Fast Fourier Bayesian Quadrature

Houston Warren
The University of Sydney

Fabio Ramos
The University of Sydney & NVIDIA

Abstract

In numerical integration, Bayesian quadrature (BQ) excels at producing estimates with quantified uncertainties, particularly in sparse data settings. However, its computational scalability and kernel learning capabilities have lagged behind modern advances in Gaussian process research. To bridge this gap, we recast the BQ posterior integral as a convolution operation, which enables efficient computation via fast Fourier transform of low-rank matrices. We introduce two new methods enabled by recasting BQ as a convolution: fast Fourier Bayesian quadrature and sparse spectrum Bayesian quadrature. These methods enhance the computational scalability of BQ and expand kernel flexibility, enabling the use of *any* stationary kernel in the BQ setting. We empirically validate the efficacy of our approach through a range of integration tasks, substantiating the benefits of the proposed methodology.

1 INTRODUCTION

Several domains, from physical simulation to Bayesian inference, necessitate the computation of challenging, non-analytic integrals. Conventional techniques like quadrature rules and Monte Carlo approximation are commonly employed to mitigate this issue. Quadrature approaches often approximate the integral as a weighted sum of integrand evaluations $f(\mathbf{x})$:

$$\int f(\mathbf{x})p(\mathbf{x})d\mathbf{x} \approx \sum_{i=1}^{N} \tau_i f(\mathbf{x}_i) \ . \tag{1}$$

However, for many problems, these methodologies may be slow to converge and require a large number of model evaluations to reach a desired level of solution accuracy (Robert et al., 2018), or may even be infeasible if the solution environment or computational-budget requires fast evaluations or data efficiency.

Probabilistic numerical integration offers an alternative. These methods instead model the integrand or integral as a probability measure, which enables the use of statistical inference to achieve data-efficient results (Cockayne et al., 2019; Briol et al., 2019). Among these approaches, Bayesian quadrature (BQ) is particularly compelling for its data efficiency and robust uncertainty quantification (O'Hagan, 1991; Ghahramani and Rasmussen, 2003). These advantages stem from modeling the integrand as a Gaussian process (GP) (Rasmussen and Williams, 2006) which can be used compute closed-form integral posteriors when using specific choices of GP kernel covariance functions.

However, the power of GP methods often lies in the ability to select a kernel that incorporates known qualities of the problem under study, such as symmetries, smoothness, or periodicity. BQ's limitation in providing analytical results for only a much-reduced set of kernel functions – rather than all possible positive-definite kernel functions – limits BQ's applicability on specialized or complex integrands that could benefit from domain-derived kernels.

In addition to this modeling inflexibility, BQ scales in $\mathcal{O}(N^3)$ time due to a necessary inversion of the kernel Gram matrix $\mathbf{K}$ for calculation of the GP posterior, which can grow computationally intractable for large datasets.

Recent advances in Gaussian processes (GPs) have focused on enhancing kernel expressiveness and learning (Xie et al., 2019; He et al., 2020; Ober et al., 2021; S. Zhu et al., 2021), as well as computational scalability (Liu et al., 2019). A key breakthrough is the spectral representation of kernels (Rahimi and Recht, 2008a; Wilson and Adams, 2013), which offers both flexible kernel learning and computational efficiency. However, traditional Bayesian quadrature (BQ) has yet to capitalize on these advances, as they diverge from the limited set of kernels that permit an analytically tractable

Proceedings of the 27th International Conference on Artificial Intelligence and Statistics (AISTATS) 2024, Valencia, Spain. PMLR: Volume 238. Copyright 2024 by the author(s).

BQ integral posterior. This intractability introduces a need for Monte Carlo approximations for many kernel and measure choices, which runs counter to the original motivations of BQ to avoid such approaches.

Recent work has reformulated the Bayesian integration problem using alternative integrand model architectures (H. Zhu et al., 2020; Ott et al., 2023) or adapted low-rank inducing point methods (Hensman et al., 2015; Titsias, 2009) to the BQ setting (Adachi et al., 2022; Hayakawa et al., 2022).

However, the opportunity remains to incorporate existing advances in spectral GP methodologies – which offer both computational improvements as well as flexible kernel parametrization and learning schemes – into traditional BQ. In this paper, we build upon recent work (Warren et al., 2022; Sellier and Dellaportas, 2023) by leveraging tools from spectral analysis to address the constraints of classical BQ. Our main contributions are:

1. We recast BQ as a spectral convolution, facilitating the utilization of modern kernel learning and low-rank approximation techniques. This reformulation allows for quick and precise BQ integral posterior approximations for any stationary kernel and measure.
2. We propose two novel methods derived from this reformulation: fast Fourier Bayesian quadrature (FFBQ), which leverages the fast Fourier transform (FFT) for BQ kernel mean calculation, and sparse spectrum Bayesian quadrature (SSBQ), a low-rank and computationally efficient BQ variant.
3. We provide empirical evidence of FFBQ and SSBQ's efficacy across a range of integration problems, demonstrating their adaptability and performance in various dimensions and contexts.

2 PRELIMINARIES

This section will briefly review various preliminaries necessary to introduce the FFBQ and SSBQ methods.

2.1 Gaussian Processes

In the BQ setting, we assume that we have N noisy samples $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N = \{\mathbf{X}, \mathbf{y}\}$ of an integrand f, where the noise ϵ is i.i.d normal, i.e.: $y_i = f(\mathbf{x}_i) + \epsilon$. The motivating use cases for BQ involve situations where Monte Carlo integration or quadrature methods are prohibitively expensive, such that we assume samples of f are sparse and thus data-efficiency is necessary. This data-efficiency condition is the driving motivation for the use of Gaussian processes (Rasmussen and Williams, 2006) for setting a non-parametric prior on the integrand f given the ability for GPs to offer uncertainty-quantified predictions from sparse data. The GP prior takes the form of a joint multivariate Gaussian:

$$f \sim \mathcal{GP}(\boldsymbol{\mu}(\mathbf{x}), k_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{x}')), \tag{2}$$

$$\mathbf{y} = f(\mathbf{x}) + \epsilon. \tag{3}$$

In (2), $k_{\boldsymbol{\theta}}$ is a positive semi-definite *kernel function* with hyper-parameters $\boldsymbol{\theta}$, and $\boldsymbol{\mu}$ is a mean function, which is commonly set to $\mathbf{0}$ without loss of generality. The associated multivariate-normal GP posterior-predictive for a new data point $\{\mathbf{x}_*\}$ is given as:

$$\mu(f*) = \mathbf{K}_{*\mathbf{x}}(\mathbf{K}_{\mathbf{xx}} + \sigma^2\mathbf{I})^{-1}\mathbf{y}, \tag{4}$$

$$\mathrm{Cov}(f*) = \mathbf{K}_{**} - \mathbf{K}_{*\mathbf{x}}(\mathbf{K}_{\mathbf{xx}} + \sigma^2\mathbf{I})^{-1}\mathbf{K}_{\mathbf{x}*}, \tag{5}$$

where $\mathbf{K}$ is the Gram matrix $\mathbf{K}_{\mathbf{xx}} = k_{\boldsymbol{\theta}}(\mathbf{x}, \mathbf{x}')$, $\forall\, \mathbf{x}, \mathbf{x}' \in \mathbf{X}$, and $\{\mathbf{x}_i, y_i\}_{i=1}^N = \{\mathbf{X}, \mathbf{y}\}$ are the training data.

The choice of kernel function k, such as the widely-used squared-exponential (RBF) kernel, serves as a conduit for incorporating prior domain knowledge into the GP model, including attributes like periodicity, smoothness, or derivative information (Raissi and Karniadakis, 2018). The inductive bias from the kernel significantly contributes to the GP's robust predictive capabilities, particularly when data are sparse.

2.2 Bayesian Quadrature

As the GP posterior is multivariate Gaussian, the expectation of (4) over a Gaussian measure $p(\mathbf{x})$ is also Gaussian (Rasmussen and Williams, 2006). BQ uses this fact to form an integral estimate $\langle \bar{f} \rangle$ of the integrand f, which yields an analytical solution when the kernel k is the RBF (Gaussian) kernel (Briol et al., 2019). Formally, the mean of the BQ posterior integral estimate is defined as:

$$\begin{aligned}\langle \bar{f} \rangle &= \int k(\mathbf{x}, \mathbf{X})^T \mathbf{K}^{-1} \mathbf{y}\, p(\mathbf{x})\, d\mathbf{x} \\ &= \mathbf{y}^T \mathbf{K}^{-1} \int k(\mathbf{x}, \mathbf{X})\, p(\mathbf{x})\, d\mathbf{x} \\ &= \mathbf{y}^T \mathbf{K}^{-1} \mu_{\mathbf{x}}(\mathbf{X}),\end{aligned} \tag{6}$$

where $\mathbf{y} = f(\mathbf{x}) + \epsilon$ are integrand observations, $\mu_{\mathbf{x}}(\mathbf{X}) = \int k(\mathbf{x}, \mathbf{X})\, p(\mathbf{x})\, d\mathbf{x}$ is the kernel mean of the observed data $\mathbf{X}$, and for brevity we define $(\mathbf{K} + \sigma^2\mathbf{I})^{-1} := \mathbf{K}^{-1}$ throughout. The associated variance of $\langle \bar{f} \rangle$ is:

$$\mathbb{V}(\langle \bar{f} \rangle) = \mu_{\mathbf{xx}'} - \mu_{\mathbf{x}}(\mathbf{X})^T \mathbf{K}^{-1} \mu_{\mathbf{x}}(\mathbf{X}) \tag{7}$$

$$\mu_{\mathbf{xx}'} = \int \int k(\mathbf{x}, \mathbf{x}') p(\mathbf{x}) p(\mathbf{x}') d\mathbf{x}\; d\mathbf{x}' \tag{8}$$

where $\mu_{\mathbf{x}\mathbf{x}'}$ represents the kernel mean over both $\mathbf{x}$ and $\mathbf{x}'$. The BQ mean equates to a quadrature rule of the form in Equation 1, where weights $\boldsymbol{\tau} \equiv \mu_{\mathbf{x}}(\mathbf{X})\mathbf{K}^{-1}$ are defined by the GP fit to integrand samples $\{\mathbf{X}, \mathbf{y}\}$.

In this context, it is crucial to recognize that all error in the BQ posterior integral estimates $\langle \bar{f} \rangle$ and $\mathbb{V}(\langle \bar{f} \rangle)$ arises from the integrand GP, not the integral calculation itself. This fact emphasizes the significance of the GP model's fit, which is influenced primarily by:

1. The Gaussian assumption for integrand samples $\mathbf{y}$.
2. The selection of kernel function k and its hyperparameters $\boldsymbol{\theta}$.

This paper focuses on the latter. Traditional BQ restricts kernel choices to ensure analytical tractability, limiting its applicability. Specifically, if the integrand does not reside in the Hilbert space $\mathcal{H}$ governed by k, the GP assumptions are violated, affecting empirical performance. Numerical approximations for terms $\mu_{\mathbf{x}}(\mathbf{X}) \approx \hat{\mu}_{\mathbf{x}}(\mathbf{X})$ and $\mu_{\mathbf{x}\mathbf{x}'} \approx \hat{\mu}_{\mathbf{x}\mathbf{x}'}$ are required to extend BQ's reach beyond analytically tractable kernels.

Recent advances propose a more flexible BQ formulation leveraging spectral kernel representations, thus permitting the use of any stationary kernel. We will explore these innovations and their theoretical underpinnings next.

2.3 Spectral Kernel Methods

Representing and learning kernel functions through their spectral representations has been a significant method in recent years, with foundational works (Rahimi and Recht, 2008a; Rahimi and Recht, 2008b; Wilson and Adams, 2013; Le et al., 2013) and their derivatives proving to be a powerful means to increase kernel flexibility and computational scaling in a wide array of kernel methods.

The validity of the spectral kernel representation derives from Bochner's theorem:

Theorem 1 (Bochner's theorem (Rudin, 2011))**.** *A shift-invariant kernel $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}')$ is positive-definite if and only if it is the Fourier transform of a non-negative measure.*

The implication of Bochner's theorem is that while positive-definite kernel functions may be difficult to define in a parametric form, we may instead leverage the rich capabilities of distribution modeling to represent and approximate a kernel.

Random Fourier Features Perhaps the most well-known approach to applying Bochner's theorem is through random Fourier features (RFFs) (Rahimi and Recht, 2008a). RFFs present a means through which the theoretical results of Theorem 1 can be applied by modeling stationary kernels as the Monte Carlo approximation to the Fourier transform of a probability measure. Under the property of kernel stationarity such that $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}')$, we can derive the RFF approximate to a real-valued kernel as:

$$\begin{aligned} k(\mathbf{x} - \mathbf{x}') &= \int_{\mathcal{R}^d} p(\boldsymbol{\omega}) e^{i\boldsymbol{\omega}(\mathbf{x}-\mathbf{x}')}\, d\boldsymbol{\omega}, \\ &= \int_{\mathcal{R}^d} p(\boldsymbol{\omega}) \cos(\boldsymbol{\omega}(\mathbf{x} - \mathbf{x}'))\, d\boldsymbol{\omega} \\ &\approx \frac{1}{R} \sum_{r=1}^{R} \cos(\boldsymbol{\omega}_r^T(\mathbf{x} - \mathbf{x}'))\ , \end{aligned} \tag{9}$$

where $\boldsymbol{\omega}_r \sim p(\boldsymbol{\omega})$.

In practice, it is common to evolve (9) a further step and represent kernel evaluations as $k(\mathbf{x} - \mathbf{x}') = \Phi(\mathbf{x})^T \Phi(\mathbf{x}')$ where:

$$\Phi(\mathbf{x}) = \frac{\sqrt{2}}{\sqrt{2R}} \begin{bmatrix} \cos(\boldsymbol{\omega}_1^T \mathbf{x}) \\ \sin(\boldsymbol{\omega}_1^T \mathbf{x}) \\ \vdots \\ \cos(\boldsymbol{\omega}_R^T \mathbf{x}) \\ \sin(\boldsymbol{\omega}_R^T \mathbf{x}) \end{bmatrix}, \tag{10}$$

for which the expectation exactly evaluates to the Monte Carlo estimate in (9). Many common kernels are, in fact, the result of applying Bochner's theorem to specific measures $p(\boldsymbol{\omega})$: as an example, the RBF is the result of the Fourier transform of a Gaussian distribution $p(\boldsymbol{\omega}) \sim \mathcal{N}(0, 1)$.

The RFF formulation offers a more flexible setting for defining stationary kernels essential for Gaussian processes and other kernel methods. In this framework, each kernel is uniquely characterized by a probability measure, thereby enabling the vast tool set for parameterization of probability distributions for use in defining any valid stationary kernel – provided we can sample from $p(\boldsymbol{\omega})$. Alternatively, samples $\boldsymbol{\omega}$ can serve as learnable hyperparameters in the GP hyperparameter selection process, obviating the need to specify $p(\boldsymbol{\omega})$ directly.

Sparse Spectrum Gaussian Processes Besides kernel flexibility, the RFF approach also addresses the computational bottleneck in GPs, namely the $\mathcal{O}(N^3)$ computational complexity for Gram matrix $\mathbf{K}$ inversion in posterior-predictive estimates. A low-rank approximation proposed in Lázaro-Gredilla et al., 2010 scales in $\mathcal{O}(R^3)$ time, where R is the number of Fourier features, and typically $R << N$. This is achieved by re-expressing the GP posterior-predictive (4) using the

RFF representations in (10):

$$\begin{aligned}\mu(f*) &= \Phi(\mathbf{x}^*)^T \mathbf{A}^{-1}\Phi(\mathbf{X})\mathbf{y} \\ \mathbf{A} &= \Phi(\mathbf{X})\Phi(\mathbf{X})^T\end{aligned} \tag{11}$$

where $\mathbf{A} \in \mathbb{R}^{R\times R}$, rather than $\mathbf{K} \in \mathbb{R}^{N\times N}$, as in traditional GP inference. Typically, R is held constant at a value significantly lower than the number of data points N or increased at a much-reduced rate.

There have been numerous approaches proposed to learn the optimal $p(\boldsymbol{\omega})$ or $\boldsymbol{\omega}$ directly for a given problem setting using the RFF representation (Oliva et al., 2016; Chang et al., 2017; Tompkins et al., 2019; Li et al., 2019; Xie et al., 2019). Regardless of the method chosen, the final kernel representation shown in Equations 9 and 10 remain primarily consistent across these methods, allowing for easy implementation of different learning schemes in a unified setting.

2.4 Spectral Bayesian Quadrature

Motivated by the flexibility and success of the RFF approach in GP modeling, recent work (Warren et al., 2022; Sellier and Dellaportas, 2023) has proposed reformulating the BQ problem using kernels and measures represented through RFFs. Specifically, the first such approach, dubbed *generalized Bayesian quadrature* (GBQ), reformulates the BQ integral posterior mean estimate (6) as:

$$\begin{aligned}\langle \bar{f} \rangle = \frac{\mathbf{y}^T\mathbf{K}^{-1}}{RZ\sqrt{(2\pi)^d|\boldsymbol{\Sigma}|}} \int_{\mathbf{x}\in\mathcal{R}} \sum_{r=1}^{R} \cos(\boldsymbol{\omega}_r^T(\mathbf{x}-\mathbf{X})) \\ \times \sum_{z=1}^{Z} \cos(\boldsymbol{\rho}_z^T(\mathbf{x}-\boldsymbol{\mu}))d\mathbf{x},\end{aligned} \tag{12}$$

where the first integrand term is the RFF approximation to a stationary kernel. The second integrand term is the RFF representation of a Gaussian measure, which the authors define as the RFF approximation to an RBF kernel, through samples $\boldsymbol{\rho}$, which is then normalized by the appropriate Gaussian normalization term $[(2\pi)^d|\boldsymbol{\Sigma}|]^{-1/2}$.

The authors show that the integral in (12) can be analytically solved under such a formulation, thus yielding the closed-form solution to the BQ posterior for RFF kernels.

3 FAST AND LOW-RANK SPECTRAL BAYESIAN QUADRATURE

In the preceding section, we introduced foundational concepts that pave the way for our contribution: two novel methodologies enhancing the spectral BQ framework's flexibility and scalability.

Unlike prior approaches that primarily consider BQ in the context of online integration – constructing integral posteriors $\langle \bar{f} \rangle$ via adaptive sampling strategies (Gunter et al., 2014; Adachi et al., 2022; Hayakawa et al., 2022) – our focus shifts towards the offline scenario. Our developments build upon recent efforts by (Warren et al., 2022; Sellier and Dellaportas, 2023) aimed at augmenting the core BQ methodology's adaptability and computational efficiency. Although primarily tailored for offline applications, our methods hold potential for integration into adaptive sampling frameworks, a prospect reserved for future investigation.

We begin with the observation that the traditional BQ kernel mean $\mu_{\mathbf{x}}(\mathbf{X}) = \int k(\mathbf{x}-\mathbf{X})p(\mathbf{x})d\mathbf{x}$ is a convolution of a kernel function with a probability measure, thus enabling the use of the convolution theorem to solve the integral.

Theorem 2 (Convolution theorem (McGillem and Cooper, 1991))**.** *The convolution of functions g and h in the spatial domain is equivalent to the inverse Fourier transform of their point-wise multiplication in the frequency domain:*

$$\begin{aligned}(g * h)(\mathbf{x}) &= \int g(\mathbf{x}-\tau)h(\tau)d\tau \\ &= \mathcal{F}^{-1}\big[\mathcal{F}[g] \times \mathcal{F}[h]\big](\mathbf{x}),\end{aligned} \tag{13}$$

where $\mathcal{F}[g]$ and $\mathcal{F}^{-1}[g]$ denote the Fourier and inverse Fourier transforms of function g, respectively.

Using this result, we will derive our two novel methodologies: fast Fourier Bayesian quadrature (FFBQ) and sparse spectrum Bayesian quadrature (SSBQ).

3.1 Fast Fourier Bayesian Quadrature

Using Bochner's Theorem (1), we note that in the BQ kernel mean setting $\mathcal{F}[k](\boldsymbol{\omega}) = p_k(\boldsymbol{\omega})$, while $\mathcal{F}[p](\boldsymbol{\omega}) = k_p(\boldsymbol{\omega})$, where p_k is the Fourier dual probability measure of kernel k, and k_p is the kernel resulting from the Fourier transform of measure p. Using these spectral representations and the convolution theorem, we can rewrite the BQ kernel mean $\mu_{\mathbf{x}}(\mathbf{X})$ in (6) as:

$$\begin{aligned}\mu_{\mathbf{x}}(\mathbf{X}) &= \int\int k(\mathbf{x},\mathbf{X})p(\mathbf{x})d\mathbf{x} \\ &= \int k(\mathbf{X}-\mathbf{x})p(\mathbf{x})d\mathbf{x} \\ &\equiv \mathcal{F}^{-1}\big[\mathcal{F}[k] \circ \mathcal{F}[p]\big](\mathbf{X}) \\ &\equiv \mathcal{F}^{-1}\big[p_k \circ k_p\big](\mathbf{X}),\end{aligned} \tag{14}$$

where $\circ$ denotes point-wise multiplication. We can likewise reformulate the kernel mean variance term

$\mu_{\mathbf{x}\mathbf{x}'}$ in (7) as:

$$\begin{aligned}\mu_{\mathbf{x}\mathbf{x}'} &= \int k(\mathbf{x},\mathbf{x}')p(\mathbf{x})p(\mathbf{x}')d\mathbf{x}d\mathbf{x}' \\ &= \int \mu_{\mathbf{x}}(\mathbf{x}')p(\mathbf{x}')d\mathbf{x}' \\ &= \int \mu_{\mathbf{x}}(\mathbf{0}-\mathbf{x}')p(\mathbf{x}')d\mathbf{x}' \\ &\equiv \mathcal{F}^{-1}\big[\mathcal{F}[\mu_{\mathbf{x}}]\circ\mathcal{F}[p]\big](\mathbf{0}) \\ &\equiv \mathcal{F}^{-1}\big[p_k\circ k_p\circ k_p\big](\mathbf{0})\end{aligned} \tag{15}$$

We note that the result of (15) is a convolution over the result of (14) evaluated at $\mathbf{0}$.

There are several advantages to this reformulation of BQ. First, analytical Fourier transforms exist for many commonly used kernels and measures, enhancing flexibility compared to traditional BQ. Secondly, when analytical forms are unavailable, we can employ the fast Fourier transform (FFT) to affordably approximate spectral representations, extending BQ kernel options to include stationary kernels defined in any manner.

Definition 1 (Fast Fourier Transform (Cooley and Tukey, 1965))**.** *Let* $(x = (x_0,\ldots,x_{Z-1})$ *be a sequence of* Z *complex numbers. The Fast Fourier Transform* $\mathrm{FFT}\big[x\big]$ *computes the sequence* $X = (X_0,\ldots,X_{Z-1})$ *in* $\mathcal{O}(Z\log Z)$ *time, where* $X_k = \sum_{z=0}^{Z-1} x_z e^{-i2\pi kz/Z}$.

We next formally present our proposed FFBQ method, which incorporates Equations 14 and 15 into the larger context of BQ. We present our BQ integral mean methodology below and include methodology for integral variance in the supplement.

Definition 2 (Fast Fourier Bayesian Quadrature)**.** *Given data* $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N = \{\mathbf{X},\mathbf{y}\}$*, stationary kernel* $k_{\boldsymbol{\theta}}$ *with hyperparameters* $\boldsymbol{\theta}$*, and probability measure* $p_{\boldsymbol{\gamma}}$ *with parameters* $\boldsymbol{\gamma}$*, and a uniformly-spaced grid of* Z *coordinates* $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_z\}_{z=1}^Z$ *taken over the domain of integration, we can approximate the BQ posterior mean (6) as:*

$$\langle\bar{f}\rangle = \mathbf{y}^T\mathbf{K}^{-1}\mu_{\mathbf{x}}(\mathbf{X}) \approx \mathbf{y}^T\mathbf{K}^{-1}\hat{\mu}_{\mathbf{x}}(\mathbf{X}), \tag{16}$$

where

$$\hat{\mu}_{\mathbf{x}}(\mathbf{X}) \coloneqq \psi\Big[\mathrm{FFT}^{-1}\Big[\dot{\mathcal{F}}[k_{\boldsymbol{\theta}}]\circ\dot{\mathcal{F}}[p_{\boldsymbol{\gamma}}]\Big](\boldsymbol{\Lambda})\Big](\mathbf{X}), \tag{17}$$

in which $\dot{\mathcal{F}}$ *signifies either the analytical Fourier transform or fast Fourier transform, and* $\psi[\cdot](\mathbf{X})$ *is any interpolation operator evaluated at* $\mathbf{X}$.

The interpolation function ψ in (17) is necessary because the FFT operates on uniformly sampled d-dimensional grids $\boldsymbol{\Lambda}$, while integrand samples $\mathcal{D} = \{\mathbf{X},\mathbf{y}\}$ may be non-uniform. Section 4 shows that this interpolation step negligibly impacts FFBQ's approximation accuracy.

Definition 2 confers three advantages over Monte Carlo methods for computing kernel means. First, its data efficiency in the spectral domain, even at modest Z, achieves higher accuracy with fewer samples and allows for a higher accuracy ceiling, as validated in Section 4. Second, the algorithmic efficiency of FFT enables GPU acceleration and rapid evaluations for large Z. Third, it allows practitioners flexibility in choosing any valid stationary kernel or measure and leveraging analytical Fourier transforms if available, all without altering the formula. FFBQ is applicable as long as the kernel and measure can be forward-evaluated for grid points $\boldsymbol{\Lambda}$ and eliminates the need for measure sampling, a frequent issue in Bayesian posterior estimation.

3.2 Sparse Spectrum Bayesian Quadrature

We additionally extend spectral BQ to make use of the efficient low-rank sparse spectrum GP in Equation 11 and derive here the equivalent low-rank BQ posterior integral estimate resulting from this inclusion.

Substituting the sparse spectrum GP posterior mean (11) for the full-rank GP posterior mean (4) in the BQ integral posterior mean (6) yields:

$$\begin{aligned}\langle\bar{f}\rangle &\approx \int \mathbf{y}^T\Phi(\mathbf{x})^T\mathbf{A}^{-1}\Phi(\mathbf{X})p(\mathbf{x})d\mathbf{x} \\ &\approx \int \mathbf{y}^T\Phi(\mathbf{X})^T\mathbf{A}^{-1}\Phi(\mathbf{x})p(\mathbf{x})d\mathbf{x} \\ &\approx \mathbf{y}^T\Phi(\mathbf{X})^T\mathbf{A}^{-1}\int \Phi(\mathbf{x})p(\mathbf{x})d\mathbf{x}.\end{aligned} \tag{18}$$

Using this result, with the associated integral estimate variance provided in the supplement, we define SSBQ:

Definition 3 (Sparse-Spectrum Bayesian Quadrature)**.** *Given data* $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N = \{\mathbf{X},\mathbf{y}\}$*, stationary RFF-defined kernel* $k_{\boldsymbol{\omega}}$ *formed by* R *samples of the kernel's spectral distribution* $p_k(\boldsymbol{\omega})$*, probability measure* $p_{\boldsymbol{\gamma}}$ *with parameters* $\boldsymbol{\gamma}$*, and a uniformly-spaced grid of* Z *coordinates* $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_z\}_{z=1}^Z$ *taken over the domain of integration, the low-rank SSBQ approximation to the BQ posterior mean is:*

$$\begin{aligned}\langle\bar{f}\rangle &= \int \mathbf{y}^T\Phi(\mathbf{X})^T\mathbf{A}^{-1}\Phi(\boldsymbol{\lambda})p(\boldsymbol{\lambda})d\boldsymbol{\lambda} \\ &\approx \mathbf{y}^T\Phi(\mathbf{X})^T\mathbf{A}^{-1}\hat{\Phi}(\boldsymbol{\Lambda}),\end{aligned} \tag{19}$$

where

$$\begin{aligned}\hat{\Phi}(\boldsymbol{\Lambda}) &= \Big[\psi\big[\hat{\Phi}_1(\boldsymbol{\Lambda})\big](\mathbf{0}),\ \ldots\ ,\psi\big[\hat{\Phi}_R(\boldsymbol{\Lambda})\big](\mathbf{0})\Big]^T, \\ \hat{\Phi}_r(\boldsymbol{\Lambda}) &\coloneqq \mathrm{FFT}^{-1}\Big[\mathrm{FFT}\big[\Phi_r\big]\circ\dot{\mathcal{F}}[p_{\boldsymbol{\gamma}}]\Big](\boldsymbol{\Lambda})\end{aligned} \tag{20}$$

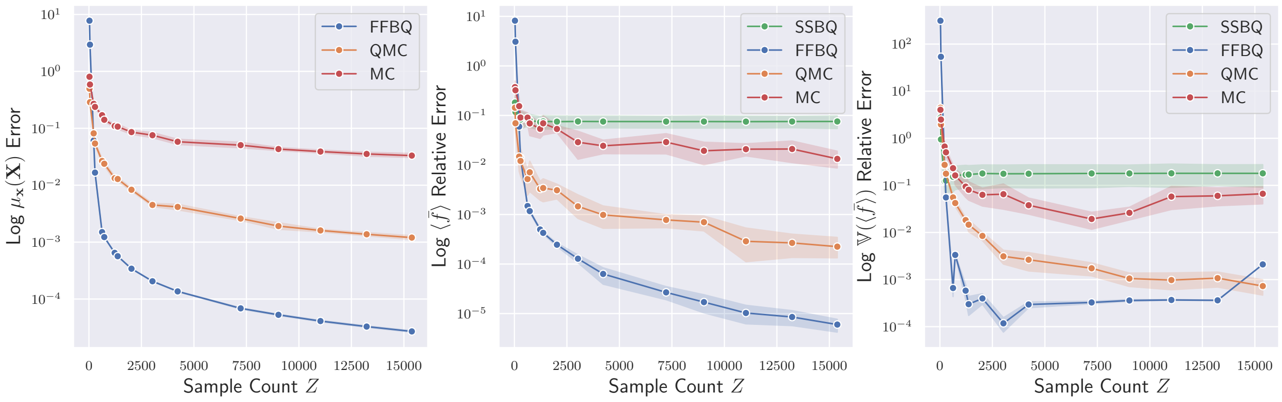

Figure 1: 2D Gaussian Mixture Analytical BQ Approximation Error of Kernel Mean, Integral Mean, and Integral Variance.

if using FFT convolution, or

$$\hat{\Phi}_r(\mathbf{\Lambda}) = \frac{1}{Z} \sum_{z=1}^{Z} \Phi_r(\mathbf{\lambda}_z) p_{\boldsymbol{\gamma}}(\mathbf{\lambda}_z) \qquad (21)$$

if using Monte Carlo convolution. $\mathbf{\Phi}$ *and* $\mathbf{\Lambda}$ *are defined as in Equation 11, and the transform operator* $\dot{\mathcal{F}}$ *and interpolation operator* $\psi[\cdot](\mathbf{X})$ *are as defined in Definition 2.*

SSBQ offers computational benefits beyond FFBQ, executing the BQ posterior integral in $\mathcal{O}(R^3)$ time while retaining desirable approximation error (Sutherland and Schneider, 2015). Similar to FFBQ, SSBQ permits any stationary kernel or measure without altering the formulation. The distribution p_k can be changed for a sampling-based approach, or $\boldsymbol{\omega}$ can be directly optimized for a parametric approach.

3.3 Computational Complexity and Considerations

In traditional analytical BQ, the computational complexity is primarily governed by Gram matrix inversion, with a complexity of $\mathcal{O}(N^3)$. However, when dealing with non-analytical kernel/measure pairs, the complexity will vary depending on kernel mean approximation method.

If we consider MC kernel mean approximation, we can observe complexities of $\mathcal{O}(NZ)$ for full-rank GPs and $\mathcal{O}(RZ)$ in the low-rank scenario, with Z representing the number of MC points. Due to the curse of dimensionality, Z can expand exponentially with dimension d, making kernel mean approximation the dominant factor in BQ computational complexity when $N^2 << Z$ or $R^2 << Z$, which is a common occurrence in high-dimensional spaces.

This varying complexity motivates the introduction of both FFBQ and SSBQ to address the dual BQ computational challenges of kernel mean approximation and Gram matrix inversion respectively. While FFT convolution scales in $\mathcal{O}(N \log N)$ compared to linear complexity of MC, our experiments shown in Figure 2 demonstrate that FFBQ convolution offers computational improvements over MC in terms of sample-efficiency in Z.

For SSBQ, we present methods for both FFT convolution and MC approximation for computing the feature map mean $\hat{\Phi}(\mathbf{\Lambda})$, tailoring the approach to the integration measure and computational context. Analytical solutions for common measures like Gaussian or uniform distributions streamline the process, but alternative measures may require R independent FFT convolutions. The choice between FFT and MC for SSBQ is problem-specific, but to our knowledge, both approaches are novel techniques for BQ using sparse spectrum GP approximation.

Together, the FFBQ and SSBQ methodologies presented here represent significant computational and flexibility improvements to traditional BQ that address the dual challenges of non-analytical BQ. Detailed derivations of integral variance $\mathbb{V}(\langle \bar{f} \rangle)$ for both methodologies are provided in the supplementary material.

4 EXPERIMENTS

We evaluate our methods against various benchmark problems, using models that vary across two axes: (1) GP kernel choice; and (2) kernel mean *operator* for solving or approximating $\mu_{\mathbf{x}}(\mathbf{X})$ and $\mu_{\mathbf{xx}'}$.

Baseline operators include analytical BQ with the Gaussian measure and RBF kernel, Monte Carlo (MC), and quasi-Monte Carlo (QMC) kernel mean ap-

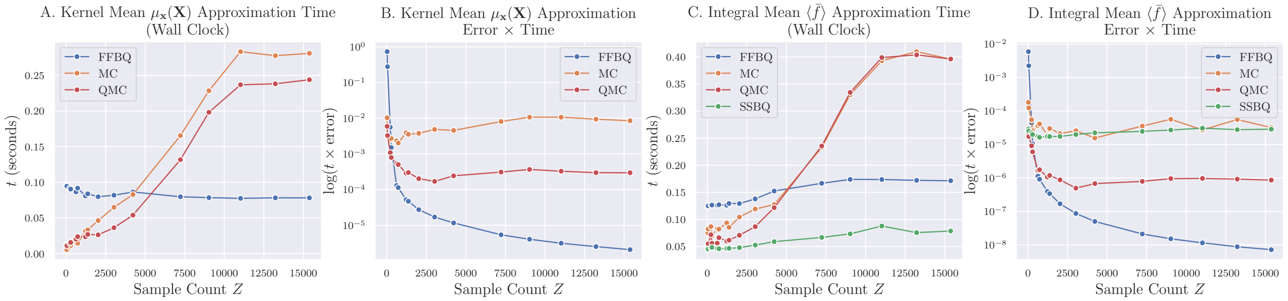

Figure 2: 2D Gaussian Mixture Wall-Clock and Sample Efficiency Ablation.

proximations. Hardware limitations cap the maximum sample size Z at 100,000 for full BQ and 500,000 for low-rank BQ in all Monte Carlo implementations, including SSBQ (21). FFBQ faced no such limitations, likely due to the optimized and GPU-parallelized nature of the FFT, constituting an implicit advantage.

We use PCHIP (Fritsch and Butland, 1984) as the interpolation function ψ in Definitions 2 and 3, leveraging its fast, performant implementations. Experiments are conducted using Jax, and all code and results are publicly available[1].

4.1 FFBQ and SSBQ Approximation Fidelity & Computational Performance

We evaluate the performance of FFBQ and SSBQ in approximating standard BQ results with analytical kernel and measure pairs. Specifically, we look at the problem of integral approximation over the infinite bounds of a 2D Gaussian mixture distribution using an RBF kernel and Gaussian measure. We present results across Z: the number of samples (or uniform grid points) $\mathbf{\Lambda}$ used by the operator for kernel mean approximation.

In this study, we aim to quantify the error introduced by FFBQ and SSBQ approximation methods to the traditional BQ algorithm, and therefore use the BQ analytical solutions as our ground truth rather than the true integral solution. Figure 1 presents our results, where we assess:

1. The quality of FFBQ kernel mean estimates $\hat{\mu}_{\mathbf{x}}(\mathbf{X})$.
2. The accuracy of integral mean estimates $\langle \bar{f} \rangle$.
3. Reliability in estimating the integral variance $\mathbb{V}(\langle \bar{f} \rangle)$.

We measure error using absolute relative error against analytical BQ solutions. For evaluating the quality of kernel mean approximations, the relative vector norm metric, $\frac{||\mu_{\mathbf{x}}(\mathbf{X})_{\text{BQ}} - \hat{\mu}_{\mathbf{x}}(\mathbf{X})||}{||\mu_{\mathbf{x}}(\mathbf{X})_{\text{BQ}}||}$ is used. In addition, we present a wall-clock computational comparison between kernel mean approximation methods in Figure 2.

FFBQ shows superior performance in approximating the kernel mean, integral mean, and integral variance over the vast majority of Z. In addition, FFBQ offers improved computational scaling (wall-clock) over MC and QMC, and exhibits significant advantages over baselines when approximation quality is jointly considered with computation time.

SSBQ also shows strong performance despite using only 10% of the full-rank size. We study this behavior further in Figure 3, measuring the approximation accuracy of SSBQ as a function of sparsity ratio $\frac{R}{N}$. SSBQ achieves less than 5% relative error while using a covariance matrix that is only 30% of the full-rank size, offering promise for more computationally scalable BQ methods.

4.2 Genz Integration Benchmarks: FFBQ and SSBQ Kernel Flexibility

Next, we evaluate the performance of FFBQ and SSBQ, using an expanded set of kernel families that are not analytically tractable in traditional BQ, on a variety of d-dimensional Genz (Genz, 1984) bounded integration benchmarks. Specifically, we adopt the Genz continuous, discontinuous, and oscillatory benchmarks to study FFBQ and SSBQ's integration capability across varied integrand geometries.

For MC, QMC, FFBQ, and SSBQ, we implement the RBF, the Matern 3/2, and fully parametric RFF kernels. We implement the full-rank RFF kernel as a product kernel multiplied by the bounded kernel from Melkumyan and Ramos, 2009, which we observed to produce more stable integral approximations. For SSBQ, we report results for the QMC variant in Equation 21. We evaluate all possible combinations of

[1] https://github.com/houstonwarren/ffbq

	$d = 2$	$d = 4$	$d = 6$
		Genz Continuous	
BQ-RBF	2.39e-03 ± 1.32e-03	2.81e-03 ± 2.00e-03	1.34e-03 ± 1.26e-03
QMC-RFF	**2.10e-03 ± 1.40e-03**	1.64e-03 ± 1.46e-03	1.55e-03 ± 7.24e-04
FFBQ-RFF	2.21e-03 ± 1.35e-03	**1.53e-03 ± 1.22e-03**	1.35e-03 ± 7.46e-04
SSBQ-RFF	5.92e-03 ± 4.82e-03	3.31e-03 ± 2.03e-03	**1.22e-03 ± 1.22e-03**
		Genz Discontinuous	
BQ-RBF	1.12e-01 ± 7.94e-02	**4.17e+00 ± 3.71e+00**	**6.02e+01 ± 3.27e+01**
QMC-M3/2	**1.02e-01 ± 9.67e-02**	1.00e+01 ± 5.97e+00	7.42e+01 ± 3.06e+01
FFBQ-M3/2	1.07e-01 ± 1.07e-01	8.91e+00 ± 4.07e+00	7.22e+01 ± 3.29e+01
SSBQ-RFF	1.06e-01 ± 1.05e-01	6.41e+00 ± 3.30e+00	6.46e+01 ± 3.92e+01
		Genz Oscillatory	
BQ-RBF	**2.83e-05 ± 1.37e-05**	7.09e-04 ± 4.70e-04	1.10e-02 ± 9.39e-03
QMC-RFF	3.63e-05 ± 1.32e-05	1.98e-03 ± 1.15e-03	3.92e-03 ± 4.12e-03
FFBQ-RFF	6.84e-04 ± 3.98e-04	3.06e-03 ± 1.96e-03	6.09e-03 ± 6.11e-03
SSBQ-RFF	1.03e-04 ± 9.07e-05	**3.05e-04 ± 3.87e-04**	**4.96e-04 ± 3.42e-04**

Table 1: 10-Fold Mean Absolute Integration Error With Standard Deviations for Genz Experiments.

kernel families and kernel mean operators; however, some combinations are restricted (such as using non-analytical kernel/measure combinations within vanilla BQ).

Importantly, all these experiments are carried out under strictly controlled conditions, with identical hyperparameters, data sizes ($N = 1000$, $R = 100$), learning rates, training epoch counts, and diagonal Gram matrix noise to ensure a fair comparison. The efficacy of the Bayesian quadrature scheme is well-supported (Briol et al., 2019), and the intention in this study is not to focus on magnitude of error – which can be strongly dependent on these factors – but rather the relative errors between methods under identical circumstances.

For each kernel mean operator, the best-performing kernel is selected based on 10-fold cross-validated GP negative-log likelihood across a range of data dimensionalities ($d \in [2, 6]$). This approach reflects a real-world scenario where practitioners would choose a kernel based on empirical performance, as analytical integral solutions for model selection are generally unavailable. The integration measures used are either Gaussian or uniform distributions, chosen based on prior knowledge of the integrand geometry.

The ultimate goal is to assess whether FFBQ and SSBQ can match or exceed the performance of established Bayesian quadrature schemes while offering flexibility or computational efficiencies. A high sample count Z is used to isolate the benefits of increased kernel flexibility within the BQ framework. Mean and standard deviations of errors across ten random seeds are included in Table 1, and full experimental setup details are provided in the supplement.

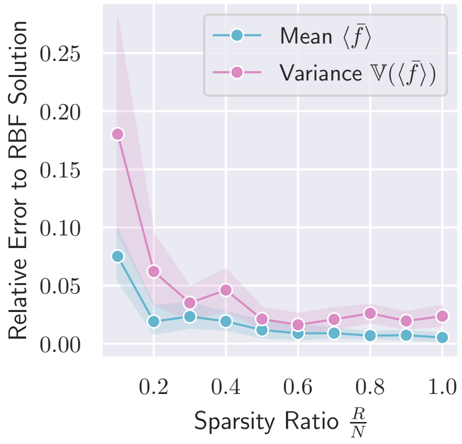

Figure 3: 2D Gaussian Mixture SSBQ Relative Approximation Error of Analytical BQ Across Low-Rank Size R / Full-Rank Size N.

The results show that FFBQ and SSBQ tend to beat other baselines as data dimensionality increases. The value of kernel flexibility is especially apparent in the Genz oscillatory benchmark, where the ideal periodic kernel can instead be captured through parametric RFF representation. SSBQ, in some instances, achieves the best performance across methods despite its low rank, especially in higher dimensions. This behavior is worthy of deeper analysis in future work.

A limitation we observed in all BQ methods, but particularly in SSBQ and FFBQ, is their sensitivity to the parameters of the integration measure and the magnitude of the jitter term added to the Gram matrix $\mathbf{K}$ during the calculation of the BQ posterior. We provide a brief ablation study in the supplement evaluating this behavior, and confirm that FFBQ and SSBQ converge to analytical BQ solutions under a wide range of reasonable jitter terms.

Additionally, although the FFT *enables* FFBQ to scale to large kernel mean sample sizes Z, it also *necessitates* high Z as dimensionality grows, given that the number of points in the uniform grid over which the FFT is performed grows exponentially with d. Even though Bayesian Quadrature methods are often limited to problems with $d < 10$, FFBQ can quickly become intractable as we approach this limit. In such cases, SSBQ paired with QMC methods are more computationally feasible, and our experiments demonstrate that they can result in superior performance.

However, we note that for isotropic (separable) integration measures or kernels, multi-d FFT convolution can be carried out as 1-d FFT convolutions in each dimension, which greatly reduces the FFBQ computational footprint. We adopt this approach where appropriate in our implementations.

4.3 Comparing SSBQ and Nyström for Low-Rank BQ

Lastly, we perform an ablation study of SSBQ against an alternative low-rank GP methodology in Nyström approximation. Specifically, we adapt the sparse variational GP (SVGP) approach of Titsias, 2009, in which R GP inducing point locations are trained through variational inference and subsequently used for low-rank inference. We use a problem setting of integration over a 2D periodic signal (Pinder and Dodd, 2022):

$$f(\mathbf{x}) = \sum_{d=1}^{2} \sin(2\mathbf{x}_d) + \mathbf{x}_d \cos(5\mathbf{x}_d) \tag{22}$$

We use QMC kernel mean approximation for both SSBQ and SVGP, and provide details of our SVGP BQ implementation in the supplement. We perform SSBQ using the RFF approximation to the RBF kernel as well as a fully parametric RFF in which we train frequencies $\boldsymbol{\omega}$. For the SVGP, we use the RBF and $\sin^2$ (periodic) kernels in order to provide similar inductive biases to the SSBQ implementations. We present results in Table 2 for sparsity ratios of $\frac{R=100}{N=1000}$ and $\frac{R=100}{N=5000}$, with means and standard deviations over 10 seeds. We additionally include standard MC integration as a baseline.

We can see that both the RBF and RFF variants of SSBQ outperform the SVGP approaches, with the RFF approach significantly outperforming other methods due to the full kernel flexibility and the periodic problem setting.

	$N = 1000$	$N = 5000$
MC	1.36 ± 1.24	0.65 ± 0.68
SVGP-RBF	1.59 ± 1.44	0.65 ± 0.70
SVGP-$\sin^2$	1.72 ± 1.24	0.65 ± 0.68
SSBQ-RBF	0.48 ± 0.20	0.16 ± 0.13
SSBQ-RFF	**0.02** ± **0.02**	**0.02** ± **0.01**

Table 2: Low-Rank BQ Integration RMSE, 2D Periodic Signal.

In the $N = 5000$ setting, SSBQ-RBF and SVGP-RBF GPs respectively had integrand GP RMSE's of 0.342 and 0.259. However, SSBQ-RBF significantly outperforms SVGP-RBF in the integration setting despite having a lower integrand GP accuracy. We hypothesize the spectral domain offers unique benefits for BQ as RFFs span the full GP domain in frequency space, while SVGP inducing points are inherently local. Further theoretical and empirical study of this conjecture is a compelling direction for future research.

5 CONCLUSION

This paper presents a novel reformulation of Bayesian quadrature as spectral convolution, from which we derive fast Fourier Bayesian quadrature and sparse spectrum Bayesian quadrature. We derived a method using the fast Fourier transform to approximate the BQ posterior mean and variance for any stationary kernel. We also presented a method to leverage the low-rank approximation of the kernel Gram matrix for improved BQ computational scaling. Finally, we demonstrated these methods' effectiveness on a variety of integration problems.

These contributions lay the groundwork for incorporating recent advancements in spectral GP representations into BQ. Their straightforward implementations, flexibility in kernel and measure selection, and compatibility with various kernel learning techniques can facilitate future BQ developments. Potential future studies might explore theoretical aspects of low-rank BQ in both spectral and spatial domains and adapt these methods to online BQ settings.

References

Adachi, Masaki et al. (Dec. 2022). "Fast Bayesian Inference with Batch Bayesian Quadrature via Kernel Recombination". In: *Advances in Neural Information Processing Systems* 35, pp. 16533–16547.

Briol, François-Xavier et al. (2019). “Probabilistic Integration: A Role in Statistical Computation?” In: *Statistical Science* 34.1, pp. 1–22.

Chang, Wei-Cheng et al. (May 23, 2017). “Data-Driven Random Fourier Features Using Stein Effect”. arXiv: `1705.08525 [cs, stat]`.

Cockayne, Jon et al. (Jan. 2019). “Bayesian Probabilistic Numerical Methods”. In: *SIAM Review* 61.4, pp. 756–789.

Cooley, James W. and John W. Tukey (1965). “An Algorithm for the Machine Calculation of Complex Fourier Series”. In: *Mathematics of Computation* 19.90, pp. 297–301. JSTOR: `2003354`.

Fritsch, F. N. and J. Butland (June 1984). “A Method for Constructing Local Monotone Piecewise Cubic Interpolants”. In: *SIAM Journal on Scientific and Statistical Computing* 5.2, pp. 300–304.

Genz, Alan (Sept. 1, 1984). “Testing Multidimensional Integration Routines”. In: *Proc. of International Conference on Tools, Methods and Languages for Scientific and Engineering Computation*, pp. 81–94.

Ghahramani, Zoubin and Carl E. Rasmussen (2003). “Bayesian Monte Carlo”. In: *Advances in Neural Information Processing Systems*. Vol. 15.

Gunter, Tom et al. (2014). “Sampling for Inference in Probabilistic Models with Fast Bayesian Quadrature”. In: *Advances in Neural Information Processing Systems*. Vol. 27. Curran Associates, Inc.

Hayakawa, Satoshi, Harald Oberhauser, and Terry Lyons (Dec. 2022). “Positively Weighted Kernel Quadrature via Subsampling”. In: *Advances in Neural Information Processing Systems* 35, pp. 6886–6900.

He, Bobby, Balaji Lakshminarayanan, and Yee Whye Teh (2020). “Bayesian Deep Ensembles via the Neural Tangent Kernel”. In: *Advances in Neural Information Processing Systems* 33, pp. 1010–1022.

Hensman, James, Alexander Matthews, and Zoubin Ghahramani (Feb. 2015). “Scalable Variational Gaussian Process Classification”. In: *Proceedings of the Eighteenth International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 351–360.

Lázaro-Gredilla, Miguel et al. (2010). “Sparse Spectrum Gaussian Process Regression”. In: *Journal of Machine Learning Research* 11.63, pp. 1865–1881.

Le, Quoc, Tamas Sarlos, and Alexander Smola (May 26, 2013). “Fastfood - Computing Hilbert Space Expansions in Loglinear Time”. In: *Proceedings of the 30th International Conference on Machine Learning*, pp. 244–252.

Li, Chun-Liang et al. (Feb. 26, 2019). *Implicit Kernel Learning*. arXiv: `1902.10214 [cs, stat]`. preprint.

Liu, Haitao et al. (Apr. 9, 2019). “When Gaussian Process Meets Big Data: A Review of Scalable GPs”. arXiv: `1807.01065 [cs, stat]`.

McGillem, Clare D. and George R. Cooper (1991). *Continuous and Discrete Signal and System Analysis*. 3rd ed. 494 pp.

Melkumyan, Arman and Fabio Ramos (July 11, 2009). “A Sparse Covariance Function for Exact Gaussian Process Inference in Large Datasets”. In: *Proceedings of the 21st International Joint Conference on Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 1936–1942.

O’Hagan, A. (Nov. 1, 1991). “Bayes–Hermite Quadrature”. In: *Journal of Statistical Planning and Inference* 29.3, pp. 245–260.

Ober, Sebastian W., Carl E. Rasmussen, and Mark van der Wilk (Dec. 1, 2021). “The Promises and Pitfalls of Deep Kernel Learning”. In: *Proceedings of the Thirty-Seventh Conference on Uncertainty in Artificial Intelligence*, pp. 1206–1216.

Oliva, Junier B. et al. (May 2, 2016). “Bayesian Nonparametric Kernel-Learning”. In: *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 1078–1086.

Ott, Katharina et al. (July 2, 2023). “Baysian Numerical Integration with Neural Networks”. In: *Proceedings of the Thirty-Ninth Conference on Uncertainty in Artificial Intelligence*, pp. 1606–1617.

Pinder, Thomas and Daniel Dodd (2022). “GPJax: A Gaussian Process Framework in JAX”. In: *Journal of Open Source Software* 7.75, p. 4455.

Rahimi, Ali and Benjamin Recht (2008a). “Random Features for Large-Scale Kernel Machines”. In: *Advances in Neural Information Processing Systems*. Vol. 20.

– (2008b). “Weighted Sums of Random Kitchen Sinks: Replacing Minimization with Randomization in Learning”. In: *Advances in Neural Information Processing Systems*. Vol. 21.

Raissi, Maziar and George Em Karniadakis (Mar. 15, 2018). “Hidden Physics Models: Machine Learning of Nonlinear Partial Differential Equations”. In: *Journal of Computational Physics* 357, pp. 125–141.

Rasmussen, Carl E. and Christopher Williams (2006). *Gaussian Processes for Machine Learning*. Cambridge, Mass: MIT Press. 248 pp.

Robert, Christian P. et al. (2018). “Accelerating MCMC Algorithms”. In: *Wiley Interdisciplinary Reviews. Computational Statistics* 10.5.

Rudin, Walter (2011). *Fourier Analysis on Groups.* Hoboken: John Wiley & Sons.

Sellier, Jeremy and Petros Dellaportas (Apr. 11, 2023). "Sparse Spectral Bayesian Permanental Process with Generalized Kernel". In: *Proceedings of The 26th International Conference on Artificial Intelligence and Statistics*, pp. 2769–2791.

Sutherland, Danica J. and Jeff Schneider (July 12, 2015). "On the Error of Random Fourier Features". In: *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, pp. 862–871.

Titsias, Michalis (Apr. 2009). "Variational Learning of Inducing Variables in Sparse Gaussian Processes". In: *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics.* PMLR, pp. 567–574.

Tompkins, Anthony et al. (Apr. 11, 2019). "Black Box Quantiles for Kernel Learning". In: *The 22nd International Conference on Artificial Intelligence and Statistics*, pp. 1427–1437.

Warren, Houston, Rafael Oliveira, and Fabio Ramos (June 18, 2022). "Generalized Bayesian Quadrature with Spectral Kernels". In: *Proceedings of the 38th Conference on Uncertainty in Artificial Intelligence.*

Wilson, Andrew and Ryan Adams (May 26, 2013). "Gaussian Process Kernels for Pattern Discovery and Extrapolation". In: *Proceedings of the 30th International Conference on Machine Learning*, pp. 1067–1075.

Xie, Jiaxuan et al. (Oct. 7, 2019). *Deep Kernel Learning via Random Fourier Features.* arXiv: `1910.02660 [cs, stat]`. preprint.

Zhu, Harrison et al. (2020). "Bayesian Probabilistic Numerical Integration with Tree-Based Models". In: *Advances in Neural Information Processing Systems.* Vol. 33, pp. 5837–5849.

Zhu, Shixiang et al. (Mar. 18, 2021). "Deep Fourier Kernel for Self-Attentive Point Processes". In: *International Conference on Artificial Intelligence and Statistics*, pp. 856–864.

Checklist

1. For all models and algorithms presented, check if you include:
   (a) A clear description of the mathematical setting, assumptions, algorithm, and/or model. [**Yes**/No/Not Applicable]
   (b) An analysis of the properties and complexity (time, space, sample size) of any algorithm. [**Yes**/No/Not Applicable]
   (c) (Optional) Anonymized source code, with specification of all dependencies, including external libraries. [**Yes**/No/Not Applicable]

2. For any theoretical claim, check if you include:
   (a) Statements of the full set of assumptions of all theoretical results. [**Yes**/No/Not Applicable]
   (b) Complete proofs of all theoretical results. [**Yes**/No/Not Applicable]
   (c) Clear explanations of any assumptions. [**Yes**/No/Not Applicable]

3. For all figures and tables that present empirical results, check if you include:
   (a) The code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL). [**Yes**/No/Not Applicable]
   (b) All the training details (e.g., data splits, hyperparameters, how they were chosen). [**Yes**/No/Not Applicable]
   (c) A clear definition of the specific measure or statistics and error bars (e.g., with respect to the random seed after running experiments multiple times). [**Yes**/No/Not Applicable]
   (d) A description of the computing infrastructure used. (e.g., type of GPUs, internal cluster, or cloud provider). [**Yes**/No/Not Applicable]

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets, check if you include:
   (a) Citations of the creator If your work uses existing assets. [Yes/No/**Not Applicable**]
   (b) The license information of the assets, if applicable. [Yes/No/**Not Applicable**]
   (c) New assets either in the supplemental material or as a URL, if applicable. [Yes/No/**Not Applicable**]
   (d) Information about consent from data providers/curators. [Yes/No/**Not Applicable**]

   (e) Discussion of sensible content if applicable, e.g., personally identifiable information or offensive content. [Yes/No/**Not Applicable**]

5. If you used crowdsourcing or conducted research with human subjects, check if you include:

   (a) The full text of instructions given to participants and screenshots. [Yes/No/**Not Applicable**]

   (b) Descriptions of potential participant risks, with links to Institutional Review Board (IRB) approvals if applicable. [Yes/No/**Not Applicable**]

   (c) The estimated hourly wage paid to participants and the total amount spent on participant compensation. [Yes/No/**Not Applicable**]

Fast Fourier Bayesian Quadrature: Supplementary Materials

1 FAST FOURIER BAYESIAN QUADRATURE (FFBQ) PROOF

In this section, we will provide the full derivations for the Fast Fourier Bayesian Quadrature (FFBQ) and Sparse Spectrum Bayesian Quadrature (SSBQ) mean estimate presented in the main text, as well as derive the variance of these estimators.

We briefly replicate here the traditional BQ mean and variance estimates from the main text. The BQ mean is defined as:

$$\begin{aligned}\langle \bar{f} \rangle &= \int k(\mathbf{x}, \mathbf{X})^T \mathbf{K}^{-1} \mathbf{y}\, p(\mathbf{x})\, d\mathbf{x} \\ &= \mathbf{y}^T \mathbf{K}^{-1} \int k(\mathbf{x}, \mathbf{X})\, p(\mathbf{x})\, d\mathbf{x} \\ &= \mathbf{y}^T \mathbf{K}^{-1} \mu_{\mathbf{x}}(\mathbf{X}),\end{aligned} \tag{1}$$

where $\mathbf{y} = f(\mathbf{x}) + \epsilon$ are integrand observations, $\mu_{\mathbf{x}}(\mathbf{X}) = \int k(\mathbf{x}, \mathbf{X})\, p(\mathbf{x})\, d\mathbf{x}$ is the kernel mean of the observed data $\mathbf{X}$, and for brevity we define $(\mathbf{K} + \sigma^2 \mathbf{I})^{-1} := \mathbf{K}^{-1}$ throughout.

The associated variance of $\langle \bar{f} \rangle$ is defined as:

$$\mathbb{V}(\langle \bar{f} \rangle) = \mu_{\mathbf{x}\mathbf{x}'} - \mu_{\mathbf{x}}(\mathbf{X})^T \mathbf{K}^{-1} \mu_{\mathbf{x}}(\mathbf{X}) \tag{2}$$

$$\mu_{\mathbf{x}\mathbf{x}'} = \int \int k(\mathbf{x}, \mathbf{x}') p(\mathbf{x}) p(\mathbf{x}') d\mathbf{x}\; d\mathbf{x}' \tag{3}$$

where $\mu_{\mathbf{x}\mathbf{x}'}$ represents the kernel mean over both $\mathbf{x}$ and $\mathbf{x}'$. We will refer to the kernel mean $\mu_{\mathbf{x}}(\mathbf{X})$ observed at integrand observations $\mathbf{X}$ as the *observed kernel mean*, and the dual kernel mean $\mu_{\mathbf{x}\mathbf{x}'}$ defined in Equation 3 as the *full kernel mean.*

We note that FFBQ is identical to traditional BQ outside of the methodology for calculating the values of or the observed kernel mean $\mu_{\mathbf{x}}(\mathbf{X})$ and full kernel mean $\mu_{\mathbf{x}\mathbf{x}'}$, for which FFBQ instead approximates through the FFT and inverse-FFT (IFFT) as $\hat{\mu}_{\mathbf{x}}(\mathbf{X})$ and $\hat{\mu}_{\mathbf{x}\mathbf{x}'}$. We will thus focus our efforts in this section on detailing the FFBQ methodologies for approximating these values, which can then be used interchangeably within the original BQ mean and variance formulations in Equations 1 and 2.

We first replicate Bochner's Theorem from the main text:

Theorem 1 (Bochner's theorem (Rudin, 2011))**.** *A shift-invariant kernel $k(\mathbf{x}, \mathbf{x}') = k(\mathbf{x} - \mathbf{x}')$ is positive-definite if and only if it is the Fourier transform of a non-negative measure.*

From Bochner's Theorem, we present the following lemma:

Lemma 1 (Fourier Transforms of Kernels and Measures)**.** *Given probability measure p, the Fourier transform $\mathcal{F}$ of p is:*

$$\mathcal{F}[p] = k_p, \tag{4}$$

where k_p is a stationary kernel function. By the properties of Fourier transforms, the double Fourier transform of p yields:

$$\mathcal{F}[\mathcal{F}[p]] \propto p \equiv \mathcal{F}[k_p] \propto p \tag{5}$$

The significance of this result arises when we combine it with properties of the convolution theorem, which we replicate here:

Theorem 2 (Convolution theorem (McGillem and Cooper, 1991))**.** *The convolution of functions g and h in the spatial domain is equivalent to the inverse Fourier transform of their point-wise multiplication in the frequency domain:*

$$\begin{aligned}(g * h)(\mathbf{x}) &= \int g(\mathbf{x} - \tau)h(\tau)d\tau \\ &= \mathcal{F}^{-1}\big[\mathcal{F}[g] \times \mathcal{F}[h]\big](\mathbf{x}),\end{aligned} \tag{6}$$

where $\mathcal{F}[g]$ and $\mathcal{F}^{-1}[g]$ denote the Fourier and inverse Fourier transforms of function g, respectively.

1.1 FFBQ Mean

Combining the convolution theorem with with Lemma 1 allows us to arrive at a reformulation for the observed kernel mean $\mu_{\mathbf{x}}(\mathbf{X})$:

Lemma 2 (Observed Kernel Mean $\mu_{\mathbf{x}}(\mathbf{X})$ as Convolution)**.**

$$\begin{aligned}\mu_{\mathbf{x}}(\mathbf{X}) &= \int\int k(\mathbf{x}, \mathbf{X})p(\mathbf{x})d\mathbf{x} \\ &= \int k(\mathbf{X} - \mathbf{x})p(\mathbf{x})d\mathbf{x} \\ &\equiv \mathcal{F}^{-1}\big[\mathcal{F}[k] \circ \mathcal{F}[p]\big](\mathbf{X}) \\ &\equiv \mathcal{F}^{-1}\big[p_k \circ k_p\big](\mathbf{X}),\end{aligned} \tag{7}$$

where $\circ$ denotes point-wise multiplication.

In (7), our choice of kernel and measure may yield closed forms for p_k and k_p, but in the case they do not, we can approximate these values using the fast Fourier transform (FFT):

Definition 1 (Fast Fourier Transform (Cooley and Tukey, 1965))**.** *Let $(x = (x_0, \ldots, x_{Z-1})$ be a sequence of Z complex numbers. The Fast Fourier Transform* $\mathrm{FFT}\big[x\big]$ *computes the sequence $X = (X_0, \ldots, X_{Z-1})$ in $\mathcal{O}(Z \log Z)$ time, where $X_k = \sum_{z=0}^{Z-1} x_z e^{-i2\pi kz/Z}$.*

Regardless of how we choose to represent p_k and k_p – whether via analytical relationships or FFT approximation – the inverse Fourier transform $\mathcal{F}^{-1}\big[p_k \circ k_p\big]$ of their point-wise multiplication will in general not be closed-form, thus requiring use of the inverse FFT (IFFT) in the FFBQ implementations.

Both the FFT and IFFT operate on a uniformly spaced d-dimensional hypergrid of function evaluations over the domain of convolution. We note this grid as $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_z\}_{z=1}^{Z}$. In BQ we assume that we have access to forward calculation of both k and p such that in combination with the FFT we can approximate p_k and k_p if their analytical forms aren't available.

If analytical forms are available, we can simply evaluate p_k and k_p on FFTFreq[$\boldsymbol{\Lambda}$], where FFTFreq denotes the FFT frequency transform, which maps uniform coordinates in the spatial domain to their equivalent coordinates in the frequency domain. This transform is a basic functionality available in any FFT implementation library.

Thus far, we have the means to calculate the observed kernel mean, using FFT convolution, at the coordinates of $\boldsymbol{\Lambda}$, ie. $\hat{\mu}_{\mathbf{x}}(\boldsymbol{\Lambda})$. However, BQ requires that we have access to the values of $\hat{\mu}_{\mathbf{x}}(\mathbf{X})$ at integrand observations $\mathbf{X}$. As $\hat{\mu}_{\mathbf{x}}(\boldsymbol{\Lambda})$ constitutes a uniformly spaced hyper-grid of $\hat{\mu}_{\mathbf{x}}$, it presents a valuable representation from which we can infer $\hat{\mu}_{\mathbf{x}}(\mathbf{X})$ using interpolation. We denote this interpolator applied to integrand values $\mathbf{X}$ as $\psi[\boldsymbol{\Lambda}](\mathbf{X})$.

Finally, we arrive at the FFBQ approximation to $\hat{\mu}_{\mathbf{x}}(\mathbf{X})$:

Definition 2 (FFBQ Observed Kernel Mean Approximation)**.** *Given data $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^{N} = \{\mathbf{X}, \mathbf{y}\}$, stationary kernel $k_{\boldsymbol{\theta}}$ with hyperparameters $\boldsymbol{\theta}$, and probability measure $p_{\boldsymbol{\gamma}}$ with parameters $\boldsymbol{\gamma}$, and a uniformly-spaced grid of Z coordinates $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_z\}_{z=1}^{Z}$ taken over the domain of integration, we can approximate the observed kernel mean $\mu_{\mathbf{x}}(\mathbf{X}) \approx \hat{\mu}_{\mathbf{x}}(\mathbf{X})$ as:*

$$\hat{\mu}_{\mathbf{x}}(\mathbf{X}) \coloneqq \psi\bigg[\mathrm{FFT}^{-1}\Big[\dot{\mathcal{F}}[k_{\boldsymbol{\theta}}] \circ \dot{\mathcal{F}}[p_{\boldsymbol{\gamma}}]\Big](\boldsymbol{\Lambda})\bigg](\mathbf{X}), \tag{8}$$

in which $\dot{\mathcal{F}}$ signifies either the analytical Fourier transform or fast Fourier transform, depending on availability from choice of kernel k and measure p, and $\psi[\cdot](\mathbf{X})$ is any interpolation operator evaluated at $\mathbf{X}$.

$\hat{\mu}_{\mathbf{x}}(\mathbf{X})$ is a drop-in replacement for $\mu_{\mathbf{x}}(\mathbf{X})$ in both the BQ mean and variance terms in Equations 1 and 2. With $\hat{\mu}_{\mathbf{x}}(\mathbf{X})$, we can fully produce the FFBQ integral mean $\langle \bar{f} \rangle$ approximation (1), and can calculate the second term $\hat{\mu}_{\mathbf{x}}(\mathbf{X})^T \mathbf{K}^{-1} \hat{\mu}_{\mathbf{x}}(\mathbf{X})$ in the BQ variance (2).

1.2 FFBQ Variance

In order to calculate the full FFBQ variance we require the approximation of the full kernel mean $\mu_{\mathbf{x}\mathbf{x}'}$, which we propose here is a simple extension of Lemma 1 to instead be a two-fold convolution of kernel k over measure p. As in Lemma 1, we make use of Bochner's and convolution Theorems (1, 2) to put forward the following:

Lemma 3 (Full Kernel Mean $\mu_{\mathbf{x}\mathbf{x}'}$ as Convolution)**.**

$$\begin{aligned}
\mu_{\mathbf{x}\mathbf{x}'} &= \int k(\mathbf{x}, \mathbf{x}')p(\mathbf{x})p(\mathbf{x}')d\mathbf{x}d\mathbf{x}' \\
&= \int \mu_{\mathbf{x}}(\mathbf{x}')p(\mathbf{x}')d\mathbf{x}' \\
&= \int \mu_{\mathbf{x}}(\mathbf{0} - \mathbf{x}')p(\mathbf{x}')d\mathbf{x}' \\
&\equiv \mathcal{F}^{-1}\big[\mathcal{F}[\mu_{\mathbf{x}}] \circ \mathcal{F}[p]\big](\mathbf{0}) \\
&\equiv \mathcal{F}^{-1}\big[p_k \circ k_p \circ k_p\big](\mathbf{0})
\end{aligned} \tag{9}$$

As in the FFBQ approximation of the observed kernel mean $\hat{\mu}_{\mathbf{x}}(\mathbf{X})$, we can make use of the FFT/IFFT for Fourier approximations and interpolation operator ψ to evaluate the resulting convolution at $\mathbf{0}$. Thus, our definition for the FFBQ approximation to $\mu_{\mathbf{x}\mathbf{x}'}$ is a simple extension of Definition 2:

Definition 3 (FFBQ Full Kernel Mean Approximation)**.** *Given data $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N = \{\mathbf{X}, \mathbf{y}\}$, stationary kernel $k_{\boldsymbol{\theta}}$ with hyperparameters $\boldsymbol{\theta}$, and probability measure $p_{\boldsymbol{\gamma}}$ with parameters $\boldsymbol{\gamma}$, and a uniformly-spaced grid of Z coordinates $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_z\}_{z=1}^Z$ taken over the domain of integration, we can approximate the full kernel mean $\mu_{\mathbf{x}\mathbf{x}'} \approx \hat{\mu}_{\mathbf{x}\mathbf{x}'}$ as:*

$$\hat{\mu}_{\mathbf{x}\mathbf{x}'} := \psi\Bigg[\mathrm{FFT}^{-1}\Big[\dot{\mathcal{F}}[k_{\boldsymbol{\theta}}] \circ \dot{\mathcal{F}}[p_{\boldsymbol{\gamma}}] \circ \dot{\mathcal{F}}[p_{\boldsymbol{\gamma}}]\Big](\boldsymbol{\Lambda})\Bigg](\mathbf{0}), \tag{10}$$

in which $\dot{\mathcal{F}}$ signifies either the analytical Fourier transform or fast Fourier transform, depending on availability from choice of kernel k and measure p, and $\psi[\cdot](\mathbf{0})$ is any interpolation operator evaluated at $\mathbf{0}$.

Using the results for approximations of observed and full kernel means from Definitions 2 and 3 yields the final forms for the FFBQ integral mean and variance approximations:

$$\langle \bar{f} \rangle \approx \mathbf{y}^T \mathbf{K}^{-1} \hat{\mu}_{\mathbf{x}}(\mathbf{X}), \tag{11}$$

$$\mathbb{V}(\langle \bar{f} \rangle) \approx \hat{\mu}_{\mathbf{x}\mathbf{x}'} - \hat{\mu}_{\mathbf{x}}(\mathbf{X})^T \mathbf{K}^{-1} \hat{\mu}_{\mathbf{x}}(\mathbf{X}). \tag{12}$$

2 SPARSE SPECTRUM BAYESIAN QUADRATURE (SSBQ) PROOF

We now turn our attention towards providing the derivations for the low-rank SSBQ mean and variance approximations. SSBQ builds upon the results presented by Lázaro-Gredilla et al., 2010 who propose a low-rank form for a GP marginal distribution:

$$\mu(f*) = \Phi(\mathbf{x}^*)^T \mathbf{A}^{-1} \Phi(\mathbf{X})\mathbf{y} \tag{13}$$

$$\mathbb{V}(f*) = \Phi(\mathbf{x}^*)^T \mathbf{A}^{-1} \Phi(\mathbf{x}^*) \tag{14}$$

$$\mathbf{A} = \Phi(\mathbf{X})\Phi(\mathbf{X})^T, \tag{15}$$

where Φ represents the RFF feature projection:

$$\Phi(\mathbf{x}) = \frac{\sqrt{2}}{\sqrt{2R}} \begin{bmatrix} \cos(\boldsymbol{\omega}_1^T \mathbf{x}) \\ \sin(\boldsymbol{\omega}_1^T \mathbf{x}) \\ \vdots \\ \cos(\boldsymbol{\omega}_R^T \mathbf{x}) \\ \sin(\boldsymbol{\omega}_R^T \mathbf{x}) \end{bmatrix}, \tag{16}$$

2.0.1 SSBQ Integral Mean

If we substitute the low-rank mean for the full-rank GP mean in the BQ mean (1) equation and reorganize terms, we arrive at the following low-rank BQ integral mean approximation:

$$\begin{aligned}\langle \bar{f} \rangle &\approx \int \mathbf{y}^T \Phi(\mathbf{x})^T \mathbf{A}^{-1} \Phi(\mathbf{X}) p(\mathbf{x}) d\mathbf{x} \\ &\approx \int \mathbf{y}^T \Phi(\mathbf{X})^T \mathbf{A}^{-1} \Phi(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} \\ &\approx \mathbf{y}^T \Phi(\mathbf{X})^T \mathbf{A}^{-1} \int \Phi(\mathbf{x}) p(\mathbf{x}) d\mathbf{x}.\end{aligned} \tag{17}$$

The question then turns to how we choose to approximate the integral $\int \Phi(\mathbf{x})p(\mathbf{x})d\mathbf{x}$, for which we propose two methodologies. The simplest methodology is to do so with Monte Carlo/Quasi-Monte Carlo:

$$\int \Phi(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} \approx \hat{\Phi} = \frac{1}{Z} \sum_{z=1}^{Z} \Phi(\mathbf{x}_z), \tag{18}$$

where $x_z \sim p(\mathbf{x})$. The resulting vector $\hat{\Phi}$ is length R, where R is the number of Fourier features used in the feature map projection in Equation 16.

Following the same logic presented in Section 1.1, we can alternatively solve this integral through convolution:

$$\begin{aligned}\int \Phi(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} &\equiv \int \Phi(\mathbf{0} - \mathbf{x}) p(\mathbf{x}) d\mathbf{x} \\ &\equiv \mathcal{F}^{-1}\big[\mathcal{F}[\Phi] \circ \mathcal{F}[p]\big](\mathbf{0}) \\ &\equiv \mathcal{F}^{-1}\big[\mathcal{F}[\Phi] \circ k_p\big](\mathbf{0}).\end{aligned} \tag{19}$$

We note that in the case of sin features, we actually repose the integral as correlation, ie. $\int \Phi(\mathbf{0}+\mathbf{x})p(\mathbf{x})d\mathbf{x}$, but the principles and implementations are nearly identical to the convolution setting.

Equations 18 and 19 are all we need, in addition to the definition for grid points $\mathbf{\Lambda}$ and interpolator ψ, to define the SSBQ mean approximation presented in the main text:

Definition 4 (Sparse-Spectrum Bayesian Quadrature Integral Mean)**.** *Given data $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N = \{\mathbf{X}, \mathbf{y}\}$, stationary RFF-defined kernel $k_{\boldsymbol{\omega}}$ formed by R samples of the kernel's spectral distribution $p_k(\boldsymbol{\omega})$, probability measure $p_{\boldsymbol{\gamma}}$ with parameters $\boldsymbol{\gamma}$, and a uniformly-spaced grid of Z coordinates $\mathbf{\Lambda} = \{\boldsymbol{\lambda}_z\}_{z=1}^Z$ taken over the domain of integration, the low-rank SSBQ approximation to the BQ posterior mean is:*

$$\begin{aligned}\langle \bar{f} \rangle &= \int \mathbf{y}^T \Phi(\mathbf{X})^T \mathbf{A}^{-1} \Phi(\boldsymbol{\lambda}) p(\boldsymbol{\lambda}) d\boldsymbol{\lambda} \\ &\approx \mathbf{y}^T \Phi(\mathbf{X})^T \mathbf{A}^{-1} \hat{\Phi}(\mathbf{\Lambda}),\end{aligned} \tag{20}$$

where

$$\begin{aligned}\hat{\Phi}(\mathbf{\Lambda}) &= \Big[\psi\big[\hat{\Phi}_1(\mathbf{\Lambda})\big](\mathbf{0}),\ \dots\ , \psi\big[\hat{\Phi}_R(\mathbf{\Lambda})\big](\mathbf{0})\Big]^T, \\ \hat{\Phi}_r(\mathbf{\Lambda}) &\coloneqq \mathrm{FFT}^{-1}\Big[\mathrm{FFT}\big[\Phi_r\big] \circ \dot{\mathcal{F}}[p_{\boldsymbol{\gamma}}]\Big](\mathbf{\Lambda})\end{aligned} \tag{21}$$

if using FFT convolution, or

$$\hat{\Phi}_r(\mathbf{\Lambda}) = \frac{1}{Z} \sum_{z=1}^{Z} \Phi_r(\boldsymbol{\lambda}_z) p_{\boldsymbol{\gamma}}(\boldsymbol{\lambda}_z) \tag{22}$$

if using Monte Carlo convolution.

We note that in the case of Monte Carlo approximation of the feature map mean (22), $\mathbf{\Lambda}$ need not actually be uniformly sampled, but in the interest reducing notation clutter we use $\mathbf{\Lambda}$ to also represent MC/QMC samples.

2.0.2 SSBQ Variance

The SSBQ variance is trivial to calculate given either the FFT or MC/QMC approximations to the vector $\hat{\Phi}$, and requires no further sampling or FFT procedures beyond what is performed in the integral mean approximation. We derive the SSBQ variance from the sparse spectrum GP variance presented in Equation 14.

The full-rank BQ integral variance estimate is simply the variance of the conditional full-rank GP integrated over $\mathbf{x}$ and $\mathbf{x}'$, which is Gaussian under linear integration (Hennig et al., 2022). We apply the same pattern here, and integrate the sparse spectrum GP variance (14) over both $\mathbf{x}$ and $\mathbf{x}'$ to obtain the SSBQ variance approximation:

$$\begin{aligned}\mathbb{V}(\langle\bar{f}\rangle) &\approx \int\int \Phi(\mathbf{x})^T\mathbf{A}^{-1}\Phi(\mathbf{x}')d\mathbf{x}d\mathbf{x}' \\ &\approx \hat{\Phi}(\mathbf{x})^T\mathbf{A}^{-1}\hat{\Phi}(\mathbf{x}')^T,\end{aligned} \tag{23}$$

As we have already calculated $\hat{\Phi}(\mathbf{x})$ and $\mathbf{A}^{-1}$ through the SSBQ integral mean calculation, no additional sampling or computation is required beyond simple low-rank matrix multiplication. We thus arrive at our definition for the SSBQ integral variance approximation:

Definition 5 (Sparse-Spectrum Bayesian Quadrature Integral Variance)**.** *Given data $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N = \{\mathbf{X}, \mathbf{y}\}$, stationary RFF-defined kernel $k_{\boldsymbol{\omega}}$ formed by R samples of the kernel's spectral distribution $p_k(\boldsymbol{\omega})$, probability measure $p_{\boldsymbol{\gamma}}$ with parameters $\boldsymbol{\gamma}$, and a uniformly-spaced grid of Z coordinates $\boldsymbol{\Lambda} = \{\boldsymbol{\lambda}_z\}_{z=1}^Z$ taken over the domain of integration, the low-rank SSBQ approximation to the BQ integral posterior variance is:*

$$\mathbb{V}(\langle\bar{f}\rangle) \approx \hat{\Phi}(\boldsymbol{\Lambda})^T\mathbf{A}^{-1}\hat{\Phi}(\boldsymbol{\Lambda})^T, \tag{24}$$

where $\hat{\Phi}(\boldsymbol{\Lambda})$ is as defined in Definition 4 through either FFT or MC approximation.

3 EXPERIMENTAL DETAILS

3.1 Gaussian Mixture Wall-Clock Experimental Details

We provide details here for the computational ablation study (Section 4.1 and Figure 2 in the main text). While our other experiments leverage modern GPU libraries and acceleration methods, we deliberately perform this experiment on CPU (M2 MacBook Air) in order to ensure that comparisons are fair between methods and not owing to specific library choices that are optimized for GPU.

Nonetheless, we still see major advantages in the FFBQ method over baselines on CPU, and there are a number of high-performing GPU implementations of the FFT that can also be leveraged to provide further computational benefits.

3.2 Genz Experimental Setup Details

We provide here extended detail on the experimental setup for the Genz integration benchmark (Genz, 1984) experiments. We highlight in the main text that our objective is compare the relative performances of methods under identical (potentially non-optimal) hyperparameters in order to clearly distinguish between their respective benefits. Table 1 provides the full parameter and experimental settings used across all methods.

We follow the same training and evaluation process for all methodologies and experiments. During the training stage, GP hyperparameters (kernel lengthscales and RFF kernel frequencies $\boldsymbol{\omega}$) are optimized through gradient descent on integrand GP negative log likelihood. These parameters are then used to initialize new GPs on an equally-sized test-set on which we evaluate the performance of each operator in BQ integration.

For each operator, we perform BQ using all possible valid integrand GP architectures. The results reported in the main text are for those integrand GPs that performed best, as evaluated by mean negative log-likelihood across random seeds, for each operator across a plurality of dimensions $d \in [2, 6]$. We then use the analytical solutions for each benchmark to calculate errors.

Parameter/Setting	Symbol	Value
Training Integrand Observation Count	N	1000
Testing Integrand Observation Count	N'	1000
Fourier Feature Count	R	100
Kernel Variance	σ_k^2	1
Gram Diagonal Noise	σ_n^2	0.1
# of Random Seeds per Experiment	-	10
Optimizer	-	Adam (Kingma and Ba, 2017)
Learning Rate	-	0.01
Training Epochs	-	200
Integration Bounds (Genz Continuous)	-	$x^d \in [0, 2]$
Integration Bounds (All Others)	-	$x^d \in [0, 1]$
CPU	-	AMD Ryzen 7 5800X
GPU	-	NVIDIA 3080 Ti GPU

Table 1: Parameters and Settings for Genz Experiments.

3.3 BQ Using Nyström SVGP

We outline here the approach for adapting the Nyström SVGP using the method of Titsias, 2009 to the BQ setting. The SVGP predictive mean is defined as:

$$\mu(f*) = \mathbf{K}_{*R}\mathbf{K}_{RR}\boldsymbol{\mu}_R, \tag{25}$$

where R is the number of inducing points and $\boldsymbol{\mu}_R$ is defined as in Equation 10 of Titsias, 2009. We follow the standard training procedures presented in the text for performing variational inference on inducing points $\mathbf{X}_R$ and kernel hyperparameters using training data from the integrand f.

If we substitute the SVGP predictive mean (25) into the BQ posterior mean (1), we find:

$$\begin{aligned}\langle\bar{f}\rangle &= \int k(\mathbf{x}, \mathbf{X}_R)^T \mathbf{K}_{RR}^{-1} \boldsymbol{\mu}_R \, p(\mathbf{x}) \, d\mathbf{x} \\ &= \boldsymbol{\mu}_R^T \mathbf{K}_{RR}^{-1} \int k(\mathbf{x}, \mathbf{X}_R) \, p(\mathbf{x}) \, d\mathbf{x},\end{aligned} \tag{26}$$

which is simply the kernel mean over the inducing points $\mathbf{X}_R$. In the experiment in the main text, we approximate the inducing-point kernel mean using QMC over a uniform integration measure, after which point the BQ posterior integral mean and variance can be calculated in the usual manner.

4 ADDITIONAL EXPERIMENTS & ABLATION STUDIES

4.1 FFBQ Diagonal Jitter Ablation

We present in Figure 1 a simple ablation study regarding the jitter term added to the diagonal of the Gram matrix $\mathbf{K}$ during FFBQ. We use the same experimental setup as Section 4.1 in the main text, where we compare the relative errors of different kernel mean operators to posterior integral estimate produced by analytical BQ with an RBF kernel and Gaussian measure. The results in Figure 1E demonstrate that suitable FFBQ hyperparameters can be determined by assessing solution convergence and reveal a wide range of effective settings. We observe that FFBQ needs comparatively larger diagonal noise σ^2 than other methods to stabilize, yet this doesn't negatively impact its performance.

4.2 FFBQ and SSBQ Variance Calibration

We perform a small study on the BQ integral variance estimates produced by each kernel mean operator. This setting shifts our focus towards evaluating the *calibration* of the full integral posterior distribution $\mathcal{N}(\langle\bar{f}\rangle, \mathbb{V}(\langle\bar{f}\rangle))$ produced by each operator. We explore this topic through the lens of results from the Genz continuous benchmark.

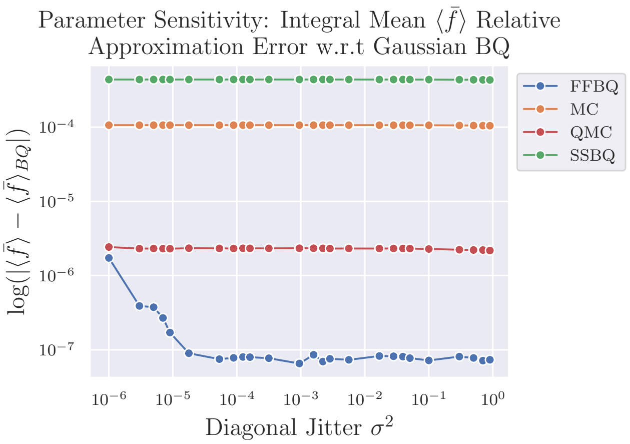

Figure 1: Relative Error of Kernel Mean Approximations to Gaussian BQ Across Jitter σ^2.

We use an identical experimental setup described in Section 3.2, with one modification. We observed that the BQ integral variance was highly sensitive, across all kernel mean operators, to the magnitude of the diagonal noise values σ_n^2 added to Gram matrix $\mathbf{K}$ to ensure numerical stability. If σ_n^2 is not jointly optimized while fitting the integrand GP, then the BQ integral variance $\mathbb{V}(\langle \bar{f} \rangle)$ takes on a subjective element conditional on the practitioners choice of noise term.

As we do not optimize σ_n^2 directly in our Genz experiments, we choose to adopt heuristics to select these values in this example. For analytical BQ, we simply use $\sigma_n^2 = \text{Tr}(\mathbf{K}) * 1e^{-4}$, while for approximated BQ (including MC methods and FFBQ/SSBQ), we adopt an eigenvalue clipping approach to ensure numerical stability, demonstrated below in Algorithm 1. We found that this approach was the most versatile for ensuring stable BQ approximations across settings and dimensionalities.

Algorithm 1 Kernel Matrix Eigenvalue Clipping

```
Require: Kernel matrix $\mathbf{K} \in \mathbb{R}^{N \times N}$
Ensure: Clipped and reprojected kernel matrix $\mathbf{K}'$
  Perform eigenvalue decomposition of $\mathbf{K}$, $\mathbf{K} = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^T$
  Identify the maximum eigenvalue $\lambda_{\text{max}} = \max_i \lambda_i$
  Initialize clipped eigenvalue diagonal matrix $\mathbf{\Lambda}'$
  for each eigenvalue $\lambda_i$ in $\mathbf{\Lambda}$ do
    Clip the eigenvalue $\lambda_i' = \max\left(\lambda_i, \frac{\lambda_{\text{max}}}{1000}\right)$:
    Set $\Lambda_{ii}' = \lambda_i'$
  end for
  Reproject the kernel matrix using the clipped eigenvalues $\mathbf{K}' = \mathbf{U}\mathbf{\Lambda}'\mathbf{U}^T$:
  return $\mathbf{K}'$
```

The use of heuristics in calculation of the BQ variance means that we are less interested in the absolute values generated by each operator, but rather the patterns that emerge as we compare the use of identical heuristics across varied settings. Figure 2 outlines our results across data dimensionality d, where calibration error represents the absolute solution Z-score, ie. $\left|\frac{\text{sol}-\langle \bar{f} \rangle}{\sqrt{\mathbb{V}(\langle \bar{f} \rangle)}}\right|$

We note that there are three GPs being represented in Figure 2, but four operators: a basic RBF GP for the BQ operator, a low-rank RFF GP for SSBQ, and a full-rank bounded RFF GP (Melkumyan and Ramos, 2009) shared by the QMC and FFBQ operators. Naturally, each GP will produce differing variance results based on the kernel in use, so it is unsurprising to see that variance estimates differ between the operators.

There are interesting trends to be be gleaned from these results. We observe that as dimensionality increases, traditional BQ and SSBQ variance estimates are relatively constant, which we would expect given that the data size N and Fourier feature size R is held constant. Conversely, we see that the QMC and FFBQ operators

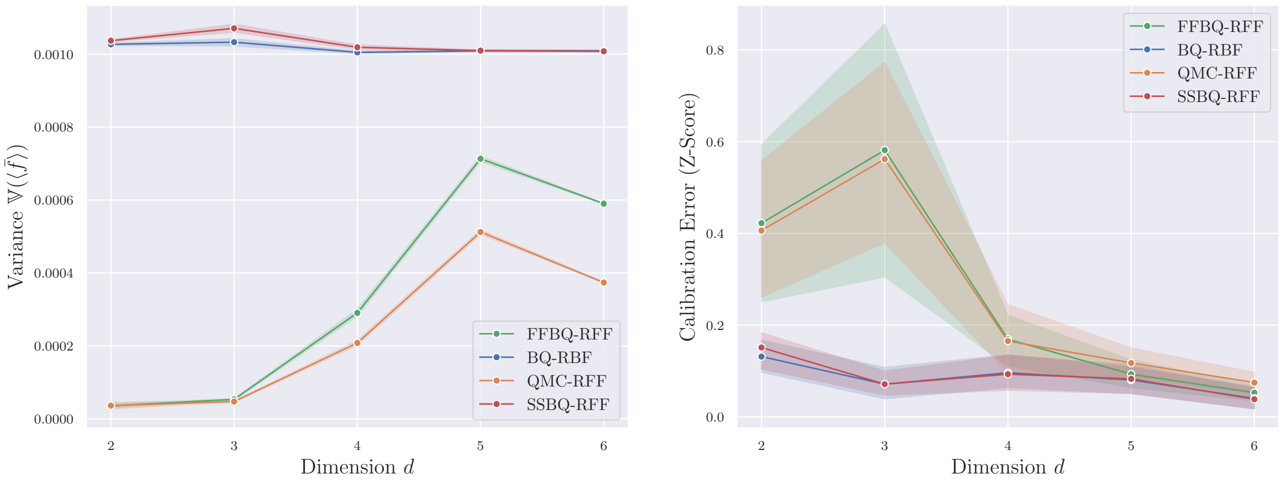

Figure 2: BQ Variance (Left) and Calibration Error (Right) Across Dimensionality d Over 10 Seeds.

produce larger variance estimates as dimensionality increases. The true solution to the integral in fact decreases as dimensionality increases, which implies that the increase in variance is due to increased approximation error of the terms $\mu_{\mathbf{x}}(\mathbf{X})$ and $\mu_{\mathbf{xx}'}$ used the the variance calculation. The calibration error of SSBQ and traditional BQ also remain stable over dimensions, while FFBQ and SSBQ calibration errors decrease as a result of the increase in variance. Regardless, all methods achieve admirable calibration error given that the solution Z-scores remains $-1 < Z < 1$ across all settings. The QMC and FFBQ operators, when using the bounded RFF GP, tend to slightly overestimate variance and thus have a higher calibration error. We can likely identify $\mu_{\mathbf{x}}(\mathbf{X})$ and $\mu_{\mathbf{xx}'}$ approximation error as the culprit in this regard.

We observed in the main text that SSBQ exhibited superior scaling in integral approximation error as dimensionality increases, despite only using 10% of the full-rank size Gram matrix. This performance is mirrored in this experiment, where the variance estimates closely match those produced by the analytical BQ solution regardless of dimensionality. This behavior is of great interest, and surprising, given that intuitively the curse of dimensionality should strongly affect the low-rank approximation accuracy if the number of features R is held constant. The experimental results we have observed suggest that further theoretical study on the approximation error and benefits of low-rank kernel methods within the BQ setting, whether through RFFs or an alternative, could be a valuable contribution to the domain.

References

Cooley, James W. and John W. Tukey (1965). “An Algorithm for the Machine Calculation of Complex Fourier Series”. In: *Mathematics of Computation* 19.90, pp. 297–301. JSTOR: `2003354`.

Genz, Alan (Sept. 1, 1984). “Testing Multidimensional Integration Routines”. In: *Proc. of International Conference on Tools, Methods and Languages for Scientific and Engineering Computation*, pp. 81–94.

Hennig, Philipp, Michael A. Osborne, and Hans Kersting (2022). *Probabilistic Numerics: Computation as Machine Learning*. Cambridge University Press. 398 pp.

Kingma, Diederik P. and Jimmy Ba (Jan. 29, 2017). *Adam: A Method for Stochastic Optimization*. arXiv: `1412.6980 [cs]`. preprint.

Lázaro-Gredilla, Miguel et al. (2010). “Sparse Spectrum Gaussian Process Regression”. In: *Journal of Machine Learning Research* 11.63, pp. 1865–1881.

McGillem, Clare D. and George R. Cooper (1991). *Continuous and Discrete Signal and System Analysis*. 3rd ed. 494 pp.

Melkumyan, Arman and Fabio Ramos (July 11, 2009). “A Sparse Covariance Function for Exact Gaussian Process Inference in Large Datasets”. In: *Proceedings of the 21st International Joint Conference on Artificial Intelligence*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 1936–1942.

Rudin, Walter (2011). *Fourier Analysis on Groups.* Hoboken: John Wiley & Sons.

Titsias, Michalis (Apr. 2009). “Variational Learning of Inducing Variables in Sparse Gaussian Processes”. In: *Proceedings of the Twelth International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 567–574.