No-Regret Algorithms for Private Gaussian Process Bandit Optimization

Abhimanyu Dubey
Media Lab and Institute for Data, Systems and Society
Massachusetts Institute of Technology
dubeya@mit.edu

Abstract

The widespread proliferation of data-driven decision-making has ushered in a recent interest in the design of privacy-preserving algorithms. In this paper, we consider the ubiquitous problem of gaussian process (GP) bandit optimization from the lens of privacy-preserving statistics. We propose a solution for differentially private GP bandit optimization that combines a uniform kernel approximator with random perturbations, providing a generic framework to create differentially-private (DP) Gaussian process bandit algorithms. For two specific DP settings - joint and local differential privacy, we provide algorithms based on efficient quadrature Fourier feature approximators, that are computationally efficient and provably no-regret for popular stationary kernel functions. Our algorithms maintain differential privacy throughout the optimization procedure and critically do not rely explicitly on the sample path for prediction, making the parameters straightforward to release as well.

1 Introduction

Gaussian Process (GP) bandit optimization (Srinivas et al., 2010) is a sequential decision problem that has a variety of human-centered applications, e.g., clinical drug trials (Costabal et al., 2019; Park et al., 2013; Peterson et al., 2017), personalized shopping recommendations (Rohde et al., 2018; Zhou et al., 2019), news feed ranking (Agarwal et al., 2018; Letham & Bakshy, 2019; Vanchinathan et al., 2014). It is increasingly becoming desirable that algorithms interacting with such data maintain the privacy of the individuals whose information is used (Cummings & Desai, 2018).

GP bandit optimization involves learning a function f via repeated interaction in rounds. At any round $t = 1, 2, ...,$ the learner is presented with a *decision set* $\mathcal{D}_t \subset \mathbb{R}^d$ from which it must select an action $\mathbf{x}_t$ and obtain a random reward $y_t = f(\mathbf{x}_t) + \varepsilon_t$. The algorithm selects actions in order to minimize regret $\mathcal{R}(T) = \sum_t [\max_{\mathbf{x}\in\mathcal{D}_t} f(\mathbf{x}) - f(\mathbf{x}_t)]$. Algorithm design is focused on minimizing *pseudoregret* $\mathbb{E}[\mathcal{R}(T)]$. In deployment settings, each round corresponds to selecting a random user i_t. The decision set $\mathcal{D}_t$ is a representation of the user' behavior and y_t refers to the users response to $\mathbf{x}_t$. In this case, privacy refers to privacy with respect to both $(\mathcal{D}_t, y_t)$ (Shariff & Sheffet, 2018).

Provably no-regret algorithms with differential privacy have been proposed for multi-armed bandits (Tossou & Dimitrakakis, 2015; Mishra & Thakurta, 2015), linear contextual bandits (Shariff & Sheffet, 2018; Agarwal & Singh, 2017) and tabular RL (Vietri et al., 2020a). For GP optimization, however, the problem is more challenging. Most applications assume f to lie in a (potentially) infinite-dimensional reproducing kernel Hilbert space (RKHS), and standard techniques for introducing privacy are inapplicable due to the *curse of dimensionality* (Liu & Guillas, 2017; Meeds & Welling, 2014): the posterior mean and variance for these methods require storing the sample path $(\mathbf{x}_t, y_t)_t$, and are $\Omega(t)$ to evaluate. Moreover, as the learnt function itself is dependent on the sample path (containing sensitive data), privatized release of the function is also a challenge (Smith et al., 2016). In this paper, we propose algorithms that guarantee differential privacy with respect to continual observation during optimization, and also the private release of learnt parameters.

Contributions. First, we propose a generic framework (and regret bound) for GP bandits that utilizes a finite-dimensional $\epsilon-$uniform approximation of infinte-

Proceedings of the 24[th] International Conference on Artificial Intelligence and Statistics (AISTATS) 2021, San Diego, California, USA. PMLR: Volume 130.

dimensional kernels and integrates random perturbations to the GP posterior, allowing for various no-regret private GP algorithms based on the kernel approximation method and privacy guarantee required.

Next, In the joint differentially private (JDP) setting (Defn. 6), we propose a novel `GP-UCB` algorithm (Alg. 1) for stationary kernels admitting a decomposable Fourier transform (Assumption 1) that satisfies (α, β)-JDP while obtaining $\widetilde{\mathcal{O}}(\sqrt{T\gamma_T/\alpha})$[1] pseudoregret. This bound matches (up to logarithmic factors) the lower bound for isotropic kernels (Scarlett et al., 2017), and admits an identical dependence on α as linear bandits (Shariff & Sheffet, 2018). Thirdly, inspired by the recent interest in locally DP methods (Bebensee, 2019), we present a stronger variant of JDP dubbed locally-joint differential privacy (Defn. 7) for sequential decision-making that imposes constraints on each user's data separately. We propose an algorithm that achieves $(\alpha, \beta)-$local JDP with $\widetilde{\mathcal{O}}(T^{3/4}\sqrt{\gamma_T/\alpha})$ pseudoregret. We conjecture that the constraints from local JDP necessitate the $\mathcal{O}(T^{1/4})$ departure from typical near-optimal regret (Remark 7).

Our approach can be coarsely summarized with two steps - we first project f from its (infinite-dimesional) RKHS into a finite-dimensional approximating RKHS, following which, we directly perturb the posterior mean and variance of the resulting GP (in the approximating space) to ensure privacy without the curse of dimensionality, providing *provably no-regret* algorithms for private GP bandit optimization. Our approach additionally avoids the parameter release problem (Smith et al., 2016; Kusner et al., 2015) since we do not explicitly store the sample path for prediction, and rely instead on cumulative sums (Remark 1).

Organization. We first discuss crucial related work and introduce necessary notation and preliminaries, subsequent to which we introduce our general framework for `GP-UCB` using noisy approximate features. We discuss quadrature Fourier features and present our algorithm and its associated regret bounds. Next, we discuss the two models of privacy studied, and present privacy mechanisms. We defer proofs to the appendix, and present concise proof sketches in the main paper.

2 Related Work

Gaussian Process Bandits. Gaussian Processes (Williams & Rasmussen, 2006) have been widely used for the bandit optimization of unknown functions in an RKHS. The seminal work of Srinivas et al. (2010) introduced the nonparameteric `GP-UCB` algorithm, that introduced contextual-bandit style confidence bounds for optimisation in infinite-dimensional RKHSes. A variant of the *expected improvement* decision rule (Močkus, 1975) was proposed via the `GP-EI` algorithm (Snoek et al., 2012). By a stronger martingale analysis, Chowdhury & Gopalan (2017) achieve the `IGP-UCB` algorithm, that improves `GP-UCB` regret by a factor of $\mathcal{O}(\ln^{3/2} T)$. For a family of isotropic squared-exponential d-dimensional kernels, Scarlett et al. (2017) establish lower bounds on the achievable regret of $\Omega(\sqrt{T(\log T)^{d+2}})$, which matches (ignoring polylogarithmic factors) the $\widetilde{\mathcal{O}}(\sqrt{T})$ rate achieved by `IGP-UCB` and `GP-UCB`. Our work relies on the research in approximate methods for kernel approximation, which has seen a lot of recent interest. The seminal work of Rahimi & Recht (2008) proposed random Fourier features (RFF) by a Monte-Carlo approximation of the Fourier basis, with additional work establishing finite-sample convergence rates (Avron et al., 2017). We propose a noisy variant of the more efficient quadrature Fourier features (QFF) (Munkhoeva et al., 2018) that have been previously employed in GP optimization with success (Mutny & Krause, 2018). An alternative approach based on sampling fewer points from the algorithm's history based on matrix sketching has been proposed in Calandriello et al. (2019).

Differentially-Private Bandit Learning. Differentially private (DP) methods for bandit optimisation have received significant attention recently. For the multi-armed bandit case, UCB and Thompson sampling algorithms have been proposed for pure-DP (Mishra & Thakurta, 2015), with subsequent improvements (Tossou & Dimitrakakis, 2015). For the contextual linear bandit, Shariff & Sheffet (2018) introduce an algorithm that utilizes matrix perturbations that our work effectively generalizes to infinite-dimensional stationary GPs. Note that this algorithm is inapplicable for general GPs as it assumes that the features are finite-dimensional. See Basu et al. (2020) for a summary of regret bounds for private multi-armed bandits. For Gaussian process bandits and Bayesian Optimisation (BO), Kusner et al. (2015) consider the problem of *releasing* GP parameters *after* optimization under differential privacy constraints, by analysing the sensitivity of the final parameters. Our work handles a more challenging setting, where parameters must be private *throughout* the optimisation process. An application of DP to the Gaussian process regression problem was studied in the work of Smith et al. (2016), however with no regret guarantees.

3 Preliminaries

Notation. We denote vectors by lowercase solid characters, i.e., $\mathbf{x}$ and matrices by uppercase solid charac-

[1] γ_T is the *maximum information gain*, see Definition 1.

ters $\mathbf{X}$. We denote the $\mathbf{\Sigma}-$ellipsoid norm of a vector $\mathbf{x}$ as $\|x\|_{\mathbf{\Sigma}} = \sqrt{\mathbf{x}^\top \mathbf{\Sigma} \mathbf{x}}$, that a symmetric matrix $\mathbf{A}$ is PSD by $\mathbf{A} \succcurlyeq \mathbf{0}$, and the Löwner ordering of symmetric PSD matrices by $\mathbf{A} \succcurlyeq \mathbf{B}$, which implies $\mathbf{A} - \mathbf{B} \succcurlyeq \mathbf{0}$.

GP Bandit Optimization. We consider the problem of sequential reward maximization under a fixed but unknown reward function $f : \mathcal{D} \to \mathbb{R}$ over a (potentially infinite) set of actions (arms) $\mathcal{D} \subset \mathbb{R}^d$. The problem proceeds in rounds $t = 1, 2, ..., T$ where, in each round, the objective is to select an action $\mathbf{x}_t \in \mathcal{D}_t$ and obtain a reward $y_t = f(\mathbf{x}_t) + \varepsilon_t$ such that the cumulative reward $\sum_{t\in[T]} y_t$ is maximized depending on the history $(\mathbf{x}_\tau, y_\tau)_{\tau<t}$, and ε_t is sampled from a zero-mean sub-Gaussian distribution with parameter λ. Gaussian Process (GP) modeling proposes to use a Gaussian likelihood model for observations and a GP prior for the uncertainty over f. A Gaussian Process (GP) over $\mathcal{D}$, denoted by $\mathrm{GP}(\mu(\cdot), k(\cdot,\cdot))$ is a collection of random variables $(f(\mathbf{x}))_{\mathbf{x}\in\mathcal{D}}$ such that every finite subset of variables $(f(\mathbf{x}_\tau))_{\tau=1}^t$ is jointly Gaussian with mean $\mathbb{E}[f(\mathbf{x}_\tau)] = \mu(\mathbf{x}_\tau)$ and covariance $\mathbb{E}[(f(\mathbf{x}_\tau) - \mu(\mathbf{x}_\tau))(f(\mathbf{x}_{\tau'}) - \mu(\mathbf{x}_{\tau'}))] = k(\mathbf{x}_\tau, \mathbf{x}_{\tau'})$, $\tau, \tau' \in [t]$ where $k(\cdot,\cdot)$ is the kernel function associated with the reproducing kernel Hilbert space (RKHS) $\mathcal{H}_k(\mathcal{D})$ in which we assume f has norm at most B, i.e., $\|f\|_k \leq B$. We use an initial prior distribution $\mathrm{GP}(0, \rho^2 k(\cdot,\cdot))$ for some $\rho > 0$. Consequently it is also assumed that the noise samples ε_t are drawn from $\mathcal{N}(0, \lambda\rho^2)$[2]. We then obtain that the observed samples $\mathbf{y}_t = (y_\tau)_{\tau<t}$ and $f(\mathbf{x})$ are jointly Gaussian given $\mathbf{X}_t = (\mathbf{x}_\tau)_{\tau<t}$,

$$\begin{bmatrix} f(\mathbf{x}) \\ \mathbf{y}_t \end{bmatrix} \sim \mathcal{N}\left(\mathbf{0}, \begin{bmatrix} \rho^2 k(\mathbf{x},\mathbf{x}) & \rho^2 \mathbf{k}_t(\mathbf{x})^\top \\ \rho^2 \mathbf{k}_t(\mathbf{x}) & \rho^2(\mathbf{K}_t + \lambda\mathbf{I}) \end{bmatrix}\right). \quad (1)$$

Where $\mathbf{K}_t = (k(\mathbf{x}_\tau, \mathbf{x}_{\tau'}))_{\tau,\tau'}^{t,t}$ is the matrix of kernel evaluations at time t, and $\mathbf{k}_t(\mathbf{x}) = [k(\mathbf{x}_1, \mathbf{x}), ..., k(\mathbf{x}_t, \mathbf{x})]^\top$ is the vector of kernel evaluations of any input $\mathbf{x}$. Conditioned on $(\mathbf{x}_\tau, y_\tau)_{\tau<t}$, the posterior mean and variance of f is given as,

$$\mu_t(\mathbf{x}) = k_t(\mathbf{x})^\top (\mathbf{K}_t + \lambda\mathbf{I})^{-1} \mathbf{y}_t, \quad (2)$$
$$\sigma_t^2(\mathbf{x}) = \left(k(\mathbf{x},\mathbf{x}) - \mathbf{k}_t(\mathbf{x})^\top (\mathbf{K}_t + \lambda\mathbf{I})^{-1} \mathbf{k}_t(\mathbf{x})\right). \quad (3)$$

The kernel $k(\cdot,\cdot)$ additionally admits a representation in terms of its feature space Φ such that $k(\mathbf{x},\mathbf{x}') = \Phi(\mathbf{x})^\top \Phi(\mathbf{x})$, where $\Phi : \mathbb{R}^d \to \mathbb{R}^m$ is the feature embedding. This provides an alternative representation of the posterior mean and variance,

$$\mu_t(\mathbf{x}) = (\mathbf{\Sigma}_t + \lambda\mathbf{I})^{-1} \Phi(\mathbf{X})^\top \mathbf{y}_t, \quad (4)$$
$$\sigma_t^2(\mathbf{x}) = \rho^2 \Phi(\mathbf{x})^\top (\mathbf{\Sigma}_t + \lambda\mathbf{I})^{-1} \Phi(\mathbf{x}), \text{ for} \quad (5)$$

$\mathbf{\Sigma}_t = \Phi(\mathbf{X}_t)^\top \Phi(\mathbf{X}_t), \Phi(\mathbf{X}_t) = [\Phi(\mathbf{x}_1)^\top, ..., \Phi(\mathbf{x}_{t-1})^\top]^\top$.

[2]The algorithm only requires ε_t to be λ-sub-Gaussian, i.e., the *agnostic* setting (Srinivas et al., 2010).

Φ can potentially be infinite-dimensional (e.g., for squared-exponential k), and hence this representation is not applicable to many popular kernel families. The regret achieved by existing algorithms depends on the *maximum information gain*, a quantity that depends on the covariance structure of the feature space.

Definition 1 (Information Gain (Srinivas et al., 2010)). *For $y_t = f(\mathbf{x}_t) + \varepsilon_t$, let $A \subset \mathcal{X}$ be a finite subset such that $|A| = T$. Let $\mathbf{y}_A = \mathbf{f}_A + \boldsymbol{\varepsilon}_A$ where $\mathbf{f}_A = (f(\boldsymbol{x}_i))_{\boldsymbol{x}_i \in A}$ and $\boldsymbol{\varepsilon}_A \sim \mathcal{N}(0, \rho^2)$. The information gain is $\gamma_T \triangleq \max_{A\subset\mathcal{X}:|A|=T} H(\mathbf{y}_A) - H(\mathbf{y}_A|f)$, where $H(\cdot)$ is the entropy of a random variable. For linear k, $\gamma_T = \mathcal{O}(d \log T)$. For RBF k, $\gamma_T = \mathcal{O}((\log T)^{d+1})$. For Matérn k with $\nu > 1$, $\gamma_T = \mathcal{O}(T^{\frac{d(d+1)}{2\nu+d(d+1)}}(\log T))$.*

Differential Privacy (DP). Differential Privacy (Dwork & Roth, 2014) is a cryptographically secure framework to introduce privacy, widely prevalent in machine learning. Let algorithm $\mathcal{A} : \mathcal{X} \to \mathcal{Y}$, let $X, X' \in \mathcal{X}$ operate on samples from $\mathcal{X}$ producing outputs in $\mathcal{Y}$. An algorithm is $(\alpha, \beta)-$differentially private if for any two inputs $X, X' \in \mathcal{X}$ that differ in only one entry and any $\mathcal{S} \subset \mathcal{Y}$,

$$\mathbb{P}(\mathcal{A}(X) \in \mathcal{S}) \leq e^\alpha \mathbb{P}(\mathcal{A}(X') \in \mathcal{S}) + \beta. \quad (6)$$

In the continual observation setting of sequential decision-making, this would imply that the algorithm be private with respect to all values $(\mathbf{x}_\tau, y_\tau)_{\tau=1}^T$ at each $t \in [T]$. However, as demonstrated in Shariff & Sheffet (2018), any algorithm DP with respect to $(\mathbf{x}_t, y_t)$ at the instance t provably incurs $\Omega(T)$ regret. Therefore we adopt the notion of *joint* differential privacy, which does not require privacy with respect to the inputs $(\mathbf{x}_t, y_t)$ at each instant $t \in [T]$ (Section 5.1). We additionally consider the stronger notion of *locally* joint DP, which additionally requires that the algorithm cannot access $(\mathbf{x}_\tau, y_\tau)_{\tau<t}$ directly (Section 5.2).

4 Noisy Proximal Features & GP-UCB

The primary challenge in creating differentially-private algorithms for *bandit estimation* in arbitrary RKHSes is the curse of dimensionality - the two central quantities μ_t and σ_t^2 both require the point-wise kernel evaluations ($\mathbf{k}_t(\mathbf{x})$) and the kernel Gram matrix ($\mathbf{K}_t$) at all times, potentially requiring $\mathcal{O}(\sqrt{t})$ noise in order to preserve privacy. In this paper, we tackle this hurdle by optimizing f under a surrogate RKHS $\mathcal{F}_m$ that of finite dimension m instead of the original (potentially infinite-dimensional) RKHS $\mathcal{H}_k$. To ensure a reasonable bound on the regret, we require that $\mathcal{F}_m$ approximates $\mathcal{H}_k$ closely, as formalized below.

Definition 2 (Uniform Approximation). *Let $k : \mathcal{D} \times \mathcal{D} \to \mathbb{R}$ be a stationary kernel with associated RKHS*

$\mathcal{H}_k$, *and* $\Phi : \mathcal{D} \to \mathbb{R}^m$. *Then* Φ *ϵ-uniformly approximates k iff* $\sup_{\mathbf{x},\mathbf{x}' \in \mathcal{D}} |k(\mathbf{x},\mathbf{x}') - \Phi(\mathbf{x})^\top \Phi(\mathbf{x})| \leq \epsilon$. *The corresponding* ***approximating space*** *defined by* Φ *is given by* $\mathcal{F}_m(\Phi) \triangleq \left\{ f(\cdot) = \boldsymbol{\theta}^\top \Phi(\cdot) | \boldsymbol{\theta} \in \mathbb{R}^m \right\}$.

Therefore, if $\mathcal{F}_m$ (resp. Φ) can approximate $\mathcal{H}_k$ without many features, one can devise an *approximate* Gaussian process algorithm directly using Φ.

$$\mathbf{G}_t = \Phi(\mathbf{X}_t)^\top \Phi(\mathbf{X}_t) + \lambda \mathbf{I}, \mathbf{u}_t = \mathbf{G}_t^{-1} \Phi(\mathbf{X}_t)^\top \mathbf{y}_t. \quad (7)$$

These parameters allow us to obtain the posterior mean $\mu_t(\mathbf{x}) = \mathbf{u}_t^\top \Phi(\mathbf{x})$ and variance $\sigma_t^2(\mathbf{x}) = \rho^2 \|\Phi(\mathbf{x})\|^2_{\mathbf{G}_t^{-1}}$. However, these parameters are obviously not differentially private with respect to the sequences $(\mathbf{X}_t, \mathbf{y}_t)$. An efficient way to achieve privacy is to ensure that at each instant t, $(\mathbf{G}_t, \mathbf{u}_t)$ are differentially-private with respect to the sequence $(\mathbf{x}_\tau, y_\tau)_{\tau < t}$ (Shariff & Sheffet, 2018). This can be achieved by carefully perturbing $(\mathbf{G}_t, \mathbf{u}_t)$ with random noise $(\mathbf{H}_t, \mathbf{h}_t)$ to create differentially-private parameters. While the exact form of $\mathbf{H}_t, \mathbf{h}_t$ will be specified by the nature of privacy (see Section 5), we can represent a variety of noise models by spectral bounds, summarized by the following abstraction.

Definition 3 (Spectral Bounds on Noise)**.** *For a sequence of perturbations* $(\mathbf{H}_t)_{t=1}^T$ *and* $(\mathbf{h}_t)_{t=1}^T$, *the bounds* $0 < \lambda_{\min} \leq \lambda_{\max}$ *are* $(\zeta/2T)-$*accurate if with probability at least* $1 - \zeta/2T$, *for each* t *in* $[T]$*:*

$$\|\mathbf{H}_t\| \leq \lambda_{\max}, \|\mathbf{H}_t^{-1}\| \leq 1/\lambda_{\min}, \|\mathbf{h}_t\|_{\mathbf{H}_t^{-1}} \leq \kappa.$$

Let us use the shorthand $\mathbf{G}_t = \mathbf{\Sigma}_t + \lambda \mathbf{I}$, where $\mathbf{\Sigma}_t = \Phi(\mathbf{X}_t)^\top \Phi(\mathbf{X}_t)$. The perturbed $\mathbf{\Sigma}_t$ and $\mathbf{u}_t$ are given as $\widetilde{\mathbf{\Sigma}}_t = \mathbf{\Sigma}_t + \mathbf{H}_t, \widetilde{\mathbf{u}}_t = \mathbf{u}_t + \mathbf{h}_t$ for any sequence $(\mathbf{H}_t, \mathbf{h}_t)$.

4.1 GP-UCB with Noisy Proximal Features

Our algorithm is built on the GP-UCB algorithm (Srinivas et al., 2010) that constructs a confidence ellipsoid around the posterior $\widetilde{\mu}_t$ such that the function f lies within the confidence ellipsoid with high probability. The key observation, is that we do not need to optimize for f directly. Given an ϵ-uniformly approximating feature Φ (resp. $\mathcal{F}_m$), then the following result guarantees the existence of a function close to f in $\mathcal{F}_m$.

Lemma 1 (Existence of Proximal Space (Lemma 4 of Mutny & Krause (2018)))**.** *Let k be a kernel defining the RKHS $\mathcal{H}_k$ and $f \in \mathcal{H}_k$, such that the spectral characteristic function is bounded by B. Assuming that the defining points of f come from the set $\mathcal{D}$, let $\mathcal{F}_m$ be an approximating space with a mapping Φ such that this mapping is an ϵ-approximation to the kernel k. Then there exists $\widehat{\mu} \in \mathcal{F}_m$ (with corresponding feature $\widehat{\boldsymbol{\theta}}$ such that $\widehat{\mu}(\mathbf{x}) = \langle \widehat{\boldsymbol{\theta}}, \Phi(\mathbf{x}) \rangle$), such that* $\sup_{\mathbf{x} \in \mathcal{D}} |\widehat{\mu}(\mathbf{x}) - f(\mathbf{x})| \leq B\epsilon$.

Algorithm 1 APPROXIMATE GP-UCB

Input: m, Φ that ϵ-uniformly approximates k.
PRIVATIZER **Initialize**: $\mathbf{\Sigma}_1 = \mathbf{0}, \mathbf{u}_1 = \mathbf{0}$.
for round $t = 1, 2, ..., T$ **do**
  SERVER:
  Receive $\mathcal{D}_t$ from environment.
  Receive $\widetilde{\mathbf{\Sigma}}_t = \mathbf{\Sigma}_t + \mathbf{H}_t, \widetilde{\mathbf{u}}_t = \mathbf{u}_t + \mathbf{h}_t \leftarrow$PRIVATIZER.
  Set $\mathbf{V}_t \leftarrow \widetilde{\mathbf{\Sigma}}_t + \lambda \mathbf{I}, \widetilde{\boldsymbol{\theta}}_t \leftarrow \mathbf{V}_t^{-1} \widetilde{\mathbf{u}}_t$.
  Compute β_t based on Theorem 1.
  Select $\mathbf{x}_t \leftarrow \arg\max_{\mathbf{x} \in \mathcal{D}_t} \langle \widetilde{\boldsymbol{\theta}}_t, \Phi(\mathbf{x}) \rangle + \beta_t \|\Phi(\mathbf{x})\|_{\mathbf{V}_t^{-1}}$.
  Play arm $\mathbf{x}_t$ and obtain y_t.
  Send $(\Phi(\mathbf{x}_t), y_t) \rightarrow$PRIVATIZER.
  PRIVATIZER:
  Sending parameters:
  Obtain $\mathbf{H}_t, \mathbf{h}_t$ based on Section 5.
  Send $\widetilde{\mathbf{\Sigma}}_t = \mathbf{\Sigma}_t + \mathbf{H}_t, \widetilde{\mathbf{u}}_t = \mathbf{u}_t + \mathbf{h}_t \rightarrow$SERVER
  Updating parameters:
  Receive $\mathbf{x}_t, y_t \leftarrow$SERVER.
  Securely update $\mathbf{\Sigma}_{t+1} \leftarrow \mathbf{\Sigma}_t + \Phi(\mathbf{x})\Phi(\mathbf{x})^\top$ (Sec. 5).
  Securely update $\mathbf{u}_{t+1} \leftarrow \mathbf{u}_t + y_i \Phi(\mathbf{x})$ (Section 5).
end for

Lemma 1 implies that there exists a fixed point $\widehat{\mu} \in \mathcal{F}_m$ such that $\sup_{\mathbf{x} \in \mathcal{D}} |\widehat{\mu}(\mathbf{x}) - f(\mathbf{x})| \leq B\epsilon$. This implies that the regret incurred at any instant t when optimizing for f is at most $B\epsilon$ larger than the regret obtained when optimizing for $\widehat{\mu}$. We therefore optimize directly in the surrogate space $\mathcal{F}_m$ to learn $\widehat{\mu}$. GP-UCB with noisy approximate features selects, for a sequence $(\beta_t)_{t=1}^T$, the action $\mathbf{x}_t \in \mathcal{D}_t$ determined as:

$$\mathbf{x}_t = \arg\max_{\mathbf{x} \in \mathcal{D}_t} \widetilde{\mu}_t(\mathbf{x}) + \beta_t^{1/2} \widetilde{\sigma}_t(\mathbf{x}). \quad (8)$$

The sequence $(\beta_t)_{t=1}^T$ is chosen such that $\widetilde{\mu}_t(\mathbf{x})$ is close to $\widehat{\mu}(\mathbf{x})$ with high probability. To accomplish this, we present the central result as follows.

Theorem 1 (β_t concentration)**.** *Let $\lambda_{\min}, \lambda_{\max}$ and κ be $(\zeta/2T)$ -accurate and regularizers $\mathbf{H}_t \succcurlyeq 0$ $\forall t \in [T]$ are PSD. Let $\widehat{\mu}$ be a function in the RKHS $\mathcal{F}_m$ that ϵ-approximates $f \in \mathcal{H}_k$ (Lemma 1). Then, with probability at least $1 - \zeta/2$, for any $\mathbf{x} \in \mathcal{D}$ we have for each $t \in [T]$ simultaneously,*

$$|\widehat{\mu}(\mathbf{x}) - \widetilde{\mu}_t(\mathbf{x})| \leq \widetilde{\sigma}_t(\mathbf{x}) \Bigg(B\sqrt{\frac{\lambda_{\max}}{\rho^2} + 1} + \frac{tB\epsilon}{\rho\sqrt{\lambda_{\min}}} + \frac{\kappa}{\rho} + \sqrt{\log\det\left(\frac{\widetilde{\mathbf{\Sigma}}_t + \lambda \mathbf{I}}{\lambda + \lambda_{\min}}\right) + 2\ln\frac{2}{\zeta}} \Bigg).$$

The sequence $(\beta_t^{1/2})_{t=1}^T$ is chosen as the multiplicative factor of $\widetilde{\sigma}_t(\mathbf{x})$, i.e., $|\widehat{\mu}(\mathbf{x}) - \widetilde{\mu}_t(\mathbf{x})| \leq \beta_t^{1/2} \widetilde{\sigma}_t(\mathbf{x})$.

The complete algorithm is summarized in Algorithm 1, and prooof is presented in the appendix. Note that we describe the algorithm abstractly for any ϵ-uniformly approximating feature Φ with dimensionality m, and Theorem 1 (and the regret bound) hold

for any such feature approximation that also satisfies $\sup_{\mathbf{x}\in\mathcal{D}}\|\Phi(\mathbf{x})\| \leq 1$. The algorithm is described in two separate entities, the SERVER and the PRIVATIZER, where the privatizer entity has access to the raw rewards and contexts, and the server only obtains privatized versions of the statistics. We now present specific Φ such that we obtain an efficient algorithm.

Remark 1 (Parameter Release). *$\widetilde{\mu}$ can be determined entirely only with the parameters $\widetilde{\mathbf{\Sigma}}_t, \widetilde{\mathbf{u}}_t$ (Equation 4). If the noise variables $\mathbf{H}_t, \mathbf{h}_t$ are constructed such that the resulting parameters satisfy privacy constraints (see next section), these parameters are by design differentially private and hence $\widetilde{\mu}$ can be released without using the sample path $(\mathbf{x}_t, y_t)_{t\leq T}$.*

4.2 Noisy Quadrature Fourier Features

Bochners' theorem (Bochner, 1933) states that there exists an integral form for stationary k, where the integrand is a product of identical features of the inputs:

$$k(\mathbf{x}-\mathbf{y}) = \int_{\Omega} \begin{pmatrix}\sin(\boldsymbol{\omega}^\top\mathbf{x})\\ \cos(\boldsymbol{\omega}^\top\mathbf{x})\end{pmatrix}^\top \begin{pmatrix}\sin(\boldsymbol{\omega}^\top\mathbf{y})\\ \cos(\boldsymbol{\omega}^\top\mathbf{y})\end{pmatrix} p(\boldsymbol{\omega}). \quad (9)$$

When the above integral is approximated by a Monte-Carlo average, we obtain the powerful Random Fourier Features (RFF, (Rahimi & Recht, 2008)) approximation. Random Fourier features, while approximating a variety of kernels, are not efficient since $\epsilon_{\text{RFF}} = \mathcal{O}(m^{-1/2})$, requiring prohibitively many features m for our purpose. We consider Quadrature Fourier Features (QFF, Dao et al. (2017)), a stronger approximation that is motivated by numerical integration, and allows ϵ to decay *exponentially* in m. To define QFF, we require that the kernel be Fourier decomposable.

Assumption 1 (Decomposability of k). *Let k be a stationary kernel defined on $\mathbb{R}^d \times \mathbb{R}^d$ and $k(\mathbf{x},\mathbf{y}) \leq 1\ \forall\ \mathbf{x},\mathbf{y}\in\mathbb{R}^d$ with a Fourier transform that decomposes product-wise, i.e., $p(\boldsymbol{\omega}) = \pi_{j=1}^d p_j(\boldsymbol{\omega}_j)$*[3].

Definition 4 (Quadrature Fourier Features). *Let $\mathcal{D} = [0,1]^d$, and $\mathbf{x},\mathbf{y}\in\mathcal{D}$. Fix $m = (\bar{m})^d$ for some $\bar{m} > 1$, and let $p(\boldsymbol{\omega}) = \exp\left(-\sum_{i=1}^d \frac{\omega_i^2\nu_i^2}{2}\right)$ be the Fourier transform of k. The QFF features $\Phi(\mathbf{x})$ is defined as:*

$$\Phi(\mathbf{x})_i = \begin{cases}\sqrt{\Pi_{j=1}^d {}^1\!/_{\nu_i} Q(\omega_{i,j})\cos(\boldsymbol{\omega}_i^\top\mathbf{x})} & \text{if } i \leq \frac{m}{2}\\ \sqrt{\Pi_{j=1}^d {}^1\!/_{\nu_i} Q(\omega_{m-i,j})\sin(\boldsymbol{\omega}_{m-i}^\top\mathbf{x})} & o.w.\end{cases}$$

Φ is hence of dimensionality $2m$, and $Q(\omega_{i,j}) = \frac{2^{m-1/2}m!\sqrt{\pi}}{\nu_j m^2 H_{m-1}(\omega_{i,j})}$ and H_t is the t^{th} Hermite polynomial. The set $(\boldsymbol{\omega}_i)_{i=1}^m$ is the Cartesian product of $\{\bar{\omega}_j\}_{j=1}^{\bar{m}}$, where each element $\bar{\omega}_i\in\mathbb{R}$ and is a zero of the i^{th} Hermite polynomial. See Hildebrand (1987) for details.

[3] This is satisfied for commonly-used kernels, e.g., squared exponential. Matérn kernels are decomposable when $d = 1$. For $d > 1$, Mutny & Krause (2018) present a modified Matérn kernel that can be used a surrogate.

Theorem 2 (QFF Error (Mutny & Krause, 2018)). *Let $\Phi(\cdot), m$ and $\bar{m}$ be as defined above, $\mathcal{D} = [0,1]^d$ and $\nu = \min_i \nu_i$. Then,*

$$\sup_{\mathbf{x},\mathbf{y}\in\mathcal{D}} |k(\mathbf{x},\mathbf{y}) - \Phi(\mathbf{x})^\top\Phi(\mathbf{y})| \leq d2^{d-1}\sqrt{\tfrac{\pi}{2}}\tfrac{1}{\bar{m}^{\bar{m}}}\left(\tfrac{e}{4\nu^2}\right)^{\bar{m}}.$$

Remark 2. *Theorem 2 implies that the error ϵ decays exponentially in m when $m > \nu^{-2}$. Mutny & Krause (2018) evaluate this phase transition in detail, where a break is observed in simulations. For any known kernel k however, we can simply select $m > \nu^{-2}$ to ensure decay. Moreover, for additive kernels, it can be demonstrated that the dependence is exponential in the effective dimension, which can be much less than d.*

By adding appropriate $(\mathbf{H}_t, \mathbf{h}_t)$ to maintain privacy, we obtain *noisy* quadrature Fourier features (NQFF).

Definition 5 (Noisy Quadrature Fourier Features (NQFF)). *Let $\Phi : \mathbb{R}^d \to \mathbb{R}^m$ be an $\epsilon-$approximation QFF to the stationary kernel k, and $(\mathbf{H}_t, \mathbf{h}_t)_{t=1}^T$ be a sequence of perturbations. Then, at any instant t, we can define the noisy QFF as $\widetilde{\Phi}(\mathbf{X}_t) = \begin{bmatrix}\Phi(\mathbf{X}_t) & \mathbf{0}\\ \mathbf{0} & \mathbf{\Gamma}_t\end{bmatrix}$, where $\mathbf{\Gamma}_t^\top\mathbf{\Gamma}_t = \mathbf{H}_t$ (i.e., eigendecomposition of $\mathbf{H}_t$).*

4.3 Regret Analysis

We first present the regret bound for `GP-UCB` with generic ϵ-uniformly approximating features Φ with dimensionality m. Note that this bound is applicable to any approximation technique that satisfies $\sup_{\mathbf{x}\in\mathcal{D}}\|\Phi(\mathbf{x})\| \leq 1$, and suitable $\lambda_{\min}, \lambda_{\max}$ and κ.

Theorem 3 (Regret Bound). *Let k be a stationary kernel with the associated RKHS $\mathcal{H}_k$, and $\mathcal{F}_m$ be an RKHS with feature $\Phi(\cdot)$ of dimensionality m, that ϵ-uniformly approximates every $f\in\mathcal{H}_k$ when $\|f\| \leq B$. Furthermore, assume $\lambda_{\min}, \lambda_{\max}$ and κ such that they are $(\zeta/2T)$ -accurate and all regularizers $\mathbf{H}_t \succcurlyeq 0\ \forall t\in[T]$ are PSD. Then for $(\beta_t)_{t=1}^T$ chosen by Theorem 1, `GP-UCB` with noisy proximal features obtains the following cumulative regret with probability at least $1-\zeta$:*

$$\mathfrak{R}(T) \leq 2\sqrt{T\beta_T\gamma_T} + \frac{2T^3\sqrt{\beta_T\epsilon}}{3\rho} + 2TB\epsilon.$$

Where γ_T is the maximum information gain (Defn. 1).

Proof (Sketch). The first key observation is to bound the per-round regret from f with the per-round regret from optimizing $\widehat{\mu}$. Next, we utilize standard techniques from the analysis of `GP-UCB` to bound the regret in terms of β_t and $\widetilde{\sigma}_t$ (using Theorem 1 twice),

and finally provide a bound on $\widetilde{\sigma}_t$ in terms of the true information gain γ_T. Summing over all rounds and manipulating proves the result. ■

By replacing β_T in the result, and manipulating terms, we can conclude that if we have Φ such that $\epsilon = \mathcal{O}(\exp(-m))$ and $m = \mathcal{O}(\text{polylog}(T))$, then we can obtain sublinear regret. Using the properties of QFF from earlier, we can obtain a specific bound as follows.

Corollary 1. *Fix $m = 2(6\log T)^d$ and let k be any kernel that obeys Assumption 1. Algorithm 1 run with m-dimensional NQFF and noise $\mathbf{H}_t, \mathbf{h}_t$ that are $\zeta/2T$-accurate with constants $\lambda_{\max}, \lambda_{\min}$ and κ obtains with probability at least $1-\zeta$, cumulative pseudoregret:*

$$\mathfrak{R}(T) = \mathcal{O}\left(\sqrt{T\gamma_T}\left(\tfrac{B\sqrt{\lambda_{\max}}}{\rho} + \sqrt{\log\tfrac{1}{\zeta} + (\log T^6)^{d+1}} + \tfrac{\kappa}{\rho}\right)\right).$$

Proof (Sketch). By Theorem 2, we can coarsely bound the approximation error by setting $\bar{m} = \log T^6$ to obtain $\epsilon = \mathcal{O}(T^{-6})$. Substituting this in Theorem 3 provides us with the final result. ■

Remark 3 (Selection of m)**.** *Note that the analysis presents a bound in terms of the information gain of the true kernel k, and hence requires $m = 2(\log T^6)^d$ features. However, an alternate technique will be to bound the information gain of $\tilde{k}$, which can subsequently be bound with a term of $\mathcal{O}(\sqrt{m\log T})$. In this case, setting $m = 2(\log T^3)^d$ suffices for no-regret learning, however the obtained regret is (coarsely) $\mathcal{O}(\sqrt{T}(\log T)^{d+1})$, which can be loose if $\gamma_T = o((\log T)^{d+1})$ (e.g., when k is low-rank).*

Remark 4 (Feasibility of Kernel Approximations)**.** *The current framework requires $\epsilon = o(T^{-4})$ with $m = \mathcal{O}(\text{polylog}(T))$ to obtain a no-regret algorithm. Random Fourier Features, while capable of approximating a variety of stationary kernels, decay with $\epsilon = \mathcal{O}(m^{-1/2})$ which makes them infeasible. For finite-dimensional $\mathcal{H}_k$, the results manifestly hold with $\epsilon = 0$.*

Remark 5 (Unknown T)**.** *When T is unknown, we can use a doubling scheme to calculate m and ϵ. To calculate ϵ, we assume $T = 1$ for the first round, then assume $T = 2$ for the next, and then assume $T = 4$ for the next 2 rounds, $T = 8$ for the next 4 rounds and so on, and set $\epsilon = \mathcal{O}(t^{-5})$, for instance, within each "period" of length t between doubling of T to calculate m. We see that the regret is at most $\widetilde{\mathcal{O}}(\sqrt{t})$ for this period. Since there are at most $\mathcal{O}(\log T)$ such periods, and $t \le T$, the total regret is $\mathcal{O}((\log T)\sqrt{T})$.*

5 GP-UCB with Differential Privacy

We now present the mechanism to ensure Algorithm 1 is differentially private. Proceeding with the standard definition of differential privacy (Equation 6) for the streaming setting, however, is infeasible (i.e., leading to linear regret, see Claim 13 of Shariff & Sheffet (2018)). We therefore work with a modified notion of privacy that is the standard for sequential decision-making (Shariff & Sheffet, 2018; Vietri et al., 2020b).

Algorithm 2 PRIVATIZER under JDP

Initialize: Binary tree $\mathcal{T}$.
for round $t = 1, 2, ..., T$ **do**
  Sending parameters:
  Obtain $\widetilde{\mathbf{\Sigma}}_t, \widetilde{\mathbf{u}}_t$ by traversing $\mathcal{T}$ to node t.
  Send $\widetilde{\mathbf{\Sigma}}_t, \widetilde{\mathbf{u}}_t \rightarrow$SERVER.
  Updating parameters:
  Receive $\mathbf{x}_t, y_t \leftarrow$SERVER.
  Insert $[\Phi(\mathbf{x}_t), y_t]^\top [\Phi(\mathbf{x}_t), y_t]$ into $\mathcal{T}$.
  Update noise values $\mathbf{n}$ on the inserted path $\mathcal{T}$.
end for

Definition 6 (Joint Differential Privacy (JDP))**.** *Let $S = (\mathcal{D}_i, y_i)_{i=1}^T$ and $S' = (\mathcal{D}'_i, y'_i)_{i=1}^T$ be two sequences such that $(\mathcal{D}_i, y_i) = (\mathcal{D}'_i, y'_i)$ for all $i \neq t$, and $\mathcal{S}_{-t} \subseteq \mathcal{D}_1 \times ... \times \mathcal{D}_{t-1} \times \mathcal{D}_{t+1} \times ... \times \mathcal{D}_T$ denote a sequence of actions except the t^{th}. An algorithm $\mathcal{A}$ is (α, β)-JDP under continual observation if for any $t \in [T], S, S'$, it holds that $\mathbb{P}(\mathcal{A}(S) \in \mathcal{S}_{-t}) \le e^\alpha \mathbb{P}(\mathcal{A}(S') \in \mathcal{S}_{-t}) + \beta$.*

The only change in the JDP setting (compared to standard DP) is that the algorithm is allowed to be non-private at time t with respect to $\mathcal{D}_t$ (i.e., the active decision set). This is crucial as standard DP would imply that for any two actions $\mathbf{x}, \mathbf{x}' \in \mathcal{D}_t, \mathbb{P}(a_t = \mathbf{x}) \approx \mathbb{P}(a_t = \mathbf{x}')$ and the algorithm would incur linear regret.

5.1 Approximate GP-UCB with JDP

Our approach involves perturbing $(\mathbf{\Sigma}_t, \mathbf{u}_t)$ by noise $(\mathbf{H}_t, \mathbf{h}_t)$ to ensure JDP, and it is summarized in Algorithm 2. Observe that the estimates $(\widetilde{\mathbf{\Sigma}}_t, \widetilde{\mathbf{u}}_t)$ are noisy cumulative sums of $\mathbf{\Sigma}_t = \sum_{\tau=1}^{t-1} \Phi(\mathbf{x}_\tau)\Phi(\mathbf{x}_\tau)^\top, \mathbf{u}_t = \sum_{\tau=1}^{t-1} y_\tau \cdot \Phi(\mathbf{x}_\tau)$. This additive structure naturally suggests that we utilize a matrix variant of the tree-based mechanism (Dwork et al., 2010; Shariff & Sheffet, 2018) to maintain $(\widetilde{\mathbf{\Sigma}}_t, \widetilde{\mathbf{u}}_t)$. We consider the matrix $\mathbf{N}_t = [\Phi(\mathbf{X}_t),\ \mathbf{y}_t]^\top [\Phi(\mathbf{X}_t),\ \mathbf{y}_t] \in \mathbb{R}^{m+1\times m+1}$ and compute this matrix via the tree-based mechanism. The advantage of maintaining $\mathbf{N}_t$ is that $\mathbf{N}_{t+1} = \mathbf{N}_t + [\Phi(\mathbf{x}_t),\ y_t]^\top [\Phi(\mathbf{x}_t),\ y_t]$ and the top $m \times m$ submatrix of $\mathbf{N}_t$ is $\mathbf{\Sigma}_t$ and the first m entries of the last column of $\mathbf{N}_t$ is $\mathbf{u}_t$, giving us the required estimates.

Tree-Based Mechanism. The tree-based mechanism (Dwork et al., 2010) estimates the rolling sum of any series $\mathbf{n}_1, \mathbf{n}_2, ...$ via a binary tree. Let P_{m+1} be a probability distribution over $\mathbb{R}^{m+1\times m+1}$. A trusted entity (in our case, the PRIVATIZER), maintains a binary tree $\mathcal{T}$ whose t^{th} leaf node stores $\mathbf{n}_t = [\Phi(\mathbf{x}_t)\, y_t]^\top [\Phi(\mathbf{x}_t)\, y_t] + (1/\sqrt{2})\,(\boldsymbol{v}_t\top + \boldsymbol{v}_t)$, where $\boldsymbol{v}_t$ is a

sample from P_{m+1}. Each parent node stores the sum of its children. Now, to compute $(\widetilde{\mathbf{\Sigma}}_t, \widetilde{\mathbf{u}}_t)$ we traverse $\mathcal{T}$ to the t^{th} leaf node, and sum the values at each node. Since the path length traversed is $1+\lceil \log_2 T \rceil$, we can rewrite $\widetilde{\mathbf{\Sigma}}_t = \mathbf{\Sigma}_t + \mathbf{H}_t$ where $\mathbf{H}_t$ is the sum of at most $n = 1+\lceil \log_2 T \rceil$ samples from P_{m+1}. We now describe selecting P_{m+1} to provide a JDP guarantee.

Lemma 2 (JDP). *Let P_{m+1} be a composition of $(m+1)^2$ zero-mean normal variables with variance $\sigma^2_{\alpha,\beta}$. If $\sigma_{\alpha,\beta} > 16n(1+B^2+2\rho^2\log(8T/\beta))\ln(10/\beta)^2/\alpha^2$, then Alg. 1 with* PRIVATIZER *following Alg. 2 is $(\alpha,\beta)-$jointly differentially private.*

Proof (Sketch). First note that since y_t is sub-Gaussian with mean at most B (since $\|f\|_k \leq B$), we can apply a standard Chernoff bound to ensure that with probability at least $1-\beta/4$, for each $(y_\tau)_{\tau\in[T]}$ simultaneously, $|y_t|^2 \leq B^2 + 2\rho^2\log(8T/\beta)$. Using this bound we can ensure that each datum has a bounded L_2-norm of $1+B^2+2\rho^2\log(4T/\beta)$ (since $\|\Phi(\mathbf{x})\|_2 \leq 1$). Based on the composition for zero-concentrated DP (Bun & Steinke, 2016), we see that for (α,β)-JDP, we require that each of the at most $n = 1+\lceil\log_2 T\rceil$ nodes maintains $(\alpha/\sqrt{8n\ln(2/\beta)}, \beta/2)$-DP. With the L_2 sensitivity result from earlier, we see that $\sigma^2_{\alpha,\beta} = 16n(1+B^2+2\rho^2\log(8T/\beta))\ln(10/\beta)^2/\alpha^2$ provides (α,β)-JDP, finishing the proof. ■

Recall that our regret bound (Corollary 1) scales with the parameters $\lambda_{\min}, \lambda_{\max}$ and κ. It remains to provide these quantities under the selected P_{m+1} such that they are accurate (Defn. 3), and provide final regret bounds based on the properties of P_{m+1}. As remarked in Shariff & Sheffet (2018), we must shift the noise matrix to ensure that all noise samples $\mathbf{H}_t$ are PSD.

Lemma 3 (Accurate Spectrum under JDP). *For any $\zeta > 0$, when P_{m+1} is selected according to Lemma 2 and $\mathbf{H}_t, \mathbf{h}_t$ are constructed according to Alg. 2, the following $\lambda_{\min}, \lambda_{\max}$ and κ are $(\zeta/2T)-$accurate:*

$$\lambda_{\min} = \Lambda, \lambda_{\max} = 3\Lambda, \kappa = \sigma_{\alpha,\beta}\sqrt{\tfrac{n}{\Lambda}}\left(\sqrt{m} + \sqrt{2\ln\tfrac{2T}{\zeta}}\right).$$

Here $\Lambda = \sigma_{\alpha,\beta}\sqrt{2n}(4\sqrt{m} + 2\ln(2T/\zeta))$.

Proof. This proof is identical to Proposition 11 from Shariff & Sheffet (2018) with our noise model. ■

Corollary 2 ((α,β)-JDP Regret Bound). *Fix $m = 2(6\log T)^d$ and let k be any kernel that obeys Assumption 1. Algorithm 1 run with m-dimensional NQFF and noise such that it maintains $(\alpha,\beta)-$JDP obtains with probability at least $1-\zeta$, cumulative pseudoregret:*

$$\mathfrak{R}(T) = \mathcal{O}\left(\sqrt{T\gamma_T}\left((\ln T)^{\frac{d+2}{4}}(\tfrac{1}{\alpha}\log\tfrac{1}{\beta}\log\tfrac{1}{\zeta})^{\frac{1}{2}} + (\ln T)^{\frac{d+1}{2}}\right)\right).$$

The proof for Corollary 2 follows directly by substituting the results from Lemma 3 into Corollary 1.

Remark 6 (Dependence on m). *Since the factors $\lambda_{\min}, \lambda_{\max}$ and κ admit a dependence of $\mathcal{O}(\sqrt{m})$ on the dimensionality of Φ, we require $m = o(\sqrt{T})$ features to guarantee no-regret learning under our approach. This constraint is complementary to the constraint on m from kernel approximation (Remark 4), and mandates that even when the approximation $\tilde{k}$ has small γ_T (i.e., $\gamma_T = o(\text{polylog}(T))$, we require small m.*

5.2 Approximate GP-UCB with Local JDP

In many settings, the existence of a trusted entity (e.g., PRIVATIZER) is not possible. For instance, consider the task of a centralized server learning a bandit algorithm in the case when each user t does not wish $(\mathcal{D}_t, y_t)$ to be sent to the server at all (even to select $\mathbf{x}_t$). We can select $\mathbf{x}_t$, however, by sending the algorithm's (privatized) parameters to each user individually and collecting updated parameters after $\mathbf{x}_t$ has been played by the user t. Here, we employ an alternative definition of privacy known as local JDP.

Definition 7 (Locally Joint Differential Privacy (Local JDP)). *A mechanism $g: \mathcal{X} \to \mathcal{Z}$ is (α,β)-locally differentially private (Bebensee, 2019) (LDP) if for any $\mathbf{x}, \mathbf{x}' \in \mathcal{X}, \mathbb{P}(g(\mathbf{x}) \in \mathcal{Z}) \leq e^\alpha \mathbb{P}(g(\mathbf{x}') \in \mathcal{Z}) + \beta$. For any sequence $(\mathcal{D}_t, y_t)_{t=1}^T$, an algorithm $\mathcal{A}$ protects locally joint differentially privacy (Local JDP) if for any t, $\mathcal{A}$ is locally differentially private with respect to each $(\mathcal{D}_\tau, y_\tau)$ simultaneously where $\tau \neq t$.*

This definition combines joint differential privacy (operating globally) with local differential privacy (operating individually). It is important to note that local JDP is weaker than LDP (Bebensee, 2019), since LDP would require local privacy with respect to $(\mathcal{D}_t, y_t)$ as well. It is a stronger privacy guarantee than JDP, since it requires $\mathcal{A}$ to be private to each user simultaneously.

Lemma 4 (Local JDP implies JDP). *Any $(\alpha,\beta)-$local JDP algorithm $\mathcal{A}$ protects (α,β)-JDP for each $t \in [T]$.*

Proof (Sketch). For any $t \in [T]$, any two t-neighboring sequences S and S' only differ in the t^{th} entries $(\mathcal{D}_t, y_t)$ and $(\mathcal{D}'_t, y'_t)$. Since $\mathcal{A}$ is $(\alpha,\beta)-$locally JDP, $\mathbb{P}(a_{t'}(\mathcal{D}_t, y_t)) \in \mathcal{S}_{t'}) \leq e^\alpha \mathbb{P}(a_{t'}(\mathcal{D}'_t, y'_t)) \in \mathcal{S}_{t'}) + \beta$ for all $t' \neq t$, from which the result follows. ■

Since a trusted entity does not exist, learning is done by sending the parameters directly to the users (ref. clients). We outline a server-client protocol and associated algorithm for (α,β)-local JDP Gaussian Process bandit optimization in Algorithm 3. This algorithm requires noise added individually to $(\Phi(\mathbf{x}_t), y_t)$ (instead of $(\mathbf{\Sigma}_t, \mathbf{u}_t)$). We achieve this by perturbing $\mathbf{\Sigma}_t$

Algorithm 3 GP-UCB with Local JDP

SERVER:
Initialize: $\mathbf{\Sigma}_1 = \mathbf{0}, \mathbf{u}_1 = \mathbf{0}$.
for round $t = 1, 2, ..., T$ **do**
  Send $\widetilde{\mathbf{\Sigma}}_t, \widetilde{\mathbf{u}}_t \rightarrow$CLIENT$(t)$.
  Receive updated $\widetilde{\mathbf{\Sigma}}_{t+1}, \widetilde{\mathbf{u}}_{t+1} \leftarrow$CLIENT$(t)$.
end for
CLIENT(t):
Initialize σ_X^2 and σ_u^2 according to Lemma 5.
Receive $\mathcal{D}_t$ from environment.
Receive $\widetilde{\mathbf{\Sigma}}_t, \widetilde{\mathbf{u}}_t \leftarrow$SERVER.
Set $\mathbf{V}_t \leftarrow \widetilde{\mathbf{\Sigma}}_t + \lambda \mathbf{I}, \widetilde{\boldsymbol{\theta}}_t \leftarrow \mathbf{V}_t^{-1} \widetilde{\mathbf{u}}_t$.
Compute β_t based on Theorem 1.
Select $\mathbf{x}_t \leftarrow \arg\max_{\mathbf{x} \in \mathcal{D}_t} \langle \widetilde{\boldsymbol{\theta}}_t, \Phi(\mathbf{x}) \rangle + \beta_t \|\Phi(\mathbf{x})\|_{\mathbf{V}_t^{-1}}$.
Play arm $\mathbf{x}_t$ and obtain y_t.
Sample $\mathbf{N}_t, \mathbf{n}_t$ using σ_X^2, σ_u^2.
Send $\widetilde{\mathbf{\Sigma}}_{t+1} \rightarrow \mathbf{\Sigma}_t + \Phi(\mathbf{x}_t)\Phi(\mathbf{x}_t)^\top + \mathbf{N}_t \rightarrow$SERVER.
Send $\widetilde{\mathbf{u}}_{t+1} \rightarrow \mathbf{u}_t + y_t \Phi(\mathbf{x}_t) + \mathbf{n}_t \rightarrow$SERVER.

and $\mathbf{u}_t$ separately with $\mathbf{N}_t \in \mathbb{R}^{m \times m}$ where $\mathbf{N}_t(i,j) \sim \mathcal{N}(0, \sigma_X^2)$ for $i \geq j$ and $\mathbf{N}_t(i,j) = \mathbf{N}_t(j,i)$ otherwise and $\mathbf{n}_t \in \mathbb{R}^m$ is such that $\mathbf{n}_t(i) \sim \mathcal{N}(0, \sigma_u^2)$. The variances σ_X^2 and σ_u^2 are chosen to ensure $(\alpha/2, \beta/2)$ respectively, securing (α, β)-Local JDP.

Lemma 5 (Noise for Local JDP)**.** *Algorithm 3 is $(\alpha, \beta)-$locally JDP whenever,*

$$\sigma_X^2 \geq \frac{8}{\alpha^2} \ln \frac{5}{2\beta}, \ \sigma_u^2 \geq \frac{8}{\alpha^2} \left(B^2 + 2 \ln \frac{8m}{\delta} \right) \ln \frac{5}{\beta}.$$

Proof. We first note that the L_2-sensitivity of each element within $\Phi(\mathbf{x}_t)^\top \Phi(\mathbf{x}_t)$ is 1 by the fact that $\|\Phi(\mathbf{x})\| \leq 1$. Next, note that the L_2- sensitivity of each element of $y_t \Phi(\mathbf{x}_t)$ is with probability at least $1 - \beta/4$ at most $B + \rho\sqrt{2 \log \frac{8m}{\delta}}$ (y_t is Gaussian with mean at most B). Now, by the Gaussian mechanism for local DP (Dwork & Roth, 2014), we have that for $\sigma_x^2 \geq \frac{8 \ln(2.5/\beta)}{\alpha^2}$ and $\sigma_u \geq \frac{8(B^2 + 2 \log \frac{8m}{\delta}) \ln(5/\beta)}{\alpha}$, both $\Phi(\mathbf{x}_t)^\top \Phi(\mathbf{x}_t) + \mathbf{N}_t$ and $y_t \Phi(\mathbf{x}_t) + \mathbf{n}_t$ are $(\alpha/2, \beta/2)-$locally DP. ■

It remains to bound the spectral parameters ($\lambda_{\min}, \lambda_{\max}$ and κ) in order to obtain regret bounds.

Lemma 6. *(Spectrum for Local JDP) For any $\zeta > 0$, fix $\Lambda = \sqrt{T}(4\sqrt{m} + 2\ln(2T/\zeta))$. When P_{m+1} is selected according to Lemma 5 and $\mathbf{H}_t, \mathbf{h}_t$ are constructed according to Alg. 3, the following are $(\zeta/2T)-$accurate:*

$$\lambda_{\min} = \sigma_x \Lambda, \lambda_{\max} = 3\sigma_x \Lambda \ \textit{and,} \kappa = \sigma_u \sqrt{mT\Lambda^{-1}}.$$

Proof (Sketch). The proof is identical to Lemma 3 except critically that in this case, $\mathbf{H}_t$ (resp. $\mathbf{h}_t$) is the sum of t matrices $\mathbf{N}_t$ (resp. $\mathbf{n}_t$), with total variance $t\sigma_X^2$ (resp. $t\sigma_u^2$). Therefore, we can bound $\|\mathbf{H}_t\|_2 \leq \sigma_X \sqrt{T}(4\sqrt{m} + 2\ln(2T/\zeta))$ and $\|\mathbf{h}_t\|_2 \leq \sigma_u \sqrt{mT}$, which gives the result identical to Lemma 3. ■

Corollary 3 ($(\alpha, \beta)-$Local JDP Regret Bound)**.** *Fix $m = (6 \log T)^d$ and let k be any kernel that obeys Assumption 1. Algorithm 1 run with NQFF and noise $\mathbf{H}_t, \mathbf{h}_t$ that maintains $(\alpha, \beta)-$local JDP obtains with probability at least $1 - \zeta$, cumulative pseudoregret:*

$$\mathfrak{R}(T) = \mathcal{O}\left(T^{\frac{3}{4}} (\ln T)^{\frac{d+2}{4}} \sqrt{\frac{\gamma_T}{\alpha} \ln \frac{1}{\beta} \ln \frac{1}{\zeta}} \right).$$

The proof for Corollary 3 follows directly by substituting the results from Lemma 6 into Corollary 1.

Remark 7 (JDP vs. Locally JDP Regret)**.** *Our algorithm for the locally JDP setting obtains $\widetilde{\mathcal{O}}(T^{3/4})$ regret in contrast to the JDP regret, which is close to the minimax optimal rate of $\widetilde{\Omega}(\sqrt{T})$ for squared-exponential and Matérn kernels (Scarlett et al., 2017). It is evident that this suboptimality is introduced by the $\widetilde{\mathcal{O}}(T)$ noise added via $\mathbf{H}_t$. However, we conjecture that in the absence of any known structure between the chosen actions $\mathbf{x}_1, ..., \mathbf{x}_{t-1}$, it is impossible to add correlated noise samples (i.e., such that the overall variance is $o(T)$) while maintaing local DP, as typically the environment selects $\mathcal{D}_t$ independently of $\mathcal{D}_{t-1}$.*

6 Experiments

We conduct experiments primarily around the noisy Quadrature features for GP optimization, and consider the Joint DP setting. For more experimental results on the approximation guarantees of QFF, please refer to the appendix and experimental section of Mutny & Krause (2018), that analyse the efficiency of quadrature features in approximating stationary kernels.

We conduct experiments with input dimensionality $d = 2$ and selecting the squared-exponential kernel with variance 1, i.e., $k(\mathbf{x}, \mathbf{y}) = \exp(-\|\mathbf{x} - \mathbf{y}\|_2^2/2)$ for simplicity. This choice was made as we essentially wish to demonstrate that the algorithms are private in practice for toy experiments, as larger dimensionalities ($d > 5$) are rarely seen in practice (Mutny & Krause, 2018) and would require additive assumptions for efficient inference (Munkhoeva et al., 2018).

6.1 Experimental Setup

We construct f by randomly sampling a set of points $\mathcal{I}$ from $\mathcal{B}_d(2)$ such that $|\mathcal{I}| = 4$ and randomly generate $\boldsymbol{\alpha}$ from the unit L_1 ball $\mathcal{B}_d(1)$ (therefore, we assume $B = 1$). For any input point $\mathbf{x}$, $f(\mathbf{x})$ can then be denoted as $f(\mathbf{x}) = \sum_{i=1}^{|\mathcal{I}|} \alpha_i k(\mathbf{x}_i, \mathbf{x})$, where $\mathbf{x}_1, ..., \mathbf{x}_{|\mathcal{I}|}$

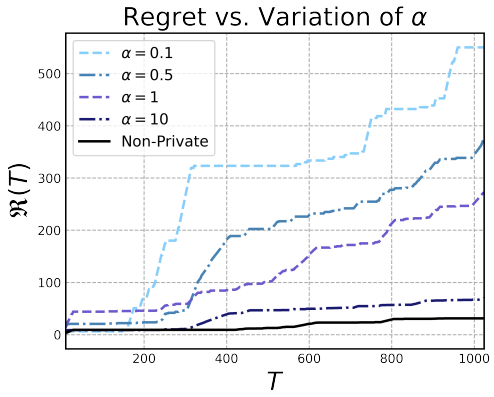

Figure 1: An experimental comparison of approximate `GP-UCB` for various values of privacy budget α.

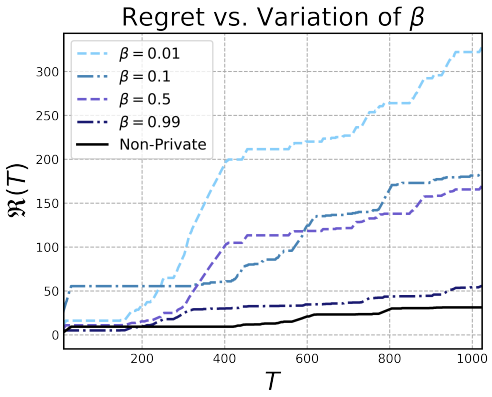

Figure 2: An experimental comparison of approximate `GP-UCB` for various values of privacy failure probability β.

belong to $\mathcal{I}$. We consider $\mathcal{D}$ to be a random sample of size n drawn from $\mathcal{B}_d(2)$ (n may be variable, but is specified prior to each experiment). We draw $\mathcal{D}_t$ such that at least 1 sample $\mathbf{x}$ from $\mathcal{D}_t$ satisfies $f(\mathbf{x}) \geq 0.8$ and the others satisfy $f(\mathbf{x}) \leq 0.6$, ensuring a suboptimality gap of at least 0.2 (this is implemented somewhat crudely by iterative sampling). At each round t, the agent is presented with a random $\mathcal{D}_t$ and it obtains a reward y_t drawn from the distribution $\text{Ber}(f(\mathbf{x}))$ and hence $|\varepsilon_t| \leq 1$ and $\mathbb{E}[y_t] = f(\mathbf{x})$. Additionally, we see that the variance $\rho^2 = f(\mathbf{x})(1-f(\mathbf{x}))$ for this case, but that is bounded from above by $1/4$. For simplicity, we restrict ourselves to Bernoulli rewards. This model, while ensuring sub-Gaussianity, also ensures that the rewards are bounded, and hence removes an additional logarithmic factor from the sensitivity analysis for the JDP setting. This can be observed by directly applying L_2-sensitivity to the JDP noise (Lemma 2), and ignoring the probabilistic argument.

Effect of α. We first examine the effect of adjusting the privacy level α. We fix $n = 25$, $\beta = 0.1$ and set $T = 1024$ (similar to Mutny & Krause (2018)). We run 20 trials and compare the performance at $\alpha = 0.1, 0.5, 1, 10$ (averaged over 20 trials). The regret scales as predicted with decreasing α (Figure 1).

Effect of β. Next, we examine the effect of adjusting the privacy failure probability β. We fix $n = 25$, $\alpha = 1$ and set $T = 1024$ (similar to Mutny & Krause (2018)). We run 20 trials and compare the performance at $\beta = 0.01, 0.1, 0.5, 0.99$ (averaged over 20 trials). The regret increases with decreasing β, summarized in Figure 2.

6.2 Additional Benchmarks

In addition to the environment proposed earlier, we additionally evaluate the JDP algorithm on previous benchmark environments. We consider the functional environments for the Camelback (`camel`), Stybtang-20 (`styb`) and Michalewicz-10 (`mw`) benchmarks from (Mutny & Krause, 2018). We observe a consistent increase in regret as the privacy budget (α) is reduced (Table 1). While the bound predicts a $\alpha^{-\frac{1}{2}}$ deterioration, we observe a larger effect, which suggests that stronger analyses can close the gap.

Alg.	`camel`	`styb`	`mw`
Non-Private	519	885	901
$\alpha = 10$	775	1667	1558
$\alpha = 1$	1029	2680	2883
$\alpha = 0.1$	3324	4493	5002

Table 1: Cumulative regret at $T = 10K$ averaged over 10 trials on functions from Mutny & Krause (2018).

7 Discussion and Concluding Remarks

In this paper, we presented the first *no-regret* algorithmic framework for differentially-private Gaussian Process bandit optimization for a class of stationary kernels in both the joint DP and local DP settings, extending the literature on private bandit estimation beyond multi-armed (Mishra & Thakurta, 2015) and linear (Shariff & Sheffet, 2018) problems. We rigorously analyse the proposed algorithms and demonstrate their provable efficiency in terms of regret, computation and privacy. Our work additionally introduces several new avenues for further research - while the dependence of the achieved pseudoregret on T is near-optimal in the JDP setting, the local JDP setting introduces an additional $\mathcal{O}(T^{1/4})$ which we conjecture is necessary owing to the nested estimation problems involved (Remark 7). Additionally, developing lower bounds on private GP regret and efficient kernel approximations for non-stationary kernels are valuable pursuits of inquiry.

Acknowledgements

We would like to thank Dr. Alex Pentland for his helpful comments, and the anonymous reviewers for their feedback and suggestions. This work was supported by the MIT Trust::Data Consortium.

References

Agarwal, D., Basu, K., Ghosh, S., Xuan, Y., Yang, Y., and Zhang, L. Online parameter selection for web-based ranking problems. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 23–32, 2018.

Agarwal, N. and Singh, K. The price of differential privacy for online learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 32–40, 2017.

Avron, H., Kapralov, M., Musco, C., Musco, C., Velingker, A., and Zandieh, A. Random fourier features for kernel ridge regression: Approximation bounds and statistical guarantees. In *International Conference on Machine Learning*, pp. 253–262, 2017.

Basu, D., Dimitrakakis, C., and Tossou, A. Differential privacy for multi-armed bandits: What is it and what is its cost?, 2020.

Bebensee, B. Local differential privacy: a tutorial. *arXiv preprint arXiv:1907.11908*, 2019.

Bochner, S. Monotone funktionen, stieltjessche integrale und harmonische analyse. *Mathematische Annalen*, 108(1):378–410, 1933.

Bun, M. and Steinke, T. Concentrated differential privacy: Simplifications, extensions, and lower bounds. In *Theory of Cryptography Conference*, pp. 635–658. Springer, 2016.

Calandriello, D., Carratino, L., Lazaric, A., Valko, M., and Rosasco, L. Gaussian process optimization with adaptive sketching: Scalable and no regret. *arXiv preprint arXiv:1903.05594*, 2019.

Chowdhury, S. R. and Gopalan, A. On kernelized multi-armed bandits. *arXiv preprint arXiv:1704.00445*, 2017.

Costabal, F. S., Matsuno, K., Yao, J., Perdikaris, P., and Kuhl, E. Machine learning in drug development: Characterizing the effect of 30 drugs on the qt interval using gaussian process regression, sensitivity analysis, and uncertainty quantification. *Computer Methods in Applied Mechanics and Engineering*, 348:313–333, 2019.

Cummings, R. and Desai, D. The role of differential privacy in gdpr compliance. In *FAT'18: Proceedings of the Conference on Fairness, Accountability, and Transparency*, 2018.

Dao, T., De Sa, C. M., and Ré, C. Gaussian quadrature for kernel features. In *Advances in neural information processing systems*, pp. 6107–6117, 2017.

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

Dwork, C., Naor, M., Pitassi, T., and Rothblum, G. N. Differential privacy under continual observation. In *Proceedings of the forty-second ACM symposium on Theory of computing*, pp. 715–724, 2010.

Hildebrand, F. B. *Introduction to numerical analysis*. Courier Corporation, 1987.

Kusner, M., Gardner, J., Garnett, R., and Weinberger, K. Differentially private bayesian optimization. In *International conference on machine learning*, pp. 918–927, 2015.

Letham, B. and Bakshy, E. Bayesian optimization for policy search via online-offline experimentation. *Journal of Machine Learning Research*, 20(145):1–30, 2019.

Liu, X. and Guillas, S. Dimension reduction for gaussian process emulation: An application to the influence of bathymetry on tsunami heights. *SIAM/ASA Journal on Uncertainty Quantification*, 5(1):787–812, 2017.

Meeds, E. and Welling, M. Gps-abc: Gaussian process surrogate approximate bayesian computation. *arXiv preprint arXiv:1401.2838*, 2014.

Mishra, N. and Thakurta, A. (nearly) optimal differentially private stochastic multi-arm bandits. In *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pp. 592–601, 2015.

Močkus, J. On bayesian methods for seeking the extremum. In *Optimization techniques IFIP technical conference*, pp. 400–404. Springer, 1975.

Munkhoeva, M., Kapushev, Y., Burnaev, E., and Oseledets, I. Quadrature-based features for kernel approximation. In *Advances in Neural Information Processing Systems*, pp. 9147–9156, 2018.

Mutny, M. and Krause, A. Efficient high dimensional bayesian optimization with additivity and quadrature fourier features. In *Advances in Neural Information Processing Systems*, pp. 9005–9016, 2018.

Park, M., Nassar, M., and Vikalo, H. Bayesian active learning for drug combinations. *IEEE transactions on biomedical engineering*, 60(11):3248–3255, 2013.

Peterson, K., Rudovic, O., Guerrero, R., and Picard, R. W. Personalized gaussian processes for future

prediction of alzheimer's disease progression. *arXiv preprint arXiv:1712.00181*, 2017.

Rahimi, A. and Recht, B. Random features for large-scale kernel machines. In *Advances in neural information processing systems*, pp. 1177–1184, 2008.

Rohde, D., Bonner, S., Dunlop, T., Vasile, F., and Karatzoglou, A. Recogym: A reinforcement learning environment for the problem of product recommendation in online advertising. *arXiv preprint arXiv:1808.00720*, 2018.

Scarlett, J., Bogunovic, I., and Cevher, V. Lower bounds on regret for noisy gaussian process bandit optimization. *arXiv preprint arXiv:1706.00090*, 2017.

Shariff, R. and Sheffet, O. Differentially private contextual linear bandits. In *Advances in Neural Information Processing Systems*, pp. 4296–4306, 2018.

Smith, M. T., Zwiessele, M., and Lawrence, N. D. Differentially private gaussian processes. *arXiv preprint arXiv:1606.00720*, 2016.

Snoek, J., Larochelle, H., and Adams, R. P. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pp. 2951–2959, 2012.

Srinivas, N., Krause, A., Kakade, S., and Seeger, M. Gaussian process optimization in the bandit setting: no regret and experimental design. In *Proceedings of the 27th International Conference on International Conference on Machine Learning*, pp. 1015–1022, 2010.

Tossou, A. and Dimitrakakis, C. Algorithms for differentially private multi-armed bandits. *arXiv preprint arXiv:1511.08681*, 2015.

Vanchinathan, H. P., Nikolic, I., De Bona, F., and Krause, A. Explore-exploit in top-n recommender systems via gaussian processes. In *Proceedings of the 8th ACM Conference on Recommender systems*, pp. 225–232, 2014.

Vietri, G., Balle, B., Krishnamurthy, A., and Wu, Z. S. Private reinforcement learning with pac and regret guarantees. *arXiv preprint arXiv:2009.09052*, 2020a.

Vietri, G., Balle, B., Krishnamurthy, A., and Wu, Z. S. Private reinforcement learning with pac and regret guarantees. *arXiv preprint arXiv:2009.09052*, 2020b.

Williams, C. K. and Rasmussen, C. E. *Gaussian processes for machine learning*, volume 2. MIT press Cambridge, MA, 2006.

Zhou, W., Li, J., Yang, Y., and Shah, F. Leverage side information for top-n recommendation with latent gaussian process. *Concurrency and Computation: Practice and Experience*, pp. e5534, 2019.