parameters, and scope, and should be reviewed by privacy/security stakeholders.
Utility degradation and downstream harms. As with other rigorous privacy frameworks, achieving stronger privacy typically requires adding noise, which can reduce accuracy and may cause harmful errors in high-stakes contexts. Our work does not deploy SRPP in real decision pipelines; we study trade-offs on benchmarks. For real deployments, we recommend conservative parameter choices when harms are high, explicit evaluation of downstream impact (including error costs and subgroup effects when appropriate), and governance processes that do not treat formal bounds as the sole determinant of acceptability.
Implementation and approximation pitfalls. Practical instantiations may involve Monte Carlo estimation, finite slice profiles, numerical optimization, or other approximations. Under-estimating sensitivity surrogates or using unstable solvers can lead to insufficient noise and invalid guarantees. We mitigate this risk by highlighting approximation failure modes, recommending conservative numerical tolerances and sanity checks, and encouraging reproducible implementations and independent auditing for high-stakes deployments.

Decision to Conduct and Publish

We believe this work has positive expected impact by making scenario-based privacy more practical in high dimensions and by promoting transparent, auditable privacy specifications tied to explicit assumptions. At the same time, SRPP does not eliminate privacy risk and should not be used to justify aggressive data use without clear communication of scope and limits, careful parameter selection, and appropriate oversight. Deployment decisions remain context-dependent.

Open Science

Our artifacts consist of source code archived on Zenodo: https://zenodo.org/records/20433043. The repository includes implementations and reproduction scripts, with public benchmarks downloaded automatically from official sources. Experiments used Python 3.x and PyTorch on a single NVIDIA A40 48 GB GPU.

Acknowledgements

This work was partially supported by the NSF (IIS-2214141, ITE-2452834), ARO (W911NF-25-1-0059), ONR (N000142412663), and Amazon.

References

[1] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, and Li Zhang. Deep learning with differential privacy. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*, CCS '16, page 308–318, 2016.

[2] Barry Becker and Ronny Kohavi. Adult. *UCI Machine Learning Repository*, 10:C5XW20, 1996.

[3] Clément Bonet, Lucas Drumetz, and Nicolas Courty. Sliced-Wasserstein Distances and Flows on Cartan-Hadamard Manifolds. *Journal of Machine Learning Research*, 26(32):1–76, 2025.

[4] Michelle Chen and Olga Ohrimenko. Protecting global properties of datasets with distribution privacy mechanisms. In *International Conference on Artificial Intelligence and Statistics*, pages 7472–7491. PMLR, 2023.

[5] Lenaic Chizat, Pierre Roussillon, Flavien Léger, François-Xavier Vialard, and Gabriel Peyré. Faster wasserstein distance estimation with the sinkhorn divergence. *Advances in neural information processing systems*, 33:2257–2269, 2020.

[6] Harald Cramér and Herman Wold. Some theorems on distribution functions. *Journal of the London Mathematical Society*, 1(4):290–294, 1936.

[7] Marco Cuturi. Sinkhorn distances: Lightspeed computation of optimal transport. *Advances in neural information processing systems*, 26, 2013.

[8] Ishan Deshpande, Yuan-Ting Hu, Ruoyu Sun, Ayis Pyrros, Nasir Siddiqui, Sanmi Koyejo, Zhizhen Zhao, David Forsyth, and Alexander G Schwing. Max-sliced wasserstein distance and its use for gans. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 10648–10656, 2019.

[9] Robert Detrano, Andras Janosi, Walter Steinbrunn, Matthias Pfisterer, Johann-Jakob Schmid, Sarbjit Sandhu, Kern H Guppy, Stella Lee, and Victor Froelicher. International application of a new probability algorithm for the diagnosis of coronary artery disease. *The American journal of cardiology*, 64(5):304–310, 1989.

[10] Ni Ding. Kantorovich mechanism for pufferfish privacy. In *International Conference on Artificial Intelligence and Statistics*, pages 5084–5103. PMLR, 2022.

[11] Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Annual international conference on the theory and applications of cryptographic techniques*, pages 486–503. Springer, 2006.

[12] Cynthia Dwork, Frank McSherry, Kobbi Nissim, and Adam Smith. Calibrating noise to sensitivity in private data analysis. In *Theory of cryptography conference*, pages 265–284. Springer, 2006.

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

[14] Vitaly Feldman, Ilya Mironov, Kunal Talwar, and Abhradeep Thakurta. Privacy amplification by iteration. In *2018 IEEE 59th Annual Symposium on Foundations of Computer Science (FOCS)*, pages 521–532. IEEE, 2018.

[15] Natasha Fernandes, Mark Dras, and Annabelle McIver. Generalised differential privacy for text document processing. In *International Conference on Principles of Security and Trust*, pages 123–148. Springer International Publishing Cham, 2019.

[16] Moritz Hardt and Aaron Roth. Beyond worst-case analysis in private singular vector computation. In *Proceedings of the forty-fifth annual ACM symposium on Theory of computing*, pages 331–340, 2013.

[17] Xi He, Ashwin Machanavajjhala, and Bolin Ding. Blowfish privacy: Tuning privacy-utility trade-offs using policies. In *Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data*, SIGMOD '14, page 1447–1458. Association for Computing Machinery, 2014.

[18] Sushil Jajodia, Pierangela Samarati, and Moti Yung. *Encyclopedia of Cryptography, Security and Privacy*. Springer, 2025.

[19] Shiva P Kasiviswanathan and Adam Smith. On the'semantics' of differential privacy: A bayesian formulation. *Journal of Privacy and Confidentiality*, 6(1), 2014.

[20] Stephan Kessler, Erik Buchmann, and Klemens Böhm. Deploying and evaluating pufferfish privacy for smart meter data. In *IEEE 12th Int. Conf. Ubiquitous Intell. Comput. IEEE 12th Int. Conf. Autonomic Trusted Comput. IEEE 15th Int. Conf. Scalable Comput. Commun. Associated Workshops (UIC-ATC-ScalCom)*, pages 229–238, 2015.

[21] Daniel Kifer and Ashwin Machanavajjhala. Pufferfish: A framework for mathematical privacy definitions. *ACM Transactions on Database Systems (TODS)*, 39(1):1–36, 2014.

[22] Philip A Knight. The sinkhorn–knopp algorithm: convergence and applications. *SIAM Journal on Matrix Analysis and Applications*, 30(1):261–275, 2008.

[23] Ios Kotsogiannis, Stelios Doudalis, Sam Haney, Ashwin Machanavajjhala, and Sharad Mehrotra. One-sided differential privacy. In *2020 IEEE 36th International Conference on Data Engineering (ICDE)*, pages 493–504. IEEE, 2020.

[24] Giulia Luise, Alessandro Rudi, Massimiliano Pontil, and Carlo Ciliberto. Differential properties of sinkhorn approximation for learning with wasserstein distance. *Advances in Neural Information Processing Systems*, 31, 2018.

[25] Pascal Massart. The tight constant in the dvoretzky-kiefer-wolfowitz inequality. *The annals of Probability*, pages 1269–1283, 1990.

[26] H Brendan McMahan, Daniel Ramage, Kunal Talwar, and Li Zhang. Learning differentially private recurrent language models. In *International Conference on Learning Representations*, 2018.

[27] Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th Computer Security Foundations Symposium (CSF)*, pages 263–275, 2017.

[28] Khai Nguyen, Nhat Ho, Tung Pham, and Hung Bui. Distributional sliced-wasserstein and applications to generative modeling. *arXiv preprint arXiv:2002.07367*, 2020.

[29] Sloan Nietert, Ziv Goldfeld, Ritwik Sadhu, and Kengo Kato. Statistical, robustness, and computational guarantees for sliced wasserstein distances. *Advances in Neural Information Processing Systems*, 35:28179–28193, 2022.

[30] Chaoyue Niu, Zhenzhe Zheng, Shaojie Tang, Xiaofeng Gao, and Fan Wu. Making big money from small sensors: Trading time-series data under pufferfish privacy. In *IEEE INFOCOM 2019-IEEE Conference on Computer Communications*, pages 568–576. IEEE, 2019.

[31] Theshani Nuradha and Ziv Goldfeld. Pufferfish privacy: An information-theoretic study. *IEEE Transactions on Information Theory*, 69(11):7336–7356, 2023.

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

[33] Gabriel Peyré, Marco Cuturi, et al. Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, 11(5-6):355–607, 2019.

[34] Clément Pierquin, Aurélien Bellet, Marc Tommasi, and Matthieu Boussard. Rényi pufferfish privacy: General additive noise mechanisms and privacy amplification by iteration via shift reduction lemmas. In *International Conference on Machine Learning (ICML 2024)*, 2024.

[35] Julien Rabin, Gabriel Peyré, Julie Delon, and Marc Bernot. Wasserstein barycenter and its application to texture mixing. In *International Conference on Scale Space and Variational Methods in Computer Vision*, pages 435–446. Springer, 2011.

[36] Alice Silva. Using data mining to predict secondary school student performance. 2008.

[37] Shuang Song, Kamalika Chaudhuri, and Anand D Sarwate. Stochastic gradient descent with differentially private updates. In *2013 IEEE global conference on signal and information processing*, pages 245–248. IEEE, 2013.

[38] Shuang Song, Yizhen Wang, and Kamalika Chaudhuri. Pufferfish privacy mechanisms for correlated data. In *Proceedings of the 2017 ACM International Conference on Management of Data*, SIGMOD '17, page 1291–1306, 2017.

[39] Mayuri Sridhar, Hanshen Xiao, and Srinivas Devadas. Pac-private algorithms. In *2025 IEEE Symposium on Security and Privacy (SP)*, pages 3839–3857. IEEE, 2025.

[40] Eric Wong, Frank Schmidt, and Zico Kolter. Wasserstein adversarial examples via projected sinkhorn iterations. In *International conference on machine learning*, pages 6808–6817. PMLR, 2019.

[41] Hanshen Xiao and Srinivas Devadas. Pac privacy: Automatic privacy measurement and control of data processing. In *Annual International Cryptology Conference*, pages 611–644. Springer, 2023.

[42] Hanshen Xiao, Jun Wan, and Srinivas Devadas. Geometry of sensitivity: Twice sampling and hybrid clipping in differential privacy with optimal gaussian noise and application to deep learning. In *Proceedings of the 2023 ACM SIGSAC Conference on Computer and Communications Security*, pages 2636–2650, 2023.

[43] Xiaokui Xiao and Yufei Tao. Output perturbation with query relaxation. *Proceedings of the VLDB Endowment*, 1(1):857–869, 2008.

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

[45] Ashkan Yousefpour, Igor Shilov, Alexandre Sablayrolles, Davide Testuggine, Karthik Prasad, Mani Malek, John Nguyen, Sayan Ghosh, Akash Bharadwaj, Jessica Zhao, et al. Opacus: User-friendly differential privacy library in pytorch. *arXiv preprint arXiv:2109.12298*, 2021.

[46] Tao Zhang, Bradley A Malin, Netanel Raviv, and Yevgeniy Vorobeychik. Differential confounding privacy and inverse composition. In *2025 IEEE International Symposium on Information Theory (ISIT)*, pages 1–6. IEEE, 2025.

[47] Tao Zhang and Yevgeniy Vorobeychik. Breaking the gaussian barrier: Residual-pac privacy for automatic privatization. *arXiv preprint arXiv:2506.06530*, 2025.

[48] Tao Zhang and Yevgeniy Vorobeychik. Sliced rényi pufferfish privacy: Tractable privatization mechanism and private learning with gradient clipping. Cryptology ePrint Archive, Paper 2026/321, 2026.

[49] Wanrong Zhang, Olga Ohrimenko, and Rachel Cummings. Attribute privacy: Framework and mechanisms. In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '22, page 757–766, New York, NY, USA, 2022. Association for Computing Machinery.