# UNLEARNING OFFLINE STOCHASTIC MULTI-ARMED BANDITS

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Machine unlearning aims to unlearn data points from a learned model, offering a principled way to mitigate privacy risks without full retraining, while preserving the utility of the learned model. Prior work has mainly studied unsupervised / supervised machine unlearning, leaving unlearning for sequential decision-making systems far less understood. We initiate the first study of a foundational sequential decision-making problem: offline stochastic multi-armed bandits (MAB). We formalize the privacy constraint for offline MAB and measure utility by the post-unlearning decision quality. We conduct a systematic study under two data-generation models: the fixed-sample model and the distribution model. Our analysis covers single-source requests in both models and extends to multi-source deletion in the fixed-sample setting. For these settings, our algorithmic design is built on two base mechanisms: the downward-shifted Gaussian mechanism and rollback, and we propose adaptive algorithms that switch between them according to the data regime and privacy constraint. We also provide performance guarantees across the above settings and establish lower bounds under both dataset models. Experiments validate the predicted trade-offs and show the effectiveness of the proposed methods.

## 1 INTRODUCTION

Modern machine learning systems are increasingly trained on third-party and user-generated data, from images to texts used in recommendation or advertising (Chen et al. (2019); Tang et al. (2023)). When a user requests that certain records be deleted (unlearned) from the offline dataset, it is no longer enough to merely remove the raw data: trained models may still retain information about the deleted records, and this information can be extracted through inference attacks (Shokri et al. (2017); Carlini et al. (2019)). Motivated by the demand for unlearning, a growing line of work (Ginart et al. (2019); Guo et al. (2020); Gupta et al. (2021); Mehta et al. (2022); Zhang et al. (2022); Yuan et al. (2025)) studied machine unlearning—updating a trained model so that the influence of a specified subset of training data is removed, ideally without retraining from scratch.

To formalize when data removal is sufficient, Ginart et al. (2019) introduced a probabilistic notion of unlearning, inspired by differential privacy (Dwork et al. (2014)). Building on this foundation, a line of work studied supervised machine unlearning (Guo et al. (2020); Izzo et al. (2021)) and unsupervised machine unlearning (Neel et al. (2021); Gupta et al. (2021); Ullah et al. (2021)). Most of these results focused on the empirical risk minimization (ERM) setting, aiming to minimize the training loss on the retained dataset after deleting the requested samples, and several approaches are memory intensive (Bourtoule et al. (2021); Gupta et al. (2021)). To address these limitations, Sekhari et al. (2021) introduced an unlearning definition that incorporated both test-performance guarantees and explicit storage constraints. Building on this framework, Suriyakumar & Wilson (2022) studied online unlearning requests and Liu et al. (2023) investigated unlearning for minimax models.

While substantial progress has been made in supervised / unsupervised machine unlearning, the landscape is far less understood in sequential decision-making problems such as multi-armed bandits (MAB) or reinforcement learning (RL). Recent work (Ye et al. (2025)) has begun exploring reinforcement unlearning and highlights that RL agents may memorize sensitive properties of environments. However, existing formulations primarily target environment-level unlearning, whereas



Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

in offline decision-making systems, deletion requests naturally arise at the level of offline data samples, which are the basic units of standard machine unlearning guarantees. Moreover, they do not quantitatively formalize a privacy-style constraint, and provide no sharp characterization of the resulting privacy-utility trade-off. As a result, unlearning for decision-making still remains largely unexplored.

In this work, we study data-level unlearning for offline stochastic MAB, where utility is measured by the sub-optimality of the final arm recommendation. Unequal arm coverage is a central challenge in offline learning and motivates pessimistic decision rules, which discount poorly supported arms through lower confidence bounds (LCBs) and admit minimax-optimal guarantees (Rashidinejad et al. (2021); Li et al. (2022)). We consider an output-level threat model for a deployed bandit service operating in a trusted environment: before a deletion request arrives, the service has trained on a dataset, deployed the learned model and retained compact statistics. Upon receiving a request, it may additionally use limited request-specific metadata. This stateful formulation is consistent with prior deletion frameworks that distinguish observable-output guarantees from guarantees on the entire internal state (Neel et al. (2021)), permit pre-deletion use while holding historically induced interactions fixed in the counterfactual execution (Cohen et al. (2023)), and preserve internal state across sequential deletion requests (Gupta et al. (2021)).

The threat model is natural for a trusted recommendation environment: after receiving a request, the system internally knows the current learner decision and some deletion-related information. An external observer may subsequently observe the updated recommendation, but cannot inspect the trusted checkpoint or deletion logs. Retaining this historical system state allows the mechanism to preserve stronger performance guarantees in the decision-making setting.

Unlike in many supervised and unsupervised unlearning settings, exact rollback is computationally inexpensive in offline stochastic MAB. However, its low computational cost does not guarantee good post-unlearning utility: under the same privacy constraint, exact rollback can incur substantially greater sub-optimality than noise-based unlearning mechanisms in certain regimes. With the threat model and the distinction discussed above, we therefore address two questions: (1) Under a given privacy constraint, which base unlearning mechanism yields better sub-optimality? (2) How can we design unlearning algorithms built on these base mechanisms that adaptively select the better one across different privacy constraints?

**Our contributions.** *Formulation.* We initiate the study of $(\varepsilon, \delta)$-unlearning for offline stochastic MAB and measure utility by post-unlearning sub-optimality. Our definition builds on the general unlearning framework of Sekhari et al. (2021); Suriyakumar & Wilson (2022); Liu et al. (2023), which captures both test-performance and storage constraints, but adapts it to the offline bandit setting. We consider both fixed-sample and distribution models, as well as single- and multi-source requests.

*Algorithms and theory.* We develop two complementary mechanisms—downward-shifted Gaussian mechanism and rollback—and adaptive algorithms that select between them using retained-data statistics, request metadata, and public parameters. We derive both upper bounds across the above settings and matching or near-matching single-source lower bounds in several regimes. Results are summarized in Table 1. At a high level, the upper bounds compare two types of unlearning costs: Gaussian mechanism incurs a privacy-noise term such as $\frac{k\gamma\sqrt{\ln(1+N\gamma)}}{N(a_0)}$, while rollback is controlled by the post-deletion sample size such as $N(a_0) - k$; both are further limited by the intrinsic offline learning error determined by the coverage of $a^*$.

*Extensive experiments.* We evaluate our algorithms on both synthetic and real-world offline MAB datasets under $\mathcal{M}_f$ and $\mathcal{M}_d$. The results validate the predicted trade-offs and show that our adaptive algorithms consistently match or outperform the baselines by selecting between the Gaussian mechanism and rollback according to the data regime and privacy constraint.

### 1.1 Related work

**Machine unlearning.** Machine unlearning was introduced by Cao & Yang (2015), followed by formal data-deletion and certified-removal frameworks (Ginart et al. (2019); Guo et al. (2020)). Subsequent work has developed certified or approximate unlearning methods based on stability,

Table 1: Theoretical results for the unlearning problem under various settings.

| **Model$^\star$ and Unlearning Type$^\circ$** | **Results$^\dagger$** |
|---|---|
| $\mathcal{M}_f$, Single | UB: $\begin{cases} O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)}}+\frac{k\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N(a_0)}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma<\gamma_0, \\ O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)-k}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma\geq\gamma_0. \end{cases}$ <br> LB: $\Omega\left(e^{-\varepsilon}\sqrt{\frac{1}{N(a_0)-k}}\right)$ |
| $\mathcal{M}_f$, Multi | UB: $\begin{cases} O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N_{\min}}}+\frac{k_{\max}\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N_{\min}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma<\gamma_0', \\ O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N_{\min}-k_{\max}}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma\geq\gamma_0'. \end{cases}$ |
| $\mathcal{M}_d$, Single | $C^*\in[2,\infty)$: UB: $O\left(\min\left\{\sqrt{\frac{C^*\ln(Nm)}{\max\{1,N-2kC^*\}}}, \sqrt{\frac{C^*\ln(Nm)}{N}}+\frac{k\gamma C^*(1+\sqrt{\ln(1+k\gamma C^*)})}{N}\right\}\right)$ <br> LB: $\Omega\left(e^{-\varepsilon}\sqrt{\frac{C^*}{N-k}}\right)$ |
| | $C^*\in(1,2)$: UB: $O\left(e^{-\frac{N-k}{2}\ln\frac{C^*}{8(C^*-1)}+\frac{N+k}{2}\ln\frac{2}{C^*}}\right)$ <br> LB: $\Omega\left((2-C^*)\,e^{-\varepsilon-(N-k)\left(\ln\left(\frac{2}{C^*-1}\right)+\frac{(2-C^*)^2}{15}\right)}\right)$ |

$^\star$ $\mathcal{A}$ is the set of base arms, $|\mathcal{A}|=m$. $N(a)$ denotes the sample count of arm $a$ in the offline dataset $D$, where $|D|=N$. For $(\varepsilon,\delta)$-unlearning, $\gamma$ denotes the analytic Gaussian noise multiplier defined in (1). The fixed-sample model ($\mathcal{M}_f$) assumes that $N(a^*)\geq N_*$, while the distribution model ($\mathcal{M}_d$) assumes that the behavior policy satisfies $d(a^*)\geq\frac{1}{C^*}$. More details are given in Section 2.

$^\circ$ Single-source unlearning refers to the case where only one arm, denoted by $a_0$, makes an unlearning request $U$. Here, $U$ is the set of data samples to be unlearned, with $|U|=k$. Multi-source unlearning refers to the case where at least two arms make unlearning requests, denoted by $\{a_{u_i}\}_{i=1}^{\ell}$. For arm $a_{u_i}$, let $k_i=|U_i|$, where $U_i$ is the corresponding unlearning request.

$^\dagger$ UB means upper bound and LB means lower bound. For single-source, $\gamma_0$ satisfies (2). For multi-source, $N_{\min}=\min_{i\in[\ell]}N(a_{u_i}), k_{\max}=\max_{i\in[\ell]}k_i, \gamma_0'$ satisfies (11).

gradient updates, and second-order approximations (Neel et al. (2021); Ullah et al. (2021); Izzo et al. (2021); Sekhari et al. (2021); Suriyakumar & Wilson (2022); Liu et al. (2023)). More recent studies have begun to consider unlearning in decision-making systems, including recommender systems (Liu et al. (2022)), online learning (Hu et al. (2025)), and reinforcement learning (Ye et al. (2025)). Our work studies a complementary setting: certified unlearning for *offline stochastic bandits*, where utility is determined by the post-unlearning arm decision rather than a parameter-space or prediction loss. We characterize this decision-level cost through finite-sample upper and lower bounds.

**Offline bandit learning.** Offline decision-making relies critically on coverage and pessimism to control distribution shift (Rashidinejad et al. (2021)). Related theory has been developed for offline linear contextual bandits (Li et al. (2022)) and combinatorial bandits (Liu et al. (2025)). We consider two standard offline data models: a fixed-sample model with predetermined per-arm counts (Xiao et al. (2021); Li et al. (2022); Lee et al. (2022); Wang et al. (2022)), and a distribution model in which samples are generated by a behavior policy (Rashidinejad et al. (2021); Zhou & ZHANG (2024); Liu et al. (2025)). Our focus differs from these works in asking how certified deletion changes effective coverage and, consequently, the quality of the final arm recommendation.

**Differential privacy.** Differential privacy has been extensively studied in stochastic and contextual bandits (Mishra & Thakurta (2015); Basu et al. (2019); Ren et al. (2020); Zheng et al. (2020); Chen et al. (2020); Tao et al. (2022); Zhou & ZHANG (2024)). DP also provides a sufficient route to certified unlearning (Guo et al. (2020); Sekhari et al. (2021); Liu et al. (2023)), but it provides

uniform protection against neighboring dataset changes rather than targeting a specific deletion request, and can therefore incur unnecessary utility loss. In contrast, we exploit the structure of the requested deletion and characterize when the Gaussian mechanism or exact rollback yields smaller post-unlearning sub-optimality.

## 2 PRELIMINARIES

We now introduce the model of offline stochastic MAB with its learning and unlearning problems.

### 2.1 LEARNING FOR OFFLINE STOCHASTIC MAB

**Offline dataset.** We focus on offline stochastic MAB. The environment has a set of base arms $\mathcal{A}$ denoted by $[m] = \{1, \dots, m\}$. Pulling arm $a \in \mathcal{A}$ yields a random reward $r \in [0, 1]$ drawn from a fixed but unknown distribution $R(a)$ with mean $\mu(a) = \mathbf{E}_{r\sim R(a)}[r]$. An offline dataset $D = \{(a_i, r_i)\}_{i=1}^N$ of size $N$ is generated from these distributions, where the reward component always follows $r_i \sim R(a_i)$ and the arm component $\{a_i\}_{i=1}^N$ depends on the dataset model. Let $a^* = \arg\max_{a\in\mathcal{A}} \mu(a)$ denote the best arm. We let $N(a)$ be the sample count of arm $a$ in $D$ and $\mathcal{Z}$ be the data point space. Therefore $D \in \mathcal{Z}^N$.

To comprehensively study unlearning, we consider two standard offline dataset models which decide the per-arm sample count in $D$: the fixed-sample model $\mathcal{M}_f$ and the distribution model $\mathcal{M}_d$. Under $\mathcal{M}_f$ (Li et al. (2022); Lee et al. (2022); Wang et al. (2022)), the sample-count vector $\boldsymbol{N} \in \mathbb{N}^m$ is fixed in advance with $\|\boldsymbol{N}\|_1 = N$. The only randomness in $D(\boldsymbol{N})$ comes from the rewards. Under $\mathcal{M}_d$ (Rashidinejad et al. (2021); Zhou & ZHANG (2024); Liu et al. (2025)), each arm $a_i$ is sampled i.i.d. from a behavior policy $d \in \Delta(\mathcal{A})$, and then $r_i \sim R(a_i)$. Equivalently, each data point is sampled i.i.d. from the joint distribution $\mathcal{D} := d \otimes R$, i.e., $(a_i, r_i) \sim \mathcal{D}$ and $D \sim \mathcal{D}^N$.

**Data coverage condition.** Under the fixed-sample model $\mathcal{M}_f$, to guarantee the learning algorithm's performance, we require that for every dataset $D(\boldsymbol{N})$, $a^*$ satisfies $N(a^*) \geq N_*$ for some constant $N_*$. Under the distribution model $\mathcal{M}_d$, following Rashidinejad et al. (2021), we assume a finite concentrability coefficient $C^* > 1$ such that $d(a^*) \geq \frac{1}{C^*}$. The parameter $C^*$ captures deviation between the behavior policy $d$ and the distribution induced by the optimal algorithm.

**Performance Metrics.** A learning algorithm $\pi$ trains on $D$ and outputs an arm $\hat{a} \in \mathcal{A}$, i.e., $\pi : \mathcal{Z}^N \to \mathcal{A}$. The goal of the learning algorithm is to minimize the sub-optimality

$$\mathcal{M}_f : \inf_\pi \sup_{D(\boldsymbol{N})} \mathbf{E}\left[\mu(a^*) - \mu(\hat{a})\right], \qquad \mathcal{M}_d : \inf_\pi \sup_{D\sim\mathcal{D}^N} \mathbf{E}\left[\mu(a^*) - \mu(\hat{a})\right].$$

Here, the supremum is over the arm reward distributions, while the expectation is over data-generation and algorithmic randomness. We use the LCB algorithm of Rashidinejad et al. (2021) by default, as it is minimax optimal under both $\mathcal{M}_f$ and $\mathcal{M}_d$ for $C^* \geq 2$; see Section 3. For $\mathcal{M}_d$ with $C^* \in (1, 2)$, we obtain a sharper guarantee via imitation learning; see Section 3.3.

### 2.2 UNLEARNING FOR OFFLINE STOCHASTIC MAB

Given an offline dataset $D$, an unlearning request specifies a subset $U \subseteq D$ of size $|U| = k$ to be deleted. An unlearning algorithm $\pi'$ is allowed to access a statistic $T(D) \in \mathcal{T}$ to facilitate efficient unlearning. Throughout this paper, $T(D)$ has size $O(m)$, storing only a constant amount of information per arm and is therefore much smaller than the full dataset. We associate each unlearning request $U$ with a request descriptor $\rho(U)$. For requests $U$ from arm $\{a_i\}_{i=1}^\ell$ containing $\{k_i\}_{i=1}^\ell$ samples, we set $\rho(U) = \{(a_i, k_i)\}_{i=1}^\ell$. We define $\mathcal{R}_k = \left\{\{(a_i, k_i)\}_{i=1}^\ell : (a_i \in \mathcal{A}, \forall i \in [\ell]) \text{ and } (\ell \leq m) \text{ and } \left(\sum_{i=1}^\ell k_i = k\right)\right\}$. Therefore $\rho(U) \in \mathcal{R}_k$. $\pi'$ then takes $(U, \emptyset, T(D))$ as input and outputs an updated arm $\hat{a}' \in \mathcal{A}$, i.e., $\pi' : \left[(\mathcal{Z}^k \times \{\emptyset\}) \cup (\{\emptyset\} \times \mathcal{R}_k)\right] \times \mathcal{T} \to \mathcal{A}$. To the best of our knowledge, we are the first to formalize $(\varepsilon, \delta)$-unlearning for offline stochastic bandits.

**Definition 1** ($(\varepsilon, \delta)$-bandit unlearning ($(\varepsilon, \delta)$-BUL)). *A privacy-preserving data-removal mechanism $\pi' : \left[(\mathcal{Z}^k \times \{\emptyset\}) \cup (\{\emptyset\} \times \mathcal{R}_k)\right] \times \mathcal{T} \to \mathcal{A}$ performs $(\varepsilon, \delta)$-BUL for a learning algorithm $\pi$ :*

$\mathcal{Z}^N \to \mathcal{A}$, *if for all* $D \in \mathcal{Z}^N, U \subseteq D, A \subseteq \mathcal{A}$:

$$\mathbf{Pr}\left[\pi'([U,\emptyset], T(D)) \in A\right] \leq e^{\varepsilon} \cdot \mathbf{Pr}\left[\pi'([\emptyset, \rho(U)], T(D \setminus U)) \in A\right] + \delta,$$
$$\mathbf{Pr}\left[\pi'([\emptyset, \rho(U)], T(D \setminus U)) \in A\right] \leq e^{\varepsilon} \cdot \mathbf{Pr}\left[\pi'([U,\emptyset], T(D)) \in A\right] + \delta.$$

Definition 1 compares the actual execution, which receives $[U, \emptyset]$ and $T(D)$, with a reference execution initialized from $T(D \setminus U)$ and given only $[\emptyset, \rho(U)]$. The descriptor $\rho(U)$ retains only the source identities and deletion counts, ensuring the same deletion context without revealing the deleted samples.

The $(\varepsilon, \delta)$-BUL requirement acts as a constraint in our optimization problem. Our objective is to minimize the expected sub-optimality of $\hat{a}'$ relative to $a^*$, subject to $\pi'$ and $\pi$ satisfying $(\varepsilon, \delta)$-BUL. More formally,

$$\text{Under } \mathcal{M}_f: \ \texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) = \inf_{\pi', \pi:(\varepsilon,\delta)\text{-BUL}} \sup_{D(\boldsymbol{N}),\, U \subseteq D,\ |U|=k} \mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right],$$

$$\text{Under } \mathcal{M}_d: \ \texttt{SubOpt}(N, C^*, k, \varepsilon, \delta) = \inf_{\pi', \pi:(\varepsilon,\delta)\text{-BUL}} \sup_{D \sim \mathcal{D}^N,\, U \subseteq D,\ |U|=k} \mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right].$$

For notational simplicity, we omit the dependence on $m$ in `SubOpt`, since $m$ is not central to our analysis. Throughout this paper, we assume that $U$ is selected independently of the reward in each data point. The restriction to reward-independent requests ensures that the empirical mean of each arm computed from $D \setminus U$ remains unbiased. In the definitions above, sup is over the choice of reward distributions of the arms in $D$ and the deletion set $U$, while the expectation is taken over the remaining randomness in the generation of $D$ and in the outputs of $\pi, \pi'$. We call the request single-source if all samples in $U$ come from one arm, and multi-source if $U$ contains samples from at least two arms.

## 3 Algorithms and Results

In this section, we first present the learning algorithm and its sub-optimality guarantee, and then introduce two unlearning primitives, the downward-shifted Gaussian mechanism and rollback. In Section 3.1, we propose an adaptive algorithm for single-source unlearning under both $\mathcal{M}_f$ and $\mathcal{M}_d$ and establish upper bounds. In Section 3.2, we prove lower bounds for single-source unlearning under both models. In Section 3.3, we introduce a mixing algorithm, present an improved learning-unlearning pair for $\mathcal{M}_d$ when $C^* \in (1, 2)$ and finally extend to multi-source unlearning under $\mathcal{M}_f$ and establish an upper bound.

**Algorithm 1** LCB algorithm (Rashidinejad et al. (2021))

**Input:** Dataset: $D = \{(a_i, r_i)\}_{i=1}^N$, a confidence level: $\tau \in (0, 1)$

Set $N(a) \leftarrow \sum_{i=1}^N \mathbb{1}[a_i = a]$ for all $a \in \mathcal{A}$
**for** $a \in \mathcal{A}$ **do**
  **if** $N(a) = 0$ **then**
    Set the empirical mean reward $\hat{\mu}(a) \leftarrow 0$ and the penalty $b(a) \leftarrow 1$
  **else**
    Compute the empirical mean reward $\hat{\mu}(a) \leftarrow \frac{\sum_{i=1}^N r_i \mathbb{1}[a_i = a]}{N(a)}$
    Compute the penalty $b(a) \leftarrow \sqrt{\frac{\ln \frac{m}{\tau}}{2N(a)}}$

**Output:** $\hat{a} = \arg\max_a \hat{\mu}(a) - b(a)$

**The Lower Confidence Bound algorithm.** Let $\hat{\mu}(a), b(a)$ be the empirical mean and penalty of arm $a$ respectively. Algorithm 1 calculates the LCB: $\hat{\mu}(a) - b(a)$ for every arm based on $D$ and outputs the arm with the largest LCB. We have an upper bound as follows.

**Lemma 2** (Theorem 1 of (Rashidinejad et al. (2021))). *For any offline dataset* $D = \{(a_i, r_i)\}_{i=1}^N$, $|\mathcal{A}| = m$, *the arm* $\hat{a}$ *returned by Algorithm 1 satisfies* $\mathcal{M}_f: \mathbf{E}\left[\mu(a^*) - \mu(\hat{a})\right] \leq \sqrt{\frac{2 \ln \frac{m}{\tau}}{N_*}} + \tau$, *and* $\mathcal{M}_d: \mathbf{E}\left[\mu(a^*) - \mu(\hat{a})\right] \leq 2\sqrt{\frac{C^* \ln \frac{m}{\tau}}{N}} + 2\tau$.

**Two unlearning base algorithms.** Our algorithms build on two complementary primitives: the downward-shifted Gaussian mechanism and rollback. The Gaussian mechanism perturbs the affected LCB with calibrated noise, together with a downward shift $\beta\sigma$ that suppresses the probability of promoting a suboptimal source arm. Its utility cost is therefore controlled by the privacy-noise scale. Rollback instead recomputes the affected statistics on $D \setminus U$, and its cost is governed by the resulting loss of arm coverage. Our adaptive algorithms compare these two costs and select the mechanism with the smaller utility guarantee.

**Notations.** Throughout Section 3, let $\widehat{\boldsymbol{\mu}}$, $\widehat{\boldsymbol{N}}$ be the empirical mean and sample-count vectors computed from $D$. Let $\boldsymbol{f} : \mathcal{Z}^* \to \mathbb{R}^m$ map a dataset to its LCB vector. For any vector $\boldsymbol{\nu}$, let $\boldsymbol{\nu}_{(i)}$ denote its $i$-th coordinate. So $\widehat{\boldsymbol{f}}_{(i)}(D)$ is the LCB of the $i$-th arm. For all algorithms involving Gaussian perturbation, assume $\varepsilon > 0$ and $0 < \delta < 1$. Following the analytic Gaussian mechanism of Balle & Wang (2018), let $\Phi$ denote the standard Gaussian cumulative distribution function and define the Gaussian noise multiplier as

$$\gamma := \inf\left\{x > 0 : \Phi\left(\frac{1}{2x} - \varepsilon x\right) - e^{\varepsilon}\Phi\left(-\frac{1}{2x} - \varepsilon x\right) \le \delta\right\}. \tag{1}$$

We further write $D' = D \setminus U$ and defer the corresponding empty-request executions to Appendix A.

### 3.1 Unlearning Algorithms and Upper Bounds

**Algorithm 2** Single-source unlearning algorithm

**Input:** Unlearning request: $U$ from $a_0$, $|U| = k$, additional statistics $T(D)$: $\widehat{\boldsymbol{f}}(D)$, $\widehat{\boldsymbol{\mu}}$, $\widehat{\boldsymbol{N}}$, a confidence level: $\tau \in (0,1), \varepsilon \in (0,\infty), \delta \in (0,1)$

1: Set $\widehat{\boldsymbol{f}}(D') \leftarrow \widehat{\boldsymbol{f}}(D), \widehat{\boldsymbol{N}}' \leftarrow \widehat{\boldsymbol{N}} - k\boldsymbol{I}\{a = a_0\}, \widehat{\boldsymbol{\mu}}' \leftarrow \widehat{\boldsymbol{\mu}}$
2: Compute the new empirical mean reward $\hat{\mu}'(a_0) \leftarrow \frac{\hat{\mu}(a_0)\cdot N(a_0) - \sum_{(a_0,r_i)\in U} r_i}{N'(a_0)}$
3: Compute $\sigma \leftarrow \frac{3k\gamma}{2N(a_0)}$, $\beta \leftarrow \sqrt{2\ln(1+N\sigma)}$ and $\gamma_0$ from (2)
4: **if** $\max_{a\neq a_0}\{\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}}\} < \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$ and $\gamma < \gamma_0$ **then**
5: Sample $\nu \in \mathbb{R}$ from $\mathcal{N}(0,\sigma^2)$
6: Set $\widehat{\boldsymbol{f}}_{(a_0)}(D') \leftarrow \widehat{\boldsymbol{f}}_{(a_0)}(D) + \nu - \beta\sigma$ ▷ *Downward-shifted Gaussian mechanism*
7: **else**
8: Compute the new penalty $b'(a_0) \leftarrow \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$
9: Set $\widehat{\boldsymbol{f}}_{(a_0)}(D') \leftarrow \hat{\mu}'(a_0) - b'(a_0)$ ▷ *Rollback*

**Output:** $\hat{a}' = \arg\max_a \widehat{\boldsymbol{f}}_{(a)}(D')$

We consider single-source unlearning under both models, where $U$ comes from arm $a_0$ and $|U| = k$. Algorithm 2 takes as input the unlearning request $U$, additional statistics $\widehat{\boldsymbol{f}}(D)$, $\widehat{\boldsymbol{\mu}}$, $\widehat{\boldsymbol{N}}$, and a confidence level $\tau$. Under $\mathcal{M}_f$, $\widehat{\boldsymbol{N}} = \boldsymbol{N}$ is fixed, whereas under $\mathcal{M}_d$ it is sampled from behavior policy $d$. The key design is to adaptively compare the downward-shifted Gaussian mechanism with rollback. First, we use the retained data to check whether $a_0$ remains statistically plausible as the optimal arm, i.e., $\max_{a\neq a_0}\left\{\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}}\right\} < \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$. If not, rollback is used. Otherwise, the algorithm compares the corresponding Gaussian and rollback utility guarantees and selects the smaller one. For analysis, we characterize their unique crossing point by $\gamma_0$, defined by

$$\frac{3k\gamma_0\left(\sqrt{2\ln\left(1+\frac{3k\gamma_0 N}{2N(a_0)}\right)} + \frac{1}{\sqrt{2\pi}}\right)}{2N(a_0)} = \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)-k}} - \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)}}. \tag{2}$$

Thus, the Gaussian guarantee is smaller precisely when $\gamma < \gamma_0$. The algorithm need not solve for $\gamma_0$ explicitly and can directly compare the two guarantees; details are provided in Appendix A.1.

**Theorem 3.** *Algorithm 2 and Algorithm 1 are $(\varepsilon, \delta)$-unlearning when $k \le N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$ under both $\mathcal{M}_f$ and $\mathcal{M}_d$.*

To more finely characterize how the sample counts of the unlearned arm $a_0$ and the optimal arm $a^*$ in $D$ affect sub-optimality, we first study $\mathcal{M}_f$, where the per-arm sample count is fixed:

**Theorem 4** ($\mathcal{M}_f$)**.** *Consider any dataset $D(\mathbf{N})$ satisfying $N(a^*) \ge N_*$, and any reward-independent unlearning request $U$ of size $k$ from $a_0$. When $k \le N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$, by setting $\tau = \frac{1}{N}$, the arm $\hat{a}'$ returned by Algorithm 2 satisfies*

$$\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) =$$
$$\begin{cases} O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)}} + \frac{k\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N(a_0)}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma < \gamma_0, \\ O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)-k}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma \ge \gamma_0. \end{cases}$$

**Remark.** *The first line corresponds to the downward-shifted Gaussian branch, while the second corresponds to exact rollback. In both cases, the term depending on $N(a_0)$ captures the additional cost when the deleted arm is optimal, whereas the $N_*$ term is the intrinsic offline-learning error when the optimal arm is unaffected. When $k \le N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$ is violated, Algorithm 2 defaults to exact rollback, for which the rollback guarantee continues to hold.*

We further consider $\mathcal{M}_d$, where the sample-count vector can vary across datasets. Despite this additional randomness, Algorithm 2 continues to achieve good performance as under $\mathcal{M}_f$. The resulting bounds depend on $N$, $k$, and the coverage parameter $C^*$, rather than on a fixed realization of the sample-count vector. We next give the upper bound of $SubOpt(N, C^*, k, \varepsilon, \delta)$.

**Theorem 5** ($\mathcal{M}_d$)**.** *Consider any offline dataset $D \sim \mathcal{D}^N$, where the behavior policy satisfies $d(a^*) \ge \frac{1}{C^*}$ for some $C^* > 1$, and any reward-independent unlearning request $U$ of size $k$ from $a_0$. When $N > 8C^* \ln N$ and $k \le N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$, by setting $\tau = \frac{1}{N}$, the arm $\hat{a}'$ returned by Algorithm 2 satisfies* $\texttt{SubOpt}(N, C^*, k, \varepsilon, \delta) = O\left(\min\left\{\sqrt{\frac{C^*\ln(Nm)}{\max\{1, N-2kC^*\}}}, \sqrt{\frac{C^*\ln(Nm)}{N}} + \frac{k\gamma C^*(1+\sqrt{\ln(1+k\gamma C^*)})}{N}\right\}\right)$.

**Remark.** *The first term inside the minimum is the rollback guarantee, which reflects the loss of effective coverage after deletion, whereas the second is the Gaussian guarantee, which consists of the intrinsic offline-learning error and the additional privacy-noise cost. Thus, Algorithm 2 adaptively attains the better of the two guarantees.*

### 3.2 Lower Bounds

In this section, we present the lower bounds under both models for single-source unlearning. The proof idea is to utilize the definition of $(\varepsilon, \delta)$-BUL and apply Le Cam's method on two constructed bandit instances. Detailed proofs are given in Appendix A.2.

**Theorem 6** ($\mathcal{M}_f$)**.** *Let $\varepsilon \ge 0$, $\delta \le \frac{1}{8\sqrt{e}}$. Suppose $N(a) \ge N^*$ for some $a \ne a_0$. There exist bandit instances and unlearning request $U$ from $a_0$, such that* $\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) = \Omega\left(e^{-\varepsilon}\sqrt{\frac{1}{N(a_0)-k}}\right)$.

**Remark.** *If $N_* \ge N(a_0) - k$ and $\gamma \ge \gamma_0$, the upper bound is dominated by the rollback term. For relatively small $\varepsilon$, Theorem 6 matches this dependence up to logarithmic factors, showing that the coverage loss $N(a_0) - k$ is statistically unavoidable in this regime.*

**Theorem 7** ($\mathcal{M}_d$)**.** *Let $\varepsilon \ge 0$, $\delta \le \frac{1}{8\sqrt{e}}$ and $N \ge 8C^*$. There exist valid bandit instances and unlearning request $U$ from $a_0$, when $C^* \in [2, \infty)$, if $k \le \frac{N}{2C^*}$,* $\texttt{SubOpt}(N, C^*, k, \varepsilon, \delta) =$

$\Omega\left(e^{-\varepsilon}\sqrt{\frac{C^*}{N-k}}\right)$, *while* $C^* \in (1,2)$, *if* $k \leq \frac{N}{2}\left(1-\frac{1}{C^*}\right)$, $\texttt{SubOpt}(N, C^*, k, \varepsilon, 0) = \Omega\left((2-C^*)\, e^{-\varepsilon-(N-k)\left(\ln\left(\frac{2}{C^*-1}\right)+\frac{(2-C^*)^2}{15}\right)}\right)$.

**Remark.** *When $C^* \in [2,\infty)$ and $N - 2kC^* = \Theta(N)$, the upper and lower bounds match in their $\sqrt{\frac{C^*}{N}}$ dependence up to logarithmic factors for relatively small $\varepsilon$. For $C^* \in (1,2)$, the lower bound instead decays exponentially in $N$, motivating the imitation-learning-based design in Section 3.3.*

### 3.3 Extensions

**Improved Results for $C^* \in (1,2)$.** For known $C^* \in (1,2)$, imitation learning achieves a faster rate in $N$ by returning the most frequent arm in $D$ (Rashidinejad et al. (2021)). Building on this, we propose Algorithm 6, a $(0,0)$-BUL imitation-learning method for this regime. Its pseudo-code and analysis are deferred to Appendix A.4, yielding the following upper bound.

**Theorem 8** ($\mathcal{M}_d$)**.** *Consider any offline dataset $D \sim \mathcal{D}^N$, where $d(a^*) \geq \frac{1}{C^*}$ for some $C^* \in (1,2)$, and any reward-independent unlearning request $U$ of size $k$ from $a_0$. When $k \leq \min\left\{\frac{3N}{4}, \frac{(2-C^*)N}{C^*}\right\}$, the arm $\hat{a}'$ returned by Algorithm 6 satisfies $\texttt{SubOpt}(N, C^*, k, \varepsilon, 0) = O\left(e^{-\frac{(N-k)}{2}\ln\frac{C^*}{8(C^*-1)}+\frac{(N+k)}{2}\ln\frac{2}{C^*}}\right)$.*

**Remark.** *By switching the learning algorithm to an imitation learning rule and using rollback for unlearning, Theorem 8 yields a sharper upper bound with an $e^{-N}$ dependence on $N$ when $C^* \to 1$. This term matches the lower bound in Theorem 7 when $C^* \in (1,2)$.*

**Multi-Source Unlearning Algorithm Under $\mathcal{M}_f$.** For multi-source unlearning under $\mathcal{M}_f$, the unlearning request $U$ may involve multiple arms $a_{u_1}, \ldots, a_{u_\ell}$ and can be denoted as $U = \bigcup_{i=1}^{\ell} U_i$, where $U_i$ denotes the subset of deleted samples associated with arm $a_{u_i}$ for $1 \leq i \leq \ell$, and $|U_i| = k_i$. We further define $N_{\min} = \min_{i\in[\ell]} N(a_{u_i}), k_{\max} = \max_{i\in[\ell]} k_i$. We design Algorithm 7, which is $(\varepsilon,\delta)$-BUL with Algorithm 1. Its pseudo-code and corresponding analysis are deferred to Appendix A.5. Accordingly, the form of sub-optimality becomes $\texttt{SubOpt}(\boldsymbol{N}, N_*, \{k_i\}_{i=1}^{\ell}, \varepsilon, \delta)$.

**Theorem 9** ($\mathcal{M}_f$)**.** *Consider any offline dataset $D(\boldsymbol{N})$ satisfying $N(a^*) \geq N_*$, and any reward-independent unlearning request $U = \bigcup_{i=1}^{\ell} U_i$, with $|U_i| = k_i$ from arm $a_{u_i}$. When $k_i \leq N(a_{u_i}) - \sqrt{\frac{N(a_{u_i})\ln\frac{m}{\tau}}{2}}, \forall i \in [\ell]$, $k_{\max} < N_{\min}$, by setting $\tau = \frac{1}{N}$, the arm $\hat{a}'$ returned by Algorithm 7 satisfies*

$$\texttt{SubOpt}(\boldsymbol{N}, N_*, \{k_i\}_{i=1}^{\ell}, \varepsilon, \delta) = \begin{cases} O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N_{\min}}} + \frac{k_{\max}\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N_{\min}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma < \gamma_0', \\ O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N_{\min}-k_{\max}}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma \geq \gamma_0'. \end{cases}$$

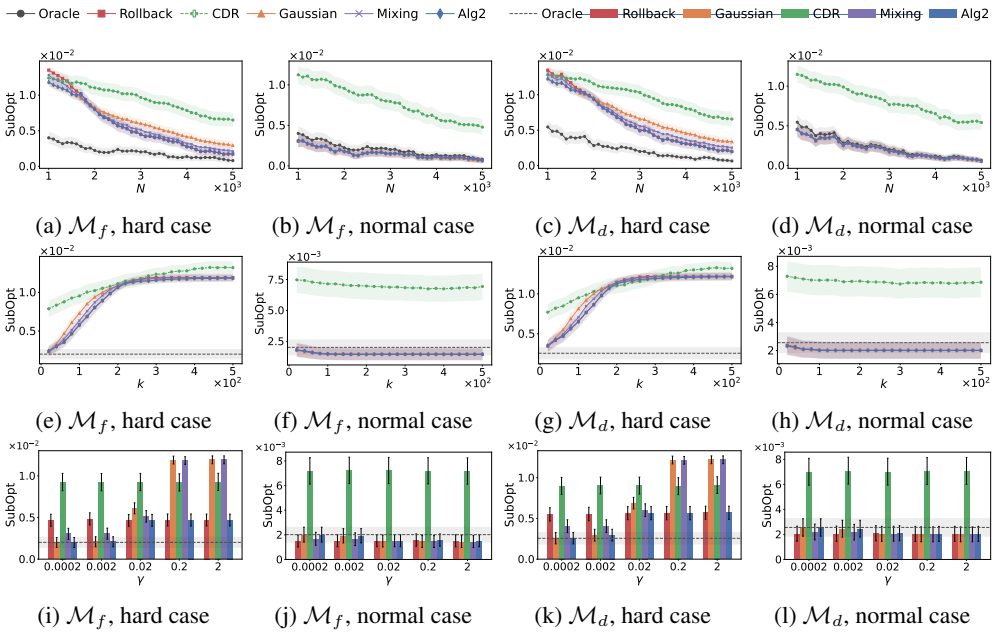
**Remark 6.** Following the batching idea in Guo et al. (2020), when requests need not be handled immediately, they can be accumulated and processed as a single batch. The resulting batch can then be handled by our multi-source algorithm, providing a practical way to address the online setting.

## 4 Experiments

In this section, we report experimental results on the real-world Yelp dataset[1]. We compare Algorithm 2 with an Oracle, Gaussian mechanism, rollback, the mixing algorithm, and the certified data removal method of Guo et al. (2020) (CDR) under both $\mathcal{M}_f$ and $\mathcal{M}_d$. The Oracle runs Algorithm 1

[1] https://business.yelp.com/data/resources/open-dataset/.

on the original dataset without deletion, while all other methods operate after deletion. The mixing algorithm uses $k' = \frac{k}{2}$ in Algorithm 4. Additional experimental details, synthetic results, the $C^* \in (1, 2)$ comparison between Algorithm 6 and Algorithm 2, and results for Algorithm 4 and Algorithm 7 are deferred to Appendix B.




(a) $\mathcal{M}_f$, hard case (b) $\mathcal{M}_f$, normal case (c) $\mathcal{M}_d$, hard case (d) $\mathcal{M}_d$, normal case

(e) $\mathcal{M}_f$, hard case (f) $\mathcal{M}_f$, normal case (g) $\mathcal{M}_d$, hard case (h) $\mathcal{M}_d$, normal case

(i) $\mathcal{M}_f$, hard case (j) $\mathcal{M}_f$, normal case (k) $\mathcal{M}_d$, hard case (l) $\mathcal{M}_d$, normal case

Figure 1: Comparison of Algorithm 2 with baselines under $\mathcal{M}_f$ and $\mathcal{M}_d$ ($C^* = 5$) on Yelp dataset.

We first construct a five-arm offline bandit instance from Yelp-derived user-item feature data, where each user-item pair induces a reward scaled to $[0, 1]$. Based on this constructed reward pool, we generate 200 independent datasets. Under $\mathcal{M}_f$ we use round-robin sampling, while under $\mathcal{M}_d$ the best arm is sampled with probability $\frac{1}{C^*}$ with $C^* = 5$ and the remaining mass is distributed uniformly. To efficiently evaluate deletions, we adopt a prefix-sharing strategy (details in Appendix B.1). In both settings, each prefix undergoes 5 independent deletion requests. We consider the hard case $a_0 = a^*$, which stresses the deletion-induced term in our bounds, and a normal case where $a_0$ is the second-best arm, which isolates the intrinsic offline-learning term. Results are averaged over 10 runs per configuration. We vary one of $N$, $k$, or $\gamma$ while fixing the remaining two parameters to $(N, k, \gamma) = (3000, 80, 0.02)$. Error bars report 95% confidence intervals across 200 dataset seeds.

Figure 1 reports the Yelp results. Under $\mathcal{M}_f$, Algorithm 2 is consistently robust. In the hard case, for $N \geq 3600$, it reduces sub-optimality by 23–46%, 7–26%, and 58–74% relative to the Gaussian mechanism, mixing, and CDR, respectively; for $N \leq 1600$, it improves over rollback and CDR by 6–13% and 5–16%. In the normal case, it achieves 72–86% lower sub-optimality than CDR and remains within 21% of the best baseline. The same trend holds under $\mathcal{M}_d$. In the hard case, for $N \geq 3500$, Algorithm 2 improves over the Gaussian mechanism, mixing, and CDR by 29–41%, 13–23%, and 56–70%, respectively, while for $N \leq 1600$ it improves over rollback and CDR by 4–9% and 4–13%. In the normal case, it reduces sub-optimality by 60–89% relative to CDR and remains within 3% of the best baseline. Overall, these results demonstrate that Algorithm 2 notably outperforms CDR while remaining competitive with the strongest baseline across regimes.

## 5 Conclusion and Future Work

We introduce unlearning for offline stochastic MAB and characterize the statistical trade-off between Gaussian mechanism and rollback through upper and lower bounds across multiple offline-data regimes. Extending this framework to richer contextual and reinforcement-learning settings is an important direction.

### AI USE STATEMENT

In this work, we used generative AI tools, primarily ChatGPT, to aid and polish the writing of the manuscript, including improving grammar, wording, readability, organization, and LaTeX presentation. We also used generative AI to assist in generating synthetic datasets used in the experiments and to assist in proving mathematical claims, including checking intermediate derivations and refining proof arguments.

All AI-assisted writing, synthetic data generation, and mathematical arguments were reviewed and verified by the authors. AI-generated suggestions were treated as intermediate inputs and were independently checked before being incorporated into the paper. We take responsibility for the final content of this work, including all text, mathematical claims, proofs, and experimental artifacts produced with the aid of generative AI.

### ETHICS STATEMENT

This work studies machine unlearning for offline stochastic multi-armed bandits from a theoretical and algorithmic perspective. Our experiments are conducted on synthetic instances (see Appendix B) and do not involve human subjects, personally identifiable information, or sensitive real-world datasets. The proposed methods are intended to improve the ability of learning systems to remove the influence of requested data while preserving computational efficiency and statistical utility. As with machine unlearning more broadly, formal guarantees rely on the assumptions specified by the underlying model and should not be interpreted as providing legal or regulatory compliance in settings where these assumptions do not hold. We are not aware of additional ethical concerns specific to this work.

### REPRODUCIBILITY STATEMENT

We have taken several steps to facilitate reproducibility. All assumptions, problem formulations, and algorithmic procedures used in our theoretical results are stated explicitly in the main paper, and complete proofs of the theoretical claims are provided in the appendix. The experimental setup, including the construction of synthetic instances, evaluation metrics, baselines, and hyperparameter settings, is described in the experimental section and supplementary materials. The supplementary material includes the implementation used to produce all reported experimental results, together with the necessary configuration files and instructions for reproducing the experiments. Randomized experiments are run with controlled random seeds, and the reported results are obtained using the procedures described in the paper and supplementary material.

### AUTHOR CONTRIBUTIONS

The first two authors contributed equally to this work and are designated as co-first authors.

## REFERENCES

Borja Balle and Yu-Xiang Wang. Improving the Gaussian mechanism for differential privacy: Analytical calibration and optimal denoising. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pp. 394–403. PMLR, 2018.

D Basu, C Dimitrakakis, and A Tossou. Privacy in multi-armed bandits: Fundamental definitions and lower bounds. *arXiv preprint arXiv:1905.12298*, 2019.

Lucas Bourtoule, Varun Chandrasekaran, Christopher A Choquette-Choo, Hengrui Jia, Adelin Travers, Baiwu Zhang, David Lie, and Nicolas Papernot. Machine unlearning. In *IEEE Symposium on Security and Privacy (S&P 2021)*, pp. 141–159, 2021.

Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *IEEE Symposium on Security and Privacy (S&P 2015)*, pp. 463–480, 2015.

Nicholas Carlini, Chang Liu, Úlfar Erlingsson, Jernej Kos, and Dawn Song. The secret sharer: Evaluating and testing unintended memorization in neural networks. In *USENIX Security Symposium (USENIX 2019)*, pp. 267–284, 2019.

Minmin Chen, Alex Beutel, Paul Covington, Sagar Jain, Francois Belletti, and Ed H Chi. Top-k off-policy correction for a reinforce recommender system. In *ACM International Conference on Web Search and Data Mining (WSDM 2019)*, pp. 456–464, 2019.

Xiaoyu Chen, Kai Zheng, Zixin Zhou, Yunchang Yang, Wei Chen, and Liwei Wang. (locally) differentially private combinatorial semi-bandits. In *International Conference on Machine Learning (ICML 2020)*, pp. 1757–1767, 2020.

Aloni Cohen, Adam Smith, Marika Swanberg, and Prashant Nalini Vasudevan. Control, confidentiality, and the right to be forgotten. In *Proceedings of the Conference on Computer and Communications Security (CCS 2023)*, pp. 3358–3372, 2023.

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

Antonio Ginart, Melody Guan, Gregory Valiant, and James Y Zou. Making ai forget you: Data deletion in machine learning. *Advances in Neural Information Processing Systems*, 32:3513–3526, 2019.

Chuan Guo, Tom Goldstein, Awni Hannun, and Laurens Van Der Maaten. Certified data removal from machine learning models. In *International Conference on Machine Learning (ICML 2020)*, pp. 3832–3842, 2020.

Varun Gupta, Christopher Jung, Seth Neel, Aaron Roth, Saeed Sharifi-Malvajerdi, and Chris Waites. Adaptive machine unlearning. *Advances in Neural Information Processing Systems*, 34:16319–16330, 2021.

Yaxi Hu, Bernhard Schölkopf, and Amartya Sanyal. Online learning and unlearning. *arXiv preprint arXiv:2505.08557*, 2025.

Zachary Izzo, Mary Anne Smart, Kamalika Chaudhuri, and James Zou. Approximate data deletion from machine learning models. In *International Conference on Artificial Intelligence and Statistics (AISTATS 2021)*, pp. 2008–2016, 2021.

Jonathan Lee, George Tucker, Ofir Nachum, and Bo Dai. Model selection in batch policy optimization. In *International Conference on Machine Learning (ICML 2022)*, pp. 12542–12569, 2022.

Gene Li, Cong Ma, and Nati Srebro. Pessimism for offline linear contextual bandits using $\ell_p$ confidence sets. *Advances in Neural Information Processing Systems*, 35:20974–20987, 2022.

Jiaqi Liu, Jian Lou, Zhan Qin, and Kui Ren. Certified minimax unlearning with generalization rates and deletion capacity. *Advances in Neural Information Processing Systems*, 36:62821–62852, 2023.

Wenyan Liu, Juncheng Wan, Xiaoling Wang, Weinan Zhang, Dell Zhang, and Hang Li. Forgetting fast in recommender systems. *arXiv preprint arXiv:2208.06875*, 2022.

Xutong Liu, Xiangxiang Dai, Jinhang Zuo, Siwei Wang, Carlee Joe-Wong, John Lui, and Wei Chen. Offline learning for combinatorial multi-armed bandits. *International Conference on Machine Learning (ICML 2025)*, 267:38251–38289, 2025.

Ronak Mehta, Sourav Pal, Vikas Singh, and Sathya N Ravi. Deep unlearning via randomized conditionally independent hessians. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR 2022)*, pp. 10422–10431, 2022.

Nikita Mishra and Abhradeep Thakurta. Nearly optimal differentially private stochastic multi-arm bandits. In *Proceedings of the Conference on Uncertainty in Artificial Intelligence (UAI 2015)*, pp. 592–601, 2015.

Seth Neel, Aaron Roth, and Saeed Sharifi-Malvajerdi. Descent-to-delete: Gradient-based methods for machine unlearning. In *Algorithmic Learning Theory (ALT 2021)*, pp. 931–962, 2021.

Paria Rashidinejad, Banghua Zhu, Cong Ma, Jiantao Jiao, and Stuart Russell. Bridging offline reinforcement learning and imitation learning: A tale of pessimism. *Advances in Neural Information Processing Systems*, 34:11702–11716, 2021.

Wenbo Ren, Xingyu Zhou, Jia Liu, and Ness B Shroff. Multi-armed bandits with local differential privacy. *arXiv preprint arXiv:2007.03121*, 2020.

Ayush Sekhari, Jayadev Acharya, Gautam Kamath, and Ananda Theertha Suresh. Remember what you want to forget: Algorithms for machine unlearning. *Advances in Neural Information Processing Systems*, 34:18075–18086, 2021.

Reza Shokri, Marco Stronati, Congzheng Song, and Vitaly Shmatikov. Membership inference attacks against machine learning models. In *IEEE Symposium on Security and Privacy (S&P 2017)*, pp. 3–18, 2017.

Vinith Suriyakumar and Ashia C Wilson. Algorithms that approximate data removal: New results and limitations. *Advances in Neural Information Processing Systems*, 35:18892–18903, 2022.

Gary Tang, Jiangwei Pan, Henry Wang, and Justin Basilico. Reward innovation for long-term member satisfaction. In *ACM Conference on Recommender Systems (RecSys 2023)*, pp. 396–399, 2023.

Youming Tao, Yulian Wu, Peng Zhao, and Di Wang. Optimal rates of (locally) differentially private heavy-tailed multi-armed bandits. In *International Conference on Artificial Intelligence and Statistics (AISTATS 2022)*, pp. 1546–1574, 2022.

Enayat Ullah, Tung Mai, Anup Rao, Ryan A Rossi, and Raman Arora. Machine unlearning via algorithmic stability. In *Conference on Learning Theory (COLT 2021)*, pp. 4126–4142, 2021.

Yifei Wang, Tavor Z Baharav, Yanjun Han, Jiantao Jiao, and David Tse. Beyond the best: Estimating distribution functionals in infinite-armed bandits. *Advances in Neural Information Processing Systems*, 35:9262–9273, 2022.

Chenjun Xiao, Yifan Wu, Jincheng Mei, Bo Dai, Tor Lattimore, Lihong Li, Csaba Szepesvari, and Dale Schuurmans. On the optimality of batch policy optimization algorithms. In *International Conference on Machine Learning (ICML 2021)*, pp. 11362–11371, 2021.

Dayong Ye, Tianqing Zhu, Congcong Zhu, Derui Wang, Kun Gao, Zewei Shi, Sheng Shen, Wanlei Zhou, and Minhui Xue. Reinforcement unlearning. To appear in *Network and Distributed System Security Symposium (NDSS 2025)*, 2025.

Hongbang Yuan, Zhuoran Jin, Pengfei Cao, Yubo Chen, Kang Liu, and Jun Zhao. Towards robust knowledge unlearning: An adversarial framework for assessing and improving unlearning robustness in large language models. *Proceedings of the AAAI Conference on Artificial Intelligence (AAAI 2025)*, 39:25769–25777, 2025.

Zijie Zhang, Yang Zhou, Xin Zhao, Tianshi Che, and Lingjuan Lyu. Prompt certified machine unlearning with randomized gradient smoothing and quantization. *Advances in Neural Information Processing Systems*, 35:13433–13455, 2022.

Kai Zheng, Tianle Cai, Weiran Huang, Zhenguo Li, and Liwei Wang. Locally differentially private (contextual) bandits learning. *Advances in Neural Information Processing Systems*, 33:12300–12310, 2020.

Xingyu Zhou and Komo Wei ZHANG. Locally private and robust multi-armed bandits. *Advances in Neural Information Processing Systems*, 37:7362–7399, 2024.

# A Missing Proofs in Section 3

## A.1 Algorithms and Upper Bounds for Single-Source Unlearning

Remember that the actual unlearning request $U$ is from $a_0$, $|U| = k$, $D' = D \setminus U$ and $N(a_0)$ is the sample count of $a_0$ in $D$. We present the single-source unlearning algorithm when the unlearning request is $\emptyset$.

**Algorithm 3** Single-source unlearning algorithm

**Input:** Request descriptor $\rho(U) := (a_0, k)$, additional statistics $T(D \setminus U)$: $\widehat{\boldsymbol{f}}(D \setminus U)$, $\widehat{\boldsymbol{\mu}}'$, $\widehat{\boldsymbol{N}}'$, a confidence level: $\tau \in (0,1)$, $\varepsilon \in (0,\infty)$, $\delta \in (0,1)$

Compute $\widehat{\boldsymbol{f}}(D') \leftarrow \widehat{\boldsymbol{f}}(D \setminus U)$, $\widehat{\boldsymbol{N}} \leftarrow \widehat{\boldsymbol{N}}' + k\boldsymbol{I}\{a = a_0\}$
Compute $\sigma \leftarrow \frac{3k\gamma}{2N(a_0)}$, $\beta \leftarrow \sqrt{2\ln(1+N\sigma)}$ and $\gamma_0$
**if** $\max_{a \neq a_0}\{\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}}\} < \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$ and $\gamma < \gamma_0$ **then**
    Sample $\nu \in \mathbb{R}$ from $\mathcal{N}(0, \sigma^2)$
    Set $\widehat{\boldsymbol{f}}_{(a_0)}(D') \leftarrow \widehat{\boldsymbol{f}}_{(a_0)}(D \setminus U) + \nu - \beta\sigma$ ▷ *Downward-shifted Gaussian mechanism*
**else**
    Compute the new penalty $b'(a_0) \leftarrow \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$
    Set $\widehat{\boldsymbol{f}}_{(a_0)}(D') \leftarrow \hat{\mu}'(a_0) - b'(a_0)$ ▷ *Rollback*

**Output:** $\hat{a}' = \arg\max_a \widehat{\boldsymbol{f}}_{(a)}(D')$

Note that the above algorithm adds the downward-shifted Gaussian noise only if Algorithm 2 adds the downward-shifted Gaussian noise, and these noises are sampled from the same distribution. Next we prove $(\varepsilon, \delta)$-BUL for Algorithm 2 and Algorithm 1. The first step is to calculate $\Delta_{\boldsymbol{f}}$, which is the $\ell_2$ sensitivity of $\boldsymbol{f}$.

**Lemma 10.** *For any dataset $D$, any unlearning request $U$ from $a_0$, when $k \leq N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$, we have $\Delta_{\boldsymbol{f}} \leq \frac{3k}{2N(a_0)}$.*

*Proof.* For any dataset $D$, after the unlearning of $U \subseteq D$, $\boldsymbol{f}_{(a_0)}(D')$ becomes $\hat{\mu}'(a_0) - b'(a_0)$ where $N'(a_0) = N(a_0) - k$, $\hat{\mu}'(a_0) = \frac{\hat{\mu}(a_0)\cdot N(a_0) - \sum_{(a_0,r_i)\in U} r_i}{N'(a_0)}$ and $b'(a_0) = \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$. For $a \neq a_0$, $\boldsymbol{f}_{(a)}(D') = \boldsymbol{f}_{(a)}(D)$. Now we calculate the sensitivity when the dataset changes from $D$ to $D \setminus U = D'$:

$$
\begin{aligned}
\Delta_{\boldsymbol{f}} &= \max_{D,D'} \|\boldsymbol{f}(D) - \boldsymbol{f}(D')\|_2 \\
&= \max_{D,D'} \left| \hat{\mu}(a_0) - \frac{\hat{\mu}(a_0)\cdot N(a_0) - \sum_{(a_0,r_i)\in U} r_i}{N(a_0) - k} \right. \\
&\qquad \left. + \sqrt{\ln\frac{m}{\tau}} \left( \sqrt{\frac{1}{2(N(a_0)-k)}} - \sqrt{\frac{1}{2N(a_0)}} \right) \right| \\
&\leq \max_{D,D'} \left| \frac{N(a_0)\cdot \sum_{(a_0,r_i)\in U} r_i - k\sum_{(a_0,r_i)\in D} r_i}{N(a_0)\,(N(a_0)-k)} + \frac{k\sqrt{\ln\frac{m}{\tau}}}{2\,(N(a_0)-k)\,\sqrt{2N(a_0)}} \right| \\
&\overset{(a)}{=} \left| \frac{k}{N(a_0)} + \frac{k\sqrt{\ln\frac{m}{\tau}}}{2\,(N(a_0)-k)\,\sqrt{2N(a_0)}} \right| \overset{(b)}{\leq} \frac{3k}{2N(a_0)},
\end{aligned}
$$

where $(a)$ comes from choosing $r_i = 1, \forall (a_0, r_i) \in U$ and $r_j = 0, \forall (a_0, r_j) \in D \setminus U$, $(b)$ comes from $k \leq N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$. □

**Theorem 11** (Theorem 3 restated)**.** *Algorithm 2 and Algorithm 1 are $(\varepsilon, \delta)$-unlearning when $k \leq N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$ under both $\mathcal{M}_f$ and $\mathcal{M}_d$.*

*Proof.* Let $D' = D \setminus U$ and define the sensitivity upper bound $\overline{\Delta}_{\boldsymbol{f}} := \frac{3k}{2N(a_0)}$. By Lemma 10, since $U$ is a single-source request from $a_0$,

$$\left|\boldsymbol{f}_{(a_0)}(D) - \boldsymbol{f}_{(a_0)}(D')\right| = \|\boldsymbol{f}(D) - \boldsymbol{f}(D')\|_2 \leq \Delta_{\boldsymbol{f}} \leq \overline{\Delta}_{\boldsymbol{f}}.$$

Moreover, $\boldsymbol{f}_{(a)}(D) = \boldsymbol{f}_{(a)}(D')$ for every $a \neq a_0$.

Let $\widehat{\boldsymbol{f}}_U$ denote the LCB vector released by the actual execution $\pi'([U, \emptyset], T(D))$, and let $\widehat{\boldsymbol{f}}_\emptyset$ denote the corresponding vector released by the reference execution $\pi'([\emptyset, \rho(U)], T(D'))$.

We first observe that the switching decision is identical in the two executions. Indeed, the retained-data screening condition

$$\max_{a \neq a_0} \left\{ \hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}} \right\} < \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$$

depends only on the retained-data statistics $\widehat{\boldsymbol{\mu}}'$ and $\widehat{\boldsymbol{N}}'$. Moreover, $\gamma$ is determined by the public privacy parameters, while the reference execution recovers $N(a_0) = N'(a_0) + k$ from the retained count and the request descriptor. Hence, the comparison $\gamma < \gamma_0$ is also identical in the two executions.

Suppose first that $\max_{a \neq a_0} \left\{ \hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}} \right\} < \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$ and $\gamma < \gamma_0$, so that both executions apply the downward-shifted Gaussian mechanism. In the actual execution, $\widehat{\boldsymbol{f}}_{U,(a_0)} = \boldsymbol{f}_{(a_0)}(D) + \nu_U - \beta\sigma$, whereas in the reference execution, $\widehat{\boldsymbol{f}}_{\emptyset,(a_0)} = \boldsymbol{f}_{(a_0)}(D') + \nu_\emptyset - \beta\sigma$, where $\nu_U, \nu_\emptyset \sim \mathcal{N}(0, \sigma^2)$ are sampled from the same distribution. The deterministic downward shift is identical in the two executions therefore does not change the distance between the two Gaussian centers:

$$\left|\left(\boldsymbol{f}_{(a_0)}(D) - \beta\sigma\right) - \left(\boldsymbol{f}_{(a_0)}(D') - \beta\sigma\right)\right| = \left|\boldsymbol{f}_{(a_0)}(D) - \boldsymbol{f}_{(a_0)}(D')\right| \leq \overline{\Delta}_{\boldsymbol{f}}.$$

Moreover, Algorithm 2 sets $\sigma = \frac{3k\gamma}{2N(a_0)} = \gamma\overline{\Delta}_{\boldsymbol{f}}$. Consequently,

$$\Phi\left(\frac{\overline{\Delta}_{\boldsymbol{f}}}{2\sigma} - \frac{\varepsilon\sigma}{\overline{\Delta}_{\boldsymbol{f}}}\right) - e^{\varepsilon}\Phi\left(-\frac{\overline{\Delta}_{\boldsymbol{f}}}{2\sigma} - \frac{\varepsilon\sigma}{\overline{\Delta}_{\boldsymbol{f}}}\right) = \Phi\left(\frac{1}{2\gamma} - \varepsilon\gamma\right) - e^{\varepsilon}\Phi\left(-\frac{1}{2\gamma} - \varepsilon\gamma\right) \leq \delta,$$

where the last inequality follows from (1). Since the analytic Gaussian privacy loss is monotone in the distance between the two centers, Theorem 8 and Lemma 7 of Balle & Wang (2018) imply that, for every measurable set $B \subseteq \mathbb{R}$,

$$\mathbf{Pr}\left[\widehat{\boldsymbol{f}}_{U,(a_0)} \in B\right] \leq e^{\varepsilon}\mathbf{Pr}\left[\widehat{\boldsymbol{f}}_{\emptyset,(a_0)} \in B\right] + \delta, \qquad \mathbf{Pr}\left[\widehat{\boldsymbol{f}}_{\emptyset,(a_0)} \in B\right] \leq e^{\varepsilon}\mathbf{Pr}\left[\widehat{\boldsymbol{f}}_{U,(a_0)} \in B\right] + \delta.$$

For every $a \neq a_0$, the two executions have the same coordinate $\boldsymbol{f}_{(a)}(D) = \boldsymbol{f}_{(a)}(D')$.

Appending these identical coordinates preserves $(\varepsilon, \delta)$ indistinguishability. Thus, for every measurable set $R \subseteq \mathbb{R}^m$,

$$\mathbf{Pr}\left[\widehat{\boldsymbol{f}}_U \in R\right] \leq e^{\varepsilon}\mathbf{Pr}\left[\widehat{\boldsymbol{f}}_\emptyset \in R\right] + \delta, \qquad \mathbf{Pr}\left[\widehat{\boldsymbol{f}}_\emptyset \in R\right] \leq e^{\varepsilon}\mathbf{Pr}\left[\widehat{\boldsymbol{f}}_U \in R\right] + \delta.$$

Suppose instead that the retained-data screening condition and $\gamma < \gamma_0$ do not both hold. Both executions perform rollback and reconstruct the LCB vector on the retained dataset $D'$. Hence, $\widehat{\boldsymbol{f}}_U = \boldsymbol{f}(D') = \widehat{\boldsymbol{f}}_\emptyset$, and consequently $\mathbf{Pr}\left[\widehat{\boldsymbol{f}}_U \in R\right] = \mathbf{Pr}\left[\widehat{\boldsymbol{f}}_\emptyset \in R\right]$ for every measurable set $R \subseteq \mathbb{R}^m$. Finally, the output arm is obtained through the deterministic map $g(\boldsymbol{f}) := \arg\max_{a \in \mathcal{A}} \boldsymbol{f}_{(a)}$.

Applying the post-processing property Dwork et al. (2014) to the two LCB vectors gives, for every $A \subseteq \mathcal{A}$,

$$\mathbf{Pr}\left[\pi'([U, \emptyset], T(D)) \in A\right] \leq e^{\varepsilon}\mathbf{Pr}\left[\pi'([\emptyset, \rho(U)], T(D')) \in A\right] + \delta,$$

and

$$\mathbf{Pr}\left[\pi'([\emptyset, \rho(U)], T(D')) \in A\right] \leq e^{\varepsilon}\mathbf{Pr}\left[\pi'([U, \emptyset], T(D)) \in A\right] + \delta.$$

Therefore, Algorithm 2 and Algorithm 1 satisfy $(\varepsilon, \delta)$-unlearning. Since the argument is pointwise in $D$ and does not depend on how the offline dataset is generated, the result holds under both $\mathcal{M}_f$ and $\mathcal{M}_d$. □

**Theorem 12** (Theorem 4 restated)**.** *Consider any offline dataset $D(\boldsymbol{N})$ satisfying $N(a^*) \geq N_*$, and any reward-independent unlearning request $U$ of size $k$ from $a_0$. When $k \leq N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$, by setting $\tau = \frac{1}{N}$, the arm $\hat{a}'$ returned by Algorithm 2 satisfies when $\gamma < \gamma_0$,*

$$\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) = O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)}} + \frac{k\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N(a_0)}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right).$$

*while when $\gamma \geq \gamma_0$,*

$$\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) = O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)-k}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right).$$

*Proof.* Let $D' = D \setminus U$. Denote by $\mathcal{E}_1$ and $\mathcal{E}_2$ the simultaneous confidence event on $D$ and $D'$:

$$\mathcal{E}_1 = \left\{|\hat{\mu}(a) - \mu(a)| \leq \sqrt{\frac{\ln\frac{m}{\tau}}{2N(a)}},\ \forall a \in \mathcal{A}\right\}, \qquad \mathcal{E}_2 = \left\{|\hat{\mu}'(a) - \mu(a)| \leq \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}},\ \forall a \in \mathcal{A}\right\}.$$

Applying Hoeffding's inequality to each arm and taking a union bound gives $\mathbf{Pr}\left[\mathcal{E}_1^c\right] \leq 2\tau$ and $\mathbf{Pr}\left[\mathcal{E}_2^c\right] \leq 2\tau$.

For convenience, we define

$$\mathcal{S} := \left\{\max_{a \neq a_0}\left\{\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}}\right\} < \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}\right\} \cap \{\gamma < \gamma_0\}. \tag{3}$$

We first consider the case where $\mathcal{S}$ holds. Then Algorithm 2 takes the downward-shifted Gaussian branch. Recall that $\sigma = \frac{3k\gamma}{2N(a_0)}$, $\beta = \sqrt{2\ln(1+N\sigma)}$.

We first consider $a_0 = a^*$. On $\mathcal{E}_1$, for every arm $a \in \mathcal{A}$, $\widehat{\boldsymbol{f}}_{(a)}(D) = \hat{\mu}(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N(a)}} \leq \mu(a)$, while

$$\widehat{\boldsymbol{f}}_{(a^*)}(D) = \hat{\mu}(a^*) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N(a^*)}} \geq \mu(a^*) - 2\sqrt{\frac{\ln\frac{m}{\tau}}{2N(a^*)}} = \mu(a^*) - \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}}.$$

If $\hat{a}' = a^*$, the sub-optimality is zero. Otherwise, since only the coordinate corresponding to $a^* = a_0$ is perturbed, the definition of $\hat{a}'$ implies $\widehat{\boldsymbol{f}}_{(\hat{a}')}(D) \geq \widehat{\boldsymbol{f}}_{(a^*)}(D) + \nu - \beta\sigma$.

Therefore, $\mu(\hat{a}') \geq \mu(a^*) - \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}} + \nu - \beta\sigma$, and hence both cases are summarized by

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}} + (\beta\sigma - \nu)_+.$$

Taking expectation with respect to the Gaussian randomness, we have

$$\mathbf{E}_\nu\left[(\beta\sigma - \nu)_+\right] = \sigma\mathbf{E}\left[(\beta - Z)_+\right] \leq \sigma\left(\beta + \frac{1}{\sqrt{2\pi}}\right), \qquad Z \sim \mathcal{N}(0,1).$$

Therefore, accounting for the failure of $\mathcal{E}_1$ by using that the sub-optimality is at most one,

$$\mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}\right]\right] \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)}} + \sigma\left(\beta + \frac{1}{\sqrt{2\pi}}\right) + \mathbf{Pr}\left[\mathcal{E}_1^c\right]$$

$$\leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)}} + \sigma\left(\beta + \frac{1}{\sqrt{2\pi}}\right) + 2\tau = O\left(\sqrt{\frac{\ln\frac{m}{\tau}}{N(a_0)}} + \frac{k\gamma}{N(a_0)}\left[1 + \sqrt{\ln\left(1 + \frac{Nk\gamma}{N(a_0)}\right)}\right] + \tau\right).$$

We next consider $a_0 \neq a^*$. If $\hat{a}' \neq a_0$, both $\hat{a}'$ and $a^*$ are unperturbed. By the maximality of $\hat{a}'$, $\widehat{\boldsymbol{f}}_{(\hat{a}')}(D) \geq \widehat{\boldsymbol{f}}_{(a^*)}(D)$, which gives $\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}}$. If $\hat{a}' = a_0$, then

$$\widehat{\boldsymbol{f}}_{(a_0)}(D) + \nu - \beta\sigma \geq \widehat{\boldsymbol{f}}_{(a^*)}(D).$$

Using $\widehat{\boldsymbol{f}}_{(a_0)}(D) \leq \mu(a_0)$ and $\widehat{\boldsymbol{f}}_{(a^*)}(D) \geq \mu(a^*) - \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}}$, we obtain

$$\mu(a^*) - \mu(a_0) \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}} + \nu - \beta\sigma.$$

Combining the two subcases,

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}} + (\nu - \beta\sigma)_+.$$

Moreover, let $\phi$ and $\Phi$ denote the probability density function and cumulative distribution function of the standard Gaussian distribution, respectively. Since $\mathbf{E}_\nu\left[(\nu - \beta\sigma)_+\right] = \sigma\left[\phi(\beta) - \beta(1 - \Phi(\beta))\right] \leq \sigma\phi(\beta)$, and $e^{-\frac{\beta^2}{2}} = \frac{1}{1+N\sigma}$, we obtain $\mathbf{E}_\nu\left[(\nu - \beta\sigma)_+\right] \leq \frac{\sigma}{\sqrt{2\pi}(1+N\sigma)} \leq \frac{1}{\sqrt{2\pi}N}$. Therefore,

$$\begin{aligned}\mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}\right]\right] &\leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}} + \frac{1}{\sqrt{2\pi}N} + \mathbf{Pr}\left[\mathcal{E}_1^c\right] \\ &\leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}} + \frac{1}{\sqrt{2\pi}N} + 2\tau = O\left(\sqrt{\frac{\ln\frac{m}{\tau}}{N_*}} + \frac{1}{N} + \tau\right),\end{aligned}$$

where the second inequality follows from $N(a^*) \geq N_*$.

Combining the cases $a_0 = a^*$ and $a_0 \neq a^*$, the contribution of the Gaussian branch satisfies

$$\mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}\right]\right] = O\Bigg(\max\left\{\sqrt{\frac{\ln\frac{m}{\tau}}{N(a_0)}} + \frac{k\gamma}{N(a_0)}\left[1 + \sqrt{\ln\left(1 + \frac{Nk\gamma}{N(a_0)}\right)}\right], \sqrt{\frac{\ln\frac{m}{\tau}}{N_*}}\right\} + \frac{1}{N} + \tau\Bigg).$$

Setting $\tau = \frac{1}{N}$ and using $N_* \leq N$, the $O(\frac{1}{N})$ terms are dominated by the displayed statistical term.

We next bound the contribution of $\mathcal{S}^c$ while still assuming $\gamma < \gamma_0$. In this regime, $\mathcal{S}^c$ means that the screening condition fails, so Algorithm 2 executes rollback.

If $a_0 = a^*$, failure of the screening condition implies that there exists $a \neq a^*$ such that

$$\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}} \geq \hat{\mu}'(a^*) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a^*)}}.$$

On $\mathcal{E}_2$,

$$\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}} \leq \mu(a) \leq \mu(a^*) \leq \hat{\mu}'(a^*) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a^*)}}.$$

Hence all inequalities hold with equality. In particular, $\mu(a) = \mu(a^*)$, $\widehat{\boldsymbol{f}}_{(a)}(D') = \mu(a^*)$. Moreover, for every $b \in \mathcal{A}$, $\widehat{\boldsymbol{f}}_{(b)}(D') \leq \mu(b) \leq \mu(a^*)$. Thus every maximizer of the retained-data LCB is optimal, and rollback incurs zero sub-optimality on $\mathcal{S}^c \cap \mathcal{E}_2$. Consequently,

$$\mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}^c\right]\right] \leq 2\tau, \qquad a_0 = a^*.$$

If $a_0 \neq a^*$, deletion does not affect the optimal-arm count: $N'(a^*) = N(a^*) \geq N_*$.

On $\mathcal{E}_2$, the standard LCB argument gives

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N'(a^*)}} \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}}.$$

Then we have

$$\mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}^c\right]\right] \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}} + 2\tau, \qquad a_0 \neq a^*.$$

Hence, when $\gamma < \gamma_0$,

$$\begin{aligned}\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) &= O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)}} + \frac{k\gamma}{N(a_0)}\left[1 + \sqrt{\ln\left(1 + \frac{Nk\gamma}{N(a_0)}\right)}\right], \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right) \\ &= O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)}} + \frac{k\gamma\left(1 + \sqrt{\ln(1+N\gamma)}\right)}{N(a_0)}, \sqrt{\frac{\ln(Nm)}{N_*}}.\right\}\right)\end{aligned}$$

We now consider the case where $\mathcal{S}$ does not hold. In this case, Algorithm 2 executes rollback and returns the LCB maximizer on $D'$. On $\mathcal{E}_2$, for every arm $a \in \mathcal{A}$, $\widehat{\boldsymbol{f}}_{(a)}(D') \leq \mu(a)$, while $\widehat{\boldsymbol{f}}_{(a^*)}(D') \geq \mu(a^*) - \sqrt{\frac{2\ln\frac{m}{\tau}}{N'(a^*)}}$. Since $\hat{a}' = \arg\max_{a\in\mathcal{A}} \widehat{\boldsymbol{f}}_{(a)}(D')$, we have $\widehat{\boldsymbol{f}}_{(\hat{a}')}(D') \geq \widehat{\boldsymbol{f}}_{(a^*)}(D')$, and consequently $\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N'(a^*)}}$.

If $a_0 = a^*$, then $N'(a^*) = N(a_0) - k$, and thus

$$\mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}^c\right]\right] \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0) - k}} + \mathbf{Pr}\left[\mathcal{E}_2^c\right] \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0) - k}} + 2\tau.$$

If $a_0 \neq a^*$, the deletion does not change the number of observations of the optimal arm, so $N'(a^*) = N(a^*) \geq N_*$. Therefore,

$$\mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}^c\right]\right] \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}} + \mathbf{Pr}\left[\mathcal{E}_2^c\right] \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}} + 2\tau.$$

Taking the worse of the two possible deletion sources, the rollback branch satisfies

$$\mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}^c\right]\right] = O\left(\max\left\{\sqrt{\frac{\ln\frac{m}{\tau}}{N(a_0) - k}}, \sqrt{\frac{\ln\frac{m}{\tau}}{N_*}}\right\} + \tau\right).$$

Finally, when $\gamma \geq \gamma_0$, setting $\tau = \frac{1}{N}$ gives

$$\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) \leq \mathbf{E}\left[(\mu(a^*) - \mu(\hat{a}'))\,\mathbb{1}\left[\mathcal{S}^c\right]\right] = O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0) - k}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right).$$

□

We first give the lemma for binomial random variable $N(a^*)$.

**Lemma 13** (**Lemma 4** in (Rashidinejad et al. (2021))). *With probability at least* $1 - e^{-\frac{Nd(a^*)}{8}}$*, one has* $N(a^*) \geq \frac{1}{2}Nd(a^*)$.

**Theorem 14** (Theorem 5 restated)**.** *Consider any offline dataset $D \sim \mathcal{D}^N$, where the behavior policy satisfies $d(a^*) \geq \frac{1}{C^*}$ for some $C^* > 1$, and any reward-independent unlearning request $U$ of size $k$ from $a_0$. When $N > 8C^* \ln N$ and $k \leq N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$, by setting $\tau = \frac{1}{N}$, the arm $\hat{a}'$ returned by Algorithm 2 satisfies*

$$\begin{aligned}&\texttt{SubOpt}(N, C^*, k, \varepsilon, \delta)\\ &= O\left(\min\left\{\sqrt{\frac{C^*\ln(Nm)}{\max\{1, N-2kC^*\}}}, \sqrt{\frac{C^*\ln(Nm)}{N}} + \frac{k\gamma C^*(1+\sqrt{\ln(1+k\gamma C^*)})}{N}\right\}\right).\end{aligned}$$

*Proof.* We first consider the case where $a_0 = a^*$. Denote by $\mathcal{E}_1$ the event that $N(a^*) \geq \frac{N}{2C^*}$. By Lemma 13 and $d(a^*) \geq \frac{1}{C^*}$, $\mathbf{Pr}[\mathcal{E}_1^c] \leq e^{-\frac{N}{8C^*}} \leq \frac{1}{N} = \tau$, where the second inequality follows from $N > 8C^* \ln N$. As in the fixed-sample analysis above, let $\mathcal{E}_2$ and $\mathcal{E}_3$ denote the simultaneous confidence events on $D$ and $D' = D \setminus U$, respectively:

$$\mathcal{E}_2 = \left\{|\hat{\mu}(a) - \mu(a)| \leq \sqrt{\frac{\ln\frac{m}{\tau}}{2N(a)}},\ \forall a \in \mathcal{A}\right\}, \qquad \mathcal{E}_3 = \left\{|\hat{\mu}'(a) - \mu(a)| \leq \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}},\ \forall a \in \mathcal{A}\right\}.$$

Conditioning on the realized arm labels and the reward-independent deletion locations, the retained rewards remain independent samples from their corresponding arm distributions. Hence, by Hoeffding's inequality and a union bound, $\mathbf{Pr}[\mathcal{E}_2^c] \leq 2\tau$, $\mathbf{Pr}[\mathcal{E}_3^c] \leq 2\tau$. Let $\mathcal{E} = \mathcal{E}_1 \cap \mathcal{E}_2 \cap \mathcal{E}_3$. Then $\mathbf{Pr}[\mathcal{E}^c] \leq 5\tau$. Since rewards lie in $[0,1]$, the contribution of $\mathcal{E}^c$ to the expected sub-optimality is at most $5\tau$. On $\mathcal{E}$, since $a_0 = a^*$, $N(a_0) = N(a^*) \geq \frac{N}{2C^*}$.

For the realized count $N(a_0)$, define the Gaussian and rollback bounds

$$R_g = \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)}} + \sigma\left(\beta + \frac{1}{\sqrt{2\pi}}\right), \quad R_r = \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)-k}},$$

where $\sigma = \frac{3k\gamma}{2N(a_0)}$, $\beta = \sqrt{2\ln(1+N\sigma)}$.

By the fixed-sample analysis above, when $\mathcal{S}$ ($\mathcal{S}$ follows definition in (3)) holds, the downward-shifted Gaussian branch satisfies $\mathbf{E}_\nu[\mu(a^*) - \mu(\hat{a}')] \leq R_g$. Moreover, the second condition in $\mathcal{S}$ is exactly

$$\sigma\left(\beta + \frac{1}{\sqrt{2\pi}}\right) < \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)-k}} - \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)}},$$

which is equivalent to $R_g < R_r$. Thus, whenever the first condition in $\mathcal{S}$ holds, Algorithm 2 selects the better of the Gaussian and rollback bounds.

It remains to consider the case where the first condition in $\mathcal{S}$ does not hold. Then there exists an arm $a \neq a^*$ such that

$$\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}} \geq \hat{\mu}'(a^*) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a^*)}}.$$

On $\mathcal{E}_3$,

$$\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}} \leq \mu(a) \leq \mu(a^*) \leq \hat{\mu}'(a^*) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a^*)}}.$$

Therefore all the inequalities above must hold with equality. In particular, $\mu(a) = \mu(a^*)$, and the same argument shows that every maximizer of the retained-data LCB is optimal. Hence rollback incurs zero sub-optimality in this case. Consequently, on $\mathcal{E}$,

$$\mathbf{E}_\nu[\mu(a^*) - \mu(\hat{a}')] \leq \min\{R_g, R_r\}.$$

We next upper-bound $R_g$ and $R_r$ by deterministic quantities depending on $N$ and $C^*$. Since $N(a_0) \geq \frac{N}{2C^*}$, we have $\sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)}} \leq 2\sqrt{\frac{C^*\ln\frac{m}{\tau}}{N}}$, $\sigma \leq \frac{3k\gamma C^*}{N}$, $N\sigma = \frac{3Nk\gamma}{2N(a_0)} \leq 3k\gamma C^*$. There-

fore,

$$R_g \le 2\sqrt{\frac{C^* \ln \frac{m}{\tau}}{N}} + \frac{3k\gamma C^*}{N}\left[\sqrt{2\ln(1+3k\gamma C^*)} + \frac{1}{\sqrt{2\pi}}\right]$$
$$= O\left(\sqrt{\frac{C^* \ln \frac{m}{\tau}}{N}} + \frac{k\gamma C^*(1+\sqrt{\ln(1+k\gamma C^*)})}{N}\right).$$

For rollback, $N(a_0) - k \ge \frac{N}{2C^*} - k = \frac{N-2kC^*}{2C^*}$. Hence, when $N - 2kC^* > 0$, $R_r \le 2\sqrt{\frac{C^* \ln \frac{m}{\tau}}{N-2kC^*}}$.

When $N - 2kC^* \le 0$, the sub-optimality is trivially at most one, so the same big-$O$ bound remains valid after replacing the denominator by $\max\{1, N - 2kC^*\}$. Therefore,

$$R_r = O\left(\sqrt{\frac{C^* \ln \frac{m}{\tau}}{\max\{1, N-2kC^*\}}}\right).$$

Combining the two bounds and setting $\tau = \frac{1}{N}$, we obtain when $a_0 = a^*$,

$$\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right] = O\Bigg(\min\Bigg\{\sqrt{\frac{C^* \ln(Nm)}{\max\{1, N-2kC^*\}}}, \sqrt{\frac{C^* \ln(Nm)}{N}} + \frac{k\gamma C^*(1+\sqrt{\ln(1+k\gamma C^*)})}{N}\Bigg\} + \frac{1}{N}\Bigg).$$

The $O(\frac{1}{N})$ term accounts for the confidence and coverage failures and is absorbed by the leading terms.

We now consider the case where $a_0 \neq a^*$. On $\mathcal{E}_1$, deletion does not change the optimal-arm count, and hence $N'(a^*) = N(a^*) \ge \frac{N}{2C^*}$. If $\mathcal{S}$ holds, the Gaussian analysis in the fixed-sample case gives $\mathbf{E}_\nu\left[\mu(a^*) - \mu(\hat{a}')\right] \le \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}} + \frac{1}{\sqrt{2\pi}N}$. If $\mathcal{S}$ does not hold, Algorithm 2 executes rollback, and the retained-data LCB bound gives $\mu(a^*) - \mu(\hat{a}') \le \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}}$. Therefore, regardless of the branch taken,

$$\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right] = O\left(\sqrt{\frac{C^* \ln(Nm)}{N}}\right).$$

Finally, $\max\{1, N - 2kC^*\} \le N$, so both terms in

$$\min\left\{\sqrt{\frac{C^* \ln(Nm)}{\max\{1, N-2kC^*\}}}, \sqrt{\frac{C^* \ln(Nm)}{N}} + \frac{k\gamma C^*(1+\sqrt{\ln(1+k\gamma C^*)})}{N}\right\}$$

are at least of order $\sqrt{\frac{C^* \ln(Nm)}{N}}$. Hence the bound for $a_0 = a^*$ also provides a valid uniform upper bound when $a_0 \neq a^*$. Therefore, $\texttt{SubOpt}(N, C^*, k, \varepsilon, \delta) = O\left(\min\left\{\sqrt{\frac{C^* \ln(Nm)}{\max\{1,N-2kC^*\}}}, \sqrt{\frac{C^* \ln(Nm)}{N}} + \frac{k\gamma C^*(1+\sqrt{\ln(1+k\gamma C^*)})}{N}\right\}\right)$. □

**How to obtain $\gamma_0$.** We characterize their unique crossing point by $\gamma_0$, defined by

$$\sigma(\beta + \frac{1}{\sqrt{2\pi}}) = \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)-k}} - \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)}},$$

i.e.

$$\frac{3k\gamma_0(\sqrt{2\ln(1+\frac{3k\gamma_0 N}{2N(a_0)})} + \frac{1}{\sqrt{2\pi}})}{2N(a_0)} = \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)-k}} - \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a_0)}}$$

Denote the left-hand side by $B_G$ and the right-hand side by $B_R$. The quantity $B_G$ is monotonically increasing in $\gamma_0$, whereas $B_R$ is independent of $\gamma_0$. Therefore, $\gamma < \gamma_0 \iff B_G < B_R$.

Consequently, $\gamma_0$ serves only as an implicit characterization of the switching threshold. The algorithm does not need to solve for $\gamma_0$ explicitly; it can directly compare $B_G$ and $B_R$ to determine whether to apply the Gaussian mechanism or rollback.

### A.2 Lower Bounds for Single-Source Unlearning

**Theorem 15** (Theorem 6 restated)**.** *Let $\varepsilon \geq 0$, $\delta \leq \frac{1}{8\sqrt{e}}$. For $\boldsymbol{N}$ such that at least two arms have sample counts greater than $N^*$, there exist valid bandit instances and unlearning request $U$ from $a_0$, such that* $\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) = \Omega\left(e^{-\varepsilon}\sqrt{\frac{1}{N(a_0)-k}}\right)$.

*Proof.* We construct two instances $I_1$ and $I_2$ to prove the lower bound, where the rewards are drawn from some Bernoulli distributions supported on $[0, 1]$. Let $\boldsymbol{\mu}_{I_1}$ and $\boldsymbol{\mu}_{I_2}$ represent the vectors of arm mean rewards in $I_1$ and $I_2$. In particular,

$$\boldsymbol{\mu}_{I_1} = (\frac{1}{2} + \Delta, \frac{1}{2}), \quad \boldsymbol{\mu}_{I_2} = (\frac{1}{2} - \Delta, \frac{1}{2}).$$

Here $0 \leq \Delta \leq \frac{1}{5}$ and its value will be specified later. Let $D_1(\boldsymbol{N}), D_2(\boldsymbol{N})$ denote the two datasets respectively, where $a_0 = 1$ and $\boldsymbol{N} = (N(a_0), N - N(a_0))^\top$. In addition, the reward distribution underlying $D_1(\boldsymbol{N})$ is $\boldsymbol{\mu}_{I_1}$, while that underlying $D_2(\boldsymbol{N})$ is $\boldsymbol{\mu}_{I_2}$. Finally, let $U_1, U_2$ be the corresponding unlearning request from $D_1(\boldsymbol{N}), D_2(\boldsymbol{N})$ respectively.

Let $\mathbf{E}_{D,U}[]$ represent the expectation conditioned on $D, U$. From the definition, for any algorithm $\pi, \pi'$, we have

$$\begin{aligned}
\mathbf{E}_{D_1(\boldsymbol{N}),U_1}\left[\mu(a^*) - \mu(\hat{a}')\right] &= \mathbf{Pr}\left[\hat{a}' = 2\right] \cdot \Delta = \mathbf{Pr}\left[\pi'([U_1, \emptyset], T(D_1(\boldsymbol{N}))) = 2\right] \cdot \Delta \\
&\overset{(a)}{\geq} e^{-\varepsilon}\left(\mathbf{Pr}\left[\pi'([\emptyset, \rho(U_1)], T(D_1(\boldsymbol{N}) \setminus U_1)) = 2\right] - \delta\right) \cdot \Delta \\
&= e^{-\varepsilon}\mathbf{Pr}\left[\pi'([\emptyset, \rho(U_1)], T(D_1(\boldsymbol{N}) \setminus U_1)) = 2\right] \cdot \Delta - e^{-\varepsilon}\delta\Delta,
\end{aligned}$$

where $(a)$ is because $\pi$, $\pi'$ are $(\varepsilon, \delta)$-BUL. Similarly for $D_2, U_2$, we have

$$\mathbf{E}_{D_2(\boldsymbol{N}),U_2}\left[\mu(a^*) - \mu(\hat{a}')\right] \geq e^{-\varepsilon}\mathbf{Pr}\left[\pi'([\emptyset, \rho(U_2)], T(D_2(\boldsymbol{N}) \setminus U_2)) = 1\right] \cdot \Delta - e^{-\varepsilon}\delta\Delta.$$

Next we generate $U_1$ by selecting $k$ points in the data points of $a_0$ from $D_1(\boldsymbol{N})$ uniformly and randomly, while $U_2$ is generated in the same way from $D_2(\boldsymbol{N})$. Therefore $U$ and the rewards of each point will be mutually independent and $\rho(U_1) = \rho(U_2) = (a_0, k)$. Denote $\boldsymbol{N}' = (N(a_0) - k, N - N(a_0))^\top$ as the sample-count vector after unlearning request $U$. Then we apply the classic Le Cam's method to give a lower bound of $\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta)$. Denote $\texttt{TV}(\cdot, \cdot)$ as the total variation distance between two distributions, $\texttt{KL}(\cdot\|\cdot)$ as the KL-divergence of two distributions. We have

$$\begin{aligned}
&\max_{i\in\{1,2\}} \mathbf{E}_{D_i(\boldsymbol{N}),U_i}\left[\mu(a^*) - \mu(\hat{a}')\right] \\
&\geq \frac{1}{2}\left(\mathbf{E}_{D_1(\boldsymbol{N}),U_1}\left[\mu(a^*) - \mu(\hat{a}')\right] + \mathbf{E}_{D_2(\boldsymbol{N}),U_2}\left[\mu(a^*) - \mu(\hat{a}')\right]\right) \\
&\geq \frac{e^{-\varepsilon}\Delta}{2}\left(\mathbf{Pr}\left[\pi'([\emptyset, \rho(U_1)], T(D_1(\boldsymbol{N}) \setminus U_1)) = 2\right] + \mathbf{Pr}\left[\pi'([\emptyset, \rho(U_2)], T(D_2(\boldsymbol{N}) \setminus U_2)) = 1\right]\right) \\
&\quad - e^{-\varepsilon}\delta\Delta \\
&\overset{(a)}{=} \frac{e^{-\varepsilon}\Delta}{2}\left(\mathbf{Pr}\left[\pi'([\emptyset, \rho(U_1)], T(D_1(\boldsymbol{N}'))) = 2\right] + \mathbf{Pr}\left[\pi'([\emptyset, \rho(U_2)], T(D_2(\boldsymbol{N}'))) = 1\right]\right) - e^{-\varepsilon}\delta\Delta \\
&\geq \frac{e^{-\varepsilon}\Delta}{2}\left(1 - \texttt{TV}\left(\pi'([\emptyset, \rho(U_1)], T(D_1(\boldsymbol{N}'))), \pi'([\emptyset, \rho(U_2)], T(D_2(\boldsymbol{N}')))\right)\right) - e^{-\varepsilon}\delta\Delta \\
&\overset{(b)}{\geq} \frac{e^{-\varepsilon}\Delta}{4}\exp\left(-\texttt{KL}\left(\pi'([\emptyset, \rho(U_1)], T(D_1(\boldsymbol{N}')))\|\pi'([\emptyset, \rho(U_2)], T(D_2(\boldsymbol{N}')))\right)\right) - e^{-\varepsilon}\delta\Delta \\
&\overset{(c)}{\geq} \frac{e^{-\varepsilon}\Delta}{4}\exp\left(-\texttt{KL}\left(D_1(\boldsymbol{N}')\|D_2(\boldsymbol{N}')\right)\right) - e^{-\varepsilon}\delta\Delta \\
&\overset{(d)}{\geq} \frac{e^{-\varepsilon}\Delta}{4}\exp\left(-10\Delta^2(N(a_0) - k)\right) - e^{-\varepsilon}\delta\Delta,
\end{aligned}$$

where $(a)$ is because $U$ and the reward are mutually independent, $(b)$ is from Bretagnolle–Huber inequality, $(c)$ is from the fact that $\rho(U_1) = \rho(U_2) = (a_0, k)$, so appending the common descriptor

$(a_0, k)$ does not change the KL divergence, and that, conditional on $(a_0, k)$, the two unlearning executions with input $\rho(U_1)$ and $\rho(U_2)$ apply the same randomized mapping to $T(D_1(\boldsymbol{N}'))$ and $T(D_2(\boldsymbol{N}')$. We then apply the data processing inequality, $(d)$ is from direct calculation and the fact that $\ln \frac{1+2\Delta}{1-2\Delta} \le 5\Delta$ when $0 \le \Delta \le \frac{1}{5}$. By choosing $\Delta = \min\left\{\frac{1}{5}, \frac{1}{2}\sqrt{\frac{1}{5(N(a_0)-k)}}\right\}$, when $\delta \le \frac{1}{8\sqrt{e}}$, we have

$$\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta) \ge \frac{e^{-\varepsilon}}{16}\sqrt{\frac{1}{5e(N(a_0)-k)}} = \Omega\left(e^{-\varepsilon}\sqrt{\frac{1}{N(a_0)-k}}\right).$$

□

**Theorem 16** (Theorem 7 restated)**.** *Let $\varepsilon \ge 0$, $\delta \le \frac{1}{8\sqrt{e}}$ and $N \ge 8C^*$. There exist valid bandit instances and unlearning request $U$ from $a_0$, when $C^* \in [2, \infty)$, if $k \le \frac{N}{2C^*}$,*

$$\texttt{SubOpt}(N, C^*, k, \varepsilon, \delta) = \Omega\left(e^{-\varepsilon}\sqrt{\frac{C^*}{N-k}}\right),$$

*while $C^* \in (1, 2)$, if $k \le \frac{N}{2}(1 - \frac{1}{C^*})$,*

$$\texttt{SubOpt}(N, C^*, k, \varepsilon, 0) = \Omega\left((2-C^*)\, e^{-\varepsilon-(N-k)\left(\ln\left(\frac{2}{C^*-1}\right)+\frac{(2-C^*)^2}{15}\right)}\right).$$

*Proof.* For $C^* \ge 2$, we construct two closely related instances $I_1$ and $I_2$ as in Theorem 15, where $\boldsymbol{\mu}_{I_1}$ and $\boldsymbol{\mu}_{I_2}$ represent the vectors of arm mean rewards in $I_1$ and $I_2$. In particular,

$$\boldsymbol{\mu}_{I_1} = (\frac{1}{2} + \Delta, \frac{1}{2}), \quad \boldsymbol{\mu}_{I_2} = (\frac{1}{2} - \Delta, \frac{1}{2}).$$

The only difference is the value of $\Delta$, which will be specified later. For the behavior policy $d$, let $d_{I_1}(1) = d_{I_2}(1) = \frac{1}{C^*}$, $d_{I_1}(2) = d_{I_2}(2) = 1 - \frac{1}{C^*}$. Since $C^* \ge 2$, we have $1 - \frac{1}{C^*} \ge \frac{1}{C^*}$ for $I_2$. Let $D_1(N, C^*)$ denote the dataset obtained by drawing $N$ i.i.d. samples from $d_{I_1} \otimes \boldsymbol{\mu}_{I_1}$ and $D_2(N, C^*)$ denote the dataset which draws $N$ i.i.d. samples from $d_{I_2} \otimes \boldsymbol{\mu}_{I_2}$. Finally, let $U_1, U_2$ be the unlearning request from $D_1(N, C^*), D_2(N, C^*)$ respectively.

We next construct single-source requests. Since the two instances have the same behavior policy, we may couple their action labels so that they are identical. Let

$$X := N_{D_i}(1) = \sum_{i=1}^{N} \mathbb{1}\left[a_i = 1\right], \quad X \sim \text{Binomial}\left(N, \frac{1}{C^*}\right),$$

where the value of $X$ is the same under the coupling for $i \in \{1, 2\}$. Define the deletion source

$$S := \begin{cases} 1, & X \ge k, \\ 2, & X < k. \end{cases}$$

If $X \ge k$, select $k$ observations of arm 1 uniformly at random and delete them. If $X < k$, select $k$ observations of arm 2 uniformly at random and delete them. Use the same deletion randomization under the coupling and denote the resulting requests by $U_1$ and $U_2$. Each realized request deletes exactly $k$ observations from a single arm, and its selection is independent of all rewards. Moreover, under the above coupling,

$$\rho(U_1) = \rho(U_2) = (S, k) \quad \text{almost surely.}$$

Thus the descriptor $(S, k)$ may be random, but it is the same random descriptor under the two reward instances. Let $D_i^- := D_i(N, C^*) \setminus U_i$ for $i \in \{1, 2\}$. Let $\mathbf{E}_{D,U}[]$ denote expectation over the dataset, the reward-independent request randomization, and the internal randomness of the unlearning mechanism. For any learning–unlearning pair $\pi, \pi'$, we have

$$\begin{aligned}\mathbf{E}_{D_1(N,C^*),U_1}\left[\mu(a^*) - \mu(\hat{a}')\right] &= \mathbf{Pr}\left[\pi'([U_1, \emptyset], T(D_1(N, C^*))) = 2\right]\Delta \\ &\overset{(a)}{\ge} e^{-\varepsilon}\left(\mathbf{Pr}\left[\pi'([\emptyset, \rho(U_1)], T(D_1^-)) = 2\right] - \delta\right)\Delta, \end{aligned} \tag{4}$$

where $(a)$ follows from the reverse inequality in Definition 1, applied pointwise to each realization of $D_1(N, C^*)$ and $U_1$, and then averaged. Similarly,

$$\mathbf{E}_{D_2(N,C^*),U_2}\left[\mu(a^*) - \mu(\hat{a}')\right] \geq e^{-\varepsilon}\left(\mathbf{Pr}\left[\pi'([\emptyset, \rho(U_2)], T(D_2^-)) = 1\right] - \delta\right)\Delta. \tag{5}$$

We then apply Le Cam's method. Fix any learning–unlearning pair $(\pi, \pi')$ satisfying Definition 1. Throughout this argument, $\mathcal{L}_i(Y)$ denotes the distribution of a random variable $Y$ under instance $I_i$, and $\mathbf{E}_{I_i}[\cdot]$ denotes expectation under $I_i$, including the randomness of the dataset, the deletion rule, and the algorithms. Under the coupling above, let $A = (a_t)_{t=1}^N$ denote the common action-label sequence, and let $\mathcal{K} \subseteq [N]$ denote the common set of $k$ deleted indices. Write $r_{i,t}$ for the reward at position $t$ under instance $I_i$, so that

$$D_i^- = \{(A_t, r_{i,t}) : t \notin \mathcal{K}\}, \qquad i \in \{1, 2\}.$$

Define $J_k(X) := X - k\mathbb{1}[X \geq k]$ as the number of retained observations from arm 1. Let

$$Q_i := \mathcal{L}_i\left(\pi'([\emptyset, \rho(U_i)], T(D_i^-))\right)$$

denote the reference-output distribution. From (4) and (5), we have

$$\begin{aligned}
&\max_{i\in\{1,2\}} \mathbf{E}_{D_i(N,C^*),U_i}\left[\mu(a^*) - \mu(\hat{a}')\right] \\
&\geq \frac{1}{2}\left(\mathbf{E}_{D_1(N,C^*),U_1}\left[\mu(a^*) - \mu(\hat{a}')\right] + \mathbf{E}_{D_2(N,C^*),U_2}\left[\mu(a^*) - \mu(\hat{a}')\right]\right) \\
&\geq \frac{e^{-\varepsilon}\Delta}{2}\left(Q_1(\{a_2\}) + Q_2(\{a_1\}) - 2\delta\right) \\
&\geq \frac{e^{-\varepsilon}\Delta}{2}\left(1 - \mathtt{TV}(Q_1, Q_2)\right) - e^{-\varepsilon}\delta\Delta \\
&\overset{(a)}{\geq} \frac{e^{-\varepsilon}\Delta}{4}\exp\left(-\mathtt{KL}(Q_1 \| Q_2)\right) - e^{-\varepsilon}\delta\Delta \\
&\overset{(b)}{\geq} \frac{e^{-\varepsilon}\Delta}{4}\exp\left(-\mathtt{KL}(\mathcal{L}_1\big(\rho(U_1), D_1^-\big) \| \mathcal{L}_2\big(\rho(U_2), D_2^-\big))\right) - e^{-\varepsilon}\delta\Delta \\
&\overset{(c)}{\geq} \frac{e^{-\varepsilon}\Delta}{4}\exp\left\{-\mathbf{E}\left[J_k(X)\right]\mathtt{KL}\left(\mathtt{Ber}\left(\frac{1}{2} + \Delta\right) \| \mathtt{Ber}\left(\frac{1}{2} - \Delta\right)\right)\right\} - e^{-\varepsilon}\delta\Delta \\
&\overset{(d)}{\geq} \frac{e^{-\varepsilon}\Delta}{4}\exp\left(-\frac{10(N-k)\Delta^2}{C^*}\right) - e^{-\varepsilon}\delta\Delta.
\end{aligned}$$

$(a)$ follows from the Bretagnolle–Huber inequality. For $(b)$, the reference execution applies the same randomized mapping

$$(\rho, x) \longmapsto \pi'([\emptyset, \rho], T(x))$$

under both instances. Then the data processing inequality gives the result. For $(c)$, the two instances use the same behavior policy and the same label-only deletion rule. Therefore, $\mathcal{L}_1(A, \mathcal{K}) = \mathcal{L}_2(A, \mathcal{K})$. Let $R_{i,-\mathcal{K}} = (R_{i,t})_{t\in[N]\setminus\mathcal{K}}$, with the retained positions listed in increasing order. The pair $(\rho(U_i), D_i^-)$ is a deterministic function of $(A, \mathcal{K}, R_{i,-\mathcal{K}})$: the action labels and deleted indices determine the descriptor, and the retained labels and rewards determine $D_i^-$. Hence, by data processing,

$$\mathtt{KL}(\mathcal{L}_1\big(\rho(U_1), D_1^-\big) \| \mathcal{L}_2\big(\rho(U_2), D_2^-\big)) \leq \mathtt{KL}\left(\mathcal{L}_1(A, \mathcal{K}, R_{1,-\mathcal{K}}) \,\|\, \mathcal{L}_2(A, \mathcal{K}, R_{2,-\mathcal{K}})\right).$$

Conditional on $A$ and $\mathcal{K}$, the descriptor and the retained sample locations are fixed. Because deletion locations are chosen independently of rewards conditional on the action labels, the retained rewards are independent samples from their respective arm distributions. The two conditional reward distributions differ only on arm 1, from which exactly $J_k(X)$ observations remain. Consequently,

$$\mathtt{KL}\left(\mathcal{L}_1(R_{1,-\mathcal{K}} \mid A, \mathcal{K}) \,\|\, \mathcal{L}_2(R_{2,-\mathcal{K}} \mid A, \mathcal{K})\right) = J_k(X)\mathtt{KL}\left(\mathtt{Ber}\left(\frac{1}{2} + \Delta\right) \| \mathtt{Ber}\left(\frac{1}{2} - \Delta\right)\right).$$

Applying the chain rule for KL divergence therefore gives

$$\begin{aligned}
&\mathtt{KL}\left(\mathcal{L}_1(A, \mathcal{K}, R_{1,-\mathcal{K}}) \,\|\, \mathcal{L}_2(A, \mathcal{K}, R_{2,-\mathcal{K}})\right) \\
&= \mathtt{KL}\left(\mathcal{L}_1(A, \mathcal{K}) \,\|\, \mathcal{L}_2(A, \mathcal{K})\right) + \mathbf{E}_{I_1}\left[\mathtt{KL}\left(\mathcal{L}_1(R_{1,-\mathcal{K}} \mid A, \mathcal{K}) \,\|\, \mathcal{L}_2(R_{2,-\mathcal{K}} \mid A, \mathcal{K})\right)\right] \\
&= \mathbf{E}\left[J_k(X)\right]\mathtt{KL}\left(\mathtt{Ber}\left(\frac{1}{2} + \Delta\right) \| \mathtt{Ber}\left(\frac{1}{2} - \Delta\right)\right).
\end{aligned}$$

The expectation in the middle line is over $(A, \mathcal{K})$ under $I_1$. Since these variables have the same law under both instances, $\mathbf{E}[J_k(X)]$ is the same under $I_1$ and $I_2$. This proves $(c)$. $(d)$ follows from Chernoff's inequality:

$$\mathbf{E}[J_k(X)] = \frac{N}{C^*} - k\mathbf{Pr}[X \geq k] \leq \frac{N}{C^*} - k + k\mathbf{Pr}\left[X < \frac{N}{2C^*}\right] \leq \frac{N}{C^*} - k + \frac{k}{e} \leq \frac{N-k}{C^*},$$

and $\mathtt{KL}(\frac{1}{2}+\Delta\|\frac{1}{2}-\Delta) = 2\Delta \ln \frac{1+2\Delta}{1-2\Delta} \leq 10\Delta^2$ for $0 \leq \Delta \leq \frac{1}{5}$. Choose $\Delta = \min\left\{\frac{1}{5}, \frac{1}{2}\sqrt{\frac{C^*}{5(N-k)}}\right\}$. Since $\delta \leq \frac{1}{8\sqrt{e}}$, it follows that

$$\mathtt{SubOpt}(N, C^*, k, \varepsilon, \delta) \geq \frac{e^{-\varepsilon}}{16}\sqrt{\frac{C^*}{5e(N-k)}} = \Omega\left(e^{-\varepsilon}\sqrt{\frac{C^*}{N-k}}\right).$$

When $C^* \in (1, 2)$, we construct new instances that are different in both the reward distributions as well as the behavior policy. More specifically,

$$\boldsymbol{\mu}_{I_1} = (\frac{1}{2}+\Delta, \frac{1}{2}, 0), \quad \boldsymbol{\mu}_{I_2} = (\frac{1}{2}, \frac{1}{2}+\Delta, 0).$$

Here we set $\Delta = \frac{2-C^*}{8}$. For the behavior policy $d$ on $\mathcal{A}$, $d_{I_1} = \left(\frac{1}{C^*}, \frac{C^*-1}{C^*} - \frac{k}{N}, \frac{k}{N}\right)$, $d_{I_2} = \left(\frac{C^*-1}{C^*} - \frac{k}{N}, \frac{1}{C^*}, \frac{k}{N}\right)$. Therefore in both instances, $a^*$ satisfies $d(a^*) \geq \frac{1}{C^*}$. The notations $D_1(N, C^*), D_2(N, C^*), U_1, U_2$ are defined analogously. Denote the event $V$ as $\{N(a_3) \geq k\}$. Since the mean of this binomial random variable is the integer $k$, the binomial median property gives $\mathbf{Pr}[V] \geq \frac{1}{2}$ for both instances. On $V$, choose $k$ observations of arm 3 uniformly at random and delete them. On $V^c$, choose an arm with the largest sample count among arms $1, 2, 3$, using fixed tie-breaking, and delete $k$ observations uniformly from that arm. This fallback is always feasible under the assumed condition $k \leq \frac{N}{2}\left(1 - \frac{1}{C^*}\right)$. Indeed, since $1 < C^* < 2$,

$$k < \frac{N}{4}, \qquad \max_{a\in\{1,2,3\}} N_{D_i}(a) \geq \frac{N}{3} > k.$$

Thus every realization produces a valid exact-$k$ single-source request $U_i$. Conditional on the action labels, the deletion locations are selected independently of all rewards. We use only the contribution of $V$ to the lower bound; the nonnegative contribution of $V^c$ can be discarded.

On $V$, $\rho(U_1) = \rho(U_2) = \{(3, k)\}$. For $i \in \{1, 2\}$, let $Q_i$ denote the conditional output law of the reference execution:

$$Q_i(B) := \mathbf{Pr}\left[\pi'([\emptyset, \rho(U_i)], T(D_i^-)) \in B \,\middle|\, V\right], \qquad B \subseteq \mathcal{A},$$

and let $\mathcal{L}_i(D_i^-|V)$ denote the conditional distribution of the retained dataset. We then have

$$\begin{aligned}
&\max_{i\in\{1,2\}} \mathbf{E}_{D_i(N,C^*),U_i}[\mu(a^*) - \mu(\hat{a}')] \\
&\geq \frac{\mathbf{Pr}[V]}{2}\left(\mathbf{E}_{D_1(N,C^*),U_1}[\mu(a^*) - \mu(\hat{a}') \mid V] + \mathbf{E}_{D_2(N,C^*),U_2}[\mu(a^*) - \mu(\hat{a}') \mid V]\right) \\
&\overset{(a)}{\geq} \frac{\mathbf{Pr}[V]\, e^{-\varepsilon}\Delta}{2}\left(Q_1(\mathcal{A}\setminus\{1\}) + Q_2(\{1\}) - 2\delta\right) \\
&\geq \frac{\mathbf{Pr}[V]\, e^{-\varepsilon}\Delta}{2}\left(1 - \mathtt{TV}(Q_1, Q_2) - 2\delta\right) \\
&\overset{(b)}{\geq} \frac{e^{-\varepsilon}\Delta}{8}\exp\{-\mathtt{KL}(Q_1\|Q_2)\} - e^{-\varepsilon}\delta\Delta \\
&\overset{(c)}{\geq} \frac{e^{-\varepsilon}\Delta}{8}\exp\left\{-\mathtt{KL}(\mathcal{L}_1(D_1^-|V)\|\mathcal{L}_2(D_2^-|V))\right\} - e^{-\varepsilon}\delta\Delta.
\end{aligned}$$

Here, $(a)$ follows from the reverse $(\varepsilon, \delta)$-BUL inequality, applied pointwise and then averaged conditional on $V$. Step $(b)$ follows from the Bretagnolle–Huber inequality and $\mathbf{Pr}[V] \geq \frac{1}{2}$. For $(c)$, conditional on $V$, the descriptor is the same deterministic value $(3, k)$ under both instances. The reference execution therefore applies the same randomized mapping

$$x \longmapsto \pi'([\emptyset, (3, k)], T(x)).$$

The mechanism may depend on $(3, k)$; however, this common deterministic input does not provide additional information for distinguishing the two instances conditional on $V$. Since $T(D_i^-)$ is a deterministic function of $D_i^-$, the data processing inequality gives

$$\mathtt{KL}(Q_1 \| Q_2) \le \mathtt{KL}\left(\mathcal{L}_1(T(D_1^-) \mid V) \,\middle\|\, \mathcal{L}_2(T(D_2^-) \mid V)\right) \le \mathtt{KL}\left(\mathcal{L}_1(D_1^- \mid V) \,\middle\|\, \mathcal{L}_2(D_2^- \mid V)\right).$$

Let $\mathcal{S}_3 := \{i \in [N] : a_i = 3\}$, $M := |\mathcal{S}_3|$, and, on the event $V = \{M \ge k\}$, let $\mathcal{K} \subseteq \mathcal{S}_3$ denote the uniformly selected set of $k$ deleted indices. Arm 3 has the same behavior probability $\frac{k}{N}$ under $I_1$ and $I_2$, and the same deletion rule is used under both instances.

Let $P_i^\circ := \mathcal{L}_i\big((a_j, r_j) | a_j \ne 3\big), i \in \{1, 2\}$ be the conditional distribution of one generic action–reward observation of $I_i$, given that its action is either arm 1 or arm 2. Let $P_3$ denote the distribution of an observation from arm 3, since they are the same under both $I_1$ and $I_2$. Conditional on $(\mathcal{S}_3, \mathcal{K})$, the retained dataset $\mathcal{L}_i(D_i^- | \mathcal{S}_3, \mathcal{K})$ has a distribution of $\bigotimes_{j \notin \mathcal{S}_3} P_i^\circ \otimes \bigotimes_{j \in \mathcal{S}_3 \setminus \mathcal{K}} P_3$ under instance $I_i$. Therefore,

$$\begin{aligned}
&\mathtt{KL}\left(\mathcal{L}_1\left(D_1^- \,\middle|\, \mathcal{S}_3, \mathcal{K}, V\right) \,\middle\|\, \mathcal{L}_2\left(D_2^- \,\middle|\, \mathcal{S}_3, \mathcal{K}, V\right)\right) \\
&= (N - M)\mathtt{KL}(P_1^\circ \| P_2^\circ) + (M - k)\mathtt{KL}(P_3 \| P_3) \\
&= (N - M)\mathtt{KL}(P_1^\circ \| P_2^\circ).
\end{aligned}$$

Thus, the retained observations from arm 3 contribute zero KL divergence because they have the same distribution under the two instances. For the KL calculation, retaining the original sample indices and the variables $(\mathcal{S}_3, \mathcal{K})$ can only provide additional information. Hence,

$$\begin{aligned}
&\mathtt{KL}(\mathcal{L}_1(D_1^- | V) \| \mathcal{L}_2(D_2^- | V)) \\
&\le \mathtt{KL}(\mathcal{L}_1(\mathcal{S}_3, \mathcal{K}, D_1^- | V) \| \mathcal{L}_2(\mathcal{S}_3, \mathcal{K}, D_2^- | V)) \\
&\overset{(a)}{=} \mathtt{KL}(\mathcal{L}_1(\mathcal{S}_3, \mathcal{K} | V) \| \mathcal{L}_2(\mathcal{S}_3, \mathcal{K} | V)) \\
&\quad + \mathbf{E}_{(\mathcal{S}_3, \mathcal{K}) \sim \mathcal{L}_1(\cdot | V)} \left[\mathtt{KL}\left(\mathcal{L}_1\left(D_1^- | \mathcal{S}_3, \mathcal{K}, V\right) \| \mathcal{L}_2\left(D_2^- | \mathcal{S}_3, \mathcal{K}, V\right)\right)\right] \\
&\overset{(b)}{=} \mathbf{E}_{(\mathcal{S}_3, \mathcal{K}) \sim \mathcal{L}_1(\cdot | V)} \left[(N - M)\mathtt{KL}(P_1^\circ \| P_2^\circ)\right] \\
&\le (N - k)\mathtt{KL}(P_1^\circ \| P_2^\circ),
\end{aligned}$$

where $(a)$ follows from the chain rule for KL divergence. $(b)$ follows the fact that $(\mathcal{S}_3, \mathcal{K})|V$ has the same distribution under the two instances, while the last inequality uses $M \ge k$ on $V$.

It remains to calculate the one-observation divergence $\mathtt{KL}(P_1^\circ \| P_2^\circ)$. Conditional on $A \ne a_3$, the action distributions under $I_1$ and $I_2$ are, respectively, $d'_{I_1} = \left(\frac{N}{C^*(N-k)}, \frac{C^*(N-k)-N}{C^*(N-k)}\right)$ and $d'_{I_2} = \left(\frac{C^*(N-k)-N}{C^*(N-k)}, \frac{N}{C^*(N-k)}\right)$. Therefore, by the chain rule for the joint action–reward distribution,

$$\begin{aligned}
\mathtt{KL}(P_1^\circ \| P_2^\circ) &= \mathtt{KL}(d'_{I_1} \| d'_{I_2}) + \sum_{a=1}^{2} d'_{I_1}(a)\mathtt{KL}\left(\mu_{I_1}(a) \,\|\, \mu_{I_2}(a)\right) \\
&= \frac{2N - C^*(N-k)}{C^*(N-k)} \ln\left(\frac{N}{C^*(N-k) - N}\right) + \frac{N}{C^*(N-k)}\mathtt{KL}\left(\frac{1}{2} + \Delta \,\middle\|\, \frac{1}{2}\right) \\
&\quad + \frac{C^*(N-k) - N}{C^*(N-k)}\mathtt{KL}\left(\frac{1}{2} \,\middle\|\, \frac{1}{2} + \Delta\right).
\end{aligned}$$

Since $\Delta = \frac{2 - C^*}{8}$, $\mathtt{KL}\left(\frac{1}{2} + \Delta \| \frac{1}{2}\right) \le 4\Delta^2$, and $\mathtt{KL}\left(\frac{1}{2} \| \frac{1}{2} + \Delta\right) \le \frac{64}{15}\Delta^2$. It follows that the sum of the last two terms is at most $\frac{(2-C^*)^2}{15}$. Moreover, $k \le \frac{N}{2}\left(1 - \frac{1}{C^*}\right)$. Therefore,

$$\mathtt{KL}(P_1^\circ \| P_2^\circ) \le \frac{3 - C^*}{C^* + 1} \ln\left(\frac{2}{C^* - 1}\right) + \frac{(2 - C^*)^2}{15} \le \ln\left(\frac{2}{C^* - 1}\right) + \frac{(2 - C^*)^2}{15}.$$

Since there are at most $N - k$ such observations on $V$, the chain rule for KL divergence gives

$$\begin{aligned}
\mathtt{SubOpt}(N, C^*, k, \varepsilon, 0) &\ge \frac{2 - C^*}{64} e^{-\varepsilon - (N-k)\left(\ln\left(\frac{2}{C^*-1}\right) + \frac{(2-C^*)^2}{15}\right)} \\
&= \Omega\left((2 - C^*)\, e^{-\varepsilon - (N-k)\left(\ln\left(\frac{2}{C^*-1}\right) + \frac{(2-C^*)^2}{15}\right)}\right).
\end{aligned}$$

□

### A.3 THE MIXING ALGORITHM

In this section, we study the mixing algorithm and investigate whether it can outperform the two base algorithms: Gaussian mechanism or rollback. For clarity of exposition, we present the algorithm under $\mathcal{M}_f$ (we can obtain similar conclusion under $\mathcal{M}_d$). When $a_0 \neq a^*$, rollback is always preferable, since it does not degrade the performance of the underlying learning algorithm. Accordingly, Algorithm 4 retains the initial check $\max_{a\neq a_0}\{\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}}\} \geq \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$.

Let $U' \subseteq U$ denote the subset of points that are first handled via rollback, and write $|U'| = k'$. After rolling back $U'$, the algorithm adds Gaussian noise to the LCB vector with $\sigma = \frac{3(k-k')\gamma}{2(N(a_0)-k')}$. The parameter $k'$ interpolates between the two extremes: $k' = k$ recovers pure rollback, whereas $k' = 0$ recovers the pure Gaussian mechanism.

**Algorithm 4** Single-source unlearning mixing algorithm under $\mathcal{M}_f$

**Input:** Unlearning request: $U$ from $a_0$, $|U| = k$, additional statistics $T(D)$: $\widehat{\boldsymbol{f}}(D)$, $\widehat{\boldsymbol{\mu}}$, $\widehat{\boldsymbol{N}}$, a confidence level: $\tau \in (0,1)$, $\varepsilon \in (0,\infty)$, $\delta \in (0,1)$, mixing parameter: $k' \in [0,k]$

1: Set $\widehat{\boldsymbol{f}}(D'') \leftarrow \widehat{\boldsymbol{f}}(D), \widehat{\boldsymbol{N}}' \leftarrow \widehat{\boldsymbol{N}} - k\boldsymbol{I}\{a = a_0\}, \widehat{\boldsymbol{\mu}}' \leftarrow \widehat{\boldsymbol{\mu}}$
2: Compute the new empirical mean reward $\hat{\mu}'(a_0) \leftarrow \frac{\hat{\mu}(a_0)\cdot N(a_0) - \sum_{(a_0,r_i)\in U} r_i}{N'(a_0)}$
3: **if** $\max_{a\neq a_0}\{\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}}\} \geq \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$ **then** $k' \leftarrow k$
4: Compute $\widehat{\boldsymbol{N}}'' \leftarrow \widehat{\boldsymbol{N}} - k'\boldsymbol{I}\{a = a_0\}, \widehat{\boldsymbol{\mu}}'' \leftarrow \widehat{\boldsymbol{\mu}}$ and choose $k'$ data points in $U$ uniformly and randomly denote them as $U'$
5: Compute the new empirical mean reward $\hat{\mu}''(a_0) \leftarrow \frac{\hat{\mu}(a_0)\cdot N(a_0) - \sum_{(a_0,r_i)\in U'} r_i}{N''(a_0)}$
6: Compute $\sigma \leftarrow \frac{3(k-k')\gamma}{2(N(a_0)-k')}$, $\beta \leftarrow \sqrt{2\ln(1+N\sigma)}$, $b''(a_0) \leftarrow \sqrt{\frac{\ln\frac{m}{\tau}}{2N''(a_0)}}$
7: Set $\widehat{\boldsymbol{f}}_{(a_0)}(D'') \leftarrow \hat{\mu}''(a_0) - b''(a_0)$
8: Sample $\nu \in \mathbb{R}$ from $\mathcal{N}(0,\sigma^2)$
9: Set $\widehat{\boldsymbol{f}}_{(a_0)}(D'') \leftarrow \widehat{\boldsymbol{f}}_{(a_0)}(D'') + \nu - \beta\sigma$

**Output:** $\hat{a}' = \arg\max_a \widehat{\boldsymbol{f}}_{(a)}(D'')$

**Algorithm 5** Single-source unlearning mixing algorithm under $\mathcal{M}_f$ with input $\emptyset$

**Input:** Request descriptor $\rho(U) := (a_0, k)$, additional statistics $T(D \setminus U)$: $\widehat{\boldsymbol{f}}(D \setminus U)$, $\widehat{\boldsymbol{\mu}}'$, $\widehat{\boldsymbol{N}}'$, a confidence level: $\tau \in (0,1)$, $\varepsilon \in (0,\infty)$, $\delta \in (0,1)$, mixing parameter: $k' \in [0,k]$

1: **if** $\max_{a\neq a_0}\{\hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}}\} \geq \hat{\mu}'(a_0) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$ **then** $k' \leftarrow k$
2: Compute $\widehat{\boldsymbol{f}}(D') \leftarrow \widehat{\boldsymbol{f}}(D \setminus U), \widehat{\boldsymbol{N}} \leftarrow \widehat{\boldsymbol{N}}' + k\boldsymbol{I}\{a = a_0\}$
3: Compute $\sigma \leftarrow \frac{3(k-k')\gamma}{2(N(a_0)-k')}$, $\beta \leftarrow \sqrt{2\ln(1+N\sigma)}$, $b'(a_0) \leftarrow \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a_0)}}$
4: Set $\widehat{\boldsymbol{f}}_{(a_0)}(D') \leftarrow \hat{\mu}'(a_0) - b'(a_0)$
5: Sample $\nu \in \mathbb{R}$ from $\mathcal{N}(0,\sigma^2)$
6: Set $\widehat{\boldsymbol{f}}_{(a_0)}(D') \leftarrow \widehat{\boldsymbol{f}}_{(a_0)}(D') + \nu - \beta\sigma$

**Output:** $\hat{a}' = \arg\max_a \widehat{\boldsymbol{f}}_{(a)}(D')$

**Theorem 17.** *Algorithm 4 and Algorithm 1 are $(\varepsilon,\delta)$-unlearning when $k \leq N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$.*

*Proof.* Let $D' = D \setminus U$ and $D'' = D \setminus U'$, where $U' \subseteq U$ and $|U'| = k'$. After rolling back $U'$, the sample count of $a_0$ is $N''(a_0) = N(a_0) - k'$, and the remaining request $U \setminus U'$ contains $k - k'$

points from $a_0$. From $k \le N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$, we have

$$k - k' \le N(a_0) - k' - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}} \le N(a_0) - k' - \sqrt{\frac{(N(a_0)-k')\ln\frac{m}{\tau}}{2}}.$$

Define $\overline{\Delta}_{\boldsymbol{f},k'} := \frac{3(k-k')}{2(N(a_0)-k')}$, which is the quantity denoted by $\Delta_f$ in the two mixing algorithms. Applying Lemma 10 to $D''$ and the remaining request $U \setminus U'$ gives $\left|\boldsymbol{f}_{(a_0)}(D'') - \boldsymbol{f}_{(a_0)}(D')\right| \le \overline{\Delta}_{\boldsymbol{f},k'}$. For every $a \ne a_0$, $\boldsymbol{f}_{(a)}(D'') = \boldsymbol{f}_{(a)}(D')$, since all points in $U$ are from $a_0$.

We next compare the actual and reference executions. The initial check depends only on the retained-data statistics and the request descriptor $\rho(U) = (a_0, k)$. Hence, the two executions make the same decision on whether to set $k' = k$ and consequently use the same sensitivity bound $\overline{\Delta}_{\boldsymbol{f},k'}$, noise scale $\sigma$, and shift parameter $\beta$.

Suppose first that $k' < k$. For the analysis, we may augment the reference execution with an independent dummy draw of $U'$ from the same uniform distribution, since its output does not depend on this additional randomness. Conditional on any realization of $U'$, the $a_0$ coordinate in the actual execution is $\widehat{\boldsymbol{f}}_{U,(a_0)} = \boldsymbol{f}_{(a_0)}(D'') + \nu_U - \beta\sigma$, whereas the corresponding coordinate in the reference execution is $\widehat{\boldsymbol{f}}_{\emptyset,(a_0)} = \boldsymbol{f}_{(a_0)}(D') + \nu_\emptyset - \beta\sigma$, where $\nu_U, \nu_\emptyset \sim \mathcal{N}(0,\sigma^2)$.

Since the common downward shift cancels, the distance between the two Gaussian centers satisfies

$$\left|\left(\boldsymbol{f}_{(a_0)}(D'') - \beta\sigma\right) - \left(\boldsymbol{f}_{(a_0)}(D') - \beta\sigma\right)\right| = \left|\boldsymbol{f}_{(a_0)}(D'') - \boldsymbol{f}_{(a_0)}(D')\right| \le \overline{\Delta}_{\boldsymbol{f},k'}.$$

Moreover, the algorithms set $\sigma = \gamma\Delta_f = \gamma\overline{\Delta}_{\boldsymbol{f},k'}$. Therefore,

$$\Phi\left(\frac{\overline{\Delta}_{\boldsymbol{f},k'}}{2\sigma} - \frac{\varepsilon\sigma}{\overline{\Delta}_{\boldsymbol{f},k'}}\right) - e^{\varepsilon}\Phi\left(-\frac{\overline{\Delta}_{\boldsymbol{f},k'}}{2\sigma} - \frac{\varepsilon\sigma}{\overline{\Delta}_{\boldsymbol{f},k'}}\right) = \Phi\left(\frac{1}{2\gamma} - \varepsilon\gamma\right) - e^{\varepsilon}\Phi\left(-\frac{1}{2\gamma} - \varepsilon\gamma\right) \le \delta,$$

where the last inequality follows from (1). By Theorem 8 and Lemma 7 of Balle & Wang (2018), the two randomized coordinates satisfy the two-sided $(\varepsilon,\delta)$ inequalities.

The preceding inequalities hold conditional on every realization of $U'$. Averaging over the uniform random choice of $U'$ therefore preserves the same $(\varepsilon,\delta)$ inequalities. Since all coordinates other than $a_0$ are identical in the two executions, the entire LCB vectors satisfy $(\varepsilon,\delta)$ indistinguishability.

If $k' = k$, then $D'' = D'$, $\overline{\Delta}_{\boldsymbol{f},k'} = \Delta_f = \sigma = \beta = 0$, and both executions reduce exactly to rollback. Hence, their output distributions are identical.

Finally, applying $\arg\max$ to the LCB vector is deterministic post-processing. Therefore, Algorithm 4 and Algorithm 1 satisfy $(\varepsilon,\delta)$-unlearning. □

**Theorem 18** ($\mathcal{M}_f$). *Consider any offline dataset $D(\boldsymbol{N})$ satisfying $N(a^*) \ge N_*$, and any reward-independent unlearning request $U$ of size $k$ from $a_0$. When $k \le N(a_0) - \sqrt{\frac{N(a_0)\ln\frac{m}{\tau}}{2}}$, by setting $\tau = \frac{1}{N}$, the arm $\hat{a}'$ returned by Algorithm 4 satisfies*

$$\begin{aligned}&\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta)\\ &= O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)-k'}} + \frac{(k-k')\gamma(1+\sqrt{\ln(1+N\gamma)})}{N(a_0)-k'}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right).\end{aligned}$$

*Proof.* Let $D' = D \setminus U, D'' = D \setminus U'$. Denote $\mathcal{E}_1$ as the simultaneous confidence event on $D'$ and $\mathcal{E}_2$ as the simultaneous confidence event on $D''$:

$$\mathcal{E}_1 = \left\{|\hat{\mu}'(a) - \mu(a)| \le \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}},\ \forall a \in \mathcal{A}\right\}, \qquad \mathcal{E}_2 = \left\{|\hat{\mu}''(a) - \mu(a)| \le \sqrt{\frac{\ln\frac{m}{\tau}}{2N''(a)}},\ \forall a \in \mathcal{A}\right\}.$$

Since $U$ is independent of the rewards and $U'$ is selected uniformly and independently of the rewards, Hoeffding's inequality and a union bound give $\mathbf{Pr}[\mathcal{E}_1^c] \le 2\tau$, $\mathbf{Pr}[\mathcal{E}_2^c] \le 2\tau$.

We first consider the case where $a_0 = a^*$. Suppose the initial check holds. Then there exists some $a \neq a^*$ such that

$$\hat{\mu}'(a) - \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a)}} \geq \hat{\mu}'(a^*) + \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a^*)}}.$$

On $\mathcal{E}_1$,

$$\hat{\mu}'(a) - \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a)}} \leq \mu(a) \leq \mu(a^*) \leq \hat{\mu}'(a^*) + \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a^*)}}.$$

Therefore all the inequalities above hold with equality. In particular, $\mu(a) = \mu(a^*)$. Moreover, if $\tilde{a}$ is any maximizer of the retained-data LCB, then $\widehat{\boldsymbol{f}}_{(\tilde{a})}(D') \geq \widehat{\boldsymbol{f}}_{(a)}(D') \geq \mu(a^*)$, while on $\mathcal{E}_1$, $\widehat{\boldsymbol{f}}_{(\tilde{a})}(D') \leq \mu(\tilde{a}) \leq \mu(a^*)$. Hence $\mu(\tilde{a}) = \mu(a^*)$. Since the algorithm sets $k' = k$ in this case and performs rollback, its sub-optimality is zero on $\mathcal{E}_1$.

We next consider the case where the initial check does not hold. Then the algorithm first rolls back $k'$ points and applies the downward-shifted Gaussian mechanism to the remaining $k - k'$ deletions. Recall that $N''(a_0) = N(a_0) - k'$, $\Delta_{\boldsymbol{f}} = \frac{3(k-k')}{2(N(a_0)-k')}$, $\sigma = \frac{3(k-k')\gamma}{2(N(a_0)-k')}$, and $\beta = \sqrt{2\ln(1+N\sigma)}$.

On $\mathcal{E}_2$, $\widehat{\boldsymbol{f}}_{(a^*)}(D'') \geq \mu(a^*) - \sqrt{\frac{2\ln \frac{m}{\tau}}{N(a_0)-k'}}$, while $\widehat{\boldsymbol{f}}_{(a)}(D'') \leq \mu(a), \forall a \in \mathcal{A}$.

If $\hat{a}' = a^*$, the sub-optimality is zero. Otherwise, by the definition of $\hat{a}'$, $\widehat{\boldsymbol{f}}_{(\hat{a}')}(D'') \geq \widehat{\boldsymbol{f}}_{(a^*)}(D'') + \nu - \beta\sigma$. Hence

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln \frac{m}{\tau}}{N(a_0)-k'}} + \beta\sigma - \nu.$$

Combining the two output cases gives $\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln \frac{m}{\tau}}{N(a_0)-k'}} + (\beta\sigma - \nu)_+$. Let $Z \sim \mathcal{N}(0,1)$. Since $\nu = \sigma Z$, $\mathbf{E}_\nu\left[(\beta\sigma - \nu)_+\right] = \sigma\mathbf{E}\left[(\beta - Z)_+\right] \leq \sigma\left(\beta + \frac{1}{\sqrt{2\pi}}\right)$. Therefore, accounting for the confidence-failure events,

$$\begin{aligned}
\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right] &\leq \sqrt{\frac{2\ln \frac{m}{\tau}}{N(a_0)-k'}} + \sigma\left(\beta + \frac{1}{\sqrt{2\pi}}\right) + 4\tau \\
&= O\left(\sqrt{\frac{\ln \frac{m}{\tau}}{N(a_0)-k'}} + \frac{(k-k')\gamma}{N(a_0)-k'}\left[1 + \sqrt{\ln\left(1 + \frac{N(k-k')\gamma}{N(a_0)-k'}\right)}\right] + \tau\right) \\
&= O\left(\sqrt{\frac{\ln(Nm)}{N(a_0)-k'}} + \frac{(k-k')\gamma(1+\sqrt{\ln(1+N\gamma)})}{N(a_0)-k'}\right).
\end{aligned}$$

The second case is $a_0 \neq a^*$. If the initial check holds, Algorithm 4 sets $k' = k$ and performs rollback. Since the deletion does not change the samples of the optimal arm, $N'(a^*) = N(a^*) \geq N_*$. On $\mathcal{E}_1$, the standard retained-data LCB guarantee gives $\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln \frac{m}{\tau}}{N_*}}$. If instead the initial check does not hold, the algorithm first rolls back $k'$ points from $a_0$ and subsequently applies the downward-shifted Gaussian mechanism. Since $a_0 \neq a^*$, $N''(a^*) = N(a^*) \geq N_*$.

If $\hat{a}' \neq a_0$, both $\hat{a}'$ and $a^*$ use their unperturbed LCBs on $D''$. By maximality, $\widehat{\boldsymbol{f}}_{(\hat{a}')}(D'') \geq \widehat{\boldsymbol{f}}_{(a^*)}(D'')$, and hence $\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln \frac{m}{\tau}}{N_*}}$. If $\hat{a}' = a_0$, then $\widehat{\boldsymbol{f}}_{(a_0)}(D'') + \nu - \beta\sigma \geq \widehat{\boldsymbol{f}}_{(a^*)}(D'')$.

Using $\widehat{\boldsymbol{f}}_{(a_0)}(D'') \leq \mu(a_0)$ and $\widehat{\boldsymbol{f}}_{(a^*)}(D'') \geq \mu(a^*) - \sqrt{\frac{2\ln \frac{m}{\tau}}{N_*}}$, we obtain

$$\mu(a^*) - \mu(a_0) \leq \sqrt{\frac{2\ln \frac{m}{\tau}}{N_*}} + \nu - \beta\sigma.$$

Combining the two output cases gives

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln \frac{m}{\tau}}{N_*}} + (\nu - \beta\sigma)_+.$$

As in the single-source analysis,

$$\mathbf{E}_\nu\left[(\nu-\beta\sigma)_+\right] \le \frac{\sigma}{\sqrt{2\pi}(1+N\sigma)} \le \frac{1}{\sqrt{2\pi}N}.$$

Therefore,

$$\mathbf{E}\left[\mu(a^*)-\mu(\hat{a}')\right] = O\left(\sqrt{\frac{\ln\frac{m}{\tau}}{N_*}}+\frac{1}{N}+\tau\right).$$

Taking the worse of the cases $a_0 = a^*$ and $a_0 \neq a^*$ and setting $\tau = \frac{1}{N}$ gives

$$\begin{aligned}&\texttt{SubOpt}(\boldsymbol{N}, N_*, k, \varepsilon, \delta)\\ &= O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N(a_0)-k'}}+\frac{(k-k')\gamma(1+\sqrt{\ln(1+N\gamma)})}{N(a_0)-k'}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right)\end{aligned}$$

which completes the proof. □

**Comparison.** We further investigate whether an intermediate mixing parameter can improve over the two base mechanisms. Define

$$f(x) := \sqrt{\frac{\ln(Nm)}{N(a_0)-x}}+\frac{(k-x)\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N(a_0)-x}, \qquad 0\le x\le k.$$

Let $C := \gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)$. Then

$$f'(x) = \frac{\frac{1}{2}\sqrt{\ln(Nm)\,(N(a_0)-x)}-C\,(N(a_0)-k)}{(N(a_0)-x)^2}.$$

The denominator is positive, while the numerator is strictly decreasing in $x$. Hence $f'(x)$ either keeps the same sign or changes sign only from positive to negative. Therefore, any stationary point in $(0,k)$ is a maximum, and the minimum is attained at an endpoint $\min_{0\le x\le k} f(x) = \min\{f(0), f(k)\}$. Here, $f(0) = \sqrt{\frac{\ln(Nm)}{N(a_0)}}+\frac{k\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N(a_0)}$ corresponds to the Gaussian mechanism under the same relaxed upper bound, whereas $f(k) = \sqrt{\frac{\ln(Nm)}{N(a_0)-k}}$ corresponds to pure rollback. Thus, under this upper-bound comparison, an intermediate mixing parameter cannot improve upon the better of the two base mechanisms.

### A.4 Improved Results for $C^* \in (1,2)$

**Algorithm 6** Single-source unlearning algorithm under $\mathcal{M}_d$ ($C^* \in (1,2)$)

**Input:** Unlearning request: $U$ from $a_0$, $|U| = k$, additional statistics $T(D)$: $\widehat{\boldsymbol{f}}(D)$, $\widehat{\boldsymbol{\mu}}$, $\widehat{\boldsymbol{N}}$

1: Set $\widehat{\boldsymbol{N}}' \leftarrow \widehat{\boldsymbol{N}}$, $\widehat{\boldsymbol{N}}'_{(a_0)} \leftarrow \widehat{\boldsymbol{N}}_{(a_0)} - k$

**Output:** $\hat{a}' = \arg\max_a \widehat{\boldsymbol{N}}'_{(a)}$

When the unlearned request is $\emptyset$, Algorithm 6 simply outputs the learned arm $\hat{a}' = \arg\max_a \widehat{\boldsymbol{N}}'_{(a)}$, which is the arm with the biggest sample count computed from $D \setminus U$. Then we have

**Theorem 19.** *Algorithm 6 and imitation learning are $(0,0)$-unlearning.*

*Proof.* When the inputs are $[\emptyset, \rho(U)], T(D\setminus U)$, denote $\widehat{\boldsymbol{N}}'_1$ as the sample-count vector at the output step. Similarly, let $\widehat{\boldsymbol{N}}'_2$ denote the corresponding vector when the inputs are $[U, \emptyset], T(D)$. In both cases, the algorithm outputs the same vector, i.e. $\widehat{\boldsymbol{N}}'_1 = \widehat{\boldsymbol{N}}'_2 = \widehat{\boldsymbol{N}}'$, where $\widehat{\boldsymbol{N}}'$ is the sample-count vector of $D \setminus U$. Viewing $g(\widehat{\boldsymbol{N}}) = \arg\max_a \widehat{\boldsymbol{N}}_{(a)}$ as a post-processing map, we obtain that Algorithm 6 and the imitation learning rule are $(0,0)$-unlearning. □

**Theorem 20** (Theorem 8 restated)**.** *Consider any offline dataset $D \sim \mathcal{D}^N$, where $d(a^*) \geq \frac{1}{C^*}$ for some $C^* \in (1,2)$, and any reward-independent unlearning request $U$ of size $k$ from $a_0$. When $k \leq \min\left\{\frac{3N}{4}, \frac{(2-C^*)N}{C^*}\right\}$, the arm $\hat{a}'$ returned by Algorithm 6 satisfies*

$$\texttt{SubOpt}(N, C^*, k, \varepsilon, 0) = O\left(e^{-\frac{(N-k)}{2}\ln\frac{C^*}{8(C^*-1)}+\frac{(N+k)}{2}\ln\frac{2}{C^*}}\right).$$

*Proof.* When $a_0 = a^*$, we bound $\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right]$ as follows,

$$\begin{aligned}\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right] &\leq \mathbf{Pr}\left[\hat{a}' \neq a^*\right] \leq \mathbf{Pr}\left[\exists a \neq a^*, N(a) \geq N(a^*) - k\right] \\ &\leq \mathbf{Pr}\left[N - N(a^*) \geq N(a^*) - k\right] = \mathbf{Pr}\left[N(a^*) \leq \frac{N+k}{2}\right].\end{aligned}$$

Since $k \leq \frac{(2-C^*)N}{C^*}$, we have that $\frac{N+k}{2} \leq \frac{N}{C^*}$. Then applying Chernoff's bound for binomial random variables, we have

$$\begin{aligned}\mathbf{Pr}\left[N(a^*) \leq \frac{N+k}{2}\right] &\leq \exp\left(-N \cdot \texttt{KL}\left(\texttt{Ber}\left(\frac{N+k}{2N}\right) \middle\| \texttt{Ber}\left(\frac{1}{C^*}\right)\right)\right) \\ &= \exp\left(-N\left(\frac{N+k}{2N}\ln\frac{(N+k)C^*}{2N} + \frac{N-k}{2N}\ln\frac{(N-k)C^*}{2N(C^*-1)}\right)\right) \\ &\overset{(a)}{\leq} \exp\left(\frac{N+k}{2}\ln\frac{2}{C^*} - \frac{N-k}{2}\ln\frac{C^*}{8(C^*-1)}\right),\end{aligned}$$

where $(a)$ is from $k \leq \frac{3N}{4}$. When $C^* \to 1$, the second term $\frac{N-k}{2}\ln\left(\frac{C^*}{8(C^*-1)}\right)$ dominates, and the resulting probability is on the order of $(C^*-1)^{\frac{N-k}{2}}$, which matches the lower bound in Theorem 7 in its dependence on $N, C^*, k$.

When $a_0 \neq a^*$, we know that

$$\begin{aligned}\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right] &\leq \mathbf{Pr}\left[N - N(a^*) - k \geq N(a^*)\right] \\ &\leq \mathbf{Pr}\left[N(a^*) \leq \frac{N-k}{2}\right] \leq \mathbf{Pr}\left[N(a^*) \leq \frac{N+k}{2}\right].\end{aligned}$$

Therefore this bound is smaller than the corresponding bound in the case $a_0 = a^*$. In conclusion,

$$\texttt{SubOpt}(N, C^*, k, \varepsilon, 0) = O\left(e^{-\frac{(N-k)}{2}\ln\frac{C^*}{8(C^*-1)}+\frac{(N+k)}{2}\ln\frac{2}{C^*}}\right).$$

□

### A.5 Multi-Source Unlearning under $\mathcal{M}_f$

Our multi-source construction preserves the same two-branch structure as in the single-source setting: the algorithm either applies a downward-shifted Gaussian mechanism to one requested source or performs exact rollback. The only additional difficulty is that the source to be perturbed must itself be selected without invalidating the unlearning guarantee. We address this by combining a common retained-data candidate set with a randomized source selection inside the Gaussian branch. In particular, even in the multi-source setting, at most one coordinate is perturbed by Gaussian noise. When there is only one candidate source, the construction reduces to the usual analytic Gaussian mechanism with noise multiplier $\gamma$.

The resulting rate is consistent with the single-source guarantee under $\mathcal{M}_f$: when $\ell = 1$, we have $N_{\min} = N(a_{u_1}) = N(a_0)$ and $k_{\max} = k_1 = k$, and the multi-source upper bound reduces to the corresponding single-source rate up to universal constants. Following the batching idea of Guo et al. (2020), requests that need not be processed immediately may be accumulated and handled as a single multi-source batch.

Let $U = \bigcup_{i=1}^{\ell} U_i$, $D' = D \setminus U$, and $|U_i| = k_i$, where $U_i$ consists of samples from arm $a_{u_i}$. Recall that $N_{\min} := \min_{i\in[\ell]} N(a_{u_i})$ and $k_{\max} := \max_{i\in[\ell]} k_i$. For every requested source, define

$\Delta_i := \frac{3k_i}{2N(a_{u_i})}$ and $\overline{\Delta} := \max_{i\in[\ell]} \Delta_i$. Under the sensitivity condition in Lemma 10,

$$\left|\widehat{\boldsymbol{f}}_{(a_{u_i})}(D) - \widehat{\boldsymbol{f}}_{(a_{u_i})}(D')\right| \le \Delta_i \le \overline{\Delta} \le \frac{3k_{\max}}{2N_{\min}}, \qquad \forall i \in [\ell].$$

We first screen the requested sources using only the retained data. Define

$$\mathcal{J} := \left\{ i \in [\ell] : \max_{a \neq a_{u_i}} \left\{ \hat{\mu}'(a) - \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a)}} \right\} \le \hat{\mu}'(a_{u_i}) + \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a_{u_i})}} \right\}. \tag{6}$$

Thus, $\mathcal{J}$ contains the requested sources that remain statistically plausible as an optimal arm after removing $U$. For $i \in \mathcal{J}$, write $f_i := \widehat{\boldsymbol{f}}_{(a_{u_i})}(D)$, $h_i := \widehat{\boldsymbol{f}}_{(a_{u_i})}(D')$. The actual execution can compute both $f_i$ and $h_i$, whereas the empty-request execution only needs $h_i$. Both executions construct the same refined candidate set

$$\mathcal{H} := \left\{ i \in \mathcal{J} : h_i \ge \max_{s\in\mathcal{J}} h_s - 2\overline{\Delta} \right\}. \tag{7}$$

Whenever $\mathcal{J} \neq \emptyset$, the set $\mathcal{H}$ is nonempty. For $j \in \mathcal{H}$, define the actual- and empty-request selection probabilities

$$p_j := \frac{\exp\left(\frac{\varepsilon f_j}{4\overline{\Delta}}\right)}{\sum_{i\in\mathcal{H}} \exp\left(\frac{\varepsilon f_i}{4\overline{\Delta}}\right)}, \qquad q_j := \frac{\exp\left(\frac{\varepsilon h_j}{4\overline{\Delta}}\right)}{\sum_{i\in\mathcal{H}} \exp\left(\frac{\varepsilon h_i}{4\overline{\Delta}}\right)}. \tag{8}$$

To specify the randomized source selection, define the analytic Gaussian privacy profile

$$\Psi_\varepsilon(x) := \Phi\left(\frac{1}{2x} - \varepsilon x\right) - e^{\varepsilon}\Phi\left(-\frac{1}{2x} - \varepsilon x\right).$$

By (1) and the continuity of the Gaussian privacy profile, $\Psi_\varepsilon(\gamma) = \delta$. We further define

$$\delta_{\mathrm{G}} := \Psi_{\varepsilon/2}(2\gamma) = \Phi\left(\frac{1}{4\gamma} - \varepsilon\gamma\right) - e^{\varepsilon/2}\Phi\left(-\frac{1}{4\gamma} - \varepsilon\gamma\right), \qquad \omega := \frac{1-\delta}{1-\delta_{\mathrm{G}}}. \tag{9}$$

The following lemma guarantees that $\omega$ is a valid probability.

**Lemma 21** (Half-budget analytic Gaussian calibration)**.** *For every $\varepsilon > 0$ and $x > 0$, $\Psi_{\varepsilon/2}(2x) \le \Psi_\varepsilon(x)$. Consequently, $0 \le \delta_{\mathrm{G}} \le \delta < 1$ and $\omega \in (0, 1]$.*

*Proof.* Let $\phi$ denote the standard Gaussian density, and set $\mu := \frac{1}{x}$, $c := \frac{\varepsilon}{\mu} = \varepsilon x$. For $t \ge 0$, define $F_c(t) := \Phi(t-c) - e^{2ct}\Phi(-t-c)$. Since $e^{2ct}\phi(t+c) = \phi(t-c)$, $F_c'(t) = 2e^{2ct}\{\phi(t+c) - c\Phi(-t-c)\}$. By the Gaussian Mills inequality, $\Phi(-t-c) \le \frac{\phi(t+c)}{t+c}$, and hence $c\Phi(-t-c) \le \frac{c}{t+c}\phi(t+c) \le \phi(t+c)$. Therefore $F_c'(t) \ge 0$. Finally, $\Psi_\varepsilon(x) = F_c\left(\frac{\mu}{2}\right)$, and $\Psi_{\varepsilon/2}(2x) = F_c\left(\frac{\mu}{4}\right)$, which proves the claimed inequality. Taking $x = \gamma$ gives $\delta_{\mathrm{G}} \le \delta$ and therefore $0 < \omega \le 1$. □

Let

$$B_G'(\gamma) := 28\overline{\Delta}\gamma\left(1 + \sqrt{\ln(1+N\gamma)}\right), \qquad B_R' := \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min} - k_{\max}}} - \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min}}}. \tag{10}$$

Define $\gamma_0'$ as the unique positive solution of

$$28\overline{\Delta}\gamma_0'\left(1 + \sqrt{\ln(1+N\gamma_0')}\right) = \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min} - k_{\max}}} - \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min}}}. \tag{11}$$

Because $k_{\max} > 0$, $k_{\max} < N_{\min}$, and $\overline{\Delta} > 0$, the right-hand side is strictly positive, while the left-hand side is continuous and strictly increasing from zero to infinity. Thus, the solution exists and is unique, and

$$B_G'(\gamma) < B_R' \quad \Longleftrightarrow \quad \gamma < \gamma_0'. \tag{12}$$

Importantly, $\gamma_0'$ does not depend on the realized rewards, $\mathcal{J}$, $\mathcal{H}$, or the selected source.

**Algorithm 7** Multi-source unlearning algorithm

**Input:** Unlearning request $U = \bigcup_{i=1}^{\ell} U_i$ from $a_{u_1}, \ldots, a_{u_\ell}$, $|U_i| = k_i$, additional statistics $T(D)$: $\widehat{\boldsymbol{f}}(D)$, $\widehat{\boldsymbol{\mu}}$, $\widehat{\boldsymbol{N}}$, a confidence level $\tau \in (0,1)$, $\varepsilon \in (0,\infty)$, $\delta \in (0,1)$

Set $\widehat{\boldsymbol{f}}(D') \leftarrow \widehat{\boldsymbol{f}}(D)$, $\widehat{\boldsymbol{N}}' \leftarrow \widehat{\boldsymbol{N}} - \sum_{i=1}^{\ell} k_i \boldsymbol{I}\{a = a_{u_i}\}$, $\widehat{\boldsymbol{\mu}}' \leftarrow \widehat{\boldsymbol{\mu}}$
**for** $i = 1, \ldots, \ell$ **do**
  Compute $\hat{\mu}'(a_{u_i}) \leftarrow \frac{\hat{\mu}(a_{u_i})N(a_{u_i}) - \sum_{(a_{u_i}, r) \in U_i} r}{N'(a_{u_i})}$
  Compute $b'(a_{u_i}) \leftarrow \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a_{u_i})}}$ and set $\widehat{\boldsymbol{f}}_{(a_{u_i})}(D') \leftarrow \hat{\mu}'(a_{u_i}) - b'(a_{u_i})$
  Compute $\Delta_i \leftarrow \frac{3k_i}{2N(a_{u_i})}$
Compute $\overline{\Delta} \leftarrow \max_{i \in [\ell]} \Delta_i$, $B_G'$ and $B_R'$ from (10) and construct $\mathcal{J}$ from (6)
**if** $\mathcal{J} \neq \emptyset$ and $B_G' < B_R'$ **then**
  Construct $\mathcal{H}$ from (7)
  **if** $|\mathcal{H}| = 1$ **then** Let $j$ be the unique element of $\mathcal{H}$ and set $\sigma_j \leftarrow \gamma \Delta_j$
  **else**
    Compute $\delta_G$ and $\omega$ from (9), and sample $Z \sim \text{Bernoulli}(\omega)$
    **if** $Z = 1$ **then** Sample $j \in \mathcal{H}$ according to $p_j$ in (8)
    **else**
      Set $j \leftarrow \min \arg\max_{i \in \mathcal{H}} \widehat{\boldsymbol{f}}_{(a_{u_i})}(D)$
    Set $\sigma_j \leftarrow 2\gamma \Delta_j$
  Compute $\beta_j \leftarrow \sqrt{2 \ln(1 + N\sigma_j)}$ and sample $\nu_j \sim \mathcal{N}(0, \sigma_j^2)$
  Set $\widehat{\boldsymbol{f}}_{(a_{u_j})}(D') \leftarrow \widehat{\boldsymbol{f}}_{(a_{u_j})}(D) + \nu_j - \beta_j \sigma_j$ ▷ *Downward-shifted Gaussian mechanism*
**else**
  Keep $\widehat{\boldsymbol{f}}(D')$ unchanged ▷ *Rollback*

**Output:** $\hat{a}' = \arg\max_a \widehat{\boldsymbol{f}}_{(a)}(D')$

For the empty-request execution, the request descriptor is $\rho(U) = \{(a_{u_i}, k_i)\}_{i=1}^{\ell}$.

**Algorithm 8** Multi-source unlearning algorithm with empty request

**Input:** Request descriptor $\rho(U) = \{(a_{u_i}, k_i)\}_{i=1}^{\ell}$, additional statistics $T(D \setminus U)$: $\widehat{\boldsymbol{f}}(D \setminus U)$, $\widehat{\boldsymbol{\mu}}'$, $\widehat{\boldsymbol{N}}'$, a confidence level $\tau \in (0,1)$, $\varepsilon \in (0,\infty)$, $\delta \in (0,1)$

Set $\widehat{\boldsymbol{f}}(D') \leftarrow \widehat{\boldsymbol{f}}(D \setminus U)$, $\widehat{\boldsymbol{N}} \leftarrow \widehat{\boldsymbol{N}}' + \sum_{i=1}^{\ell} k_i \boldsymbol{I}\{a = a_{u_i}\}$
**for** $i = 1, \ldots, \ell$ **do**
  Compute $\Delta_i \leftarrow \frac{3k_i}{2N(a_{u_i})}$
Compute $\overline{\Delta} \leftarrow \max_{i \in [\ell]} \Delta_i$, $B_G'$ and $B_R'$ from (10) and construct $\mathcal{J}$ from (6)
**if** $\mathcal{J} \neq \emptyset$ and $B_G' < B_R'$ **then**
  Construct $\mathcal{H}$ from (7)
  **if** $|\mathcal{H}| = 1$ **then** Let $j$ be the unique element of $\mathcal{H}$ and set $\sigma_j \leftarrow \gamma \Delta_j$
  **else**
    Compute $\delta_G$ and $\omega$ from (9), and sample $Z \sim \text{Bernoulli}(\omega)$
    **if** $Z = 1$ **then** Sample $j \in \mathcal{H}$ according to $q_j$ in (8)
    **else**
      Set $j \leftarrow \min \arg\max_{i \in \mathcal{H}} \widehat{\boldsymbol{f}}_{(a_{u_i})}(D')$
    Set $\sigma_j \leftarrow 2\gamma \Delta_j$
  Compute $\beta_j \leftarrow \sqrt{2 \ln(1 + N\sigma_j)}$ and sample $\nu_j \sim \mathcal{N}(0, \sigma_j^2)$
  Set $\widehat{\boldsymbol{f}}_{(a_{u_j})}(D') \leftarrow \widehat{\boldsymbol{f}}_{(a_{u_j})}(D') + \nu_j - \beta_j \sigma_j$ ▷ *Downward-shifted Gaussian mechanism*
**else**
  Keep $\widehat{\boldsymbol{f}}(D')$ unchanged ▷ *Rollback*

**Output:** $\hat{a}' = \arg\max_a \widehat{\boldsymbol{f}}_{(a)}(D')$

**Remark.** *All sets used in the switching decision and in the candidate refinement are functions only of the retained-data state, the request descriptor, and public parameters. The actual and empty-request executions therefore construct the same $\mathcal{J}$ and $\mathcal{H}$ and take the same Gaussian-versus-rollback branch. Whenever an exact maximizer is required, both algorithms use the smallest maximizing index. The two exact maximizers inside the exceptional component need not coincide; their possible discrepancy is accounted for by the $\delta$ term in the unlearning guarantee.*

**Lemma 22** (Common candidate set)**.** *Suppose $\left|\widehat{\boldsymbol{f}}_{(a_{u_i})}(D) - \widehat{\boldsymbol{f}}_{(a_{u_i})}(D')\right| \le \Delta_i \le \overline{\Delta}$ holds and $\mathcal{J} \neq \emptyset$. Let $i^* := \min \arg\max_{i\in\mathcal{J}} f_i$. Then $i^* \in \mathcal{H}$, and every $j \in \mathcal{H}$ satisfies $0 \le \max_{i\in\mathcal{J}} f_i - f_j \le 4\overline{\Delta}$. Moreover, suppose the simultaneous confidence event on $D'$ holds. If the optimal arm is a requested source, say $a^* = a_{u_s}$, then $s \in \mathcal{J}$.*

*Proof.* Let $r \in \arg\max_{i\in\mathcal{J}} h_i$. Since $i^*$ maximizes $f_i$ over $\mathcal{J}$, $\left|\widehat{\boldsymbol{f}}_{(a_{u_i})}(D) - \widehat{\boldsymbol{f}}_{(a_{u_i})}(D')\right| \le \overline{\Delta}$ gives $h_{i^*} \ge f_{i^*} - \overline{\Delta} \ge f_r - \overline{\Delta} \ge h_r - 2\overline{\Delta}$. Thus $i^* \in \mathcal{H}$. For any $j \in \mathcal{H}$,

$$f_{i^*} \le h_{i^*} + \overline{\Delta} \le \max_{i\in\mathcal{J}} h_i + \overline{\Delta}, \qquad f_j \ge h_j - \overline{\Delta} \ge \max_{i\in\mathcal{J}} h_i - 3\overline{\Delta}.$$

Subtracting the two displays proves $0 \le \max_{i\in\mathcal{J}} f_i - f_j \le 4\overline{\Delta}$.

It remains to prove the screening claim. Suppose, toward a contradiction, that $a^* = a_{u_s}$ but $s \notin \mathcal{J}$. Then there exists an arm $a \neq a^*$ such that

$$\hat{\mu}'(a^*) + \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a^*)}} < \hat{\mu}'(a) - \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}}.$$

On the retained-data confidence event, the left-hand side is at least $\mu(a^*)$, while the right-hand side is at most $\mu(a)$. This would imply $\mu(a^*) < \mu(a)$, contradicting the optimality of $a^*$. Therefore $s \in \mathcal{J}$. □

**Theorem 23** (Multi-source $(\varepsilon,\delta)$-unlearning)**.** *Algorithm 7, Algorithm 8, and Algorithm 1 satisfy $(\varepsilon,\delta)$-unlearning when $k_i \le N(a_{u_i}) - \sqrt{\frac{N(a_{u_i})\ln\frac{m}{\tau}}{2}}$, $\forall i \in [\ell]$, and $k_{\max} < N_{\min}$.*

*Proof.* Let $D' = D \setminus U$. Both executions first construct the same retained-data LCB vector $\widehat{\boldsymbol{f}}(D')$. The actual execution computes it by removing $U$ from the stored statistics of $D$, whereas the empty-request execution receives the same statistics through $T(D \setminus U)$. Consequently, they construct the same sets $\mathcal{J}$ and $\mathcal{H}$, the same sensitivity bounds $\Delta_i$ and $\overline{\Delta}$, and the same quantities $B'_G$ and $B'_R$. Hence the Gaussian-versus-rollback decision is identical in the two executions.

If $\mathcal{J} = \emptyset$ or $B'_G \ge B'_R$, both executions perform exact rollback and release the same vector $\widehat{\boldsymbol{f}}(D')$. Their output distributions are therefore identical. It remains to consider the Gaussian branch.

Suppose first that $|\mathcal{H}| = 1$, and let $j$ denote its unique element. Both executions perturb the same coordinate using $\sigma_j = \gamma\Delta_j$. The two Gaussian centers are $f_j - \beta_j\sigma_j$ and $h_j - \beta_j\sigma_j$. Hence their distance is at most $\Delta_j$. Therefore, the analytic Gaussian mechanism of Balle & Wang (2018) and (1) imply two-sided $(\varepsilon,\delta)$ indistinguishability for this coordinate. Every other coordinate equals its retained-data LCB in both executions.

Now suppose that $|\mathcal{H}| \ge 2$. We first analyze the component corresponding to $Z = 1$. Write

$$p_j = \frac{\exp\left(\frac{\varepsilon f_j}{4\overline{\Delta}}\right)}{\sum_{i\in\mathcal{H}}\exp\left(\frac{\varepsilon f_i}{4\overline{\Delta}}\right)}, \qquad q_j = \frac{\exp\left(\frac{\varepsilon h_j}{4\overline{\Delta}}\right)}{\sum_{i\in\mathcal{H}}\exp\left(\frac{\varepsilon h_i}{4\overline{\Delta}}\right)}.$$

For every $j \in \mathcal{H}$,

$$\frac{p_j}{q_j} = \exp\left(\frac{\varepsilon(f_j - h_j)}{4\overline{\Delta}}\right)\frac{\sum_{i\in\mathcal{H}}\exp\left(\frac{\varepsilon h_i}{4\overline{\Delta}}\right)}{\sum_{i\in\mathcal{H}}\exp\left(\frac{\varepsilon f_i}{4\overline{\Delta}}\right)} \le e^{\varepsilon/4}e^{\varepsilon/4} = e^{\varepsilon/2}.$$

The reverse ratio is bounded in the same way. Thus, the source-selection step in the $Z = 1$ component satisfies two-sided $(\varepsilon/2, 0)$ indistinguishability.

Conditional on selecting a fixed $j$, both executions use $\sigma_j = 2\gamma\Delta_j$. At the largest possible center distance $\Delta_j$, the analytic Gaussian privacy profile with privacy parameter $\varepsilon/2$ is

$$\Phi\left(\frac{\Delta_j}{2\sigma_j} - \frac{(\varepsilon/2)\sigma_j}{\Delta_j}\right) - e^{\varepsilon/2}\Phi\left(-\frac{\Delta_j}{2\sigma_j} - \frac{(\varepsilon/2)\sigma_j}{\Delta_j}\right) = \Phi\left(\frac{1}{4\gamma} - \varepsilon\gamma\right) - e^{\varepsilon/2}\Phi\left(-\frac{1}{4\gamma} - \varepsilon\gamma\right) = \delta_{\mathrm{G}}.$$

By Theorem 8 and Lemma 7 of Balle & Wang (2018), the conditional Gaussian step therefore satisfies two-sided $(\varepsilon/2, \delta_{\mathrm{G}})$ indistinguishability.

For completeness, consider the augmented output that contains the selected index and the perturbed LCB vector. For an arbitrary measurable event $\mathcal{B}$, let $\mathcal{B}_j$ denote its section conditional on the selected index being $j$, and let $P_j, Q_j$ be the corresponding conditional Gaussian distributions. Then

$$\sum_{j\in\mathcal{H}} p_j P_j(\mathcal{B}_j) \le e^{\varepsilon/2}\sum_{j\in\mathcal{H}} p_j Q_j(\mathcal{B}_j) + \delta_{\mathrm{G}} \le e^{\varepsilon}\sum_{j\in\mathcal{H}} q_j Q_j(\mathcal{B}_j) + \delta_{\mathrm{G}}.$$

The reverse inequality follows identically. Hence the entire $Z = 1$ component is $(\varepsilon, \delta_{\mathrm{G}})$-indistinguishable. Let $P_0, Q_0$ denote the two distributions of this component and let $P_1, Q_1$ denote the distributions of the $Z = 0$ component. The latter component may select different exact maximizers in the two executions, so we conservatively treat $P_1, Q_1$ as arbitrary distributions. The full Gaussian-branch distributions are

$$P = \omega P_0 + (1-\omega)P_1, \qquad Q = \omega Q_0 + (1-\omega)Q_1.$$

For every measurable event $\mathcal{B}$,

$$P(\mathcal{B}) \le \omega\left(e^{\varepsilon}Q_0(\mathcal{B}) + \delta_{\mathrm{G}}\right) + 1 - \omega \le e^{\varepsilon}Q(\mathcal{B}) + \omega\delta_{\mathrm{G}} + 1 - \omega = e^{\varepsilon}Q(\mathcal{B}) + \delta,$$

where the last equality follows from (9). The same calculation with $P$ and $Q$ interchanged proves the reverse inequality. Thus, in every case, the released LCB vectors satisfy two-sided $(\varepsilon, \delta)$ indistinguishability. Applying $g(\boldsymbol{f}) := \arg\max_{a\in\mathcal{A}} \boldsymbol{f}_{(a)}$ is deterministic post-processing. Therefore Algorithm 7, Algorithm 8, and Algorithm 1 satisfy $(\varepsilon, \delta)$-unlearning. □

We next establish the uniform relation that allows the source-selection cost to be expressed using the analytic multiplier $\gamma$ over the full parameter range $\varepsilon > 0$ and $\delta \in (0, 1)$.

**Lemma 24** (Full-range selection attenuation)**.** *Let $N \ge 2$ and let $\omega$ be defined in* (9)*. Then, for every $\varepsilon > 0$ and $\delta \in (0, 1)$, $\omega \min\left\{1, \frac{\ln N}{\varepsilon}\right\} \le 6\gamma\left(1 + \sqrt{\ln(1 + N\gamma)}\right)$.*

*Proof.* Let $r := \varepsilon\gamma^2$, $z := \ln(1 + N\gamma)$. Suppose first that $r \ge 1/4$. Then $1/\varepsilon \le 4\gamma^2$, and since $\omega \le 1$, $\omega \min\left\{1, \frac{\ln N}{\varepsilon}\right\} \le \min\left\{1, 4\gamma^2 \ln N\right\}$.

If $\gamma \ge N^{-1/2}$, then $\ln N \le 2z$ and $\min\left\{1, 4\gamma^2 \ln N\right\} \le \min\left\{1, 8\gamma^2 z\right\} \le \sqrt{8}\,\gamma\sqrt{z}$. If $\gamma < N^{-1/2}$, then $4\gamma^2 \ln N \le 4\gamma\frac{\ln N}{\sqrt{N}} \le \frac{8}{e}\gamma$. Thus, in both subcases, $\omega \min\left\{1, \frac{\ln N}{\varepsilon}\right\}$ is at most $3\gamma(1 + \sqrt{z})$. Since $r < 1/4$, we have $\frac{1}{2\gamma} - \varepsilon\gamma > \frac{1}{4\gamma} > 0$, $\frac{1}{2\gamma} + \varepsilon\gamma > \frac{1}{2\gamma} > 0$, $\frac{1}{4\gamma} - \varepsilon\gamma > 0$. Using $\Psi_\varepsilon(\gamma) = \delta$, the Gaussian Mills inequality, and $e^{\varepsilon}\phi(\frac{1}{2\gamma} + \varepsilon\gamma > \frac{1}{2\gamma}) = \phi(\frac{1}{2\gamma} - \varepsilon\gamma)$, we obtain

$$\begin{aligned} 1 - \delta &= \Phi(-\frac{1}{2\gamma} + \varepsilon\gamma) + e^{\varepsilon}\Phi(-\frac{1}{2\gamma} - \varepsilon\gamma > \frac{1}{2\gamma}) \\ &\le \frac{\phi(\frac{1}{2\gamma} - \varepsilon\gamma)}{\frac{1}{2\gamma} - \varepsilon\gamma} + e^{\varepsilon}\frac{\phi(\frac{1}{2\gamma} + \varepsilon\gamma > \frac{1}{2\gamma})}{\frac{1}{2\gamma} + \varepsilon\gamma > \frac{1}{2\gamma}} \le 6\gamma\phi(\frac{1}{2\gamma} - \varepsilon\gamma). \end{aligned}$$

On the other hand, $1 - \delta_{\mathrm{G}} \ge \Phi(-\frac{1}{4\gamma} + \varepsilon\gamma)$.

If $\gamma \ge 1/4$, then $\omega \le 1 \le 4\gamma$. If $\gamma < 1/4$, then $(\frac{1}{2\gamma} - \varepsilon\gamma) - (\frac{1}{4\gamma} - \varepsilon\gamma) = \frac{1}{4\gamma} > 1$.

Because $\frac{1}{4\gamma} - \varepsilon\gamma > 0$ and $\phi$ is decreasing on the positive half-line,

$$\Phi(-\frac{1}{4\gamma} + \varepsilon\gamma) = \int_{\frac{1}{4\gamma} - \varepsilon\gamma}^{\infty} \phi(t)\,dt \ge \int_{\frac{1}{4\gamma} - \varepsilon\gamma}^{\frac{1}{4\gamma} - \varepsilon\gamma + 1} \phi(t)\,dt \ge \phi(\frac{1}{4\gamma} - \varepsilon\gamma + 1) \ge \phi(\frac{1}{2\gamma} - \varepsilon\gamma).$$

Consequently, $\omega = \frac{1-\delta}{1-\delta_{\mathrm{G}}} \le 6\gamma$ in either subcase. Notice $\min\{1, \ln N/\varepsilon\} \le 1$. Q.E.D. □

**Lemma 25** (Source-selection and Gaussian costs)**.** *In the Gaussian branch, let $j$ be the source selected by Algorithm 7, then*

$$\mathbf{E}_j\left[\max_{i\in\mathcal{J}} f_i - f_j\right] \le 24\overline{\Delta}\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right).$$

*Furthermore, for every selected $j$,*

$$\sigma_j\left(\beta_j + \frac{1}{\sqrt{2\pi}}\right) \le 4\overline{\Delta}\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right), \qquad \mathbf{E}_{\nu_j}\left[(\nu_j - \beta_j\sigma_j)_+\right] \le \frac{1}{\sqrt{2\pi}N}.$$

*Proof.* If $|\mathcal{H}| = 1$, Lemma 22 implies that its unique element maximizes $f_i$ over $\mathcal{J}$, so $\max_{i\in\mathcal{J}} f_i - f_j = 0$. Suppose now that $|\mathcal{H}| \ge 2$. Conditional on $Z = 0$, the actual execution selects an exact maximizer of $f_i$ over $\mathcal{H}$. Because an $f_i$-maximizer over $\mathcal{J}$ belongs to $\mathcal{H}$, this again gives $\max_{i\in\mathcal{J}} f_i - f_j = 0$. Conditional on $Z = 1$, the entropy bound for exponential weights gives $\mathbf{E}\left[\max_{i\in\mathcal{J}} f_i - f_j \mid Z = 1\right] \le \frac{4\overline{\Delta}}{\varepsilon}\ln|\mathcal{H}|$.

In addition, Lemma 22 gives the deterministic bound $\max_{i\in\mathcal{J}} f_i - f_j \le 4\overline{\Delta}$. Since $|\mathcal{H}| \le \ell \le N$,

$$\mathbf{E}_j\left[\max_{i\in\mathcal{J}} f_i - f_j\right] \le 4\overline{\Delta}\omega\min\left\{1, \frac{\ln N}{\varepsilon}\right\} \le 24\overline{\Delta}\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right).$$

For the Gaussian cost, both the singleton and non-singleton cases satisfy $\sigma_j \le 2\gamma\Delta_j \le 2\gamma\overline{\Delta}$. Moreover, $k_{\max} < N_{\min}$ implies $\overline{\Delta} < 3/2$, and hence $\beta_j \le \sqrt{2\ln(1+3N\gamma)}$.

For $N\gamma \ge 0$, $\ln(1+3N\gamma) \le \ln 3 + \ln(1+N\gamma)$, which implies

$$\sqrt{2\ln(1+3N\gamma)} + \frac{1}{\sqrt{2\pi}} \le 2\left(1+\sqrt{\ln(1+N\gamma)}\right).$$

Finally, conditional on $j$, write $\nu_j = \sigma_j Z$ with $Z \sim \mathcal{N}(0,1)$. Since $\beta_j = \sqrt{2\ln(1+N\sigma_j)}$,

$$\mathbf{E}_{\nu_j}\left[(\nu_j - \beta_j\sigma_j)_+\right] \le \sigma_j\phi(\beta_j) = \frac{\sigma_j}{\sqrt{2\pi}(1+N\sigma_j)} \le \frac{1}{\sqrt{2\pi}N}.$$

□

**Theorem 26** (Theorem 9 restated)**.** *Consider any offline dataset $D(\boldsymbol{N})$ satisfying $N(a^*) \ge N_*$, and any reward-independent unlearning request $U = \bigcup_{i=1}^{\ell} U_i$, with $|U_i| = k_i$ from arm $a_{u_i}$. When $k_i \le N(a_{u_i}) - \sqrt{\frac{N(a_{u_i})\ln\frac{m}{\tau}}{2}}, \forall i \in [\ell]$, $k_{\max} < N_{\min}$, by setting $\tau = \frac{1}{N}$, the arm $\hat{a}'$ returned by Algorithm 7 satisfies*

$$\texttt{SubOpt}(\boldsymbol{N}, N_*, \{k_i\}_{i=1}^{\ell}, \varepsilon, \delta) =$$
$$\begin{cases} O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N_{\min}}} + \frac{k_{\max}\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N_{\min}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma < \gamma_0', \\ O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N_{\min}-k_{\max}}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right), \gamma \ge \gamma_0'. \end{cases}$$

*Proof.* Let $D' = D \setminus U$. Define the simultaneous confidence events on $D$ and $D'$, respectively, by

$$\mathcal{E}_1 := \left\{|\hat{\mu}(a) - \mu(a)| \le \sqrt{\frac{\ln\frac{m}{\tau}}{2N(a)}},\ \forall a \in \mathcal{A}\right\}, \qquad \mathcal{E}_2 := \left\{|\hat{\mu}'(a) - \mu(a)| \le \sqrt{\frac{\ln\frac{m}{\tau}}{2N'(a)}},\ \forall a \in \mathcal{A}\right\}.$$

Because the request is reward-independent, Hoeffding's inequality and a union bound give $\mathbf{Pr}[\mathcal{E}_1^c] \le 2\tau$, $\mathbf{Pr}[\mathcal{E}_2^c] \le 2\tau$. Let $\mathcal{E} := \mathcal{E}_1 \cap \mathcal{E}_2$. Then $\mathbf{Pr}[\mathcal{E}^c] \le 4\tau$.

We first consider the regime $\gamma < \gamma_0'$. By (12), $B_G'(\gamma) < B_R'$.

Suppose first that the optimal arm is requested, so that $a^* = a_{u_s}$ for some $s \in [\ell]$. On $\mathcal{E}$, Lemma 22 gives $s \in \mathcal{J}$. Consequently, $\mathcal{J} \neq \emptyset$ and the algorithm takes the Gaussian branch. Since $s \in \mathcal{J}$ and $\mathcal{E}_1$ holds,

$$f_j = \max_{i\in\mathcal{J}} f_i - (\max_{i\in\mathcal{J}} f_i - f_j) \geq f_s - (\max_{i\in\mathcal{J}} f_i - f_j)$$
$$\geq \mu(a^*) - \sqrt{\frac{2\ln\frac{m}{\tau}}{N(a^*)}} - (\max_{i\in\mathcal{J}} f_i - f_j) \geq \mu(a^*) - \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min}}} - (\max_{i\in\mathcal{J}} f_i - f_j).$$

Fix a realization in $\mathcal{E}$ and condition on the selected source $j$. If $\hat{a}' = a_{u_j}$, the pessimism of the original-data LCB on $\mathcal{E}_1$ gives $f_j \leq \mu(a_{u_j})$. Combining this with below,

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min}}} + (\max_{i\in\mathcal{J}} f_i - f_j).$$

If $\hat{a}' \neq a_{u_j}$, its released score is its retained-data LCB. By maximality of the output, $\widehat{\boldsymbol{f}}_{(\hat{a}')}(D') \geq f_j + \nu_j - \beta_j\sigma_j$. On $\mathcal{E}_2$, $\widehat{\boldsymbol{f}}_{(\hat{a}')}(D') \leq \mu(\hat{a}')$. Therefore,

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min}}} + (\max_{i\in\mathcal{J}} f_i - f_j) + \beta_j\sigma_j - \nu_j.$$

Combining the two output cases yields

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min}}} + (\max_{i\in\mathcal{J}} f_i - f_j) + (\beta_j\sigma_j - \nu_j)_+.$$

Conditional on $j$,

$$\mathbf{E}_{\nu_j}\left[(\beta_j\sigma_j - \nu_j)_+\right] \leq \sigma_j\left(\beta_j + \frac{1}{\sqrt{2\pi}}\right).$$

Applying Lemma 25, on $\mathcal{E}$, we have

$$\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right] \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min}}} + 28\overline{\Delta}\gamma\left(1 + \sqrt{\ln(1+N\gamma)}\right).$$

Since rewards lie in $[0,1]$, the regret is at most one. Accounting for $\mathcal{E}^c$ therefore yields

$$\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right] \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min}}} + 28\overline{\Delta}\gamma\left(1 + \sqrt{\ln(1+N\gamma)}\right) + 4\tau.$$

We next consider $a^* \notin \{a_{u_1}, \ldots, a_{u_\ell}\}$. In this case deletion does not modify the optimal-arm statistics, so $N'(a^*) = N(a^*) \geq N_*$, $\widehat{\boldsymbol{f}}_{(a^*)}(D') = \widehat{\boldsymbol{f}}_{(a^*)}(D)$. On $\mathcal{E}$, $\widehat{\boldsymbol{f}}_{(a^*)}(D') \geq \mu(a^*) - \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}}$.

If $\mathcal{J} = \emptyset$, the algorithm performs rollback, and the standard LCB comparison gives

$$\mu(a^*) - \mu(\hat{a}') \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}}.$$

Suppose instead that $\mathcal{J} \neq \emptyset$, so the algorithm takes the Gaussian branch. If $\hat{a}' \neq a_{u_j}$, maximality of the output and $\widehat{\boldsymbol{f}}_{(a^*)}(D') \geq \mu(a^*) - \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}}$ again give the preceding bound. If $\hat{a}' = a_{u_j}$, then $f_j + \nu_j - \beta_j\sigma_j \geq \widehat{\boldsymbol{f}}_{(a^*)}(D')$.

Using $f_j \leq \mu(a_{u_j})$ on $\mathcal{E}_1$ gives

$$\mu(a^*) - \mu(a_{u_j}) \leq \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}} + \nu_j - \beta_j\sigma_j.$$

Consequently, in both output cases,

$$\mu(a^*) - \mu(\hat{a}') \le \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}} + (\nu_j - \beta_j\sigma_j)_+.$$

By Lemma 25, this implies

$$\mathbf{E}\left[\mu(a^*) - \mu(\hat{a}')\right] \le \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}} + \frac{1}{\sqrt{2\pi}N} + 4\tau.$$

The same display also covers the rollback case because its conditional Gaussian-tail term is zero. Hence, when $\gamma < \gamma_0'$, setting $\tau = 1/N$ gives

$$\texttt{SubOpt}\left(\boldsymbol{N}, N_*, \{k_i\}_{i=1}^{\ell}, \varepsilon, \delta\right) = O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N_{\min}}} + \frac{k_{\max}\gamma\left(1+\sqrt{\ln(1+N\gamma)}\right)}{N_{\min}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right).$$

Finally, suppose $\gamma \ge \gamma_0'$. By (12), $B_G'(\gamma) \ge B_R'$, and the algorithm performs exact rollback. If $a^* = a_{u_s}$ is requested, then $N'(a^*) = N(a_{u_s}) - k_s \ge N_{\min} - k_{\max}$.

On $\mathcal{E}_2$, the standard LCB comparison yields

$$\mu(a^*) - \mu(\hat{a}') \le \sqrt{\frac{2\ln\frac{m}{\tau}}{N_{\min} - k_{\max}}}.$$

If $a^*$ is not requested, then $N'(a^*) = N(a^*) \ge N_*$ and

$$\mu(a^*) - \mu(\hat{a}') \le \sqrt{\frac{2\ln\frac{m}{\tau}}{N_*}}.$$

Accounting for $\mathbf{Pr}\left[\mathcal{E}_2^c\right] \le 2\tau$ and setting $\tau = 1/N$ gives

$$\texttt{SubOpt}\left(\boldsymbol{N}, N_*, \{k_i\}_{i=1}^{\ell}, \varepsilon, \delta\right) = O\left(\max\left\{\sqrt{\frac{\ln(Nm)}{N_{\min} - k_{\max}}}, \sqrt{\frac{\ln(Nm)}{N_*}}\right\}\right).$$

□

# B EXPERIMENTS

The experiments are performed on a Dell Inspiron 16 PLUS 7620 laptop equipped with a 12th Generation Intel(R) Core(TM) i7-12700H CPU at 2.30 GHz and 16 GB of RAM. All experiments are conducted using CPU computation only, without GPU acceleration, cloud instances, or internal clusters. Each reported experimental run can be completed within a few hours on this machine, and the total compute required for the reported experiments is modest.

## B.1 ADDITIONAL EXPERIMENTAL DETAILS OF SECTION 4

**Dataset construction.** We construct a real-world bandit instance from the Yelp-derived[2] user-item feature data. Specifically, we use a subset of users and items and form all user-item pairs in this subset. For each pair, we compute a real-valued preference score from the corresponding user-preference vector and item-feature vector, and then normalize all scores to $[0, 1]$ to match the bounded-reward assumption in our bandit model. To obtain a finite-armed bandit instance, we partition the resulting user-item pairs into $|A| = 5$ arms by a fixed deterministic rule based on the pair index. Each arm therefore corresponds to a disjoint subset of user-item pairs rather than to a manually designed semantic category. The mean reward of each arm is estimated from the full constructed pool and is used only for evaluation and for identifying $a^*$.

[2] https://business.yelp.com/data/resources/open-dataset/.

**Experimental setup.** For the fixed-sample model $\mathcal{M}_f$, deletion requests are implemented at the block level. When varying $N$ or $\gamma$, we delete the samples of a designated arm $a_0$ contained in the first five blocks. When varying the deletion size $k$, we fix $N = 3000$ and delete all samples of $a_0$ contained in between one and 25 consecutive blocks. Under the round-robin behavior policy, each block of size $B = 100$ contains approximately $B/5 = 20$ samples per arm. Consequently, deleting between one and 25 blocks corresponds to deletion sizes ranging from $k = 20$ to $k = 500$, consistent with the range reported in Figure 1. Each prefix is evaluated using five independent deletion requests.

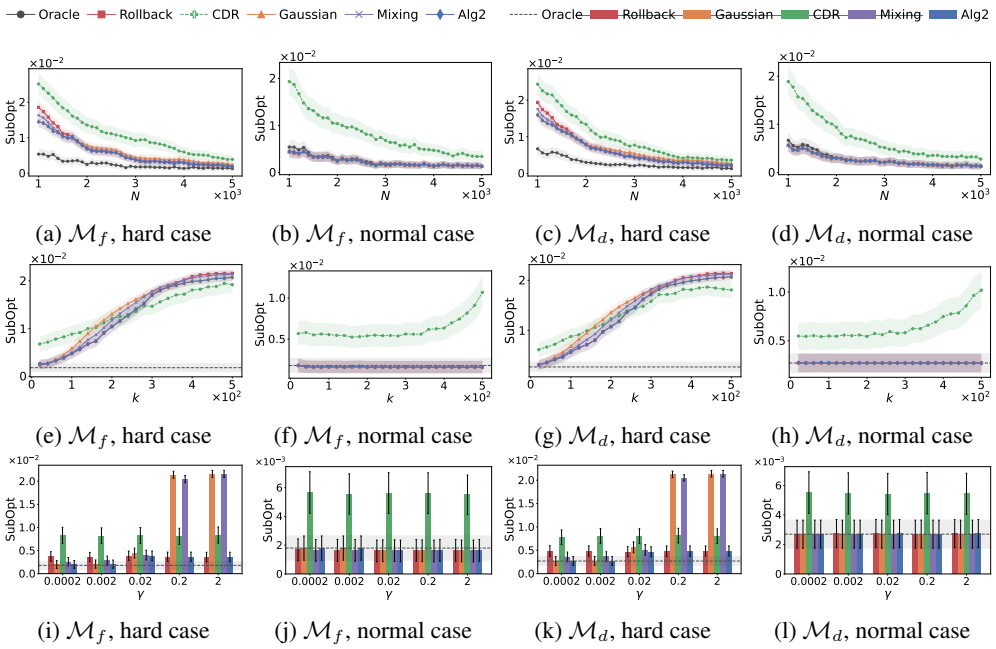
For the distribution model $\mathcal{M}_d$, we study the regime $C^* \in [2, \infty)$ using a stochastic behavior policy that selects the optimal arm with probability $1/C^*$, where $C^* = 5$, and distributes the remaining probability mass uniformly across the suboptimal arms. Unlike the fixed-sample setting, the realized arm counts in the distribution model are random and may vary across arms even under a uniform behavior policy. We consider prefixes of length $N \in \{1000, \ldots, 5000\}$. When varying $N$ or $\gamma$, each deletion request removes $k = 80$ samples from the first four blocks in a designated arm. When varying $k$, we fix $N = 3000$ and remove $k$ consecutive samples, with $k$ ranging from 20 to 500. Each prefix is evaluated using five independent deletion requests, and the results for each configuration are averaged over 10 runs.

**CDR adaptation.** CDR denotes our bandit adaptation of the certified data-removal method of Guo et al. (2020). We encode an observation from arm $a$ by the one-hot vector $e_a$ and fit a five-coordinate $\ell_2$-regularized logistic model. Since the rewards in both Yelp and synthetic experiments lie in $[0, 1]$, we use the convex soft-label logistic loss $\ell(w; (a, r)) = \log(1 + \exp(w_a)) - rw_a$; hence $\mathrm{sigmoid}(w_a)$ estimates the mean reward of arm $a$. Following the public implementation's documented configuration, we set the regularization coefficient to $10^{-3}$ and draw the objective-perturbation vector from $\mathcal{N}(0, 10^2 I)$. For a deletion batch, we apply the retained-Hessian Newton update of Guo et al. (2020) and evaluate its logistic residual certificate. If the residual exceeds the certified budget, we retrain on the retained data with fresh objective noise. Finally, to make its recommendation comparable to the other bandit methods, we score each arm by $\mathrm{sigmoid}(w'_a) - \sqrt{\ln(m/\tau)/(2N'(a))}$ and return the arm with the largest score. The same construction is applied coordinate-wise for multi-source deletion batches.

### B.2 Synthetic Results

We generate 200 independent synthetic offline bandit datasets using a round-robin behavior policy over 5 arms with Bernoulli rewards $\boldsymbol{\mu} = (0.10, 0.08, 0.06, 0.04, 0.02)$. To efficiently evaluate deletions, we adopt a prefix-sharing strategy following that in Appendix B.1. In both settings, each prefix undergoes 5 independent deletion requests. We consider the hard case $a_0 = a^*$, which stresses the deletion-induced term in our bounds, and a normal case where $a_0$ is the second-best arm, which isolates the intrinsic offline-learning term. Results are averaged over 10 runs per configuration. We vary one of $N$, $k$, or $\gamma$ while fixing the remaining two parameters to $(N, k, \gamma) = (3000, 80, 0.02)$. Error bars report 95% confidence intervals across runs.

Figure 2 shows that Algorithm 2 maintains robust and competitive performance across the evaluated settings under $\mathcal{M}_f$, yielding low sub-optimality over a wide range of $N$, $k$, and $\gamma$. In the hard case, the algorithm adaptively balances rollback and the Gaussian mechanism: it favors Gaussian mechanism when $\frac{k}{N}$ is large and selects rollback when $\frac{k}{N}$ is small. When varying $N$, for $N \geq 2100$, Algorithm 2 reduces sub-optimality by 9–29% and 46–66% relative to the Gaussian mechanism and CDR, respectively. Compared with the mixing algorithm, it remains within 2% when mixing performs better and achieves reductions of up to 15% when it performs better. For $N \leq 1200$, it achieves 16–22% and 41–42% reductions relative to rollback and CDR, respectively. In the normal case, Algorithm 2 reduces sub-optimality by 56–79% relative to CDR and remains within 6% of the best-performing baseline. Similar adaptive behavior is observed when varying $k$ or $\gamma$. Overall, Algorithm 2 consistently achieves low sub-optimality and ranks among the best-performing methods across most regimes, whereas the relative performance of the individual base mechanisms varies notably across settings.




(a) $\mathcal{M}_f$, hard case (b) $\mathcal{M}_f$, normal case (c) $\mathcal{M}_d$, hard case (d) $\mathcal{M}_d$, normal case

(e) $\mathcal{M}_f$, hard case (f) $\mathcal{M}_f$, normal case (g) $\mathcal{M}_d$, hard case (h) $\mathcal{M}_d$, normal case

(i) $\mathcal{M}_f$, hard case (j) $\mathcal{M}_f$, normal case (k) $\mathcal{M}_d$, hard case (l) $\mathcal{M}_d$, normal case

Figure 2: Comparison of Algorithm 2 with baselines under $\mathcal{M}_f$ and $\mathcal{M}_d$ ($C^* = 5$) on synthetic dataset.

When $C^* \in [2, \infty)$, we generate 200 independent synthetic offline datasets using a stochastic behavior policy that selects $a^*$ with probability $\frac{1}{C^*}$, where $C^* = 5$, and distributes the remaining probability uniformly over the suboptimal arms. All other configurations are identical to those under $\mathcal{M}_f$. Figure 2 shows that Algorithm 2 also maintains robust and competitive performance under $\mathcal{M}_d$. In the hard case, it adaptively switches between the Gaussian mechanism and rollback according to $\frac{k}{N}$. When varying $N$, for $N \geq 2100$, Algorithm 2 reduces sub-optimality by 6–28% and 27–44% relative to the Gaussian mechanism and CDR, respectively. Compared with the mixing algorithm, it remains within 3% when mixing performs better and achieves reductions of up to 13% when it performs better. For $N \leq 1300$, it achieves 13–17% and 34–38% reductions relative to rollback and CDR, respectively. In the normal case, it reduces sub-optimality by 47–73% relative to CDR and remains within 8% of the best-performing baseline. Similar trends are observed when varying $k$ or $\gamma$. Overall, Algorithm 2 maintains low sub-optimality across regimes and adaptively tracks the strongest base mechanism, while the individual baselines exhibit greater performance variability.

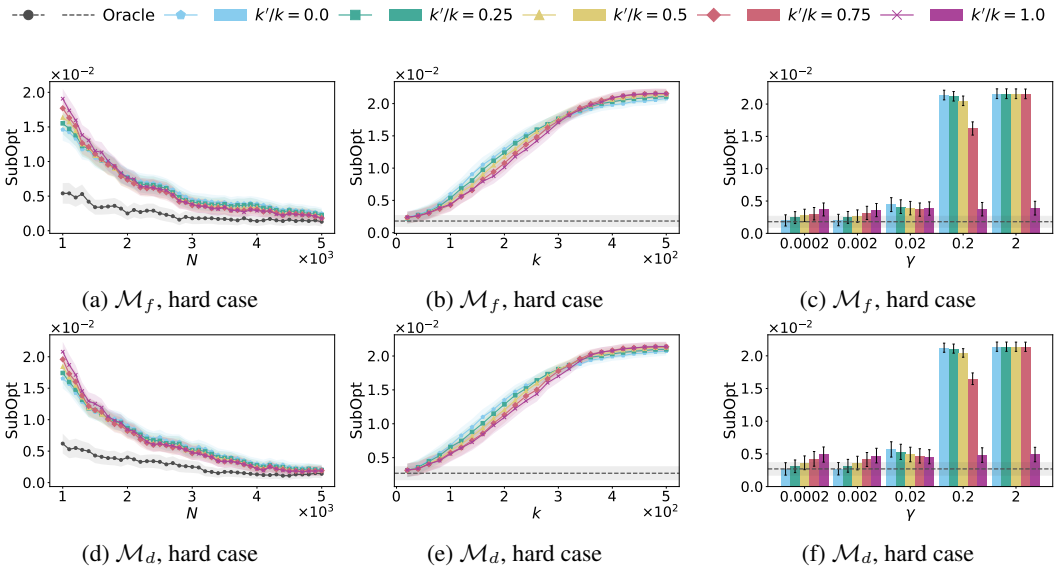
### B.3 Comparison between Algorithm 6 and Algorithm 2

We conduct this comparison on synthetic datasets, since the purpose is to isolate the special distribution-model regime $C^* \in (1, 2)$ under a controlled data-generating process. The setup for $C^* \in (1, 2)$ is identical to that for $C^* \in [2, \infty)$ and only the range of $C^*$ differs. We focus on the hard case, as the sample count of $a^*$ dominates the dataset in this regime. As shown in Table 2, when $C^* = 1.3$, Algorithm 6 achieves consistently competitive performance across different deletion scenarios, yielding more than 93% reduction in sub-optimality compared to Algorithm 2.

Table 2: Comparison of Algorithm 6 and Algorithm 2 under distribution model ($C^* = 1.3$) on synthetic dataset.

| Algorithm | Sub-optimality | | | |
|---|---|---|---|---|
| | $N = 900$ | $N = 1000$ | $N = 1100$ | $N \geq 1200$ |
| Algorithm 6 ($C^* \in (1, 2)$) | 0.0008 | 0.0000 | 0.0000 | 0.0000 |
| Algorithm 2 (single source) | 0.0120 | 0.0115 | 0.0110 | 0.0000 |

### B.4 Experiments of Algorithm 4




(a) $\mathcal{M}_f$, hard case (b) $\mathcal{M}_f$, hard case (c) $\mathcal{M}_f$, hard case

(d) $\mathcal{M}_d$, hard case (e) $\mathcal{M}_d$, hard case (f) $\mathcal{M}_d$, hard case

Figure 3: Comparison of Algorithm 4 with baselines under $\mathcal{M}_f$ and $\mathcal{M}_d$ ($C^* = 5$) on synthetic dataset.

We conduct this experiment on synthetic datasets to isolate the effect of the mixing ratio in Algorithm 4 under a controlled reward model. In the experiments below, we focus exclusively on the hard case where $a^* = a_0$. This choice is motivated by the design of Algorithm 4, in which the mixing parameter is set to $k' = k$ whenever $\max_{a \neq a_0}\{\hat{\mu}'(a) - \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a)}}\} \geq \hat{\mu}'(a_0) + \sqrt{\frac{\ln \frac{m}{\tau}}{2N'(a_0)}}$. In the normal cases ($a^* \neq a_0$), where the deleted arm is sub-optimal, the effect of the mixing ratio is typically less pronounced because the optimal arm is not directly affected by the deletion.

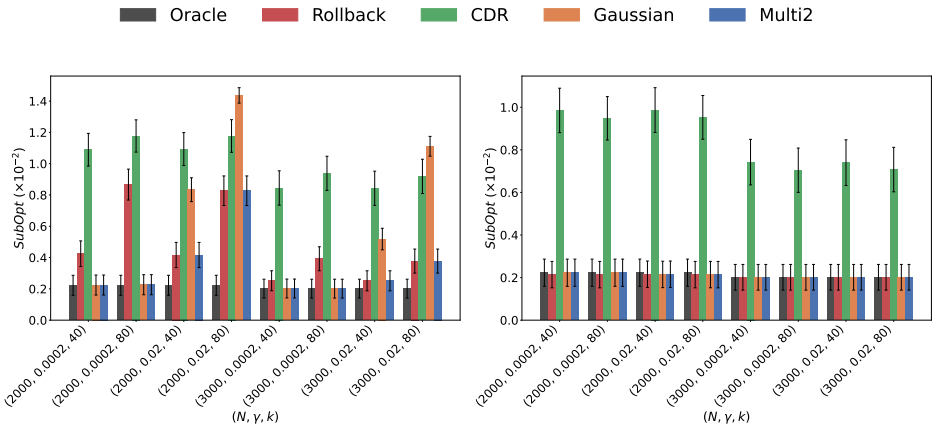
We adopt the same experimental setup as in the synthetic experiments, except for a minor implementation difference in the fixed-sample model. Specifically, each logical block of size $B = 100$ is further partitioned into four equal-sized sub-blocks of size $B/4 = 25$. This construction allows us to implement different mixing ratios $\eta \in \{0, 0.25, 0.5, 0.75, 1\}$, where $\eta = k'/k$ controls the fraction of deleted samples handled by rollback, while the remaining samples are handled by the Gaussian mechanism. In particular, $\eta = 0$ corresponds to the pure Gaussian mechanism, whereas $\eta = 1$ corresponds to pure rollback. Intermediate values of $\eta$ are implemented by assigning the corresponding number of sub-blocks to rollback.

Apart from this block-level construction in the fixed-sample model, all other aspects of the experimental protocol, including data generation, prefix-sharing, deletion strategies, and evaluation metrics, remain the same as in the synthetic experiments described above. The distribution-model experiments use the same configuration as in Appendix B.1. As shown in Figure 3, the best performance is achieved at the two extremes, $\eta = 0$ and $\eta = 1$, in almost all parameter regimes. This validates the design intuition behind Algorithm 2: it is sufficient to adaptively choose between the Gaussian mechanism and rollback, rather than tuning an intermediate mixing ratio.

### B.5 Experiments of Algorithm 7

The experimental setup follows the same protocol as described in Appendix B, except that we focus exclusively on the fixed-sample setting, in accordance with the theoretical analysis of the multi-source unlearning algorithm. For the 5-arm fixed-sample model with ordering $1 > 2 > 3 > 4 > 5$, the hard case corresponds to deleting $k$ samples each from arms 1 and 2, while the normal case corresponds to deleting $k$ samples each from arms 4 and 5. All other aspects of the experimental

protocol—including block construction, prefix-sharing, and evaluation over multiple independent runs—remain identical to those described in Appendix B.1.




(a) $\mathcal{M}_f$, hard case, Yelp dataset

(b) $\mathcal{M}_f$, normal case, Yelp dataset

(c) $\mathcal{M}_f$, hard case, synthetic dataset

(d) $\mathcal{M}_f$, normal case, synthetic dataset

Figure 4: Comparison of Algorithm 7 with baselines under $\mathcal{M}_f$ on Yelp and synthetic datasets.

Figure 4 reports the revised multi-source fixed-sample results over $N \in \{2000, 3000\}$, $\gamma \in \{0.0002, 0.02\}$, and $k \in \{40, 80\}$, using 200 independent datasets. We fix $\delta = 0.1$ and determine $\varepsilon$ from the analytic Gaussian calibration $\Psi_\varepsilon(\gamma) = \delta$ in Equation (1). This gives $\varepsilon \approx 1.2506 \times 10^7$ for $\gamma = 0.0002$ and $\varepsilon \approx 1313.0903$ for $\gamma = 0.02$. The same $(\varepsilon, \delta)$ pair is supplied to CDR in each configuration. The error bars denote 95% confidence intervals computed across the 200 dataset-seed means, after averaging repeated deletion and Gaussian-noise trials within each seed. In the hard case, where deletion requests affect the two best arms, Algorithm 7 adaptively switches between the Gaussian mechanism and rollback. When $\gamma = 0.0002$, it selects the Gaussian branch in every tested configuration. On Yelp, this reduces sub-optimality relative to rollback by 20–74%. On the synthetic instance, it improves over rollback by 3–39% in three of the four configurations; in the remaining configuration $(N, k) = (3000, 40)$, rollback is lower by 3.9%. When $\gamma = 0.02$, the algorithm selects rollback in every configuration, reducing sub-optimality relative to the Gaussian mechanism by 42–66% on Yelp and by 36–69% on the synthetic instance. Thus, the revised algorithm matches the better base mechanism in 15 of the 16 hard-case configurations across the two datasets. It also consistently outperforms CDR, reducing sub-optimality relative to CDR by 65–76% on the synthetic instance and by 30–81% on Yelp. In the normal case, where deletion requests affect the two worst arms, Algorithm 7, Oracle, rollback, and the Gaussian mechanism attain identical sub-optimality on the synthetic instance. On Yelp, Algorithm 7 remains within 4.2% of the best-performing method across all configurations and coincides with the best method in five of the eight configurations. Relative to CDR, it reduces sub-optimality by 68–78% on the synthetic instance and by 71–78% on Yelp.