Positively Weighted Kernel Quadrature via Subsampling

Satoshi Hayakawa, Harald Oberhauser, Terry Lyons
Mathematical Institute, University of Oxford
{hayakawa,oberhauser,tlyons}@maths.ox.ac.uk

Abstract

We study kernel quadrature rules with convex weights. Our approach combines the spectral properties of the kernel with recombination results about point measures. This results in effective algorithms that construct convex quadrature rules using only access to i.i.d. samples from the underlying measure and evaluation of the kernel and that result in a small worst-case error. In addition to our theoretical results and the benefits resulting from convex weights, our experiments indicate that this construction can compete with the optimal bounds in well-known examples. [1]

1 Introduction

The goal of numerical quadrature is to provide, for a given probability measure μ on a space $\mathcal{X}$, a set of points $x_1, \dots, x_n \in \mathcal{X}$ and weights $w_1, \dots, w_n \in \mathbb{R}$ such that

$$\sum_{i=1}^{n} w_i f(x_i) \approx \int_{\mathcal{X}} f(x) \, \mathrm{d}\mu(x) \tag{1}$$

holds for a large class of functions $f : \mathcal{X} \to \mathbb{R}$. Kernel quadrature focuses on the case when the function class forms a reproducing kernel Hilbert space (RKHS). What makes kernel quadrature attractive, is that the kernel choice provides a simple and flexible way to encode the regularity properties of a function class. Exploiting such regularity properties is essential when the integration domain $\mathcal{X}$ is high-dimensional or the function class is large. Additionally, the domain $\mathcal{X}$ does not have to be Euclidean, but can be any topological space that carries a positive semi-definite kernel.

More formally, given a set $Q \coloneqq \{(w_i, x_i) : i = 1, \dots, n\} \subset \mathbb{R} \times \mathcal{X}$ denote with $\mu^Q \coloneqq \sum_{i=1}^{n} w_i \delta_{x_i}$ the resulting measure on $\mathcal{X}$. We refer to Q [resp. μ^Q] as a quadrature [resp. quadrature measure], to the points $x_1, \dots, x_n$ as the support of Q [resp. μ^Q]. The aim of kernel quadrature is to construct quadrature measures μ^Q that have a small worst-case error

$$\mathrm{wce}(Q; \mathcal{H}_k, \mu) \coloneqq \sup_{\|f\|_{\mathcal{H}_k} \le 1} \left| \int_{\mathcal{X}} f(x) \, \mathrm{d}\mu^Q(x) - \int_{\mathcal{X}} f(x) \, \mathrm{d}\mu(x) \right|, \tag{2}$$

where $\mathcal{H}_k$ denotes the RKHS associated with a positive semi-definite kernel k. If the weights are positive and sum up to one, $w_i > 0$, $\sum_{i=1}^{n} w_i = 1$, then we refer to Q as a convex quadrature rule.

Contribution. The primary contribution of this article is to leverage recombination (a consequence of Carathéodory's Theorem) with spectral analysis of kernels to construct convex kernel quadrature rules and derive convergence rates. We also provide efficient algorithms that compute these quadrature rules; they only need access to i.i.d. samples from μ and the evaluation of the kernel k. See Table 1 for a comparision with other kernel quadrature constructions.

[1] Code: https://github.com/satoshi-hayakawa/kernel-quadrature

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

The table is written by using σ_n and r_n, which represent a sort of decay of the kernel with respect to μ. Typical regimes are $\sigma_n \sim n^{-\beta}$ (e.g. Sobolev) or $\sigma_n \sim \exp(-\gamma n)$ (e.g. Gaussian) depending on the 'smoothness' of the kernel [e.g. 16, 3] (see also Section B.3), and in such regimes (with $\beta \geq 2$ or $\gamma > 0$), σ_n or $r_n (\lesssim n\sigma_n)$ provide faster rates than $\mathrm{wce}^2 \sim 1/n$ of the usual Monte Carlo rate. For more examples including multivariate Sobolev spaces, see Bach [3, Section 2.3].

Limitation. Our proposed methods are based on either Mercer or Nyström approximation. Though our Mercer-based methods result in strong theoretical bounds, they require the knowledge of Mercer decomposition like [3, 7, 8], which is not available for general (k, μ). Our Nyström-based methods apply to much more general situations and outperform existing methods in experiments, but the $n/\sqrt{\ell}$ term makes their theoretical bound far from competitive. Further study is needed to bridge the gap between theory and empirical results.

Method	Bound of squared wce	Computational complexity	C	M	E
Herding [10, 4]	$1/n$	$n \cdot (n$: global optimization)	✓	✓	
SBQ [25]	Not found	$n \cdot (n^2$: global optimization)		✓	
Leveraged [3]	$\sigma_m, m = \mathcal{O}(n \log n)$	Unavailable			
DPP [7, 6]	r_{n+1}	$n^3 \cdot$ (rejection sampling)			
CVS [8]	σ_{n+1}	Unavailable		✓	
KT++ [14, 15, 56]	$(1/n^2 + 1/N)\,\mathrm{polylog}(N)$	$N \log^3 N$	✓	✓	✓
Ours:					
Mercer†	r_n	$nN_\varphi + C(n, N_\varphi)$	✓		
M. + empirical‡	$r_n + \frac{1}{N}$	$nN + n^3 \log(N/n)$	✓		✓
Nyström†	$n\sigma_n + r_{n+1} + \frac{n}{\sqrt{\ell}}$	$n\ell N_\varphi + n\ell^2 + C(n, N_\varphi)$	✓	✓	
N. + empirical‡	$n\sigma_n + r_{n+1} + \frac{n}{\sqrt{\ell}} + \frac{1}{N}$	$n\ell N + n\ell^2 + n^3 \log(N/n)$	✓	✓	✓

Table 1: Comparison on n-point kernel quadrature rules. We are omitting the $\mathcal{O}$ notation throughout the table. Note that the assumption under which the theoretical guarantee holds varies from method to method, and this table displays just a representative bound derived in the cited references. Here are remarks on the notation. (1) σ_m is the m-th eigenvalue of the integral operator $\mathcal{K}$, and $r_m = \sum_{i=m}^{\infty} \sigma_i$. (2) The symbols in the first line respectively mean C: convex, M: *not* using the knowledge of Mercer decomposition, and E: *not* using the knowledge of expectations such as $\int_{\mathcal{X}} k(x, y)\,\mathrm{d}\mu(y)$. (3) The $(m$: global optimization$)$ is indicating the cost of globally optimizing a function whose evaluation costs $\Theta(m)$. (†) Mercer/Nyström are the algorithms based on random convex hulls, see Section 2.4 and Appendix D. (‡) M./N. + empirical are the algorithms discussed in the main text.

Why Convex Weights? There are several reasons why convex weights are preferable: (i) **Positive Integral Operator:** Kernel quadrature provides an approximation of the integration operator $f \mapsto I(f) = \int f(x)\,\mathrm{d}\mu(x)$. Hence, a natural requirement is to preserve basic properties of this operator and positive weights preserve the positivity of this operator. (ii) **Uniform estimates and Robustness:** In applications, the RKHS $\mathcal{H}_k$ may be mis-specified if a quadrature rule with negative weights is applied to a function $f \notin \mathcal{H}_k$, the approximation error (1) can get arbitrary bad; in contrast, a simple estimate shows that convex weights give uniform bounds, see Appendix B.4. (iii) **Iteration:** Consider the m-fold product of quadrature formulas for approximating $\mu^{\otimes m}$ on $\mathcal{X}^m$. This is a common construction for a multidimensional quadrature formulas (e.g., for polynomials) from one-dimensional formulas [61] or numerics for stochastic differential equations [42]. In doing so, working with a probability measure is strongly preferred, since otherwise the total variation of their m-fold product gets exponentially large as m increases ($\|\mu^{\otimes m}\|_{\mathrm{TV}} = \|\mu\|_{\mathrm{TV}}^m$).

Related Literature. Roughly speaking, there have been two approaches to kernel-based quadrature formulas: kernel herding and random sampling. In kernel herding or its variants, the points $(x_i)_{i=1}^n$ are found iteratively, typically based on the Frank–Wolfe gradient descent algorithm [10, 4, 25].

In the random sampling approach, $(x_i)_{i=1}^n$ are sampled and subsequently the weights are optimized. Generically, this results only in a signed measure μ^Q but not a probability measure. Bach [3] and

Belhadji et al. [7] use the eigenvalues and the eigenfunctions of the integral operator $\mathcal{K} : f \mapsto \int_{\mathcal{X}} k(\cdot, y) f(y) \, \mathrm{d}\mu(y)$ to obtain a Mercer-type decomposition of k [59]. Bach [3] then uses the eigenvalues and eigenfunctions of $\mathcal{K}$ to define an optimized measure from which the points (x_i) are i.i.d. sampled. This achieves a near optimal rate, but the exact sampling from this measure is usually unavailable, although for special cases, it can be done efficiently. In contrast, Belhadji et al. [7] proposes non-independent sampling based on the determinantal point process [DPP; 24]. These two papers also treat the more general quadrature problem that includes a weight function $g \in L^2(\mu)$, i.e., approximating $\int_{\mathcal{X}} f(x) g(x) \, \mathrm{d}\mu(x)$ for $f \in \mathcal{H}_k \subset L^2(\mu)$, which we do not discuss in this paper. Another recently introduced method is kernel thinning [14, 15], which aims at efficient compression of empirical measures that can be obtained by sampling like our '+ empirical' methods. Its acceleration [56] makes it a competitive candidate in terms of compressing $N \sim n^2$ points ('KT++' in Table 1).

Finally, we emphasize that the kernel quadrature literature is vast, and the distinction between herding and sampling is only a rough dichotomy, see e.g. [12, 40, 9, 27, 48, 29, 28, 57]. Beyond kernel quadrature, our algorithms can also contribute to the density estimation approach in [64] which relies on recombination based on Fourier features although we do not pursue this further in this article.

Outline. Section 2 contains our main theoretical and methodological contribution. Section 3 provides numerical experiments on common benchmarks. The Appendix contains several extensions of our main result, proofs, and further experiments and benchmarks.

2 Main Result

Assume we are given a set[2] of $n-1$ functions $\varphi_1, \ldots, \varphi_{n-1} : \mathcal{X} \to \mathbb{R}$ such that their linear combinations well approximate functions in $\mathcal{H}_k$. Then our kernel quadrature problem reduces to the construction of an n-point discrete probability measure $\mu^{Q_n} = \sum_{i=1}^n w_i \delta_{x_i}$ such that

$$\int_{\mathcal{X}} \varphi_i(x) \, \mathrm{d}\mu^{Q_n}(x) = \int_{\mathcal{X}} \varphi_i(x) \, \mathrm{d}\mu(x) \quad \text{for every } i = 1, \ldots, n. \tag{3}$$

A simple way to *approximately* construct this μ^{Q_n} is to first, sample $N \gg n$ points, $(y_i)_{i=1}^N$, from μ such that their empirical measure, $\widetilde{\mu}_N = \frac{1}{N} \sum_{i=1}^N \delta_{y_i}$, is a good approximation to μ in the sense that $\int \varphi_i \, \mathrm{d}\widetilde{\mu}_N \approx \int \varphi_i \, \mathrm{d}\mu$ for $i = 1, \ldots, n-1$, and secondly, apply a so-called recombination algorithm (Remark 1) that takes as input $(y_i)_{i=1}^N$ and n functions $\varphi_1, \ldots, \varphi_{n-1}$ and outputs a measure $\mu^{Q_n} = \sum w_i \delta_{x_i}$ by selecting a subset $(x_i)_{i=1}^n$ of the points $(y_i)_{i=1}^N$ and giving them weights $(w_i)_{i=1}^n$ such that μ^{Q_n} is a probability measure that satisfies the equation (3) with μ replaced by $\widetilde{\mu}_N$.

The challenging parts of this approach are (i) to construct functions $\varphi_1, \ldots, \varphi_{n-1}$ that approximately span the RKHS $\mathcal{H}_k$ for a small n; (ii) to arrive at good quantitative bounds despite the (probabilistic) sampling error resulting from the use of the empirical measure $\widetilde{\mu}_N$, and the function approximation error via $\varphi_1, \ldots, \varphi_{n-1}$. To address (i) we look for functions such that

$$k(x, y) \approx k_0(x, y) \coloneqq \sum_{i=1}^{n-1} c_i \varphi_i(x) \varphi_i(y) \tag{4}$$

with some $c_i \geq 0$. Two classic ways to do this are the Mercer and Nyström approximations. The remaining, item (ii) is our main contribution. Theorem 1 shows that the worst-case error, (5), is controlled by the sum of two terms: the first term stems from the kernel approximation (4), the second term stems from the sample error.

Theorem 1. *Let μ be a Borel probability measure on $\mathcal{X}$ and k a positive semi-definite kernel on $\mathcal{X}$ such that $\int_{\mathcal{X}} k(x, x) \, \mathrm{d}\mu(x) < \infty$. Further, let n be a positive integer and assume k_0 is a positive semi-definite kernel on $\mathcal{X}$ such that*

1. $k - k_0$ is a positive semi-definite kernel on $\mathcal{X}$, and 2. $\dim \mathcal{H}_{k_0} < n$.

There exists a function KQuad *such that if D_N is a set of N i.i.d. samples from μ, then $Q_n = \mathrm{KQuad}(D_N)$ is a random n-point convex quadrature that satisfies*

$$\mathbb{E}_{D_N}\left[\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2\right] \leq 8 \int_{\mathcal{X}} (k(x, x) - k_0(x, x)) \, \mathrm{d}\mu(x) + \frac{2 c_{k,\mu}}{N}. \tag{5}$$

[2] The number $n-1$ stems from Carathéodory's theorem, Remark 1, and leads to an n point quadrature rule.

where $c_{k,\mu} := \int_{\mathcal{X}} k(x,x)\,\mathrm{d}\mu(x) - \iint_{\mathcal{X}\times\mathcal{X}} k(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y)$.

Moreover, the support of Q_n *is a subset of* D_N *and given functions* $\varphi_1,\dots,\varphi_{n-1} \in L^1(\mu)$ *with* $\mathcal{H}_{k_0} \subset \operatorname{span}\{\varphi_1,\dots,\varphi_{n-1}\}$, $Q_n = \mathrm{KQuad}(D_N)$ *can be computed with Algorithm 1 in* $\mathcal{O}\big(nN + n^3\log(N/n)\big)$ *computational steps.*

The function KQuad is deterministic but since D_N is random, the resulting quadrature rule Q_n is random, hence also the resulting worst case error $\mathrm{wce}(Q;\mathcal{H}_k,\mu)$ and the expectation in (5) denotes the expectation over the N samples in D_N. The theoretical part of Theorem 1 follows from more general results that we present and prove in the Appendix: Theorem 7 proves the inequality, essentially by comparing $\mathcal{H}_k$ with $\mathcal{H}_{k_0}$; Theorem 8 proves the existence. The algorithmic part of Theorem1 is discussed in Section 2.1 below. Theorem 1 covers our two main examples for the construction of k_0, resp. the choice of $\varphi_1,\dots,\varphi_{n-1}$, and for which the error estimate gets quite explicit: the Mercer approximation, see Section 2.2, and the Nyström approximation, see Section 2.3. The former requires some knowledge about the spectrum of the kernel which is, however, known for many popular kernels; the latter works in full generality but yields worse theoretical guarantees for the convergence rate. Finally, we emphasize that N and n in Theorem 1 can be chosen independently and we will see that from a computational point the choice $N \sim n^2$ is preferable in which case (5) is faster rate than Monte Carlo, see also Table 1.

2.1 Algorithm

Algorithm 1 Kernel Quadrature with Convex Weights via Recombination KQuad

Input: A positive semi-definite kernel k on $\mathcal{X}$, a probability measure μ on $\mathcal{X}$, integers $N \geq n \geq 1$, another kernel k_0, functions $\varphi_1,\dots,\varphi_{n-1}$ on $\mathcal{X}$ with $\mathcal{H}_{k_0} \subset \operatorname{span}\{\varphi_1,\dots,\varphi_{n-1}\}$ and a set D_N of N i.i.d. samples from μ.

Output: A set $Q_n := \{(w_i,x_i) \mid i=1,\dots,n\} \subset \mathbb{R}\times\mathcal{X}$ with $w_i \geq 0$, $\sum_{i=1}^n w_i = 1$

1: Apply a Recombination Algorithm (Remark 1) with $\boldsymbol{\psi} = (\varphi_1,\dots,\varphi_{n-1},k_{1,\mathrm{diag}})^\top$, to the empirical measure $\frac{1}{N}\sum_{y\in D_N}\delta_y$ to obtain points $\{\widetilde{x}_1,\dots,\widetilde{x}_{n+1}\} \subset D_N$ and weights $\boldsymbol{v} = (v_1,\dots,v_{n+1})^\top \geq \mathbf{0}$ that satisfy $\mathbf{1}^\top\boldsymbol{v} = 1$ and $\boldsymbol{\psi}(\widetilde{\boldsymbol{x}})\boldsymbol{v} = \frac{1}{N}\sum_{i=1}^N \boldsymbol{\psi}(\widetilde{x}_i)$, where $\boldsymbol{\psi}(\widetilde{\boldsymbol{x}}) = [\boldsymbol{\psi}(\widetilde{x}_1),\dots,\boldsymbol{\psi}(\widetilde{x}_{n+1})] \in \mathbb{R}^{n\times(n+1)}$.

2: Apply SVD with the matrix $A = [\varphi_{i-1}(y_j)]_{ij} \in \mathbb{R}^{n\times(n+1)}$ with $\varphi_0 = 1$ to find a nonzero vector $\boldsymbol{u} \in \mathbb{R}^{n+1}$ such that $A\boldsymbol{u} = \mathbf{0}$ and $k_{1,\mathrm{diag}}(\widetilde{\boldsymbol{x}})^\top\boldsymbol{u} \geq 0$

3: Compute the smallest $\alpha \geq 0$ such that $\boldsymbol{v} - \alpha\boldsymbol{u} \geq \mathbf{0}$ and $v_j - \alpha u_j = 0$ for some j

4: Return $(w_i)_{i=1}^n \leftarrow (v_k - \alpha u_k)_{k\in I}$ and $(x_i)_{i=1}^n \leftarrow (\widetilde{x}_k)_{k\in I}$, where $I = \{1,\dots,n+1\}\setminus\{j\}$

Suppose we are given k_0 and $\varphi_1,\dots,\varphi_{n-1} \in L^1(\mu)$ with $\mathcal{H}_{k_0} \subset \operatorname{span}\{\varphi_1,\dots,\varphi_{n-1}\}$, and also N independent samples from μ denoted by $D_N = (y_1,\dots,y_N)$. Theorem 7 in the Appendix shows that if we construct a convex quadrature $Q_n = (w_i,x_i)_{i=1}^n$ satisfying

$$\sum_{i=1}^n w_i\boldsymbol{\varphi}(x_i) = \frac{1}{N}\sum_{i=1}^N \boldsymbol{\varphi}(y_i), \qquad \sum_{i=1}^n w_i k_{1,\mathrm{diag}}(x_i) \leq \frac{1}{N}\sum_{i=1}^N k_{1,\mathrm{diag}}(y_i), \tag{6}$$

where $\boldsymbol{\varphi} = (\varphi_1,\dots,\varphi_{n-1})^\top$ and $k_{1,\mathrm{diag}}(x) = k(x,x) - k_0(x,x)$, it satisfies the bound (5). For this problem, we can use the so-called *recombination* algorithms:

Remark 1 (Recombination)**.** *Given* $d-1$ *functions (called test functions) and a probability measure supported on* $N > d$ *points, there exists a probability measure supported on a subset of* d *points that gives the same mean to these* $d-1$ *functions. This follows from Carathéodory's theorem and is known as recombination. Efficient deterministic [38, 43, 61] as well as randomized [11] algorithms exist to compute the new probability measure supported on* d *points; e.g. deterministic algorithms perform the recombination, step 1, in* $\mathcal{O}\big(c_\varphi N + d^3\log(N/d)\big)$ *time, where* c_φ *is the cost of computing all the test functions at one sample. If each function evaluation is in constant time,* $c_\varphi = \mathcal{O}(d)$.

Let us briefly provide the intuition behind the deterministic recombination algorithms. We can solve the problem of "reducing (weighted) $2d$ points to d points in $\mathbb{R}^d$ while keeping the barycenter" by using linear programming or a variant of it. If we apply this to $2d$ points each given by a barycenter

of approximately $\frac{N}{2d}$ points, we can reduce the original problem of size N to a problem of size $d \cdot \frac{N}{2d} = \frac{N}{2}$. By repeating this procedure $\log_2(\frac{N}{d})$ times we obtain the desired measure.

Although the recombination introduced here only treats the equality constraints in (6) we can satisfy the remaining constraints just with n points by modifying it. This is done in Algorithm 1 which works as follows: First, via recombination, find an $(n+1)$-point convex quadrature $R_{n+1} = (v_i, y_i)_{i=1}^{n+1}$ that exactly integrates functions $\varphi_1, \ldots, \varphi_{n-1}, k_{1,\mathrm{diag}}$ with regard to the empirical measure $\frac{1}{N}\sum_{i=1}^{N} \delta_{y_i}$. Second, to reduce one point, find a direction ($-\boldsymbol{u}$ in the algorithm) in the space of weights on $(\widetilde{x}_i)_{i=1}^{n+1}$ that does not change the integrals of $\varphi_1, \ldots, \varphi_{n-1}$ and the constant function 1, and does not increase the integral of $k_{1,\mathrm{diag}}$. Finally, move the weight from $\boldsymbol{v}$ to the above direction until an entry becomes zero, at $\boldsymbol{v} - \alpha\boldsymbol{u}$. Such an $\alpha \geq 0$ exists, as $\boldsymbol{u}$ must have a positive entry since it is a nonzero vector whose entries sum up to one. Now we have a convex weight vector with at most n nonzero entries, so it outputs the desired quadrature satisfying (6).

2.2 Mercer Approximation

In this section and Section 2.3, we assume that k has a pointwise convergent Mercer decomposition $k(x,y) = \sum_{m=1}^{\infty} \sigma_m e_m(x) e_m(y)$ with $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$ and $(e_m)_{m=1}^{\infty} \subset L^2(\mu)$ being orthonormal [59]. If we let $\mathcal{K}$ be the integral operator $L^2(\mu) \to L^2(\mu)$ given by $f \mapsto \int_{\mathcal{X}} k(\cdot, y) f(y)\,\mathrm{d}\mu(y)$, then $(\sigma_m, e_m)_{m=1}^{\infty}$ are the eigenpairs of this operator.

The first choice of the approximate kernel k_0 is just the trucation of Mercer decomposition.

Corollary 2. *Theorem 1 applied with $k_0(x,y) = \sum_{m=1}^{n-1} \sigma_m e_m(x) e_m(y)$ yields a random convex quadrature rule Q_n such that*

$$\mathbb{E}_{D_N}\left[\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2\right] \leq 8 \sum_{m=n}^{\infty} \sigma_m + \frac{2c_{k,\mu}}{N}. \tag{7}$$

Proof. It suffices to prove the result under the assumption $\int_{\mathcal{X}} k(x,x)\,\mathrm{d}\mu(x) = \sum_{m=1}^{\infty} \sigma_m < \infty$, as otherwise the right-hand side of (7) is infinity.

For $k_1 \coloneqq k - k_0$, we have that $k_1(x,y) = \sum_{m=n}^{\infty} \sigma_m e_m(x) e_m(y)$ and it is the inner product of $\Phi(x) \coloneqq (\sqrt{\sigma_m} e_m(x))_{m=n}^{\infty}$ and $\Phi(y)$ in $\ell^2(\{n, n+1, \ldots\})$ and so positive semi-definite. Thus k and k_0 satisfies the assumption of Theorem 1, and $\int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x) = \sum_{m=n}^{\infty} \sigma_m$ applied to (5) yields the desired inequality. □

2.3 Nyström Approximation

Although the Nyström method [68, 13, 34] is primarily used for approximating a large Gram matrix by a low rank matrix, it can also be used for directly approximating the kernel function itself and this is how we use it. Given a set of ℓ points $Z = (z_i)_{i=1}^{\ell} \subset \mathcal{X}$, the vanilla Nyström approximation of $k(x,y)$ is given by

$$k(x,y) = \langle k(\cdot,x), k(\cdot,y)\rangle_{\mathcal{H}_k} \approx \langle P_Z k(\cdot,x), P_Z k(\cdot,y)\rangle_{\mathcal{H}_k} =: k^Z(x,y), \tag{8}$$

where $P_Z : \mathcal{H}_k \to \mathcal{H}_k$ is a projection operator onto $\mathrm{span}\{k(\cdot, z_i)\}_{i=1}^{\ell}$. In matrix notation, we have

$$k^Z(x,y) = k(x,Z) W^+ k(Z,y) \coloneqq [k(x,z_1), \ldots, k(x,z_\ell)] W^+ \begin{bmatrix} k(z_1,y) \\ \vdots \\ k(z_\ell,y) \end{bmatrix}, \tag{9}$$

where $W = (k(z_i,z_j))_{i,j=1}^{\ell}$ is the Gram matrix for Z and W^+ denotes its Moore–Penrose inverse. We discuss the equivalence between (8) and (9) in Appendix B.5. As k^Z is an ℓ-dimensional kernel, there exists an $(\ell+1)$-point quadrature formula that exactly integrates functions in $\mathcal{H}_{k^Z}$. For a quadrature formula, exactly integrating all the functions in $\mathcal{H}_{k^Z}$ is indeed equivalent to exactly integrating $k(z_i, \cdot)$ for all $1 \leq i \leq \ell$, as long as the Gram matrix $k(Z,Z)$ is nonsingular. Proposition 1 in the Appendix provides bound for the associated worst case error. From this viewpoint, the Nyström approximation offers a natural set of test functions.

The Nyström method has a further generalization with a low-rank approximation of $k(Z,Z)$. Concretely, by letting W_s be the best rank-s approximation of $W = k(Z,Z)$ (given by eigendecomposition), we define the following s-dimensional kernel:

$$k_s^Z(x,y) \coloneqq k(x,Z)W_s^+ k(Z,y). \tag{10}$$

Let $W = U\Lambda U^\top$ be the eigendecomposition of W, where $U = [u_1,\dots,u_\ell] \in \mathbb{R}^{\ell\times\ell}$ is a real orthogonal matrix and $\Lambda = \mathrm{diag}(\lambda_1,\dots,\lambda_\ell)$ with $\lambda_1 \geq \cdots \geq \lambda_\ell \geq 0$. Then, if $\lambda_s > 0$ we have

$$k_s^Z(x,y) = \sum_{i=1}^{s} \frac{1}{\lambda_i}(u_i^\top k(Z,x))(u_i^\top k(Z,y)). \tag{11}$$

So we can use functions $u_i^\top k(Z,\cdot)$ $(i = 1,\dots,s)$ as test functions, which is chosen from a larger dimensional space $\mathrm{span}\{k(z_i,\cdot)\}_{i=1}^\ell$. Although closer to the original usage of the Nystöm method is to obtain $u_i^\top k(Z,\cdot)$ as an approximation of i-th eigenfunction of the integral operator $\mathcal{K}$ with Z appropriately chosen with respect to μ, we have adopted an explanation suitable for the machine learning literature [13, 34].

The following is a continuous analogue of Kumar et al. [34, Theorem 2] showing the effectiveness of the Nyström method. See also Jin et al. [26] for an analysis specific to the case $s = \ell$.

Theorem 3. *Let $s \leq \ell$ be positive integers and $\delta > 0$. Let Z be an ℓ-point independent sample from μ. If we define the integral operator $\mathcal{K}_s^Z : L^2(\mu) \to L^2(\mu)$ by $f \mapsto \int_{\mathcal{X}} k_s^Z(\cdot,y)f(y)\,\mathrm{d}\mu(y)$, then we have, with probability at least $1-\delta$, in terms of the operator norm,*

$$\|\mathcal{K}_s^Z - \mathcal{K}\| \leq \sigma_{s+1} + \frac{2\sup_{x\in\mathcal{X}} k(x,x)}{\sqrt{\ell}}\left(1 + \sqrt{2\log\frac{1}{\delta}}\right). \tag{12}$$

The proof is given in Appendix C.5. By using this estimate, we obtain the following guarantee for the random convex quadrature given by Algorithm 1 and the Nyström approximation.

Corollary 4. *Let D_N be N-point independent sample from μ and let Z be an ℓ-point independent sample from μ. Theorem 1 applied with the Nyström approximation $k_0 = k_{n-1}^Z$ yields an random n-point convex quadrature rule Q_n such that, with probability at least $1-\delta$ and $k_{\max} \coloneqq \sup_{x\in\mathcal{X}} k(x,x)$,*

$$\mathbb{E}_{D_N}\left[\mathrm{wce}(Q_n;\mathcal{H}_k,\mu)^2 \,\middle|\, Z\right] \leq 8\left(n\sigma_n + \sum_{m>n}\sigma_m\right) + \frac{16(n-1)k_{\max}}{\sqrt{\ell}}\left(1+\sqrt{2\log\frac{1}{\delta}}\right) + \frac{2c_{k,\mu}}{N}.$$

Proof. From (11), $k^Z(x,y) - k_{n-1}^Z(x,y) = \sum_{i=n}^\ell \lambda_i^{-1}(u_i^\top k(Z,x))(u_i^\top k(Z,y))$ (ignore the terms with $\lambda_i = 0$ if necessary), and it is thus positive semi-definite. If we define $P_Z^\perp : \mathcal{H}_k \to \mathcal{H}_k$ as the projection operator onto the orthogonal complement of $\mathrm{span}\{k(\cdot,z_i)\}_{i=1}^\ell$, then, from (8), we also have $k(x,y) - k^Z(x,y) = \left\langle P_Z^\perp k(\cdot,x), P_Z^\perp k(\cdot,y)\right\rangle_{\mathcal{H}_k}$, so $k - k^Z$ is also positive semi-definite. In particular, $k - k_{n-1}^Z = (k - k^Z) + (k^Z - k_{n-1}^Z)$ is positive semi-definite. Also, it suffices to prove the result when $\sum_{m=1}^\infty \sigma_m < \infty$, so we can now apply Theorem 1.

For $k_1 \coloneqq k - k_{n-1}^Z$, we prove the inequality $\int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x) = \sum_{m=1}^\infty \left\langle e_m, (\mathcal{K} - \mathcal{K}_s^Z)e_m\right\rangle_{L^2} \leq (n-1)\|\mathcal{K} - \mathcal{K}_s^Z\| + \sum_{m\geq n}\sigma_m$ (see (31) in Appendix D.2 for details), and the desired inequality follows by combining Theorem 1 and Theorem 3 (i.e., (5) and (12)). □

Remark 2. *Algorithm 1 with the Nystöm approximation can be decomposed into two parts: (a) Nystöm approximation by truncated singular value decomposition (SVD) (the first $n-1$ eigenvectors from an ℓ-point sample), (b) Recombination from an N-point empirical measure. The complexity of (a) is $\mathcal{O}\big(n\ell^2\big)$, and it can also be approximated by randomized SVD in $\mathcal{O}\big(n^2\ell + \ell^2\log n\big)$ [20]. The cost of part (b) is $\mathcal{O}\big(n\ell N + n^3\log(N/n)\big)$, where $n\ell N$ stems from the evaluation of $k_{1,\mathrm{diag}}$ for all N sampling points. If we do not impose the inequality constraint regarding $k_{1,\mathrm{diag}}$, which still works well in practice, the cost of part (b) becomes $\mathcal{O}\big(\ell N + n^2\ell\log(N/n)\big)$, by using the trick $\frac{1}{N}\sum_{i=1}^N U_{n-1}^\top k(Z,y_i) = U_{n-1}^\top \frac{1}{N}\sum_{i=1}^N k(Z,y_i)$, where $U_{n-1} = [u_1,\dots,u_{n-1}] \in \mathbb{R}^{\ell\times(n-1)}$ is a truncation of the matrix that appears in the Nyström approxiamtion (10,11). So the overall complexity is $\mathcal{O}\big(n\ell N + n\ell^2 + n^3\log(N/n)\big)$ while an approximate algorithm (randomized SVD, without the inequality constraint) runs in $\mathcal{O}\big(\ell N + \ell^2\log n + n^2\ell\log(N/n)\big)$.*

2.4 Kernel Quadrature Using Expectations of Test Functions

Algorithm 1 and the bound (5) can be generally applicable once we obtain a low-rank approximation k_0 as we have seen in Section 2.2 and 2.3. However, since by construction we start by reducing the empirical measure given by D_N, it is inevitable to have the $\Omega(1/N)$ term in the error estimate and performance. We can avoid this limitation by exploiting additional knowledge of expectations.

Let k_0 and k_1 be positive definite kernels with $k = k_0 + k_1$. Let $\boldsymbol{\varphi} = (\varphi_1, \dots, \varphi_{n-1})^\top$ be the vector of test functions that spans $\mathcal{H}_{k_0}$. When we know the expectations of them, i.e., $\int_{\mathcal{X}} \boldsymbol{\varphi}(x)\,\mathrm{d}\mu(x)$, we can actually construct a convex quadrature $Q_n = (w_i, x_i)_{i=1}^n$ satisfying

$$\sum_{i=1}^n w_i \boldsymbol{\varphi}(x_i) = \int_{\mathcal{X}} \boldsymbol{\varphi}(x)\,\mathrm{d}\mu(x), \qquad \sum_{i=1}^n w_i k_1(x_i, x_i) \le \int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x) \tag{13}$$

with a positive probability by an algorithm based on random convex hulls (Appendix D, Algorithm 2).

For this Q_n, we have the following theoretical guarantee (see Theorem 6 in Appendix B):

Theorem 5. *If a convex quadrature Q_n satisfies the condition* (13)*, then we have*

$$\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2 \le 4 \int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x).$$

If k_0 is given the Mercer/Nyström approximations, we immediately have the following guarantees; they correspond to **Mercer** and **Nyström** in Table 1. See also Theorem 9 and 11 for details.

- If $k_0(x,y) = \sum_{m=1}^{n-1} \sigma_m e_m(x) e_m(y)$ is given by the Mercer approximation, we have
$$\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2 \le 4 \sum_{m=n}^{\infty} \sigma_m$$
for a convex quadrature Q_n satisfying (13).
- Let $k_0 = k_{n-1}^Z$ be given by the Nyström approximation (10) with Z being an ℓ-point independent sample from μ (with $\ell > n$). Then, for a convex quadrature Q_n satisfying (13), with probability at least $1-\delta$ (with respect to Z) and $k_{\max} := \sup_{x\in\mathcal{X}} k(x,x)$, we have
$$\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2 \le 4\left(n\sigma_n + \sum_{m>n} \sigma_m\right) + \frac{8(n-1)k_{\max}}{\sqrt{\ell}}\left(1 + \sqrt{2\log\frac{1}{\delta}}\right).$$

3 Numerical Experiments

In this section, we compare our methods with several existing methods. In all the experiments, we used the setting where we can compute $\int_{\mathcal{X}} k(x,y)\,\mathrm{d}\mu(y)$ for $x \in \mathcal{X}$ and $\iint_{\mathcal{X}\times\mathcal{X}} k(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y)$ since then we can evaluate the worst-case error of quadrature formulas explicitly. Indeed, if a quadrature formula Q_n is given by points $X = (x_i)_{i=1}^n$ and weights $\boldsymbol{w} = (w_i)_{i=1}^n$, then we have

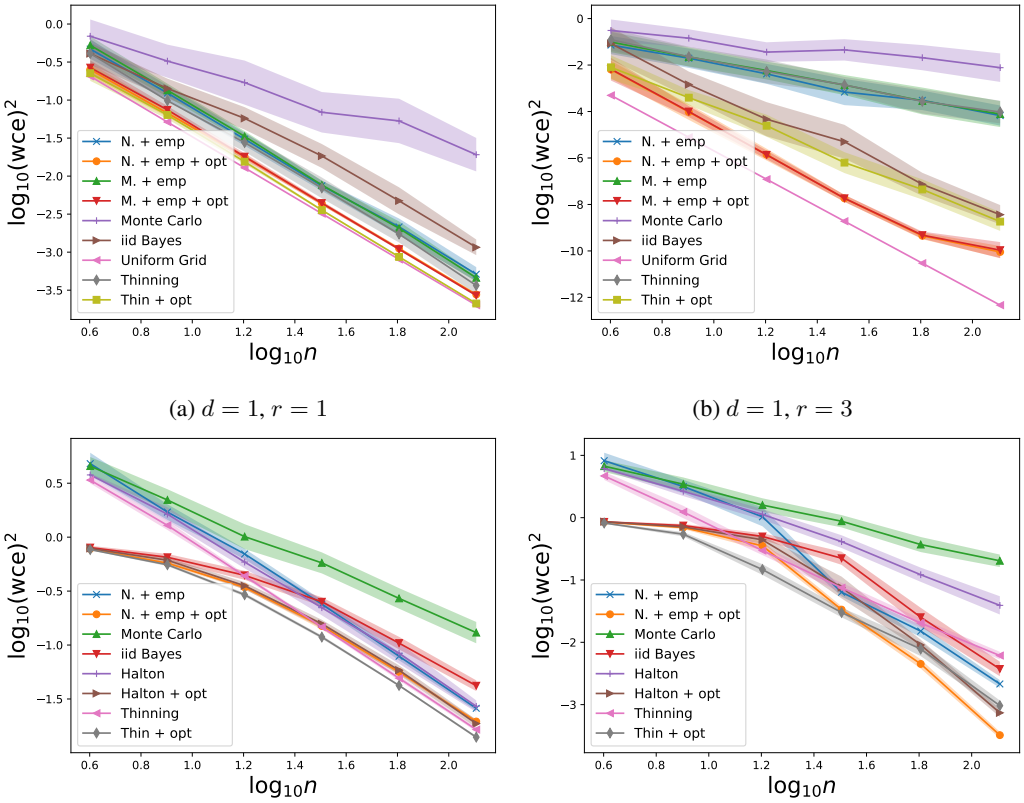
$$\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2 = \boldsymbol{w}^\top k(X,X)\boldsymbol{w} - 2\mathbb{E}_y[\boldsymbol{w}^\top k(X,y)] + \mathbb{E}_{y,y'}[k(y,y')] \tag{14}$$

for independent $y, y' \sim \mu$ under $\int_{\mathcal{X}} \sqrt{k(x,x)}\,\mathrm{d}\mu(x) < \infty$, which is a well-known formula for the worst-case error [19, 58]. An essential remark shown in Huszár and Duvenaud [25] is that the Bayesian quadrature [49] with covariance kernel k given observation at points $(x_i)_{i=1}^n$ (automatically) estimates the integral as $\sum_{i=1}^n w_i f(x_i)$ with $(w_i)_{i=1}^n$ minimizing the above expression. Once given points $(x_i)_{i=1}^n$ and additional knowledge of expectations, we can compute the optimal weights $(w_i)_{i=1}^n$ by solving a convex quadratic programming (CQP), either without any restrictions or with the condition that $(w_i)_{i=1}^n$ is convex. Although the former can be solved by matrix inversion, we have used the optimizer Gurobi[3] for both CQPs to avoid numerical instability. For the recombination part, we have modified the Python library by Cosentino et al. [11] implementing the algorithm of [61].

Our theoretical bounds are close to optimal in classic examples and we see that the algorithm even outperforms the theory in practice especially in Section 3.1. We also execute a measure reduction of a large discrete measure in terms of Gaussian RKHS and our methods shows a fast convergence rate in two ML datasets in Section 3.2. [4]

[3] Version 9.1.2, `https://www.gurobi.com/`

[4] All done on a MacBook Pro, CPU: 2.4 GHz Quad-Core Intel Core i5, RAM: 8 GB 2133 MHz LPDDR3.



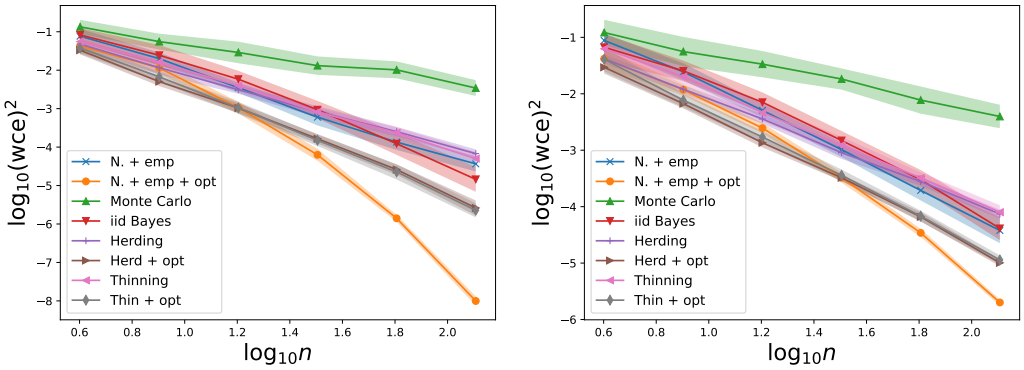
(a) $d = 1, r = 1$

(b) $d = 1, r = 3$

(c) $d = 2, r = 1$

(d) $d = 3, r = 3$

Figure 1: Periodic Sobolev spaces with kernel $k_r^{\otimes d}$: The average of $\log_{10}(\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2)$ over 20 trials is plotted for each method of obtaining Q_n. The shaded regions show their standard deviation. The worst computational time per one trial was 57 seconds of **Thin + opt** in $(d, r, n) = (3, 3, 128)$, where **Thinning** was 56 seconds and **N. + emp [+ opt]** was 22 seconds.



(a) 3D Road Network data

(b) Power Plant data

Figure 2: Measure reduction in Gaussian RKHS with two ML datasets: The average of $\log_{10}(\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2)$ over 20 trials is plotted for each method of obtaining Q_n. The shaded regions show their standard deviation. The worst computational time per one trial was 14 seconds of **Thinning [+ opt]** in Power Plant data with $n = 128$, where **N. + emp [+ opt]** was 6.3 seconds.

3.1 Periodic Sobolev Spaces with Uniform Measure

For a positive integer r, consider the Sobolev space of functions on $[0,1]$ endowed with the norm $\|f\|^2 = (\int_0^1 f(x)\,\mathrm{d}x)^2 + (2\pi)^{2r}\int_0^1 f^{(r)}(x)^2\,\mathrm{d}x$, where f and its derivatives $f^{(1)},\dots f^{(r)}$ are periodic (i.e., $f(0)=f(1)$ and so forth). This function space can be identifies as the RKHS of the kernel

$$k_r(x,y) = 1 + \frac{(-1)^{r-1}(2\pi)^{2r}}{(2r)!}B_{2r}(|x-y|)$$

for $x,y\in[0,1]$, where B_{2r} is the $2r$-th Bernoulli polynomial [66, 3]. If we let μ be the uniform measure on $[0,1]$, the normalized eigenfunctions (of the integral operator) are 1, $c_m(\cdot)=\sqrt{2}\cos(2\pi m\,\cdot)$ and $s_m(\cdot)=\sqrt{2}\sin(2\pi m\,\cdot)$ for $m=1,2,\dots$, and the corresponding eigenvalues are 1 and m^{-2r} (both for c_m and s_m). Although the rectangle formula $f\mapsto n^{-1}\sum_{i=1}^n f(i/n)$ (a.k.a. `Uniform Grid` below) is known to be optimal for this kernel [69, 47] in the sense of worst-case error, this RKHS is commonly used for testing the efficiency of general kernel quadrature methods [3, 7, 28]. We also consider its multivariate extension on $[0,1]^d$, i.e., the RKHS given by the product kernel $k_r^{\otimes d}(\boldsymbol{x},\boldsymbol{y}) \coloneqq \prod_{i=1}^d k_r(x_i,y_i)$ for $\boldsymbol{x}=(x_1,\dots,x_d),\boldsymbol{y}=(y_1,\dots,y_d)\in[0,1]^d$.

We carried out the experiment for $(d,r)=(1,1),(1,3),(2,1),(3,3)$. For each (d,r), we compared the following algorithms for n-point quadrature rules with $n\in\{4,8,16,32,64,128\}$.

N. + emp, N. + emp + opt: We used the functions $u_i^\top k(Z,\cdot)$ $(i=1,\dots,n-1)$ given by the Nyström approximation (11) with $s=n-1$ as test functions $\varphi_1,\dots,\varphi_{n-1}$ in Algorithm 1. The set Z was given as an $(\ell=)10n$-point independent sample from μ. We used $N=n^2$ samples from μ. In '**+ opt**' we additionally optimized the *convex* weights using (14)

M. + emp, M. + emp + opt ($d=1$): We used the first $n-1$ functions of the sequence of eigenfunctions $1,c_1,s_1,c_2,s_2,\dots$ as test functions $\varphi_1,\dots,\varphi_{n-1}$ in Algorithm 1. We used $N=n^2$ samples from μ. In '**+ opt**' we additionally optimized the *convex* weights using (14).

Monte Carlo, iid Bayes: With an n-point independent sample $(x_i)_{i=1}^n$ from μ, we used uniform weights $1/n$ in **Monte Carlo** and the weights optimized using (14) in **iid Bayes**.

Uniform Grid ($d=1$): We used the rectangle formula $f\mapsto n^{-1}\sum_{i=1}^n f(i/n)$. This is known to be optimal [not just up to constant, but exactly; 69, 47], and thus equivalent to the Bayesian quadrature on the uniform grid, i.e., the weights are already optimized.

Halton, Halton + opt ($d\geq 2$): For an n-point sequence given by the Halton sequence with Owen scrambling [21, 50], the uniform weights $w_i=1/n$ is adopted in **Halton** and the weights are additionally optimized using (14) in **Halton + opt**.

Thinning, Thin + opt: Given an N-point independent sample $(y_i)_{i=1}^N$ with $N=n^2$ from μ, an n-point subset $(x_i)_{i=1}^n$ taken from a KT++ algorithm (kernel thinning [14, 15] combined with Compress++ algorithm [56] with the oversampling parameter $\mathfrak{g}=\min\{4,\log_2 n\}$, implemented with GoodPoints package: `https://github.com/microsoft/goodpoints`) is adopted in **Thinning**. In **'+ opt'** we additionally optimized the *convex* weights using (14).

The results are given in Figure 1. In $d=1$, the optimal rate given by **Uniform Grid** is known to be $\mathcal{O}(n^{-2r})$. As the uniform sampling is equal to the *optimized distribution* of Bach [3] in this case, **iid Bayes** also achieves this rate up to log factors. Although our theoretical guarantee for **M. + emp** is $\mathcal{O}(n^{1-2r}+N^{-1})$ with $N=n^2$ (Corollary 2), in the case $(d,r)=(1,1)$, we can observe that in the experiment it is better than **iid Bayes** and close to the optimal error of **Uniform Grid**, but slightly worse than **Thinning**. Moreover, **N. + emp**, which does not use the information of spectral decomposition, is remarkably almost as accurate as **M. + emp** in $d=1$. Furthermore, if we additionally use the knowledge of expectations, which **iid Bayes** is already doing, **M./N. + emp + opt** become surprisingly accurate even with $N=n^2$. They are worse than **Thinn + opt** when $r=1$, but well outperform it when $r=3$. Nonlineality in the graph of these methods when $(d,r,n)=(1,3,128)$ should be from numerical accuracy of the CQP solver (see also Section E.1).

The accuracy of **N. + emp + opt** becomes more remarkable in multivariate cases. It behaves almost the same as **Halton + opt** in $d=2$ and clearly beats it in $d=3$. Also, the sudden jump of our methods around $n=30$ in $(d,r)=(3,3)$ seems to be caused by the jump of eigenvalues. Indeed, for the integral operator given by $k_3^{\otimes 3}$ with uniform measure, the eigenspace of the largest eigenvalue

1 is of dimension 27, and the next largest eigenvalue is $1/64$. Again in the latter case, **N. + emp + opt** outperforms **Thin + opt**, and these results suggest that our method works better when there is a strong spectral decay, as is explicitly incorporated in our algorithm.

Note also that we can compare Figure 1 with Belhadji et al. [7, Figure 1] which includes some other methods such as DPPs, herding and sequential Bayesian quadrature, as we did experiments under almost the same setting. In particular, in the case $(d, r) = (1, 3)$ where the eigenvalue decay is fast, we see that our method substantially outperforms the sequential Bayesian quadrature.

3.2 Measure Reduction in Machine Learning Datasets

We used two datasets from UCI Machine Learning Repository (`https://archive.ics.uci.edu/ml/datasets/`). We set μ as the equally weighted measure over (a subset of) the data points $X = (X^{(1)}, \dots, X^{(d)})^\top$ ($d = 3, 5$, respectively), where each entry is centered and normalized. We considered the Gaussian kernel $\exp(-\|x - y\|^2/(2\lambda^2))$ whose hyperparameter λ is determined by *median heuristics* [17], and compared the performance of **N. + emp**, **N. + emp + opt** (with $\ell = 10n$, $N = n^2$), **Monte Carlo**, **iid Bayes**, **Thinning**, **Thin + opt**. We also added **Herding**, an equally weighted greedy algorithm with global optimization [10], and its weight optimization **Herd + opt** within *convex* quadrature given by (14). We conducted the experiment for $n \in \{4, 8, 16, 32, 64, 128\}$.

The first is **3D Road Network Data Set** [31]. The original dataset is 3-dimensional real vectors at 434874 points. To be able to compute the worst-case error (14) efficiently to evaluate each kernel quadrature, we used a random subset $\mathcal{X}$ of size $43487 = \lfloor 434874/10 \rfloor$ (fixed throughout the experiment) and defined μ as the uniform measure on it. We determined λ with the median heuristic by using a random subset of $\mathcal{X}$ with size 10000 and used the same $\mathcal{X}$ and λ throughout the experiment. The second is **Combined Cycle Power Plant Data Set** [32, 63]. The original dataset is 5-dimensional real vectors at 9568 points. We set the whole data as $\mathcal{X}$ and defined μ as the uniform measure on it. We determined λ with median heuristics by using the whole $\mathcal{X}$.

Figure 2 shows the results. We can observe that in both experiments **N. + emp + opt** successfully exploits the fast spectral decay of Gaussian kernel and significantly outperforms other methods. Also, even without using the knowledge of any expectations, **N. + emp** (and **Thinning**) show a decent convergence rate comparable to **Herding** or **iid Bayes**, which actually use the additional information. See also the end of Section E.2 for the plot of $\mathrm{wce}(Q_n; \mathcal{H}_k, \mu')$ for another set of empirical data μ'.

4 Concluding Remarks

We leveraged a classical measure reduction tool, recombination, with spectral properties of kernels to construct kernel quadrature rules with positive weights. The resulting algorithms show strong benchmark performance despite their restriction to convex weights. Our method has also recently been applied to Bayesian inference problems [1].

Although our method is applicable to fairly general situations, the usage or performance can be limited when it is difficult or inefficient to directly sample from the target measure μ. Hence, an interesting follow up questions, is how one could replace the i.i.d. samples with smarter sampling (DPP, importance sampling, etc) before the recombination is carried out. Further, our theoretical results do not fully explain the empirical superiority; especially the $1/\sqrt{\ell}$ term does not match the experiments and it is a challenging future research question to reduce this theoretical gap. Nevertheless, we believe our method is the first generally applicable algorithm with a guarantee from the spectral decay.

Acknowledgments and Disclosure of Funding

The authors would like to thank Chris Oates and Toni Karvonen for helpful remarks and discussions. The authors are also grateful to anonymous reviewers for detailed and constructive discussions that improved the paper. Harald Oberhauser and Terry Lyons are supported by the DataSıg Program [EP/S026347/1], the Alan Turing Institute [EP/N510129/1], the Oxford-Man Institute, and the Hong Kong Innovation and Technology Commission (InnoHK Project CIMDA).

References

[1] M. Adachi, S. Hayakawa, M. Jørgensen, H. Oberhauser, and M. A. Osborne. Fast Bayesian inference with batch Bayesian quadrature via kernel recombination. In *Advances in Neural Information Processing Systems*, 2022. doi: 10.48550/arXiv.2206.04734.

[2] A. Anastasiou, A. Barp, F.-X. Briol, B. Ebner, R. E. Gaunt, F. Ghaderinezhad, J. Gorham, A. Gretton, C. Ley, Q. Liu, L. Mackey, C. J. Oates, G. Reinert, and Y. Swan. Stein's method meets statistics: A review of some recent developments. *arXiv preprint arXiv:2105.03481*, 2021.

[3] F. Bach. On the equivalence between kernel quadrature rules and random feature expansions. *The Journal of Machine Learning Research*, 18(1):714–751, 2017.

[4] F. Bach, S. Lacoste-Julien, and G. Obozinski. On the equivalence between herding and conditional gradient algorithms. In *International Conference on Machine Learning*, pages 1355–1362, 2012.

[5] C. Bayer and J. Teichmann. The proof of Tchakaloff's theorem. *Proceedings of the American Mathematical Society*, 134(10):3035–3040, 2006.

[6] A. Belhadji. An analysis of Ermakov–Zolotukhin quadrature using kernels. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

[7] A. Belhadji, R. Bardenet, and P. Chainais. Kernel quadrature with DPPs. In *Advances in Neural Information Processing Systems*, volume 32, pages 12907–12917, 2019.

[8] A. Belhadji, R. Bardenet, and P. Chainais. Kernel interpolation with continuous volume sampling. In *International Conference on Machine Learning*, pages 725–735. PMLR, 2020.

[9] F.-X. Briol, C. Oates, M. Girolami, and M. A. Osborne. Frank–Wolfe Bayesian quadrature: Probabilistic integration with theoretical guarantees. *Advances in Neural Information Processing Systems*, 28:1162–1170, 2015.

[10] Y. Chen, M. Welling, and A. Smola. Super-samples from kernel herding. In *Conference on Uncertainty in Artificial Intelligence*, pages 109–116, 2010.

[11] F. Cosentino, H. Oberhauser, and A. Abate. A randomized algorithm to reduce the support of discrete measures. In *Advances in Neural Information Processing Systems*, volume 33, pages 15100–15110, 2020.

[12] S. De Marchi, R. Schaback, and H. Wendland. Near-optimal data-independent point locations for radial basis function interpolation. *Advances in Computational Mathematics*, 23(3):317–330, 2005.

[13] P. Drineas, M. W. Mahoney, and N. Cristianini. On the Nyström method for approximating a Gram matrix for improved kernel-based learning. *The Journal of Machine Learning Research*, 6(12):2153–2175, 2005.

[14] R. Dwivedi and L. Mackey. Kernel thinning. In *Conference on Learning Theory*, pages 1753–1753. PMLR, 2021.

[15] R. Dwivedi and L. Mackey. Generalized kernel thinning. In *International Conference on Learning Representations*, 2022.

[16] G. E. Fasshauer and M. J. McCourt. Stable evaluation of Gaussian radial basis function interpolants. *SIAM Journal on Scientific Computing*, 34(2):A737–A762, 2012. URL `https://doi.org/10.1137/110824784`.

[17] D. Garreau, W. Jitkrittum, and M. Kanagawa. Large sample analysis of the median heuristic. *arXiv preprint arXiv:1707.07269*, 2017.

[18] J.-y. Gotoh, A. Takeda, and K. Tono. DC formulations and algorithms for sparse optimization problems. *Mathematical Programming*, 169(1):141–176, 2018.

[19] A. Gretton, K. Borgwardt, M. Rasch, B. Schölkopf, and A. Smola. A kernel method for the two-sample-problem. *Advances in neural information processing systems*, 19, 2006.

[20] N. Halko, P.-G. Martinsson, and J. A. Tropp. Finding structure with randomness: Probabilistic algorithms for constructing approximate matrix decompositions. *SIAM review*, 53(2):217–288, 2011.

[21] J. H. Halton. On the efficiency of certain quasi-random sequences of points in evaluating multi-dimensional integrals. *Numerische Mathematik*, 2(1):84–90, 1960.

[22] S. Hayakawa. Monte Carlo cubature construction. *Japan Journal of Industrial and Applied Mathematics*, 38:561–577, 2021.

[23] S. Hayakawa, T. Lyons, and H. Oberhauser. Estimating the probability that a given vector is in the convex hull of a random sample. *arXiv preprint arXiv:2101.04250*, 2021.

[24] J. B. Hough, M. Krishnapur, Y. Peres, B. Virág, et al. Determinantal processes and independence. *Probability surveys*, 3:206–229, 2006.

[25] F. Huszár and D. Duvenaud. Optimally-weighted herding is Bayesian quadrature. In *Conference on Uncertainty in Artificial Intelligence*, pages 377–386, 2012.

[26] R. Jin, T. Yang, M. Mahdavi, Y.-F. Li, and Z.-H. Zhou. Improved bounds for the Nyström method with application to kernel classification. *IEEE Transactions on Information Theory*, 59 (10):6939–6949, 2013.

[27] M. Kanagawa, B. K. Sriperumbudur, and K. Fukumizu. Convergence guarantees for kernel-based quadrature rules in misspecified settings. *Advances in Neural Information Processing Systems*, 29:3296–3304, 2016.

[28] M. Kanagawa, B. K. Sriperumbudur, and K. Fukumizu. Convergence analysis of deterministic kernel-based quadrature rules in misspecified settings. *Foundations of Computational Mathematics*, 20(1):155–194, 2020.

[29] T. Karvonen, C. J. Oates, and S. Särkkä. A Bayes–Sard cubature method. In *Advances in Neural Information Processing Systems*, volume 31, pages 5886–5897, 2018.

[30] T. Karvonen, C. Oates, and M. Girolami. Integration in reproducing kernel hilbert spaces of Gaussian kernels. *Mathematics of Computation*, 90(331):2209–2233, 2021.

[31] M. Kaul, B. Yang, and C. S. Jensen. Building accurate 3d spatial networks to enable next generation intelligent transportation systems. In *2013 IEEE 14th International Conference on Mobile Data Management*, volume 1, pages 137–146. IEEE, 2013.

[32] H. Kaya, P. Tüfekci, and F. S. Gürgen. Local and global learning methods for predicting power of a combined gas & steam turbine. In *Proceedings of the International Conference on Emerging Trends in Computer and Electronics Engineering*, pages 13–18, 2012.

[33] V. Koltchinskii and E. Giné. Random matrix approximation of spectra of integral operators. *Bernoulli*, 6(1):113–167, 2000.

[34] S. Kumar, M. Mohri, and A. Talwalkar. Sampling methods for the Nyström method. *The Journal of Machine Learning Research*, 13(1):981–1006, 2012.

[35] A. Kyrillidis, S. Becker, V. Cevher, and C. Koch. Sparse projections onto the simplex. In *International Conference on Machine Learning*, pages 235–243. PMLR, 2013.

[36] F. Larkin. Optimal approximation in Hilbert spaces with reproducing kernel functions. *Mathematics of Computation*, 24(112):911–921, 1970.

[37] P. Li, S. S. Rangapuram, and M. Slawski. Methods for sparse and low-rank recovery under simplex constraints. *Statistica Sinica*, 30(2):557–577, 2020.

[38] C. Litterer and T. Lyons. High order recombination and an application to cubature on Wiener space. *The Annals of Applied Probability*, 22(4):1301–1327, 2012.

[39] F. Liu, X. Huang, Y. Chen, and J. A. Suykens. Random features for kernel approximation: A survey on algorithms, theory, and beyond. *arXiv preprint arXiv:2004.11154*, 2020.

[40] Q. Liu and J. Lee. Black-box importance sampling. In *Artificial Intelligence and Statistics*, pages 952–961. PMLR, 2017.

[41] J. Lu, G. Cheng, and H. Liu. Nonparametric heterogeneity testing for massive data. *arXiv preprint arXiv:1601.06212v1*, 2016.

[42] T. Lyons and N. Victoir. Cubature on Wiener space. *Proceedings of the Royal Society of London Series A*, 460:169–198, 2004.

[43] A. Maalouf, I. Jubran, and D. Feldman. Fast and accurate least-mean-squares solvers. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 32, pages 8305–8316, 2019.

[44] H. Q. Minh. Some properties of Gaussian reproducing kernel Hilbert spaces and their implications for function approximation and learning theory. *Constructive Approximation*, 32(2): 307–338, 2010.

[45] H. Q. Minh, P. Niyogi, and Y. Yao. Mercer's theorem, feature maps, and smoothing. In *International Conference on Computational Learning Theory*, pages 154–168. Springer, 2006.

[46] K. Muandet, K. Fukumizu, B. Sriperumbudur, and B. Schölkopf. Kernel mean embedding of distributions: A review and beyond. *Foundations and Trends® in Machine Learning*, 10(1-2): 1–141, 2017.

[47] E. Novak. *Deterministic and stochastic error bounds in numerical analysis*. Springer, 1988.

[48] C. Oates, M. Girolami, and N. Chopin. Control functionals for Monte Carlo integration. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 79:695–718, 2017.

[49] A. O'Hagan. Bayes–Hermite quadrature. *Journal of Statistical Planning and Inference*, 29(3): 245–260, 1991.

[50] A. B. Owen. A randomized Halton algorithm in R. *arXiv preprint arXiv:1706.02808*, 2017.

[51] V. Pan. On the complexity of a pivot step of the revised simplex algorithm. *Computers & Mathematics with Applications*, 11(11):1127 – 1140, 1985.

[52] M. Pilanci, L. El Ghaoui, and V. Chandrasekaran. Recovery of sparse probability measures via convex programming. In *Advances in Neural Information Processing Systems*, volume 25, pages 2420–2428, 2012.

[53] A. Rahimi and B. Recht. Random features for large-scale kernel machines. In *Advances in Neural Information Processing Systems*, volume 20, pages 1177–1184, 2007.

[54] A. Sard. Best approximate integration formulas; best approximation formulas. *American Journal of Mathematics*, 71(1):80–91, 1949.

[55] R. Shamir. The efficiency of the simplex method: A survey. *Management Science*, 33(3): 301–334, 1987.

[56] A. Shetty, R. Dwivedi, and L. Mackey. Distribution compression in near-linear time. In *International Conference on Learning Representations*, 2022.

[57] L. F. South, T. Karvonen, C. Nemeth, M. Girolami, C. Oates, et al. Semi-exact control functionals from sard's method. *arXiv preprint arXiv:2002.00033*, 2020.

[58] B. K. Sriperumbudur, A. Gretton, K. Fukumizu, B. Schölkopf, and G. R. Lanckriet. Hilbert space embeddings and metrics on probability measures. *The Journal of Machine Learning Research*, 11:1517–1561, 2010.

[59] I. Steinwart and C. Scovel. Mercer's theorem on general domains: On the interaction between measures, kernels, and RKHSs. *Constructive Approximation*, 35(3):363–417, 2012.

[60] V. Tchakaloff. Formules de cubature mécanique à coefficients non négatifs. *Bulletin des Sciences Mathématiques*, 81:123–134, 1957.

[61] M. Tchernychova. *Carathéodory cubature measures*. PhD thesis, University of Oxford, 2015.

[62] A. Tompkins and F. Ramos. Fourier feature approximations for periodic kernels in time-series modelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[63] P. Tüfekci. Prediction of full load electrical power output of a base load operated combined cycle power plant using machine learning methods. *International Journal of Electrical Power & Energy Systems*, 60:126–140, 2014.

[64] P. Turner, J. Liu, and P. Rigollet. A statistical perspective on coreset density estimation. In *International Conference on Artificial Intelligence and Statistics*, pages 2512–2520. PMLR, 2021.

[65] U. Wagner and E. Welzl. A continuous analogue of the upper bound theorem. *Discrete & Computational Geometry*, 26(2):205–219, 2001.

[66] G. Wahba. *Spline Models for Observational Data*. Society for Industrial and Applied Mathematics, 1990.

[67] J. G. Wendel. A problem in geometric probability. *Mathematica Scandinavica*, 11(1):109–111, 1963.

[68] C. Williams and M. Seeger. Using the Nyström method to speed up kernel machines. In *Advances in Neural Information Processing Systems*, volume 13, pages 661–667, 2000.

[69] A. A. Zhensykbaev. Monosplines of minimal norm and the best quadrature formulae. *Russian Mathematical Surveys*, 36(4):121–180, 1981.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 4.
   (c) Did you discuss any potential negative societal impacts of your work? [No] This is an improvement on a specific numerical method and we believe it does not have a forseeable negative social impact.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 2 and Appendix.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Section 2 and Appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Included in the supplementary material as a Zip file.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 3.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Figures in Section 3.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See the footnote 3 in page 7 for the resource. We also included the computational time in each Figure (for the largest case) as well as the supplementary material (for all cases).
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 3 and the README file in the supplementary material.
   (b) Did you mention the license of the assets? [Yes] See the README file in the supplementary material.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] See the README file in the supplementary material.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Outline of the Appendix

Appendix B contains general results from which the results presented in the main text, in particular Theorem 1, follow as special cases. Appendix C contains the proofs of these theoretical results and needed technical lemmas. Appendix D shows that if the expectations $\int \varphi_i(x)\,\mathrm{d}\mu(x)$ are known, then this knowledge can be used to further improve the theoretical bounds; it also gives a simple modification of Algorithm 1 doing this efficiently. Appendix E provides additional numerical experiments and benchmarks.

B Theoretical Results and Remarks

In this section, we present theoretical results that include our main results as a special case. The proofs are given in Section C.

Notation. For simplicity, for a quadrature Q_n given by points $(x_i)_{i=1}^n$ and weights $(w_i)_{i=1}^n$ and a probability measure μ, we denote the integration of an integrable function f on $\mathcal{X}$ with respect to these measures by

$$Q_n(f) = \sum_{i=1}^n w_i f(x_i), \qquad \mu(f) = \int_{\mathcal{X}} f(x)\,\mathrm{d}\mu(x),$$

respectively. We also write the inner-product and norm of an RKHS $\mathcal{H}_k$ by $\langle \cdot, \cdot \rangle_{\mathcal{H}_k}$ and $\|\cdot\|_{\mathcal{H}_k}$. Furthermore, we use the probability simplex Δ^n and convex hull $\operatorname{conv} A$ of a set $A \subset \mathbb{R}^d$ in the proofs:

$$\Delta^n := \left\{ (w_i)_{i=1}^n \,\middle|\, w_i \geq 0,\, \sum_{i=1}^n w_i = 1 \right\},\ \operatorname{conv} A := \left\{ \sum_{i=1}^n w_i a_i \,\middle|\, (w_i) \in \Delta^n,\, a_i \in A,\, n \geq 1 \right\}.$$

B.1 Quantitative Results

We work under the following setting as in the assumption of Theorem 1.

Assumption A. *μ is a Borel probability measure on $\mathcal{X}$, and k is a positive semi-definite kernel on $\mathcal{X}$ such that $\int_{\mathcal{X}} k(x,x)\,\mathrm{d}\mu(x) < \infty$. Further, k_0 is a positive semi-definite kernel on $\mathcal{X}$ such that $k_1 := k - k_0$ is a positive semi-definite kernel on $\mathcal{X}$.*

The following is a general result regarding a quadrature formula exactly integrating functions in $\mathcal{H}_{k_0}$.

Theorem 6. *Under Assumption A, if an n-point convex quadrature Q_n on $\mathcal{X}$ satisfies $Q_n(f) = \mu(f)$ for any $f = k_0(\cdot, x)$ with $x \in \mathcal{X}$, we have*

$$\operatorname{wce}(Q_n; \mathcal{H}_k, \mu) \leq Q_n(g) + \mu(g), \tag{15}$$

where g is the function given by $g(x) = \sqrt{k_1(x,x)}$. In particular, the following assertions hold for such a quadrature Q_n:

(a) We have $\operatorname{wce}(Q_n; \mathcal{H}_k, \mu) \leq 2\|g\|_\infty = 2\sup_{x\in\mathcal{X}} \sqrt{k_1(x,x)}$.

(b) If we additionally have $Q_n(g) \leq \mu(g)$, then we have $\operatorname{wce}(Q_n; \mathcal{H}_k, \mu) \leq 2\mu(g)$.

(c) If we additionally have $Q_n(g^2) \leq \mu(g^2)$ instead of (b), we still have

$$\operatorname{wce}(Q_n; \mathcal{H}_k, \mu)^2 \leq 4 \int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x).$$

Remark 3. *For a Borel probability measure ν on $\mathcal{X}$ and a nonnegative function $h : \mathcal{X} \to \mathbb{R}_{\geq 0}$, we have an inequality $\int_{\mathcal{X}} \sqrt{h(x)}\,\mathrm{d}\nu(x) \leq \left(\int_{\mathcal{X}} h(x)\,\mathrm{d}\nu(x)\right)^{1/2}$, so the above $\mu(g)$ can be upper bounded by $\int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x)$, which is equal to the trace of the integral operator given by k_1. Also, the assumption in Theorem 6 can be weakened to the integrability of $\sqrt{k(x,x)}$ from the same inequality as you can see in the proof.*

We can combine Theorem 6 with an empirical approximation of μ to obtain the following result, which essentially implies Theorem 1.

Theorem 7. *Under Assumption A, let D_N be a set of N independent samples from μ, and $\widetilde{\mu}_N$ be its empirical measure, i.e., $\widetilde{\mu}_N = \frac{1}{N}\sum_{y\in D_N}\delta_y$. Then, if an n-point convex quadrature Q_n on $\mathcal{X}$ satisfies $Q_n(f) = \widetilde{\mu}_N(f)$ for any $f = k_0(\cdot, x)$ with $x \in \mathcal{X}$, we have*

$$\mathbb{E}\big[\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2\big] \le 2\mathbb{E}\big[(Q_n(g) + \widetilde{\mu}_N(g))^2\big] + \frac{2c_{k,\mu}}{N}, \tag{16}$$

where $g(x) \coloneqq \sqrt{k_1(x,x)}$ and $c_{k,\mu} \coloneqq \int_{\mathcal{X}} k(x,x)\,\mathrm{d}\mu(x) - \iint_{\mathcal{X}\times\mathcal{X}} k(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y)$. In particular, the following assertions hold for such a quadrature Q_n:

(a) We have $\mathbb{E}\big[\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2\big] \le 8\sup_{x\in\mathcal{X}} k_1(x,x) + 2c_{k,\mu}/N$.

(b) If we additionally always require $Q_n(g) \le \widetilde{\mu}_N(g)$, then we have

$$\mathbb{E}\big[\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2\big] \le 8\int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x) + \frac{2c_{k,\mu}}{N}.$$

The requirement $Q_n(g) \le \widetilde{\mu}_N(g)$ can be replaced by $Q_n(g^2) \le \widetilde{\mu}_N(g^2)$

Although we have assumed $k - k_0$ is positive semi-definite in the previous assertions, the uniform bound works without the assumption as follows.

Proposition 1. *Let μ be a Borel probability measure on $\mathcal{X}$. Let k and k_0 be positive semi-definite kernels on $\mathcal{X}$ satisfying $\int_{\mathcal{X}}\sqrt{k(x,x)}\,\mathrm{d}\mu(x), \int_{\mathcal{X}}\sqrt{k_0(x,x)}\,\mathrm{d}\mu(x) < \infty$. If an n-point convex quadrature Q_n on $\mathcal{X}$ satisfies $Q_n(f) = \mu(f)$ for any $k_0(\cdot, x)$ with $x \in \mathcal{X}$, we have*

$$\mathrm{wce}(Q_n; \mathcal{H}_k, \mu) \le 2\sup_{x,y\in\mathcal{X}}\sqrt{|k(x,y) - k_0(x,y)|}.$$

Furthermore, if $\dim \mathcal{H}_{k_0} < n$, there exists an n-point convex quadrature Q_n satisfying $Q_n(f) = \mu(f)$ for each $f = k_0(\cdot, x)$.

In this paper, we focus on the cases where k_0 is either given by the truncated Mercer decomposition or Nyström approximation. For many important kernels, however, we may also use the random Fourier features [53] or its periodic version [62] which can easily be combined with Proposition 1, but it is beyond the scope of this paper to choose its appropriate variant for each kernel [see 39, for a list of variants].

B.2 Existence Results

The existence of quadrature formulas satisfying the estimate of Theorem 6 or Theorem 7 is guaranteed when $\dim \mathcal{H}_{k_0} < n$.

Theorem 8. *Under Assumption A, if $\dim \mathcal{H}_{k_0} < n$, there exists an n-point convex quadrature Q_n satisfying $Q_n(f) = \mu(f)$ for each $f = k_0(\cdot, x)$. This still holds even if we additionally require $Q_n(g) \le \mu(g)$ or $Q_n(g^2) \le \mu(g^2)$ for $g(x) = \sqrt{k_1(x,x)}$.*

Remark 4. *This also implies the existence result of Q_n satisfying the condition in Theorem 7 if we replace μ by $\widetilde{\mu}_N$.*

The algorithm for constructing a kernel quadrature with Theorem 7 is given in the main body, see Algorithm 1. The ones with Theorem 6 requires further knowledge of the expectation of test functions, i.e., the values of $\int_{\mathcal{X}}\varphi_i(x)\,\mathrm{d}\mu(x)$ with $\mathcal{H}_{k_0} \subset \mathrm{span}\{\varphi_1, \dots, \varphi_{n-1}\}$. Under this additional information, we have an algorithm (Algorithm 2) based on random sampling given in the following section.

B.3 Eigenvalue estimate for Gaussian kernels

We provide a proof of a folklore estimate on the eigenvalues of integral operators given by a Gaussian kernel. Let $k(x,y) = \exp(-\frac{1}{2\ell^2}(x-y)^2)$ for an $\ell > 0$ and $x, y \in \mathbb{R}$. Then, it has the following expansion [44, 30]:

$$k(x,y) = \sum_{m=0}^{\infty}\varphi_m(x)\varphi_m(y), \qquad \varphi_m(x) = \frac{1}{\ell^m\sqrt{m!}}x^m\exp\left(-\frac{x^2}{2\ell^2}\right). \tag{17}$$

Let μ be a Borel probability measure supported on a compact domain, i.e., $\mu(\{x \in \mathbb{R} \mid |x| > R\}) = 0$ for some $R > 0$. Let us consider the RKHS given by k over $\mathcal{X} := \operatorname{supp} \mu$.

Recall that σ_n is the n-th eigenvalue of the integral operator

$$\mathcal{K} : L^2(\mu) \to L^2(\mu); \quad f \mapsto \mathcal{K}f = \int_{\mathcal{X}} k(\cdot, y) f(y) \, \mathrm{d}\mu(y).$$

From the minimax property of eigenvalues of compact Hermitian operators, we have

$$\begin{aligned}
\sigma_n &= \inf_{g_1,\dots,g_{n-1} \in L^2(\mu)} \sup_{f \in L^2(\mu) \cap \{g_1,\dots,g_{n-1}\}^\perp,\, \|f\|_{L^2(\mu)}=1} \langle f, \mathcal{K}f \rangle_{L^2(\mu)} \\
&\leq \sup_{f \in L^2(\mu) \cap \{\varphi_0,\dots,\varphi_{n-2}\}^\perp,\, \|f\|_{L^2(\mu)}=1} \langle f, \mathcal{K}f \rangle_{L^2(\mu)} ,
\end{aligned}$$

where the orgthogonal complement is taken in terms of $L^2(\mu)$-inner product and φ_m are functions given in (17). They are indeed in $L^2(\mu)$ as μ is compactly supported.

Now, let $k_n(x, y) := \sum_{m=n-1}^{\infty} \varphi_m(x)\varphi_m(y)$. For an $f \in L^2(\mu) \cap \{\varphi_0, \dots, \varphi_{n-2}\}^\perp$, we have

$$\begin{aligned}
\langle f, \mathcal{K}f \rangle_{L^2(\mu)} &= \iint_{\mathcal{X}\times\mathcal{X}} f(x) k(x, y) f(y) \, \mathrm{d}\mu(y) \, \mathrm{d}\mu(x) \\
&= \iint_{\mathcal{X}\times\mathcal{X}} f(x) k_n(x, y) f(y) \, \mathrm{d}\mu(y) \, \mathrm{d}\mu(x) \\
&\leq \iint_{\mathcal{X}\times\mathcal{X}} f(x) \sqrt{k_n(x, x)} \sqrt{k_n(y, y)} f(y) \, \mathrm{d}\mu(y) \, \mathrm{d}\mu(x) && \text{(positive definiteness)} \\
&= \left(\int_{\mathcal{X}} \sqrt{k_n(x, x)} f(x) \, \mathrm{d}\mu(x) \right)^2 \\
&\leq \left(\int_{\mathcal{X}} k_n(x, x) \, \mathrm{d}\mu(x) \right) \|f\|^2_{L^2(\mu)}. && \text{(Cauchy–Schwarz)}
\end{aligned}$$

Therefore, we have the estimate $\sigma_n \leq \int_{\mathcal{X}} k_n(x, x) \, \mathrm{d}\mu(x)$. We have

$$k_n(x, x) = \sum_{m=n-1}^{\infty} \frac{1}{m!} \left(\frac{x}{\ell}\right)^{2m} \exp\left(-\left(\frac{x}{\ell}\right)^2 \right),$$

and this can be regarded as the remainder term of the Maclaurin expansion, so there is a $\theta \in (0, 1)$ such that

$$k_n(x, x) = \frac{1}{(n-1)!} \exp\left(-\theta \left(\frac{x}{\ell}\right)^2 \right) \left(\frac{x}{\ell}\right)^{2(n-1)} \leq \frac{(x/\ell)^{2(n-1)}}{(n-1)!}.$$

In particular, if we have $|x| \leq R$ for μ-almost all x, we have a factorial decay $\sigma_n \leq \frac{(R/\ell)^{2(n-1)}}{(n-1)!}$.

B.4 Uniform Robustness

In applications, the RKHS $\mathcal{H}_k$ may be mis-specified and the quadrature rule μ^Q when computed for the mis-specified function class $\mathcal{H}_k$ but applied to a function $f \notin \mathcal{H}_k$ leads only to the attainable bound

$$\begin{aligned}
&\left| \int_{\mathcal{X}} f(x) \, \mathrm{d}\mu^Q(x) - \int_{\mathcal{X}} f(x) \, \mathrm{d}\mu(x) \right| \\
&\leq \sup_{x \in \mathcal{X}} \left| f(x) - \widetilde{f}(x) \right| (|\mu^Q|_{\mathrm{TV}} + |\mu|_{\mathrm{TV}}) + \left\| \widetilde{f} \right\|_{\mathcal{H}_k} \operatorname{wce}(\mu^Q; \mathcal{H}_k, \mu)
\end{aligned}$$

via triangle equality and standard integral estimates. Note that $|\cdot|_{\mathrm{TV}}$ denotes the total variation norm and the above applies to any $\widetilde{f} \in \mathcal{H}_k$; in particular, to the best approximation in uniform norm to f in $\mathcal{H}_k$. Since μ is a probability measure, $|\mu|_{\mathrm{TV}} = 1$ but if μ^Q is a signed measure with non-convex weights, its total variation $|\mu^Q|_{\mathrm{TV}}$ can be large, resulting in arbitrary large integration errors.

B.5 Equivalence between the projection/matrix Nyström approximations

Let k be a positive semi-definite kernel on $\mathcal{X}$, $Z = (z_i)_{i=1}^{\ell} \subset \mathcal{X}$. Let $P_Z : \mathcal{H}_k \to \mathcal{H}_k$ be the projection operator onto $\mathrm{span}\{k(\cdot, z_i) \mid i = 1, \dots, \ell\}$. For arbitrary $x, y \in \mathcal{X}$, we can write

$$P_Z k(\cdot, x) = \sum_{i=1}^{\ell} a_i k(\cdot, z_i), \qquad P_Z k(\cdot, y) = \sum_{i=1}^{\ell} b_i k(\cdot, z_i),$$

where $a = (a_i)_{i=1}^{\ell}, b = (b_i)_{i=1}^{\ell} \in \mathbb{R}^{\ell}$. From the properties of projection, we have

$$k(z_j, x) = \langle k(\cdot, z_j), k(\cdot, x)\rangle_{\mathcal{H}_k} = \langle k(\cdot, z_j), P_Z k(\cdot, x)\rangle_{\mathcal{H}_k} = \sum_{i=1}^{\ell} a_i k(z_j, z_i).$$

In matrix notation, we have $k(Z, x) = k(Z, Z)a$, and $k(Z, y) = k(Z, Z)b$ from the same argument. Thus, by conbining it with the property of Moore–Penrose inverse, we have

$$\begin{aligned}
\langle P_Z k(\cdot, x), P_Z k(\cdot, y)\rangle_{\mathcal{H}_k} &= a^\top k(Z, Z) b \\
&= a^\top k(Z, Z) k(Z, Z)^+ k(Z, Z) b && \text{(Moore–Penrose)} \\
&= k(x, Z) k(Z, Z)^+ k(Z, y).
\end{aligned}$$

This shows the desired equivalence.

C Proofs

C.1 Proof of Theorem 6

Before proceeding to the proof of the theorem, we prepare a couple of assertions. The following is a well-known estimate proven by using the Cauchy–Schwarz inequality [see e.g., 46, Lemma 3.1 and its proof].

Proposition 2. *Let k be a positive semi-definite kernel on $\mathcal{X}$, and ν be a Borel probability measure with $\int_{\mathcal{X}} \sqrt{k(x, x)}\, \mathrm{d}\nu(x) < \infty$. Then, for each $f \in \mathcal{H}_k$, we have*

$$\left| \int_{\mathcal{X}} f(x)\, \mathrm{d}\nu(x) \right| \le \|f\|_{\mathcal{H}_k} \int_{\mathcal{X}} \sqrt{k(x, x)}\, \mathrm{d}\nu(x).$$

By using the proposition, we obtain the following technical lemma.

Lemma 1. *Let k and k_1 be a positive semi-definite kernels on $\mathcal{X}$ such that $k - k_1$ is also positive semi-definite. Let ν be a Borel probability measure on $\mathcal{X}$. Then, for any $n \ge 1$, $a_1, \dots, a_n \in \mathbb{R}$, $x_1, \dots, x_n \in \mathcal{X}$, if we let $f = \sum_{i=1}^n a_i k(\cdot, x_i)$ and $f_1 = \sum_{i=1}^n a_i k_1(\cdot, x_i)$, then we have*

$$\left| \int_{\mathcal{X}} f_1(x)\, \mathrm{d}\nu(x) \right| \le \|f\|_{\mathcal{H}_k} \int_{\mathcal{X}} \sqrt{k_1(x, x)}\, \mathrm{d}\nu(x).$$

Proof. From the positive semi-definiteness of $k_0 \coloneqq k - k_1$, we have

$$\begin{aligned}
\|f_1\|_{\mathcal{H}_{k_1}}^2 = \sum_{i,j=1}^n a_i a_j k_1(x_i, x_j) &\le \sum_{i,j=1}^n a_i a_j k_1(x_i, x_j) + \sum_{i,j=1}^n a_i a_j k_0(x_i, x_j) \\
&= \sum_{i,j=1}^n a_i a_j k(x_i, x_j) = \|f\|_{\mathcal{H}_k}^2.
\end{aligned}$$

Hence it suffices to prove $|\nu(f_1)| \le \|f_1\|_{\mathcal{H}_{k_1}} \nu(g)$ for $g(x) \coloneqq \sqrt{k_1(x, x)}$, but it directly follows from Proposition 2. □

Proof of Theorem 6. Note first that, for each $f \in \mathcal{H}_{k_0}$, f is integrable with respect to μ. Indeed, we have

$$|f(x)| = |\langle f, k_0(\cdot, x)\rangle_{\mathcal{H}_{k_0}}| \le \|f\|_{\mathcal{H}_{k_0}} \|k_0(\cdot, x)\|_{\mathcal{H}_{k_0}} = \|f\|_{\mathcal{H}_{k_0}} \sqrt{k_0(x, x)} \le \|f\|_{\mathcal{H}_{k_0}} \sqrt{k(x, x)},$$

and it is integrable from assumption, so the equality $Q_n(f) = \mu(f)$ with $f = k_0(\cdot, x)$ is attained at a finite value.

Once we establish (15), the item (b) is clear, and (a) follows from the fact that $Q_n(g)$ and $\mu(g)$ are both integrals of the function g with respect to a probability measure. Also, (c) is justified as follows:

$$\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2 \le (Q_n(g) + \mu(g))^2 \le 2Q_n(g)^2 + 2\mu(g)^2 \le 2Q_n(g^2) + 2\mu(g^2) \le 4\mu(g^2),$$

where $\mu(Q_n)^2 \le \mu(Q_n^2)$ and $\mu(g)^2 \le \mu(g^2)$ follows from the Cauchy–Schwarz.

To prove (15),we first prove

$$|Q_n(f) - \mu(f)| \le \|f\|_{\mathcal{H}_k}(Q_n(g) + \mu(g)) \tag{18}$$

for any f of the form $f = \sum_{i=1}^n a_i k(\cdot, x_i)$ with $n \ge 0$ and $a_1, \dots, a_n \in \mathbb{R}$. Given such an f, we have $Q_n(f_0) = \mu(f_0)$ for $f_0 \coloneqq \sum_{i=1}^n a_i k_0(\cdot, x_i)$ from the assumption. Thus, by letting $f_1 \coloneqq f - f_0 = \sum_{i=1}^n a_i k_1(\cdot, x_i)$, we have

$$Q_n(f) - \mu(f) = (Q_n(f) - \mu(f)) - (Q_n(f_0) - \mu(f_0)) = Q_n(f_1) - \mu(f_1).$$

As we have $|\nu(f_1)| \le \|f\|_{\mathcal{H}_k}\nu(g)$ for $\nu = Q_n, \mu$ from Lemma 1, we obtain $|Q_n(f_1) - \mu(f_1)| \le \|f\|_{\mathcal{H}_k}(Q_n(g) + \mu(g))$, and so (18) is shown for f of the form $f = \sum_{i=1}^n a_i k(\cdot, x_i)$.

Finally, we generalize (18) to any $f \in \mathcal{H}_k$. Let $\widetilde{f} \in \mathcal{H}_k$ can be written in the form $\sum_{i=1}^n a_i k(\cdot, x_i)$. If we let $h(x) = \sqrt{k(x,x)}$, from Proposition 2, we have

$$|Q_n(f - \widetilde{f})| \le \|f - \widetilde{f}\|_{\mathcal{H}_k} Q_n(h), \qquad |\mu(f - \widetilde{f})| \le \|f - \widetilde{f}\|_{\mathcal{H}_k}\mu(h).$$

Note that $\mu(h) < \infty$ follows from the integrability of $k(x,x)$ in Assumption A. Therefore, we have

$$\begin{aligned}
|Q_n(f) - \mu(f)| &\le |Q_n(\widetilde{f}) - \mu(\widetilde{f})| + |Q_n(f - \widetilde{f}) - \mu(f - \widetilde{f})| \\
&\le \|\widetilde{f}\|_{\mathcal{H}_k}(Q_n(g) + \mu(g)) + \|f - \widetilde{f}\|_{\mathcal{H}_k}(Q_n(h) + \mu(h)) \\
&\le \|f\|_{\mathcal{H}_k}(Q_n(g) + \mu(g)) + \|f - \widetilde{f}\|_{\mathcal{H}_k}(Q_n(g) + \mu(g) + Q_n(h) + \mu(h)).
\end{aligned}$$

As we can make $\|f - \widetilde{f}\|_{\mathcal{H}_k}$ arbitrarily small from the definition of $\mathcal{H}_k$, the proof of (15) is completed by taking the limit. □

C.2 Proof of Theorem 7

Proof. Denote $D_N = \{y_1, \dots, y_N\}$ and note that the result follows from (16) and

$$\mathbb{E}\big[\widetilde{\mu}_N(g)^2\big] = \mathbb{E}\big[\widetilde{\mu}_N(g^2)\big] \le \mathbb{E}\left[\frac{1}{N}\sum_{i=1}^N g(y_i)^2\right] = \int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x), \tag{19}$$

where the first inequality is given by the Cauchy–Schwarz.

Indeed, (a) is an immediate consequence of (16) and Q_n and $\widetilde{\mu}_N$ making a probability measure, and (b) is obtained as $2\mathbb{E}\big[(Q_n(g) + \widetilde{\mu}_N(g))^2\big] \le 8\mathbb{E}\big[\widetilde{\mu}_N(g)^2\big] \le 8\int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x)$ by using (19) and the requirement $Q_n(g) \le \widetilde{\mu}_N(g)$.

When the requirement is $Q_n(g^2) \le \widetilde{\mu}_N(g^2)$, as we have $Q_n(g)^2 \le Q_n(g^2)$ and $\widetilde{\mu}_N(g)^2 \le \widetilde{\mu}_N(g^2)$ by the Cauchy–Schwarz, we also have by the AM–GM,

$$\begin{aligned}
2\mathbb{E}\big[(Q_n(g) + \widetilde{\mu}_N(g))^2\big] &\le 4\mathbb{E}\big[Q_n(g)^2\big] + 4\mathbb{E}\big[\widetilde{\mu}_N(g)^2\big] \\
&\le 4\mathbb{E}\big[Q_n(g^2)\big] + 4\mathbb{E}\big[\widetilde{\mu}_N(g^2)\big] \\
&\le 8\mathbb{E}\big[\widetilde{\mu}_N(g^2)\big] \le 8\int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x)
\end{aligned}$$

For showing (16), we remark that we always have

$$\mathrm{wce}(Q_n; \mathcal{H}_k, \widetilde{\mu}_N) \le Q_n(g) + \widetilde{\mu}_N(g) \tag{20}$$

by applying Theorem 6 with $\widetilde{\mu}_N$ instead of μ.

Let $h(\mu), h(\widetilde{\mu}_N), h(Q_n) \in \mathcal{H}_k$ be the kernel mean embeddings of μ, $\widetilde{\mu}_N$ and μ^{Q_n}, i.e.,

$$h(\mu) \coloneqq \int_{\mathcal{X}} k(\cdot, x)\,\mathrm{d}\mu(x), \quad h(\widetilde{\mu}_N) \coloneqq \frac{1}{N}\sum_{i=1}^{N} k(\cdot, y_i), \quad h(Q_n) \coloneqq \sum_{i=1}^{n} w_i k(\cdot, x_i),$$

where $(w_i)_{i=1}^n$ and $(x_i)_{i=1}^n$ are weights and points defining the quadrature Q_n. Remark that $h(\mu)$ is well-defined as $\int_{\mathcal{X}} k(x,x)\,\mathrm{d}\mu(x) < \infty$ [46, Lemma 3.1]. As we can rewrite the worst-case error as

$$\mathrm{wce}(Q_n; \mathcal{H}_k, \widetilde{\mu}_N) = \|h(Q_n) - h(\widetilde{\mu}_N)\|_{\mathcal{H}_k}, \qquad \mathrm{wce}(Q_n; \mathcal{H}_k, \mu) = \|h(Q_n) - h(\mu)\|_{\mathcal{H}_k},$$

by triangle inequality and the AM–GM, we obtain

$$\begin{aligned}
\mathbb{E}\big[\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2\big] &\le \mathbb{E}\big[(\mathrm{wce}(Q_n; \mathcal{H}_k, \widetilde{\mu}_N) + \|h(\mu) - h(\widetilde{\mu}_N)\|_{\mathcal{H}_k})^2\big] \\
&\le 2\mathbb{E}\big[\mathrm{wce}(Q_n; \mathcal{H}_k, \widetilde{\mu}_N)^2\big] + 2\mathbb{E}\big[\|h(\mu) - h(\widetilde{\mu}_N)\|_{\mathcal{H}_k}^2\big] \\
&\le 2\mathbb{E}\big[(Q_n(g) + \widetilde{\mu}_N(g))^2\big] + 2\mathbb{E}\big[\|h(\mu) - h(\widetilde{\mu}_N)\|_{\mathcal{H}_k}^2\big],
\end{aligned}$$

where we have used (20) in the last inequality. It now suffices to prove $\mathbb{E}\big[\|h(\mu) - h(\widetilde{\mu}_N)\|_{\mathcal{H}_k}^2\big] = c_{k,\mu}/N$ for showing (16).

Indeed, we have

$$\begin{aligned}
&\mathbb{E}\big[\|h(\mu) - h(\widetilde{\mu}_N)\|_{\mathcal{H}_k}^2\big] = \mathbb{E}\big[\|h(\mu)\|_{\mathcal{H}_k}^2\big] - 2\mathbb{E}\big[\langle h(\mu), h(\widetilde{\mu}_N)\rangle_{\mathcal{H}_k}\big] + \mathbb{E}\big[\|h(\mu)\|_{\mathcal{H}_k}^2\big] \\
&= \iint_{\mathcal{X}\times\mathcal{X}} k(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y) - \frac{2}{N}\sum_{i=1}^{N}\int_{\mathcal{X}} \mathbb{E}[k(x, y_i)]\,\mathrm{d}\mu(x) + \frac{1}{N^2}\sum_{i,j=1}^{N}\mathbb{E}[k(y_i, y_j)] \\
&= \frac{1}{N^2}\sum_{i=1}^{N}\mathbb{E}[k(y_i, y_i)] + \left(1 - 2 + \frac{N(N-1)}{N^2}\right)\iint_{\mathcal{X}\times\mathcal{X}} k(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y) = \frac{c_k}{N},
\end{aligned}$$

since $\iint_{\mathcal{X}\times\mathcal{X}} k(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y) = \int_{\mathcal{X}} \mathbb{E}[k(x, y_i)]\,\mathrm{d}\mu(x) = \mathbb{E}[k(y_i, y_j)]$ holds for $i \neq j$. Thus, the proof is completed. □

C.3 Proof of Proposition 1

Proof. As Q_n exactly integrates the functions in $\mathcal{H}_{k_0}$, we have $\mathrm{wce}(Q_n; \mathcal{H}_{k_0}, \mu) = 0$. So, if we set $Q_n(f) = \sum_{i=1}^n w_i f(x_i)$, then we have, from (14) with kernel k_0,

$$0 = \sum_{i,j=1}^{n} w_i w_j k_0(x_i, x_j) - 2\sum_{i=1}^{n} w_i \int_{\mathcal{X}} k_0(x_i, y)\,\mathrm{d}\mu(y) + \iint_{\mathcal{X}\times\mathcal{X}} k_0(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y). \quad (21)$$

If we extract this from the formula (14) for the kernel k, we have, by letting $k_1 \coloneqq k - k_0$,

$$\begin{aligned}
\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2 &= \mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2 - \mathrm{wce}(Q_n; \mathcal{H}_{k_0}, \mu)^2 \\
&= \sum_{i,j=1}^{n} w_i w_j k_1(x_i, x_j) - 2\sum_{i=1}^{n} w_i \int_{\mathcal{X}} k_1(x_i, y)\,\mathrm{d}\mu(y) \\
&\quad + \iint_{\mathcal{X}\times\mathcal{X}} k_1(x,y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y).
\end{aligned}$$

So, if we define $M \coloneqq \sup_{x\in\mathcal{X}}|k_1(x,y)| = \sup_{x\in\mathcal{X}}|k(x,y) - k_0(x,y)|$, we have

$$\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2 \le \left(\sum_{i,j=1}^{n} w_i w_j M + 2\sum_{i=1}^{n} w_i \int_{\mathcal{X}} M\,\mathrm{d}\mu(y) + \iint_{\mathcal{X}\times\mathcal{X}} M\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y)\right) = 4M,$$

as Q_n is a convex quadrature. The existence follows from almost the same proof as in the proof of Theorem 6, but in this case it directly follows from Tchakaloff's thorem [60, 5]. □

C.4 Proof of Theorem 8

Proof. We prove the existence of the version $Q_n(g) \le \mu(g)$. The other follows just by replacing every g in the proof below by g^2.

Let $\varphi_1, \dots, \varphi_{n-1} \in \mathcal{H}_{k_0}$ satisfy $\mathcal{H}_{k_0} = \mathrm{span}\{\varphi_1, \dots, \varphi_{n-1}\}$. Also, let $y, y_1, y_2, \dots$ be independent samples from μ. Now, consider the vector-valued function $\boldsymbol{\psi} = (\varphi_1, \dots, \varphi_{n-1}, g)^\top \in \mathbb{R}^n$. Note that $\mathbb{E}[\|\boldsymbol{\psi}(y)\|] < \infty$ follows from the integrability of elements in $\mathcal{H}_{k_0}$ and g with respect to μ. Therefore, by [22, Theorem 11], with probability 1, there exists an N such that $\mathbb{E}[\boldsymbol{\psi}(y)] \in \mathrm{conv}\{\boldsymbol{\psi}(y_1), \dots, \boldsymbol{\psi}(y_N)\}$. So, in particular, there exist deterministic points $x_1, \dots, x_N \in \mathcal{X}$ satisfying $\mathbb{E}[\boldsymbol{\psi}(y)] \in \mathrm{conv}\{\boldsymbol{\psi}(x_1), \dots, \boldsymbol{\psi}(x_N)\}$. For such $(x_i)_{i=1}^N$, consider an optimal solution that is also a *basic* feasible solution of the following linear programming problem:

$$\begin{aligned} &\text{minimize} && \sum_{i=1}^N w_i g(x_i) \\ &\text{subject to} && [\boldsymbol{\varphi}(x_1) \cdots \boldsymbol{\varphi}(x_N)]\, \boldsymbol{w} = \int_{\mathcal{X}} \boldsymbol{\varphi}(x)\, \mathrm{d}\mu(x),\ \boldsymbol{w} \ge \mathbf{0}, \end{aligned} \tag{22}$$

where $\boldsymbol{\varphi} = (1, \varphi_1, \dots, \varphi_{n-1})^\top \in \mathbb{R}^n$ is another vector-valued function (note that its first coordinate is constant so that any feasible solution of (22) sums up to one). Such a basic solution $\boldsymbol{w}$ has at most n nonzero entries, say $(w_{i_1}, \dots, w_{i_n}) \in \Delta^n$ with $1 \le i_1 < \cdots < i_n \le N$. Then, the quadrature Q_n given by weights $(w_{i_j})_{j=1}^n$ and points $(x_{i_j})_{j=1}^n$ satisfies $Q_n(\boldsymbol{\varphi}) = \mu(\boldsymbol{\varphi})$ and $Q_n(g) \le \mu(g)$. The latter follows from the optimality of $\boldsymbol{w}$ and the fact that $\mathbb{E}[\psi(y)] \in \mathrm{conv}\{\psi(y_1), \dots, \psi(y_N)\}$ leads to a feasible solution with the objective $\mathbb{E}[g(y)] = \mu(g)$). □

C.5 Proof of Theorem 3

We prove the theorem by using an existing bound regarding the Nystöm approximation for matrices, which is more common in the machine learning literature.

Let $A = (A_{ij})_{i,j=1}^N \in \mathbb{R}^{N \times N}$ be a symmetric positive semi-definite matrix. Let us denote it as $A = [\boldsymbol{a}_1, \dots, \boldsymbol{a}_N]$ by using $\boldsymbol{a}_1, \dots, \boldsymbol{a}_N \in \mathbb{R}^N$. Then, we independently sample $i_1, \dots, i_\ell$ from $\{1, \dots, N\}$ uniformly, and construct a submatrix $B = (a_{i_j i_k})_{j,k=1}^\ell$. If we let B_s be the best rank-s approximation of B and B_s^+ its pseudoinverse, the matrix

$$\widetilde{A} = [\boldsymbol{a}_{i_1}, \dots, \boldsymbol{a}_{i_\ell}] B_s^+ \begin{bmatrix} \boldsymbol{a}_{i_1} \\ \vdots \\ \boldsymbol{a}_{i_\ell} \end{bmatrix} \tag{23}$$

works as a rank-s approximation of A.

We use the following result on this matrix version:

Proposition 3 ([34, Theorem 2])**.** *For a positive semi-definite matrix A, the rank-s approximation $\widetilde{A}$ given above satisfies, with probability at least $1 - \delta$, the following:*

$$\|A - \widetilde{A}\|_2 \le \|A - A_s\|_2 + \frac{2N}{\sqrt{\ell}} A_{\max} \left(1 + \sqrt{\frac{D_{\max}^A}{A_{\max}} \frac{N - \ell}{N - 1/2} \frac{1}{\beta(\ell, N)} \log \frac{1}{\delta}} \right),$$

where $\beta(\ell, N) = 1 - \frac{1}{2 \max\{\ell, N - \ell\}}$, $A_{\max} = \max_i A_{ii}$, $D_{\max}^A = \max_{i,j}(A_{ii} + A_{jj} - 2A_{ij})$ and A_s is the best rank-s approximation of A.

As $D_{\max}^A \le 2A_{\max}$, if we have $N \ge 2\ell$, it holds that

$$\frac{D_{\max}^A}{A_{\max}} \frac{N - \ell}{N - 1/2} \frac{1}{\beta(\ell, N)} \le 2 \frac{N - \ell}{N - 1/2} \frac{N - \ell - 1/2}{N - \ell} \le 2,$$

and we can just state

$$\|A - \widetilde{A}\|_2 \le \|A - A_s\|_2 + \frac{2N}{\sqrt{\ell}} A_{\max} \left(1 + \sqrt{2 \log \frac{1}{\delta}} \right). \tag{24}$$

We show the following lemma as a consequence of this proposition.

Lemma 2. *Let $s \le \ell$ be positive integers and $\delta > 0$. Let $k : \mathcal{X} \times \mathcal{X}$ be a symmetric and positive semi-definite kernel and $y_1, y_2, \ldots$ be i.i.d. random variables taking values in $\mathcal{X}$. For each N, define the $N \times N$ matrices $K(N), K_s(N), K_s^Z(N)$ by*

$$K(N)_{ij} = \frac{k(y_i, y_j)}{N}, \quad K_s(N)_{ij} = \frac{1}{N}\sum_{m=1}^{s} \sigma_m e_m(y_i) e_m(y_j), \quad K_s^Z(N)_{ij} = \frac{k_s^Z(y_i, y_j)}{N},$$

where $Z = (y_1, \ldots, y_\ell)$.

Then, there exists a sequence $\varepsilon_N \to 0$ such that

$$\|K(N) - K_s^Z(N)\|_2 \le \|K(N) - K_s(N)\|_2 + \frac{2\sup_x k(x,x)}{\sqrt{\ell}}\left(1 + \sqrt{2\log\frac{1}{\delta}}\right) \tag{25}$$

is met with probability at least $1 - \delta - \varepsilon_N$.

Proof. We assume $N \ge 2\ell$. Let $i_1, \ldots, i_\ell$ be independent uniform samples from $\{1, \ldots, N\}$. Consider the event E_N that $i_1, \ldots, i_\ell$ are all different. Then, $\mathbb{P}(E_N) = \prod_{i=1}^{\ell} \frac{N+1-i}{N}$ converges to 1 as $N \to \infty$, and let $\varepsilon_N = 1 - \mathbb{P}(E_N)$. By using Proposition 3, (25) and $\max_i K(N)_{ii} \le N^{-1}\sup_x k(x,x)$, we have that the probability

$$\mathbb{P}\left(\|K(N) - \widetilde{K}_s(N)\|_2 \le \|K(N) - K_s(N)\|_2 + \frac{2\sup_x k(x,x)}{\sqrt{\ell}}\left(1 + \sqrt{2\log\frac{1}{\delta}}\right) \,\middle|\, E_N\right)$$

is at least $(1 - \delta - \varepsilon_N)/\mathbb{P}(E_N) \ge 1 - \delta - \varepsilon_N$, where $\widetilde{K}_s(N)$ is the rank-s Nyström approximation of the matrix $K(N)$ by using indices $i_1, \ldots, i_\ell$. From (23), if we take $\widetilde{W} = k(y_{i_j}, y_{i_k})_{j,k=1}^{\ell}$ and $\widetilde{W}_s$ its best rank-s approximation, it actually satisfies

$$\widetilde{K}_s(N)_{ij} = \frac{1}{N} k(y_i, D)\widetilde{W}_s^+ k(D, y_j) = \frac{1}{N} k_s^D(y_i, y_j),$$

where $D = (y_{i_1}, \ldots, y_{i_\ell})$ and k_s^D is the Nyström approximation given in the main body.

As $y_1, \ldots, y_N$ are i.i.d. samples, we can see that $(Z, (y_i)_{i=1}^N)$ (without any conditioning) and $(D, (y_i)_{i=1}^N)$ conditioned on E_N actually have the same distribution, so we are done. □

We finally prove the result for the Nyström approximation of integral operators.

Proof of Theorem 3. Take a sufficiently large N and let us use $K(N), K_s(N), K_s^Z(N)$ defined in the previous lemma with $y_1, y_2, \ldots$ independently sampled from μ.

It suffices to consider the case $C_k \coloneqq \sup_{x \in \mathcal{X}} k(x,x) < \infty$. It is clear that $K_s(N)_{ii} \le K(N)_{ii} \le C_k/N$, and from (10), we also have

$$k_s^Z(x,x) = k(x,Z)W_s^+ k(Z,x) \le k(x,Z)W^+ k(Z,x) = \|P_Z k(\cdot,x)\|_{\mathcal{H}_k}^2 \le \|k(\cdot x)\|_{\mathcal{H}_k}^2 = k(x,x),$$

and so $K_s^Z(N)_{ii} \le C_k/N$.

For a matrix $A(N) \in \mathbb{R}^{N \times N}$ defined by $A(N)_{ij} = (1 - \delta_{ij})(K(N) - K_s^Z(N))$, i.e., the matrix given by deleting the diagonal, we have $\|A(N)\|_2 \to \|\mathcal{K}_s^Z - \mathcal{K}\|$ as $N \to \infty$ almost surely [33, Theorem 3.1]. Since we have observed that $\|K(N) - K_s^Z(N) - A(N)\|_2 \le C_k/N$, we have

$$\|K(N) - K_s^Z(N)\|_2 \to \|\mathcal{K}_s^Z - \mathcal{K}\|, \qquad N \to \infty$$

almost surely. The same argument yields $\|K(N) - K_s(N)\|_2 \to \sigma_{s+1}$, as it converges to the norm of the integral operator given by the kernel $\sum_{m \ge s+1} \sigma_m e_m(x) e_m(y)$.

Now, by letting A_N be the event that (25) holds (so $\mathbb{P}(A_N) \ge 1 - \delta - \varepsilon_N$), the desired inequality (12) almost surely holds under the event $\limsup A_N = \bigcap_{N > \ell} \bigcup_{M \ge N} A_M$. Indeed, under this event we can just take the limit of both sides of (25) for an appropriate subsequence of $(2\ell, 2\ell + 1, \ldots)$. As we have

$$\mathbb{P}(\limsup A_N) = \lim_{N \to \infty} \mathbb{P}\left(\bigcup_{M \ge N} A_N\right) \ge \lim_{N \to \infty}(1 - \delta - \varepsilon_N) = 1 - \delta,$$

the proof is completed. □

D Kernel Quadrature when Expectations are Known

When we use an approximate kernel k_0 and know exact expectation of test functions $\varphi_1, \dots, \varphi_{n-1}$ with $\mathcal{H}_{k_0} \subset \mathrm{span}\{\varphi_1, \dots, \varphi_{n-1}\}$, we can obtain an n-point kernel quadrature that exactly integrates $\varphi_1, \dots, \varphi_{n-1}$ by Algorithm 2.

Algorithm 2 Kernel Quadrature with Random Convex Hulls

Input: A positive semi-definite kernel k on $\mathcal{X}$, a probability measure μ on $\mathcal{X}$, integers $N \geq n \geq 1$, another kernel k_0 and functions $\varphi_1, \dots, \varphi_{n-1}$ on $\mathcal{X}$ with $\mathcal{H}_{k_0} \subset \mathrm{span}\{\varphi_1, \dots, \varphi_{n-1}\}$

Output: With some probability, returns $Q_n := \{(w_i, x_i) \mid i = 1, \dots, n\} \subset \mathbb{R} \times \mathcal{X}$ with $(w_i) \in \Delta^n$

1: Calculate the expectations $\int_{\mathcal{X}} \varphi_1(x) \,\mathrm{d}\mu(x), \dots, \int_{\mathcal{X}} \varphi_{n-1}(x) \,\mathrm{d}\mu(x)$

2: Sample $y_1, \dots, y_N$ independently from μ

3: For a vector-valued function $\boldsymbol{\varphi} = (\varphi_1, \dots, \varphi_{n-1})^\top$ and $k_{1,\mathrm{diag}}(x) = k(x,x) - k_0(x,x)$, solve the linear programming problem ($|\cdot|_0$ denotes the number of nonzero entries)

$$\begin{aligned} \text{minimize} \quad & \boldsymbol{w}^\top k_{1,\mathrm{diag}}(\boldsymbol{x}) \\ \text{subject to} \quad & [\boldsymbol{\varphi}(y_1) \cdots \boldsymbol{\varphi}(y_N)]\, \boldsymbol{w} = \int_{\mathcal{X}} \boldsymbol{\varphi}(x) \,\mathrm{d}\mu(x), \\ & \boldsymbol{w} \geq \boldsymbol{0},\ \boldsymbol{1}^\top \boldsymbol{w} = 1,\ |\boldsymbol{w}|_0 \leq n. \end{aligned} \tag{26}$$

to obtain points $\{x_1, \dots, x_n\} \subset \{y_1, \dots, y_N\}$ and weights $(w_i) \in \Delta^n$ satisfying

$$\sum_{i=1}^{n} w_i \boldsymbol{\varphi}(x_i) = \int_{\mathcal{X}} \boldsymbol{\varphi}(x) \,\mathrm{d}\mu(x)$$

if (26) is feasible.

We make several remarks on this algorithm. First, the problem (26) is, strictly speaking, not a linear programming (LP), as it includes the sparsity constraint $|\boldsymbol{w}|_0 \leq n$. However, as it only contains n equality constraints, its *basic feasible solution* always satisfies $|\boldsymbol{w}|_0 \leq n$ and the simplex algorithm automatically gives such a sparse (and optimal) solution even if we do not explicitly impose this constraint, so we call it an LP for simplicity. Second, *this algorithm occasionally fails to output Q_n* as, with some probability, the LP has no feasible solution. Although we can repeat the algorithm until we succeed, the number N should be chosen appropriately. See Remark 5 for this point. Finally, our algorithm has possibly related approaches such as sparse optimization and Sard's method, see Remark 6 and 7.

Remark 5. *A simple approach for constructing a quadrature formula [22] was recently proposed: randomly sample candidate points and find a solution by using a linear programming (LP) solver. Indeed, for an independent sample $y_1, \dots, y_N \sim \mu$, we can construct a quadrature formula with convex weights exactly integrating the functions in $\mathcal{F} = \mathrm{span}\{\varphi_1, \dots, \varphi_{n-1}\}$ using a subset of these points if and only if we have*

$$\int_{\mathcal{X}} \boldsymbol{\varphi}(x) \,\mathrm{d}\mu(x) \in \mathrm{conv}\{\boldsymbol{\varphi}(y_1), \dots, \boldsymbol{\varphi}(y_N)\}, \tag{27}$$

where $\boldsymbol{\varphi} = (\varphi_1, \dots, \varphi_{n-1})^\top : \mathcal{X} \to \mathbb{R}^{n-1}$ and $\mathrm{conv}\, A$ denotes the convex hull of A. Several sharp estimates for the probability of the event (27) *are available in Hayakawa et al. [23]. Under the event* (27)*, we can find a desired rule by using the simplex method for the LP problem* (26)*.*

Remark 6. *From the viewpoint of subsampling, a direct way to obtain quadrature formulas with convex weights supported on a small number of points, is to first sample N candidate points $D_N = (x_1, \dots, x_N)$ and then solve the following sparse optimization problem:*

$$\begin{aligned} \textit{minimize} \quad & \boldsymbol{w}^\top k(D_N, D_N) \boldsymbol{w} - 2\boldsymbol{w}^\top \textstyle\int_{\mathcal{X}} k(D_N, y) \,\mathrm{d}\mu(y) \\ \textit{subject to} \quad & \boldsymbol{w} \geq \boldsymbol{0},\ \boldsymbol{1}^\top \boldsymbol{w} = 1,\ |\boldsymbol{w}|_0 \leq n, \end{aligned} \tag{28}$$

where $k(D_N, D_N)$ is the corresponding $N \times N$ Gram matrix. Unfortunately, exactly solving this problem is computationally challenging, in particular in contrast to our approach that exploits the spectral properties of k and μ. Nevertheless, one could use sparse optimization to obtain

an approximate solution of (28)*: although the simplex constraint ($\boldsymbol{w} \geq \mathbf{0}$, $\mathbf{1}^\top \boldsymbol{w} = 1$) makes it impossible to exploit the classical ℓ_1 regulatization, there are possible alternatives under this constraint [52, 35, 37] or use the DC (difference of convex functions) algorithm to incorporate the sparsity constraint to find a local minima [18]. This is a promising research direction, and our general sample estimates might provide a first step towards this direction.*

Remark 7. *Sard's method [54, 36] for constructing numerical integration rules uses the n degree of freedom (of choosing weights in our setting) separately; m ($\leq n$) for exactness over a certain m-dimensional space of test functions, and the remaining $n - m$ for minimizing an error criterion such as the worst-case error. In the context of kernel quadrature, one way to use Sard's method with exactness over $\mathcal{F}$ (an m-dimensional space of test functions) is as follows [29, 57]:*

$$\begin{array}{rl} \textit{minimize} & \boldsymbol{w}^\top k(D_n, \boldsymbol{x}_n)\boldsymbol{w} - 2\boldsymbol{w}^\top \int_{\mathcal{X}} k(\boldsymbol{x}_n, y)\,\mathrm{d}\mu(y) \\ \textit{subject to} & \boldsymbol{w}^\top f(D_n) = \int_{\mathcal{X}} f(y)\,\mathrm{d}\mu(y),\ \forall f \in \mathcal{F}, \end{array} \tag{29}$$

where $D_n = (x_1, \ldots, x_n)^\top$, $f(D_n) = (f(x_1), \ldots, f(x_n))^\top$. This amounts to solving a convex quadratic programming for $\boldsymbol{w}$ in an $(n-m)$-dimensional subspace of $\mathbb{R}^n$ (without constraint). This is similar to our approach in that it enforces exactness in a certain finite-dimensional space of test functions. One key difference is that Sard's approach aims for a quadrature formula on a given set of points, whereas our method determines also the points themelves. Hence, the combination of these two approaches seems to be an interesting future research topic.[5]

Computational complexity. A tricky part of this approach, essentially based on random convex hulls, is that the algorithm possibly does not output a quadrature formula. Hence, the following quantity plays an important role to estimate the essential complexity of the algorithm:

$$N_\varphi = \inf\left\{ N \geq 1 \,\middle|\, \mathbb{P}(\mathbb{E}[\boldsymbol{\varphi}(y)] \in \operatorname{conv}\{\boldsymbol{\varphi}(y_1), \ldots, \boldsymbol{\varphi}(y_N)\}) \geq \frac{1}{2} \right\},$$

where $y, y_1, y_2, \ldots$ are independent samples from μ. This value is known to be finite and estimated under a variety of conditions on $\boldsymbol{\varphi}(y)$ [65, 23]. If we have some knowledge of μ, we can just keep trying the algorithm with $N = N_\varphi$ until it succeeds, and its expected computational time is $\mathcal{O}(nN_\varphi + C(n, N_\varphi))$, where $C(a, b)$ is the (expected) cost of solving an $a \times b$ LP with a simplex method. Note that, though the worst-case computational time of the simplex method is exponential, it is empirically $\mathcal{O}(ab \min\{a, b\})$ in practice [51, 55]. In addition, $N_\varphi = \mathcal{O}(n)$ holds in examples with some symmetry [67, 22], so in that case we have a heuristic complexity estimate of $\mathcal{O}\big(n^3\big)$.

Choice of approximate kernels. Similarly to the empirical version discussed in the main text, we prove quantitative estimates when k_0 is given by the Mercer approximation or Nyström approximation. Remark that the necessary information for using these methods is different. Whereas using the Mercer approximation requires the knowledge of Mercer decomposition $k(x, y) = \sum_{m=1}^\infty \sigma_m e_m(x) e_m(y)$ and their exact integration $\int_{\mathcal{X}} e_m(x)\,\mathrm{d}\mu(x)$, the Nyström approximation only requires the exact integral values of kernel, $\int_{\mathcal{X}} k(x, y)\,\mathrm{d}\mu(y)$, and so is more generally applicable. See the following sections for details.

In the following, we assume that the kernel attains the Mercer decomposition $k(x, y) = \sum_{m=1}^\infty \sigma_m e_m(x) e_m(y)$, where $\sigma_1 \geq \sigma_2 \geq \cdots \geq 0$ and $(e_m)_{m=1}^\infty$ is an orthonormal set of $L^2(\mu)$.

D.1 Algorithm 2 with Mercer Approximation

If we use the truncated Mercer decomposition as an approximate kernel, we have the following result.

Theorem 9. *If Algorithm 2 with $k_0 \coloneqq \sum_{m=1}^{n-1} \sigma_m e_m(x) e_m(y)$ and $\varphi_i = e_i$ successfully outputs a convex quadrature Q_n, then it satisfies the following:*

(a) If $C \coloneqq \sup_{m \geq 1} \|e_m\|_\infty < \infty$, we have $\operatorname{wce}(Q_n; \mathcal{H}_k, \mu)^2 \leq 4C^2 \sum_{m=n}^\infty \sigma_m$.

[5] For example, we can pick the first m eigenfunctions of the integral operator as test functions, and find n points and weights that minimizes the worst-case error while exactly integrating the test functions from a larger set of candidate points. An obvious challenge is that a quadratic programming does not supply sparsity, whereas the approach of this paper has been fully based on the sparsity of a basic feasible solution of an LP problem.

(b) As N in Algorithm 2 tends to infinity, we have

$$\mathbb{P}\left(\mathrm{wce}(Q_n;\mathcal{H}_k,\mu)^2 \le 4\sum_{m=n}^{\infty}\sigma_m\right) \to 1.$$

Proof. As in the proof of Corollary 2, $k-k_0$ is positive semi-definite. Thus, when $\sum_{m=n}^{\infty}\sigma_m<\infty$, the kernel and the measure μ satisfies Assumption A. So Thorem 6(a) implies (a) of this theorem, since $k_1(x,x)=\sum_{m=n}^{\infty}\sigma_m e_m(x)^2 \le C^2\sum_{m=n}^{\infty}\sigma_m$.

For (b), if we have $Q_n(k_{1,\mathrm{diag}}) \le \mu(k_{1,\mathrm{diag}})$, then Theorem 6 implies

$$\mathrm{wce}(Q_n;\mathcal{H}_k,\mu)^2 \le 4\int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x) = \int_{\mathcal{X}}\sum_{m=n}^{\infty}\sigma_m e_m(x)^2\,\mathrm{d}\mu(x) = 4\sum_{m=n}^{\infty}\sigma_m.$$

So it suffices to prove $\mathbb{P}(Q_n(k_{1,\mathrm{diag}}) \le \mu(k_{1,\mathrm{diag}})) \to 1$ as $N\to\infty$, and it is shown by considering the optimal basic feasible solution of the LP (26) and the following fact [23, Proposition 4]:

$$\mathbb{P}(\mathbb{E}[\boldsymbol{\psi}(y)] \in \mathrm{conv}\{\boldsymbol{\psi}(y_1),\dots,\psi(y_N)\}) \to 1, \qquad N\to\infty,$$

where $\psi=(\varphi_1,\dots,\varphi_{n-1},k_{1,\mathrm{diag}})^\top$. Indeed, under the event $\boldsymbol{\psi}(y)\in\mathrm{conv}\{\boldsymbol{\psi}(y_1),\dots,\psi(y_N)\}$, the LP becomes feasible and $Q_n(k_{1,\mathrm{diag}}) \le \mu(k_{1,\mathrm{diag}})$ follows from the optimality. See the proof of Theorem 8 (Section C.4) for a more detailed explanation if necessary. □

Note that the boundedness of C is a typical assumption [see 41, Assumption 3.2 and references therein], while it does not necessarily hold [45, Section 3]. Under some assumptions, we can quantify the probability that the LP (26) becomes feasible.

Sampling bound. Suppose 1 is an eigenfunction of $\mathcal{K}$. This is satisfied, e.g., in the following cases:

- μ is a Haar measure on a compact group and k is shift-invariant.
- k is a kernel based on Stein's identity [48, 57, 2] with respect to μ.

In this case, we have a theoretical bound of the required N in Algorithm 2 as follows.

Theorem 10. *Suppose 1 is an eigenfunction of $\mathcal{K}$, i.e., $\int_{\mathcal{X}} k(\cdot,y)\,\mathrm{d}\mu(y)$ is a constant function. Then, for each $n\ge 2$ and $N \ge 6(n-1)\sup_{x\in\mathcal{X}}\sum_{m=1}^{n-1} e_m(x)^2$, Algorithm 2 returns a feasible quadrature with probability at least $1-2^{1-n}$, i.e., for an independent sample $y_1,\dots,y_N$ from μ, we have*

$$\mathbb{P}\left(\int_{\mathcal{X}}\boldsymbol{\varphi}(x)\,\mathrm{d}\mu(x) \in \mathrm{conv}\{\boldsymbol{\varphi}(y_1),\dots,\boldsymbol{\varphi}(y_N)\}\right) \ge 1-\frac{1}{2^{n-1}},$$

where $\boldsymbol{\varphi}=(e_1,\dots,e_{n-1})^\top$. If the value $C=\sup_{m\ge1}\|e_m\|_\infty$ is finite, $N\ge 6C(n-1)^2$ is also sufficient for the above estimate.

Proof. This follows from the existing results [23, Theorem 14 and Proposition 17]. □

D.2 Algorithm 2 with Nyström Approximation

Although the method discussed in the previous section requires the knowledge of Mercer decomposition, if we make use of the Nyström approximation, we only require the values of $\int_{\mathcal{X}} k(x,y)\,\mathrm{d}\mu(y)$ for $x\in\mathcal{X}$.

Recall that $k_s^Z(x,y)$ is the rank-s Nyström approximation of the kernel k based on the point set $Z=(z_1,\dots,z_\ell)$. From (11), we can use $\varphi_i^Z := u_i^\top k(Z,\cdot)$ as test functions.

Theorem 11. *Let $n\le\ell$ and $\delta>0$, and let Z be an ℓ-point independent sample from μ. If Algorithm 2 with $k_0=k_{n-1}^Z$ and $\varphi_i=\varphi_i^Z$ successfully outputs a convex quadrature Q_n, then with probability at least $1-\delta$, we have*

$$\mathrm{wce}(Q_n;\mathcal{H}_k,\mu)^2 \le 4n\sigma_n + 4\sum_{m=n+1}^{\infty}\sigma_m + \frac{8(n-1)\sup_{x\in\mathcal{X}}k(x,x)}{\sqrt{\ell}}\left(1+\sqrt{2\log\frac{1}{\delta}}\right)$$

Proof. As in the proof of Corollary 4, $k - k^Z_{n-1}$ is positive semi-definite. Also, we can assume $\sum_{m=1}^\infty \sigma_m < \infty$, as otherwise the right-hand side is infinity. Thus k and $k_0 = k^Z_{n-1}$ satisfy Assumption A.

Note that for a function of the form $c(x,y) = a \cdot b(x)b(y)$ with $a \in \mathbb{R}$ and $b \in L^2(\mu)$, and an orthonomal set $(f_i)_{i\in I} \subset L^2(\mu)$ of $L^2(\mu)$ with $b \in \overline{\operatorname{span}\{f_i \mid i \in I\}}$, we have

$$\begin{aligned}\sum_{i\in I} \iint_{\mathcal{X}\times\mathcal{X}} f_i(x)c(x,y)f_i(y)\,\mathrm{d}\mu(x)\,\mathrm{d}\mu(y) &= a\sum_{i\in I}\langle b, f_i\rangle_{L^2}^2 \\ &= a\|b\|_{L^2}^2 = \int_{\mathcal{X}} c(x,x)\,\mathrm{d}\mu(x). \end{aligned} \tag{30}$$

If we here use the orthonormal set $(e_m)_{m=1}^\infty$ that appears in the Mercer decomposition, by letting $k_1 \coloneqq k - k^Z_{n-1}$ and using the linear extension of (30), we have

$$\begin{aligned}\int_{\mathcal{X}} k_1(x,x)\,\mathrm{d}\mu(x) &= \int_{\mathcal{X}} (k(x,x) - k_0(x,x))\,\mathrm{d}\mu(x) \\ &= \sum_{m=1}^\infty \left\langle e_m, (\mathcal{K} - \mathcal{K}^Z_{n-1})e_m\right\rangle_{L^2} \\ &\le \sum_{m=n}^\infty \langle e_m, \mathcal{K}e_m\rangle_{L^2} + \sum_{m=1}^{n-1}\|\mathcal{K} - \mathcal{K}^Z_{n-1}\|\|e_m\|_{L^2}^2 \\ &= \sum_{m=n}^\infty \sigma_m + (n-1)\|\mathcal{K} - \mathcal{K}^Z_{n-1}\|. \end{aligned} \tag{31}$$

Then, by combining this with Theorem 3 and Theorem 6(c), we obtain the desired estimate. □

Remark 8. *If we denote by N_φ the required number of samples, the computational complexity of the above algorithm becomes $\mathcal{O}\big(n\ell N_\varphi + n\ell^2 + C(n, N_\varphi)\big)$, including the cost of computing the Nyström approximation as well as test functions at N_φ samples (see also Remark 2).*

E Additional Numerical Experiments

In this section, we provide additional experiments on Algorithm 2 using random convex hulls, as well as the approximated version of the **N. + emp** described in Remark 2. Section E.1 shows the comparison of Algorithm 2 (with Mercer/Nyström approximation) with some of the methods mentioned in the main text under the periodic Sobolev spaces with uniform measure. Section E.2 investigates Algorithm 2 (with Nystöm approximation) as well as the approximate but fast algorithm for **N. + emp**, under the setting of empirical measure reduction.

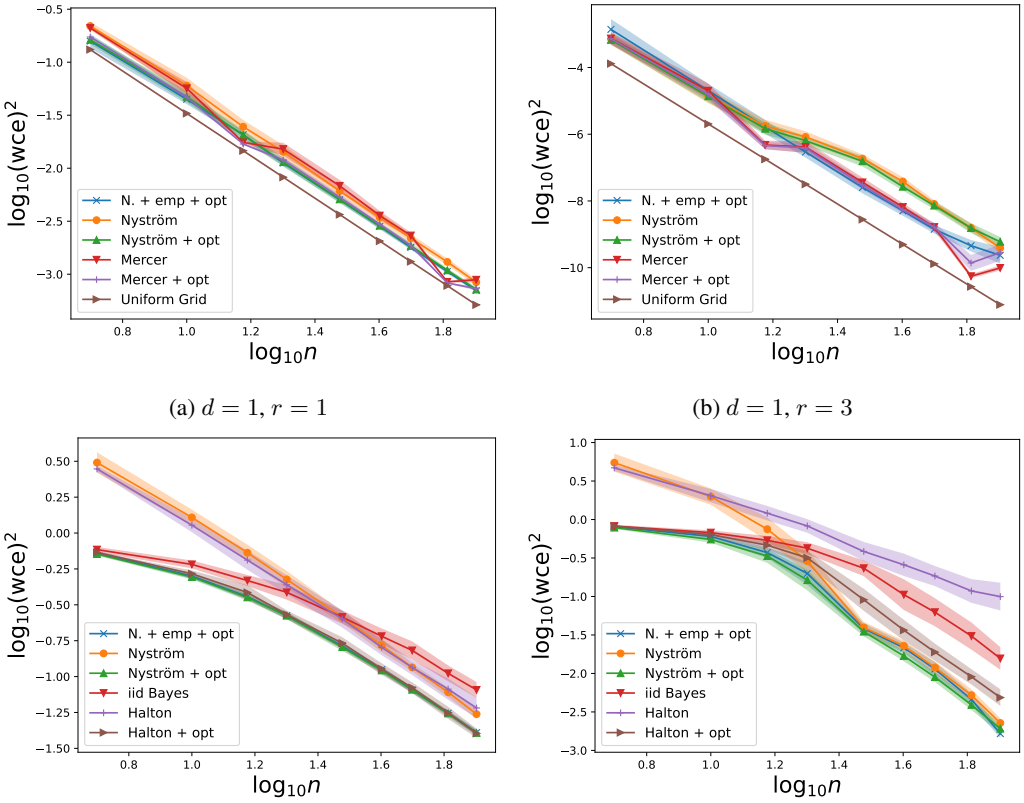
E.1 Periodic Sobolev Spaces with Uniform Measure

We conducted experiments under the same setting as in Section 3.1, except that we additionally have the following methods:

Nyström, Nyström + opt: We used the same test functions as **N. + emp** with the random set Z of size $\ell = 10n$, but for Algorithm 2. We used $N = 10n$ samples for the LP (26). In **Nyström + opt** we additionally optimized the convex weights using (14).

Mercer, Mercer + opt ($d = 1$): We used the same test functions as **M. + emp**, but for Algorithm 2. We used $N = 10n$ samples for the LP (26). In **Mercer + opt** we additionally optimized the convex weights using (14).

The results are given in Figure 3. The weights of **Nyström** and **Mercer** are already almost optimized as they exactly integrate a certain family of functions, so the additional CQP (14) does not change the error so much. Surprising is that **N. + emp + opt** is almost as good as **Nyström + opt** or even better. This implies that the recombination points with respect to a moderately large ($N = n^2$ in this case) empirical measure can provide a good convergence rate in Bayesian quadrature [25], even though the (equally weighted) empirical measure itself is not that close to the true measure.



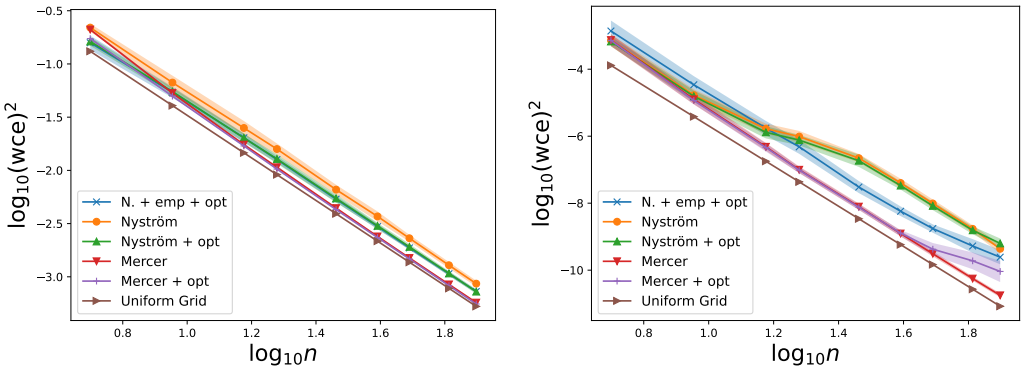
(a) $d = 1, r = 1$

(b) $d = 1, r = 3$

(c) $d = 2, r = 1$

(d) $d = 3, r = 3$

Figure 3: Periodic Sobolev spaces with kernel $k_r^{\otimes d}$: The average of $\log_{10}(\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2)$ over 50 trials is plotted for each method of obtaining Q_n. The shaded regions are sample standard deviation. The worst computational time per one trial was 5 seconds of **N. + emp** and **N. + emp + opt** in $(d, r, n) = (3, 3, 80)$, while **Nyström** and **Nyström + opt** ran in 0.9 seconds under the same setting. There were no infeasible LPs.



(a) $d = 1, r = 1$

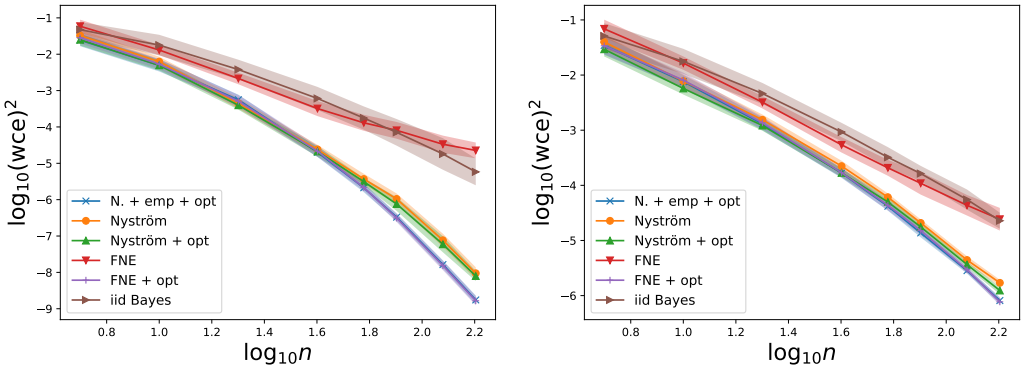
(b) $d = 1, r = 3$

Figure 4: Supplemental experiments for Figure 3. n is all odd.

Odd behavior of 'Mercer'. As we can see in Figure 3(a,b), the methods based on the exact Mercer decomposition becomes very close to optimal when $n = 15, 65$. As it seemed to be caused by the parity of n, we carried out another experiment for $n \in \{5, 9, 15, 19, 29, 39, 49, 65, 79\}$ (Figure 4), then **Mercer** and its optimization clearly became the best methods except the exact optimal **Unifrom Grid**. It might be related to the structure of the periodic Sobolev space, that has, for each eigenvalue except for 1, two-dimensional eigenspace ($\cos$ and $\sin$), but needs further investigation. Also, in the case $(d, r) = (1, 3)$, we see '**+ opt**' make the quadrature less accurate for a big n, but it is theoretically almost impossible, so it seems to be caused by numerical accuracy of the CQP solver.

E.2 Measure Reduction in Machine Learning Datasets

We conducted experiments under the same setting as in Section 3.2. We additionally adopted **Nyström, Nyström + opt** (with $N = 20n$), and **FNE, FNE + opt**, where **FNE** (stands for 'fast N. + emp') is the approximate algorithm for **N. + emp** by omitting the inequality in (6) and using the randomized SVD [20] (see Remark 2).



(a) 3D Road Network data

(b) Power Plant data

Figure 5: Measure reduction in Gaussian RKHS with two ML datasets: The average of $\log_{10}(\mathrm{wce}(Q_n; \mathcal{H}_k, \mu)^2)$ over 50 trials is plotted for each method of obtaining Q_n. The shaded regions are sample standard deviation. The worst computational time per one trial was 13 seconds of **N. + emp** and **N. + emp + opt** in 3D Road Network data with $n = 160$. There were 7 infeasible LPs (and 800 feasible LPs) in the experiment (a) with **Nyström** or **Nyström + opt**. There were no infeasible LPs in (b).

The results are given in Figure 5. **N. + emp + opt** and **FNE + opt** show almost the same convergence. While in the largest case $n = 160$, the average runtime of (**N. + emp + opt**, **FNE + opt**) was $(13.0, 2.07)$ seconds in 3D Road Network data and $(12.8, 2.09)$ seconds in Power Plant data, respectively. Although our theoretical guarantee no longer holds for **FNE**, it accelerates the algorithm while surprisingly maintaining the accuracy. **Nystöm** or **Nyström + opt** behave much better than **iid Bayes**, but are slightly less accurate than **N. + emp + opt** and **FNE + opt**, whereas they have good theoretical guarantees (Theorem 11). Their computational time was basically between that of **FNE + opt** and **N. + emp + opt**.

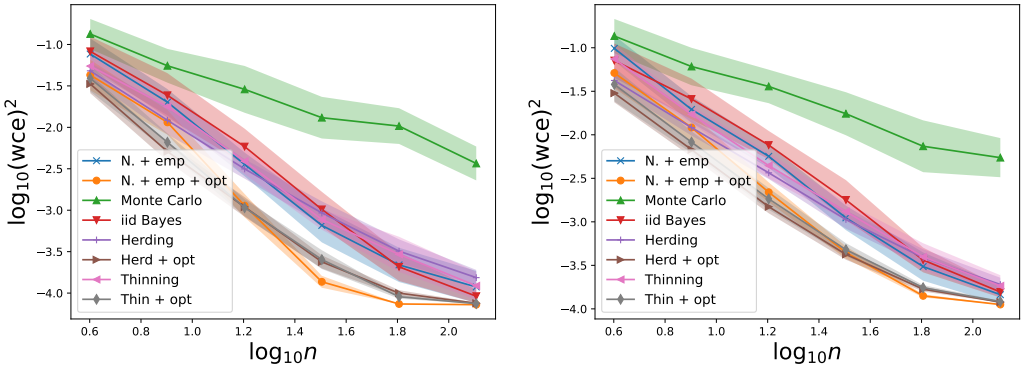
Comparison with another empirical measure. The setting of 'ML datasets' treated here is empirical measures given by some real data, so it is also just an approximation of a true distribution from the viewpoint of frequentists. Therefore, if we want to evaluate the performance of measure reduction methods with regard to the true distribution, we should measure the worst-case error using it. As it is not feasible in reality, we take another empirical measure μ' (of the same size as but different from the empirical measure μ, used in the construction of a kernel quadrature rule Q_n), and plot the quantities of $\mathrm{wce}(Q_n; \mathcal{H}_k, \mu')$ to better estimate the actual performance of Q_n in this section.

The overall setting is the same as in Section 3.2, except the following points:

- In the 3D Road Network Data Set, we used another random 43487-point subset from the remaining $434874 - 43487$ data points to define μ'.

- In the Combined Cycle Power Plant Data Set, we used exactly a half of the whole data points to define μ (so the size of $\operatorname{supp}\mu$ is different from the original experiment) and the other half to define μ'.

Note that μ and μ' were randomly taken at first and fixed throughout the experiment. The median heuristics as well as the normalization of the data (for both of the points in μ and μ') was carried out by using the statistical information solely given by μ.



(a) 3D Road Network data

(b) Power Plant data

Figure 6: Measure reduction in Gaussian RKHS with two ML datasets with another empirical measure: The average of $\log_{10}(\mathrm{wce}(Q_n; \mathcal{H}_k, \mu')^2)$ over 20 trials is plotted for each method of obtaining Q_n. The shaded regions show their standard deviation. The worst computational time per one trial was 14 seconds of **Thinning [+ opt]** in Power Plant data with $n = 128$, where **N. + emp [+ opt]** was 6.2 [6.1] seconds.

The results are given in Figure 6. We can see that, though our methods are still competitive, the error eventually becomes dominated by the (MMD-)distance between μ and μ' as n gets larger. This is inevitable as we are only using the empirical measure μ to construct Q_n, so in an application to this kind of setting, we can just pick any method whose error is sufficiently small compared to the 'inevitable' error.