Batched Nonparametric Bandits via k-Nearest Neighbor UCB

Anonymous Author(s)
Affiliation
Address
email

Abstract

We study sequential decision-making in batched nonparametric contextual bandits, where actions are selected over a finite horizon divided into a small number of batches. Motivated by constraints in domains such as medicine and marketing—where online feedback is limited—we propose a nonparametric algorithm that combines adaptive k-nearest neighbor (k-NN) regression with the upper confidence bound (UCB) principle. Our method, `BaNk-UCB`, is fully nonparametric, adapts to the context dimension, and is simple to implement. Unlike prior work relying on parametric or binning-based estimators, `BaNk-UCB` uses local geometry to estimate rewards and adaptively balances exploration and exploitation. We provide near-optimal regret guarantees under standard Lipschitz smoothness and margin assumptions, using a theoretically motivated batch schedule that balances regret across batches and achieves minimax-optimal rates. Empirical evaluations on synthetic and real-world datasets demonstrate that `BaNk-UCB` consistently outperforms binning-based baselines.

1 Introduction

Many real-world decision-making problems involve using feedback from past interactions to improve future outcomes—a hallmark of adaptive sequential learning. Contextual bandits are a standard framework for modeling these problems, especially in personalized decision-making, where side information helps tailor actions to individuals [Tewari and Murphy, 2017, Li et al., 2010]. In this framework, a learner observes a context, selects an action, and receives a reward, aiming to maximize cumulative reward over time through adaptive policy updates.

However, in many practical applications—such as clinical trials [Kim et al., 2011, Lai et al., 1983] and marketing campaigns [Schwartz et al., 2017, Mao et al., 2018]—adaptivity is limited due to logistical or cost constraints. Decisions are made in batches, and feedback is only received at the end of each batch. This structure permits limited adaptation and renders traditional online bandit algorithms ineffective, motivating new methods tailored for low-adaptivity regimes with few batches.

While parametric bandits have been extended to the batched setting, they often rely on strong modeling assumptions. Nonparametric models offer greater flexibility and robustness [Rigollet and Zeevi, 2010, Qian and Yang, 2016, Reeve et al., 2018, Zhou et al., 2020], but their use in batched bandits remains limited. Existing nonparametric batched bandit methods, such as `BaSEDB` [Jiang and Ma, 2025], rely on partitioning the context space into bins and treating each bin as a local static bandit instance. While effective when contexts are uniformly distributed, such binning-based approaches can struggle in the presence of non-uniform or heterogeneous context distributions. In particular, low-density regions may receive few or no samples, leading to poor reward estimation and imbalanced exploration across the space. These limitations highlight the need for methods that adapt to the local geometry and data distribution, rather than imposing a fixed spatial discretization.

To address this gap, we propose Batched Nonparametric k-nearest neighbor-Upper Confidence Bound

(BaNk-UCB), a nonparametric algorithm for batched contextual bandits that combines adaptive k-nearest neighbor regression with UCB-based exploration. BaNk-UCB adapts neighborhood radii to local data density, eliminating the need for manual bin design. Under Lipschitz continuity and margin conditions, we prove minimax-optimal regret rates up to logarithmic factors. Empirical results on synthetic and real data show consistent improvements over binning-based methods. Our main contributions are:

- We propose BaNk-UCB, a novel nonparametric algorithm for batched contextual bandits that integrates adaptive k-nearest neighbor (k-NN) regression with upper confidence bound (UCB) exploration. The method is simple to implement and avoids biases introduced by coarse partitioning of the context space.
- We design a theoretically grounded batch schedule and establish *minimax-optimal regret bounds* under standard Lipschitz smoothness and margin conditions. This is, to our knowledge, the first such result for a k-NN-based method in the batched setting.
- We highlight how BaNk-UCB automatically adapts to the local geometry of the context distribution without requiring explicit modeling assumption, due to the adaptive neighborhood choice in k-NN regression.
- We demonstrate through extensive experiments on both synthetic and real-world datasets that BaNk-UCB consistently outperforms binning-based baselines, particularly in high-dimensional or heterogeneous contexts.

1.1 Related Work

Batched contextual bandits have received growing attention due to their relevance in settings with limited adaptivity, such as clinical trials and campaign-based interventions [Perchet et al., 2016, Gao et al., 2019]. Prior work has explored both non-contextual bandits with fixed or adaptive batch schedules [Esfandiari et al., 2021, Kalkanli and Ozgur, 2021, Jin et al., 2021], and contextual bandits, often under parametric assumptions. In particular, linear [Han et al., 2020] and generalized linear models [Ren et al., 2022] have been popular due to their analytical tractability, though such models may fail to generalize when the reward function is nonlinear or misspecified.

Nonparametric bandits have been extensively studied in the fully sequential setting. Early work by Yang and Zhu [2002] employed ϵ-greedy strategies with nonparametric reward estimation. Subsequent methods include the Adaptively Binned Successive Elimination (ABSE) algorithm [Rigollet and Zeevi, 2010, Perchet and Rigollet, 2013], which partitions the context space adaptively and uses elimination-based strategies [Even-Dar et al., 2006]. Other approaches include kernel regression methods [Qian and Yang, 2016, Hu et al., 2020], nearest neighbor algorithms [Reeve et al., 2018, Zhao et al., 2024], and Gaussian process or kernelized models [Krause and Ong, 2011, Valko et al., 2013, Arya and Sriperumbudur, 2023].

In the batched nonparametric setting, Jiang and Ma [2025] introduced BaSEDB, a batched variant of ABSE with dynamic binning and minimax-optimal regret guarantees. Other recent directions include neural network-based estimators [Gu et al., 2024], Lipschitz-constrained models [Feng et al., 2022], and semi-parametric frameworks [Arya and Song, 2025], though each makes different structural assumptions.

Our work departs from these approaches by employing adaptive k-nearest neighbor regression to estimate both reward functions and confidence bounds under batch constraints. Unlike binning-based methods, BaNk-UCB avoids discretization and instead adapts to the local geometry of the context distribution through data-driven neighborhood selection. To our knowledge, this is the first batched nonparametric algorithm based on k-NN to achieve near-optimal regret guarantees. Empirically, we show that BaNk-UCB outperforms BaSEDB, particularly in heterogeneous context spaces, leveraging the well-known ability of k-NN to adapt to local intrinsic dimension [Kpotufe, 2011].

2 Setup

We consider a batched contextual bandit problem over a finite time horizon T, where decisions are grouped into M batches to reflect limited adaptivity. At each round $t \in \{1, \dots, T\}$, a context $X_t \in \mathcal{X} \subset \mathbb{R}^d$ is observed, and the learner selects an action $a_t \in \mathcal{A} = \{1, \dots, K\}$. The learner selects an action $a_t \in \mathcal{A}$ based on X_t and receives a noisy reward:

$$Y_t = f_{a_t}(X_t) + \epsilon_t, \tag{1}$$

where $f_a(x)$ is an unknown mean reward function for $a \in \mathcal{A}$ and $x \in \mathcal{X}$. The model noise is given by ϵ_t. We make the following assumptions on the noise and context space.

Assumption 1 (Sub-Gaussian noise)**.** *We assume that the noise terms $\{\epsilon_t\}_{t=1}^T$ are independent and σ^2-sub-Gaussian; that is, for all $\lambda \in \mathbb{R}$ and all t,*

$$\mathbb{E}\left[e^{\lambda \epsilon_t}\right] \leq e^{\frac{1}{2}\lambda^2\sigma^2}. \tag{2}$$

Assumption 2 (Bounded context density)**.** *The context vectors X_t are drawn i.i.d. from a distribution with density p_X, which is supported on $\mathcal{X} \subset \mathbb{R}^d$. We assume that $p_X(x) \geq \underline{c}$ for some $\underline{c} > 0$.*

Note that, while many nonparametric bandit works assume the context space to be a cube such as $[0,1]^d$, we allow for arbitrary bounded domains with densities bounded away from zero—a setting that accommodates more general geometry in $\mathcal{X}$.

A *policy* $\pi_t : \mathcal{X} \to \mathcal{A}$ for $t = 1, \ldots, T$ determines an action $a_t \in \mathcal{A}$ at t. Based on the chosen action a_t, a reward Y_t is obtained. In the sequential setting without batch constraints, the policy π_t can depend on all the observations (X_s, Y_s) for $s < t$. In contrast, in a batched setting with M batches, where $0 = t_0 < t_1 < \cdots < t_{M-1} < t_M = T$, for $t \in [t_i, t_{i+1})$, the policy π_t can depend on observations from the previous batches, but not on any observations within the same batch. In other words, policy updates can occur only at the predetermined batch boundaries $t_1, \ldots, t_M$. This reflects the constraint that feedback is only revealed at the end of each batch.

Let $\mathcal{G} = \{t_0, t_1, \ldots, t_M\}$ represent a partition of time $\{0, 1, \ldots, T\}$ into M intervals, and $\pi = (\pi_t)_{t=1}^T$ be the sequence of policies applied at each time step. The overarching objective of the decision-maker is to devise an M-batch policy $(\mathcal{G}, \pi)$ that minimizes the expected *cumulative regret*, defined as $\mathcal{R}_T(\pi) = E[R_T(\pi)]$, where

$$R_T(\pi) = \sum_{t=1}^{T} f_*(X_t) - f_{(\pi_t(X_t))}(X_t) \tag{3}$$

where $f_*(x) = \max_{a\in\mathcal{A}} f_a(x)$ is the expected reward from the optimal choice of arms given a context x. The cumulative regret serves as a pivotal metric, quantifying the difference between the cumulative reward attained by π and that achieved by an optimal policy, assuming perfect foreknowledge of the optimal action at each time step.

We make the following assumptions on the reward functions.

Assumption 3 (Lipschitz Smoothness)**.** *We assume that the link function $f_a : \mathbb{R}^d \to \mathbb{R}$ for each arm is Lipschitz smooth, that is, there exists $L > 0$ such that for $a \in \mathcal{A}$,*

$$|f_a(x) - f_a(x')| \leq L\|x - x'\|,$$

holds for $x, x' \in \mathcal{X}$.

Assumption 4 (Margin)**.** *For some $0 < \alpha \leq d$ and for all $a \in \mathcal{A}$, there exists a $\delta_0 \in (0,1)$ and $D_\alpha > 0$ such that*

$$\mathbb{P}_X(0 < f_*(X) - f_a(X) \leq \delta) \leq D_\alpha \delta^\alpha,$$

holds for all $\delta \in [0, \delta_0]$.

The margin condition implies that the regions where the reward gap is small, i.e., where it is hard to distinguish the best arm are not too large. The exponent α controls the rate at which the measure of such regions shrinks as $\delta \to 0$. When α is small, suboptimal arms can be frequently indistinguishable from the best arm, leading to slower learning; larger α implies faster decay and enables faster convergence.

Remark 1. *Throughout this paper, we assume that $\alpha \leq 1$, because in the $\alpha > 1$ regime, the context information becomes irrelevant as one arm dominates the other (e.g., see Proposition 2.1 of Rigollet and Zeevi [2010]).*

The margin condition plays a crucial role in determining the minimax rate of regret in nonparametric bandit problems, similar to its role in classification [Mammen and Tsybakov, 1999, Tsybakov, 2004].

Notation: We use $\|\cdot\|$ to denote the Euclidean norm in $\mathbb{R}^d$. We denote $B(x, r)$ to denote a Euclidean ball with center $x \in \mathbb{R}^d$ and radius r. We denote $\lesssim$ and $\gtrsim$ to denote inequalities upto constants. The notation $f(n) = \Theta(g(n))$ indicates an asymptotic tight bound. Formally, there exist positive constants c_1, c_2 and n_0 such that for all $n \geq n_0$, $c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n)$. The notation $\tilde{O}(g(n))$ denotes an asymptotic upper bound up to logarithmic factors. For $a, b \in \mathbb{R}$, $a \vee b$ denotes the maximum of a and b, and $a \wedge b$ denotes minimum of a and b. For any batch m, let $\mathcal{F}_{t_m}$ be the filtration encoding the history up to batch m.

3 Batched Nonparametric k-Nearest Neighbor-UCB (`BaNk-UCB`) Algorithm

Recall that in the batched bandits setting, the decision at time t in batch m only depends on the information observed up to the end of the $(m-1)^{\text{th}}$ batch. We propose `BaNk-UCB` (**B**atched **N**onparametric ***k***-Nearest Neighbors **U**pper **C**onfidence **B**ound) detailed in Algorithm 1. This is based on an *adaptive k-nearest-neighbor* policy that tunes k according to the local margin (sub-optimality gap) and context density. Let us first define some useful notation. For $x \in \mathcal{X}$ and some fixed $k \leq t_{m-1}$, let $N_{t_{m-1},k}(x,a)$ be the set of k nearest neighbors of x where arm a was chosen, i.e.,

$$N_{t_{m-1},k}(x,a) := \{s \leq t_{m-1} : a_s = a \text{ and } X_s \text{ is among the } k \text{ nearest to } x\}. \tag{4}$$

For simplicity, we denote $N_{t,k}(x,a) \equiv N_{t_{m-1},k}(x,a)$ for all times t within the batch interval $(t_{m-1}, t_m]$. Then we define for $t \in (t_{m-1}, t_m]$,

$$d_{a,t,k}(x) = \max_{s \in N_{t_{m-1},k}(x,a)} \|X_s - x\|, \tag{5}$$

to be the radius of the k-NN ball around x for arm a. We adaptively select the number of neighbors, denoted $k_{t,a}(x)$, based solely on observations available up to the end of batch $(m-1)$ and specifically associated with arm a. This $k_{t,a}$ is then used in the proposed `BaNk-UCB` algorithm as described in Algorithm 1:

$$k_{t,a}(x) = \max\left\{ j \mid L d_{a,t,j}(x) \leq \frac{\ln t_{m-1}}{j} \right\}. \tag{6}$$

Note that L is the constant from the Lipschitz smoothness assumption (Assumption 3). The left hand side thus controls the bias in the estimation of f_a and the right-hand side controls the variance in the estimation, i.e., it ensures that we use large k if previous samples are relatively dense around X_t, and vice versa. The adaptive selection of k in (6) requires that the nearest observed context be sufficiently close. Specifically, we enforce $L d_{a,t,1}(X_t) \leq \sqrt{\ln t_{m-1}}$; otherwise, reliable estimation is not feasible, and we conservatively set the UCB to infinity: $\hat{f}_{a,t}(x) = \infty$. Otherwise, for $t \in (t_{m-1}, t_m]$, we calculate the upper confidence bound (UCB) as follows:

$$\hat{f}_{a,t}(x) = \frac{1}{k_{a,t}(x)} \sum_{s \in N_{t_{m-1}}(x,a)} Y_s + \xi_{a,t}(x) + L d_{a,t}(x), \tag{7}$$

where $d_{a,t}$ is as defined in (5) and $\xi_{a,t}$ is defined as:

$$\xi_{a,t}(x) = \sqrt{\frac{2\sigma^2}{k_{a,t}(x)} \ln\left(d t_{m-1}^{2d+3} |\mathcal{A}|\right)}. \tag{8}$$

Algorithm 1 BaNk-UCB for Batched Nonparametric Bandits

Input: Partition $t_0, t_1, \ldots, t_M$, with $t_0 = 0$ and $t_M = T$.
for $m = 1, \ldots, M$ **do**
    for $t = t_{m-1} + 1, \ldots, t_m$ **do**
        Receive context X_t;
        for $a \in \mathcal{A}$ **do**
            if $L d_{a,t,1}(X_t) > \sqrt{\ln t_{m-1}}$ **then**
                Set $\hat{f}_{a,t}(X_t) \leftarrow +\infty$;
            else
                Compute $k_{t,a}(X_t)$ according to (6);
                Compute $\hat{f}_{a,t}(X_t)$ according to (7);
            end if
        end for
        Choose action $a_t = \arg\max_{a \in \mathcal{A}} \hat{f}_{a,t}(X_t)$;
        Pull arm a_t;
    end for
    Observe rewards $\{Y_t, t \in t_{m-1} + 1, \ldots, t_m\}$;
end for

Here, $\xi_{a,t}(x)$ provides a high-probability bound for stochastic noise of the nearest-neighbor averaging, while $Ld_{a,t}(x)$ controls the estimation bias from finite-sample approximation. Both terms depend explicitly on prior-batch data, highlighting the critical role batch design plays in balancing estimation accuracy and cumulative regret. Finally, the algorithm selects arm a_t with the maximum UCB value,

$$a_t = \text{arg max}_{a \in \mathcal{A}} \hat{f}_{a,t}(X_t). \tag{9}$$

Note, that for (6) to hold in the initial samples, we use $\log(T)/|\mathcal{A}|$ samples for pure exploration in the beginning.

Remark 2. *The adaptive choice of $k_{a,t}(x)$ in* (6) *simultaneously balances the bias-variance and exploration-exploitation trade-offs in estimating f_a. Specifically, the bias-variance trade-off is managed by selecting a larger k when previously observed contexts are densely sampled around X_t, thereby reducing variance, and choosing a smaller k otherwise, controlling bias. Moreover, due to the Lipschitz smoothness assumption, contexts with larger optimality gaps ($f^*(x) - f_a(x)$) naturally correspond to larger radii $d_{a,t,j}(x)$, leading to smaller chosen values of k and promoting targeted exploration in regions with high uncertainty.*

4 Minimax Analysis on the Expected Regret

In this section, we demonstrate that the BaNk-UCB algorithm achieves a minimax optimal rate on the expected cumulative regret under an appropriately designed partition of grid points. Specifically, the rate matches known minimax lower bounds up to logarithmic factors. First we describe the choice of the batch grid points and then state the upper and lower bounds on the expected regret.

4.1 Batch sizes

The choice of batch sizes plays a crucial role in the performance of the batched bandit algorithms. We partition the time horizon into M batches, denoted by grid points $\mathcal{G} = \{t_1, t_2, \ldots, t_M\}$, with $t_0 = 0$. The special case $M = T$ recovers the fully sequential bandit setting, where policy updates occur at every step. Conversely, smaller M imposes fewer policy updates, introducing a trade-off between computational/operational complexity and regret accumulation. A key challenge in the batched setting is selecting the grid $\mathcal{G}$. Intuitively, to minimize total regret, no single batch should dominate the cumulative error, suggesting that the grid should balance regret across batches. If one batch incurs higher regret, reassigning time steps can improve the overall rate. This motivates a grid choice that equalizes regret across batches, up to order in T and d, as we formalize below. We choose:

$$t_1 = ad, \;\; t_m = \lfloor a t_{m-1}^{\gamma} \rfloor, \tag{10}$$

where $\gamma = \frac{1+\alpha}{2+d}$ and $a = \Theta(T^{\frac{1-\gamma}{1-\gamma^M}})$ is chosen so that $t_M = T$.

4.2 Regret bounds

In order to establish the regret rates, we first define the batch-wise expected sample density, motivated by the formulation of Zhao et al. [2024]. Let $p_a^{(m)} : \mathcal{X} \to \mathbb{R}$ is defined such that for all $A \subseteq \mathcal{X}$,

$$\mathbb{E}\left[\sum_{t=t_{m-1}}^{t_m} 1(X_t \in A, a_t = a)\right] = \int_A p_a^{(m)}(x)dx. \tag{11}$$

First let's consider the cumulative regret relate it to the batch-wise expected sample density.

Lemma 1. *The expected cumulative regret in* (3) *is given by $R_T(\pi) = \sum_{a \in \mathcal{A}} \sum_{m=1}^{M} R_a^{(m)}(\pi)$, where $R_a^{(m)}(\pi)$ is defined as:*

$$R_a^{(m)}(\pi) = \int_{\mathcal{X}} (f_*(x) - f_a(x)) p_a^{(m)}(x)dx. \tag{12}$$

Proof. Consider,

$$\begin{aligned}
R_T(\pi) &= \mathbb{E}\left[\sum_{t=1}^{T}(f_*(X_t) - f_{a_t}(X_t))\right] \\
&= \mathbb{E}\left[\sum_{m=1}^{M}\sum_{t=t_{m-1}}^{t_m}(f_*(X_t) - f_{a_t}(X_t))\right] \\
&= \sum_{a\in\mathcal{A}}\sum_{m=1}^{M}\mathbb{E}\left[\sum_{t=t_{m-1}}^{t_m}(f_*(X_t) - f_{a_t}(X_t))1(a_t = a)\right] \\
&= \sum_{a\in\mathcal{A}}\sum_{m=1}^{M}\int_{\mathcal{X}}(f_*(X_t) - f_{a_t}(X_t))\, p_a^{(m)}(x)dx.
\end{aligned}$$

□

Using the fact that the batch sizes are chosen to control for the regret to be balanced across batches, the idea is to construct an upper bound on the batch-wise arm specific regret, $R_a^{(m)}(\pi)$. Then, using Lemma 1, we can bound the expected cumulative regret.

Theorem 1. *Under Assumptions 1–4, and with the batch sizes as defined in* (10) *in Section 4.1, the regret of the proposed BaNk-UCB algorithm (π) is bounded by,*

$$R_T(\pi) \lesssim |\mathcal{A}|MT^{\frac{1-\gamma}{1-\gamma^M}}\,(\ln T)^{\gamma}\,, \tag{13}$$

where $\gamma = \frac{1+\alpha}{2+d}$.

Proof Sketch for Theorem 1. For $\epsilon > 0$, we split $R_a^{(m)}$ into two terms:

$$\begin{aligned}
R_a^{(m)} = &\int_{\mathcal{X}}(f_*(x) - f_a(x))p_a^{(m)}(x)1(f_*(x) - f_a(x) > \epsilon)dx \\
&+ \int_{\mathcal{X}}(f_*(x) - f_a(x))p_a^{(m)}(x)1(f_*(x) - f_a(x) \le \epsilon)dx.
\end{aligned} \tag{14}$$

The idea is to bound these two terms separately, where the second one can be bounded using the margin assumption (i.e., Assumption 4). The ϵ is determined theoretically based on the bound on $R_a^{(m)}$. From Lemmas 8 and 10 in the Appendix B, we get that:

$$\int_{\mathcal{X}}(f^*(x) - f_a(x))p_a^{(m)}(x)1\left(f^*(x) - f_a(x) > \epsilon\right)dx \lesssim \epsilon^{\alpha-d-1}\ln t_{m-1} + t_m\epsilon^{1+\alpha}. \tag{15}$$

Furthermore, we can bound the second term in (14) by

$$\begin{aligned}
&\int_{\mathcal{X}}(f^*(x) - f_a(x))p_a^{(m)}(x)1\left(f^*(x) - f_a(x) \le \epsilon\right)dx \\
&\qquad\overset{(\dagger)}{\le} t_m\epsilon\int p_X(x)1\left(f^*(x) - f_a(x) \le \epsilon\right)dx \\
&\qquad\overset{(\ddagger)}{\lesssim} t_m\epsilon^{1+\alpha},
\end{aligned} \tag{16}$$

where (†) follows from Lemma 2 and (‡) follows from the Margin condition. Now combining (15) and (16), we get from (14):

$$R_a^{(m)} \lesssim \epsilon^{\alpha-d-1}\ln t_{m-1} + t_m\epsilon^{1+\alpha} \tag{17}$$

By the choice of our batch end points $t_m = \lfloor a t_{m-1}^{\gamma}\rfloor$, then it is easy to see using a geometric sum in the exponent, $t_m = \Theta(T^{\frac{1-\gamma^m}{1-\gamma^M}})$ with $\gamma = \frac{1+\alpha}{2+d}$. Now, balancing the two terms in (17) and solving for ϵ, we get $\epsilon = [t_{m-1}^{-1}\ln t_{m-1}]^{\frac{1}{2+d}}$. Therefore, we have:

$$R_a^{(m)} \lesssim t_m[t_{m-1}^{-1}\ln t_{m-1}]^{\frac{1+\alpha}{2+d}} \lesssim T^{\frac{1-\gamma^m}{1-\gamma^M}} \cdot T^{-\left(\frac{1-\gamma^{m-1}}{1-\gamma^M}\right)\left(\frac{1+\alpha}{2+d}\right)} \cdot (\ln t_{m-1})^{\frac{1+\alpha}{2+d}} = T^{\frac{1-\gamma}{1-\gamma^M}}\,(\ln t_{m-1})^{\gamma}\,. \tag{18}$$

Now, using Lemma 1,

$$
\begin{aligned}
R_T(\pi) &= \sum_{a\in\mathcal{A}}\sum_{m=1}^{M} R_a^{(m)}(\pi) \\
&\lesssim \sum_{a\in\mathcal{A}}\sum_{m=1}^{M} T^{\frac{1-\gamma}{1-\gamma^M}} \left(\ln t_{m-1}\right)^{\gamma} \\
&\lesssim |\mathcal{A}| M T^{\frac{1-\gamma}{1-\gamma^M}} \left(\ln T\right)^{\gamma}.
\end{aligned}
$$

□

Next, we establish minimax lower bounds on the regret achievable by any M-batch policy $(\mathcal{G}, \pi)$ and show that it matches the upper bound in Theorem 1 up to logarithm factors. While our lower bound result matches that of Jiang and Ma [2025], we include a complete proof in the Appendix C for completeness. Notably, our hypothesis construction and proof technique differ slightly from theirs.

Theorem 2 (Minimax lower bound for nonparametric batched bandits)**.** *Let $\mathcal{F}(L, \alpha)$ denote the class of functions that satisfy Lipschitz smoothness (Assumption 3) with Lipschitz constant L and margin condition (Assumption 4). For any M-batch policy π deployed over T rounds, the minimax expected cumulative regret satisfies:*

$$
\inf_{\pi} \sup_{f_1, f_2 \in \mathcal{F}(L,\alpha)} R_T(\pi) \gtrsim T^{\frac{1-\gamma}{1-\gamma^M}}, \quad \textit{where } \gamma = \frac{\alpha+1}{2+d}.
$$

Theorem 2 characterizes the fundamental difficulty of learning within this class of problems and shows that our BaNk-UCB algorithm's upper bound matches this minimax lower bound up to logarithmic factors. Recall that,

$$
R_T(\pi) = \mathbb{E}[\sum_{t=1}^{T} (f^*(X_t) - f_{a_t}(X_t)]. \tag{19}
$$

We define the inferior sampling rate as the expected number of steps with sub-optimal actions:

$$
S_T(\pi) = \mathbb{E}[\sum_{t=1}^{T} 1(f_{a_t}(X_t) < f_*(X_t))] \tag{20}
$$

Lemma 11 characterizes the relationship between S and R and we use that in establishing a lower-bound on the batch-wise regret for any policy π in batched bandit setting.

Remark 3. *Note that, when $M \gtrsim \ln(\ln T)$ and the number of arms $|\mathcal{A}| \lesssim \ln T$, the cumulative regret simplifies to $R_T(\pi) = \tilde{O}(T^{1-\gamma})$, recovering the known minimax optimal rate for fully sequential (non-batched) nonparametric bandits [Perchet and Rigollet, 2013]. This condition implies that, surprisingly, only a relatively modest increase in the number of batches (log-logarithmic in the horizon T) is sufficient to achieve the fully sequential optimal rate. Additionally, the mild logarithmic restriction on the number of actions $|\mathcal{A}|$ reflects practical scenarios where the action set is moderately large but not excessively growing with T, highlighting the efficiency of the BaNk-UCB algorithm in nearly matching fully adaptive performance despite batching constraints.*

5 Experiments

In this section, we present numerical simulations and real-data experiments to illustrate the performance of the proposed Batched Nonparametric k-NN UCB algorithm (BaNk-UCB) in comparison to the nonparametric analogue: Batched Successive Elimination with Dynamic Binning (BaSEDB) algorithm of Jiang and Ma [2025].

5.1 Simulated Data

We consider the following simulation settings:

Setting 1: Motivated by the construction of the function class for the regret lower bound, we make the following choices for f_1 and f_2: $f_1(x) = \sum_{j=1}^{D} v_j h I\{x \in \mathcal{B}_j\}, x \in \mathcal{X}$, and $f_2(x) = 0$, where $v_j \in \{-1, 1\}$ for $j = 1, \ldots, D$, $\mathcal{B}_j$ is a ball centered at c_j with radius r. In Figure 1, we set $\mathcal{X} = [-1, 1]^d$ (with a uniform P_X) with $d = 2, r = 0.6, D = 6$, with randomly generated centers for $\mathcal{B}_j$ and Rademacher random variables $v_j, j = 1, \ldots, 6$. Note that, Setting 1 is derived from the regret lower bound construction and represents a worst-case instance for nonparametric bandits under margin conditions.

Setting 2: As illustrated in Figure 1 consider the following choice of mean reward functions: $f_1(x) = \|x\|_2$ and $f_2(x) = 0.5 - \|x\|_2$, where X is sampled uniformly from $[-1, 1]^d$, with $d = 2$. We set $T = 10000$, $L = 1$ for the Lipschitz constant in Assumption 3. We fix the number of batches

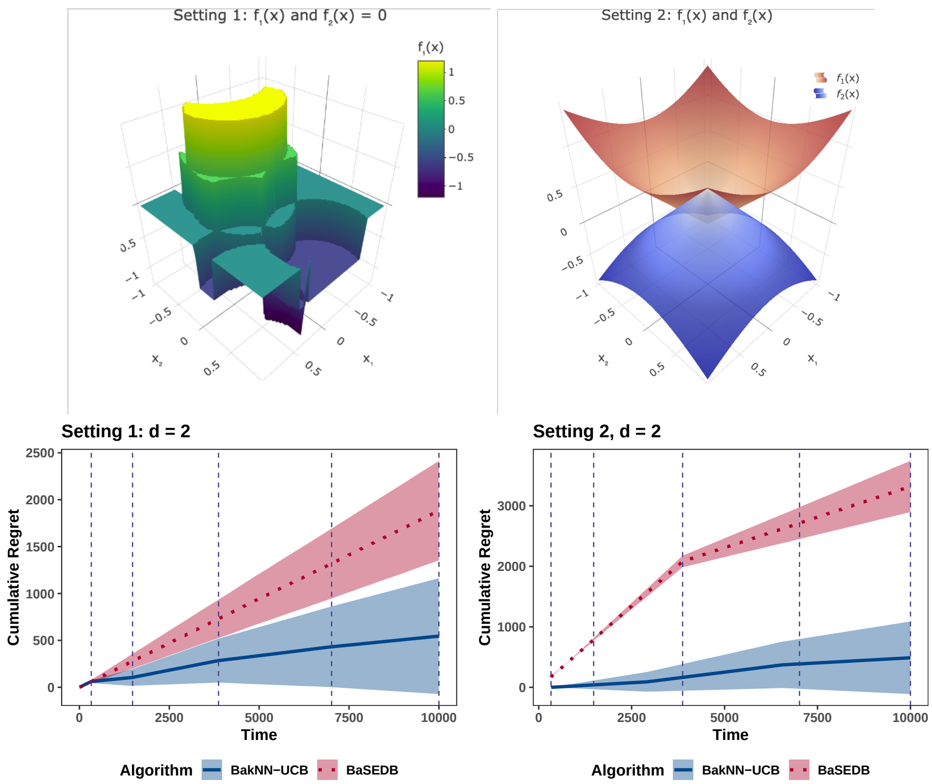

Figure 1: Top row (left to right): Reward functions for the two arms in Setting 1 and 2, respectively. Bottom row: Cumulative regret comparison for BaSEDB and BaNk-UCB algorithms over 30 runs.

to $M = 5$ to balance between frequent updates and computational efficiency, but the results remain consistent across different choices of M. For the BaSEDB algorithm, we follow the specifications described in Jiang and Ma [2025] for choosing grid points and bin-widths. For our proposed BaNk-UCB algorithm, we choose the same batch grid for a fair comparison. In Figure 1, we plot the cumulative regret averaged over 30 independent runs. In order to present an empirical assessment of the variability inherent in our simulations, the shaded regions represent empirical confidence intervals computed as ± 1.96 times the standard error across these runs. The vertical dotted blue lines denote the grid choices for the batches.

`BaNk-UCB` consistently outperforms `BaSEDB` across all experimental settings. Although our batch sizes were selected based on empirical performance, they align closely with the theoretically motivated schedule in Section 4.1. Importantly, we find that performance is robust to the specific number of batches, as long as batch endpoints follow the prescribed growth pattern. This suggests that `BaNk-UCB` does not require precise tuning of the batch schedule to perform well.

In Appendix C.1, we extend the comparison to higher-dimensional contexts ($d = 3, 4, 5$), where both methods degrade in performance, yet `BaNk-UCB` maintains a consistent advantage over `BaSEDB`. A key practical benefit of `BaNk-UCB` is its minimal tuning overhead. Unlike binning-based algorithms such as `BaSEDB`, which depend on careful calibration of bin widths, refinement rates, and arm elimination thresholds—often requiring knowledge of problem-specific parameters—`BaNk-UCB` relies on a fully data-driven nearest neighbor strategy. Its adaptively chosen k automatically balances bias and variance based on local data density, without needing explicit smoothness or margin parameters. This makes `BaNk-UCB` both more robust to misspecification and easier to implement in practice.

5.2 Real Data

We evaluate the performance of BaNk-UCB and BaSEDB algorithm on three publicly available classification datasets: (a) *Rice* [Cammeo and Osmancik, 2020], consisting of 3810 samples with 7 morphological features used to classify two rice varieties; (b) *Occupancy Detection* [Candanedo and Feldheim, 2016], with 8143 samples and 5 environmental sensor features used to predict room occupancy; and (c) *EEG Eye State* [Biermann, 2014], with 14980 samples and 14 EEG measurements used to classify eye state. In all cases, we treat the true label as the optimal action and assign a binary reward of 1 if the selected action matches the label, and 0 otherwise. We simulate a contextual bandit setting where the context x_t is observed, the learner selects an arm $a_t \in \{1, \dots, K\}$, and observes only the reward for the chosen arm. We set the number of arms K equal to the number of classes (which is $K = 2$ for the three datasets considered) and choose the number of batches to be 3, 4, and 6 respectively, based on dataset size. The number of batches was selected based on the total number of samples to ensure reasonable granularity while maintaining batch sizes that approximately align with our theoretically motivated geometric schedule.

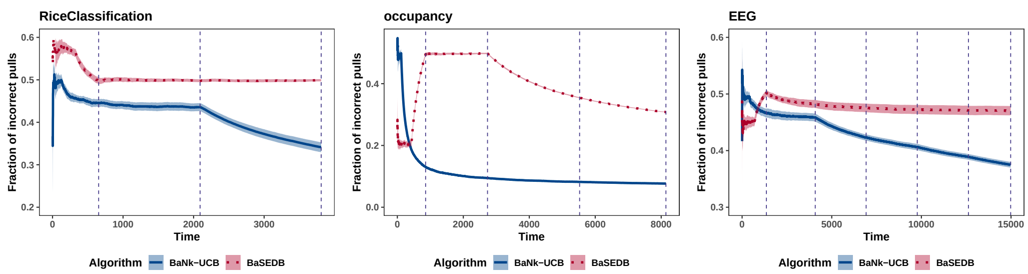

Figure 2: Rolling average fraction of incorrect decisions across three real datasets. BaNk-UCB achieves lower error and faster learning than BaSEDB.

The rolling fraction of incorrect decisions is computed using a windowed average over 30 independent random permutations of each dataset. In Figure 2, we plot the rolling fraction of incorrect decisions with shaded regions (± 1.96 standard errors) for uncertainty quantification as a function of the number of observed instances. `BaNk-UCB` consistently outperforms `BaSEDB` across all datasets. For the EEG dataset, which has the highest context dimensionality, `BaNk-UCB` exhibits faster convergence and consistently lower error, suggesting its advantage in capturing local structure in high-dimensional spaces. Batch sizes are chosen according to theoretical guidelines and are identical for both algorithms.

6 Conclusion

We introduced `BaNk-UCB`, a nonparametric algorithm for batched contextual bandits that combines adaptive k-nearest neighbor regression with the UCB principle. Unlike binning-based methods, `BaNk-UCB` leverages the local geometry of the context space and naturally adapts to heterogeneous data distributions. We established near-optimal regret guarantees under standard Lipschitz smoothness and margin conditions and proposed a theoretically grounded batch grid that balances regret across batches. In addition to its theoretical robustness, `BaNk-UCB` is resilient to batch scheduling choices and requires minimal parameter tuning, making it suitable for practical deployment in real-world systems. Empirical evaluations on both synthetic and real-world classification datasets demonstrate that `BaNk-UCB` consistently outperforms existing nonparametric baselines, particularly in high-dimensional or irregular context spaces.

While `BaNk-UCB` achieves minimax-optimal regret under standard conditions, it assumes a known Lipschitz constant, which influences the adaptive selection of neighborhood size in k-NN estimation. The algorithm also relies on batch schedules guided by theoretical principles, which may not always align with real-time operational constraints. Moreover, although k-NN performs well in moderate dimensions, its accuracy may deteriorate in very high-dimensional settings due to the curse of dimensionality. Addressing these limitations by developing adaptive strategies for estimating smoothness and margin parameters, or by integrating dimension reduction techniques, is a promising direction for future research. Additional extensions include eliminating extraneous logarithmic factors in regret bounds and generalizing the framework to infinite or structured action spaces.

References

Sakshi Arya and Hyebin Song. Semi-parametric batched global multi-armed bandits with covariates. *arXiv preprint arXiv:2503.00565*, 2025.

Sakshi Arya and Bharath K Sriperumbudur. Kernel ϵ-greedy for contextual bandits. *arXiv preprint arXiv:2306.17329*, 2023.

H. Biermann. Eeg eye state dataset. `https://archive.ics.uci.edu/ml/datasets/EEG+Eye+State`, 2014.

G. Cammeo and T. Osmancik. Rice (cammeo and osmancik). `https://archive.ics.uci.edu/ml/datasets/Rice+(Cammeo+and+Osmancik)`, 2020.

Luis M. Candanedo and Véronique Feldheim. Occupancy detection data set. `https://archive.ics.uci.edu/dataset/357/occupancy+detection`, 2016.

Hossein Esfandiari, Amin Karbasi, Abbas Mehrabian, and Vahab Mirrokni. Regret bounds for batched bandits. *Proceedings of the AAAI Conference on Artificial Intelligence*, 35(8):7340–7348, May 2021.

Eyal Even-Dar, Shie Mannor, Yishay Mansour, and Sridhar Mahadevan. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research*, 7(6), 2006.

Yasong Feng, Zengfeng Huang, and Tianyu Wang. Lipschitz bandits with batched feedback. In Alice H. Oh, Alekh Agarwal, Danielle Belgrave, and Kyunghyun Cho, editors, *Advances in Neural Information Processing Systems*, 2022.

Zijun Gao, Yanjun Han, Zhimei Ren, and Zhengqing Zhou. Batched multi-armed bandits problem. *Advances in Neural Information Processing Systems*, 32, 2019.

Quanquan Gu, Amin Karbasi, Khashayar Khosravi, Vahab Mirrokni, and Dongruo Zhou. Batched neural bandits. *ACM / IMS J. Data Sci.*, 1(1), January 2024.

Yanjun Han, Zhengqing Zhou, Zhengyuan Zhou, Jose Blanchet, Peter W Glynn, and Yinyu Ye. Sequential batch learning in finite-action linear contextual bandits. *arXiv preprint arXiv:2004.06321*, 2020.

Yichun Hu, Nathan Kallus, and Xiaojie Mao. Smooth contextual bandits: Bridging the parametric and non-differentiable regret regimes. In *Conference on Learning Theory*, pages 2007–2010. PMLR, 2020.

Rong Jiang and Cong Ma. Batched nonparametric contextual bandits. *IEEE Transactions on Information Theory*, 2025.

Tianyuan Jin, Jing Tang, Pan Xu, Keke Huang, Xiaokui Xiao, and Quanquan Gu. Almost optimal anytime algorithm for batched multi-armed bandits. In *International Conference on Machine Learning*, pages 5065–5073. PMLR, 2021.

Cem Kalkanli and Ayfer Ozgur. Batched thompson sampling. In *Advances in Neural Information Processing Systems*, volume 34, pages 29984–29994. Curran Associates, Inc., 2021.

Edward S Kim, Roy S Herbst, Ignacio I Wistuba, J Jack Lee, George R Blumenschein Jr, Anne Tsao, David J Stewart, Marshall E Hicks, Jeremy Erasmus Jr, Sanjay Gupta, et al. The battle trial: personalizing therapy for lung cancer. *Cancer discovery*, 1(1):44–53, 2011.

Samory Kpotufe. k-nn regression adapts to local intrinsic dimension. *Advances in neural information processing systems*, 24, 2011.

Andreas Krause and Cheng Ong. Contextual gaussian process bandit optimization. *Advances in neural information processing systems*, 24, 2011.

Tze Leung Lai, Herbert Robbins, and David Siegmund. Sequential design of comparative clinical trials. In *Recent Advances in Statistics*, pages 51–68. Elsevier, 1983.

Lihong Li, Wei Chu, John Langford, and Robert E Schapire. A contextual-bandit approach to personalized news article recommendation. In *Proceedings of the 19th international conference on World wide web*, pages 661–670, 2010.

Enno Mammen and Alexandre B Tsybakov. Smooth discrimination analysis. *The Annals of Statistics*, 27(6):1808–1829, 1999. doi: 10.1214/aos/1017939240.

Yizhi Mao, Miao Chen, Abhinav Wagle, Junwei Pan, Michael Natkovich, and Don Matheson. A batched multi-armed bandit approach to news headline testing. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 1966–1973. IEEE, 2018.

Vianney Perchet and Philippe Rigollet. The multi-armed bandit problem with covariates. *The Annals of Statistics*, 2013.

Vianney Perchet, Philippe Rigollet, Sylvain Chassang, and Erik Snowberg. Batched bandit problems. *The Annals of Statistics*, 44(2):660 – 681, 2016.

Wei Qian and Yuhong Yang. Kernel estimation and model combination in a bandit problem with covariates. *Journal of Machine Learning Research*, 17(149), 2016.

Henry Reeve, Joe Mellor, and Gavin Brown. The k-nearest neighbour ucb algorithm for multi-armed bandits with covariates. In Firdaus Janoos, Mehryar Mohri, and Karthik Sridharan, editors, *Proceedings of Algorithmic Learning Theory*, volume 83 of *Proceedings of Machine Learning Research*, pages 725–752. PMLR, 07–09 Apr 2018.

Zhimei Ren, Zhengyuan Zhou, and Jayant R. Kalagnanam. Batched learning in generalized linear contextual bandits with general decision sets. *IEEE Control Systems Letters*, 6:37–42, 2022.

Philippe Rigollet and Assaf Zeevi. Nonparametric bandits with covariates. *Conference on Learning Theory (COLT)*, page 54, 2010.

Eric M. Schwartz, Eric T. Bradlow, and Peter S. Fader. Customer acquisition via display advertising using multi-armed bandits. *Marketing Science*, 36(4):500–522, 2017.

Ambuj Tewari and Susan A Murphy. From ads to interventions: Contextual bandits in mobile health. *Mobile health: sensors, analytic methods, and applications*, pages 495–517, 2017.

Alexandre B. Tsybakov. Optimal aggregation of classifiers in statistical learning. *Annals of Statistics*, 32(1):135–166, 2004. doi: 10.1214/aos/1079120131.

Alexandre B. Tsybakov. *Introduction to Nonparametric Estimation*. Springer Series in Statistics. Springer, 2009. ISBN 978-0-387-79051-0. URL `https://link.springer.com/book/10.1007/b13794`.

Michal Valko, Nathan Korda, Rémi Munos, Ilias Flaounas, and Nello Cristianini. Finite-time analysis of kernelised contextual bandits. In *Proceedings of the Twenty-Ninth Conference on Uncertainty in Artificial Intelligence*, pages 654–663, 2013.

Yuhong Yang and Dan Zhu. Randomized allocation with nonparametric estimation for a multi-armed bandit problem with covariates. *The Annals of Statistics*, 30(1):100–121, 2002.

Puning Zhao, Jiafei Wu, Zhe Liu, and Huiwen Wu. Contextual bandits for unbounded context distributions. *arXiv preprint arXiv:2408.09655*, 2024.

Dongruo Zhou, Lihong Li, and Quanquan Gu. Neural contextual bandits with ucb-based exploration. In *International Conference on Machine Learning*, pages 11492–11502. PMLR, 2020.

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

2. **Limitations**

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

   Answer: [Yes] See the discussion at the end of Section 6

3. **Theory assumptions and proofs**

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

   Answer: [Yes] Assumptions 2-4 are clearly stated in Section 2. All proofs are provided in the Appendices B and C.

4. **Experimental result reproducibility**

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

   Answer: [Yes] All the implementation details needed for reproducibility are provided in Section 5. Code can be provided upon request.

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

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

   Answer: [Yes] We cited original sources for all public datasets used. We have created a private Github repository for the code and we plan on releasing code upon acceptance.

6. **Experimental setting/details**

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

   Answer: [Yes] Parameter choice details are described in Section 5 and in the Appendix C.1.

7. **Experiment statistical significance**

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

   Answer: [Yes] Standard error bands are shown in all plots in Figures 1 and 2.

8. **Experiments compute resources**

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

   Answer: [NA] Experiments were run on a standard CPU; no specialized compute resources were required.

9. **Code of ethics**

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

   Answer: [Yes]

10. **Broader impacts**

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

    Answer: [NA] The work is methodological in nature and not applied to sensitive or high-risk domains.

11. **Safeguards**

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

    Answer: [NA] Our work does not involve the release of pretrained models, generative systems, or scraped datasets. It presents a theoretical and algorithmic contribution with empirical validation on standard synthetic and benchmark datasets, which pose minimal risk of misuse.

12. **Licenses for existing assets**

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

    Answer: [Yes] All external assets used in our work—such as models, theoretical contributions, benchmark datasets and baseline implementations—are properly cited in the manuscript.

13. **New assets**

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

    Answer: [NA] We do not release any new assets with the submission. If the paper is accepted, we plan to release code with appropriate documentation.

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

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

    Answer: [NA] This work does not involve crowdsourcing or research with human subjects.

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

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

    Answer: [NA] This work does not involve human subjects, and thus no IRB or equivalent approval was required.

16. **Declaration of LLM usage**

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

    Answer: [NA] This work does not use large language models (LLMs) as part of the core methodology.

A Appendix

In this section, we provide the detailed proofs for the results in Theorem 1 and 2, respectively. First we present the supporting lemmas for establishing the upper bound for the expected regret in Section B, then in Section C we present the proof for the regret lower bound with supporting lemmas.

B Proof for the Regret Upper Bound

Recall, the batch-wise expected sample density, $p_a^{(m)}(x)$, from (11). In Lemma 2, we first construct an upper bound for $p_a^{(m)}(x)$ in terms of the context density $p_X(x)$.

Lemma 2. *The batch-wise expected sample density satisfies:*

$$p_a^{(m)}(x) \leq (t_m - t_{m-1})p_X(x),$$

for almost all $x \in \mathcal{X}$.

Proof. Note, since the event $\{X_t \in A\} \subseteq \{X_t \in A, a_t = a\}$,

$$\mathbb{E}\left[\sum_{t=t_{m-1}}^{t_m} 1(X_t \in A, a_t = a)\right] \leq (t_m - t_{m-1}) \int_A p_X(x)dx. \tag{21}$$

From (11) and (21), we get that,

$$\int_A p_a^{(m)}(x)dx \leq (t_m - t_{m-1}) \int_A p_X(x)dx,$$

for all $A \in \mathcal{X}$. Therefore, $p_a^{(m)}(x) \leq (t_m - t_{m-1})p_X(x)$ for almost all $x \in \mathcal{X}$. □

Next, we build a concentration bound on the average model noise for the k-nearest neighbors around a point x. Here, we will use the sub-Gaussianity of noise (Assumption 1) and the fact that we only observe data until the last batch, i.e., for $t \in [t_{m-1} + 1, t_m]$, we can only utilize data until time t_{m-1} for estimation.

Lemma 3. *Let $N_{t_{m-1},k}(x, a)$ denote the set of k nearest neighbors among $\{X_i : i < t_{m-1}, a_i = a\}$. Then, for all $x \in \mathcal{X}$, $a \in \mathcal{A}$, and $k \geq 1$, we have that,*

$$\mathbb{P}\left(\sup_{x,a,k}\left|\frac{1}{\sqrt{k}}\sum_{i\in\mathcal{N}_{t_{m-1},k}(x,a)} \epsilon_i\right| > u\right) \leq dt_{m-1}^{2d+1}|\mathcal{A}|e^{-\frac{u^2}{2\sigma^2}}, \tag{22}$$

where ϵ_i are independent sub-Gaussian noise terms with variance proxy σ^2.

Proof of Lemma 3. From Lemma 4 of Zhao et al. [2024], we have that of a fixed k:

$$\mathbb{P}\left(\sup_{x,a}\left|\frac{1}{\sqrt{k}}\sum_{i\in\mathcal{N}_{t_{m-1},k}(x,a)} \epsilon_i\right| > u\right) \leq dt_{m-1}^{2d}|\mathcal{A}|e^{-\frac{u^2}{2\sigma^2}}. \tag{23}$$

Then we apply a union bound over all $k \leq t_{m-1}$ to get,

$$\mathbb{P}\left(\sup_{x,a,k}\left|\frac{1}{\sqrt{k}}\sum_{i\in\mathcal{N}_{t,k}(x,a)} \epsilon_i\right| > u\right) \leq dt_{m-1}^{2d+1}|\mathcal{A}|e^{-\frac{u^2}{2\sigma^2}}.$$

□

Note, that Lemma 3 is for any batch m and we will use it to bound the batch-wise regret.

Definition 1. *Define the event $\mathcal{E}_m$ as*

$$\mathcal{E}_m := \left\{\left|\frac{1}{\sqrt{k}}\sum_{i\in\mathcal{N}_{t_{m-1},k}(x,a)} \epsilon_i\right| \leq \sqrt{2\sigma^2 \ln(dt_{m-1}^{2d+3}|\mathcal{A}|)}\ \forall\, x, a, k\right\}, \tag{24}$$

Then, from Lemma 3, it follows that $\mathbb{P}(\mathcal{E}_m) \geq 1 - 1/t_m$.

Lemma 4. *Under $\mathcal{E}_m$, we have that the following point-wise estimation error bound for $x \in \mathcal{X}$ and $t \in [t_{m-1}+1, t_m]$:*

$$f_a(x) \leq \widehat{f}_{a,t}(x) \leq f_a(x) + 2\xi_{a,t}(x) + 2Ld_{a,t}(x), \tag{25}$$

where $\xi_{a,t}(x)$ and $d_{a,t}(x)$ are as defined in (8) *and* (5)*, respectively.*

Proof. Observe that for $t \in [t_{m-1}+1, t_m]$, under event $\mathcal{E}_m$ and $x \in \mathcal{X}$:

$$\begin{aligned}
&\left|\widehat{f}_{a,t}(x) - (f_a(x) + \xi_{a,t}(x) + Ld_{a,t}(x))\right| & (26)\\
&\leq \left|\frac{1}{k_{a,t}(x)} \sum_{i \in \mathcal{N}_t(x,a)} (Y_i - f_a(x))\right| \\
&\leq \frac{1}{k_{a,t}(x)} \sum_{i \in \mathcal{N}_t(x,a)} (Y_i - f_a(X_i)) + \frac{1}{k_{a,t}(x)} \sum_{i \in \mathcal{N}_t(x,a)} (f_a(X_i) - f_a(x)) \\
&\leq \xi_{a,t}(x) + Ld_{a,t}(x), & (27)
\end{aligned}$$

where the last line uses the definition of $\mathcal{E}_m$ in (24) and the Lipschitz (smoothness) property (Assumption 3) of f_a. □

Quantities of interest: We define some important quantities of interest which are central to the proof. This includes two population quantities:

$$r_a(x) = \frac{1}{2L\sqrt{C_1}}(f_*(x) - f_a(x)), \tag{28}$$

$$n_a^{(m)}(x) = \frac{C_1 \ln t_{m-1}}{(f_*(x) - f_a(x))^2}, \tag{29}$$

in which

$$C_1 = \max\left\{4, 32\sigma^2(2d + 3 + \log(Md|\mathcal{A}|))\right\}. \tag{30}$$

The quantity $n_a^{(m)}(x)$ can be interpreted as a *local sample complexity proxy*, capturing the number of samples required near x to estimate the reward function $f_a(x)$ with sufficient precision. Then, another quantity of interest is a data-dependent quantity that measures the total number of observations until time t_{m-1} corresponding to arm a in a radius r ball around x. For any $x \in \mathcal{X}, a \in \mathcal{A}$ define,

$$n^{(m)}(x, a, r) := \sum_{t=1}^{t_{m-1}} 1(\|X_t - x\| < r, a_t = a). \tag{31}$$

Next in Lemma 5, under the event $\mathcal{E}_m$, we show that the adaptive choice of $k_{a,t}$ from (6) in our k-NN estimator is in fact upper bounded by $n_a^{(m)}(x)$. Then, in Lemma 6, we show that $n^{(m)}(x, a, r) \leq k_{a,t}(x)$, which then leads to the relationship between $n_a^{(m)}(x)$ and $n^{(m)}(x, a, r)$ in Lemma 7.

Lemma 5. *Under event $\mathcal{E}_m$ for $t \in [t_{m-1}+1, t_m]$,*

$$k_{a,t}(x) \leq n_a^{(m)}(x).$$

Proof. We prove this by contradiction. Let $k_{a,t}(x) > n_a^{(m)}(x)$. By definition of $k_{a,t}$ in (6):

$$Ld_{a,t}(x) = Ld_{a,t,k_{a,t}(x)}(x) \leq \sqrt{\frac{\ln(t_{m-1})}{k_{a,t}(x)}} \leq \sqrt{\frac{\ln t_{m-1}}{n_a^{(m)}(x)}} = 2Lr_a(x), \tag{32}$$

From Lemma 4, under $\mathcal{E}_m$,

$$\begin{aligned}
\widehat{f}_{a_t,t}(x) &\leq f_{a_t}(x) + 2\sqrt{\frac{2\sigma^2}{k_{a_t,t}(x)} \ln(dMt_{m-1}^{2d+3}|\mathcal{A}|)} + 2Lr_{a_t}(x) \\
&\leq f_{a_t}(x) + 2\sqrt{\frac{2\sigma^2}{n_{a_t}^{(m)}(x)} \ln(dMt_{m-1}^{2d+3}|\mathcal{A}|)} + 2Lr_{a_t}(x). & (33)
\end{aligned}$$

Since action a_t is selected at time t, from the proposed UCB algorithm (Algorithm 1), i.e., the choice of $a_t = \arg\max_{a\in\mathcal{A}} \hat{f}_{a,t}(X_t)$ and from Lemma 4,

$$\hat{f}_{a_t,t}(x) \geq \hat{f}_{a^*(x),t}(x) \geq f^*(x). \tag{34}$$

Combining (33) and (34) gives:

$$2\sqrt{\frac{2\sigma^2}{n_{a_t}^{(m)}(x)}\ln(dt_{m-1}^{2d+3}|\mathcal{A}|)} + 2Lr_{a_t}(x) \geq f_*(x) - f_{a_t}(x). \tag{35}$$

We now derive an inequality that contradicts with (35). From (29) and (30),

$$\begin{aligned} 2\sqrt{\frac{2\sigma^2}{n_{a_t}^{(m)}(x)}\ln(dt_{m-1}^{2d+3}|\mathcal{A}|)} &= 2\sqrt{\frac{2\sigma^2}{C_1\ln t_{m-1}}\ln(dt_{m-1}^{2d+3}|\mathcal{A}|)(f^*(x) - f_{a_t}(x))^2} \\ &\leq \frac{1}{2}\sqrt{\frac{\ln(dt_{m-1}^{2d+3}|\mathcal{A}|)}{(2d+3+\ln(d|\mathcal{A}|))\ln(t_{m-1})}}(f^*(x) - f_{a_t}(x)) \\ &< \frac{1}{2}(f^*(x) - f_{a_t}(x)). \end{aligned} \tag{36}$$

From the definition of $r_a(x)$ in (28),

$$2Lr_{a_t}(x) = \frac{1}{\sqrt{C_1}}(f^*(x) - f_{a_t}(x)) \leq \frac{1}{2}(f^*(x) - f_{a_t}(x)). \tag{37}$$

From (36) and (37),

$$2\sqrt{\frac{2\sigma^2}{n_{a_t}^{(m)}(x)}\ln(dt_{m-1}^{2d+3}|\mathcal{A}|)} + 2Lr_{a_t}(x) < f^*(x) - f_{a_t}(x). \tag{38}$$

Note that (35) contradicts (38). Hence, the desired conclusion follows. $\square$

Lemma 6. *Under $\mathcal{E}_m$, let $r_a(x) \geq \frac{2LC_1}{\sqrt{C_1}-2}$ and $k_{a,t}(x) \gtrsim \ln T$, then, we get*

$$n^{(m)}(x, a, r_a(x)) \leq k_{a,t}(x),$$

where $r_a(x)$ is as defined in (28), *$n^{(m)}(x, a, r_a(x))$ defined in* (31) *and $k_{a,t}$ as defined in* (6).

Proof of Lemma 6. We also prove Lemma 6 by contradiction. If $n^{(m)}(x, a, r_a(x)) > k_{a,t}(x)$, let

$$t = \max\{\tau < t_{m-1} \mid \|x_\tau - x\| \leq r_a(x), A_\tau = a\}. \tag{39}$$

be the last step falling in $B(x, r_a(x))$ with action a. Then $B(x, r_a(x)) \subseteq B(X_t, 2r_a(x))$, and thus there are at least $k_{a,t}(x)$ points in $B(X_t, 2r_a(x))$. Therefore, for any $x \in \mathcal{X}$, by the definition of $d_{a,t}(x)$, i.e., the distance of x to its k^{th} nearest-neighbors in (5),

$$d_{a,t}(x) < 2r_a(x). \tag{40}$$

Denote $a^*(x) = \arg\max_a f_a(x)$ as the best action at context x. Again, note that $a_t = a$ is selected only if the UCB of action a is not less than the UCB of action $a^*(x)$, i.e.,

$$\hat{f}_{a,t}(X_t) \geq \hat{f}_{a^*(X_t),t}(X_t). \tag{41}$$

From Lemma 4,

$$\hat{f}_{a,t}(X_t) \leq f_a(X_t) + 2\xi_{a,t}(X_t) + 2Ld_{a,t}(X_t), \tag{42}$$

and

$$\hat{f}_{a^*(X_t),t}(X_t) \geq f_{a^*(X_t)}(X_t) = f^*(X_t). \tag{43}$$

From (41), (42), and (43),

$$f_a(X_t) + 2\xi_{a,t}(X_t) + 2Ld_{a,t}(X_t) \geq f^*(X_t). \tag{44}$$

which yields,

$$
\begin{aligned}
d_{a,t}(X_t) &\geq \frac{f^*(X_t) - f_a(X_t) - 2\xi_{a,t}(X_t)}{2L} \\
&\geq \frac{f^*(X_t) - f_a(X_t) - 2\sqrt{\frac{2\sigma^2 \ln(dMT^{2d+3}|\mathcal{A}|)}{k_{a,t}(x)}}}{2L} \\
&\geq \frac{f^*(X_t) - f_a(X_t) - 2\sqrt{\frac{2\sigma^2 \ln(dMT^{2d+3}|\mathcal{A}|)}{\ln T}}}{2L} \\
&= \sqrt{C_1} r_a(X_t) - \frac{1}{L}\sqrt{\frac{2\sigma^2 \ln(dMT^{2d+3}|\mathcal{A}|)}{\ln T}} \\
&\geq \sqrt{C_1} r_a(X_t) - \frac{\sqrt{C_1}}{L} \\
&\geq 2 r_a(X_t),
\end{aligned} \tag{45}
$$

using the fact that $r_a(x) \geq \frac{2LC_1}{\sqrt{C_1}-2}$ and $k_{a,t}(x) \gtrsim \ln T$. Note that (45) contradicts (40). Therefore $n^{(m)}(x, a, r_a(x)) \leq k_{a,t}(x)$. That completes the proof of Lemma 6. □

Lemma 7. *For $n_a(x)$ defined in* (29) *and $n^{(m)}(x, a, r)$ as defined in* (31)*, under $\mathcal{E}_m$,*

$$n^{(m)}(x, a, r_a(x)) \leq n_a^{(m)}(x).$$

Proof. Combining the results of Lemma 5 and 6 proves Lemma 7. □

Bounding the batch-wise regret $R_a^{(m)}$: From Lemma 7 and from Lemma 3, we know that $\mathbb{P}(\mathcal{E}_m^c) \leq 1/t_m$ and $n^{(m)}(x, a, r_a(x)) < t_m$ on $\mathcal{E}_m$ gives:

$$
\begin{aligned}
\mathbb{E}\left[n^{(m)}(x, a, r_a(x)) \mid \mathcal{F}_{t_{m-1}}\right] &\leq \mathbb{P}(\mathcal{E}_m|\mathcal{F}_{t_{m-1}})\mathbb{E}\left[n^{(m)}(x, a, r_a(x)) \mid \mathcal{E}_m, \mathcal{F}_{t_{m-1}}\right] \\
&\quad + \mathbb{P}(\mathcal{E}_m^c|\mathcal{F}_{t_{m-1}})\mathbb{E}\left[n^{(m)}(x, a, r_a(x)) \mid \mathcal{E}_m^c, \mathcal{F}_{t_{m-1}}\right] \\
&\leq n_a^{(m)}(x) + 1.
\end{aligned} \tag{46}
$$

From the definition of $p_a^{(m)}$ in (11),

$$\int_{B(x, r_a(x))} p_a^{(m)}(u) du \leq n_a^{(m)}(x) + 1. \tag{47}$$

Recall $R_a^{(m)}$ from (12). We first bound $R_a^{(m)}$ for a given m to get a bound on the expected regret using Lemma 1. To bound $R_a^{(m)}$, we introduce a new random variable Z follow a distribution with probability density function (pdf) ϕ:

$$\phi(z) = \frac{1}{C_Z \left[(f^*(z) - f_a(z)) \vee \epsilon\right]^d}, \tag{48}$$

where C_Z is the normalizing constant. As discussed in Section 4, we split $R_a^{(m)}$ into two regions: one where the suboptimality gap is large (where concentration bounds dominate) and another where the margin condition helps control the measure of near-optimal points,

$$
\begin{aligned}
R_a^{(m)} = &\int_{\mathcal{X}} (f_*(x) - f_a(x)) p_a^{(m)}(x) 1(f_*(x) - f_a(x) > \epsilon) dx \\
&+ \int_{\mathcal{X}} (f_*(x) - f_a(x)) p_a^{(m)}(x) 1(f_*(x) - f_a(x) \leq \epsilon) dx.
\end{aligned}
$$

The idea is to bound these two terms separately, where the second one can be bounded using the margin assumption (i.e., Assumption 4). The ϵ is determined theoretically based on the bound on $R_a^{(m)}$. We tackle the first integral term in the following Lemma 8.

Lemma 8. *There exists a constant $C_2 > 0$ such that for any $a \in \mathcal{A}$,*

$$\int_{\mathcal{X}} (f^*(x) - f_a(x)) p_a^{(m)}(x) 1\left(f^*(x) - f_a(x) > \epsilon\right) dx$$
$$\leq C_2 C_Z \,\mathbb{E}\left[\int_{B(Z, r_a(Z))} p_a^{(m)}(u) \left(f^*(u) - f_a(u)\right) du \,\middle|\, \mathcal{F}_{t_{m-1}} \right],$$

where $Z \sim \phi$ is a density function defined over $\mathcal{X}$.

Proof. Consider,

$$\begin{aligned}
&\mathbb{E}\left[\int_{B(Z, r_a(Z))} p_a^{(m)}(u) \left(f^*(u) - f_a(u)\right) du \middle| \mathcal{F}_{t_{m-1}} \right] && (49)\\
&\overset{(a)}{=} \int_{\mathcal{X}} \int_{B(u, 2r_a(u)/3)} \phi(z) p_a^{(m)}(u) \left(f^*(u) - f_a(u)\right) dz du \\
&\geq \int_{\mathcal{X}} \left(\inf_{\|z-u\| \leq 2r_a(u)/3} \phi(z) \right) \left(\frac{2}{3}\right)^d r_a^d(u) p_a^{(m)}(u) \left(f^*(u) - f_a(u)\right) du \\
&\overset{(b)}{\geq} \left(\frac{2}{3}\right)^d \left(\frac{3}{4}\right)^d \int_{\mathcal{X}} \phi(u) r_a^d(u) p_a^{(m)}(u) \left(f^*(u) - f_a(u)\right) du \\
&= \frac{1}{2^d C_Z} \int_{\mathcal{X}} \frac{1}{\left[(f^*(u) - f_a(u)) \vee \epsilon\right]^d} r_a^d(u) p_a^{(m)}(u) \left(f^*(u) - f_a(u)\right) du \\
&\geq \frac{1}{2^d C_Z} \int_{\mathcal{X}} \mathbf{1}(f^*(u) - f_a(u) > \epsilon) \frac{1}{(f^*(u) - f_a(u))^d} \frac{(f^*(u) - f_a(u))^d}{(4L)^d} \\
&\qquad \times p_a^{(m)}(u) \left(f^*(u) - f_a(u)\right) du \\
&\geq \frac{1}{2^{3d} L^d C_Z} \int_{\mathcal{X}} p_a^{(m)}(u) \left(f^*(u) - f_a(u)\right) \mathbf{1}(f^*(u) - f_a(u) > \epsilon) du. && (50)
\end{aligned}$$

For (a), if $\|u - z\| \leq r_a(z)$, then from the definition of r_a in (28) and using the Lipschitz assumption (Assumption 3), we get that:

$$\begin{aligned}
\frac{r_a(u)}{r_a(z)} &= \frac{f^*(u) - f_a(u)}{f^*(z) - f_a(z)} \\
&= \frac{f^*(u) - f^*(z) + f_a(z) - f_a(u) + f^*(z) - f_a(z)}{f^*(z) - f_a(z)} \\
&\leq \frac{f^*(z) - f_a(z) + 2L r_a(z)}{f^*(z) - f_a(z)} \\
&= 1 + \frac{1}{\sqrt{C_1}} \\
&\leq \frac{3}{2}. && (51)
\end{aligned}$$

For (b), we have that $\|z - u\| \leq \frac{2r_a(u)}{3}$, therefore we have that:

$$|f^*(u) - f^*(z)| \leq \frac{2}{3} r_a(u), \text{ and } |f_a(u) - f_a(z)| \leq \frac{2}{3} r_a(u).$$

Therefore,

$$\begin{aligned}
|f^*(z) - f_a(z) - (f^*(u) - f_a(u))| &\leq \frac{4}{3} r_a(u) \\
\Rightarrow (f^*(z) - f_a(z)) \vee \epsilon &\leq \left((f^*(u) - f_a(u) + \frac{4}{3} r_a(u)) \right) \vee \epsilon.
\end{aligned}$$

Therefore, we get that,

$$\begin{aligned}\frac{\phi(z)}{\phi(u)} &= \frac{[(f^*(u)-f_a(u))\vee\epsilon]^d}{[(f^*(z)-f_a(z))\vee\epsilon]^d}\\ &\geq \frac{[(f^*(u)-f_a(u))\vee\epsilon]^d}{\left[(f^*(u)-f_a(u))+\frac{4}{3}Lr_a(u)\right]^d}\\ &\geq \left(\frac{3}{4}\right)^d. \end{aligned} \tag{52}$$

where (52) follows because,

$$\begin{aligned}f^*(u)-f_a(u)+\frac{4}{3}Lr_a(u) &= f^*(u)-f_a(u)+\frac{4}{3}L\cdot\frac{1}{2L\sqrt{C_1}}(f^*(u)-f_a(u))\\ &= (f^*(u)-f_a(u))\left(1+\frac{2}{3\sqrt{C_1}}\right).\end{aligned}$$

Since $\sqrt{C_1}\geq 2$, then (52) holds. □

Next, we prove an inequality that plays a key role in bounding the regret contribution from contexts where the reward gap is large.

Lemma 9.

$$\int_{\mathcal{X}}(f^*(z)-f_a(z))^{-(d-1)}1(f^*(z)-f_a(z)>\epsilon)\,dz \lesssim \begin{cases}\epsilon^{\alpha+1-d} & \textit{if } d>\alpha+1,\\ \log\left(\frac{1}{\epsilon}\right) & \textit{if } d=\alpha+1,\\ 1 & \textit{if } d<\alpha+1.\end{cases} \tag{53}$$

Proof of Lemma 9. Consider

$$\begin{aligned}&\int_{\mathcal{X}}(f^*(z)-f_a(z))^{-(d-1)}1(f^*(z)-f_a(z)>\epsilon)\,dz \qquad (54)\\ &\overset{(a)}{\leq} \frac{1}{\underline{c}}\int_{\mathcal{X}}(f^*(z)-f_a(z))^{-(d-1)}1(f^*(z)-f_a(z)>\epsilon)\,p_X(z)\,dz\\ &\overset{(b)}{=} \frac{1}{\underline{c}}\mathbb{E}\left[(f^*(X)-f_a(X))^{-(d-1)}1(f^*(X)-f_a(X)>\epsilon)\right]\\ &= \frac{1}{\underline{c}}\int_0^\infty \mathbb{P}\left(\epsilon<f^*(X)-f_a(X)<t^{-\frac{1}{d-1}}\right)dt\\ &\leq \frac{1}{\underline{c}}\int_0^{\epsilon^{-(d-1)}}\mathbb{P}\left(f^*(X)-f_a(X)<t^{-\frac{1}{d-1}}\right)dt \qquad (55)\end{aligned}$$

(a) comes from Assumption 2, which requires that $p_X(x)\geq\underline{c}$ over the support. In (b), the random variable X follows a distribution with pdf p_X.

If $d>\alpha+1$, then from Assumption 4,

$$(55)\leq\frac{D_\alpha}{\underline{c}}\int_0^{\epsilon^{-(d-1)}} t^{-\frac{\alpha}{d-1}}\,dt = \frac{D_\alpha(d-1)}{\underline{c}(d-1-\alpha)}\epsilon^{\alpha+1-d}. \tag{56}$$

If $d=\alpha+1$, then

$$(55)\leq\frac{1}{\underline{c}}\int_0^1 dt+\frac{D_\alpha}{\underline{c}}\int_1^{\epsilon^{-(d-1)}} t^{-\frac{\alpha}{d-1}}\,dt = \frac{1}{\underline{c}}+\frac{D_\alpha(d-1)}{\underline{c}}\log\left(\frac{1}{\epsilon}\right). \tag{57}$$

If $d<\alpha+1$, then

$$(55)\leq\frac{1}{\underline{c}}\int_0^1 dt+\frac{D_\alpha}{\underline{c}}\int_1^{\epsilon^{-(d-1)}} t^{-\frac{\alpha}{d-1}}\,dt \leq \frac{1}{\underline{c}}+\frac{D_\alpha(d-1)}{\underline{c}(\alpha+1-d)}. \tag{58}$$

Therefore, combining results from (55), (56), (57), and (58) we obtain:

$$\int_{\mathcal{X}}(f^*(z)-f_a(z))^{-(d-1)}1(f^*(z)-f_a(z)>\epsilon)\,dz \lesssim \begin{cases}\frac{1}{\underline{c}}\epsilon^{\alpha+1-d} & \text{if } d>\alpha+1,\\ \frac{1}{\underline{c}}\log\left(\frac{1}{\epsilon}\right) & \text{if } d=\alpha+1,\\ \frac{1}{\underline{c}} & \text{if } d<\alpha+1.\end{cases} \tag{59}$$

This proves (53). □

Lemma 10. *Suppose Assumptions 1 and 2 hold. Then, for any batch $m \in [M]$, and for all arms $a \in \mathcal{A}$, we have:*

$$\mathbb{E}\left[\int_{B(Z,r_a(Z))} p_a^{(m)}(u)(\eta^*(u)-\eta_a(u))\,du \,\Big|\, \mathcal{F}_{t_{m-1}}\right] \lesssim \frac{1}{C_Z}\left(\epsilon^{\alpha-d-1}\log t_{m-1} + t_m\epsilon^{1+\alpha}\right).$$

Here C_Z is the density lower bound constant from (48) *and $\mathcal{F}_{t_{m-1}}$ is the history until the $(m-1)^{th}$ batch.*

Proof. Consider:

$$\begin{aligned}
&\mathbb{E}\left[\int_{B(Z,r_a(Z))} p_a^{(m)}(u)(f^*(u)-f_a(u))\,du \,\Big|\, \mathcal{F}_{t_{m-1}}\right] \\
&\overset{(a)}{\leq} \frac{3}{2}\mathbb{E}\left[\int_{B(Z,r_a(Z))} p_a^{(m)}(u)(f^*(z)-f_a(z))\,du \,\Big|\, \mathcal{F}_{t_{m-1}}\right] \\
&\overset{(b)}{\leq} \frac{3}{2}\mathbb{E}\left[((n_a^{(m)}(Z)+1)\wedge(t_m p_Z(z) r_a^d(Z)))(f^*(Z)-f_a(Z)) \,\Big|\, \mathcal{F}_{t_{m-1}}\right] \\
&= \frac{3}{2}\int \left((n_a^{(m)}(z)+1)\wedge(t_m p_Z(z) r_a^d(Z))\right)(f^*(z)-f_a(z))\frac{1}{\phi_Z[(f^*(z)-f_a(z))\vee\epsilon]^d}dz \\
&= \frac{3}{2}\int \left((n_a^{(m)}(z)+1)\wedge(t_m p_Z(z) r_a^d(Z))\right)(f^*(z)-f_a(z))\frac{1}{\phi_Z[(f^*(z)-f_a(z))]^d} \\
&\qquad\qquad \times 1(f^*(z)-f_a(z)>\epsilon)dz \\
&\quad + \frac{3}{2}\int \left((n_a^{(m)}(z)+1)\wedge(t_m p_Z(z) r_a^d(Z))\right)(f^*(z)-f_a(z))\frac{1}{\phi_Z\epsilon^d}1(f^*(z)-f_a(z)\leq\epsilon)dz,
\end{aligned} \tag{60}$$

For (a):

$$\begin{aligned}
f^*(u)-f_a(u) &\leq f^*(z)-f_a(z)+2Lr_a(z) \\
&\leq f^*(z)-f_a(z)+\frac{1}{\sqrt{C_1}}(f^*(z)-f_a(z)) \\
&\leq \frac{3}{2}(f^*(z)-f_a(z)).
\end{aligned} \tag{61}$$

We get (b) from Lemma 2 and (46). In (60), we split the domain based on whether $(f^*(z)-f_a(z))$ is large or small, and use the margin assumption (Assumption 4) for the latter. Note that, If $f^*(Z)-f_a(Z)>\epsilon$, then $n_a^{(m)}(Z)=(\log t_{m-1})(f^*(Z)-f_a(Z))^{-2}$ is smaller, otherwise the bias dominates.

$$\begin{aligned}
(60) &= \frac{3}{2C_Z}\left(\int\left(\frac{C_1\ln t_{m-1}}{(f^*(z)-f_a(z))}+f^*(z)-f_a(z)\right)\frac{1}{(f^*(z)-f_a(z))^d}1(f^*(z)-f_a(z)>\epsilon)dz\right. \\
&\qquad \left. + \int t_m p_Z(z) r_a^d(Z)(f^*(z)-f_a(z))\frac{1}{\epsilon^d}1(f^*(z)-f_a(z)\leq\epsilon)dz\right) \\
&\lesssim \frac{1}{C_Z}\left(\mathbb{E}\left[(f^*(Z)-f_a(Z))^{-(d+1)}1(f^*(Z)-f_a(Z)>\epsilon)\right]\ln t_{m-1}\right. \\
&\qquad \left. + \frac{t_m}{\epsilon^d}\mathbb{E}\left[(f^*(Z)-f_a(Z))^{d+1}1(f^*(Z)-f_a(Z)\leq\epsilon)\right]\right) \\
&\overset{(c)}{\lesssim} \frac{1}{C_Z}\left(\epsilon^{\alpha-d-1}\ln t_{m-1}+t_m\epsilon^{1+\alpha}\right),
\end{aligned}$$

where the first term in (c) comes from the dominating term in Lemma 9 and for the second term we use the Margin assumption as follows:

$$\begin{aligned}
\int_{\mathcal{X}}(f^*(z)-f_a(z))1(f^*(z)-f_a(z)<\epsilon)dz &\leq \frac{1}{\underline{c}}\mathbb{E}\left[(f^*(X)-f_a(X))1(f^*(X)-f_a(X)<\epsilon)\right] \\
&\leq \frac{L_0}{\underline{c}}\epsilon^{\alpha+1}.
\end{aligned} \tag{62}$$

This concludes the proof. ∎

C Proof for Regret Lower Bound

In this section, we prove that a lower bound on the expected regret for the batched nonparametric bandits framework. First, we state a well-known Lemma from Perchet and Rigollet [2013].

Lemma 11. *There exists a constant C_0 such that the expected cumulative regret R is related to the inferior sampling rate defined in* (20).

$$R \geq C_0 S^{\frac{\alpha+1}{\alpha}} T^{-\frac{1}{\alpha}}. \tag{63}$$

For proof of Lemma 11, we refer the reader to Perchet and Rigollet [2013]. Next, we provide a proof for Theorem 2.

Proof of Theorem 2. For establishing the lower bound, we only discuss the case with only two arms, say, $\mathcal{A} = \{-1, 1\}$. Construct B disjoint balls with centers $a_1, a_2, \ldots, a_B$ with radius h. The probability measure $\mathbb{P}_X$ is assumed to be absolutely continuous with respect to the Lebesgue measure such that the density function p_X is given by:

$$p_X(x) = \sum_{j=1}^{B} 1(x \in \mathcal{B}_j), \tag{64}$$

where $\mathcal{B}_j = \{x' \mid \|x' - a_j\| \leq h\}$ for $x \in \mathcal{X}$ is the jth ball of radius h centered at a_j. To ensure that the pdf is well defined, we need $\int p_X(x)dx = 1$, which means that B and h satisfy: $Bh^d \mathrm{Vol}_d = 1$, where Vol_d is the volume of a d-dimensional unit ball.

We consider the two mean rewards functions to be $f_1(x) = f_v(x) \in \mathcal{F}(L, \alpha)$ and $f_2(x) = 0 \in \mathcal{F}(L, \alpha)$ with, $f_v(x) = \sum_{j=1}^{D} v_j h I\{x \in \mathcal{B}_j\}, x \in \mathcal{X}$, where $v_j \in \{-1, 1\}$ for $j = 1, \ldots, D$. Note that,

$$P(0 < |f_v(u)| \leq t) \leq \begin{cases} Dh^d \mathrm{Vol}_d & \text{if } t \geq h \\ 0 & \text{if } t < h. \end{cases} \tag{65}$$

This is because the only non-zero values that f can take are $\pm h$ and when $t < h$, the above probability is 0. For the case when $t \geq h$, the set $\{0 < |f_v(x)| \leq t\}$ is just the union of all intervals where $|f(u)| = h$, hence $P(0 < |f_v(X)| \leq t) = P(X \in \cup_{j=1}^{D} \mathcal{B}_j) = Dh^d \mathrm{Vol}_d$. For $f \in \mathcal{F}(\alpha, \eta)$, we want it to satisfy the margin condition which requires:

$$Dh^d \mathrm{Vol}_d \leq D_\alpha h^\alpha. \tag{66}$$

Note that, this implies that $D\mathrm{Vol}_d \leq D_\alpha h^{\alpha - d}$ which means that in the construction of f_v, D is chosen to satisfy the margin condition for any $h > 0$. We denote the space of functions that satisfy (66):

$$\mathcal{G}_v = \{f_1(x) = f_v(x), f_2(x) = 0 | x \in \{-1, 1\}^D\}.$$

Let $\mathcal{F}(L, \alpha)$ denote the function class satisfying both the Lipschitz condition (Assumption 3) with Lipschitz constant L and Margin condition (Assumption 4). Also, note that, in the batched setting, we have,

$$\sup_{f_1, f_2 \in \mathcal{F}(L, \alpha)} R_T(\pi) \geq \sup_{1 \leq i \leq M} \sup_{f_1, f_2 \in \mathcal{F}(L, \alpha)} R_{t_i}(\pi), \tag{67}$$

therefore, we bound the per-batch regret R_{t_i} using the per-batch inferior sampling rate and Lemma 11. Recall that $\mathcal{T} = \{t_0, t_1, \ldots, t_M\}$ denote the batches in our algorithm. For X_t, consider,

$$\begin{aligned} S_{t_i} &= \sum_{j=1}^{D} \sum_{t=1}^{t_i} P(X_t \in \mathcal{B}_j, a_t \neq a^*(X_t)) && (68) \\ &\geq \sum_{j=1}^{D} \sum_{t=1}^{t_i} \int_{\mathcal{B}_j} P(a_t \neq a^*(X_t) | X_t = x) p_X(x) dx \\ &= \sum_{j=1}^{D} \sum_{t=1}^{t_i} \int_{\mathcal{B}_j} P(a_t \neq v_j | X_t = x) p_X(x) dx \\ &= \sum_{j=1}^{D} \sum_{t=1}^{t_i} \mathbb{E} \left[\int_{\mathcal{B}_j} 1\{\pi(x | \mathcal{F}_{t_{i-1}}) \neq v_j\} p_X(x) dx \right], && (69) \end{aligned}$$

where $\mathcal{F}_{t_{i-1}} = \sigma(X_1, Y_1, a_1, \ldots, X_{t_{i-1}}, Y_{t_{i-1}}, a_{t_{i-1}})$, and $\pi(x|\mathcal{F}_{t_{i-1}})$ denotes the arm choice given the information until the previous batch. Define $\hat{v}_j(t) = \text{sign}\left(\int_{\mathcal{B}_j} \pi(x|\mathcal{F}_{t_{i-1}}) p_X(x) dx\right)$. Intuitively, $\hat{v}_j(t)$ represents the average action the policy π takes across all contexts in ball $\mathcal{B}_j$, weighted by the covariate density p_X, so it is the learner's guess for the true hidden label v_j. Then by the definition of $\hat{v}_j(t)$, it follows that,

$$\int_{\mathcal{B}_j} 1\{\pi(x|\mathcal{F}_{t_{i-1}}) = \hat{v}_j(t)\} p_X(x) dx \geq \int_{\mathcal{B}_j} 1\{\pi(x|\mathcal{F}_{t_{i-1}}) = -\hat{v}_j(t)\} p_X(x) dx. \tag{70}$$

Since,

$$\int_{\mathcal{B}_j} 1\{\pi(x|\mathcal{F}_{t_{i-1}}) = \hat{v}_j(t)\} p_X(x) dx + \int_{\mathcal{B}_j} 1\{\pi(x|\mathcal{F}_{t_{i-1}}) = -\hat{v}_j(t)\} p_X(x) dx = \int_{\mathcal{B}_j} p_X(x) dx, \tag{71}$$

then,

$$\int_{\mathcal{B}_j} 1\{\pi(x|\mathcal{F}_{t_{i-1}}) = \hat{v}_j(t)\} p_X(x) dx \geq \frac{1}{2} \int_{\mathcal{B}_j} p_X(x) dx. \tag{72}$$

If $\hat{v}_j(t) \neq v_j$, then the policy π is agreeing with the wrong label so, $\{\pi(x) = \hat{v}_j(t)\} \subseteq \{\pi(x) \neq v_j\}$, therefore,

$$\mathbb{P}(\pi(x) \neq v_j \mid \hat{v}_j(t) \neq v_j) \geq \mathbb{P}(\pi(x) = \hat{v}_j(t) \mid \hat{v}_j(t) \neq v_j)$$

Therefore, given the event $\hat{v}_j(t) \neq v_j$, we get:

$$\int_{\mathcal{B}_j} 1\{\pi(x|\mathcal{F}_{t_{i-1}}) \neq v_j\} p_X(x) dx \geq \int_{\mathcal{B}_j} 1\{\pi(x|\mathcal{F}_{t_{i-1}}) = \hat{v}_j(t)\} p_X(x) dx \geq \frac{1}{2} \int_{\mathcal{B}_j} p_X(x) dx. \tag{73}$$

Therefore, from (69) and (73),

$$\begin{aligned} S_{t_j} &\geq \sum_{j=1}^{D} \sum_{t=1}^{t_i} \frac{1}{2} P(\hat{v}_j(t) \neq v_j) \int_{\mathcal{B}_j} p(u) du \\ &\geq \sum_{j=1}^{D} \sum_{t=1}^{t_i} \frac{1}{2} h^d \text{Vol}_d P(\hat{v}_j(t) \neq v_j) \end{aligned} \tag{74}$$

Now, we can bound this error probability of hypothesis testing between two probability distributions. Let $V_1, \ldots, V_D$ be the vector of D Rademacher random variables such that $P(V_j = 1) = P(V_j = -1) = 1/2$, and V_j for different values of j are i.i.d. Denote $\mathbb{P}^{t_{i-1}}_{X,Y|V_j=v_j}$ as the joint distribution of $(X_n)_{n=1}^{t_{i-1}}, X_t$ and $(Y_n)_{n=1}^{t_{i-1}}$ given $V_j = v_j$. Then,

$$\begin{aligned} P(\hat{v}_j(t) \neq v_j) &\geq \frac{1}{2}\left(1 - \mathbb{TV}(\mathbb{P}^{t_{i-1}}_{X,Y|V_j=1}, \mathbb{P}^{t_{i-1}}_{X,Y|V_j=-1})\right) \\ &\geq \frac{1}{2}\left(1 - \sqrt{\frac{1}{2} K(\mathbb{P}^{t_{i-1}}_{X,Y|V_j=1}, \mathbb{P}^{t_{i-1}}_{X,Y|V_j=-1})}\right), \end{aligned} \tag{75}$$

in which the second step uses the Pinsker's inequality [Tsybakov, 2009], and $K(p, q)$ denotes the Kullback-Leibler (KL) divergence between distributions p and q. Using Lemma 12, we get that,

$$P(\hat{v}_j(t) \neq v_j) \geq \frac{1}{2}\left(1 - \sqrt{t_{i-1} h^{d+2}}\right). \tag{76}$$

Note that, this bound follows because the only difference in distributions occurs when $X_t \in B_j$ and $a_t = 1$, with the reward differing between $\text{Bern}(h)$ and $\text{Bern}(0)$. Now, plugging in (76) in (74), we get:

$$
\begin{aligned}
S_{t_i} &\geq \frac{1}{4}\sum_{j=1}^{D}\sum_{\ell=1}^{i}(t_\ell - t_{\ell-1})h\left(1-\sqrt{t_{i-1}h^{d+2}}\right)\\
&\geq \frac{D}{4}\sum_{\ell=1}^{i}(t_\ell - t_{\ell-1})h\left(1-\sqrt{t_{i-1}h^{d+2}}\right)\\
&\geq \frac{D_\alpha}{4}\sum_{\ell=1}^{i}(t_\ell - t_{\ell-1})h^{\alpha}\left(1-\sqrt{t_{i-1}h^{d+2}}\right),
\end{aligned}
$$

where (77) follows from (66). We use the convention that $t_0 = 0$. Now, choosing $h = (\frac{t_{i-1}}{2})^{-1/(d+2)}$, we get,

$$
S_{t_i} \geq \begin{cases} c_* \dfrac{t_i}{t_{i-1}^{\alpha/(d+2)}} & \text{when } i > 1 \\ c_* t_1 & \text{when } i = 1 \end{cases}, \tag{77}
$$

for some $c_* > 0$. Now, combining the previous arguments in (67) and using Lemma 11:

$$
\begin{aligned}
\sup_{f_1,f_2\in\mathcal{F}(L,\alpha)} R_T(\pi) &\geq \sup_{1\leq i\leq M}\ \sup_{f_1,f_2\in\mathcal{F}(L,\alpha)} R_{t_i}(\pi)\\
&\geq \sup_{1\leq i\leq M}\ \sup_{f_1,f_2\in\mathcal{G}_v} C_0 S_{t_i}^{\frac{\alpha+1}{\alpha}} t_i^{-\frac{1}{\alpha}}\\
&\gtrsim \left\{ t_1, \frac{t_2}{t_1^{\frac{\alpha+1}{d+2}}}, \frac{t_3}{t_3^{\frac{\alpha+1}{d+2}}}, \dots, \frac{T}{t_{M-1}^{\frac{\alpha+1}{d+2}}} \right\}\\
&\gtrsim \tilde{c} T^{\frac{1-\gamma}{1-\gamma^M}},
\end{aligned}
$$

where $\gamma = \dfrac{\alpha+1}{d+2}$, and we assume $t_i = \lfloor a t_{i-1}^{\frac{1+\alpha}{d+2}} \rfloor$, where $a = O(T^{\frac{1-\gamma}{1-\gamma^M}})$. This completes the minimax lower bound, showing that no M-batch algorithm can outperform the rate achieved by `BaNk-UCB` up to logarithmic factors. ☐

Lemma 12 (KL-divergence lower bound)**.** *Suppose the context density $p_X(x)$ is uniform over disjoint balls $\mathcal{B}_j$ of radius h, with $p_X(x) = 1$ on $\cup_j \mathcal{B}_j$. Let $\mathbb{P}^t_{V_j=v}$ denote the distribution over the learner's trajectory up to time t under $V_j = v$. Then the KL divergence between the two distributions satisfies*

$$
\mathbb{KL}\left(\mathbb{P}^t_{X,Y|V_j=1} \,\middle\|\, \mathbb{P}^t_{X,Y|V_j=-1}\right) \leq 2th^{2+d}. \tag{78}
$$

Proof of Lemma 12. We apply the chain rule for KL divergence as described in Lemma 13 over the interaction sequence:

$$
\begin{aligned}
&\mathrm{KL}(\mathbb{P}^t_{X,Y|v_j=+1} \,\|\, \mathbb{P}^t_{X,Y|v_j=-1})\\
&\quad= \sum_{s=1}^{t} \mathbb{E}_{\mathbb{P}^{s-1}_{X,Y|v_j=+1}}\left[\mathrm{KL}\left(\mathbb{P}(X_s,a_s,Y_s \mid \mathcal{F}_{s-1}, v_j=+1), \mathbb{P}(X_s,a_s,Y_s \mid \mathcal{F}_{s-1}, v_j=-1)\right)\right],
\end{aligned} \tag{79}
$$

where $\mathcal{F}_{s-1}$ denotes the full history up to round $s-1$.
At each round s, note that: $X_s \sim p_X$ is independent of v_j, $a_s \sim \pi_s(\cdot \mid X_s, \mathcal{F}_{s-1})$ is the same under both v_j and only the reward distribution $Y_s \mid X_s, a_s$ depends on v_j. Therefore, for all s, the distributions of (X_s, a_s) under both environments are identical, and we can apply the chain rule for KL at the level of the conditional reward distributions:

$$
\begin{aligned}
&\mathrm{KL}\left(\mathbb{P}(X_s,a_s,Y_s \mid \mathcal{F}_{s-1}, v_j=+1), \mathbb{P}(X_s,a_s,Y_s \mid \mathcal{F}_{s-1}, v_j=-1)\right)\\
&= \mathbb{E}_{X_s\sim p_X,\, a_s\sim\pi_s(\cdot|X_s,\mathcal{F}_{s-1})}\left[\mathrm{KL}\left(\mathbb{P}(Y_s \mid X_s, A_s, \mathcal{F}_{s-1}, v_j=+1), \mathbb{P}(Y_s \mid X_s, A_s, \mathcal{F}_{s-1}, v_j=-1)\right)\right].
\end{aligned} \tag{80}
$$

Using the fact that the reward distributions only differ when $X_s \in \mathcal{B}_j$ and $A_s = 1$, and that the KL between Bern(h) and Bern(0) is at most $2h^2$, we get the pointwise bound:

$$\mathrm{KL}\left(\mathbb{P}(Y_s \mid U_s, A_s, \mathcal{F}_{s-1}, v_j = +1), \mathbb{P}(Y_s \mid U_s, A_s, \mathcal{F}_{s-1}, v_j = -1)\right) \leq 2h^2 \cdot \mathbf{1}(U_s \in \mathcal{B}_j, A_s = 1).$$

Putting this in (80), taking the expectation

$$\begin{aligned}\mathbb{E}_{X_s \sim p_U,\, a_s \sim \pi_s(\cdot|X_s, \mathcal{F}_{s-1})}&\left[\mathrm{KL}\left(\mathbb{P}(Y_s \mid X_s, A_s, \mathcal{F}_{s-1}, v_j = +1), \mathbb{P}(Y_s \mid X_s, A_s, \mathcal{F}_{s-1}, v_j = -1)\right)\right] \\ &\leq 2h^2 \cdot \mathbb{P}_{v_j=+1}(X_s \in \mathcal{B}_j, a_s = 1) \\ &\leq 2h^2 \cdot \mathbb{P}(X_s \in \mathcal{B}_j) = 2h^2 \cdot h^d = 2h^{2+d}.\end{aligned}$$

Summing over $s = 1$ to t in (79) gives:

$$\mathbb{KL}\left(\mathbb{P}^t_{U,Y|V_j=1}, \mathbb{P}^t_{U,Y|V_j=-1}\right) \leq \sum_{s=1}^{t} 2h^{d+2} = 2th^{d+2}.$$

□

Lemma 13 (Chain rule for KL divergence in sequential models)**.** *Let $Z_{1:t} = (Z_1, Z_2, \dots, Z_t)$ be a sequence of random variables (e.g., observations generated in rounds of a bandit process), and let P and Q be two distributions over $Z_{1:t}$ such that $P \ll Q$ (i.e., P is absolutely continuous with respect to Q). Then:*

$$\mathrm{KL}(P(Z_{1:t}) \,\|\, Q(Z_{1:t})) = \sum_{s=1}^{t} \mathbb{E}_{P(Z_{1:s-1})}\left[\mathrm{KL}\left(P(Z_s \mid Z_{1:s-1}) \,\|\, Q(Z_s \mid Z_{1:s-1})\right)\right]. \tag{81}$$

Proof of Lemma 13. We use the chain rule for joint distributions:

$$\begin{aligned}P(Z_{1:t}) &= P(Z_1) \cdot P(Z_2 \mid Z_1) \cdots P(Z_t \mid Z_{1:t-1}), \\ Q(Z_{1:t}) &= Q(Z_1) \cdot Q(Z_2 \mid Z_1) \cdots Q(Z_t \mid Z_{1:t-1}).\end{aligned}$$

Then the KL divergence between the full joint distributions is:

$$\begin{aligned}\mathrm{KL}(P(Z_{1:t}) \,\|\, Q(Z_{1:t})) &= \int P(Z_{1:t}) \log \frac{P(Z_{1:t})}{Q(Z_{1:t})}\, dZ_{1:t} \\ &= \int P(Z_{1:t}) \sum_{s=1}^{t} \log \frac{P(Z_s \mid Z_{1:s-1})}{Q(Z_s \mid Z_{1:s-1})}\, dZ_{1:t} \\ &= \sum_{s=1}^{t} \int P(Z_{1:t}) \log \frac{P(Z_s \mid Z_{1:s-1})}{Q(Z_s \mid Z_{1:s-1})}\, dZ_{1:t}.\end{aligned}$$

Now for each s, we marginalize over $Z_{s+1:t}$ and write:

$$\int P(Z_{1:t}) \log \frac{P(Z_s \mid Z_{1:s-1})}{Q(Z_s \mid Z_{1:s-1})}\, dZ_{1:t} = \int P(Z_{1:s}) \log \frac{P(Z_s \mid Z_{1:s-1})}{Q(Z_s \mid Z_{1:s-1})}\, dZ_{1:s}.$$

This is the definition of:

$$\mathbb{E}_{P(Z_{1:s-1})}\left[\mathrm{KL}(P(Z_s \mid Z_{1:s-1}) \,\|\, Q(Z_s \mid Z_{1:s-1}))\right].$$

Summing over $s = 1$ to t completes the proof. □

C.1 Additional Experiments in Higher Dimensions

We extend the numerical experiments from Section 5.1 to evaluate algorithm performance in higher-dimensional contexts. Specifically, we consider $d \in \{3, 4, 5\}$ while keeping the underlying data-generating mechanisms for both experimental settings unchanged. As expected, the performance of both `BaSEDB` and `BaNk-UCB` deteriorates with increasing dimension, consistent with the theoretical prediction from Theorem 1 and Theorem 2 that regret decays more slowly when d is large due to the corresponding decrease in the parameter γ.
Despite the increased difficulty, `BaNk-UCB` continues to outperform `BaSEDB` across all settings, including the more challenging Setting 1. These results highlight the robustness of `BaNk-UCB` in moderate to high-dimensional settings, where the benefits of adapting to local geometry become even more pronounced.

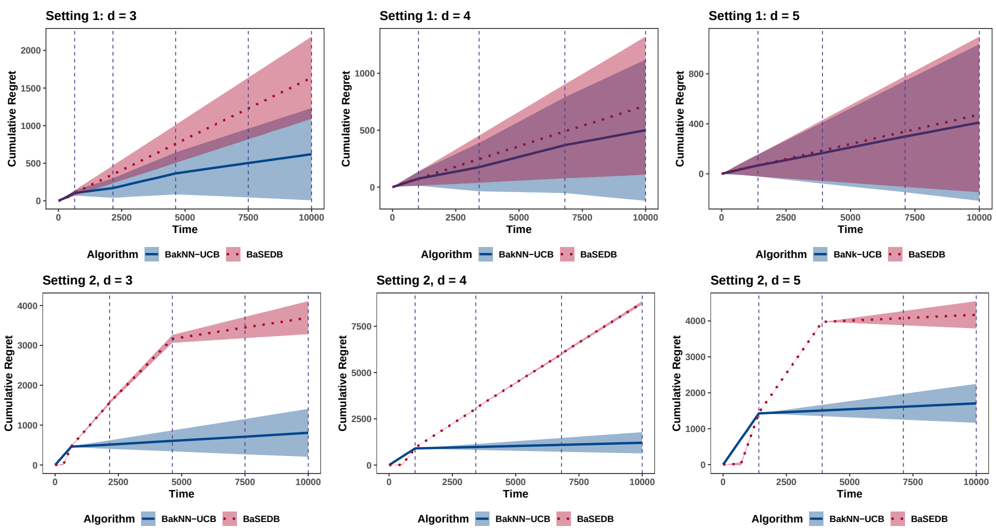

Figure 3: Average cumulative regret over 30 runs for BaSEDB and BaNk-UCB under Settings 1 and 2 with $d \in \{3, 4, 5\}$. Vertical dashed lines denote batch boundaries.