Privacy amplification by random allocation

Vitaly Feldman
Apple
vitaly.edu@gmail.com

Moshe Shenfeld[*]
The Hebrew university of Jerusalem
moshe.shenfeld@mail.huji.ac.il

Abstract

We consider the privacy amplification properties of a sampling scheme in which a user's data is used in k steps chosen randomly and uniformly from a sequence (or set) of t steps. This sampling scheme has been recently applied in the context of differentially private optimization [Chua et al., 2024a, Choquette-Choo et al., 2025] and is also motivated by communication-efficient high-dimensional private aggregation [Asi et al., 2025]. Existing analyses of this scheme either rely on privacy amplification by shuffling which leads to overly conservative bounds or require Monte Carlo simulations that are computationally prohibitive in most practical scenarios.

We give the first theoretical guarantees and numerical estimation algorithms for this sampling scheme. In particular, we demonstrate that the privacy guarantees of random k-out-of-t allocation can be upper bounded by the privacy guarantees of the well-studied independent (or Poisson) subsampling in which each step uses the user's data with probability $(1 + o(1))k/t$. Further, we provide two additional analysis techniques that lead to numerical improvements in several parameter regimes. Altogether, our bounds give efficiently-computable and nearly tight numerical results for random allocation applied to Gaussian noise addition.

1 Introduction

One of the central tools in the analysis of differentially private algorithms are so-called *privacy amplification* guarantees, where amplification results from sampling of the inputs. In these results one starts with a differentially private algorithms (or a sequence of such algorithms) and a randomized selection (or sampling) to determine which of the n elements in a dataset to run each of the t algorithms on. Importantly, the random bits of the sampling scheme and the selected data elements are not revealed. For a variety of sampling schemes this additional uncertainty is known to lead to improved privacy guarantees of the resulting algorithm, that is, privacy amplification.

In the simpler, single step case a DP algorithm is run on a randomly chosen subset of the dataset. As first shown by Kasiviswanathan et al. [2011], if each element of the dataset is included in the subset with probability λ (independently of other elements) then the privacy of the resulting algorithm is better (roughly) by a factor λ. This basic result has found numerous applications, most notably in the analysis of the differentially private stochastic gradient descent (DP-SGD) algorithm [Bassily et al., 2014]. In DP-SGD gradients are computed on randomly chosen batches of data points and then privatized through clipping and Gaussian noise addition. Privacy analysis of this algorithm is based on the so called Poisson sampling: elements in each batch and across batches are chosen randomly and independently of each other. The absence of dependence implies that the algorithm can be analyzed relatively easily as a direct composition of single step amplification results. The downside of this simplicity is that such sampling is less efficient and harder to implement within the standard ML pipelines. As a result, in practice some form of shuffling is used to define the batches in

[*] Work partially done while author was an intern at Apple

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

DP-SGD leading to a well-recognized discrepancy between the implementations of DP-SGD and their analysis [Chua et al., 2024b,c, Annamalai et al., 2024].

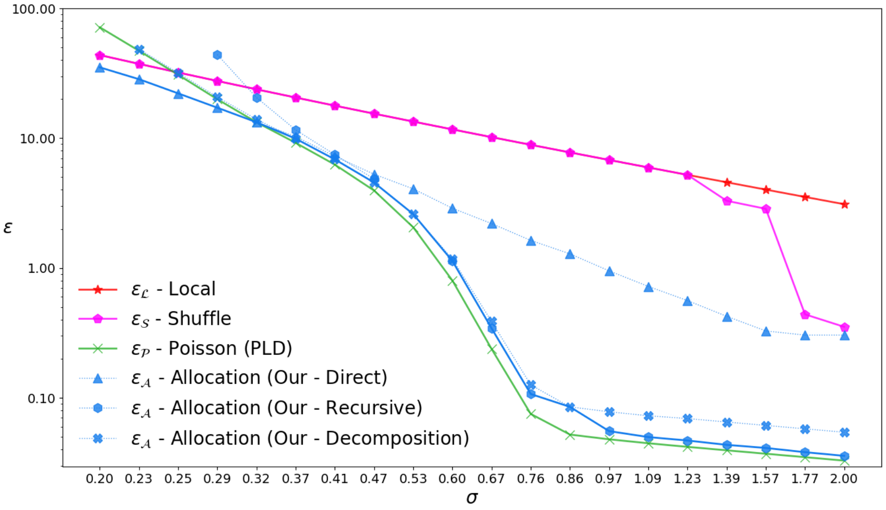


Figure 1: Upper bounds on privacy parameter ε as a function of the noise parameter σ for various schemes and the local algorithm (no amplification), all using the Gaussian mechanism with fixed parameters $\delta = 10^{-10}$, $t = 10^6$. In the Poisson scheme $\lambda = 1/t$. The "flat" part of the RDP based calculation is due to computational limitations, which was computed for the range $\alpha \in [2, 60]$.

Motivated by the shuffle model of federated data analysis [Bittau et al., 2017], Cheu et al. [2019], Erlingsson et al. [2019] have studied the privacy amplification of the shuffling scheme. In this scheme, the n elements are randomly and uniformly permuted and the i-th element in the permuted order is used in the i-th step of the algorithm. This sampling scheme can be used to analyze the implementations of DP-SGD used in practice [Erlingsson et al., 2019, Feldman et al., 2021]. However, the analysis of this sampling scheme is more involved and nearly tight results are known only for relatively simple pure DP ($\delta = 0$) algorithms [Feldman et al., 2021, 2023, Girgis et al., 2021]. In particular, applying these results to Gaussian noise addition requires using (ε, δ)-guarantees of the Gaussian noise. This leads to an additional $\sqrt{\ln(1/\delta)}$ factor in the asymptotic analysis and significantly worse numerical results (see Fig. 1 for comparison and discussion in Section 4.2).

Note that shuffling differs from Poisson subsampling in that participation of elements is dependent both in each step (or batch) and across the steps. If the participation of elements in each step is dependent (by fixing the total number of participating elements) but the steps are independent then the sampling scheme can be tightly analyzed as a direct composition of fixed subset size sampling steps (e.g., using bound in Balle et al. [2018], Zhu et al. [2022]). However, a more problematic aspect of Poisson sampling is the stochasticity in the number of times each element is used in all steps. For example, using Poisson sampling with sampling rate $1/t$ over t batches will result in a roughly $1/e$ probability of not using the sample which implies dropping approximately 37% of the data, and the additional sampling randomness may increase the resulting variance as demonstrated in Appendix H. In a distributed setting it is also often necessary to limit the maximum number of times a user participates in the analysis due to time or communication constraints on the protocol [Chen et al., 2024, Asi et al., 2025]. Poisson sampling does not allow to fully exploit the available limit which hurts the utility.

Motivated by the privacy analysis of DP-SGD and the problem of communication-efficient high-dimensional private aggregation with two servers [Asi et al., 2025], we analyze sampling schemes where each element participates in exactly k randomly chosen steps out of the total t, independently of other elements. We refer to this sampling as k-out-of-t *random allocation*. For $k = 1$, this scheme is a special case of the *random check-in* model of defining batches for DP-SGD in [Balle et al., 2020]. Their analysis of this variant relies on the amplification properties of shuffling and thus does not lead to better privacy guarantees than those that are known for shuffling. Very recently, Chua et al. [2024a] have studied such sampling (referring to it as *balls-and-bins sampling*) in the context of training neural networks via DP-SGD. Their main results show that from the point of view of utility (namely, accuracy of the final model) random allocation with $k = 1$ is essentially identical to shuffling and is noticeably better than Poisson sampling. Concurrently, Choquette-Choo et al. [2025] considered the same sampling scheme for the matrix mechanism in the context of DP-FTRL. The privacy analysis in these two works reduces the problem to analyzing the divergence of a specific pair of distributions on $\mathbb{R}^t$. They then used Monte Carlo simulations to estimate the privacy parameters of this pair. Their numerical results suggest that privacy guarantees of 1-out-of-t random allocation are similar to those of the Poisson sampling with rate of $1/t$. While very encouraging, such simulations have several limitations which we discuss in Appendix G.1, most notably, achieving high-confidence estimates for small δ and supporting composition appear to be computationally impractical. This approach also does not lead to provable privacy guarantees and does not lend itself to asymptotic analysis (such as the scaling of the privacy guarantees with t).

1.1 Our contribution

We provide three new analyses for the random allocation setting that result in provable guarantees that nearly match or exceed those of the Poisson subsampling at rate k/t. The analyses rely on different techniques and lead to incomparable numerical results. We describe the specific results below and illustrate the resulting bounds in Fig. 1.

In our main result we show that the privacy of random allocation is upper bounded by that of the Poisson scheme with sampling probability $\approx k/t$ up to lower order terms which are asymptotically vanishing in t/k. Specifically, we upper bound it by the k-wise composition of Poisson subsampling with rate $(1+\gamma)k/t$ applied to a dominating pair of distributions for each step of the original algorithm (Def. 2.6) with an additional $t\delta_0 + \delta'$ added to the δ parameter. Here, $\gamma = O\left(e^{\varepsilon_0}\sqrt{\frac{k\ln(k/\delta')}{t}}\right)$ and ε_0, δ_0 are the privacy parameters of the original algorithm. The formal statement of this result that includes all the constants can be found in Thm. 4.1. Additionally, we show in Thm. 4.1 this lower order term can be recursively bounded using (ε', δ') parameters of the same algorithm for some $\varepsilon' > \varepsilon$. This leads to significant numerical improvements in our results.

Our analysis relies on several simplification steps. Using a dominating pair of distributions for the steps of the original algorithm, we first derive an explicit dominating pair of distributions for random allocation (extending a similar result for Gaussian noise in [Chua et al., 2024a, Choquette-Choo et al., 2025]). Equivalently, we reduce the allocation for general multi-step adaptive algorithms to the analysis of random allocation for a single (non-adaptive) randomizer on two inputs. We also analyze only the case of $k = 1$ and then use a reduction from general k to $k = 1$. Finally, our analysis of the non-adaptive randomizer for $k = 1$ relies on a decomposition of the allocation scheme into a sequence of posterior sampling steps for which we then prove a high-probability bound on subsampling probability in each step.

We note that, in general, the privacy of the composition of subsampling of the dominating pair of distributions can be worse than the privacy of the sampling scheme of a concrete algorithm, even if this pair tightly dominates it. This is true for both Poisson and random allocation schemes. However, all existing analyses of the Poisson sampling are effectively based on composition of subsampling for a dominating pair of distributions. Moreover, if the algorithm has a pair of neighboring datasets inducing this dominating pair, then our upper bound can be stated directly in terms of the Poisson subsampling scheme with respect to this pair. Such *dominating input* exists for many standard algorithms including those based on Gaussian and Laplace noise addition.

While our result shows asymptotic equivalence of allocation and Poisson subsampling, it may lead to suboptimal bounds for small values of t/k and large ε_0. We address this using two additional techniques which are also useful as starting points for the recursive version of our main result.

We first show that ε of random allocation with $k = 1$ is at most a constant (≈ 1.6) factor times larger than ε of the Poisson sampling with rate $1/t$ for the same δ (see Theorem 4.3). This upper bound does not asymptotically approach Poisson subsampling but applies in all parameter regimes. To prove this upper bound we observe that Poisson subsampling is essentially a mixture of random allocation schemes with various values of k. We then prove a monotonicity property of random allocations showing that increasing k leads to worse privacy. Combining these results with the advanced joint convexity property Balle et al. [2018] gives the upper bound.

Finally, we give a direct analysis of the divergence for the dominating pair of distributions. Due to the asymmetric nature of the add/remove privacy our bounds require different techniques for each of the directions. In the remove direction we derive a closed form expression for the Rényi DP [Mironov, 2017] of the dominating pair of distributions for allocation in terms of the RDP parameters of the original algorithm (Theorem 4.4). This method has two important advantages. First, it gives a precise bound on the RDP parameters of integer order (as opposed to just an upper bound). Secondly, it is particularly easy to use in the typical setting where composition is used in addition to a sampling scheme (for example, when $k > 1$ or in multi-epoch DP-SGD). The primary disadvantage of this technique is that the conversion from RDP bounds to the regular (ε, δ) bounds is known to be somewhat lossy (typically within 10-20$\%$ range in multi-epoch settings). The same loss is also incurred when Poisson sampling is analyzed via RDP (referred to as moment accounting [Abadi et al., 2016]). Two more limitations of this technique result from the restriction to the range $\alpha \geq 2$, and the computational complexity when α is in the high tens.

For the add direction we give an approximate upper bound in terms of the usual composition of a different, explicitly defined randomizer over the same domain. While this bound is approximate, the divergence for the add direction is typically significantly lower than the one for the remove direction and therefore, in our evaluations, this approximation has either minor or no effect on the maximum. Overall, in our evaluations of this method for Gaussian distribution in most regimes the resulting bounds are almost indistinguishable from those obtained via RDP for Poisson distribution (see Fig. 6 for examples). In fact, in some regimes it is better than Poisson sampling (Figure 5).

Numerical evaluation: In Section 5 we provide numerical evaluation and comparisons of our bounds to those for Poisson sampling as well as other relevant bounds.[2] Our evaluations across many parameter regimes give bounds on the privacy of random allocation that are very close, typically within 10% of those for the Poisson subsampling with rate k/t. This means that random allocation can be used to replace Poisson subsampling with only a minor loss in privacy. At the same time, in many cases, the use of random allocation can improve utility. In the context of training neural networks via DP-SGD this was shown in [Chua et al., 2024a]. Application of our bounds also lead to improvement over Poisson subsampling in [Asi et al., 2025]. We demonstrate that even disregarding some practical disadvantages of Poisson subsampling, random allocation has a better privacy-utility trade-off for mean estimation in low-dimensional regime. This improvement stems from the fact that random allocation computes the sum exactly whereas Poisson subsampling introduces additional variance. At the same time in the high-dimensional regime noise due to privacy dominates the final error and thus the trade-off boils down to the difference in the privacy bounds.

1.2 Related work

Our work builds heavily on tools and ideas developed for analysis of privacy amplification by subsampling, composition and shuffling. We have covered the work directly related to ours earlier and will describe some of the tools and their origins in the preliminaries. A more detailed technical and historical overview of subsampling and composition for DP can be found in the survey by Steinke [2022]. The shuffle model was first proposed by Bittau et al. [2017]. The formal analysis of the privacy guarantees in this model was initiated in [Erlingsson et al., 2019, Cheu et al., 2019]. The sequential shuffling scheme that we discuss here was defined by Erlingsson et al. [2019] who proved the first general privacy amplification results for this scheme albeit only for pure DP algorithms. Improved analyses and extensions to approximate DP were given in [Balle et al., 2019, 2020, Feldman et al., 2021, 2023, Girgis et al., 2021, Koskela et al., 2022].

DP-SGD was first defined and theoretically analyzed in the convex setting by Bassily et al. [2014]. Its use in machine learning was spearheaded by the landmark work of Abadi et al. [2016] who significantly improved the privacy analysis via the moments accounting technique and demonstrated the practical utility of the approach. In addition to a wide range of practical applications, this work has motivated the development of more advanced techniques for analysis of sampling and composition. At the same time most analyses used in practice still assume Poisson subsampling when selecting batches whereas some type of shuffling is used in implementation. It was recently shown that it

[2] Python implementation of all methods is available in a GitHub project and in a package.

results in an actual difference between the reported and true privacy level in some regimes [Chua et al., 2024b,c, Annamalai et al., 2024].

In a concurrent and independent work Dong et al. [2025] considered the same sampling method (referring to it as *Balanced Iteration Subsampling*). Their results are closest in spirit to our direct bounds. Specifically, they provide RDP-based bounds for the same dominating pair of distributions in the Gaussian case for both add and remove directions. Their bound for general k is incomparable to ours as it is based on a potentially loose upper bound for divergences of order $\alpha > 2$, while using an exact extension of their approximation to $k > 1$. In contrast, our RDP-based bound uses a reduction from general k to $k = 1$ that is potentially loose but our computation for the $k = 1$ case is exact (for the remove direction which is typically larger than the add direction). In our numerical comparisons, the bounds in Dong et al. [2025] are comparable or worse than our direct bounds and are often significantly worse than the bounds from our main result. We discuss these differences in more detail and provide numerical comparison in Appendix G.2.

2 Preliminaries

We denote the domain of *elements* by $\mathcal{X}$ and the set of possible *outputs* by $\mathcal{Y}$. Given a dataset $\boldsymbol{s} \in \mathcal{X}^*$ and an output $y \in \mathcal{Y}$, we denote by $P_M(y|\boldsymbol{s}) \coloneqq \underset{Y \sim M(\boldsymbol{s})}{\mathbb{P}}(Y = y)$ the probability of observing the output y as the output of some randomized algorithm M which was given dataset $\boldsymbol{s}$ as input.

2.1 Sampling schemes

In this work, we consider *t-step algorithms* defined using an algorithm M that takes some subset of the dataset and a sequence of previous outputs as input. Formally, let $\mathcal{Y}^{<t} = \bigcup_{i<t} \mathcal{Y}^i$. M takes a dataset in $\mathcal{X}^*$ and a view $\boldsymbol{v} \in \mathcal{Y}^{<t}$ as its inputs and outputs a value in $\mathcal{Y}$. A t-step algorithm defined by M first uses some scheme to define t subsets $\boldsymbol{s}^1, \ldots, \boldsymbol{s}^t \subseteq \boldsymbol{s}$, then sequentially computes $y_i = M\left(\boldsymbol{s}^i, \boldsymbol{v}^{i-1}\right)$, where $\boldsymbol{v}^i \coloneqq (y_1, \ldots, y_i)$ are the intermediate *views* consisting of the outputs produced so far, and $\boldsymbol{v}^0 = \emptyset$. Such algorithms include DP-SGD, where each step consists of a call to the Gaussian mechanism (A.2), with gradient vectors adaptively defined as a function of previous outputs.

The assignment of the elements in $\boldsymbol{s}$ to the various subsets can be done in a deterministic manner (e.g., $\boldsymbol{s}^1 = \ldots = \boldsymbol{s}^t = \boldsymbol{s}$), or randomly using a *sampling scheme*. We consider the following three sampling schemes.

1. *Poisson scheme* parametrized by sampling probability $\lambda \in [0, 1]$, where each element is added to each subset $\boldsymbol{s}^i$ with probability λ independent of the other elements and other subsets,
2. *Shuffling scheme* which uniformly samples a permutation π over $[n]$ where n is the sample size, and sets $\boldsymbol{s}^i = \{x_{\pi(i)}\}$ (in this case, the sample size and number of steps must match).
3. *Random allocation scheme* parameterized by a number of selected steps $k \in [t]$, which uniformly samples k indices $\boldsymbol{i} = (i_1, \ldots, i_k) \subseteq [t]$ for each element and adds them to the corresponding subsets $\boldsymbol{s}^{i_1}, \ldots, \boldsymbol{s}^{i_k}$.

For a t-step algorithm defined by M, we denote by $\mathcal{P}_{t,\lambda}(M) : \mathcal{X}^* \to \mathcal{Y}^t$ an algorithm using M with the Poisson sampling scheme, $\mathcal{S}_n(M) : \mathcal{X}^n \to \mathcal{Y}^n$ for the shuffling scheme, and $\mathcal{A}_{t,k}(M) : \mathcal{X}^* \to \mathcal{Y}^t$ when M is used with the random allocation scheme. When $k = 1$ we omit it from the notation for clarity.

2.2 Privacy notions

We consider the standard add/remove adjacency notion of privacy in which datasets $\boldsymbol{s}, \boldsymbol{s}' \in \mathcal{X}^*$ are neighboring if $\boldsymbol{s}$ can be obtained from $\boldsymbol{s}'$ via adding or removing one of the elements. To appropriately define sampling schemes that operate over a fixed number of elements we augment the domain with a "null" element $\perp$, that is, we define $\mathcal{X}' \coloneqq \mathcal{X} \cup \{\perp\}$. When a t-step algorithm assigns $\perp$ to M we treat it as an empty set, that is, for any $\boldsymbol{s} \in \mathcal{X}^*$, $\boldsymbol{v} \in \mathcal{Y}^*$ we have $M(\boldsymbol{s}, \boldsymbol{v}) = M((\boldsymbol{s}, \perp), \boldsymbol{v})$. We say that two datasets $\boldsymbol{s}, \boldsymbol{s}' \in \mathcal{X}^n$ are *neighbors* and denote it by $\boldsymbol{s} \simeq \boldsymbol{s}'$, if one of the two can be created by replacing a single element in the other dataset by $\perp$.

We rely on the hockey-stick divergence to quantify the privacy loss.

Definition 2.1. Given $\kappa \geq 0$ and two distributions P, Q over some domain Ω, the *hockey-stick divergence* between them is defined to be

$$\boldsymbol{H}_\kappa \left(P \parallel Q \right) \coloneqq \int_\Omega \left[P(\omega) - \kappa Q(\omega) \right]_+ d\omega = \mathop{\mathbb{E}}_{\omega \sim Q} \left[\left[e^{\ell(\omega; P, Q)} - \kappa \right]_+ \right] = \mathop{\mathbb{E}}_{\omega \sim P} \left[\left[1 - \kappa e^{\ell(\omega; Q, P)} \right]_+ \right],$$

where $\ell\left(\omega; P, Q\right) \coloneqq \ln\left(\frac{P(\omega)}{Q(\omega)}\right)$; $\frac{P(\omega)}{Q(\omega)}$ is the ratio of the probabilities for countable domain or the Radon Nikodym derivative in the continuous case, and $\left[x\right]_+ \coloneqq \max\{0, x\}$. When P, Q are distributions induced by neighboring datasets $\boldsymbol{s} \simeq \boldsymbol{s}'$ and an algorithm M, we refer to the log probability ratio as the *privacy loss random variable* and denote it by $\ell_M\left(y; \boldsymbol{s}, \boldsymbol{s}'\right)$. We omit M from the notation when it is clear from the context.

Definition 2.2 ([Balle et al., 2018])**.** Given an algorithm $M : \mathcal{X}^* \to \mathcal{Y}$, the privacy profile $\delta_M : \mathbb{R} \to [0, 1]$ is defined to be the maximal hockey-stick divergence between the distributions induced by any neighboring datasets and past view. Formally, $\delta_M(\varepsilon) \coloneqq \sup_{\boldsymbol{s} \simeq \boldsymbol{s}' \in \mathcal{X}^*, \boldsymbol{v} \in \mathcal{Y}^*} \left(\boldsymbol{H}_{e^\varepsilon}\left(M(\boldsymbol{s}, \boldsymbol{v}) \parallel M(\boldsymbol{s}', \boldsymbol{v}) \right) \right)$.

Since the hockey-stick divergence is asymmetric in the general case, we use $\vec{\delta}_M$ and $\overset{\rightarrow}{\simeq}$ to denote the *remove* direction where $\perp \in \boldsymbol{s}'$ and $\overleftarrow{\delta}_M$, $\overset{\leftarrow}{\simeq}$ to denote the *add* direction when $\perp \in \boldsymbol{s}$. Notice that $\delta_M(\varepsilon) = \max\{\vec{\delta}_M(\varepsilon), \overleftarrow{\delta}_M(\varepsilon)\}$.

Another useful divergence notion is the *Rényi divergence*.

Definition 2.3. Given $\alpha > 1$ and two distributions P, Q over some domain Ω, the *Rényi divergence* between them is defined to be $\boldsymbol{R}_\alpha\left(P \| Q\right) \coloneqq \frac{1}{\alpha - 1} \ln\left(\mathop{\mathbb{E}}_{\omega \sim Q}\left[e^{\alpha \cdot \ell(\omega; P, Q)} \right] \right)$.

We can now formally define our privacy notions.

Definition 2.4 ([Dwork et al., 2006])**.** Given $\varepsilon > 0$; $\delta \in [0, 1]$, an algorithm M will be called (ε, δ)*-differentially private (DP)*, if $\delta_M(\varepsilon) \leq \delta$.

Definition 2.5 ([Mironov, 2017])**.** Given $\alpha \geq 1$; $\rho > 0$, an algorithm M will be called (α, ρ)*-Rényi differentially private (RDP)*, if $\sup_{\boldsymbol{s} \simeq \boldsymbol{s}' \in \mathcal{X}^*, \boldsymbol{v} \in \mathcal{Y}^*} \left(\boldsymbol{R}_\alpha\left(M(\boldsymbol{s}, \boldsymbol{v}) \| M(\boldsymbol{s}', \boldsymbol{v}) \right) \right) \leq \rho$.

One of the most common algorithms is the Gaussian mechanism N_σ, which simply reports the sum of (some function of) the elements in the dataset with the addition of Gaussian noise. One of its main advantages is that we have closed form expressions of its privacy (Lemma A.2).

2.3 Dominating pair of distributions

A key concept for characterizing the privacy guarantees of an algorithm is that of a *dominating pair* of distributions.

Definition 2.6 ([Zhu et al., 2022])**.** Given distributions P, Q over some domain Ω, and P', Q' over Ω', we say (P', Q') *dominate* (P, Q) if for all $\kappa \geq 0$ we have $\boldsymbol{H}_\kappa\left(P \parallel Q\right) \leq \boldsymbol{H}_\kappa\left(P' \parallel Q'\right)$. If $\delta_M(\varepsilon) \leq \boldsymbol{H}_{e^\varepsilon}\left(P \parallel Q\right)$ for all $\varepsilon \in \mathbb{R}$, we say (P, Q) is a *dominating pair* of distributions for M. If the inequality can be replaced by an equality for all ε, we say it is a *tightly dominating pair*. If there exist some $\boldsymbol{s} \simeq \boldsymbol{s}' \in \mathcal{X}^*$ such that $P = M(\boldsymbol{s})$, $Q = M(\boldsymbol{s}')$ we say $(\boldsymbol{s}, \boldsymbol{s}')$ are the dominating pair of datasets for M. By definition, a dominating pair of input datasets is tightly dominating.

We use the notion of dominating pair to define a dominating randomizer, which captures the privacy guarantees of the algorithm independently of its algorithmic adaptive properties.

Definition 2.7. Given a t-step algorithm defined by M, we say that $R : \{\perp, *\} \to \mathcal{Y}$ is a *dominating randomizer* for M and set $R(*) = P$ and $R(\perp) = Q$, where (P, Q) is the tightly dominating pair of $M(\cdot, \cdot)$ w.r.t. $\overset{\rightarrow}{\simeq}$ over all indexes $i \in [t]$ and input partial views $\boldsymbol{v}^{i-1}$.[3].

[3] Such a pair always exists [Zhu et al., 2022, Proposition 8]

The definition of the Poisson and random allocation schemes naturally extends to the case where the internal algorithm is a randomizer. In this case $\mathcal{P}_{t,\lambda}\left(R\right) : \{*, \perp\} \to \mathcal{Y}^t$ and $\mathcal{A}_{t,\lambda}\left(R\right) : \{*, \perp\} \to \mathcal{Y}^t$.

3 General reduction

We first prove two general claims which reduce the bound on arbitrary algorithms, datasets, and number of allocations, to the case of a single allocation ($k = 1$) of a simple non-adaptive randomizer receiving a single element. Missing proofs can be found in Appendix B

From the definition of the dominating randomizer, for any $\varepsilon \in \mathbb{R}$ we have $\delta_M(\varepsilon) \leq \delta_R(\varepsilon)$. We now prove that this is also the case for allocation scheme, that is $\delta_{\mathcal{A}_{t,k}(M)}(\varepsilon) \leq \delta_{\mathcal{A}_{t,k}(R)}(\varepsilon)$, and that the supremum over neighboring datasets for $\mathcal{A}_{t,k}\left(R\right)$ is achieved by the pair of datasets containing a single element, that is $\boldsymbol{s} = \{*\}$, $\boldsymbol{s}' = \{\perp\}$. This results from the fact that random allocation can be viewed as a two steps process, where first all elements but one are allocated, then the remaining one is allocated and the algorithm is ran for t steps. From the convexity of the hockey-stick divergence we can upper bound the privacy profile of the random allocation scheme by the worst case allocation of all elements but the removed one. From Lemma A.3, each intermediate call to the mechanism is a post process of the randomizer, which can be used to recursively define a randomized mapping from the random allocation over the randomizer to the allocation over the mechanism. Using the same lemma, this mapping implies that $\delta_{\mathcal{A}_{t,k}(M)}(\varepsilon) \leq \delta_{\mathcal{A}_{t,k}(R)}(\varepsilon)$.

Theorem 3.1. *Given $t \in \mathbb{N}$; $k \in [t]$ and a t-step algorithm defined by M dominated by a randomizer R, we have $\delta_{\mathcal{A}_{t,k}(M)}(\varepsilon) \leq \delta_{\mathcal{A}_{t,k}(R)}(\varepsilon)$.*

A special case of this result for Gaussian noise addition and $k = 1$ was given by Chua et al. [2024a, Theorem 1], and in the context of the matrix mechanism by Choquette-Choo et al. [2025, Lemma 3.2]. The same bound for the Poisson scheme is a direct result from the combination of Claim C.9 and Zhu et al. [2022, Theorem 11].

Next we show how to translate any bound on the privacy profile of the random allocation with $k = 1$ to the case of $k > 1$ by decomposing it to k calls to a 1 out of t/k steps allocation process.

Lemma 3.2. *For any $k \in \mathbb{N}$, $\varepsilon > 0$ we have $\delta_{\mathcal{A}_{t,k}(R)}(\varepsilon) \leq \delta^{\otimes k}_{\mathcal{A}_{\lfloor t/k \rfloor}(R)}(\varepsilon)$, where $\otimes k$ denotes the composition of k runs of the algorithm or scheme which in our case is $\mathcal{A}_{\lfloor t/k \rfloor}\left(R\right)$.*

Combining these two results, the privacy profile of the random allocation scheme is bounded by the (composition of the) hockey-stick divergence between $\mathcal{A}_t\left(R; *\right)$ and $\mathcal{A}_t\left(R; \perp\right) = R^{\otimes t}(\perp)$ in both directions, which we bound in three different ways in the following section.

4 Privacy bounds

4.1 Truncated Poisson bound

Roughly speaking, our main theorem states that random allocation is asymptotically identical to the Poisson scheme with sampling probability $\approx k/t$ up to lower order terms. Formal proofs and missing details of this section can be found in Appendix C.

Theorem 4.1. *Given $\varepsilon_0 > 0$; $\delta_0 \in [0, 1]$ and a $(\varepsilon_0, \delta_0)$-DP randomizer R, for any $\varepsilon, \delta > 0$ we have $\delta_{\mathcal{A}_t(R)}(\varepsilon) \leq \delta_{\mathcal{P}_{t,\eta}(R)}(\varepsilon) + t\delta_0 + \delta$, where $\eta := \frac{1}{t(1-\gamma)}$ and $\gamma := \min\left\{\cosh(\varepsilon_0) \cdot \sqrt{\frac{2}{t}\ln\left(\frac{1}{\delta}\right)}, 1 - \frac{1}{t}\right\}$.*

Furthermore, for any $\varepsilon, \varepsilon' > 0$ and randomizer R we have $\overrightarrow{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \leq \overrightarrow{\delta}_{\mathcal{P}_{t,\eta}(R)}(\varepsilon) + \tau \cdot \overleftarrow{\delta}_{\mathcal{A}_t(R)}\left(\varepsilon'\right)$ and $\overleftarrow{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \leq \overleftarrow{\delta}_{\mathcal{P}_{t,\eta}(R)}(\varepsilon) + \tau e^{2\varepsilon'} \cdot \overleftarrow{\delta}_{\mathcal{A}_t(R)}\left(\varepsilon'\right)$, where $\eta := \frac{e^{2\varepsilon'}}{t}$ and $\tau := \frac{1}{e^{\varepsilon'}(e^{\varepsilon'}-1)}$.

Since η corresponds to a sampling probability of $\frac{1}{t}$ up to a lower order term in t, this implies that the privacy of random allocation scheme is asymptotically upper-bounded by the Poisson scheme. While this holds for sufficiently large value of t, in many practical parameter regimes the second part of the theorem provides tighter bounds.

While the recursive expression might seem to lead to a vacuous loop, it is in fact a useful tool. Notice that $\overleftarrow{\delta}_{\mathcal{A}_t(R)}\left(\varepsilon'\right) / \overleftarrow{\delta}_{\mathcal{A}_t(R)}(\varepsilon)$ quickly diminishes as ε'/ε grows, so it suffices to set $\varepsilon' = C\varepsilon$

for some constant $1 \ll C$ for the second term to become negligible. Both parts of this theorem follow from Lemma C.1 which bounds the privacy profile of the random allocation scheme by that of the corresponding Poisson scheme with sampling probability η, with an additional term roughly corresponding to a tail bound on the privacy loss of the allocation scheme.

4.2 Asymptotic analysis

Combining Theorem 4.1 with Lemma 3.2 and applying it to the Gaussian mechanism results in the next corollary.

Corollary 4.2. *Given $\varepsilon, \delta > 0$, for any $\sigma > 8 \cdot \max\left\{\sqrt{\ln(t/\delta)}, \sqrt{\frac{k}{t}}\ln(t/\delta)\right\}$, we have $\delta_{\mathcal{A}_{t,k}(N_\sigma)}(\varepsilon) \leq \delta_{\mathcal{P}_{t,2k/t}(N_\sigma)}(\varepsilon) + 2\delta$, where N_σ is the Gaussian mechanism.*

Using this Corollary we can derive asymptotic bounds on the privacy guarantees of the Gaussian mechanism amplified by random allocation. Since the Gaussian mechanism is dominated by the one-dimensional Gaussian randomizer (Claim D.3) where $R(*) = \mathcal{N}(1, \sigma^2)$ and $R(\perp) = \mathcal{N}(0, \sigma^2)$, this corollary implies that for sufficiently large σ, the random allocation scheme with the Gaussian mechanism $\mathcal{A}_{t,k}(N_\sigma)$ is (ε, δ)-DP for any $\varepsilon > C \cdot \max\left\{\frac{k\sqrt{\ln(t/\delta)}}{\sigma\sqrt{t}}, \frac{k^2\sqrt{\ln(t/\delta)}}{t^{1.5}}\right\}$ for some universal constant C (Lemma D.2). We note that the dependence of ε on σ; δ; k; and t matches that of the Poisson scheme for $\lambda = k/t$ (Lemma D.1) up to an additional logarithmic dependence on t (Poisson scales with $\ln(1/\delta)$), unlike the shuffle scheme which acquires an additional $\sqrt{\ln(1/\delta)}$ by converting approximate the DP randomizer to pure DP first, resulting in the bound $\varepsilon \geq C' \cdot \frac{k \cdot \ln(1/\delta)}{\sigma\sqrt{t}}$ [Feldman et al., 2021]. A detailed comparison can be found in Appendix D.

The recursive bound (second part of Theorem 4.1) provides similar asymptotic guarantees for arbitrary mechanisms, when $\varepsilon_0 \leq 1$ for the local mechanism M. In this case, the privacy parameter of its corresponding Poisson scheme $\varepsilon_{\mathcal{P}}$ is approximately linear in the sampling probability. Setting the sampling probability to $1/t$ and combining amplification by subsampling with advanced composition implies $\varepsilon_{\mathcal{P}} = O\left(\eta\varepsilon_0\sqrt{t \cdot \ln(1/\delta)}\right) = O\left(\varepsilon_0\sqrt{\frac{\ln(1/\delta)}{t}}\right)$. Setting $\varepsilon' = C\varepsilon_{\mathcal{P}}$ for some constant $1 \ll C < 1/\varepsilon_{\mathcal{P}}$, we get $\varepsilon_{\mathcal{A}} = O\left(\eta\varepsilon_0\sqrt{t \cdot \ln(1/\delta)}\right) = O\left((1 + \varepsilon_{\mathcal{P}})\varepsilon_0\sqrt{\frac{\ln(1/\delta)}{t}}\right) = O(\varepsilon_{\mathcal{P}} + \varepsilon_{\mathcal{P}}^2)$.

While Theorem 4.1 provides a full asymptotic characterization of the random allocation scheme, the bounds it induces could be suboptimal for small t or large ε_0. In the following section we provide several bounds that hold in all parameter regimes. We also use these to "bootstrap" the recursive bound.

4.3 Poisson Decomposition

Theorem 4.3. *For any $\varepsilon > 0$ we have $\overrightarrow{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \leq \vec{\gamma} \cdot \overrightarrow{\delta}_{\mathcal{P}_{t,\lambda}(R)}(\vec{\varepsilon})$ and $\overleftarrow{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \leq \overleftarrow{\gamma} \cdot \overleftarrow{\delta}_{\mathcal{P}_{t,\lambda}(R)}(\overleftarrow{\varepsilon})$, where $\vec{\gamma} := \frac{1}{1-(1-\lambda)^t}$, $\overleftarrow{\gamma} := 1 + e^\varepsilon(\vec{\gamma} - 1)$, $\vec{\varepsilon} := \ln(1 + (e^\varepsilon - 1)/\vec{\gamma})$, and $\overleftarrow{\varepsilon} := -\ln\left(1 - (1 - e^{-\varepsilon})/\vec{\gamma}\right)$.*

We remark that while this theorem provides separate bounds for the add and remove adjacency notions[4], numerical analysis seems to indicate that the bound on the remove direction is always larger than the one for the add direction.

Setting $\lambda := 1/t$ yields $\vec{\gamma} \approx e/(e-1) \approx 1.6$, which bounds the difference between these two sampling methods up to this factor in ε in the $\varepsilon < 1$ regime.

Formal proofs and missing details can be found in Appendix E.

4.4 Direct analysis

The previous bounds rely on a reduction to Poisson scheme. In this section we bound the privacy profile of the random allocation scheme directly, which is especially useful in the low privacy regime

[4] An earlier version of this work has mistakenly stated that an upper bound for the remove direction applies to both directions.

where the privacy profile of random allocation is lower than that of Poisson. Formal proofs and missing details of this section can be found in Appendix F.

Our main result expresses the RDP of the random allocation scheme in the remove direction in terms of the RDP parameters of the randomizer, and provides an approximate bound in the the add direction[5]. While the privacy bounds induced by RDP are typically looser than those relying on full analysis and composition of the privacy loss distribution (PLD), the gap nearly vanishes as the number of composed calls to the randomizer grows, as depicted in Figure 6.

Theorem 4.4. *Given two integers $t, \alpha \in \mathbb{N}$, we denote by $\mathbf{\Pi}_t(\alpha)$ the set of integer partitions of α consisting of $\leq t$ elements. Given a partition $\Pi \in \mathbf{\Pi}_t(\alpha)$, we denote by $\binom{\alpha}{\Pi} = \frac{\alpha!}{\prod_{p \in \Pi} p!}$, and denote by $C(\Pi)$ the list of counts of unique values in P (e.g. if $\alpha = 9$ and $\Pi = [1, 2, 3, 3]$ then $C(\Pi) = [1, 1, 2]$). For any $\alpha \geq 2$ and randomizer R we have*

$$\boldsymbol{R}_\alpha \left(\mathcal{A}_t \left(R; * \right) \| \mathcal{A}_t \left(R; \perp \right) \right) = \frac{1}{\alpha - 1} \ln \left(\frac{1}{t^\alpha} \sum_{\Pi \in \mathbf{\Pi}_t(\alpha)} \binom{t}{C(\Pi)} \binom{\alpha}{\Pi} \prod_{p \in \Pi} e^{(p-1) \boldsymbol{R}_p(R(*) \| R(\perp))} \right).$$

For the add direction we use a different bound.

Theorem 4.5. *Given $\gamma \in [0, 1]$ and a randomizer R, we define a new randomizer R_γ which given an input x samples $y \propto \frac{P_R(y|x)^\gamma \cdot P_R(y|\perp)^{1-\gamma}}{Z_\gamma}$, where Z_γ is the normalizing factor.*

For any $\varepsilon > 0$ we have $\overleftarrow{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \leq \boldsymbol{H}_{e^{\varepsilon'}} \left(R^{\otimes t}(\perp) \middle\| R_{1/t}^{\otimes t}(*) \right)$, where $\varepsilon' := \varepsilon - t \cdot \ln(Z_{1/t})$.

These two theorems follow from Lemmas F.1 and F.3 for $P = R(*)$ and $Q = R(\perp)$.

As is the case in Theorem 4.3, numerical analysis seems to indicate the bound on the remove direction always dominates the one for the add direction.

Since we have an exact expression for the hockey-stick and Rényi divergences of the Gaussian mechanism, these two theorems immediately imply the following corollary.

Corollary 4.6. *Given $\sigma > 0$, and a Gaussian mechanism N_σ, we have for any integer $\alpha \geq 2$*

$$\boldsymbol{R}_\alpha \left(\mathcal{A}_t \left(N_\sigma; 1 \right) \| N_\sigma^{\otimes t}(0) \right) = \frac{1}{\alpha - 1} \left(\ln \left(\sum_{\Pi \in \mathbf{\Pi}_t(\alpha)} \binom{t}{C(\Pi)} \binom{\alpha}{\Pi} e^{\sum_{p \in \Pi} \frac{p^2}{2\sigma^2}} \right) - \alpha \left(\frac{1}{2\sigma^2} + \ln(t) \right) \right),$$

and $\overleftarrow{\delta}_{\mathcal{A}_t(N_\sigma)}(\varepsilon) \leq \delta_{N_{\sigma'}}(\varepsilon')$ where $\sigma' := \sqrt{t}\sigma$ and $\varepsilon' := \varepsilon - \frac{1-1/t}{2\sigma^2}$ for all $\varepsilon \in \mathbb{R}$.

Corollary 4.6 gives a simple way to exactly compute integer RDP parameters of random allocation with Gaussian noise in the remove direction. Interestingly, they closely match RDP parameters of the Poisson scheme with rate $1/t$ in most regimes (e.g. Fig. 6). In fact, in some (primarily large ε) parameter regimes the bounds based on RDP of allocation are lower than the PLD-based bounds for Poisson subsampling (Fig. 5). The restriction to integer values has negligible effect, which can be further mitigated using [Wang et al., 2019, Corollary 10], which upper bounds the fractional Rényi divergence by a linear combination of the Rényi divergence of its rounded integer values. We also note that $|\mathbf{\Pi}_t(\alpha)|$ is sub-exponential in α which leads to performance issues in the very high privacy ($\varepsilon \ll 1$) regime (Large σ values in Fig 1). Since the typical value of α used for accounting is in the low tens, this quantity can be efficiently computed using several technical improvements which we discuss in Appendix F.1. On the other hand, in the very low privacy regime ($\varepsilon \gg 1$), the α that leads to the best bound on ε is typically in the range $[1, 2]$ which cannot be computed exactly using our method. Finally, we remark that while this result is stated only for $k = 1$, it can be extended to $k > 1$ using the same argument as in Lemma 3.2. In fact, RDP based bounds are particularly convenient for subsequent composition which is necessary to obtain bounds for $k > 1$ or multi-epoch training algorithms.

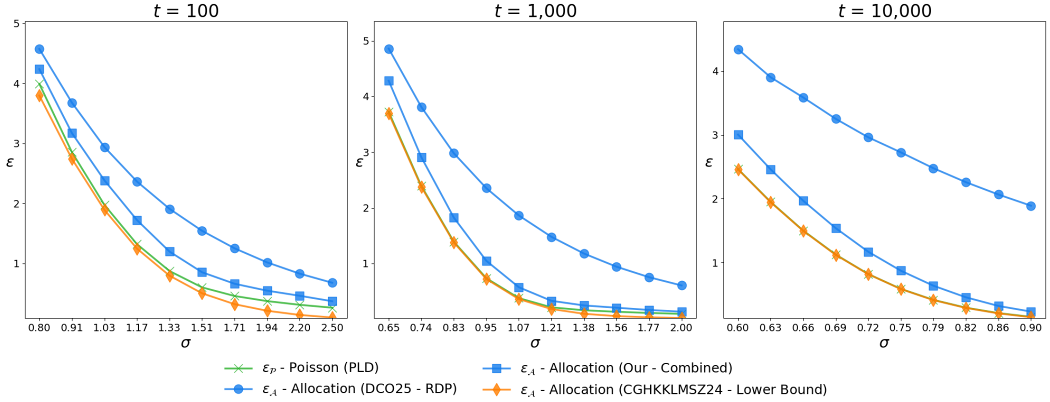


Figure 2: Bounds on privacy parameter ε as a function of the noise parameter σ for various values of t, all using the Gaussian mechanism with $\delta = 10^{-10}$. We compare the minimum over all our methods to the independent results in Dong et al. [2025], lower bound by Chua et al. [2024a], and to the Poisson scheme with $\lambda = 1/t$.

5 Numerical evaluation

In this section we demonstrate that numerical implementations of our results give the first nearly-tight and provable bounds on privacy amplification of random allocation with Gaussian noise, notably showing (Fig. 1, 2) that they nearly match bounds known for Poisson subsampling. Compared to the Monte Carlo-based technique by Chua et al. [2024a] (G.1), we show in many regimes our results match these bounds up to constants in δ (logarithmic in ε), and the computational limitation of the MC technique in the low δ and high confidence level regime. We additionally compare our results to the RDP-based approximation by Dong et al. [2025] (G.2), and demonstrate the advantage of our tight RDP analysis in the regime where $k \ll t$. Their bound is tighter than our direct analysis in the $k \approx t$ regime, where the effect of amplification is small and ε is prohibitively large.

We demonstrate the utility degradation induced by Poisson subsampling relative to random allocation using the simple setting of estimating the mean of a Bernoulli distribution from a sampled dataset (App. H). We derive theoretical approximations for the mean square error of the two schemes and match them with numerical simulations, that demonstrate random allocation always has lower error for sufficiently large sample size. Together with the results of Chua et al. [2024a], our results imply that random allocation (or balls-and-bins sampling) has the utility benefits of shuffling while having the privacy benefits of Poisson subsampling. This provides a (reasonably) practical way to reconcile a long-standing and concerning discrepancy between the practical implementations of DP-SGD and its commonly-used privacy analyses.

6 Discussion

This work provides the first theoretical guarantees and numerical estimation algorithms for the random allocation sampling scheme. Its main analysis shows that its privacy guarantees are asymptotically identical to those of the Poisson scheme. We provide two additional analyses which lead to tighter bounds in some setting (Fig. 1). The resulting combined bound of the random allocation remains close to that of the Poisson scheme in many practical regimes (Fig. 2, 9), and even exceeds it in some. Unlike the Poisson scheme, our bounds are analytical and do not rely on numerical PLD analysis, which results in some remaining slackness. Further, unlike PLD-based bounds, our (ε, δ) bounds do not lend themselves for tight privacy accounting of composition. Both of these limitations are addressed in our subsequent work [Feldman and Shenfeld, 2025] where we show that PLD of random allocation can be approximated efficiently, leading to tighter and more general numerical bounds.

[5] An earlier version of this work has mistakenly stated that an upper bound for the remove direction applies to both directions.

Acknowledgments and Disclosure of Funding

We are grateful to Kunal Talwar for proposing the problem that we investigate here as well as suggesting the decomposition-based approach for the analysis of this problem (established in Theorem 4.3). We thank Thomas Steinke and Christian Lebeda for pointing out that an earlier version of our work mistakenly did not analyze the add direction of our direct bound. We also thank Hilal Asi, Hannah Keller, Guy Rothblum and Katrina Ligett for thoughtful comments and motivating discussions of these results and their application in [Asi et al., 2025]. Shenfeld's work was supported by the Apple Scholars in AI/ML PhD Fellowship.

References

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*, pages 308–318, 2016.

Meenatchi Sundaram Muthu Selva Annamalai, Borja Balle, Emiliano De Cristofaro, and Jamie Hayes. To shuffle or not to shuffle: Auditing dp-sgd with shuffling. *arXiv preprint arXiv:2411.10614*, 2024.

Hilal Asi, Vitaly Feldman, Hannah Keller, Guy N. Rothblum, and Kunal Talwar. PREAMBLE: Private and efficient aggregation of block sparse vectors and applications. Cryptology ePrint Archive, Paper 2025/490, 2025.

Borja Balle and Yu-Xiang Wang. Improving the gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *International Conference on Machine Learning*, pages 394–403. PMLR, 2018.

Borja Balle, Gilles Barthe, and Marco Gaboardi. Privacy amplification by subsampling: Tight analyses via couplings and divergences. *Advances in neural information processing systems*, 31, 2018.

Borja Balle, James Bell, Adrià Gascón, and Kobbi Nissim. The privacy blanket of the shuffle model. In *Advances in Cryptology–CRYPTO 2019: 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Proceedings, Part II 39*, pages 638–667. Springer, 2019.

Borja Balle, Peter Kairouz, Brendan McMahan, Om Thakkar, and Abhradeep Guha Thakurta. Privacy amplification via random check-ins. *Advances in Neural Information Processing Systems*, 33: 4623–4634, 2020.

Raef Bassily, Adam Smith, and Abhradeep Thakurta. Private empirical risk minimization: Efficient algorithms and tight error bounds. In *2014 IEEE 55th annual symposium on foundations of computer science*, pages 464–473. IEEE, 2014.

Andrea Bittau, Úlfar Erlingsson, Petros Maniatis, Ilya Mironov, Ananth Raghunathan, David Lie, Mitch Rudominer, Ushasree Kode, Julien Tinnes, and Bernhard Seefeld. Prochlo: Strong privacy for analytics in the crowd. In *Proceedings of the 26th symposium on operating systems principles*, pages 441–459, 2017.

Clément L Canonne, Gautam Kamath, and Thomas Steinke. The discrete gaussian for differential privacy. *Advances in Neural Information Processing Systems*, 33:15676–15688, 2020.

Wei-Ning Chen, Dan Song, Ayfer Ozgur, and Peter Kairouz. Privacy amplification via compression: Achieving the optimal privacy-accuracy-communication trade-off in distributed mean estimation. *Advances in Neural Information Processing Systems*, 36, 2024.

Albert Cheu, Adam Smith, Jonathan Ullman, David Zeber, and Maxim Zhilyaev. Distributed differential privacy via shuffling. In *Advances in Cryptology–EUROCRYPT 2019: 38th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Darmstadt, Germany, May 19–23, 2019, Proceedings, Part I 38*, pages 375–403. Springer, 2019.

Christopher Choquette-Choo, Arun Ganesh, Saminul Haque, Thomas Steinke, and Abhradeep Guha Thakurta. Near-exact privacy amplification for matrix mechanisms. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu, editors, *International Conference on Representation Learning*, volume 2025, pages 98772–98802, 2025.

Christopher A Choquette-Choo, Arun Ganesh, Thomas Steinke, and Abhradeep Guha Thakurta. Privacy amplification for matrix mechanisms. In *The Twelfth International Conference on Learning Representations*, 2023.

Lynn Chua, Badih Ghazi, Charlie Harrison, Pritish Kamath, Ravi Kumar, Ethan Jacob Leeman, Pasin Manurangsi, Amer Sinha, and Chiyuan Zhang. Balls-and-bins sampling for dp-sgd. In *The 28th International Conference on Artificial Intelligence and Statistics*, 2024a.

Lynn Chua, Badih Ghazi, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Amer Sinha, and Chiyuan Zhang. How private are dp-sgd implementations? In *Forty-first International Conference on Machine Learning*, 2024b.

Lynn Chua, Badih Ghazi, Pritish Kamath, Ravi Kumar, Pasin Manurangsi, Amer Sinha, and Chiyuan Zhang. Scalable dp-sgd: Shuffling vs. poisson subsampling. *Advances in Neural Information Processing Systems*, 37:70026–70047, 2024c.

Andy Dong, Wei-Ning Chen, and Ayfer Ozgur. Leveraging randomness in model and data partitioning for privacy amplification. In *Forty-second International Conference on Machine Learning*, 2025.

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

Cynthia Dwork, Krishnaram Kenthapadi, Frank McSherry, Ilya Mironov, and Moni Naor. Our data, ourselves: Privacy via distributed noise generation. In *Advances in Cryptology-EUROCRYPT 2006: 24th Annual International Conference on the Theory and Applications of Cryptographic Techniques, St. Petersburg, Russia, May 28-June 1, 2006. Proceedings 25*, pages 486–503. Springer, 2006.

Úlfar Erlingsson, Vitaly Feldman, Ilya Mironov, Ananth Raghunathan, Kunal Talwar, and Abhradeep Thakurta. Amplification by shuffling: From local to central differential privacy via anonymity. In *Proceedings of the Thirtieth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 2468–2479. SIAM, 2019.

Vitaly Feldman and Moshe Shenfeld. Efficient computation of the privacy loss distribution for random allocation, 2025. URL `https://openreview.net/forum?id=DuFNAlQ8Lw`.

Vitaly Feldman, Audra McMillan, and Kunal Talwar. Hiding among the clones: A simple and nearly optimal analysis of privacy amplification by shuffling. In *2021 IEEE 62nd Annual Symposium on Foundations of Computer Science (FOCS)*, pages 954–964. IEEE, 2021.

Vitaly Feldman, Audra McMillan, and Kunal Talwar. Stronger privacy amplification by shuffling for rényi and approximate differential privacy. In *Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA)*, pages 4966–4981. SIAM, 2023.

Antonious M. Girgis, Deepesh Data, Suhas Diggavi, Peter Kairouz, and Ananda Theertha Suresh. Shuffled model of federated learning: Privacy, accuracy and communication trade-offs. *IEEE Journal on Selected Areas in Information Theory*, 2(1):464–478, 2021.

Peter Kairouz, Sewoong Oh, and Pramod Viswanath. The composition theorem for differential privacy. In *International conference on machine learning*, pages 1376–1385. PMLR, 2015.

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

Shiva Prasad Kasiviswanathan, Homin K Lee, Kobbi Nissim, Sofya Raskhodnikova, and Adam Smith. What can we learn privately? *SIAM Journal on Computing*, 40(3):793–826, 2011.

Antti Koskela, Mikko A Heikkilä, and Antti Honkela. Numerical accounting in the shuffle model of differential privacy. *Transactions on Machine Learning Research*, 2022.

Seng Pei Liew and Tsubasa Takahashi. Shuffle gaussian mechanism for differential privacy. *arXiv preprint arXiv:2206.09569*, 2022.

Xin Lyu. Composition theorems for interactive differential privacy. *Advances in Neural Information Processing Systems*, 35:9700–9712, 2022.

Ilya Mironov. Rényi differential privacy. In *2017 IEEE 30th computer security foundations symposium (CSF)*, pages 263–275. IEEE, 2017.

Mehta Neelesh B., Wu Jingxian, Molisch Andreas F., and Zhang Jin. Approximating a sum of random variables with a lognormal. *Transactions on Wireless Communications*, 6(7):2690–2699, 2007.

Thomas Steinke. Composition of differential privacy & privacy amplification by subsampling. *arXiv preprint arXiv:2210.00597*, 2022.

Salil Vadhan and Tianhao Wang. Concurrent composition of differential privacy. In *Theory of Cryptography: 19th International Conference, TCC 2021, Raleigh, NC, USA, November 8–11, 2021, Proceedings, Part II 19*, pages 582–604. Springer, 2021.

Salil Vadhan and Wanrong Zhang. Concurrent composition theorems for differential privacy. In *Proceedings of the 55th Annual ACM Symposium on Theory of Computing*, pages 507–519, 2023.

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

Yuqing Zhu, Jinshuo Dong, and Yu-Xiang Wang. Optimal accounting of differential privacy via characteristic function. In *International Conference on Artificial Intelligence and Statistics*, pages 4782–4817. PMLR, 2022.

A Missing definitions and claims

Since Rényi divergence is effectively a bound on the moment generating function, it can be used to bound the hockey-stick divergence which is effectively a tail bound.

Lemma A.1 (Rényi bounds Hockey-stick, Prop. 12 in Canonne et al. [2020])**.** *Given two distributions P, Q, if $\boldsymbol{R}_\alpha\left(P \| Q\right) \leq \rho$ then $\boldsymbol{H}_{e^\varepsilon}\left(P \;\|\; Q\right) \leq \frac{1}{\alpha-1} e^{(\alpha-1)(\rho-\varepsilon)}\left(1-\frac{1}{\alpha}\right)^{\alpha}$.*

.

Lemma A.2 (Gaussian mechanism DP guarantees, [Balle and Wang, 2018, Mironov, 2017])**.** *Given $d \in \mathbb{N}$; $\sigma > 0$, let $\mathcal{X} = \mathcal{Y} := \mathbb{R}^d$. The* Gaussian mechanism *N_σ is defined as $N_\sigma(\boldsymbol{s}) := \mathcal{N}(\sum_{x \in \boldsymbol{s}} x, \sigma^2 I_d)$.*

If the domain of N_σ is the unit ball in $\mathbb{R}^d$, we have $\delta_{N_\sigma}(\varepsilon) = \Phi\left(\frac{1}{2\sigma} - \varepsilon\sigma\right) - e^{\varepsilon}\Phi\left(-\frac{1}{2\sigma} - \varepsilon\sigma\right)$, where Φ is the CDF of the standard Normal distribution, and for any $\alpha \geq 1$ N_σ is $(\alpha, \alpha/(2\sigma^2)$-RDP.

An important property of domination is its equivalence to existence of postprocessing.

Lemma A.3 (Post processing, Thm. II.5 [Kairouz et al., 2015])**.** *Given distributions P, Q over some domain Ω, and P', Q' over Ω', (P, Q) dominate (P', Q') if and only if there exists a randomized function $\varphi : \Omega \to \Omega'$ such that $P' = \varphi(P)$ and $Q' = \varphi(Q)$.*

B Missing proofs from Section 3

Proof of Lemma 3.1. Given $n \in \mathbb{N}$, a dataset $\boldsymbol{s} \in \mathcal{X}^{n-1}$ and element $x \in \mathcal{X}$, the random allocation scheme $\mathcal{A}_{t,k}\left(M; (\boldsymbol{s}, x)\right)$ can be decomposed into two steps. First all elements in $\boldsymbol{s}$ are allocated, then x is allocated and the outputs are sampled based on the allocations. We denote by $\boldsymbol{a}^{t,k}(n-1)$ the set of all possible allocations of $n-1$ elements into k out of t steps, and for any $a \in \boldsymbol{a}^{t,k}(n-1)$ denote by $\mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, x))$ the allocation scheme conditioned on the allocation of $\boldsymbol{s}$ according to a. Given the neighboring datasets $(\boldsymbol{s}, x)$ and $(\boldsymbol{s}, \perp)$ we have,

$$
\begin{aligned}
&\boldsymbol{H}_\kappa\left(\mathcal{A}_{t,k}\left(M; (\boldsymbol{s}, x)\right) \| \mathcal{A}_{t,k}\left(M; (\boldsymbol{s}, \perp)\right)\right) \\
&\quad = \boldsymbol{H}_\kappa\left(\frac{1}{|\boldsymbol{a}^{t,k}(n-1)|} \sum_{a \in \boldsymbol{a}^{t,k}(n-1)} \mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, x)) \;\middle\|\; \frac{1}{|\boldsymbol{a}^{t,k}(n-1)|} \sum_{a \in \boldsymbol{a}^{t,k}(n-1)} \mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, \perp))\right) \\
&\quad \leq \max_{a \in \boldsymbol{a}^{t,k}(n-1)} \left(\boldsymbol{H}_\kappa\left(\mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, x)) \;\middle\|\; \mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, \perp))\right)\right),
\end{aligned}
$$

and similarly,

$$
\boldsymbol{H}_\kappa\left(\mathcal{A}_{t,k}\left(M; (\boldsymbol{s}, \perp)\right) \;\|\; \mathcal{A}_{t,k}\left(M; (\boldsymbol{s}, x)\right)\right) \leq \max_{a \in \boldsymbol{a}^{t,k}(n-1)} \left(\boldsymbol{H}_\kappa\left(\mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, \perp)) \;\middle\|\; \mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, x))\right)\right),
$$

where the inequality results from the quasi-convexity of the hockey-stick divergence.

Fixing a to be the allocation that maximizes the right-hand side of the inequality, we denote by $\boldsymbol{s}_1, \ldots, \boldsymbol{s}_t$ the subsets defined by a.[6] From the definition of the randomizer, for any index $j \in [t]$ and input prefix view $\boldsymbol{v}^{j-1}$ we have $(R(*), R(\perp))$ dominates $(M((\boldsymbol{s}^j, x), \boldsymbol{v}^{j-1}), M((\boldsymbol{s}^j, \perp), \boldsymbol{v}^{j-1}))$, so from Lemma A.3, there exists a randomized mapping $\varphi_{\boldsymbol{v}^{j-1}}$ such that $M((\boldsymbol{s}^j, x), \boldsymbol{v}^{j-1}) = \varphi_{\boldsymbol{v}^{j-1}}(R(*))$ and $M((\boldsymbol{s}^j, \perp), \boldsymbol{v}^{j-1}) = \varphi_{\boldsymbol{v}^{j-1}}(R(\perp))$.[7] Using these mappings, we will recursively define another randomized mapping φ. Given an output view $\boldsymbol{v} \in \mathcal{Y}^t$, we define $\boldsymbol{v}' \sim \varphi(\boldsymbol{v})$ by sequentially sampling $y'_j \sim \varphi_{\boldsymbol{v}'^{j-1}}(y_j)$ for $j = 1, \ldots, t$, where $\boldsymbol{v}'^j := (y_1, \ldots, y_j)$ and $\boldsymbol{v}'^0 := \emptyset$.

We will now prove that $\mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, x)) = \varphi(\mathcal{A}_{t,k}\left(R; *\right))$ and $\mathcal{A}^a_{t,k}(M; (\boldsymbol{s}, \perp)) = \varphi(\mathcal{A}_{t,k}\left(R; \perp\right))$, which by invoking Lemma A.3 again, implies $\mathcal{A}_{t,k}\left(M\right)$ is dominated by $\mathcal{A}_{t,k}\left(R\right)$ and completes the proof.

[6] Conceptually, this is equivalent to considering the random allocation scheme over a single element x, with a sequence of mechanisms $M_{\boldsymbol{s}^j}$ defined by the various subsets.

[7] We note that φ depends on $\boldsymbol{s}^j$ and x as well. We omit them from notations for simplicity, since they are fixed at this point of the argument.

From the law of total probability we have,

$$P_{\mathcal{A}^a_{t,k}(M;(\boldsymbol{s},x))}(\boldsymbol{v}) = \frac{1}{\binom{t}{k}} \sum_{\boldsymbol{i}\subseteq[t],|\boldsymbol{i}|=k} P_{\mathcal{A}^a_{t,k}(M;(\boldsymbol{s},x))}(\boldsymbol{v}|\boldsymbol{i}) = \frac{1}{\binom{t}{k}} \sum_{\boldsymbol{i}\subseteq[t],|\boldsymbol{i}|=k} \prod_{j\in[t]} P_{\mathcal{A}^a_{t,k}(M;(\boldsymbol{s},x))}(\boldsymbol{v}^j|\boldsymbol{v}^{j-1},\boldsymbol{i})$$

and

$$P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}) = \frac{1}{\binom{t}{k}} \sum_{\boldsymbol{i}\subseteq[t],|\boldsymbol{i}|=k} P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}|\boldsymbol{i}) = \frac{1}{\binom{t}{k}} \sum_{\boldsymbol{i}\subseteq[t],|\boldsymbol{i}|=k} \prod_{j\in[t]} P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^j|\boldsymbol{v}^{j-1},\boldsymbol{i}),$$

where $\boldsymbol{i} \subseteq [t]$ denots the allocation of the single element x.

Using these identities, it suffices to prove that for any subset of indexes $\boldsymbol{i}$, index $j \in [t]$, and input prefix view $\boldsymbol{v}^{j-1}$, we have $P_{\mathcal{A}^a_{t,k}(M;(\boldsymbol{s},x))}(\boldsymbol{v}^j|\boldsymbol{v}^{j-1},\boldsymbol{i}) = P_{\varphi(\mathcal{A}_{t,k}(R;*))}(\boldsymbol{v}^j|\boldsymbol{v}^{j-1},\boldsymbol{i})$ and $P_{\mathcal{A}^a_{t,k}(M;(\boldsymbol{s},\perp))}(\boldsymbol{v}^j|\boldsymbol{v}^{j-1},\boldsymbol{i}) = P_{\varphi(\mathcal{A}_{t,k}(R;\perp))}(\boldsymbol{v}^j|\boldsymbol{v}^{j-1},\boldsymbol{i})$.

From the definition,

$$\begin{aligned} P_{\mathcal{A}^a_{t,k}(M;(\boldsymbol{s},x))}(\boldsymbol{v}^j|\boldsymbol{v}^{j-1},\boldsymbol{i}) &= P_{\mathcal{A}^a_{t,k}(M;(\boldsymbol{s},x))}(y_j \mid \boldsymbol{v}^{j-1},\boldsymbol{i}) \\ &= \begin{cases} P_{M((\boldsymbol{s}^j,x),\boldsymbol{v}^{j-1})}(y_j) & j \in \boldsymbol{i} \\ P_{M((\boldsymbol{s}^j,\perp),\boldsymbol{v}^{j-1})}(y_j) & j \notin \boldsymbol{i} \end{cases} \\ &= \begin{cases} P_{\varphi_{\boldsymbol{v}^{j-1}}(R(*))}(y_j) & j \in \boldsymbol{i} \\ P_{\varphi_{\boldsymbol{v}^{j-1}}(R(\perp))}(y_j) & j \notin \boldsymbol{i} \end{cases} \\ &= P_{\varphi(\mathcal{A}_{t,k}(R;*))}(y_j \mid \boldsymbol{v}^{j-1},\boldsymbol{i}) \\ &= P_{\varphi(\mathcal{A}_{t,k}(R;*))}(\boldsymbol{v}^j \mid \boldsymbol{v}^{j-1},\boldsymbol{i}). \end{aligned}$$

In the case of the null element, the allocation doesn't have any effect so we have,

$$P_{\mathcal{A}^a_{t,k}(M;(\boldsymbol{s},\perp))}(\boldsymbol{v}^j|\boldsymbol{v}^{j-1}) = P_{M((\boldsymbol{s}^j,\perp),\boldsymbol{v}^{j-1})}(y_j) = P_{\varphi_{\boldsymbol{v}^{j-1}}(R(\perp))}(y_j) = P_{\varphi(\mathcal{A}_{t,k}(R;\perp))}(\boldsymbol{v}^j \mid \boldsymbol{v}^{j-1})$$

which completes the proof. □

Proof. [Proof of Lemma 3.2] Notice that the random allocation of k indexes out of t can be described as a two steps process, first randomly splitting t into k subsets of size t/k, [8] then running $\mathcal{A}_{t/k,1}(R)$ on each of the k copies of the scheme. Using the same convexity argument as in the proof of Lemma 3.1, the privacy profile of $\mathcal{A}_{t,k}(R)$ is upper bounded by the composition of k copies of $\mathcal{A}_{t/k,1}(R)$. □

We remark that this lemma holds for arbitrary t-step algorithms (and not just non-adaptive randomizers) but in the adaptive case the usual sequential composition should be replaced by concurrent composition [Vadhan and Wang, 2021], which was recently proven to provide the same privacy guarantees [Lyu, 2022, Vadhan and Zhang, 2023].

C Missing proofs from Section 4.1

Lemma C.1 (Analytic bound)**.** *For any $\varepsilon > 0$ and $\eta \in [1/t, 1]$ we have $\vec{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \leq \vec{\delta}_{\mathcal{P}_{t,\eta}(R)}(\varepsilon) + \vec{\beta}_{\mathcal{A}_t(R)}(\eta)$ and $\overleftarrow{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \leq \overleftarrow{\delta}_{\mathcal{P}_{t,\eta}(R)}(\varepsilon) + \overleftarrow{\beta}_{\mathcal{A}_t(R)}(\eta)$, where $\vec{\beta}_{\mathcal{A}_t(R)}(\eta) := P_{\mathcal{A}_t(R;*)}(B(\eta))$, $\overleftarrow{\beta}_{\mathcal{A}_t(R)}(\eta) := P_{\mathcal{A}_t(R;\perp)}(B(\eta))$, and $B(\eta) := \left\{\boldsymbol{v} \in \mathcal{Y}^{t-1} \;\middle|\; \max_{i\in[t-1]} \left(\ell\left(\boldsymbol{v}^i; \perp, *\right)\right) > \ln(t\eta)\right\}$.*

Theorem 4.1 follows from this lemma by identifying the total loss with a sum of the independent losses per step and using maximal Azuma-Hoeffding inequality. Theorem4.1 follows from this lemma by using a simple relationship between the privacy loss tail bound and the privacy profile.

The proof of this lemma consists of a sequences of reductions.

[8] For simplicity we assume that t is divisible by k.

Proof. Following [Erlingsson et al., 2019], we introduce the posterior sampling scheme (Definition C.3), where the sampling probability depends on the previous outputs. Rather than selecting in advance a single step, at each step the scheme chooses to include the element with posterior probability induced by the previously released outputs $\lambda_{\boldsymbol{v}^i} := P_{\mathcal{A}_t(R;*)}\left(i+1 \in \boldsymbol{I} | \boldsymbol{v}^i\right)$, where $\boldsymbol{I}$ is the subset of chosen steps.

Though this scheme seems like a variation of the Poisson scheme, we prove (Lemma C.4) that in fact its output is distributed like the output of random allocation, which implies they share the same privacy guarantees. The crucial difference between these two schemes is the fact that unlike random allocation, the distribution over the outputs of any step of the posterior scheme is independent of the distribution over output of previous steps given the view and the dataset, since there is no shared randomness (such as the chosen allocation).

We then define a truncated variant of the posterior distribution (Definition C.5), where the sampling probability is capped by some threshold, and bound the difference between the privacy profile of the truncated and original posterior distributions, by the probability that the posterior sampling probability will exceed the truncation threshold (Lemma C.6).

Finally, we bound the privacy profile of the truncated posterior scheme by the privacy profile of the Poisson scheme with sampling probability corresponding to the truncation threshold, using the fact the privacy loss is monotonically increasing in the sampling probability (Lemma C.8), which completes the proof. Part of these last two lemmas is a special case of the tail bound that recently proved in [Choquette-Choo et al., 2023, Theorem 3.1].

Formally,

$$\vec{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \overset{(1)}{=} \vec{\delta}_{\mathcal{T}_t(R)}(\varepsilon) \overset{(2)}{\leq} \vec{\delta}_{\mathcal{T}_{t,\eta}(R)}(\varepsilon) + \vec{\beta}_{\mathcal{A}_t(R)}(\eta) \overset{(3)}{\leq} \vec{\delta}_{\mathcal{P}_{t,\eta}(R)}(\varepsilon) + \vec{\beta}_{\mathcal{A}_t(R)}(\eta),$$

where (1) results from Lemma C.4, (2) from Lemma C.6, and (3) from Lemma C.8.

The same proof can be repeated as is for the add direction. □

Remark C.2. Repeating the previous lemmas while changing the direction of the inequalities and the sign of the lower order terms, we can similarly prove that the random allocation scheme upper bounds the Poisson scheme up to lower order terms, which implies they are asymptotically identical.

Throughout the rest of the section, claims will be stated in terms of R and $*$ for simplicity, but can be generalized to M and x. We note that $\mathcal{A}_{t,k}\left(R;\perp\right) = R^{\otimes t}(\perp)$ where $R^{\otimes t}(\perp)$ denotes t sequential calls to $R(\perp)$, an identity which will be used several times throughout the next section

C.1 Posterior scheme

We start by introducing the posterior sampling scheme, where the sampling probability depends on the previous outputs.

Definition C.3 (Posterior probability and scheme)**.** Given a subset size $k \in [t]$, an index $i \in [t-1]$, a view $\boldsymbol{v}^i \in \mathcal{Y}^i$, and a randomizer R, the $i+1$ *posterior probability* of the k allocation out of t given $\boldsymbol{v}^i$ is the probability that the index $i+1$ was one of the k steps chosen by the random allocation scheme, given that the view $\boldsymbol{v}^i$ was produced by the first i rounds of $\mathcal{A}_t\left(R;*\right)$. Formally, $\lambda_{\boldsymbol{v}^i,k} := P_{\mathcal{A}_{t,k}(R;*)}\left(i+1 \in \boldsymbol{I} | \boldsymbol{v}^i\right)$, where $\boldsymbol{I}$ is the subset of chosen steps.

The *posterior scheme* is a function $\mathcal{T}_{t,k}\left(R\right) : \{*,\perp\} \to \mathcal{Y}^t$ parametrized by a randomizer R, number of steps t, and number of selected steps k, which given $*$, sequentially samples

$$y_{i+1} \sim \left(\lambda_{\boldsymbol{v}^i,k} \cdot R(*) + (1-\lambda_{\boldsymbol{v}^i,k}) \cdot R(\perp)\right),$$

where $\lambda_{\boldsymbol{v}^0,k} = k/t$, and $\mathcal{T}_{t,k}\left(R;\perp\right) = \mathcal{A}_{t,k}\left(R;\perp\right)$. As before, we omit k from the notations where $k = 1$.

Though this scheme seems like a variation of the Poisson scheme, the following lemma shows that in fact its output is distributed like the output of random allocation.

Lemma C.4. *For any subset size $k \in [t]$ and randomized R, $\mathcal{A}_{t,k}\left(R;*\right)$ and $\mathcal{T}_{t,k}\left(R;*\right)$ are identically distributed, which implies $\vec{\delta}_{\mathcal{A}_{t,k}(R)}(\varepsilon) = \vec{\delta}_{\mathcal{T}_{t,k}(R)}(\varepsilon)$ and $\overleftarrow{\delta}_{\mathcal{A}_{t,k}(R)}(\varepsilon) = \overleftarrow{\delta}_{\mathcal{T}_{t,k}(R)}(\varepsilon)$ for any randomizer and all $\varepsilon \geq 0$.*

Proof. We notice that for all $j \in [t-1]$ and $\boldsymbol{v}^j \in \mathcal{Y}^j$,

$$\begin{aligned}
P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^{j+1}|\boldsymbol{v}^j) &= \frac{P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^{j+1})}{P_{\mathcal{A}_{t,k}(R)}(\boldsymbol{v}^j)} \\
&\overset{(1)}{=} \frac{\sum_{\boldsymbol{i} \subseteq [t], |\boldsymbol{i}|=k} P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^{j+1}, \boldsymbol{I} = \boldsymbol{i})}{P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^j)} \\
&\overset{(2)}{=} \frac{\sum_{\boldsymbol{i} \subseteq [t], |\boldsymbol{i}|=k} P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^j, \boldsymbol{I} = \boldsymbol{i}) \cdot P_{\mathcal{A}_{t,k}(R;*)}(y_{j+1}|\boldsymbol{I} = \boldsymbol{i}, \boldsymbol{v}^j)}{P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^j)} \\
&\overset{(3)}{=} \left(\sum_{\boldsymbol{i} \subseteq [t], |\boldsymbol{i}|=k, j+1 \notin \boldsymbol{i}} \frac{P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^j, \boldsymbol{I} = \boldsymbol{i})}{P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^j)} \right) P_{R(\perp)}(y_{j+1}) \\
&\quad + \left(\sum_{\boldsymbol{i} \subseteq [t], |\boldsymbol{i}|=k, j+1 \in \boldsymbol{i}} \frac{P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^j, \boldsymbol{I} = \boldsymbol{i})}{P_{\mathcal{A}_{t,k}(R;*)}(\boldsymbol{v}^j)} \right) P_{R(*)}(y_{j+1}) \\
&= P_{\mathcal{A}_{t,k}(R;*)}(j+1 \notin \boldsymbol{I}|\boldsymbol{v}^j) \cdot P_{R(\perp)}(y_{j+1}|\boldsymbol{v}^j) + P_{\mathcal{A}_{t,k}(R;*)}(j+1 \in \boldsymbol{I}|\boldsymbol{v}^j) \cdot P_{R(*)}(y_{j+1}) \\
&= (1 - \lambda_{\boldsymbol{v}^j,k}) \cdot P_{R(\perp)}(y_{j+1}) + \lambda_{\boldsymbol{v}^j,k} \cdot P_{R(*)}(y_{j+1}) \\
&\overset{(3)}{=} P_{\mathcal{T}_{t,k}(R;*)}(\boldsymbol{v}^{j+1}|\boldsymbol{v}^j),
\end{aligned}$$

where (1) denotes the subset of steps selected by the allocation scheme by $\boldsymbol{I}$ so $\boldsymbol{I} = \boldsymbol{i}$ denotes the selected subset was $\boldsymbol{i}$, (2) results from the definition $\boldsymbol{v}^{j+1} = (\boldsymbol{v}^j, y_{j+1})$ and Bayes law, (3) from the fact that if $j+1 \in \boldsymbol{I}$ then y_{j+1} depends only on a $*$ and if $j+1 \notin \boldsymbol{I}$ then y_{j+1} depends only on $\perp$, and (4) is a direct result of the posterior scheme definition.

Since $P(\boldsymbol{v}|*) = \prod_{i \in [t-1]} P(\boldsymbol{v}^{j+1}|*, \boldsymbol{v}^j)$ for any scheme, this completes the proof. □

C.2 Truncated scheme

Next we define a truncated variant of the posterior distribution and use it to bound its privacy profile.

Definition C.5 (Truncated scheme)**.** The *truncated posterior scheme* is a function $\mathcal{T}_{t,k,\eta}(R) : \{*, \perp\} \to \mathcal{Y}^t$ parametrized by a randomized R, number of steps t, number of selected steps k, and threshold $\eta \in [0,1]$, which given $*$, sequentially samples

$$y_{i+1} \sim \left(\lambda^{\eta}_{\boldsymbol{v}^i,k} \cdot R(*) + (1 - \lambda^{\eta}_{\boldsymbol{v}^i,k}) \cdot R(\perp) \right),$$

where $\lambda^{\eta}_{\boldsymbol{v}^i,k} := \min\{\lambda_{\boldsymbol{v}^i,k}, \eta\}$, and $\mathcal{T}_{t,k,\eta}(R; \perp) = \mathcal{T}_{t,k}(R; \perp)$.

Next we relate the privacy profiles of the posterior and truncated schemes.

Lemma C.6. *Given a randomizer R, for any $\eta \in [k/t, 1]$; $\varepsilon > 0$ we have*

$$\overrightarrow{\delta}_{\mathcal{T}_{t,k}(R)}(\varepsilon) \leq \overrightarrow{\delta}_{\mathcal{T}_{t,k,\eta}(R)}(\varepsilon) + \overrightarrow{\beta}_{\mathcal{A}_t(R)}(\eta) \quad \textit{and} \quad \overleftarrow{\delta}_{\mathcal{T}_{t,k}(R)}(\varepsilon) \leq \overleftarrow{\delta}_{\mathcal{T}_{t,k,\eta}(R)}(\varepsilon) + \overleftarrow{\beta}_{\mathcal{A}_t(R)}(\eta)$$

where $\overrightarrow{\beta}_{\mathcal{A}_t(R)}(\eta)$ and $\overleftarrow{\beta}_{\mathcal{A}_t(R)}(\eta)$ were defined in Lemma C.1.

The proof of this lemma relies on the relation between $\lambda_{\boldsymbol{v}^i}$ and the privacy loss of the random allocation scheme, stated in the next claim.

Claim C.7. *Given $i \in [t-1]$, a randomizer R, and a view $\boldsymbol{v}^i$ we have $\lambda_{\boldsymbol{v}^i} = \frac{1}{t} e^{\ell_{\mathcal{A}_t(R)}(\boldsymbol{v}^i; \perp, *)}$.*

Proof. From the definition,

$$\begin{aligned}\lambda_{\boldsymbol{v}^i} &= P_{\mathcal{A}_t(R;*)}\left(I = i+1 | \boldsymbol{v}^i\right) \\ &\overset{(1)}{=} \frac{P_{\mathcal{A}_t(R;*)}\left(\boldsymbol{v}^i | I = i+1\right) \cdot P\left(I = i+1\right)}{P_{\mathcal{A}_t(R;*)}\left(\boldsymbol{v}^i\right)} \\ &\overset{(2)}{=} \frac{1}{t} \cdot \frac{P_{\mathcal{A}_t(R;\perp)}\left(\boldsymbol{v}^i\right)}{P_{\mathcal{A}_t(R;*)}\left(\boldsymbol{v}^i\right)} \\ &= \frac{1}{t} e^{\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i;\perp,*\right)},\end{aligned}$$

where (1) results from Bayes law and (2) from the fact $P\left(I = i+1\right) = \frac{1}{t}$ and $P_{\mathcal{A}_t(R;*)}\left(\boldsymbol{v}^i | I = i+1\right) = P_{\mathcal{A}_t(R;\perp)}\left(\boldsymbol{v}^i\right)$. □

Proof of Lemma C.6. Denoting $\mathcal{B}^t_\eta := \left\{\boldsymbol{v} \in \mathcal{Y}^t \mid \max_{i \in [t-1]}\left(\lambda_{\boldsymbol{v}^i}\right) > \eta\right\}$, for any $\mathcal{C} \subseteq \mathcal{Y}^t$ we have

$$\begin{aligned}&\underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{C}\right) \\ &\quad= \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{C}/\mathcal{B}^t_\eta\right) + \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{C} \cap \mathcal{B}^t_\eta\right) \\ &\quad\overset{(1)}{=} \underset{\boldsymbol{V} \sim \mathcal{T}_{t,\eta}(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{C}/\mathcal{B}^t_\eta\right) + \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{C} \cap \mathcal{B}^t_\eta\right) \\ &\quad\overset{(2)}{\leq} e^\varepsilon \underset{\boldsymbol{V} \sim \mathcal{T}_{t,\eta}(R;\perp)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{C}/\mathcal{B}^t_\eta\right) + \boldsymbol{H}_{e^\varepsilon}\left(\mathcal{T}_{t,\eta}\left(R;*\right) \parallel \mathcal{T}_{t,\eta}\left(R;\perp\right)\right) + \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{B}^t_\eta\right) \\ &\quad\leq e^\varepsilon \underset{\boldsymbol{V} \sim \mathcal{T}_{t,\eta}(R;\perp)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{C}\right) + \boldsymbol{H}_{e^\varepsilon}\left(\mathcal{T}_{t,\eta}\left(R;*\right) \parallel \mathcal{T}_{t,\eta}\left(R;\perp\right)\right) + \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{B}^t_\eta\right),\end{aligned}$$

which after reordering the terms implies

$$\boldsymbol{H}_{e^\varepsilon}\left(\mathcal{T}_t\left(R;*\right) \parallel \mathcal{T}_t\left(R;\perp\right)\right) \overset{(2)}{\leq} \boldsymbol{H}_{e^\varepsilon}\left(\mathcal{T}_{t,\eta}\left(R;*\right) \parallel \mathcal{T}_{t,\eta}\left(R;\perp\right)\right) + \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{B}^t_\eta\right),$$

and similarly,

$$\boldsymbol{H}_{e^\varepsilon}\left(\mathcal{T}_t\left(R;\perp\right) \parallel \mathcal{T}_t\left(R;*\right)\right) \overset{(2)}{\leq} \boldsymbol{H}_{e^\varepsilon}\left(\mathcal{T}_{t,\eta}\left(R;\perp\right) \parallel \mathcal{T}_{t,\eta}\left(R;*\right)\right) + \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;\perp)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{B}^t_\eta\right),$$

where (1) results from the definition of the truncated posterior scheme and the set $\mathcal{B}^t_\eta$, (2) from the fact that for any couple of distributions P, Q over some domain $\mathcal{Y}$

$$\boldsymbol{H}_{e^\varepsilon}\left(P \parallel Q\right) = \sup_{\mathcal{C} \subseteq \mathcal{Y}^t}\left(\underset{Y \sim P}{\mathbb{P}}\left(Y \in \mathcal{C}\right) - e^\varepsilon \underset{Y \sim Q}{\mathbb{P}}\left(Y \in \mathcal{C}\right)\right),$$

and (3) from the definition of $\vec{\beta}_{\mathcal{A}_t(R)}(\eta)$.

Combining this with Claim C.7 we get

$$\begin{aligned}\underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{B}^t_\eta\right) &= \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\max_{i \in [t-1]}\left(\lambda_{\boldsymbol{v}^i}\right) > \eta\right) \\ &= \underset{\boldsymbol{V} \sim \mathcal{T}_t(R;*)}{\mathbb{P}}\left(\max_{i \in [t-1]}\left(\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i;\perp,*\right)\right) > \ln(t\eta)\right) \\ &= \vec{\beta}_{\mathcal{A}_t(R)}(\eta),\end{aligned}$$

and similarly

$$\underset{\boldsymbol{V} \sim \mathcal{A}_t(R;\perp)}{\mathbb{P}}\left(\boldsymbol{V} \in \mathcal{B}^t_\eta\right), = \overleftarrow{\beta}_{\mathcal{A}_t(R)}(\eta).$$

□

We now relate the truncated scheme's privacy profile to Poisson.

Lemma C.8. *Given $\eta \in [0,1]$ and a randomizer R, we have $\vec{\delta}_{\mathcal{T}_{t,\eta}(R)}(\varepsilon) \leq \vec{\delta}_{\mathcal{P}_{t,\eta}(R)}(\varepsilon)$ and $\overleftarrow{\delta}_{\mathcal{T}_{t,\eta}(R)}(\varepsilon) \leq \overleftarrow{\delta}_{\mathcal{P}_{t,\eta}(R)}(\varepsilon)$ for all $\varepsilon > 0$.*

The proof makes use of the following result.

Claim C.9 (Theorem 10 in [Zhu et al., 2022])**.** *If a pair of distributions (P, Q) dominates a randomizer R and (P', Q') dominate R', then $(P \times P', Q \times Q')$ dominate the composition of R and R'.*

Proof of Lemma C.8. We first notice that the the hockey-stick divergence between a mixture distribution $\lambda P + (1-\lambda)Q$ and its second component Q is monotonically increasing in its mixture parameter λ. For any $0 \leq \lambda \leq \lambda' \leq 1$ and two distributions P_0, P_1 over some domain, denoting $Q_\lambda \coloneqq (1-\lambda)P_0 + \lambda P_1$ we have, $Q_{\lambda'} = \frac{1-\lambda'}{1-\lambda} Q_\lambda + \frac{\lambda'-\lambda}{1-\lambda} P_1$. From the quasi-convexity of the hockey-stick divergence, for any $\kappa \geq 0$ we have

$$\boldsymbol{H}_\kappa\left(Q_{\lambda'} \,\|\, P_1\right) = \boldsymbol{H}_\kappa\left(\frac{1-\lambda'}{1-\lambda} Q_\lambda + \frac{\lambda'-\lambda}{1-\lambda} P_1 \,\|\, P_1\right) \leq \boldsymbol{H}_\kappa\left(Q_\lambda \,\|\, P_1\right),$$

and similarly, $\boldsymbol{H}_\kappa\left(P_1 \,\|\, Q_{\lambda'}\right) \leq \boldsymbol{H}_\kappa\left(P_1 \,\|\, Q_\lambda\right)$.

Using this fact we get that the privacy profile of a single call to a Poisson subsampling algorithm is monotonically increasing in its sampling probability w.r.t. both $\overset{\rightarrow}{\simeq}$ and $\overset{\leftarrow}{\simeq}$, so the privacy profile of every step of $\mathcal{T}_{t,\eta}(R)$ is upper bounded by that of $\mathcal{P}_{1,\eta}(R)$, and from Claim C.9 its t times composition is the dominating pair of $\mathcal{P}_{t,\eta}(R)$, which completes the proof. □

C.3 Proof of Lemma C.1

Proof. The proof directly results from combining the previous lemmas.

$$\vec{\delta}_{\mathcal{A}_t(R)}(\varepsilon) \overset{(1)}{=} \vec{\delta}_{\mathcal{T}_t(R)}(\varepsilon) \overset{(2)}{\leq} \vec{\delta}_{\mathcal{T}_{t,\eta}(R)}(\varepsilon) + \vec{\beta}_{\mathcal{A}_t(R)}(\eta) \overset{(3)}{\leq} \vec{\delta}_{\mathcal{P}_{t,\eta}(R)}(\varepsilon) + \vec{\beta}_{\mathcal{A}_t(R)}(\eta),$$

where (1) results from Lemma C.4, (2) from Lemma C.6, and (3) from Lemma C.8.

The same proof can be repeated as is for the add direction. □

C.4 Proof of the first part of Theorem 4.1

We first reduce the analysis of general approximate-DP algorithms to that of pure-DP ones, paying an additional $t\delta_0$ term in the probability.

Claim C.10. *Given $\varepsilon_0 > 0$; $\delta_0 \in [0,1]$ and a $(\varepsilon_0, \delta_0)$-DP randomizer R, there exists a randomized $\hat{R}$ which is ε_0-DP, such that $\vec{\beta}_{\mathcal{A}_{t,k}(R)}(\eta) \leq \vec{\beta}_{\mathcal{A}_{t,k}\left(\hat{R}\right)}(\eta) + t\delta_0$ and $\overleftarrow{\beta}_{\mathcal{A}_{t,k}(R)}(\eta) \leq \overleftarrow{\beta}_{\mathcal{A}_{t,k}\left(\hat{R}\right)}(\eta) + t\delta_0$, where $\beta_{\mathcal{A}_{t,k}(R)}(\eta)$ was defined in Lemma C.1.*

Proof of Claim C.10. From Lemma 3.7 in [Feldman et al., 2021], there exists a randomizer $\hat{R}$ which is ε_0-DP, such that $\hat{R}(\perp) = R(\perp)$ and $D_{TV}(R(*) \| \hat{R}(*)) \leq \delta_0$.

For any $i \in [t]$ consider the posterior scheme $\mathcal{T}_{t,k,(i)}\left(\hat{R}\right)$ which $\forall j < i$ returns

$$y_{j+1} \sim \left(\lambda_{\boldsymbol{v}^j,k} \cdot R(*) + (1-\lambda_{\boldsymbol{v}^j,k}) \cdot R(\perp)\right),$$

and $\forall j \geq i$ returns

$$y_{j+1} \sim \left(\lambda_{\boldsymbol{v}^i,k} \cdot \hat{R}(*) + (1-\lambda_{\boldsymbol{v}^j,k}) \cdot \hat{R}(\perp)\right).$$

Notice that $\mathcal{T}_{t,k,(0)}\left(\hat{R}\right) = \mathcal{T}_{t,k}(R)$ and $\mathcal{T}_{t,k,(t)}\left(\hat{R}\right) = \mathcal{T}_{t,k}\left(\hat{R}\right)$. From the definition, for any $i \in [t]$ we have $D_{TV}\left(\mathcal{T}_{t,k,(i-1)}\left(\hat{R}; *\right) \| \mathcal{T}_{t,k,(i)}\left(\hat{R}; *\right)\right) \leq \delta_0$, which implies $D_{TV}\left(\mathcal{T}_{t,k}(R; *) \| \mathcal{T}_{t,k}\left(\hat{R}; *\right)\right) \leq t\delta_0$.

Combining this inequality with the fact that for any two distributions P, Q over domain Ω and a subset $\mathcal{C} \subseteq \Omega$ we have $P(\mathcal{C}) \leq Q(\mathcal{C}) + D_{TV}(P \| Q)$ completes the proof. □

Next we provide a closed form expression for the privacy loss of the random allocation scheme.

Claim C.11. *Given an index $i \in [t]$ and a view $\boldsymbol{v}^i \in \mathcal{Y}^i$ we have,*

$$\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i; *, \perp\right) = \ln\left(\frac{1}{t}\left(t - i + \sum_{j\in[i]} e^{\ell_R(y_j;*,\perp)}\right)\right).$$

Proof. From the definition,

$$P_{\mathcal{A}_t(R;*)}(\boldsymbol{v}^i | I = j) = \begin{cases} \left(\prod_{m=0}^{i-1} P_{R(\perp)}(y_j)\right) P_{R(*)}(y_i) \left(\prod_{j=i+1}^{t} P_{R(\perp)}(y_j)\right) & j \leq i \\ \prod_{m\in[i]} P_{R(\perp)}(y_m) & j > i \end{cases}$$

which implies,

$$\frac{P_{\mathcal{A}_t(R;*)}(\boldsymbol{v}^i | I = j)}{P_{\mathcal{A}_t(R;\perp)}(\boldsymbol{v}^i)} = \begin{cases} \frac{P_{R(*)}(y_i)}{P_{R(\perp)}(y_i)} & j \leq i \\ 1 & j > i \end{cases}.$$

Using this identity we get,

$$\begin{aligned} \ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i; *, \perp\right) &= \ln\left(\frac{P_{\mathcal{A}_t(R;*)}(\boldsymbol{v}^i)}{P_{\mathcal{A}_t(R;\perp)}(\boldsymbol{v}^i)}\right) \\ &= \ln\left(\frac{\frac{1}{t}\sum_{j=[t]} P_{\mathcal{A}_t(R;*)}(\boldsymbol{v}^i | I = j)}{P_{\mathcal{A}_t(R;\perp)}(\boldsymbol{v}^i | \perp)}\right) \\ &= \ln\left(\frac{1}{t}\left(t - i + \sum_{j\in[i]} \frac{P_{R(*)}(y_j | \boldsymbol{v}^{j-1})}{P_{R(\perp)}(y_j | \boldsymbol{v}^{j-1})}\right)\right) \\ &= \ln\left(\frac{1}{t}\left(t - i + \sum_{j\in[i]} e^{\ell_R(y_j;*,\perp)}\right)\right). \end{aligned}$$

□

We are now ready to prove the corollary.

Proof of Theorem 4.1. Using Claim C.10 we can limit our analysis to a ε_0-pure DP randomizer. We have,

$$\begin{aligned} \vec{\beta}_{\mathcal{A}_t(R)}(\eta) &= \underset{\boldsymbol{V}\sim\mathcal{A}_t(R;*)}{\mathbb{P}}\left(\max_{i\in[t-1]}\left(\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{V}^i; \perp, *\right)\right) > \ln(t\eta)\right) \\ &\overset{(1)}{=} \underset{\boldsymbol{V}\sim\mathcal{A}_t(R;*)}{\mathbb{P}}\left(\max_{i\in[t]}\left(\frac{1}{t - i + \sum_{j\in[i]} e^{\ell_R(Y_j;*,\perp)}}\right) > \eta\right) \\ &= \frac{1}{t}\sum_{l\in[t]} \underset{\boldsymbol{V}\sim\mathcal{A}_t(R;*)}{\mathbb{P}}\left(\max_{i\in[t]}\left(\frac{1}{t + \sum_{j\in[i]}\left(e^{\ell_R(Y_j;*,\perp)} - 1\right)}\right) > \eta \mid I = l\right) \\ &= \frac{1}{t}\sum_{l\in[t]} \underset{\boldsymbol{V}\sim\mathcal{A}_t(R;*)}{\mathbb{P}}\left(\max_{i\in[t]}\left(\sum_{j\in[i]}\left(1 - e^{\ell_R(Y_j;*,\perp)}\right)\right) > t\left(1 - \frac{1}{t\eta}\right) \mid I = l\right), \end{aligned}$$

where (1) results from Claim C.11, and similarly,

$$\overleftarrow{\beta}_{\mathcal{A}_t(R)}(\eta) \leq \underset{\boldsymbol{V}\sim\mathcal{A}_t(R;\perp)}{\mathbb{P}}\left(\max_{i\in[t]}\left(\sum_{j\in[i]}(1 - e^{\ell_R(Y_j;*,\perp)})\right) > t\left(1 - \frac{1}{t\eta}\right)\right).$$

We can now define the following martingale; $D_0 \coloneqq 0$, $\forall j \in [t-1] : D_j \coloneqq 1 - e^{\ell_R(Y_j;*,\perp)}$, and $S_i \coloneqq \sum_{j=0}^{i} D_j$. Notice that this is a sub-martingale since for any $j \in [t-1]$

$$\underset{Y\sim R(\perp)}{\mathbb{E}}\left[1 - e^{\ell_R(Y;*,\perp)}\right] = 1 - \underset{Y\sim R(\perp)}{\mathbb{E}}\left[\frac{P_{R(*)}(Y)}{P_{R(\perp)}(Y)}\right] = 0$$

and

$$\mathop{\mathbb{E}}_{Y\sim R(*)}\left[1-e^{\ell_R(Y;*,\perp)}\right]=1-\exp\left(\boldsymbol{R}_2\left(R(*)\|R(\perp)\right)\right)\leq 0,$$

where $\boldsymbol{R}_\alpha$ is the α-Rényi divergence (Definition 2.3).

From the fact R is ε_0-DP we have $1-e^{-\varepsilon_0}\leq D_j\leq 1-e^{\varepsilon_0}$ almost surely, so the range of D_j is bounded by $e^{\varepsilon_0}-e^{-\varepsilon_0}=2\cosh(\varepsilon_0)$, and we can invoke the Maximal Azuma-Hoeffding inequality and get for any $l\in[t]$,

$$\begin{aligned}
&\mathop{\mathbb{P}}_{\boldsymbol{V}\sim\mathcal{A}_t(R;*)}\left(\max_{i\in[t]}\left(\sum_{j\in[i]}\left(1-e^{\ell_R(Y_j;*,\perp)}\right)\right)>t\left(1-\frac{1}{t\eta}\right)\mid I=l\right)\\
&\quad\leq\mathop{\mathbb{P}}_{\boldsymbol{V}\sim\mathcal{A}_t(R;\perp)}\left(\max_{i\in[t]}\left(\sum_{j\in[i]}(1-e^{\ell_R(Y_j;*,\perp)})\right)>t\left(1-\frac{1}{t\eta}\right)\right)\\
&\quad=\mathop{\mathbb{P}}_{\boldsymbol{V}\sim\mathcal{A}_t(R;\perp)}\left(\max_{i\in[t]}(C_i)>t\left(1-\frac{1}{t\eta}\right)\right)\\
&\quad\leq\exp\left(-\frac{t}{2}\left(\frac{t\eta-1}{t\eta\cosh(\varepsilon_0)}\right)^2\right).\\
&\quad\leq\exp\left(-\frac{t}{2}\left(\frac{\gamma}{\cosh(\varepsilon_0)}\right)^2\right).\\
&\quad\leq\delta,
\end{aligned}$$

where the last two steps result from the definition of η and γ. □

C.5 Proof of the second part of Theorem 4.1

The proof makes use of the following result.

Claim C.12 (Part 2 of lemma 3.3 in Kasiviswanathan and Smith [2014]). *Given $\varepsilon>0$; $\delta\in[0,1]$ and a (ε,δ)-DP algorithm M, for any neighboring datasets $\boldsymbol{s}\simeq\boldsymbol{s}'$ we have,*

$$\mathop{\mathbb{P}}_{Y\sim M(\boldsymbol{s})}\left(\ell\left(Y;\boldsymbol{s},\boldsymbol{s}'\right)>2\varepsilon\right)\leq\frac{2\delta}{1-e^{-\varepsilon}}.^{9}$$

Proof. Consider the following algorithm based on the randomizer $R_\eta:\{*,\perp\}\times\mathcal{Y}^*\to\mathcal{Y}$ which is defined by $R_\eta(\perp;\boldsymbol{v})=R(\perp)$, and $R_\eta(*;\boldsymbol{v})=\begin{cases}R(*) & \ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i;\perp,*\right)<\ln(t\eta)\\ R(\perp) & \ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i;\perp,*\right)\geq\ln(t\eta)\end{cases}$.

Given a view $\boldsymbol{v}$, denote

$$i_{\boldsymbol{v}}:=\begin{cases}\underset{i\in[t-1]}{\arg\min}\left(\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i;\perp,*\right)\geq\ln(t\eta)\right) & \max\limits_{i\in[t-1]}\left(\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i;\perp,*\right)\right)>\ln(t\eta)\\ 0 & \max\limits_{i\in[t-1]}\left(\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i;\perp,*\right)\right)\leq\ln(t\eta)\end{cases},$$

the first index where the privacy loss exceeds $\ln(t\eta)$ if such index exists and 0 otherwise, and notice that,

$$\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v};\perp,*\right)\overset{(1)}{=}\ln\left(\frac{1}{\frac{1}{t}\sum_{i\in[t]}e^{\ell_R(y_i;*,\perp)`}}\right)\overset{(2)}{=}\ln\left(\frac{t}{t-i_{\boldsymbol{v}}+\sum_{i\in[i_{\boldsymbol{v}}]}e^{\ell_R(y_i;*,\perp)}}\right),$$

where (1) results from Claim C.11 and (2) from the definition of R_η which implies the distribution of $R_\eta(*;\boldsymbol{v}^j)=R_\eta(\perp;\boldsymbol{v}^j)$ for all $j>i_{\boldsymbol{v}}$.

[9] We note the journal version has a typo in the δ part of the statement, which does not match the proof. We use the corrected version which appears in the Arxiv version.

Using this fact we get,

$$
\begin{aligned}
\vec{\beta}_{\mathcal{A}_t(R)}(\eta) &= \underset{\boldsymbol{V}\sim\mathcal{A}_t(R;*)}{\mathbb{P}}\left(\max_{i\in[t-1]}\left(\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^i;\perp,*\right)\right) > \ln(t\eta)\right) \\
&\overset{(1)}{=} \underset{\boldsymbol{V}\sim\mathcal{A}_t(R;*)}{\mathbb{P}}\left(\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}^{i_{\boldsymbol{v}}};\perp,*\right) > \ln(t\eta)\right) \\
&\overset{(2)}{=} \underset{\boldsymbol{V}\sim\mathcal{A}_t(R;*)}{\mathbb{P}}\left(\frac{1}{t-i_{\boldsymbol{v}}+\sum_{j\in[i_{\boldsymbol{v}}]}e^{\ell_R(y_j;*,\perp)}} > \eta\right) \\
&\overset{(3)}{=} \underset{\boldsymbol{V}\sim\mathcal{A}_t(R_\eta;*)}{\mathbb{P}}\left(\frac{1}{t-i_{\boldsymbol{v}}+\sum_{j\in[i_{\boldsymbol{v}}]}e^{\ell_R(y_j;*,\perp)}} > \eta\right) \\
&\overset{(4)}{=} \underset{\boldsymbol{V}\sim\mathcal{A}_t(R_\eta;*)}{\mathbb{P}}\left(\ell_{\mathcal{A}_t(R_\eta)}\left(\boldsymbol{V};\perp,*\right) > \ln(t\eta)\right) \\
&\overset{(5)}{\leq} \frac{1}{t\eta}\cdot \underset{\boldsymbol{V}\sim\mathcal{A}_t(R_\eta;\perp)}{\mathbb{P}}\left(\ell_{\mathcal{A}_t(R_\eta)}\left(\boldsymbol{V};\perp,*\right) > \ln(t\eta)\right) \\
&\overset{(6)}{\leq} \frac{2}{t\eta-\sqrt{t\eta}}\cdot\overleftarrow{\delta}_{\mathcal{A}_t(R_\eta)}\left(\frac{\ln(t\eta)}{2}\right) \\
&\overset{(7)}{\leq} \frac{2}{t\eta-\sqrt{t\eta}}\cdot\overleftarrow{\delta}_{\mathcal{A}_t(R)}\left(\frac{\ln(t\eta)}{2}\right) \\
&\overset{(8)}{=} \tau\cdot\overleftarrow{\delta}_{\mathcal{A}_t(R)}\left(\varepsilon'\right),
\end{aligned}
$$

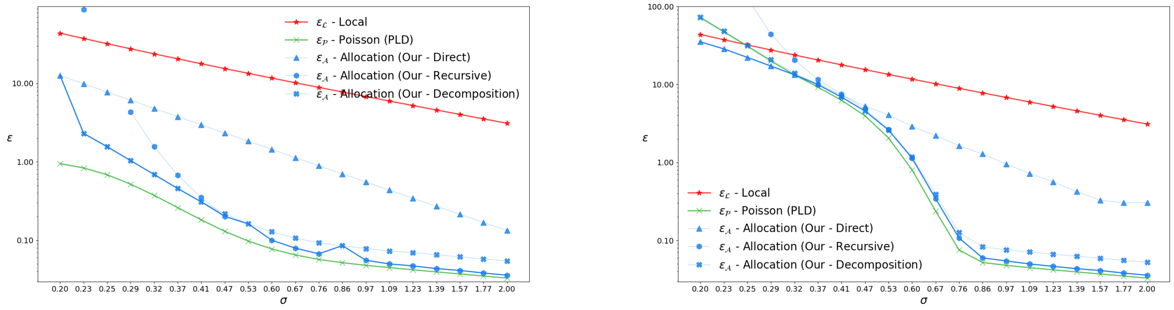
and repeating all steps but (5) we similarly get,

$$
\overleftarrow{\beta}_{\mathcal{A}_t(R)}(\eta) \leq \tau e^{2\varepsilon'}\cdot\overleftarrow{\delta}_{\mathcal{A}_t(R)}\left(\varepsilon'\right),
$$

where (1) results from the definition of $i_{\boldsymbol{v}}$, (2) from Claim C.11, (3) from the definition of R_η, (4) from the previous identity, (5) from the definition of the privacy loss, (6) from Claim C.12, (7) from the fact R_η is dominated by R, and (8) from the definition of η and τ. □

C.6 Separate directions

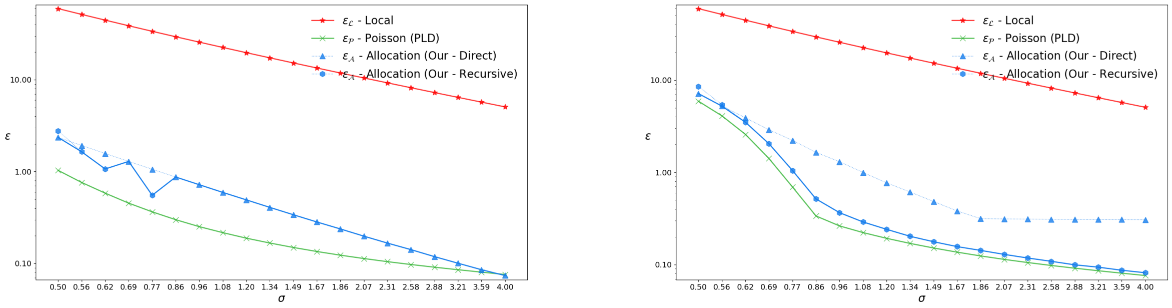
For completeness we present the results of figure 1 for the add and remove directions separately for a single and multiple allocations.



(a) Add direction

(b) Remove direction

Figure 3: Upper bounds on privacy parameter ε or the add and remove directions as a function of the noise parameter σ for various schemes, all using the Gaussian mechanism with fixed parameters $\delta = 10^{-10}$, $t = 10^6$, the same setting as Figure 1



(a) Add direction

(b) Remove direction

Figure 4: Upper bounds on privacy parameter ε or the add and remove directions as a function of the noise parameter σ for various schemes, all using the Gaussian mechanism with fixed parameters $\delta = 10^{-10}$, $t = 10^6$, $k = 10$, the same setting as Figure 1

D Asymptotic analysis

We start by recalling the asymptotic bounds for the Poisson scheme due to Abadi et al. [2016].[10]

Lemma D.1 ([Abadi et al., 2016]). *There exists constants $c_1, c_2 > 0$ such that for any $t \in \mathbb{N}$; $\lambda \in [0, 1/16]$; $\delta \in [0, 1]$, if $t \geq \ln(1/\delta)$ and $\sigma > \max\left\{1, c_1 \frac{\sqrt{\ln(1/\delta)}}{\lambda\sqrt{t}}\right\}$ then the Poisson scheme with the Gaussian mechanism $\mathcal{P}_{t,\lambda}\left(N_\sigma\right)$ is (ε, δ)-DP for any $\varepsilon \geq c_2 \max\left\{\frac{\lambda\sqrt{t \cdot \ln(1/\delta)}}{\sigma}, \lambda^2\sqrt{t \cdot \ln(1/\delta)}\right\}$.*

This is a direct result of the fact the Gaussian mechanism is dominated by the one-dimensional Gaussian randomizer (Claim D.3) where $R(*) = \mathcal{N}(1, \sigma^2)$ and $R(\perp) = \mathcal{N}(0, \sigma^2)$. Combining this Lemma with Corollary 4.2 implies a similar result for the random allocation scheme.

Lemma D.2. *There exist constants c_1, c_2 such that for any $t \in \mathbb{N}$; $k \in [t/16]$; $\delta \in [0, 1]$; if*

$$\sigma \geq c_1 \cdot \max\left\{\sqrt{\ln(t/\delta)}, \sqrt{\frac{k}{t}}\ln(t/\delta), \frac{\sqrt{t \cdot \ln(1/\delta) \cdot \ln(t/k)}}{k}\right\},$$

then the random allocation scheme with the Gaussian mechanism $\mathcal{A}_{t,k}\left(N_\sigma\right)$ is (ε, δ)-DP for any $\varepsilon \geq c_2 \max\left\{\frac{k\sqrt{\ln(1/\delta)}}{\sigma\sqrt{t}}, \frac{k^2\sqrt{\ln(1/\delta)}}{t^{1.5}}\right\}$.

The second term in the bound on ε is due to the privacy profile of the Poisson scheme, and applies only in the uncommon regime when $\sigma > t/k$. One important difference between the privacy guarantees of the Poisson and random allocation schemes is in the bounds on σ, which are stricter for random allocation in the $k > \sqrt{t}$ regime (Remark D.4).

The proof of this Lemma is based on the identity of the dominating pair of the Gaussian mechanism.

Claim D.3 (Gaussian randomizer [Abadi et al., 2016]). *Given $\sigma > 0$, the Gaussian mechanism N_σ is tightly dominated by the pair of distributions $(\mathcal{N}(1, \sigma^2), \mathcal{N}(0, \sigma^2))$, which induce a Gaussian randomizer where $* := 1$ and $\perp := 0$. This pair can be realized by datasets of arbitrary size n of vectors in dimension d by the pair $((\overbrace{\bar{0}, \ldots, \bar{0}}^{n-1 \text{ times}}, e_1), (\overbrace{\bar{0}, \ldots, \bar{0}}^{n \text{ times}}))$.*

We note that the dominating pair of the Gaussian is one dimensional, regardless of the dimension of the original algorithm.

[10] This is a variant of Abadi et al. [2016, Theorem 1] that is better suited for comparison. We prove this version in Appendix C.3.

Proof of Lemma D.2. From Theorem 4.1, each of the schemes has a privacy profile $\delta_{\mathcal{A}_{t/k}(R)}(\varepsilon) \leq \delta_{\mathcal{P}_{t/k,\eta}(R)}(\varepsilon) + t/k\delta_0 + \delta/k$. Applying the union bound to the $t/k\delta_0$ and δ/k terms, and using the fact that the composition of Poisson schemes is a longer Poisson scheme completes the proof of the first part.

From Lemma A.2 we have $\varepsilon_0 = \frac{\sqrt{2\ln(1.25/\delta_0)}}{\sigma} = \frac{\sqrt{2\ln(1.25t/\delta)}}{\sigma}$ (see e.g., Dwork and Roth [2014] for exact derivation). From the first bound on σ we get $\varepsilon_0 \leq 1$ and therefore $\cosh(\varepsilon_0) = (e^{\varepsilon_0} - e^{\varepsilon_0})/2 \leq 3\varepsilon_0/2$. Combining this with the second bound on σ we get,

$$\gamma \leq 3\varepsilon_0\sqrt{\frac{k}{2t}\ln\left(\frac{k}{\delta}\right)} \leq 3\frac{\sqrt{2\ln(1.25t/\delta)}}{\sigma}\sqrt{\frac{k}{2t}\ln\left(\frac{k}{\delta}\right)} \leq \frac{3\sqrt{k}\ln(1.25t/\delta)}{\sqrt{t}\sigma} \leq 1/2,$$

which implies $\eta \leq \frac{2k}{t}$ and $\delta_{\mathcal{P}_{t,\eta}(N_\sigma)}(\varepsilon) \leq \delta_{\mathcal{P}_{t,2k/t}(N_\sigma)}(\varepsilon)$, since the Poisson scheme's privacy profile is monotonic in the sampling probability as proven in Lemma C.8. □

Remark D.4. While the asymptotic bound on ε for the Poisson and random allocation schemes is identical up to the additional logarithmic dependence on t, only the third bound on σ stated for random allocation is required for Poisson. Notice that if $\sqrt{t} > k$ the third term upper bounds the first one, and if additionally $\ln(1/\delta) \leq \frac{t^2}{k^3}$ the second term is bounded by the third one as well. While the first condition might not hold when each element is allocated to many steps, the latter does not hold only when $t < \ln^2(1/\delta)$ which is an uncommon regime of parameters.

E Missing proofs from Section 4.3

Lemma E.1. *Given $1 \leq k \leq k' \leq t$ and a randomizer R dominated by a randomizer R we have $\vec{\delta}_{\mathcal{A}_{t,k}(R)}(\varepsilon) \leq \vec{\delta}_{\mathcal{A}_{t,k'}(R)}(\varepsilon)$ and $\overleftarrow{\delta}_{\mathcal{A}_{t,k}(R)}(\varepsilon) \leq \overleftarrow{\delta}_{\mathcal{A}_{t,k'}(R)}(\varepsilon)$. Furthermore, for any sequence of integers $k \leq k_1 < \ldots < k_j \leq t$, and non-negative $\lambda_1, \ldots, \lambda_j$ s.t. $\lambda_1 + \ldots + \lambda_j = 1$, the privacy profile of $\mathcal{A}_{t,k}(R)$ is upper-bounded by the privacy profile of $\lambda_1\mathcal{A}_{t,k_1}(R) + \ldots + \lambda_j\mathcal{A}_{t,k_j}(R)$, where we use convex combinations of algorithms to denote an algorithm that randomly chooses one of the algorithms with probability given in the coefficient.*

Proof. To prove this claim, we recall the technique used in the proof of Lemma C.1. We proved in Lemma C.4 that $\mathcal{A}_{t,k}(R;*)$ and $\mathcal{T}_{t,k}(R;*)$ are identically distributed. From the non-adaptivity assumption, this is just a sequence of repeated calls to the mixture algorithm $\lambda_{\boldsymbol{v}^i,k,*} \cdot R(*) + (1 - \lambda_{\boldsymbol{v}^i,k,*}) \cdot R(\perp)$.

Next we recall the fact proven in Lemma C.8 that the hockey-stick divergence between this mixture algorithm and $R(\perp)$ is monotonically increasing in λ. Since $\lambda_{\boldsymbol{v}^i,k',*} \geq \lambda_{\boldsymbol{v}^i,k,*}$ for any $k' > k$, this means the pair of distributions $(\lambda_{\boldsymbol{v}^i,k',*} \cdot R(*) + (1 - \lambda_{\boldsymbol{v}^i,k',*}) \cdot R(\perp), R(\perp))$ dominates the pair $(\lambda_{\boldsymbol{v}^i,k',*} \cdot R(*) + (1 - \lambda_{\boldsymbol{v}^i,k',*}) \cdot R(\perp), R(\perp))$ for any iteration i and view $\boldsymbol{v}^i$ (this domination holds in both directions). Using Claim C.9 this implies we can iteratively apply this for all step and get $\delta_{\mathcal{A}_{t,k}(R)}(\varepsilon) \leq \delta_{\mathcal{A}_{t,k'}(R)}(\varepsilon)$ for any $\varepsilon > 0$, thus completing the proof of the first part.

The proof of the second part is identical, since the posterior sampling probability induced by any mixture of $\mathcal{A}_{t,k_1}(R), \ldots, \mathcal{A}_{t,k_j}(R)$ is greater than the one induced by $\mathcal{A}_{t,k}(R)$ the same reasoning follows. □

Lemma E.2. *For any $\lambda \in [0,1]$, element $x \in \mathcal{X}$, and randomizer R we have,*

$$\mathcal{P}_{t,\lambda}(R;x) = \sum_{k=0}^{t} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,k}(R;x),$$

where $B_{t,\lambda}$ is the PDF of the binomial distribution with parameters t, λ and $\mathcal{A}_{t,0}(R;x) := R^{\otimes t}(\perp)$.

Proof. This results from the fact that flipping t coins with bias λ can be modeled as first sampling an integer $k \in \{0, 1, \ldots, t\}$ from a binomial distribution with parameters (t, λ), then uniformly sampling $i_1, \ldots, i_k \in [t]$, and setting the coins to 1 for those indexes. □

The proof of the next claim is a generalization of the advance joint convexity property [Balle et al., 2018, Theorem 2].

Claim E.3. *Given $\lambda \in [0,1]$; $\kappa > 1$ and two distribution P, Q over some domain, we have*

$$\boldsymbol{H}_{\kappa}\left((1-\lambda)Q + \lambda P \;\|\; Q\right) = \lambda \boldsymbol{H}_{\kappa'}\left(P \;\|\; Q\right)$$

$$\boldsymbol{H}_{\kappa}\left(P \;\|\; (1-\lambda)P + \lambda Q\right) = \begin{cases} \beta \boldsymbol{H}_{\kappa''}\left(P \;\|\; Q\right) & \kappa \le \frac{1}{1-\lambda} \\ 0 & \kappa > \frac{1}{1-\lambda} \end{cases},$$

where $\kappa' := 1 + \frac{\kappa - 1}{\lambda}$, $\kappa'' := 1 + \frac{\kappa-1}{1-\kappa+\kappa\lambda}$, and $\beta := 1 - \kappa + \kappa\lambda$.

Proof. The identity for $\boldsymbol{H}_{\kappa}\left((1-\lambda)P + \lambda Q \;\|\; P\right)$ is a direct result of the advanced joint convexity property [Balle et al., 2018, Theorem 2]. For the second part notice that,

$$\begin{aligned} \boldsymbol{H}_{\kappa}\left(P \;\|\; (1-\lambda)P + \lambda Q\right) &= \int_{\Omega} \left[P(\omega) - \kappa\left((1-\lambda)P(\omega) + \lambda Q(\omega)\right)\right]_{+} d\omega \\ &= \int_{\Omega} \left[(1 - \kappa + \kappa\lambda)P(\omega) - \lambda\kappa Q(\omega)\right]_{+} d\omega \\ &= \begin{cases} \beta \int_{\Omega} \left[P(\omega) - \kappa'' Q(\omega)\right]_{+} d\omega & \alpha \le \frac{1}{1-\lambda} \\ 0 & \kappa > \frac{1}{1-\lambda} \end{cases} \\ &= \begin{cases} \beta \boldsymbol{H}_{\kappa''}\left(P \;\|\; Q\right) & \kappa \le \frac{1}{1-\lambda} \\ 0 & \kappa > \frac{1}{1-\lambda} \end{cases}. \end{aligned}$$

□

Lemma E.4. *For any $\lambda \in [0,1]$; $\varepsilon > 0$ and randomizer R we have*

$$\boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{P}^{+}_{t,\lambda}(R;*) \;\middle\|\; \mathcal{P}^{+}_{t,\lambda}(R;\perp)\right) = \vec{\gamma} \boldsymbol{H}_{e^{\vec{\varepsilon}}}\left(\mathcal{P}_{t,\lambda}\left(R;*\right) \;\|\; \mathcal{P}_{t,\lambda}\left(R;\perp\right)\right)$$

$$\boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{P}^{+}_{t,\lambda}(R;\perp) \;\middle\|\; \mathcal{P}^{+}_{t,\lambda}(R;*)\right) = \overleftarrow{\gamma} \boldsymbol{H}_{e^{\overleftarrow{\varepsilon}}}\left(\mathcal{P}_{t,\lambda}\left(R;\perp\right) \;\|\; \mathcal{P}_{t,\lambda}\left(R;*\right)\right),$$

where

$$\mathcal{P}^{+}_{t,\lambda}(R;x) = \vec{\gamma} \sum_{k \in [t]} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,k}\left(R;x\right)$$

is the Poisson scheme conditioned on allocating the element at least once, and $\vec{\gamma}$, $\overleftarrow{\gamma}$, $\vec{\varepsilon}$, and $\overleftarrow{\varepsilon}$ were defined in Theorem 4.3.

Proof. First notice that,

$$\begin{aligned} \mathcal{P}_{t,\lambda}\left(R;x\right) &\overset{(1)}{=} \sum_{k=0}^{t} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,k}\left(R;x\right) \\ &= B_{t,\lambda}(0) \cdot \mathcal{A}_{t,0}\left(R;x\right) + \sum_{k \in [t]} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,k}\left(R;x\right) \\ &\overset{(2)}{=} (1-\lambda)^{t} \cdot \mathcal{A}_{t,0}\left(R;x\right) + \sum_{k \in [t]} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,k}\left(R;x\right) \\ &\overset{(3)}{=} (1 - 1/\vec{\gamma}) \cdot R^{\otimes t}(\perp) + 1/\vec{\gamma} \cdot \mathcal{P}^{+}_{t,\lambda}(R;x), \end{aligned}$$

where (1) results from Lemma E.2, (2) from the definition of the binomial distribution, λ' and $\mathcal{P}^{+}_{t,\lambda}(R)$, and (3) from the definition of λ' and the fact $\mathcal{A}_{t,0}\left(R;x\right) = R^{\otimes t}(\perp)$.

Taking the converse of Claim E.3 we have,

$$\boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{P}^{+}_{t,\lambda}(R;*) \;\middle\|\; \mathcal{P}^{+}_{t,\lambda}(R;\perp)\right) = \vec{\gamma} \boldsymbol{H}_{e^{\vec{\varepsilon}}}\left(\mathcal{P}_{t,\lambda}\left(R;*\right) \;\|\; \mathcal{P}_{t,\lambda}\left(R;\perp\right)\right).$$

Similarly, from the previous claim we get,

$$\boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{P}^{+}_{t,\lambda}(R;\perp) \;\middle\|\; \mathcal{P}^{+}_{t,\lambda}(R;*)\right) = \overleftarrow{\gamma} \boldsymbol{H}_{e^{\overleftarrow{\varepsilon}}}\left(\mathcal{P}_{t,\lambda}\left(R;\perp\right) \;\|\; \mathcal{P}_{t,\lambda}\left(R;*\right)\right).$$

which completes the proof. □

We can now prove the main theorem.

Proof. [Proof of Theorem 4.3] The proof of this theorem consists of several key steps. First, we show (Lemma E.1) that increasing the number of allocations can only harm the privacy, that is, for any sequence of integers $k \leq k_1 < \ldots < k_j \leq t$, and non-negative $\lambda_1, \ldots, \lambda_j$ s.t. $\lambda_1 + \ldots + \lambda_j = 1$, the privacy profile of $\mathcal{A}_{t,k}\left(R\right)$ is upper-bounded by the privacy profile of $\lambda_1 \mathcal{A}_{t,k_1}\left(R\right) + \ldots + \lambda_j \mathcal{A}_{t,k_j}\left(R\right)$, where we use convex combinations of algorithms to denote an algorithm that randomly chooses one of the algorithms with probability given in the coefficient.

Next, we notice (Lemma E.2) that the Poisson scheme can be decomposed into a sequence of random allocation schemes, by first sampling the number of steps in which the element will participate from the Binomial distribution and then running the random allocation scheme for the corresponding number of steps,

$$\mathcal{P}_{t,\lambda}\left(R; x\right) = \sum_{k=0}^{t} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,k}\left(R; x\right),$$

where $B_{t,\lambda}$ is the PDF of the binomial distribution with parameters t, λ and $\mathcal{A}_{t,0}\left(R; x\right) := R^{\otimes t}(\perp)$.

We then define the Poisson scheme conditioned on allocating the element at least once

$$\mathcal{P}^{+}_{t,\lambda}(R; x) = \vec{\gamma} \sum_{k \in [t]} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,k}\left(R; x\right),$$

and use a generalized version of the advanced joint convexity (Claim E.3) to relate its privacy profile that of the Poisson scheme (Lemma E.4).

Formally,

$$\begin{aligned}
\vec{\delta}_{\mathcal{A}_t(R)}(\varepsilon) &= \boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{A}_{t,k}\left(R; *\right) \,\|\, \mathcal{A}_{t,k}\left(R; \perp\right)\right) \\
&\overset{(1)}{=} \boldsymbol{H}_{e^{\varepsilon}}\left(\vec{\gamma} \sum_{k \in [t]} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,1}\left(R; *\right) \,\|\, \mathcal{A}_{t,k}\left(R; \perp\right)\right) \\
&\overset{(2)}{\leq} \boldsymbol{H}_{e^{\varepsilon}}\left(\vec{\gamma} \sum_{k \in [t]} B_{t,\lambda}(k) \cdot \mathcal{A}_{t,k}\left(R; *\right) \,\|\, \mathcal{A}_{t,k}\left(R; \perp\right)\right) \\
&\overset{(3)}{=} \boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{P}^{+}_{t,\lambda}(R; *) \,\middle\|\, \mathcal{P}^{+}_{t,\lambda}(R; \perp)\right) \\
&\overset{(4)}{=} \vec{\gamma} \boldsymbol{H}_{e^{\vec{\varepsilon}}}\left(\mathcal{P}_{t,\lambda}\left(R; *\right) \,\|\, \mathcal{P}_{t,\lambda}\left(R; \perp\right)\right) \\
&= \vec{\gamma} \cdot \vec{\delta}_{\mathcal{P}_{t,\lambda}(R)}(\vec{\varepsilon}),
\end{aligned}$$

where (1) results from the fact $\sum_{k \in [t]} B_{t,\lambda}(k) = 1/\vec{\gamma}$, (2) from Lemma E.1, (3) from the definition of $\mathcal{P}^{+}_{t,\lambda}$ and the fact that $\mathcal{P}_{t,\lambda}\left(R; \perp\right) = R^{\otimes t}(\perp) = \mathcal{A}_{t,\lambda}\left(R; \perp\right)$, and (4) from the first part of Lemma E.4.

Repeating the same proof using the second part of Lemma E.4 proves the bound on $\overleftarrow{\delta}_{\mathcal{A}_t(R)}$. □

Combining the Poisson decomposition perspective shown in Lemma E.2 with the monotonicity in number of allocations shown in Lemma E.1, additionally implies the following corollary.

Corollary E.5. *For any $\lambda \in [0,1]$; $k \in [t]$, we have $\delta_{\mathcal{P}_{t,\lambda,k}(R)}(\varepsilon) \leq \delta_{\mathcal{P}_{t,\lambda}(R)}(\varepsilon)$, where $\mathcal{P}_{t,\lambda,k}\left(R\right)$ denotes the Poisson scheme where the number of allocations is upper bounded by k.*

Proof Theorem 4.3. The proof directly results from combining the previous lemmas,

$$
\begin{aligned}
\vec{\delta}_{\mathcal{A}_t(R)}(\varepsilon) &= \boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{A}_{t,k}\left(R;*\right) \;\|\; \mathcal{A}_{t,k}\left(R;\perp\right)\right) \\
&\overset{(1)}{=} \boldsymbol{H}_{e^{\varepsilon}}\left(\vec{\gamma}\sum_{k\in[t]} B_{t,\lambda}(k)\cdot\mathcal{A}_{t,1}\left(R;*\right) \;\|\; \mathcal{A}_{t,k}\left(R;\perp\right)\right) \\
&\overset{(2)}{\leq} \boldsymbol{H}_{e^{\varepsilon}}\left(\vec{\gamma}\sum_{k\in[t]} B_{t,\lambda}(k)\cdot\mathcal{A}_{t,k}\left(R;*\right) \;\|\; \mathcal{A}_{t,k}\left(R;\perp\right)\right) \\
&\overset{(3)}{=} \boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{P}^{+}_{t,\lambda}(R;*) \;\middle\|\; \mathcal{P}^{+}_{t,\lambda}(R;\perp)\right) \\
&\overset{(4)}{=} \vec{\gamma}\boldsymbol{H}_{e^{\vec{\varepsilon}}}\left(\mathcal{P}_{t,\lambda}\left(R;*\right) \;\|\; \mathcal{P}_{t,\lambda}\left(R;\perp\right)\right) \\
&= \vec{\gamma}\cdot\vec{\delta}_{\mathcal{P}_{t,\lambda}(R)}(\vec{\varepsilon}),
\end{aligned}
$$

where (1) results from the fact $\sum_{k\in[t]} B_{t,\lambda}(k) = 1/\vec{\gamma}$, (2) from Lemma E.1, (3) from the definition of $\mathcal{P}^{+}_{t,\lambda}$ and the fact that $\mathcal{P}_{t,\lambda}\left(R;\perp\right) = R^{\otimes t}(\perp) = \mathcal{A}_{t,\lambda}\left(R;\perp\right)$, and (4) from the first part of Lemma E.4.

Repeating the same proof using the second part of Lemma E.4 proves the bound on $\overleftarrow{\delta}_{\mathcal{A}_t(R)}$. □

Proof. Notice that,

$$
\begin{aligned}
\delta_{\mathcal{P}_{t,\lambda,k}(R)}(\varepsilon) &\overset{(1)}{\leq} \delta_{\mathcal{P}_{t,\lambda,k}(R)}(\varepsilon) \\
&= \boldsymbol{H}_{e^{\varepsilon}}\left(\mathcal{P}_{t,\lambda,k}\left(R;*\right) \;\|\; \mathcal{P}_{t,\lambda,k}\left(R;\perp\right)\right) \\
&\overset{(2)}{=} \boldsymbol{H}_{e^{\varepsilon}}\left(\left(\sum_{i=0}^{k-1} B_{t,\lambda}(i)\mathcal{A}_{t,i}\left(R;*\right)\right) + \left(\sum_{i=k}^{t} B_{t,\lambda}(i)\right)\mathcal{A}_{t,k}\left(R;*\right) \;\|\; \mathcal{P}_{t,\lambda,k}\left(R;\perp\right)\right) \\
&\overset{(3)}{\leq} \boldsymbol{H}_{e^{\varepsilon}}\left(\sum_{i=0}^{t} B_{t,\lambda}(i)\mathcal{A}_{t,i}\left(R;*\right) \;\|\; \mathcal{P}_{t,\lambda}\left(R;\perp\right)\right) \\
&= \delta_{\mathcal{P}_{t,\lambda}(R)}(\varepsilon),
\end{aligned}
$$

where (1) results from Lemma 3.1, (2) from Lemma E.2 and the definition of $\mathcal{P}_{t,\lambda,k}\left(R\right)$, and (3) from Lemma E.1. □

F Missing proofs from Section 4.4

Lemma F.1. *Given $t,\alpha\in\mathbb{N}$ and two distributions P,Q over some domain Ω, we have*

$$
\boldsymbol{R}_{\alpha}\left(\bar{P}\|Q^{\otimes t}\right) = \frac{1}{\alpha-1}\ln\left(\frac{1}{t^{\alpha}}\sum_{\Pi\in\boldsymbol{\Pi}_t(\alpha)}\binom{t}{C(\Pi)}\binom{\alpha}{\Pi}\prod_{p\in\Pi} e^{(\alpha-1)\boldsymbol{R}_p(P\|Q)}\right),
$$

where $\bar{P} \coloneqq \frac{1}{t}\sum_{i\in[t]}\left(Q^{\otimes(i-1)}\cdot P\cdot Q^{\otimes(t-i)}\right)$, $Q^{\otimes(i-1)}\cdot P\cdot Q^{t-i}$ denotes the distribution induced by sampling all elements from Q, except for the ith one which is sampled from P.

We start by proving a supporting claim.

Claim F.2. *Given $\alpha,t\in\mathbb{N}$ and a list of integers $i_1,\ldots,i_t\geq 0$ such that $i_1+\ldots+i_t=\alpha$, denote by $P(i_1,\ldots,i_t)$ the integer partition of α associated with this list, e.g. if $i_1=1,i_2=0,i_3=2,i_4=1$, then $P=[1,1,2]$. Given an integer partition P of α, we have $|B_P|=\binom{t}{C(\Pi)}$ where,*

$$
B_P = \left\{i_1,\ldots,i_t\geq 0 \mid P(i_1,\ldots,i_t)=P\right\},
$$

and $C(\Pi)$ was defined in Theorem 4.4.

Proof. Given a partition P with unique counts $C(\Pi) = (c_1, \ldots, c_j)$, and an assignments $i_1, \ldots, i_t$ such that $i_1, \ldots, i_t \geq 0$ and $P(i_1, \ldots, i_t) = P$, there are $\binom{t}{c_1}$ ways to assign the first value to c_1 indexes of the possible t, $\binom{t-c_1}{c_2}$ ways to assign the second value to c_2 indexes of of the remaining $t - c_1$ indexes, and so on. Multiplying these terms completes the proof. □

Proof of Lemma F.1. Given a set of integers $i_1, \ldots, i_t \geq 0$ such that $i_1 + \ldots + i_t = \alpha$ we have,

$$\prod_{k\in[t]} \mathop{\mathbb{E}}_{\boldsymbol{V}\sim Q^{\otimes t}}\left[\left(\frac{P(\omega)}{Q(\omega)}\right)^{i_k}\right] = \prod_{p\in\Pi} \mathop{\mathbb{E}}_{Y\sim Q}\left[\left(\frac{P(\omega)}{Q(\omega)}\right)^{p}\right],$$

where P is the integer partition of α defined by $i_1, \ldots, i_t$, e.g. if $i_1 = 1, i_2 = 0, i_3 = 2, i_4 = 1$, then $P = [1, 1, 2]$. This is a result of the fact Y_k are all identically distributed. Notice that the same partition corresponds to many assignments, e.g. $P = [1, 1, 2]$ corresponds to $i_1 = 0, i_2 = 1, i_3 = 1, i_4 = 2$ as well. The number of assignments that correspond to a partition P is $\binom{t}{C(\Pi)}$. Using this fact we get,

$$\begin{aligned}
e^{(\alpha-1)\boldsymbol{R}_\alpha\left(\bar{P}\|Q^{\otimes t}\right)} &\overset{(1)}{=} \mathop{\mathbb{E}}_{\boldsymbol{V}\sim Q^{\otimes t}}\left[\left(\frac{\frac{1}{t}\sum_{i\in[t]}\left(Q^{\otimes(i-1)}\cdot P \cdot Q^{\otimes(t-i)}\right)(\boldsymbol{V})}{Q^{\otimes t}(\boldsymbol{V})}\right)^{\alpha}\right] \\
&= \mathop{\mathbb{E}}_{\boldsymbol{V}\sim Q^{\otimes t}}\left[\left(\frac{1}{t}\sum_{i\in[t]}\frac{P(\omega_i)}{Q(\omega_i)}\right)^{\alpha}\right] \\
&\overset{(2)}{=} \frac{1}{t^\alpha}\mathop{\mathbb{E}}_{\boldsymbol{V}\sim Q^{\otimes t}}\left[\sum_{\substack{i_1,\ldots,i_t\in[\alpha];\\ i_1+\ldots+i_t\geq 0}}\binom{\alpha}{i_1,\ldots,i_t}\prod_{k\in[t]}\left(\frac{P(\omega_i)}{Q(\omega_i)}\right)^{i_k}\right] \\
&\overset{(3)}{=} \frac{1}{t^\alpha}\sum_{\substack{i_1,\ldots,i_t\geq 0;\\ i_1+\ldots+i_t=\alpha}}\binom{\alpha}{i_1,\ldots,i_t}\prod_{k\in[t]}\mathop{\mathbb{E}}_{\boldsymbol{V}\sim Q^{\otimes t}}\left[\left(\frac{P(\omega_i)}{Q(\omega_i)}\right)^{i_k}\right] \\
&\overset{(4)}{=} \frac{1}{t^\alpha}\sum_{\substack{i_1,\ldots,i_t\geq 0;\\ i_1+\ldots+i_t=\alpha}}\binom{\alpha}{i_1,\ldots,i_t}\prod_{p\in\Pi(i_1,\ldots,i_t)}\mathop{\mathbb{E}}_{\omega\sim Q}\left[\left(\frac{P(\omega)}{Q(\omega)}\right)^{p}\right] \\
&\overset{(5)}{=} \frac{1}{t^\alpha}\sum_{\Pi\in\boldsymbol{\Pi}_t(\alpha)}\binom{t}{C(\Pi)}\binom{\alpha}{\Pi}\prod_{p\in\Pi}\mathop{\mathbb{E}}_{\omega\sim Q}\left[\left(\frac{P(\omega)}{Q(\omega)}\right)^{p}\right] \\
&= \frac{1}{t^\alpha}\sum_{\Pi\in\boldsymbol{\Pi}_t(\alpha)}\binom{t}{C(\Pi)}\binom{\alpha}{\Pi}\prod_{p\in\Pi}e^{(\alpha-1)\boldsymbol{R}_p(P\|Q)},
\end{aligned}$$

where (1) results from the definition of $\bar{P}$, (2) is the multinomial theorem, (3) results from the fact ω_i and ω_j are independent for any $i \neq j$, (4) from the fact ω_k are all identically and independently distributed with $P(i_1, \ldots, i_t)$ defined in Claim F.2, and (5) results from Claim F.2. □

Lemma F.3. *Given $\lambda \in [0, 1]$ and two distributions P, Q over some domain Ω, denote $P_\lambda := \frac{P^\lambda Q^{1-\lambda}}{Z_\lambda}$ where Z_λ is the normalizing factor.*

Given $t \in \mathbb{N}$, for any $\varepsilon \in \mathbb{R}$ we have $\boldsymbol{H}_{e^\varepsilon}\left(Q^{\otimes t} \,\middle\|\, \bar{P}\right) \leq \boldsymbol{H}_{e^{\varepsilon'}}\left(Q^{\otimes t} \,\middle\|\, P_{1/t}^{\otimes t}\right)$, where $\bar{P} := \frac{1}{t}\sum_{i\in[t]}\left(Q^{\otimes(i-1)} \cdot P \cdot Q^{\otimes(t-i)}\right)$, $Q^{\otimes(i-1)} \cdot P \cdot Q^{t-i}$ denotes the distribution induced by sampling all elements from Q, except for the ith one which is sampled from P.

Proof. By definition,

$$\ell(\omega; P_\lambda, Q) = \ln\left(\frac{P_\lambda(\omega)}{Q(\omega)}\right) = \ln\left(\frac{P^\lambda(\omega)Q^{1-\lambda}(\omega)}{Q(\omega)Z_\lambda}\right) = \lambda\ln\left(\frac{P(\omega)}{Q(\omega)}\right) - \ln(Z_\lambda) = \lambda\ell(\omega; P, Q) - \ln(Z_\lambda).$$

Given $\varepsilon \in \mathbb{R}$ denote $\varepsilon' \coloneqq \varepsilon - t \cdot \ln(Z_{1/t})$

$$\begin{aligned}
\boldsymbol{H}_{e^\varepsilon}\left(Q^{\otimes t} \;\middle\|\; \bar{P}\right) &= \int_{\Omega^t} \left[Q^{\otimes t}(\boldsymbol{v}) - e^{\varepsilon}\bar{P}(\boldsymbol{v})\right]_+ d\boldsymbol{v} \\
&= \int_{\Omega^t} \left[Q^{\otimes t}(\boldsymbol{v}) - e^{\varepsilon}\frac{1}{t}\sum_{i\in[t]}\left(Q^{\otimes(i-1)} \cdot P \cdot Q^{\otimes(t-i)}\right)(\boldsymbol{v})\right]_+ d\boldsymbol{v} \\
&= \int_{\Omega^t} Q^{\otimes t}(\boldsymbol{v}) \left[1 - e^{\varepsilon}\frac{1}{t}\sum_{i\in[t]} e^{\ell(\omega_i;P,Q)}\right]_+ d\boldsymbol{v} \\
&\overset{(1)}{\leq} \int_{\Omega^t} Q^{\otimes t}(\boldsymbol{v}) \left[1 - e^{\varepsilon + \frac{1}{t}\sum_{i\in[t]}\ell(\omega_i;P,Q)}\right]_+ d\boldsymbol{v} \\
&\overset{(2)}{=} \int_{\Omega^t} Q^{\otimes t}(\boldsymbol{v}) \left[1 - e^{\varepsilon' + \sum_{i\in[t]}\ell\left(\omega_i;P_{1/t},Q\right)}\right]_+ d\boldsymbol{v} \\
&= \int_{\Omega^t} Q^{\otimes t}(\boldsymbol{v}) \left[1 - e^{\varepsilon'}\left(\prod_{i\in[t]}\frac{P_{1/t}(\omega_i)}{Q(\omega_i)}\right)\right]_+ d\boldsymbol{v} \\
&= \int_{\Omega^t} \left[Q^{\otimes t}(\boldsymbol{v}) - e^{\varepsilon'} P_{1/t}^{\otimes t}(\boldsymbol{v})\right]_+ d\boldsymbol{v} \\
&= \boldsymbol{H}_{e^{\varepsilon'}}\left(Q^{\otimes t} \;\middle\|\; P_{1/t}^{\otimes t}\right)
\end{aligned}$$

where (1) results from Jensen's inequality and (2) from the definition of $P_{1/t}$ and the previous claim. □

Proof of Corollary 4.6. From the definition of the Rényi divergence for the Gaussian mechanism,

$$\begin{aligned}
\boldsymbol{R}_\alpha\left(\mathcal{A}_t\left(N_\sigma;1\right)\|\mathcal{A}_t\left(N_\sigma;0\right)\right) &= \boldsymbol{R}_\alpha\left(\mathcal{A}_t\left(N_\sigma;1\right)\|N_\sigma^{\otimes t}(0)\right) \\
&= \frac{1}{\alpha-1}\ln\left(\frac{1}{t^\alpha}\sum_{\Pi\in\boldsymbol{\Pi}_t(\alpha)}\binom{t}{C(\Pi)}\binom{\alpha}{\Pi}\prod_{p\in\Pi}e^{\boldsymbol{R}_p(N_\sigma(1)\|N_\sigma(0))}\right) \\
&= \frac{1}{\alpha-1}\ln\left(\frac{1}{t^\alpha}\sum_{\Pi\in\boldsymbol{\Pi}_t(\alpha)}\binom{t}{C(\Pi)}\binom{\alpha}{\Pi}\prod_{p\in\Pi}e^{\frac{p(p-1)}{2\sigma^2}}\right) \\
&= \frac{1}{\alpha-1}\ln\left(\frac{e^{-\frac{\alpha}{2\sigma^2}}}{t^\alpha}\sum_{\Pi\in\boldsymbol{\Pi}_t(\alpha)}\binom{t}{C(\Pi)}\binom{\alpha}{\Pi}e^{\sum_{p\in\Pi}\frac{p^2}{2\sigma^2}}\right) \\
&= -\frac{\alpha}{\alpha-1}\left(\frac{1}{2\sigma^2} + \ln(t)\right) + \frac{1}{\alpha-1}\ln\left(\sum_{\Pi\in\boldsymbol{\Pi}_t(\alpha)}\binom{t}{C(\Pi)}\binom{\alpha}{\Pi}e^{\sum_{p\in\Pi}\frac{p^2}{2\sigma^2}}\right).
\end{aligned}$$

For the add direction we notice that from the definition, for any $x \in \mathbb{R}$

$$P_R^\lambda(x|*)P_R^{1-\lambda}(x|\bot) = \frac{1}{\sqrt{2\pi}\sigma}e^{-\lambda\frac{(x-1)^2}{2\sigma^2}-(1-\lambda)\frac{x^2}{2\sigma^2}} = \frac{1}{\sqrt{2\pi}\sigma}e^{-\frac{(x-\lambda)^2}{2\sigma^2}-\frac{\lambda(1-\lambda)}{2\sigma^2}}$$

so $R_\lambda(*) = \mathcal{N}(\lambda,\sigma^2)$ and $Z_\lambda = e^{\frac{\lambda(1-\lambda)}{2\sigma^2}}$.

Setting $\lambda = 1/t$ we get,

$$\begin{aligned}
\overleftharpoon{\delta}_{\mathcal{A}_t(N_\sigma)}(\varepsilon) &\overset{(1)}{\leq} \boldsymbol{H}_{e^{\varepsilon'}}\left(\mathcal{N}^{\otimes t}(0,\sigma^2) \;\|\; \mathcal{N}^{\otimes t}(1/t,\sigma^2)\right) \\
&\overset{(2)}{=} \boldsymbol{H}_{e^{\varepsilon'}}\left(\mathcal{N}^{\otimes t}(0,t^2\sigma^2) \;\|\; \mathcal{N}^{\otimes t}(1,t^2\sigma^2)\right) \\
&\overset{(3)}{=} \boldsymbol{H}_{e^{\varepsilon'}}\left(\mathcal{N}(0,t\sigma^2) \;\|\; \mathcal{N}(1,t\sigma^2)\right),
\end{aligned}$$

where (1) results from Theorem 4.5, (2) from the fact that the hockey-stick divergence between two Gaussians with the same scale depends only on the ratio of the difference between their means and their scale, and (3) from the fact that the t-composition hockey-stick divergence between two Gaussians with the same scale amounts to dividing their scale by $\sqrt{t}$. □

We remark that the expression in Corollary 4.6 for the remove direction was previously computed in Liew and Takahashi [2022], up to the improvement of using integer partitions. In this (unpublished) work the authors give an incorrect proof that datasets $(0, \ldots, 0, 1)$ and $(0, \ldots, 0)$ are a dominating pair of datasets for the shuffle scheme applied to Gaussian mechanism. Their analysis of the RDP bound for this pair of distributions is correct (even if significantly longer) and the final expression is identical to ours.

Figure 5 provides a clear example of the advantage of direct analysis, in regimes where the privacy guarantees of the random allocation scheme are better than those of the Poisson scheme. Even though RDP-based bounds on Poisson are not as tight, the RDP-based of allocation is superior to other methods that rely on reduction to Poisson. On the other hand, figure 6 indicates that the gap between our RDP-based bound and Poisson's PLD-based one in other regimes, is mainly due to the fact it relies on RDP, and not a property of random allocation. It additionally reflects the fact that the gap between PLD and RDP based analysis vanishes as the number of epochs grows.

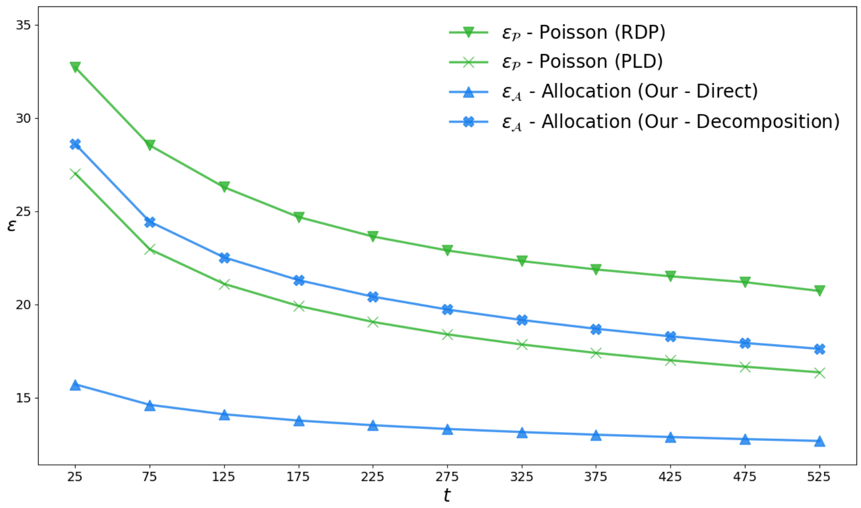


Figure 5: Upper bounds on privacy ε as a function of the number of steps t for the Poisson and random allocation schemes, for fixed parameters $\sigma = 0.3$, $\delta = 10^{-4}$.

F.1 Implementation details

Computation time of the naive implementation of our RDP calculation ranges between second and minutes on a typical personal computer, depending on the α value and other parameters, but can be improved by several orders of magnitude using several programming and analytic steps which we briefly discuss here.

On the programming side, we used vectorization and hashing to reduce runtime. To avoid overflow we computed most quantities in log form, and used and the LSE trick. While significantly reducing the runtime, programming improvements cannot escape the inevitable exponential (in α) nature of this method. Luckily, in most settings, α^* - the α value which induces the tightest bound on ε is typically in the low 10s. Unfortunately, finding α^* requires computing $\boldsymbol{R}_\alpha$, so reducing the range of α values for which $\boldsymbol{R}_\alpha$ is crucial.

We do so by proving an upper bound on α^* in terms of a known bound on ε.

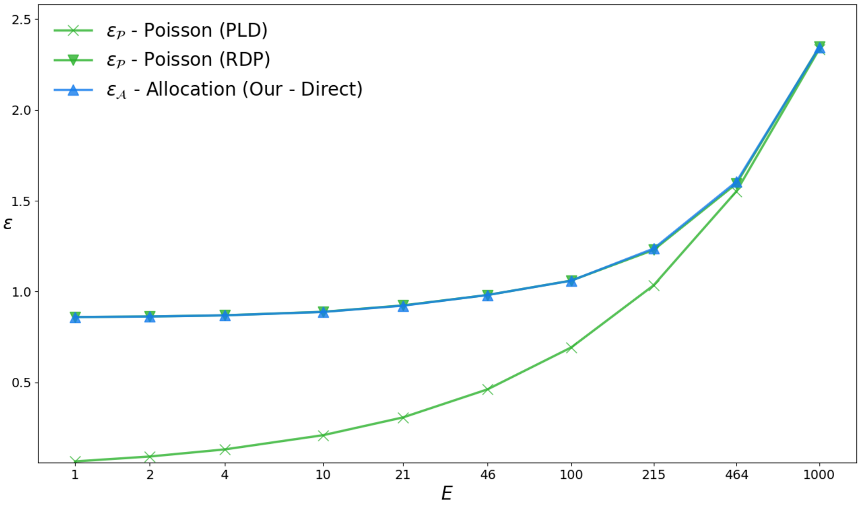


Figure 6: Upper bounds on privacy parameter ε for various schemes all using the Gaussian mechanism, as a function of E the number of "epochs" - times the scheme was sequentially computed, for fixed parameters $\sigma = 1, \delta = 10^{-8}, t = 10^4$.

Claim F.4. *Given $\delta \in (0,1)$ and two distributions P, Q and, denote by $\varepsilon(\delta) \coloneqq \inf_{x>0}(\delta(x) < \delta)$. Given $\varepsilon > 0$, if $\varepsilon(\delta) \leq \varepsilon$ and $\boldsymbol{R}_\alpha\left(P \| Q\right) > \varepsilon$, then $\alpha^* < \alpha$.*

A direct implication of this Lemma is that searching on monotonically increasing values of α and using the best bound on ε achieved at any point to check the relevancy of α, we don't have to compute many values of α greater than α^* before we stop.

Proof. Denote by $\gamma_\delta(\alpha)$ the bound on ε achieved using $\boldsymbol{R}_\alpha\left(P \| Q\right)$. From Lemma A.1, $\gamma_\delta(\alpha) = \boldsymbol{R}_\alpha\left(P \| Q\right) + \phi(\alpha)$ for a non negative ϕ (except for the range $\alpha > 1/(2\delta)$ which provides a vacuous bound). Since $\boldsymbol{R}_\alpha\left(P \| Q\right)$ is monotonically non-decreasing in α we have for any $\alpha' \geq \alpha$,

$$\gamma_\delta(\alpha') \geq \boldsymbol{R}_{\alpha'}\left(P \| Q\right) \geq \boldsymbol{R}_\alpha\left(P \| Q\right) \geq \varepsilon,$$

so it cannot provide a better bound on α. □

G Comparison to other techniques

For completeness, we state how one can directly estimate the hockey-stick divergence of the entire random allocation scheme. This technique was first presented in the context of the Gaussian mechanism by Chua et al. [2024a].

We first provide an exact expression for the privacy profile of the random allocation scheme.

Lemma G.1. *For any randomizer R and $\varepsilon > 0$ we have,*

$$\delta_{\mathcal{A}_t(R)}(\varepsilon) = \mathop{\mathbb{E}}_{\boldsymbol{V} \sim R^{\otimes t}(\perp)} \left[\left[\frac{1}{t} \sum_{i \in [t]} e^{\ell(Y_i; *, \perp)} - e^{\varepsilon} \right]_+ \right].$$

Given $\sigma > 0$, if N_σ is a Gaussian mechanism with noise scale σ we have,

$$\delta_{\mathcal{A}_t(N_\sigma)}(\varepsilon) = \mathop{\mathbb{E}}_{\boldsymbol{Z} \sim \mathcal{N}(\bar{0}, \sigma^2 I_t)} \left[\left[\frac{1}{t} \sum_{i \in [t]} e^{\frac{2Z_i - 1}{2\sigma^2}} - e^{\varepsilon} \right]_+ \right]$$

We note that up to simple algebraic manipulations, this hockey-stick divergence is essentially the expectation of the right tail of the sum of t independent log-normal random variables, which can be approximated as a single log-normal random variable [Neelesh B. et al., 2007], but this approximation typically provide useful guarantees only for large number of steps.

Proof. Denote by I the index of the selected allocation. Notice that for any $i \in [t]$ we have,

$$P_{\mathcal{A}_t(R;*)}(\boldsymbol{v}|I=i) = \left(\prod_{j=1}^{i-1} P_{R(\perp)}(y_j)\right) P_{R(*)}(y_i) \left(\prod_{j=1}^{i-1} P_{R(\perp)}(y_j)\right) = P_{\mathcal{A}_t(R;\perp)}(\boldsymbol{v}) \cdot \frac{P_{R(*)}(y_i)}{P_{R(\perp)}(y_i)}$$

$$\Rightarrow P_{\mathcal{A}_t(R;*)}(\boldsymbol{v}) = \frac{1}{t}\sum_{i\in[t]} P_{\mathcal{A}_t(R;*)}(\boldsymbol{v}|I=i) = \frac{1}{t} P_{\mathcal{A}_t(R;\perp)}(\boldsymbol{v}) \sum_{i\in[t]} \frac{P_{R(*)}(y_i)}{P_{R(\perp)}(y_i)}$$

Using this identity we get,

$$\ell_{\mathcal{A}_t(R)}\left(\boldsymbol{v}; *, \perp\right) = \ln\left(\frac{P_{\mathcal{A}_t(R;*)}(\boldsymbol{v})}{P_{\mathcal{A}_t(R;\perp)}(\boldsymbol{v})}\right) = \ln\left(\frac{1}{t}\sum_{i\in[t]} \frac{P_{R(*)}(y_i)}{P_{R(\perp)}(y_i)}\right) = \ln\left(\frac{1}{t}\sum_{i\in[t]} e^{\ell_R(Y_i;*,\perp)}\right).$$

Plugging this into the definition of the hockey-stick divergence completes the proof of the first part.

The second part is a direct result of the dominating pair for the random allocation scheme of the Gaussian mechanism (Claim D.3). □

G.1 Monte Carlo simulation Chua et al. [2024a]

Using Monte Carlo simulation to estimate this quantity, is typically done using the $\mathop{\mathbb{E}}_{\omega\sim P}\left[\left[1-\alpha e^{-\ell(\omega;P,Q)}\right]_+\right]$ representation of the hockey-stick divergence, so that numerical stability can be achieved by bounding the estimates quantity $\in [0,1]$.

A naive estimation will require an impractical number of experiments, especially in the low δ and high confidence level regimes. These challenges can be partially mitigated using importance sampling and order statistics, a new technique recently presented by Chua et al. [2024a]. Still, this technique suffers from several limitations. It can only account for the setting of $k = 1$ and does not provide a full PLD, and so cannot be composed. It can only estimate δ, so plotting ε as a function of some other parameter is computationally prohibitive. Figure 7 illustrates simultaneously the tightness of our bounds, which are within a constant from the lower bound in the delta regime, and the limitations of the MC methods which become loose in the $\delta < 10^{-4}$ regime for the chosen parameters.

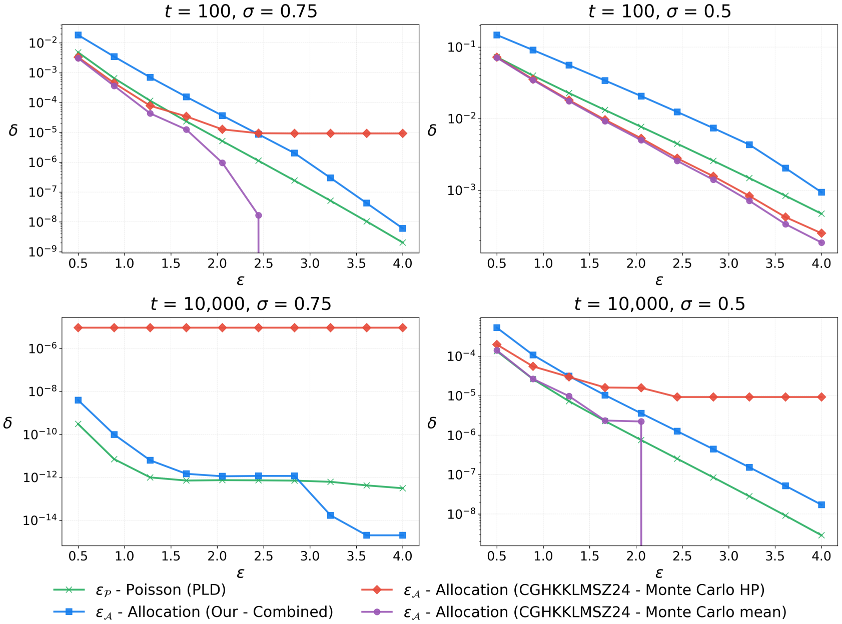


Figure 7: Comparison of δ bound for Poisson scheme with various bounds for the random allocation scheme, for several σ and t values; our combined methods, the high probability and the average estimations of using Monte Carlo simulation with order statistics, $5 \cdot 10^5$ samples and 99% confidence level, by Chua et al. [2024a].

We additionally repeat the analysis using the experimental setting presented in Chua et al. [2024a, Figure 2], both in the form of Figure 1 and Figure 7. The choice of t depends on the experimental settings, the dataset (Criteo pCTR or Criteo search conversion logs) and the batch size.

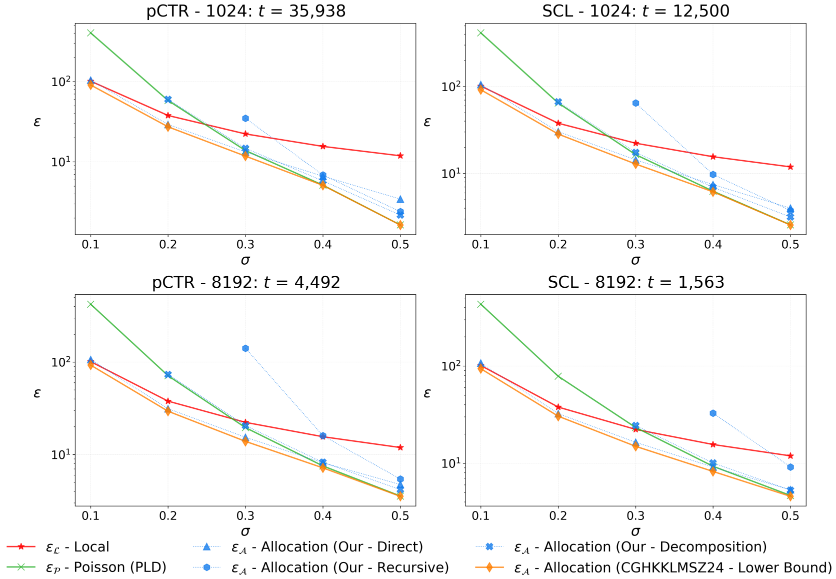


Figure 8: Upper bounds on privacy parameter ε as a function of the noise parameter σ for various schemes and the local algorithm (no amplification), all using the Gaussian mechanism, with privacy parameters $\delta = 10^{-7}$ and various values of t, following the experimental parameters following the experimental settings of Chua et al. [2024a, Figure 2]. In the Poisson scheme $\lambda = 1/t$.

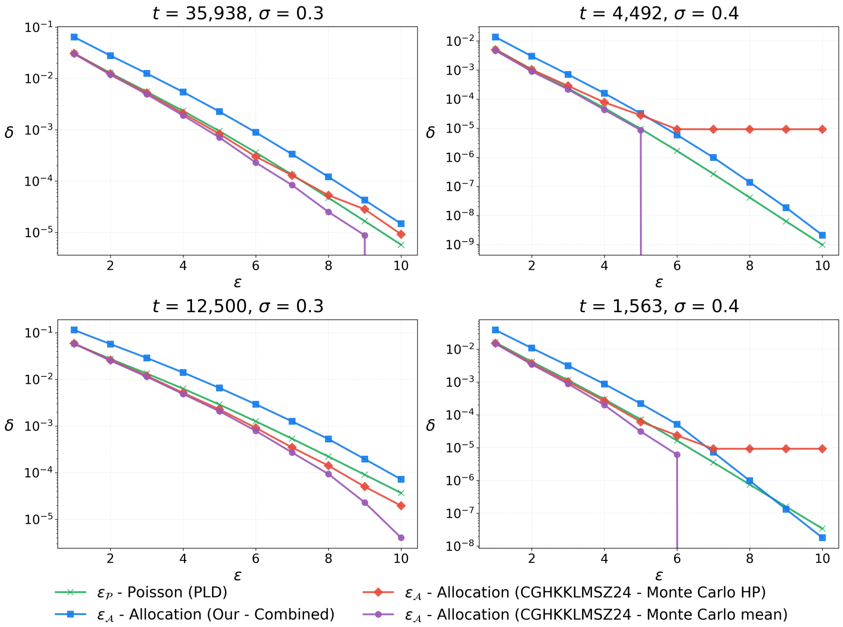


Figure 9: Comparison of δ bound for Poisson scheme with various bounds for the random allocation scheme, for several σ and t values; our combined methods, the high probability and the average estimations of using Monte Carlo simulation with order statistics, $5 \cdot 10^5$ samples and 99% confidence level, following the experimental settings of Chua et al. [2024a, Figure 2].

G.2 RDP-based bound by Dong et al. [2025]

A recent independent work by Dong et al. [2025] considered the same setting under the name Balanced Iteration Subsampling. In Theorem 3.1 they provide two RDP bounds for the remove direction and one for add, that are comparable to Theorem 4.4 in our work. Since the bound for the remove direction always dominated the add direction, we focus on it. The first one is tight but computationally expensive even for the case of $k = 1$, as it sums over $O(t^{k\alpha})$ terms (in the case of $k = 1$ their expression matches the one proposed by Liew and Takahashi [2022], which is mathematically identical to our, but requires $O(t^{\alpha})$ summands rather than our $O(2^{\alpha})$ ones.). The second bound they propose requires summing only over a linear (in k) number of terms which is significantly more efficient than our term, but is lossy. This gap is more pronounced in some parameter regimes, mainly when the α used for inducing the best bound on ε is large. On the other hand, this method allows for direct analysis of the $k > 1$ case, while our analysis relies on the reduction to composition of k runs of the random allocation process with a selection of 1 out of t/k steps.

Figure 10 depicts the spectrum of these effects. For small values of k, our RDP based bounds are tighter than the loose bound proposed by Dong et al. [2025], while for the large values of k when ε is quite large our composition based analysis is looser.

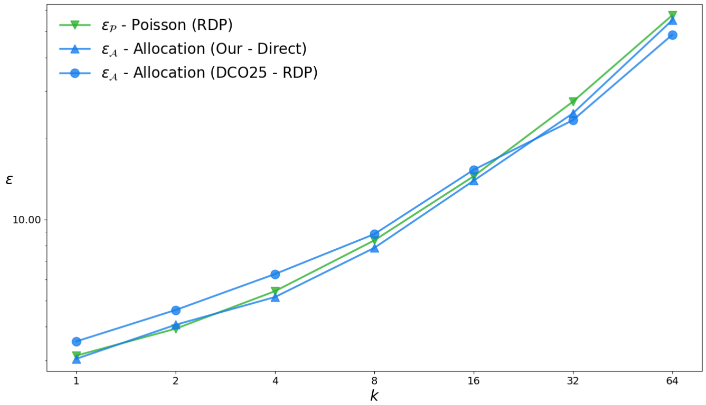


Figure 10: Upper bounds on privacy parameter ε as a function of the the number of allocations k for the Poisson and random allocation schemes, all using the Gaussian mechanism with fixed parameters $\delta = 10^{-6}$, $t = 2^{10}$, $\sigma = 0.6$. The y-axis uses logarithmic scale to emphasize the relative performance.

H Privacy-utility tradeoff

The results of Chua et al. [2024a] show that in the context of training DP-SGD, random allocation (or balls-and-bins sampling) has the utility benefits of shuffling while having the privacy benefits of Poisson subsampling. Here we investigate the privacy-utility trade-off in a simple-to-analyze setting of mean estimation over a Boolean hypercube, that illustrates one possible source of this relative advantage.

We start with the one-dimensional setting. Consider a dataset $\boldsymbol{s} \in \{0,1\}^n$ sampled iid from a Bernoulli distribution with expectation $p \in [0,1]$, where p is estimated from the data elements using one of the two schemes. Formally, at each iteration, the algorithm reports a noisy sum of the elements in the corresponding subset y_i, and the estimated expectation is $\hat{p} := \frac{1}{n}\sum_{i\in[t]} y_i$.

Since $\hat{p}$ is averaged over the various steps, in the case of random allocation with the Gaussian mechanism we have $\hat{p}_{\mathcal{A}} = \frac{1}{n}\left(\sum_{x\in\boldsymbol{s}} x + \sum_{i\in[t]}\xi_i\right)$ where ξ_i is the noise added at step i. From the property of the Gaussian mechanism $\sum_{i\in[t]}\xi_i$ is a Gaussian random variable with variance $t\sigma^2$, and from the definition of the distribution, $\sum_{x\in\boldsymbol{s}} x \sim \mathrm{Bin}(n,p)$, so $\hat{p}_{\mathcal{A}} \sim \frac{1}{n}\left(\mathrm{Bin}(n,p) + \mathcal{N}(0,t\sigma^2)\right)$. In particular this implies $\mathbb{E}[\hat{p}_{\mathcal{A}}] = p$ and $\mathrm{Var}(\hat{p}_{\mathcal{A}}) = \frac{p(1-p)}{n} + \frac{t\sigma^2}{n^2}$, where the first term is the sampling noise and the second is the privacy noise.

Poisson subsampling adds some complexity to the analysis, but can be well approximated for large sample size. The estimation $\hat{p}_{\mathcal{P}}$ follows a similar distribution to that of $\hat{p}_{\mathcal{A}}$, with an additional step. First we sample $u \sim \mathrm{Bin}(n,p)$, then - following the insight introduced in Lemma E.2 - we sample $v_i \sim \mathrm{Bin}(t, 1/t)$ for all $i \in [u]$, which amounts to sampling $m \sim \mathrm{Bin}(u \cdot t, 1/t)$. We note that $\mathbb{E}[m] = u$ and $\mathrm{Var}(m) = u \cdot t \cdot \frac{1}{t}\left(1 - \frac{1}{t}\right) \approx u$. Since w.h.p. $u \approx p \cdot n$, we get $\mathbb{E}[\hat{p}_{\mathcal{P}}] = p$ and $\mathrm{Var}(\hat{p}_{\mathcal{A}}) \approx \frac{p(1-p)}{n} + \frac{p}{n} + \frac{t\sigma^2}{n^2}$, where the first term is the sampling noise, the second is the Poisson sampling noise, and the third is the privacy noise.

The noise scale required to guarantee some fixed privacy parameters using the random allocation scheme is typically larger than the one required by the Poisson scheme, as shown in Figures 1. But following the asymptotic analysis discussed in Section 4.2 we have $t\sigma^2 \approx \frac{\ln(1/\delta)}{\varepsilon^2}$ for both schemes, with constants differing by $\approx 10\%$ in most practical parameter regimes, as shown in Figures 1 and 2, which implies the privacy noise $\frac{t\sigma^2}{n^2}$ is typically slightly larger for the random allocation scheme. On the other hand, for $p \to 1$, the Poisson sampling noise is arbitrarily larger than the sampling noise. Since the privacy bound becomes negligible as n increases, we get that the random allocation scheme asymptotically (in n) dominates the Poisson scheme, as illustrated in Figure 11. In the $\varepsilon = 1, d = 1$

case, the induced σ is sufficiently small, so the gap is dominated by the additional Poisson sampling noise, when $\varepsilon = 0.1$, this effect becomes dominate only for relatively large sample size.

In the high-dimensional setting the privacy noise dominates the sampling noise and therefore the privacy-utility tradeoff is dominated by the difference in the (known) privacy guarantees of the two schemes. In Figure 11 we give an example of this phenomenon for $d = 1000$.

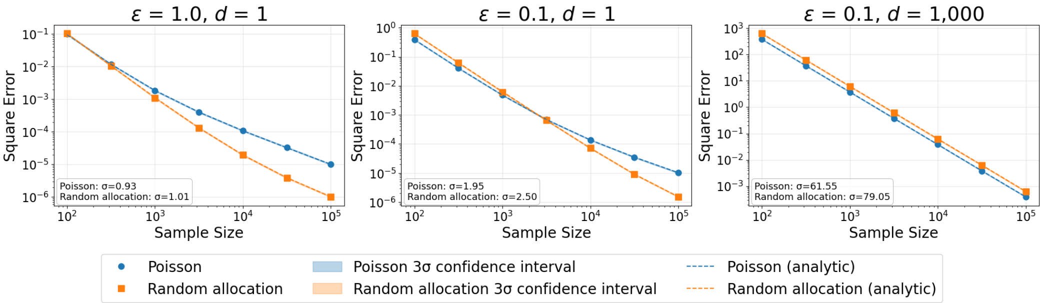


Figure 11: Analytical and empirical square error for the Poisson and random allocation scheme for the setting discussed in Appendix H, for various values of ε and d (which corresponds to an increase in sensitivity). We set $p = 0.9$, $t = 10^3$, $\delta = 10^{-10}$. The experiment was carried 10^4 times, so the 3-std confidence intervals are barely visible.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: All claims are mathematically proven.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss both tightness of the results and computation limitations when applied.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   [Yes]

Justification: All the claims are detailed and all the proofs appear in the appendices.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: The paper discusses the parameters and the supplementary material includes the code.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: The code is provided as supplementary material. No data is involved.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: or [NA]

   Justification: The paper only contains numerical analysis without any usage of data. All relevant details were provided.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [NA]

   Justification: The paper mostly contains numerical analysis without any usage sampling. when MC based methods are considered, CI are provided.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [NA]

   The paper only contains numerical analysis, the longest of which runs for several minutes on a personal computer.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: This is a purely theoretical work.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: This paper provides several new tools for analyzing the privacy of machine learning algorithms. We do not anticipate any impacts beyond those typical for such results.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
    - The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
    - The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
    - If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: No data or models were released in this work.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: Details can be found in the README file.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: No new assets were introduced.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No such thing was done in this paper.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

Justification: No such thing was done in this paper.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: LLMs only assisted with some of the technical coding tasks.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.