# Optimal algorithmic complexity of inference in quantum kernel methods

**Elies Gil-Fuster**[*,1,2], **Seongwook Shin**[1,3], **Sofiene Jerbi**[1,4], **Jens Eisert**[1,2,4], **Maximilian J. Kramer**[†,1]
[1] Freie Universität Berlin
[2] Fraunhofer Heinrich Hertz Institute Berlin
[3] Sejong University
[4] Helmholtz-Zentrum Berlin
[*] emgilfuster@gmail.com
[†] m.kramer@fu-berlin.de

## Abstract

Quantum kernel methods are among the leading candidates for achieving quantum advantage in supervised learning. A key bottleneck is the cost of inference: evaluating a weighted sum of $N$ kernel values to additive precision $\varepsilon$, where $\alpha$ is the vector of trained coefficients. The standard approach estimates each term independently via sampling, yielding a query complexity of $\mathcal{O}(N\|\alpha\|_2^2/\varepsilon^2)$. In this work, we combine two independent improvements: estimating the kernel values via quantum amplitude estimation and collecting the sum under a single observable. We show that the improved approach achieves a query complexity of $\mathcal{O}(\|\alpha\|_1/\varepsilon)$, removing the dependence on $N$ from the query count and yielding a quadratic improvement in both $\|\alpha\|_1$ and $\varepsilon$. We prove a matching lower bound of $\Omega(\|\alpha\|_1/\varepsilon)$, establishing query-optimality in our oracle access model. Beyond query complexity, we also analyze how these improvements translate into gate costs and show that the query-optimal strategy is not always optimal in practice from the perspective of gate complexity. We thus provide both a query-optimal algorithm and practical guiding principles for the choice of strategy depending on hardware capabilities, along with a complete landscape of intermediate methods to guide practitioners. All algorithms require only amplitude estimation as a subroutine and are thus natural candidates for early-fault-tolerant implementations.

## 1 Introduction

Quantum kernel methods [Schuld, 2021, Tanner et al., 2026] are among the leading candidates for achieving quantum advantage in supervised learning [Biamonte et al., 2017, Acampora et al., 2025], as showcased by the provable learning separation of Liu et al. [2021]. The mainstream approach is based on *embedding quantum kernels* [Hubregtsen et al., 2022, Gil-Fuster et al., 2024b], where a kernel function $k$ is specified via an $n$-qubit *quantum feature map* $x \mapsto \rho(x)$ as $k(x, x') = \mathrm{Tr}\{\rho(x)\rho(x')\}$. After training – finding suitable parameters $\alpha \in \mathbb{R}^N$ from a training set of size $N$ – one must evaluate the resulting hypothesis on new data. This *inference* step amounts to computing $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$, and its cost has been identified as a critical scalability bottleneck [Nakayama et al., 2024]. Commonly used techniques [Havlíček et al., 2019, Tanner et al., 2026] estimate each of the $N$ terms independently via sampling, resulting in a query complexity of $\mathcal{O}(N\|\alpha\|_2^2/\varepsilon^2)$. This scaling can be improved along two independent axes: In the way individual kernel values are estimated (*sampling* versus *quantum amplitude estimation*), and how the sum over training points is approximated (*term-by-term* versus *via a single observable*). We systematically investigate these choices in Section 3 and Appendix C. Ultimately, we show that in the early fault-tolerant regime [Eisert and Preskill, 2025], the optimal combination – encoding the full sum as a single observable and applying quantum amplitude

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

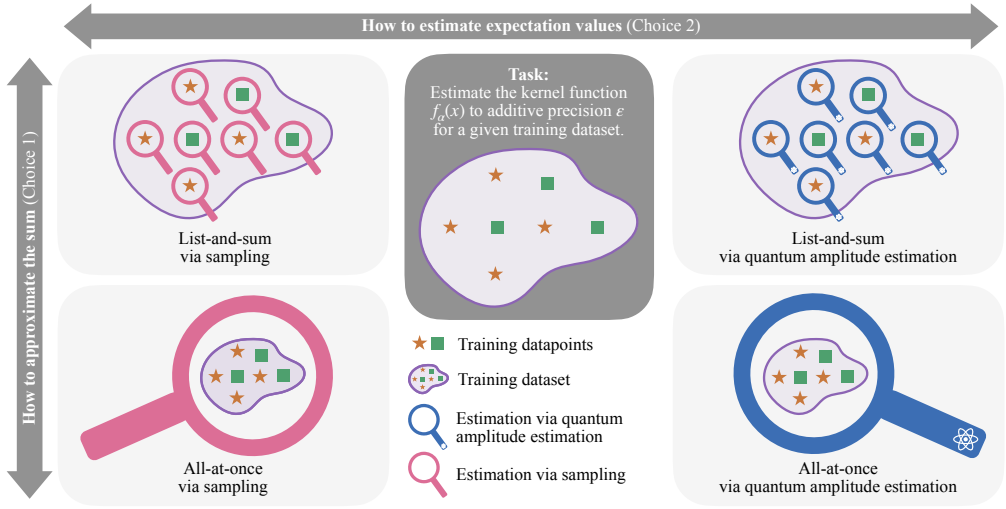



Figure 1: Schematic overview of the four algorithmic strategies for inference in quantum kernel methods. The two axes correspond to the two independent design choices: how the sum over training points is approximated (*list-and-sum* versus all-at-once, vertical axis), and how expectation values are estimated (sampling versus quantum amplitude estimation, horizontal axis). In list-and-sum, each kernel value $k(x, x_i)$ is estimated independently and results are combined classically. In the *all-at-once* approach, the full inference sum is encoded as a single observable. Moving from sampling to quantum amplitude estimation yields a quadratic improvement in precision within each row.

estimation – achieves a query complexity of $\mathcal{O}(\|\alpha\|_1/\varepsilon)$, where $\|\alpha\|_1 = \sum_{i=1}^N |\alpha_i|$ (Theorem 4). We complement this with a matching lower bound of $\Omega(\|\alpha\|_1/\varepsilon)$, up to constant factors (Theorem D.1). Beyond query complexity, our results provide a practical guide to quantum kernel inference: given hardware capabilities, Algorithm C.1 and Theorem 4 identify the appropriate strategy and its runtime, while Theorem 5 characterizes the corresponding gate complexity. Among the strategies analyzed here, *list-and-sum* with adaptive budget via quantum amplitude estimation has the smallest leading polynomial gate-count upper bound in the stated regime. Throughout, we use *gate-optimal* in this comparative sense, suppressing polylogarithmic factors.

## 2 Preliminaries

### 2.1 Problem set-up and notation

Suppose we are given a learning task defined on $\mathcal{X} \times \mathcal{Y}$, where a quantum kernel model has already been trained. We call the parameters of the kernel model $\alpha \in \mathbb{R}^N$, where $N$ is the size of the training set. We consider quantum feature maps realized via a data-dependent unitary $U(x)$ applied to the all-0 state vector as $|\phi(x)\rangle = U(x)|0\rangle$, with $\rho(x) = |\phi(x)\rangle\langle\phi(x)|$. In this case, the kernel function becomes $k(x, x') = |\langle\phi(x)|\phi(x')\rangle|^2$ for $x, x' \in \mathcal{X}$, and the inference task reduces to evaluating

$$f_\alpha(x) := \sum_{i=1}^N \alpha_i k(x, x_i) = \sum_{i=1}^N \alpha_i |\langle 0|U^\dagger(x)U(x_i)|0\rangle|^2. \tag{1}$$

Here, $S = \{(x_i, y_i)\}_{i=1}^N$ is a training set of size $N$, where the co-variates $x_i \in \mathcal{X}$ are $d$-dimensional real vectors, and the labels $y_i \in \{\pm 1\}$ (in the case of binary classification) or real-valued (in the case of regression). For simplicity, we restrict our discussion to binary classification and regression; standard approaches beyond these have been discussed by, e.g., Schölkopf and Smola [2002]. The vector $\alpha = (\alpha_1, \ldots, \alpha_N) \in \mathbb{R}^N$ contains the trained parameters, with each $\alpha_i$ being a weight assigned to a training point $x_i$.

We refer to $N$ as the *size of the training set* for simplicity. In practice, sparsity is often imposed over the coefficients during training (e.g., via $\ell_1$-regularization), so that $N$ only includes the non-zero coefficients, sometimes dubbed *support vectors*. While number of support vectors can be significantly

smaller than the size of the training set, we do not make any specific assumptions. We summarize the above in the following formal definition for the computational task of inference.

**Definition 1** (Problem set-up)**.** *Let $S = \{(x_i, y_i)\}_{i=1}^N$ be a training set of size $N \in \mathbb{N}$, with $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$, let $\alpha = (\alpha_i)_{i=1}^N \in \mathbb{R}^N$ be a vector of trained parameters, let $U(x)$ be an input-dependent unitary quantum-feature-map involving $G$-many gates from a fixed gate set, with $U(x)|0\rangle = |\phi(x)\rangle$, and let $\varepsilon > 0$ be a precision parameter. The goal is to provide a quantum algorithm which, for any new input $x$, evaluates the kernel function $f_\alpha(x)$ in Eq. (1) to additive precision $\varepsilon$ with probability at least $2/3$.*

**Definition 2** (Norms)**.** *We use the notation $\|x\|_p := \left(\sum_{i=1}^N |x_i|^p\right)^{1/p}$ for $p \in \{2/3, 1, 2\}$, although it is not a norm for $0 < p < 1$ as it does not fulfill the triangle inequality. For a Hermitian operator $\mathcal{M}$ with eigenvalues $\{\lambda_i\}_i$, we use $\|\mathcal{M}\|_\infty = \max_i |\lambda_i|$ and refer to it as the operator norm.*

**Definition 3** (Short-hand notation)**.** *We separate the coefficient vector into three parts: (1) its norm $\|\alpha\|_1$, (2) a probability distribution $p = (p_i)_{i=1}^N$ with $p_i = |\alpha_i|/\|\alpha\|_1$, and (3) a vector of signs $s = (s_i)_{i=1}^N$, with $s_i = \mathrm{sign}(\alpha_i)$. With this, we can re-write Eq. (1) as*

$$f_\alpha(x) = \|\alpha\|_1 \sum_{i=1}^N p_i s_i k(x, x_i) = \|\alpha\|_1 \mathbb{E}_{i\sim p}[s_i k(x, x_i)]. \tag{2}$$

*Accounting for the positive and negative terms in the sum, we introduce the intermediate variables $f_\pm \in [0, 1]$ as $f_\pm := \sum_{i=1}^N \delta_{s_i,\pm 1} p_i k(x, x_i)$. We can re-write Eq. (1) in terms of these normalized quantities as $f_\alpha(x) = \|\alpha\|_1 (f_+ - f_-)$.*

Below, we define the oracle access model used throughout the work and specify the associated gate costs per query.

**Definition 4** (Access model with associated gate costs)**.** *To tackle the goal from Definition 1, we assume the following quantum oracle access model:*

- Single-input oracle $U(x)$: *given any input $x$ and an $n$-qubit register, we can implement $U(x)$, $U^\dagger(x)$, and controlled versions thereof, with an associate cost of $G$-many gates.*
- Coefficient quantum-state oracle $W(\alpha)$: *Given a $(\lceil \log_2(N) \rceil + 1)$-qubit register, we can implement $W(\alpha)$, which applied onto $|0\rangle$ prepares a quantum state encoding the coefficients as amplitudes $|\alpha\rangle = \sum_{i=1}^N \sqrt{p_i}|i\rangle \otimes |s_i\rangle$, with $p_i$ and $s_i$ as in Definition 3, and $s_i$ encoding the sign of the $i^{th}$ coefficient as an eigenstate of the $Z$ operator as $Z|s_i\rangle = s_i|s_i\rangle$. We can also implement the controlled version of $W(\alpha)$, as well as $W^\dagger(\alpha)$. The associated cost per query in terms of gates is $N$.*

*We consider the following classical oracles to access the coefficient vector $\alpha$:* Reading oracle *$O_R(\alpha, i)$, read the $i^{th}$ element $\alpha_i$;* Norm oracle *$O_N(\alpha)$, compute the norms $\|\alpha\|_1$, $\|\alpha\|_2$, or the pseudo-norm $\|\alpha\|_{2/3}$; and* Sampling oracle *$O_C(\alpha)$, sample an index $i$ with probability $p_i$ (that is, proportional to the magnitude of the coefficients).*

In Section 3.2, we measure algorithmic cost by the total number of *quantum* oracle queries, which include the single-input oracle, the coefficient quantum-state oracle, and the training-set oracle (which we introduce below, in Definition 6). Below, we provide explicit circuit constructions that confirm that the runtime of our algorithms is indeed dominated by the quantum oracle queries. We note that the gate cost per query varies across algorithms, yet in a way that is mostly independent of $\varepsilon$ and $\|\alpha\|_1$, and hence it does not affect the asymptotic comparisons in runtime and query complexity we perform in this work. We comment on resource estimates in terms of gates in Section 3.3.

The access model we propose in Definition 4 not only allows us to perform a formal analysis of query complexity, but it also accurately reflects realistic implementations. The oracles we introduce can be directly instantiated on hardware, and they do not require any special computational power, nor do they hide exponential overheads. Still, we assume *black-box* access to $U(x)$, $O^\dagger(S)$, and $W(\alpha)$ to remove the possibility of optimized implementations which would exploit the structure of the circuits involved. This is a conscious design choice, as it is our goal to focus on the fundamental complexity of the computational task, which in turn provides worst-case guarantees applicable to any specific quantum feature map. As is common in query complexity studies, nothing prevents more-efficient procedures from existing under the assumption that $U(x)$ is of a particular known form.

### 2.2 Amplitude estimation

The inference task in Definition 1 can be viewed as an instance of amplitude estimation, which we now define.

**Definition 5** (Amplitude estimation problem)**.** *Let $V \in \mathbb{C}^{d\times d}$ be a unitary such that*

$$V\,|0\ldots0\rangle\,|0\rangle = a\,|\psi_{\text{good}}\rangle\,|0\rangle + \sqrt{1-a^2}\,|\psi_{\text{bad}}\rangle\,|1\rangle\,, \tag{3}$$

*where $|\psi_{\text{good}}\rangle$ and $|\psi_{\text{bad}}\rangle$ are normalized quantum state vectors, and $a \in [0,1]$. Given access to $V$, $V^\dagger$, and controlled-$V$, the task is to output an estimate $\hat{a}$ such that $a-\varepsilon < \hat{a} < a+\varepsilon$ with probability at least $2/3$.*

Two approaches are relevant: *sampling*, which estimates $a$ from repeated measurements of $V|0\rangle$ with sampling complexity $\mathcal{O}(\varepsilon^{-2})$, and *quantum amplitude estimation* (QAE), which uses coherent applications of $V$ and $V^\dagger$ to achieve $\mathcal{O}(\varepsilon^{-1})$ [Brassard et al., 2002, Aaronson and Rall, 2020]. The latter scaling is optimal [Nayak and Wu, 1999] and requires inverse access [Tang and Wright, 2025, Tang et al., 2025]. We provide formal statements and further background in Appendix A.

### 2.3 Related work

The naive approach to inference, which we call *list-and-sum (fixed budget)* below, was already outlined by, e.g., Havlíček et al. [2019] and Schuld and Killoran [2019], and re-instated countless times [Schuld, 2021, Tanner et al., 2026]. To the best of our knowledge, a formal runtime analysis of this approach had not been performed before; we provide it in Theorems 4 and 5. Liu et al. [2021] established a learning separation, proving that quantum kernel methods can learn functions outside of $\mathsf{P}/\mathsf{poly}$ under standard cryptographic assumptions, making the scalability of inference practically relevant.

Beyond the black-box oracle model we consider, existing works have studied reducing inference cost via *surrogates*, either classically [Schreiber et al., 2023, Landman et al., 2022, Sweke et al., 2023, Sahebi et al., 2025], exploiting random Fourier features [Rahimi and Recht, 2007]; or quantumly [Nakayama et al., 2024], also using shadow tomography [Aaronson, 2018, Haug et al., 2023, King et al., 2025]. Surrogates can reduce inference cost in restricted scenarios, but cannot work in the general setting we consider (Definitions 1 and 4).

The trade-off between term-by-term and global estimation strategies that we exploit here has parallels in classical signal processing: in compressed sensing, coherent measurements satisfying the restricted isometry property recover $s$-sparse vectors from $\mathcal{O}(s\log N)$ measurements rather than $\mathcal{O}(N)$ term-by-term ones [Eldar and Kutyniok, 2012]. A related trade-off arises in basis optimization for electronic structure Hamiltonians in quantum chemistry [Rubin et al., 2018, Low and Chuang, 2019].

## 3 Results

We highlight two algorithmic choices to make: one regarding how the target quantity $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x,x_i)$ (or parts thereof) is estimated, and the other regarding how the sum is approximated. In particular, we consider the following possibilities:

Choice 1. **How to estimate expectation values:** either via *sampling* (see Theorem A.1), yielding precision $\mathcal{O}(1/\sqrt{M})$ from $M$ shots, or via *quantum amplitude estimation* (see Theorem A.3), yielding precision $\mathcal{O}(1/M)$ from $M$ coherent queries.

Choice 2. **How to approximate the sum:** the naive option is to *list-and-sum*, i.e., estimate each term independently and add the results classically, either with a *fixed budget* for each term or following an *adaptive budget* scheme. Alternatively, all terms can be grouped into a single *all-at-once* observable $\mathcal{M}$.

These algorithmic choices are illustrated schematically in Figure 1 to provide an intuitive overview. A priori it could be unclear whether such an *all-at-once* observable exists, or whether it can be efficiently implemented. We address this first in Section 3.1. A unified algorithm and a theorem describing the runtime complexities follows in Section 3.2. For a detailed presentation and analysis of each of the ramifications of the unified algorithm, refer to Appendices B and C. We comment on

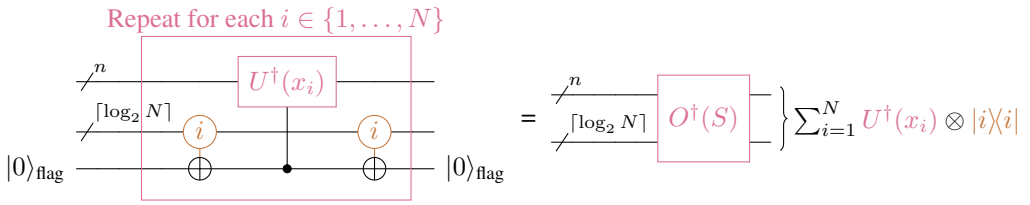



Figure 2: Realizing the "training-set oracle" $O^\dagger(S)$ in Definition 6, as $N$ controlled calls to the inverse of the single-input oracle $U(x)$, one for each input in the training set $x_i \in S$. The associated gate cost is $\mathcal{O}(N(G + \log N))$. This construction resembles ideas in the *linear combination of unitaries* (LCU) framework [Childs and Wiebe, 2012].

resource estimates in Section 3.3 and end by providing a matching query-complexity lower bound in Section 3.4.

### 3.1 Existence and implementation of an all-at-once observable

Instead of estimating each kernel term individually, we reinterpret the weighted sum as a single expectation value by embedding the coefficients into a quantum state. This allows us to evaluate the entire inference expression in one coherent procedure, rather than aggregating many separate estimates. We show that the full inference sum $f_\alpha(x)$ can be cast as the expectation value of a single observable, rather than estimating each kernel value separately. Similar versions of the following results were used by, e.g., González et al. [2022], Nakayama et al. [2024], Gil-Fuster et al. [2024a, 2025], though not as formal statements.

We start by introducing another quantum oracle that can be constructed from those in our access model.

**Definition 6** (Training-set oracle $O^\dagger(S)$)**.** *Let $S = \{(x_i, y_i)\}_{i=1}^N$ be the training set. Given two registers, an $n$-qubit "data" register and a $\lceil \log_2(N) \rceil$-qubit "index" register, which indexes each element $i \in \{1, \ldots, N\}$, we call* training-set oracle $O^\dagger(S)$

$$O^\dagger(S) := \sum_{i=1}^{N} U^\dagger(x_i)_{\text{data}} \otimes |i\rangle\langle i|_{\text{idx}} \equiv \prod_{i=1}^{N} \left( U^\dagger(x_i)_{\text{data}} \otimes |i\rangle\langle i|_{\text{idx}} + \mathbb{I}_{\text{data}} \otimes (\mathbb{I} - |i\rangle\langle i|)_{\text{idx}} \right). \quad (4)$$

We note that this construction does not require *quantum random-access memory* (QRAM). The training inputs $\{x_i\}$ are classically known and fixed after training, so the controlled unitaries can be compiled into a circuit offline. The $\mathcal{O}(N)$ gate cost per query is inherent to any procedure – classical or quantum – that must touch all $N$ training points.

**Lemma 1** (Query complexity training-set-oracle)**.** *The training-set oracle can be realized with $N$ controlled queries to the single-input oracle $U(x_i)$. The associated gate cost is $\mathcal{O}(N(G + \log N))$, and the procedure uses one extra qubit.*

*Proof.* See Fig. 2 for a visual proof. The associated gate cost is, for each $i \in \{1, \ldots, N\}$: $\mathcal{O}(\log N)$ gates to perform a multi-controlled bitflip on the flag register conditioned on the index being $i$, $\mathcal{O}(G)$ to call the controlled version of the $G$ gates in $U(x_i)$, and $\mathcal{O}(\log N)$ to uncompute the flag bitflip, for a total of $\mathcal{O}(G + \log N)$. Since these must be repeated $N$ times, the total gate cost of the training-set oracle is $\mathcal{O}(N(G + \log N))$. □

We consider the training-set oracle $O^\dagger(S)$ as another quantum oracle in the sense of Definition 4. In Appendix B.2, we consider the possibility of a *programmable* embedding unitary $U(x)$, together with a more-efficient implementation of the training-set oracle using the programmable single-input oracle. Since such a programmable oracle is an extra assumption, however fair, we do not include it in the main text.

**Theorem 2** (All-at-once observable)**.** *Consider the problem set-up from Definition 1. There exists a bounded Hermitian observable $\mathcal{M}(\alpha, S)$, with $\|\mathcal{M}(\alpha, S)\|_\infty \leq \|\alpha\|_1$, and a quantum state vector $|\phi(x)\rangle$, such that*

$$\langle \phi(x) | \mathcal{M}(\alpha, S) | \phi(x) \rangle = f_\alpha(x). \quad (5)$$

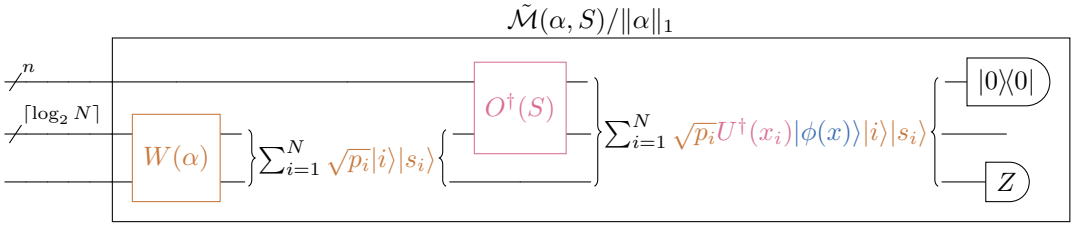



Figure 3: Quantum circuit encoding the *all-at-once* observable as a single expectation value. The *all-at-once* observable $\tilde{\mathcal{M}}(\alpha, S)$ is the collection of gates and fixed observables inside the box, up to a factor of $\|\alpha\|_1$. The intermediate-step formulas serve the purpose of illustrating the action of the gates onto the input state vector $|\phi(x)\rangle$.

*Moreover, this expectation value can be implemented using* $n + \lceil \log_2 N \rceil + 2$ *qubits and* $\mathcal{O}(N(G + \log N))$ *gates.*

*Proof sketch.* Set $\mathcal{M}(\alpha, S) = \sum_{i=1}^N \alpha_i |\phi(x_i)\rangle\langle\phi(x_i)|$ and verify directly. We give Figure 3 as visual proof, and we present full details and a rigorous argument in Appendix B. □

The implementation follows the "adjoint" method (using $U(x)$ and $U^\dagger(x)$), as opposed to the SWAP test; the main difference is a trade-off of depth versus width [Schuld, 2021, Hubregtsen et al., 2022]. To perform quantum amplitude estimation, recall from Definition 5 that we must provide a unitary gate $V$ such that $V|0\rangle = a|\psi_{\text{good}}\rangle|0\rangle + \sqrt{1-a^2}|\psi_{\text{bad}}\rangle|1\rangle$. Here, $a$ is the quantity we would like to estimate, and the precise form of $|\psi_{\text{good,bad}}\rangle$ is not important, as long as we are able to implement $V$ and $V^\dagger$ multiple times. Our general strategy is to relate $a$ to $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$. Following the short-hand notation from Definition 3, we use quantum amplitude estimation for $f_+$ and $f_-$ independently, and then we reconstruct $f_\alpha(x)$ from their difference. We provide a single unitary gate $V$ which allows us to estimate both.

**Definition 7** (Amplitude-encoding unitary)**.** *Consider the problem set-up from Definition 1 and the notation from Definition 3. We define the* amplitude-encoding unitary $V$ *via its action:*

$$V|0\rangle = \sqrt{f_+}|\psi_+\rangle|0,0\rangle + \sqrt{f_-}|\psi_-\rangle|0,1\rangle + \alpha|\psi_{trash}\rangle|1,0\rangle + \beta|\psi'_{trash}\rangle|1,1\rangle. \tag{6}$$

*We have* $\alpha^2 + \beta^2 = 1 - f_+ - f_-$.

With this, specializing $|\psi_{\text{good}}\rangle = |\psi_\pm\rangle$ and collecting the rest under $|\psi_{\text{bad}}\rangle$ for both $f_\pm$ becomes straightforward. We give a circuit depiction for how to construct $V$ in Figure 4, further details can be found in Appendix B.

**Lemma 3** (Single-observable encoded in amplitude)**.** *Under the conditions of Theorem 2, there exists a quantum circuit using* $n + \lceil \log_2(N) \rceil + 3$ *qubits which implements the amplitude-encoding unitary in Definition 7.*

*Proof sketch.* Refer to Figure 4 as visual proof, full details are in Appendix C.2.2. □

### 3.2 Unified algorithm and runtime complexity

With the *all-at-once* observable in hand, the two algorithmic choices can be combined freely, giving rise to a landscape of algorithms (further details can be found in Appendix C, in particular in the unified algorithm summarized in Algorithm C.1). For each combination, we prove the resulting runtime complexity; the results are collected in terms of query complexity in Theorem 4 and in terms of gates in Theorem 5.

Moreover, for completeness, we also analyze a *sample-and-average* strategy in Appendix E, which conceptually interpolates between *list-and-sum* and *all-at-once*. Our analysis yields the same query-complexity upper bound for sampling and quantum amplitude estimation within this template.

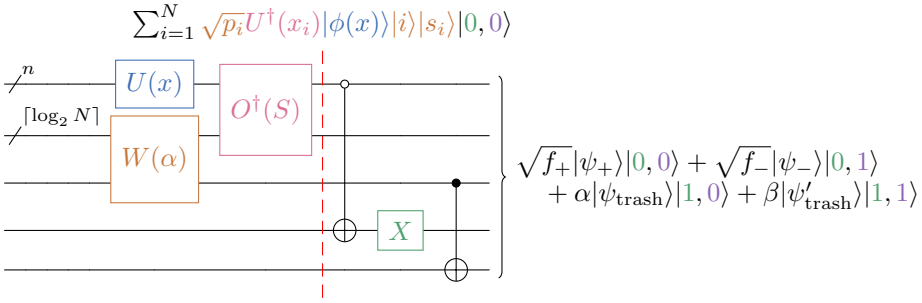



Figure 4: Quantum circuit encoding the *all-at-once* observable as amplitudes. Explicit implementation of the unitary $V$ described in Lemma 3.

As a technical comment, we note that analyzing the *list-and-sum* strategy involves manipulating individual estimators for each of the terms in the sum. Specifically for estimators based on quantum amplitude estimation, each of the individual estimators may be biased. In our results, we use the nondestructive estimator of Cornelissen and Hamoudi [2023] (see also Rall and Fuller [2023]), which is nearly unbiased, with a bias that can be suppressed at multiplicative-logarithmic cost, see Theorem C.1 and Lemma C.4 for full details. The estimator used in the *all-at-once* strategy is unaffected by this issue, since it combines two amplitude estimates through their tail bounds rather than by averaging.

The displayed list-and-sum bounds apply in the regime where the continuous total query budget is at least $N$; per-term allocations are rounded up to at least one query. For the gate-complexity bounds, we assume $n \in \mathcal{O}(G)$, so the $\mathcal{O}(n)$ reflection overhead is absorbed into $\mathcal{O}(G)$.

**Theorem 4** (Query complexity of inference in quantum kernel methods)**.** *There exist algorithms for estimating the quantum kernel model prediction in Eq.* (1) *to additive precision $\varepsilon$ with constant success probability (see Definition 1). The algorithmic runtime depends on the algorithmic choices as follows:*

| ***Strategy*** | ***Sampling*** | ***Quantum amplitude estimation*** |
|---|---|---|
| *1(a): list-and-sum, fixed* | $\mathcal{O}\left(\frac{N\|\alpha\|_2^2}{\varepsilon^2}\right)^{\ddagger}$ *[Cor. C.2]* | $\tilde{\mathcal{O}}\left(\frac{N\|\alpha\|_2}{\varepsilon}\right)$ *[Cor. C.6]* |
| *1(b): list-and-sum, adaptive* | $\mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right)$ *[Cor. C.3]* | $\tilde{\mathcal{O}}\left(\frac{\|\alpha\|_{2/3}}{\varepsilon}\right)^{\diamond}$ *[Cor. C.7]* |
| *1(c): all-at-once* | $\mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right)$ *[Thm. C.8]* | $\mathcal{O}\left(\frac{\|\alpha\|_1}{\varepsilon}\right)^{\star}$ *[Thm. C.9]* |

*We highlight ‡ as the naive approach (1(a), sampling), ⋆ as query-optimal (1(c), quantum amplitude estimation), and ⋄ as gate-optimal (1(b), quantum amplitude estimation; see Theorem 5).*

*Proof.* Each case follows from the corresponding statement in square brackets; the full proofs are given in Appendix C. □

### 3.3 Resource estimates

In terms of runtime complexity, Theorem 4 dictates that the best algorithmic choice depends on the available hardware. On near-term devices without coherent multi-query access, using an adaptive budget for *list-and-sum* already yields the same runtime scaling as measuring the more-costly *all-at-once* observable. Conversely, the *all-at-once* approach is more beneficial in the early-fault-tolerant regime, where the runtime scaling is directly manifested by the 1-norm of the vector of coefficients, versus the $2/3$-norm achieved by *list-and-sum* with an adaptive budget. Explicitly, within any case for Choice 1, quantum amplitude estimation yields a quadratic improvement over sampling. We expand beyond query-complexity analysis in Theorem 5, where we show the asymptotic scaling of the number of gates associated to each choice.

**Theorem 5** (Number of gates for inference in quantum kernel methods)**.** *There exist algorithms for estimating the quantum kernel model prediction in Eq. (1) to additive precision $\varepsilon$ with constant success probability (see Definition 1). The number of gates depends on the algorithmic choices as follows:*

| ***Strategy*** | ***Sampling*** | ***Quantum amplitude estimation*** |
|---|---|---|
| *1(a): list-and-sum, fixed* | $\mathcal{O}\left(\frac{NG\|\alpha\|_2^2}{\varepsilon^2}\right)^{\ddagger}$ | $\tilde{\mathcal{O}}\left(\frac{NG\|\alpha\|_2}{\varepsilon}\right)$ |
| *1(b): list-and-sum, adaptive* | $\mathcal{O}\left(\frac{G\|\alpha\|_1^2}{\varepsilon^2}\right)$ | $\tilde{\mathcal{O}}\left(\frac{G\|\alpha\|_{2/3}}{\varepsilon}\right)^{\diamond}$ |
| *1(c): all-at-once* | $\tilde{\mathcal{O}}\left(\frac{NG\|\alpha\|_1^2}{\varepsilon^2}\right)$ | $\tilde{\mathcal{O}}\left(\frac{(NG+n)\|\alpha\|_1}{\varepsilon}\right)^{\star}$ |

*We highlight ‡ as the naive approach (1(a), sampling), ⋆ as query-optimal (1(c), quantum amplitude estimation; see Theorem 4), and ⋄ as gate-optimal (1(b), quantum amplitude estimation).*

*Proof.* We import the query complexity runtime results from Theorem 4 and then we account for the differing gate costs of each oracle query. Our access model in Definition 4 specifies that the single-input oracle $U(x)$ requires $G$ gates, and the coefficient quantum-state oracle $W(\alpha)$ requires $N$ gates. Further, we have shown how to construct the training-set oracle $O^\dagger(S)$ using a number of gates in $\mathcal{O}(N(G+\log N))$. Since the *list-and-sum* algorithms only use the single-input oracle, once for $x$ and once for $x_i$, their gate cost is $2G$ times the number of queries. Via sampling, this results in $\mathcal{O}(\varepsilon^{-2}NG\|\alpha\|_2^2)$ gates with a fixed budget, and $\mathcal{O}(\varepsilon^{-2}G\|\alpha\|_1^2)$ with an adaptive budget. Via quantum amplitude estimation, the scaling in the number of gates is, in turn, $\tilde{\mathcal{O}}(\varepsilon^{-1}NG\|\alpha\|_2)$ with a fixed budget and $\tilde{\mathcal{O}}(\varepsilon^{-1}G\|\alpha\|_{2/3})$ with an adaptive budget.

In the *all-at-once* approaches, the circuits used for sampling and quantum amplitude estimation are not identical, as seen when comparing Figures 3 and 4. Nevertheless, both circuits make the same number of oracle queries, visually: they contain the same number and types of boxes. Namely, each measurement of $\tilde{\mathcal{M}}(\alpha, S)$ and each query to the unitary $V$ both require one call to the single-input oracle $U(x)$, one call to the quantum-state oracle $W(\alpha)$, and one call to the training-set oracle $O^\dagger(S)$. Implementing the unitary $V$ requires one multi-controlled $X$ gate, which has an associated cost of $\mathcal{O}(n)$ gates. For sampling, the gate costs per measurement add up to $G + N + \mathcal{O}(N(G+\log N))$, which scales as $\mathcal{O}(N(G+\log N))$. For quantum amplitude estimation, the extra multi-controlled gate results in a scaling of $\mathcal{O}(N(G+\log N)+n)$. The total number of gates used in the *all-at-once* algorithms is, then, $\tilde{\mathcal{O}}(\varepsilon^{-2}NG\|\alpha\|_1^2)$ via sampling and $\tilde{\mathcal{O}}(\varepsilon^{-1}(NG+n)\|\alpha\|_1)$ via quantum amplitude estimation, where we have removed the logarithmic factors for simplicity. □

**Query-optimality versus gate-optimality.** When comparing the *naive* and *query-optimal* approaches in Theorem 4, there is an $N$-factor speed-up in the query complexity, on top of the quadratic improvement over the norm of $\alpha$ and the precision $\varepsilon$. In terms of gates, as stated in Theorem 5, the same is no longer true: both *list-and-sum* with fixed budget and *all-at-once* have an $N$ factor in their gate cost scaling. From the perspective of number of gates, it is not immediately clear anymore that *all-at-once* is better than *list-and-sum* with adaptive budget. In the case of sampling, *list-and-sum* with adaptive budget now has an $N$-factor *improvement* over *all-at-once*. In the case of quantum amplitude estimation, *all-at-once* has both an $N$ and an $n$ factor. Below, we make a principled comparison between the runtime scalings. Let us define $\tilde{G}_L \coloneqq \varepsilon^{-1}G\|\alpha\|_{2/3}$ and $G_A \coloneqq \varepsilon^{-1}(NG+n)\|\alpha\|_1$ the respective scaling of the numbers of gates. The subscript $L$ stands for "list-and-sum", and the subscript $A$ stands for "all-at-once", and the notation $\tilde{G}_L$ indicates that we drop logarithmic factors.

$$\left.\begin{aligned} &\frac{G_L}{G_A} = \frac{\varepsilon^{-1}G\|\alpha\|_{2/3}}{\varepsilon^{-1}(NG+n)\|\alpha\|_1} = \frac{\|\alpha\|_{2/3}}{(N+\frac{n}{G})\|\alpha\|_1} \\ &\|\alpha\|_1 \leq \|\alpha\|_{2/3} \leq \sqrt{N}\|\alpha\|_1 \end{aligned}\right\} \Rightarrow \left(\sqrt{N} + \frac{n}{G\sqrt{N}}\right) G_L \leq G_A \leq \left(N + \frac{n}{G}\right) G_L. \quad (7)$$

Thus, after suppressing logarithmic factors, the displayed gate-count upper bound for *all-at-once* is no smaller than that for *list-and-sum* with adaptive budget. See Appendix B.2 for a refined discussion using a programmable quantum embedding $U(x)$.

**Practical guidelines.** As a guide to the eye for the practitioner, our results suggest the following guidelines for choosing an inference strategy:

- Use *all-at-once* combined with quantum amplitude estimation when coherent multi-query access is available, as this achieves query-optimal scaling.
- Use *list-and-sum with adaptive budget* combined with quantum amplitude estimation when gate cost dominates, as it has the best leading gate-count bound among the strategies analyzed here.
- Sampling-based approaches are primarily competitive in near-term regimes without coherent access, but are asymptotically suboptimal in precision.

### 3.4 Matching lower bound

The query complexity of the *all-at-once* via quantum amplitude estimation algorithm (cf. Algorithm C.9 and Theorem C.9) cannot be improved in its dependence on $\|\alpha\|_1$ and $\varepsilon$.

In Appendix D we construct a family of oracle instances, with a single training point, on which any quantum algorithm that estimates $f_\alpha(x)$ to additive precision $\varepsilon$ with constant success probability must make

$$\Omega\left(\frac{\|\alpha\|_1}{\varepsilon}\right) \tag{8}$$

queries to the single-input oracle, counting inverses, controlled versions, and the training-set oracle alike (Theorem D.1). The bound constrains every algorithm equipped with the oracles of Definition 4, not only those routing their computation through the amplitude-encoding unitary $V$: the hardness sits inside the feature-map oracle, leaving all other oracles instance-independent and hence free of charge, and the proof reduces to the two-point approximate-counting bound of Nayak and Wu [1999], with each oracle call simulated at unit cost by bit-oracle queries via phase kickback. Since each application of $V$ or $V^\dagger$ makes two such calls, any routine accessing the instance only through $V$, $V^\dagger$, and controlled versions thereof – in particular any quantum-amplitude-estimation routine run on $V$ – must apply them $\Omega(\|\alpha\|_1/\varepsilon)$ times (Corollary D.2). Together with the matching upper bound of Theorem C.9, Algorithm C.9 is therefore query-optimal in these two parameters, up to constant factors.

Some caveats delimit the claim. The bound holds for $64 \leq \|\alpha\|_1/\varepsilon \leq 2^{n-1}$, and neither end restricts the conclusion: below it the task is solved with $\mathcal{O}(1)$ queries, while above it the number of feature qubits $n$ is a free parameter of the construction, so hard instances exist at every ratio. Since the hard family has $N = 1$, where all norms of $\alpha$ coincide, we do not claim optimality with respect to a finer parameterization of $\alpha$. Quantum amplitude estimation is likewise already optimal for each individual kernel term – estimating $k(x, x_i)$ to additive error $\varepsilon_i$ requires $\Omega(\varepsilon_i^{-1})$ queries whenever $2^{1-n} \leq \varepsilon_i \leq 1/64$ – so any improvement over *list-and-sum* must change how the sum is approximated, not how its terms are estimated.

## 4 Discussion

In this work, we have scrutinized and revisited evaluation methods for inference in quantum kernel methods, which are among the top candidates to achieve quantum advantage in machine learning. Our results have practical implications for the long-term viability of quantum kernels versus other approaches in *quantum machine learning* (QML). The cost of inference in kernel methods has been cited as a critical flaw that could prevent scalability [Nakayama et al., 2024]. Our results show that the dependence on the training set size can be confined to the per-query gate cost, so that the number of oracle queries – and hence the scaling with precision – becomes independent of $N$. We show that the query-complexity scaling of the naive approach $\mathcal{O}\big(N\|\alpha\|_2^2/\varepsilon^2\big)$ can be improved down to $\mathcal{O}(\|\alpha\|_1/\varepsilon)$. By providing matching lower bounds, we showed that this improved scaling is optimal within our access model. Importantly, our lower bound concerns query complexity only; no matching bound accompanies the gate-count statements, and an inference algorithm with gate cost sublinear in $N$ is not ruled out by our bound.

In the regime of exponential kernel concentration [Thanasilp et al., 2024], where $k(x, x') \approx \delta_{x,x'}$, our upper bounds remain valid but the model itself loses predictive power. Our results are therefore most

relevant for kernels whose values do not vanish asymptotically [Liu et al., 2021, Henderson et al., 2025], which are precisely the settings where quantum kernel methods yield *quantum advantage*. Extending our construction to non-unitary feature maps (mixed states $\rho(x)$) remains an open question.

Our results confirm that using quantum amplitude estimation for the *all-at-once* observable is optimal in terms of query complexity. Taking into account hardware requirements for both noisy and early-fault-tolerant devices, it may well be that this approach is beyond budget. Accordingly, we provide a zoo of algorithms that nicely interpolate between the *naive* and *optimal* ones, and formally analyze their runtimes. As such, this work aims to constitute a *useful handbook for practitioners*. Indeed, we hope that our analysis clarifies some of the critical design choices involved in realistic implementations for the inference step in quantum kernel methods.

Finally, an exciting open question is whether our improvements for inference can be extended to speed-up the *training* phase of quantum kernels. The usual pipeline in training kernel methods involves inverting an $N \times N$ matrix, and the end-to-end complexity is usually characterized as $\mathcal{O}(N^3)$. Anticipating potential paths to reduce the cost, one could take Eq. (1) without assuming that the parameters $\alpha$ have already been trained, and instead define a loss function with respect to those parameters. Then, one option would be to perform (stochastic) gradient descent [Shalev-Shwartz and Ben-David, 2014] on those parameters. Gradient-based optimization of kernel methods has been considered also in classical machine learning, mostly as a proof technique, and it comes with performance guarantees owing to the fact that usual regularized loss functions are convex with respect to $\alpha$ [Schölkopf and Smola, 2002]. The representer theorem [Schuld, 2021, Schölkopf and Smola, 2002] further states that the optimal choice of $\alpha$ actually corresponds to the best possible linear function in terms of fitting the training set. Beyond straightforward gradient-based optimization, one could consider more advanced forms of quantum optimization [Leng et al., 2023, Catli et al., 2025], which in turn enjoy favorable performance guarantees for convex loss functions.

**Note added**

During the final stages of publication, new results by Ambainis et al. [2026] provide unbiased estimators for quantum amplitude estimation. With these, our analysis of *list-and-sum* strategies can be simplified.

## Acknowledgments and Disclosure of Funding

This work has been inspired by conversations with Tom O'Brien of Google Quantum AI. The authors thank Armando Bellante, Carlos Bravo-Prieto, Vedran Dunjko, Tom O'Brien, Louis Schatzki and Franz Schreiber for useful references and comments in an earlier version of this manuscript. E. G.-F. is a recipient of the 2023 Google PhD Fellowship. Moreover, the authors acknowledge support by the BMFTR (PraktiQOM, QuSol, HYBRID++), the Quantum Flagship (Millenion, PasQuans2), the DFG (SPP 2514, CRC 183), the Munich Quantum Valley, Berlin Quantum, the Clusters of Excellence (MATH+, ML4Q), and the European Research Council (DebuQC).

The authors declare no other competing interests.

## References

Scott Aaronson. Shadow tomography of quantum states. In *Proceedings of the 50th annual ACM SIGACT symposium on theory of computing*, pages 325–338, 2018.

Scott Aaronson and Patrick Rall. *Quantum approximate counting, simplified*, page 24–32. Society for Industrial and Applied Mathematics, 2020. ISBN 9781611976014. doi: 10.1137/1.9781611976014.5.

Giovanni Acampora, Andris Ambainis, Natalia Ares, Leonardo Banchi, Pallavi Bhardwaj, Daniele Binosi, G. Andrew D. Briggs, Tommaso Calarco, Vedran Dunjko, Jens Eisert, Olivier Ezratty, Paul Erker, Federico Fedele, Elies Gil-Fuster, Martin Gärttner, Mats Granath, Markus Heyl, Iordanis Kerenidis, Matthias Klusch, Anton Frisk Kockum, Richard Kueng, Mario Krenn, Jörg Lässig, Antonio Macaluso, Sabrina Maniscalco, Florian Marquardt, Kristel Michielsen, Gorka Muñoz-Gil, Daniel Müssig, Hendrik Poulsen Nautrup, Sophie A. Neubauer, Evert van Nieuwenburg, Roman Orus, Jörg Schmiedmayer, Markus Schmitt, Philipp Slusallek, Filippo Vicentini, Christof

Weitenberg, and Frank K. Wilhelm. Quantum computing and artificial intelligence: status and perspectives. *arXiv preprint arXiv:2505.23860*, 2025.

Andris Ambainis, András Gilyén, and Martins Kokainis. Tight bounds for hybrid quantum-classical query algorithms. *arXiv preprint arXiv:2610.06803*, 2026.

Jacob Biamonte, Peter Wittek, Nicola Pancotti, Patrick Rebentrost, Nathan Wiebe, and Seth Lloyd. Quantum machine learning. *Nature*, 549:195–202, 2017.

Gilles Brassard, Peter Høyer, Michele Mosca, and Alain Tapp. Quantum amplitude amplification and estimation. *Quantum Computation and Information*, page 53–74, 2002. ISSN 0271-4132. doi: 10.1090/conm/305/05215.

Ahmet Burak Catli, Sophia Simon, and Nathan Wiebe. Exponentially better bounds for quantum optimization via dynamical simulation. *arXiv preprint arXiv:2502.04285*, 2025.

Andrew M. Childs and Nathan Wiebe. Hamiltonian simulation using linear combinations of unitary operations. *Quantum Information and Computation*, 12:901–924, 2012. doi: 10.26421/qic12.11-12-1.

Arjan Cornelissen and Yassine Hamoudi. *A Sublinear-Time Quantum Algorithm for Approximating Partition Functions*, page 1245–1264. Society for Industrial and Applied Mathematics, January 2023. ISBN 9781611977554. doi: 10.1137/1.9781611977554.ch46.

Jens Eisert and John Preskill. Mind the gaps: The fraught road to quantum advantage. *arXiv preprint arXiv:2510.19928*, 2025.

Yonina C. Eldar and Gitta Kutyniok. *Compressed sensing: theory and applications*. Cambridge University Press, 2012.

Elies Gil-Fuster, Jens Eisert, and Carlos Bravo-Prieto. Understanding quantum machine learning also requires rethinking generalization. *Nat. Comm.*, 15:1–12, 2024a.

Elies Gil-Fuster, Jens Eisert, and Vedran Dunjko. On the expressivity of embedding quantum kernels. *Machine Learning: Science and Technology*, 5:025003, 2024b.

Elies Gil-Fuster, Casper Gyurik, Adrian Perez-Salinas, and Vedran Dunjko. On the relation between trainability and dequantization of variational quantum learning models. In *The Thirteenth International Conference on Learning Representations*, 2025.

Fabio A González, Alejandro Gallego, Santiago Toledo-Cortés, and Vladimir Vargas-Calderón. Learning with density matrices and random features. *Quantum Machine Intelligence*, 4(2):23, 2022.

Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery. ISBN 0897917855. doi: 10.1145/237814.237866.

Tobias Haug, Chris N. Self, and Myungshik S. Kim. Quantum machine learning of large datasets using randomized measurements. *Machine Learning: Science and Technology*, 4:015005, 2023. doi: 10.1088/2632-2153/acb0b4.

Vojtech Havlíček et al. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567: 209–212, 2019.

Laura J. Henderson, Kerstin Beer, Salini Karuvade, Riddhi Gupta, Angela White, and Sally Shrapnel. Quantum advantage without exponential concentration: Trainable kernels for symmetry-structured data. *arXiv preprint arXiv:2509.14337*, 2025.

Thomas Hubregtsen, David Wierichs, Elies Gil-Fuster, Peter-Jan H. S. Derks, Paul K. Faehrmann, and Johannes Jakob Meyer. Training quantum embedding kernels on near-term quantum computers. *Phys. Rev. A*, 106:04243, 2022.

Robbie King, David Gosset, Robin Kothari, and Ryan Babbush. Triply efficient shadow tomography. *PRX Quantum*, 6, 2025. ISSN 2691-3399. doi: 10.1103/prxquantum.6.010336.

Jonas Landman, Slimane Thabet, Constantin Dalyac, Hela Mhiri, and Elham Kashefi. Classically approximating variational quantum machine learning with random Fourier features. *arXiv preprint arXiv:2210.13200*, 2022.

Jiaqi Leng, Ethan Hickman, Joseph Li, and Xiaodi Wu. Quantum Hamiltonian descent. *arXiv preprint arXiv:2303.01471*, 2023.

Yao-Ting Lin. A note on quantum phase estimation. *arXiv preprint arXiv:2304.02241*, 2023.

Yunchao Liu, Srinivasan Arunachalam, and Kristan Temme. A rigorous and robust quantum speed-up in supervised machine learning. *Nature Phys.*, 17:1013–1017, 2021.

Guang Hao Low and Isaac L. Chuang. Hamiltonian simulation by qubitization. *Quantum*, 3:163, 2019. doi: 10.22331/q-2019-07-12-163.

Nikhil S. Mande and Ronald de Wolf. Tight bounds for quantum phase estimation and related problems. In Inge Li Gørtz, Martin Farach-Colton, Simon J. Puglisi, and Grzegorz Herman, editors, *31st Annual European Symposium on Algorithms (ESA 2023)*, volume 274 of *Leibniz International Proceedings in Informatics (LIPIcs)*, pages 81:1–81:16, Dagstuhl, Germany, 2023. Schloss Dagstuhl – Leibniz-Zentrum für Informatik. ISBN 978-3-95977-295-2. doi: 10.4230/LIPIcs.ESA.2023.81.

Akimoto Nakayama, Hayata Morisaki, Kosuke Mitarai, Hiroshi Ueda, and Keisuke Fujii. Explicit quantum surrogates for quantum kernel models. *arXiv preprint arXiv:2408.03000*, 2024.

Ashwin Nayak and Felix Wu. The quantum query complexity of approximating the median and related statistics. In *Proceedings of the Thirty-First Annual ACM Symposium on Theory of Computing*, STOC '99, page 384–393, New York, NY, USA, 1999. Association for Computing Machinery. ISBN 1581130678. doi: 10.1145/301250.301349.

Michael A. Nielsen and Isaac L. Chuang. *Quantum information and quantum computation*. Cambridge University Press Cambridge, 2000.

Ali Rahimi and Benjamin Recht. Random features for large-scale kernel machines. In *Adv. Neur. Inf. Proc. Sys.*, volume 20, 2007.

Patrick Rall and Bryce Fuller. Amplitude estimation from quantum signal processing. *Quantum*, 7: 937, 2023. ISSN 2521-327X. doi: 10.22331/q-2023-03-02-937.

Nicholas C. Rubin, Ryan Babbush, and Jarrod McClean. Application of fermionic marginal constraints to hybrid quantum algorithms. *New Journal of Physics*, 20(5):053020, May 2018. ISSN 1367-2630. doi: 10.1088/1367-2630/aab919.

Mehrad Sahebi, Alice Barthe, Yudai Suzuki, Zoë Holmes, and Michele Grossi. On dequantization of supervised quantum machine learning via random Fourier features. *arXiv preprint arXiv:2505.15902*, 2025.

Franz J. Schreiber, Jens Eisert, and Johannes Jakob Meyer. Classical surrogates for quantum learning models. *Phys. Rev. Lett.*, 131:100803, 2023.

Maria Schuld. Quantum machine learning models are kernel methods. *arXiv preprint arXiv:2101.11020*, 2021.

Maria Schuld and Nathan Killoran. Quantum machine learning in feature Hilbert spaces. *Phys. Rev. Lett.*, 122 4:040504, 2019.

Bernhard Schölkopf and Alexander J. Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. Adaptive computation and machine learning. MIT Press, 2002. ISBN 978-0-262-19475-4.

Shai Shalev-Shwartz and Shai Ben-David. *Understanding machine learning: From theory to algorithms*. Cambridge University Press, 2014.

Ryan Sweke, Eric Recio, Sofiene Jerbi, Elies Gil-Fuster, Bryce Fuller, Jens Eisert, and Johannes J. Meyer. Potential and limitations of random Fourier features for dequantizing quantum machine learning. *arXiv preprint arXiv:2309.11647*, 2023.

Ewin Tang and John Wright. Amplitude amplification and estimation require inverses. *arXiv preprint arXiv:2507.23787*, 2025.

Ewin Tang, John Wright, and Mark Zhandry. Conjugate queries can help. *arXiv preprint arXiv:2510.07622*, 2025.

John Tanner, Chon-Fai Kam, and Jingbo Wang. Non-variational supervised quantum kernel methods: a review. *arXiv preprint arXiv:2604.07896*, 2026.

Supanut Thanasilp, Samson Wang, Marco Cerezo, and Zoë Holmes. Exponential concentration in quantum kernel methods. *Nature Comm.*, 15:5200, 2024.

# Supplementary Material

## A Amplitude estimation toolbox

In this appendix, we collect standard results on amplitude estimation that are used throughout the paper. We present two approaches to the problem defined in Definition 5: a *sampling-based method* achieving a scaling of $\mathcal{O}(\varepsilon^{-2})$, and *quantum amplitude estimation* achieving $\mathcal{O}(\varepsilon^{-1})$. We also state the matching lower bound and the necessity of inverse access. Our presentation closely follows Tang and Wright [2025].

### A.1 Sampling-based amplitude estimation

In the classical setting, the amplitude $a$ is not directly accessible; one can only sample from the distribution induced by measuring $V|0\rangle$. Each measurement of the auxiliary register yields outcome $|0\rangle$ with probability $p = a^2$, so estimating $a$ reduces to estimating the Bernoulli parameter $p$ and taking a square root.

**Theorem A.1** (Sampling estimator: variance and concentration)**.** *Let $X \in [0,1]$ and let $Y_1, \dots, Y_M$ be i.i.d. Bernoulli random variables with $\mathbb{E}[Y_k] = X$. Define the estimator $\hat{X} := \sum_{k=1}^{M} Y_k/M$. Then the mean of the estimator fulfills $\mathbb{E}[\hat{X}] = X$ and its variance $\mathrm{Var}(\hat{X}) \leq 1/4M$. Further, for all $\varepsilon > 0$, the following tail bound holds $\mathbb{P}(|\hat{X} - X| \geq \varepsilon) \leq 2\exp(-2M\varepsilon^2)$. In particular, to guarantee $\mathbb{P}(|\hat{X} - X| \leq \varepsilon) \geq 1 - \delta$, it suffices to choose $M \in \mathcal{O}(\varepsilon^{-2}\log(\delta^{-1}))$.*

The proof is via standard Chernoff-based arguments, together with the facts that bounded random variables are sub-Gaussian, and that the observations $Y_1, \dots, Y_M$ are indeed independent.

**Corollary A.2** (Estimation of amplitudes)**.** *Let $a \in [0,1]$ and define $X = a^2$. Let $\hat{X}$ be as in Theorem A.1, and define $\hat{a} := \sqrt{\hat{X}}$. Then for $a \in \Omega(\varepsilon)$, the estimator $\hat{a}$ satisfies $|\hat{a} - a| \leq \varepsilon$ with probability at least $1 - \delta$ using, $M \in \mathcal{O}(\varepsilon^{-2}\log(\delta^{-1}))$ samples.*

### A.2 Quantum amplitude estimation

Quantum amplitude amplification and quantum amplitude estimation, introduced by Brassard et al. [2002], generalize the quadratic speedup achieved by Grover's search algorithm [Grover, 1996]. These techniques serve as fundamental building blocks in quantum algorithm design to speed-up certain subroutines.

**Theorem A.3** (Quantum amplitude estimation [Brassard et al., 2002, Aaronson and Rall, 2020])**.** *Let $a \in [0,1]$ and $V$ an $n_V$-qubit unitary as in Definition 5. Given access to $V$ and $V^\dagger$, there exists an estimator $\hat{a}$ such that for all $\varepsilon > 0$ and $\delta \in (0,1)$, $\mathbb{P}(|\hat{a} - a| \geq \varepsilon) \leq \delta$, using $M \in \mathcal{O}(\varepsilon^{-1}\log(\delta^{-1}))$ applications of $V$ and $V^\dagger$. More precisely, there exists a constant $C > 0$ such that $\mathbb{P}(|\hat{a} - a| \geq \varepsilon) \leq 2\exp(-CM\varepsilon)$ for all $a \in \Omega(\varepsilon)$. The procedure from Aaronson and Rall [2020] uses $n_V + 1$ qubits, and the total number of gates is $\mathcal{O}(M(G_V + n_V))$, where $G_V$ is the number of gates necessary to implement $V$ or $V^\dagger$.*

**Remark** The concentration bound in Theorem A.3 implies that the estimator behaves as if it had variance of order $1/M^2$, in contrast to the $1/M$ variance of sampling-based amplitude estimation from Theorem A.1.

Aaronson and Rall [2020] provide constructions for quantum amplitude estimation without using quantum phase estimation or controlled applications of $V$ (which was required by Brassard et al. [2002]). Their analysis assumes $a = \Omega(\varepsilon)$, ensuring the estimation problem is non-trivial: in this regime, additive error $\varepsilon$ corresponds to constant relative error. Tang and Wright [2025] argue that related techniques can be extended to regimes where $a$ is comparable to or smaller than $\varepsilon$. Finally, the query complexity of Theorem A.3 is optimal up to constant factors.

**Theorem A.4** (Optimality of quantum amplitude estimation [Nayak and Wu, 1999])**.** *Any quantum algorithm that estimates $a$ to additive error $\varepsilon$ with constant success probability requires $\Omega(\varepsilon^{-1})$ applications of $V$ and $V^\dagger$.*

Theorem A.4 follows from lower bounds for quantum counting [Nayak and Wu, 1999], which reduce to phase estimation lower bounds. Lin [2023] and Mande and de Wolf [2023] provide refinements and extensions to arbitrary success probabilities. In particular, the dependence on the failure probability $\delta$ is also tight [Mande and de Wolf, 2023]. Tang and Wright [2025] and Tang et al. [2025] recently proved that the quadratic speedup for quantum amplitude estimation requires access to the inverses.

**Theorem A.5** (Amplitude estimation requires inverse access [Tang and Wright, 2025])**.** *Suppose we have only access to the unitary $V \in \mathbb{C}^{d\times d}$ from above and no access to the inverse, then any algorithm for amplitude estimation must use $\Omega(\min(d, \varepsilon^{-2}))$ applications of $V$.*

Finally, we record a variant of amplitude estimation that controls not only the tails but also the bias of the estimate, suppressing the latter at logarithmic cost. We use this variant in our analysis of the list-and-sample algorithms, in Section 3.2 and Appendix C.1.

**Theorem A.6** (Nearly unbiased amplitude estimation [Cornelissen and Hamoudi, 2023])**.** *Let $|\psi\rangle$ be a quantum state and $\Pi$ a projector, and write $p = \|\Pi|\psi\rangle\|^2$. For every $t \geq 4$ and $b \in (0, 1)$ there is a quantum algorithm that outputs an estimate $\hat{p} \in [-2\pi, 2\pi]$ satisfying*

$$\left|\mathbb{E}[\hat{p}] - p\right| \leq b \qquad \textit{and} \qquad \mathrm{Var}[\hat{p}] \leq \frac{91\,p}{t^2} + b, \tag{A.1}$$

*using $\mathcal{O}\big(t \log\log(t) \log(\frac{t}{b})\big)$ applications, in expectation, of the reflections $\mathbb{I} - 2|\psi\rangle\langle\psi|$ and $\mathbb{I} - 2\Pi$ and controlled versions thereof; it consumes a single copy of $|\psi\rangle$, which is restored at the end with probability at least $1 - b$.*

# B The all-at-once observable

This appendix provides the formal results behind the *all-at-once* approach. We first establish the proofs that the inference sum $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ admits an exact representation as a single expectation value, and give an explicit circuit implementation with gate cost $\mathcal{O}(N(G + \log N))$. We then show that, if the feature-map unitary $U(x)$ admits a programmable implementation controlled by an $s$-bit description of the input, this cost can be reduced to $\tilde{\mathcal{O}}((N + G)s)$.

## B.1 Existence and implementation

**Theorem 2** (Single-observable representation)**.** *Consider the problem set-up from Definition 1, in particular the task of estimating the kernel function in Eq. (1). Then, there exists a bounded Hermitian observable $\mathcal{M}(\alpha, S)$, with $\|\mathcal{M}(\alpha, S)\|_\infty \leq \|\alpha\|_1$, and a quantum state $|\phi(x)\rangle$, such that*

$$\mathbb{E}_{|\phi(x)\rangle}\left[\mathcal{M}(\alpha, S)\right] = \langle\phi(x)|\mathcal{M}(\alpha, S)|\phi(x)\rangle = \sum_{i=1}^N \alpha_i k(x, x_i) =: f_\alpha(x). \tag{B.1}$$

*Proof.* A simple calculation confirms that it is sufficient to set $\mathcal{M}(\alpha, S) = \sum_{i=1}^N \alpha_i |\phi(x_i)\rangle\langle\phi(x_i)|$, since

$$\begin{aligned}\langle\phi(x)|\mathcal{M}(\alpha, S)|\phi(x)\rangle &= \langle\phi(x)|\left(\sum_{i=1}^N \alpha_i |\phi(x_i)\rangle\langle\phi(x_i)|\right)|\phi(x)\rangle \\ &= \sum_{i=1}^N \alpha_i \langle\phi(x)|\phi(x_i)\rangle\langle\phi(x_i)|\phi(x)\rangle \\ &= \sum_{i=1}^N \alpha_i k(x, x_i) = f_\alpha(x).\end{aligned} \tag{B.2}$$

The operator norm of the observable

$$\|\mathcal{M}(\alpha, S)\|_\infty = \left\|\sum_{i=1}^N \alpha_i |\phi(x_i)\rangle\langle\phi(x_i)|\right\|_\infty \leq \sum_{i=1}^N \|\alpha_i |\phi(x_i)\rangle\langle\phi(x_i)|\|_\infty = \sum_{i=1}^N |\alpha_i| = \|\alpha\|_1 \tag{B.3}$$

results from the triangle inequality. □

**Algorithm B.1** Single-observable sampling.

**Input:** $x \in \mathcal{X}$ ▷ New input.

**Output:** $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ ▷ Inference, Eq. (1).

1: Initialize $|0\rangle_{\text{data}}|0\rangle_{\text{idx}}|0\rangle_{\text{sign}}$. ▷ $n + \lceil \log_2(N) \rceil + 1$ qubits.

2: Single-input oracle $U(x)_{\text{data}}$: $|\tilde{\phi}(x)\rangle = |\phi(x)\rangle_{\text{data}}|0\rangle_{\text{idx}}|0\rangle_{\text{sign}}$ ▷ $G$ gates.

3: Quantum-state coefficient oracle $W(\alpha)_{\text{idx,sign}}$: ▷ $N$ gates.

$$|\phi(x)\rangle_{\text{data}}|0\rangle_{\text{idx}}|0\rangle_{\text{sign}} \mapsto \sum_{i=1}^N \sqrt{p_i}|\phi(x)\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}$$

4: Training-set oracle $O^\dagger(S)$: ▷ $\mathcal{O}(N(G + \log N))$ gates.

$$\sum_{i=1}^N \sqrt{p_i}|\phi(x)\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}} \mapsto \sum_{i=1}^N \sqrt{p_i}U^\dagger(x_i)|\phi(x)\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}$$

5: Norm coefficient oracle $O_N(\alpha)$: $\|\alpha\|_1$.

6: **return** Expectation value of observable: $\|\alpha\|_1|0\rangle\langle 0|_{\text{data}} \otimes \mathbb{I}_{\text{idx}} \otimes Z_{\text{sign}}$ ▷ Total number of gates $\mathcal{O}(N(G + \log N))$

Theorem 2 confirms that, in principle, the inference step in kernel methods, i.e., estimating $f_\alpha(x)$ for a data input $x$, can be achieved via a single expectation value, instead of taking a different expectation value for each element in the training set and then summing the results, each weighted by the corresponding $\alpha_i$. As presented, it may be unclear how to estimate the expectation value of the given observable with respect to the quantum state on a quantum device. A direct approach may attempt to construct a unitary gate that maps the $i^{\text{th}}$ computational basis element $|i\rangle$ into the quantum feature vector $|\phi(x_i)\rangle$, for $i \in \{1, \ldots, N\}$. Such a transformation may be costly to obtain just from having access to the state-preparation unitary gate $U(x)$. We next present Algorithm B.1, which can be used to implement the *all-at-once* observable using a few auxiliary qubits. We spell out each oracle call separately to reflect the sequential circuit structure. Note that Algorithm B.1 follows the "adjoint" method to estimate the kernel function (using $U(x)$ and its inverse $U^\dagger(x)$), as opposed to the SWAP test. This is an arbitrary choice, and the main difference between the two approaches is a trade-off of depth versus width [Hubregtsen et al., 2022, Schuld, 2021]. In the adjoint method, the depth of the circuit is at least twice the depth of $U(x)$. In the SWAP method, the number of qubits in the circuit is at least twice the number of qubits $U(x)$ acts on.

**Lemma B.1** (Single-observable implementation)**.** *Under the conditions of Theorem 2, there exists a quantum circuit with $n + \lceil \log_2(N) \rceil + 2$ qubits and $\mathcal{O}(N(G + \log N))$ gates which implements $\mathcal{M}(\alpha, S)$, as specified in Algorithm B.1. The algorithm makes one call to each of the following oracles: single-input oracle, training-set oracle, norm coefficient oracle, and quantum-state coefficient oracle. This algorithm involves measuring the expectation value of an auxiliary observable $\tilde{\mathcal{M}}(\alpha, S)$ fulfilling $\|\tilde{\mathcal{M}}(\alpha, S)\|_\infty = \|\alpha\|_1$.*

*Proof.* We prove the statement directly by providing a construction. Consider three registers: an $n$-qubit "data" register $\mathcal{H}_{\text{data}}$ and an $N$-dimensional "index" register $\mathcal{H}_{\text{idx}}$ (equivalently, a $\lceil \log_2(N) \rceil$-qubit register), and a 1-qubit "sign" register. Recall the notation $\alpha_i = \|\alpha\|_1 s_i p_i$, where $s_i = \operatorname{sign} \alpha_i$ and $p_i = |\alpha_i|/\|\alpha\|_1$. Calling the single-input oracle on the data register, we obtain the quantum state vector

$$U(x)|0\rangle_{\text{data}}|0\rangle_{\text{idx,sign}} \mapsto |\phi(x)\rangle_{\text{data}}|0\rangle_{\text{idx,sign}}. \tag{B.4}$$

We refer to this state vector as $|\tilde{\phi}(x)\rangle_{\text{data,idx,sign}}$. We next introduce an observable $\tilde{\mathcal{M}}(\alpha, S)$ which fulfills $f_\alpha(x) = \langle\tilde{\phi}(x)|\tilde{\mathcal{M}}(\alpha, S)|\tilde{\phi}(x)\rangle$. It then follows that

$$\langle\tilde{\phi}(x)|\tilde{\mathcal{M}}(\alpha, S)|\tilde{\phi}(x)\rangle = \langle\phi(x)|\mathcal{M}(\alpha, S)|\phi(x)\rangle, \tag{B.5}$$

where $\mathcal{M}(\alpha, S)$ is the observable defined in Theorem 2, and the r.h.s. expectation value is defined solely on the data register. To specify $\mathcal{M}(\alpha, S)$, we make use of three oracle calls, namely to the

quantum-state coefficient oracle $W(\alpha)$, the training-set oracle $O^\dagger(S)$, and the norm coefficient oracle $O_N(\alpha)$. With these, it is sufficient to set

$$\tilde{\mathcal{M}}(\alpha, S) \tag{B.6}$$
$$= \|\alpha\|_1 \left(O^\dagger(S)_{\text{data,idx}} \cdot W(\alpha)_{\text{idx,sign}}\right)^\dagger (|0\rangle\langle 0|_{\text{data}} \otimes \mathbb{I}_{\text{idx}} \otimes Z_{\text{sign}}) \left(O^\dagger(S)_{\text{data,idx}} \cdot W(\alpha)_{\text{idx,sign}}\right).$$

Intuitively, we recognized that the weighted sum over $\alpha$ can be instantiated using a construction reminiscent of the *linear combination of unitaries* (LCU) framework [Childs and Wiebe, 2012]. The extra sign register allows us to flexibly combine both positive and negative coefficients. We show that $\langle\tilde{\phi}(x)|\tilde{\mathcal{M}}(\alpha, S)|\tilde{\phi}(x)\rangle = f_\alpha(x)$. For clarity, as depicted in Fig. 3, instead of keeping the Heisenberg picture of $\tilde{\mathcal{M}}(\alpha, S)$, we apply the unitary gates $W(\alpha)$ and $O^\dagger(S)$ on the quantum state, one at a time,

$$\begin{aligned}
&\left(O^\dagger(S)_{\text{data,idx}} \cdot W(\alpha)_{\text{idx,sign}}\right) |\tilde{\phi}(x)\rangle \qquad (B.7)\\
=&O^\dagger(S)_{\text{data,idx}} \left(|\phi(x)\rangle_{\text{data}} \otimes \left(\sum_{i=1}^N \sqrt{p_i}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}\right)\right)\\
=&\left(\sum_{j=1}^N U^\dagger(x_j)_{\text{data}} \otimes |j\rangle\langle j|_{\text{idx}}\right)\left(|\phi(x)\rangle_{\text{data}} \otimes \left(\sum_{i=1}^N \sqrt{p_i}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}\right)\right)\\
=&\sum_{i,j=1}^N \sqrt{p_i}\left(U^\dagger(x_j)|\phi(x)\rangle_{\text{data}}\right) \otimes \left(|j\rangle_{\text{idx}} \underbrace{\langle j|i\rangle}_{\delta_{i,j}}\right) \otimes |s_i\rangle_{\text{sign}}\\
=&\sum_{i=1}^N \sqrt{p_i}\left(U^\dagger(x_i)|\phi(x)\rangle_{\text{data}}\right)|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}.
\end{aligned}$$

We consider the expectation value of the fixed observable $\|\alpha\|_1 (|0\rangle\langle 0|_{\text{data}} \otimes \mathbb{I}_{\text{idx}} \otimes Z_{\text{sign}})$ with respect to this quantum state, to get

$$\begin{aligned}
&\left(\sum_{i=1}^N \sqrt{p_i}\left(U^\dagger(x_i)|\phi(x)\rangle_{\text{data}}\right)|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}\right)^\dagger \|\alpha\|_1 (|0\rangle\langle 0|_{\text{data}} \otimes \mathbb{I}_{\text{idx}} \otimes Z_{\text{sign}}) \qquad (B.8)\\
&\quad\times \left(\sum_{j=1}^N \sqrt{p_j}\left(U^\dagger(x_j)|\phi(x)\rangle_{\text{data}}\right)|j\rangle_{\text{idx}}|s_j\rangle_{\text{sign}}\right)\\
=&\|\alpha\|_1 \sum_{i,j=1}^N \sqrt{p_i p_j}\langle\phi(x)|\underbrace{U(x_i)|0\rangle}_{|\phi(x_i)\rangle}\underbrace{\langle 0|U^\dagger(x_j)}_{\langle\phi(x_j)|}|\phi(x)\rangle \underbrace{\langle i|\mathbb{I}|j\rangle}_{\delta_{i,j}}\langle s_i|\underbrace{Z|s_j\rangle}_{s_j|s_j\rangle}\\
=&\sum_{i=1}^N \underbrace{s_i \underbrace{\|\alpha\|_1 p_i}_{|\alpha_i|}}_{\alpha_i} |\langle\phi(x)|\phi(x_i)\rangle|^2\\
=&\sum_{i=1}^N \alpha_i k(x, x_i) = f_\alpha(x).
\end{aligned}$$

We next bound the operator norm of $\tilde{\mathcal{M}}(\alpha, S)$. Making use of the invariance of norms under unitary transformations, the norm of $\tilde{\mathcal{M}}(\alpha, S)$ is equal to the norm

$$\|\tilde{\mathcal{M}}(\alpha, S)\|_\infty = \left\|\|\alpha\|_1 (|0\rangle\langle 0|_{\text{data}} \otimes \mathbb{I}_{\text{idx}} \otimes Z_{\text{sign}})\right\|_\infty = \|\alpha\|_1 \||0\rangle\langle 0|\|_\infty \|\mathbb{I}\|_\infty \|Z\|_\infty = \|\alpha\|_1 \tag{B.9}$$

of the fixed observable. We exploit the factoring of norms under the tensor product in the second line. Indeed, the Hermitian observable $\mathcal{M}(\alpha, S)$ fulfills $\|\mathcal{M}(\alpha, S)\|_\infty \leq \|\alpha\|_1$. □

These statements together prove Theorem 2.

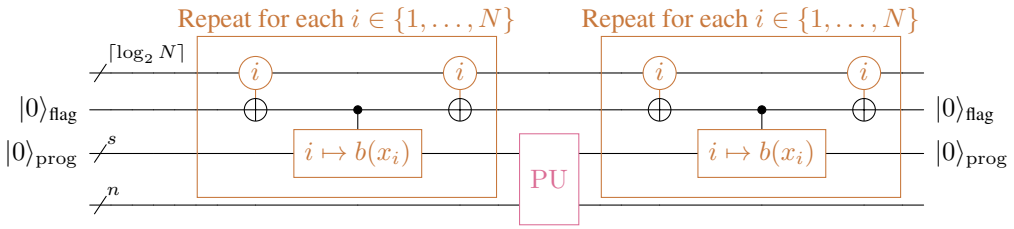



Figure B.1: Realizing the "training-set oracle" $O^\dagger(S)$ in Definition 6, using the programmable single-input oracle in Definition B.1. The input-data information is encoded onto the $s$-qubit program register as a sequence of $N$ controlled multi-target bitflips, and the (programmable) single-input oracle $\mathrm{PU}_{\mathrm{prog,data}}$ is called only once, coherently in superposition over programs. The associated gate cost is $\mathcal{O}((N+G)s)$. This construction improves on that presented in Fig. 2, where no access to the programmable oracle is assumed.

### B.2 Reduced gate cost via programmable feature maps

The number of gates required to implement the training-set oracle $O^\dagger(S)$ could be further reduced with an extra assumption, namely that the encoding unitary $U(x)$ can be programmed with a length-$s$ bitstring $b(x) \in \{0,1\}^s$ that depends only on $x$. In this case, we consider an auxiliary $s$-qubit "program" register, which controls the single-input oracle $U(x)$ acting on the $n$-qubit data register. To be precise, we assume we have access to the following programmable oracle:

**Definition B.1** (Programmable single-input oracle)**.** *Let $b : \mathcal{X} \to \{0,1\}^s$ be an efficiently computable map. The* programmable single-input oracle $\mathrm{PU}_{\mathrm{prog,data}}$ *is the unitary*

$$\mathrm{PU}_{\mathrm{prog,data}} = \sum_{p \in \{0,1\}^s} |p\rangle\langle p|_{\mathrm{prog}} \otimes U_p, \tag{B.10}$$

*where $\{U_p\}_{p\in\{0,1\}^s}$ is a family of $n$-qubit unitaries satisfying $U_{b(x)} = U(x)$ for all $x \in \mathcal{X}$. In particular, on inputs of the form $|b(x)\rangle_{\mathrm{prog}} \otimes |\psi\rangle_{\mathrm{data}}$, the oracle acts as*

$$\mathrm{PU}_{\mathrm{prog,data}}\left(|b(x)\rangle_{\mathrm{prog}} \otimes |\psi\rangle_{\mathrm{data}}\right) = |b(x)\rangle_{\mathrm{prog}} \otimes \left(U(x)|\psi\rangle_{\mathrm{data}}\right). \tag{B.11}$$

*The associated gate cost is $\mathcal{O}(Gs)$.*

We emphasize that the $\mathcal{O}(Gs)$ gate cost is not automatic: a black-box family $\{U_p\}$ of $2^s$ arbitrary unitaries would require $\mathcal{O}(2^s G)$ gates to implement coherently via multi-controlled gates. The improved scaling reflects a structural assumption on $U(x)$ – namely, that it admits an implementation as a sequence of $G$ elementary gates, each parameterized by $\mathcal{O}(1)$ bits of the program register $b(x)$. Under this assumption, each elementary gate is realized with $\mathcal{O}(s)$ controlled primitives, yielding a total cost of $\mathcal{O}(Gs)$ rather than the naive $\mathcal{O}(2^s G)$. This is a meaningful assumption: it is weaker than quantum random-access memory but stronger than the black-box access of Definition 4.

**Lemma B.2** (Training-set-oracle from programmable single-input oracle)**.** *The training-set oracle can be realized with a single query to the programmable single-input oracle* $\mathrm{PU}_{\mathrm{prog,data}}$*. The associated gate cost is $\tilde{\mathcal{O}}((N+G)s)$, and the procedure uses $s+1$ extra qubits.*

*Proof.* See Figure B.1 for a visual proof. The associated gate cost to load the programs in the auxiliary register is, for each $i \in \{1, \ldots, N\}$: $\mathcal{O}(\log N)$ gates to perform a multi-controlled bitflip on the flag register conditioned on the index being $i$ plus $\mathcal{O}(s)$ controlled bitflips on the program register conditioned on the flag register. Then, calling the programmable single-input oracle in superposition uses $\mathcal{O}(Gs)$ gates. Uncomputing the program register has the same cost as loading it. In total, the number of gates scale as $\mathcal{O}(N \log N + (N+G)s) = \tilde{\mathcal{O}}((N+G)s)$. □

**Gate complexity from programmable single-input oracle.** Assuming access to the programmable single-input oracle from Definition B.1, the number of gates for the *all-at-once* algorithms in Theorem 5 reduces to $\tilde{\mathcal{O}}((N+G)s\|\alpha\|_1^2/\varepsilon^2)$ via sampling and $\tilde{\mathcal{O}}(((N+G)s+n)\|\alpha\|_1/\varepsilon)$ via quantum amplitude estimation. Compared to the bounds in Theorem 5, the multiplicative factor of $N$ on $G$ is replaced by a multiplicative factor of $s$, at the cost of an additive $\mathcal{O}(Ns)$ term per

**Algorithm C.1** Unified inference for quantum kernel methods.

**Input:** $x \in \mathcal{X}$ (new input), $\varepsilon > 0$ (additive precision), trained coefficients $\alpha \in \mathbb{R}^N$.
**Output:** Estimate of $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ to $\varepsilon$ additive precision.

**Choice 1: How to approximate the sum.**
(a) LIST-AND-SUM (FIXED BUDGET): estimate every $k(x, x_i)$ with the same budget $M$. [Algorithms C.3 and C.5]
(b) LIST-AND-SUM (ADAPTIVE BUDGET): estimate $k(x, x_i)$ with a separate budget $M_i$. [Algorithms C.4 and C.6]
(c) ALL-AT-ONCE: encode $f_\alpha(x)$ as a single expectation value $\langle\tilde{\phi}(x)|\tilde{\mathcal{M}}(\alpha, S)|\tilde{\phi}(x)\rangle$. [Algorithms B.1 and C.9]

**Choice 2: How to estimate expectation values.**
(I) SAMPLING: measure $M$ times; precision $\mathcal{O}(1/\sqrt{M})$ per estimate. [Theorem A.1]
(II) QUANTUM AMPLITUDE ESTIMATION: use $V^{(\dagger)}$ coherently $M$ times; precision $\mathcal{O}(1/M)$. [Theorem A.3]

**switch** Choice 1: **do**
  **case** (a) or (b) ▷ List-and-sum
    **for** $i \in \{1, \ldots, N\}$ **do**
      Reading oracle $O_R(\alpha, i)$: obtain $\alpha_i$.
      **switch** Choice 2: **do**
        **case** (I) ▷ Sampling
          Estimate $k(x, x_i)$ using $M_i$ shots: $\hat{k}^{(M_i)}(x, x_i)$.
        **case** (II) ▷ Quantum amplitude estimation
          Estimate $k(x, x_i)$ using $M_i$ coherent queries: $\hat{k}^{(M_i)}(x, x_i)$.
      Compute $\alpha_i \hat{k}^{(M_i)}(x, x_i)$.
    **end for**
    **return** $\hat{f}_\alpha(x) = \sum_{i=1}^N \alpha_i \hat{k}(x, x_i)$.
  **case** (c) ▷ All-at-once
    **switch** Choice 2: **do**
      **case** (I) ▷ Sampling
        Measure $M$ times the *all-at-once* observable (Figure 3, Algorithm B.1).
        **return** $\hat{f}_\alpha^{(M)}(x)$.
      **case** (II) ▷ Quantum amplitude estimation
        Consider the *amplitude-encoding* unitary $V$ from Figure 4 (Algorithm C.8).
        Run quantum amplitude estimation on $V$ to estimate $\sqrt{f_\pm}$ to precision $\mathcal{O}(1/M)$.
        **return** $\hat{f}_\alpha^{(2M)}(x) = \|\alpha\|_1(\hat{f}_+ - \hat{f}_-)$.

query. In the regime $s \ll N$ – the natural setting in which the programmable oracle assumption is meaningful – this represents a substantial improvement. Relative to *list-and-sum adaptive*, *all-at-once* now incurs only an additive $\mathcal{O}(Ns)$ overhead and an $s$ multiplicative factor on $G$, rather than the $N$ multiplicative factor of Theorem 5.

## C A zoo of algorithms

In this appendix, we provide detailed analyses and proofs for each of the algorithmic combinations presented in Algorithm C.1 and summarized in Theorem 4. For each combination of choices – how to approximate the sum (*list-and-sum* with *fixed* or *adaptive* budget, or *all-at-once*) and how to estimate expectation values (*sampling* or *quantum amplitude estimation*) – we state the corresponding algorithm, derive the optimal parameter choices where applicable, and prove the resulting runtime complexity. The results are collected in Table C.1.

Table C.1: Total cost required to ensure $\varepsilon$ additive precision for each of the algorithms below. We highlight *list-and-sum* with fixed budget via sampling as the *naive* approach ($\ddagger$), *all-at-once* via quantum amplitude estimation as *query-optimal* ($\star$), and *list-and-sum* with adaptive budget via quantum amplitude estimation as *gate-optimal* ($\diamond$).

| **Strategy** | **Sampling** | **Quantum amplitude estimation** | **Section reference** |
|---|---|---|---|
| List-and-sum, fixed | $\mathcal{O}\left(\frac{N\|\alpha\|_2^2}{\varepsilon^2}\right)^{\ddagger}$ [Cor. C.2] | $\tilde{\mathcal{O}}\left(\frac{N\|\alpha\|_2}{\varepsilon}\right)$ [Cor. C.6] | C.1 |
| List-and-sum, adaptive | $\mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right)$ [Cor. C.3] | $\tilde{\mathcal{O}}\left(\frac{\|\alpha\|_{2/3}}{\varepsilon}\right)^{\diamond}$ [Cor. C.7] | C.1 |
| All-at-once | $\mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right)$ [Thm. C.8] | $\mathcal{O}\left(\frac{\|\alpha\|_1}{\varepsilon}\right)^{\star}$ [Thm. C.9] | C.2 |

## C.1 List-and-sum

We first provide a template for all *list-and-sum* theorems and algorithms, in Theorem C.1 and Algorithm C.2. We next specialize the template into the four possible choices. We collect these results in Table C.2.

**Theorem C.1** (List-and-sum algorithm template analysis)**.** *Let $X_1, \dots, X_N \in [0,1]$ and $X = \sum_{i=1}^N \alpha_i X_i$, let $r \in \{1,2\}$, and let $\varepsilon > 0$. Let $\beta_1, \dots, \beta_N \geq 0$ be bias parameters and let $\gamma \geq 0$ be a variance floor. Assume we have mutually independent $M_i$-query estimators $\hat{X}_i$ of $X_i$ obeying*

$$\left|\mathbb{E}_{\hat{X}}[\hat{X}_i] - X_i\right| \leq \beta_i \qquad \text{and} \qquad \mathrm{Var}_{\hat{X}}[\hat{X}_i] \leq \frac{c}{M_i^r} + \gamma \tag{C.1}$$

*for some constant $c > 0$. Define the $M$-query estimator $\hat{X} \coloneqq \sum_{i=1}^N \alpha_i \hat{X}_i$, with $M = \sum_{i=1}^N M_i$ and $M \geq N$, and write $B \coloneqq \sum_{i=1}^N |\alpha_i| \beta_i$ for the accumulated bias. Assume the variance floor obeys*

$$\gamma \|\alpha\|_2^2 \leq \frac{\varepsilon^2}{48}. \tag{C.2}$$

*Then the following numbers of queries $M$ are sufficient to ensure $|\hat{X} - X| < \varepsilon + B$ with probability at least $2/3$:*

1. *With fixed budget $M_i = M/N$ for all $i \in \{1, \dots, N\}$: $M \in \mathcal{O}(\varepsilon^{-2/r} N \|\alpha\|_2^{2/r})$.*
2. *With optimal query allocation $M_i \in \mathcal{O}(\varepsilon^{-2/r} |\alpha_i|^{2/(r+1)} \|\alpha\|_{2/(r+1)}^{2/(r(r+1))})$, which in turn yields $M \in \mathcal{O}(\varepsilon^{-2/r} \|\alpha\|_{2/(r+1)}^{2/r})$.*

*For any $\delta \in (0,1)$, a median-of-means amplification – running $\mathcal{O}(\log(\delta^{-1}))$ independent copies of the estimator and returning their median – reduces the failure probability to $\delta$ at a multiplicative overhead of $\mathcal{O}(\log(\delta^{-1}))$ in the number of queries, without changing the $\varepsilon$-dependence.*

*Proof.* Write $\mu_i \coloneqq \mathbb{E}_{\hat{X}}[\hat{X}_i]$ and $Z \coloneqq \sum_{i=1}^N \alpha_i (\hat{X}_i - \mu_i)$, and split the estimation error into a centered part and a deterministic bias,

$$|\hat{X} - X| \leq |Z| + \sum_{i=1}^N |\alpha_i| |\mu_i - X_i| \leq |Z| + B, \tag{C.3}$$

where the second inequality uses the first hypothesis in Eq. (C.1). The bias contribution $B$ is deterministic, so it suffices to ensure $|Z| < \varepsilon$ with probability at least $2/3$. By construction $\mathbb{E}_{\hat{X}}[Z] = 0$, and by independence of the estimators together with the second hypothesis in Eq. (C.1) and the floor condition Eq. (C.2),

$$\mathrm{Var}_{\hat{X}}[Z] = \sum_{i=1}^N \alpha_i^2 \mathrm{Var}_{\hat{X}}[\hat{X}_i] \leq c \sum_{i=1}^N \frac{\alpha_i^2}{M_i^r} + \gamma \|\alpha\|_2^2 \leq c \sum_{i=1}^N \frac{\alpha_i^2}{M_i^r} + \frac{\varepsilon^2}{48}. \tag{C.4}$$

Chebyshev's inequality gives $\mathbb{P}[|Z| \geq \varepsilon] \leq \mathrm{Var}_{\hat{X}}[Z]/\varepsilon^2$, so it is enough to certify $\mathrm{Var}_{\hat{X}}[Z] \leq \varepsilon^2/3$. By Eq. (C.4), this holds as soon as the query budgets satisfy

$$\sum_{i=1}^{N} \frac{\alpha_i^2}{M_i^r} \leq \kappa, \qquad \kappa := \frac{\varepsilon^2}{4c} \in \Theta(\varepsilon^2), \tag{C.5}$$

since then $\mathrm{Var}_{\hat{X}}[Z] \leq \varepsilon^2/4 + \varepsilon^2/48 = 13\varepsilon^2/48 \leq \varepsilon^2/3$, whence $\mathbb{P}[|Z| < \varepsilon] \geq 35/48 > 2/3$. We stress that the variance floor $\gamma$ does not depend on the budgets $M_i$ and therefore plays no role in the allocation problem below; it enters solely through the standing hypothesis Eq. (C.2).

We next give shot allocations realizing Eq. (C.5). In the simpler case of *fixed budget* $M_i = M/N$ for all $i$, we directly obtain

$$\sum_{i=1}^{N} \frac{\alpha_i^2}{M_i^r} = \frac{N^r}{M^r} \sum_{i=1}^{N} \alpha_i^2 = \frac{N^r \|\alpha\|_2^2}{M^r} \leq \kappa \qquad \Longleftrightarrow \qquad M \geq \frac{N\|\alpha\|_2^{2/r}}{\kappa^{1/r}}, \tag{C.6}$$

so that $M \in \mathcal{O}(\varepsilon^{-2/r} N \|\alpha\|_2^{2/r})$ suffices. To find the optimal query allocation (partly inspired by Rubin et al. [2018]), we consider the optimization problem

$$\min_{M_i > 0} M = \sum_{i=1}^{N} M_i, \qquad \text{s.t.} \sum_{i=1}^{N} \frac{\alpha_i^2}{M_i^r} \leq \kappa, \tag{C.7}$$

in which integrality and the lower bounds $M_i \geq 1$ are temporarily dropped; both are restored by the rounding step at the end. We may assume $\alpha_i \neq 0$ for all $i$, as indices with $\alpha_i = 0$ contribute to neither the objective nor the constraint and can be dropped. Each map $M_i \mapsto \alpha_i^2/M_i^r$ is convex and strictly decreasing on $M_i > 0$, so the feasible region – a sublevel set of a convex function – is convex and the linear objective makes this a convex program; since the objective is increasing in each $M_i$ while the constraint function is decreasing, the constraint is active at any minimum, so it may be imposed as an equality, and the stationary point of the Lagrangian computed below is the global minimum. The Lagrangian of the problem is

$$\mathcal{L} = \sum_{i=1}^{N} M_i + \lambda \left( \sum_{i=1}^{N} \frac{\alpha_i^2}{M_i^r} - \kappa \right), \tag{C.8}$$

and the first order condition yields

$$\frac{\partial \mathcal{L}}{\partial M_i} = 1 - r\lambda \frac{\alpha_i^2}{M_i^{r+1}} = 0, \tag{C.9}$$

$$M_i^{r+1} = r\lambda\alpha_i^2, \tag{C.10}$$

$$M_i = \left(r\lambda\alpha_i^2\right)^{\frac{1}{(r+1)}}. \tag{C.11}$$

Introducing this condition into the constraint, we obtain

$$\sum_{i=1}^{N} \frac{\alpha_i^2}{M_i^r} = \sum_{i=1}^{N} \frac{\alpha_i^2}{(r\lambda\alpha_i^2)^{\frac{r}{r+1}}} = \frac{1}{(r\lambda)^{\frac{r}{r+1}}} \sum_{i=1}^{N} \left(\alpha_i^2\right)^{1-\frac{r}{r+1}} = \frac{\sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}}}{(r\lambda)^{\frac{r}{r+1}}} = \kappa, \tag{C.12}$$

$$\frac{\sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}}}{\kappa r^{\frac{r}{r+1}}} = \lambda^{\frac{r}{r+1}}, \tag{C.13}$$

$$\lambda = \frac{\left(\sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}}\right)^{\frac{r+1}{r}}}{r\kappa^{\frac{r+1}{r}}}. \tag{C.14}$$

We can use the value of the multiplier $\lambda$ to reach the final result given by

$$M_i = \left(r\lambda\alpha_i^2\right)^{\frac{1}{r+1}} = \left( r \frac{\left(\sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}}\right)^{\frac{r+1}{r}}}{r\kappa^{\frac{r+1}{r}}} \alpha_i^2 \right)^{\frac{1}{r+1}} \tag{C.15}$$

$$= \frac{|\alpha_i|^{\frac{2}{r+1}}}{\sqrt[r]{\kappa}} \left( \sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}} \right)^{\frac{1}{r}} = \frac{|\alpha_i|^{\frac{2}{r+1}}}{\sqrt[r]{\kappa}} \|\alpha\|_{\frac{2}{r+1}}^{\frac{2}{r(r+1)}}.$$

Table C.2: Runtime scaling of *list-and-sum* algorithms.

| **Strategy** | **Sampling** ($r = 1$) | **Quantum amplitude estimation** ($r = 2$) |
|---|---|---|
| Fixed budget | $M \in \mathcal{O}\big(\varepsilon^{-2} N \|\alpha\|_2^2\big)$ | $M \in \tilde{\mathcal{O}}\big(\varepsilon^{-1} N \|\alpha\|_2\big)$ |
| Adaptive budget | $M_i \in \mathcal{O}\big(\varepsilon^{-2} \|\alpha_i\| \|\alpha\|_1\big)$<br>$M \in \mathcal{O}\big(\varepsilon^{-2} \|\alpha\|_1^2\big)$ | $M_i \in \tilde{\mathcal{O}}\Big(\varepsilon^{-1} \|\alpha_i\|^{2/3} \|\alpha\|_{2/3}^{1/3}\Big)$<br>$M \in \tilde{\mathcal{O}}\big(\varepsilon^{-1} \|\alpha\|_{2/3}\big)$ |

Spelling out the $\varepsilon$ dependence of $\kappa$, we obtain

$$M_i \in \mathcal{O}\left(\varepsilon^{-\frac{2}{r}} |\alpha_i|^{\frac{2}{r+1}} \|\alpha\|^{\frac{2}{r(r+1)}}_{\frac{2}{r+1}}\right). \tag{C.16}$$

With this, we can reach the final result

$$\begin{aligned} M = \sum_{i=1}^{N} M_i &= \frac{1}{\sqrt[r]{\kappa}} \left(\sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}}\right) \|\alpha\|^{\frac{2}{r(r+1)}}_{\frac{2}{r+1}} \\ &= \frac{1}{\sqrt[r]{\kappa}} \left(\sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}}\right) \left(\sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}}\right)^{\frac{1}{r}} \\ &= \frac{1}{\sqrt[r]{\kappa}} \left(\sum_{i=1}^{N} |\alpha_i|^{\frac{2}{r+1}}\right)^{\frac{r+1}{r}} = \frac{\|\alpha\|^{\frac{2}{r}}_{\frac{2}{r+1}}}{\sqrt[r]{\kappa}}, \end{aligned} \tag{C.17}$$

the full expression including $\varepsilon$ in turn becoming

$$M \in \mathcal{O}\Big(\varepsilon^{-\frac{2}{r}} \|\alpha\|^{\frac{2}{r}}_{\frac{2}{r+1}}\Big). \tag{C.18}$$

Rounding each budget up to $\max\{1, \lceil M_i \rceil\}$ restores integrality and the constraint $M_i \geq 1$ dropped above, only tightens Eq. (C.5), and increases the total budget by at most $N$, which is absorbed because the continuous total budget before rounding is at least $N$.

Finally, we address the amplification of the success probability. Let $K = 2\lceil 5 \log(\delta^{-1}) \rceil + 1$, let $\hat{X}^{(1)}, \ldots, \hat{X}^{(K)}$ be independent copies of the estimator, each with budget $M$, and let $\tilde{X}$ denote their median, i.e., the middle order statistic, $K$ being odd. Call a copy *good* if it lands in the open interval $I := (X - \varepsilon - B, X + \varepsilon + B)$, which by the above happens independently with probability at least $35/48$ for each copy. We stress that $I$ is centred at the true value $X$ and is the same for all copies, so the bias does not interfere with the argument. If more than $K/2$ copies are good, then $\tilde{X} \in I$: were $\tilde{X}$ to reach or exceed $\sup I$, the $(K+1)/2$ largest copies would all reach or exceed $\sup I$ and hence be bad, leaving at most $(K-1)/2 < K/2$ good copies – a contradiction – and symmetrically below $\inf I$. By Hoeffding's inequality applied to the number of good copies,

$$\mathbb{P}\left[\#\{\text{good copies}\} \leq \frac{K}{2}\right] \leq \exp\left(-2K\left(\frac{35}{48} - \frac{1}{2}\right)^2\right) = \exp\left(-\frac{121K}{1152}\right) \leq \delta, \tag{C.19}$$

where the last inequality uses $K \geq 10 \log(\delta^{-1}) \geq 1152/121 \log(\delta^{-1})$. Hence $|\tilde{X} - X| < \varepsilon + B$ holds with probability at least $1 - \delta$ at a total cost of $\mathcal{O}(M \log(\delta^{-1}))$ queries. □

Below, we spell out the resulting costs for all four *list-and-sum* variants as corollaries of Theorem C.1. The two sampling variants instantiate the template with an exactly unbiased estimator and no variance floor, so that both the accumulated bias $B$ and the floor condition Eq. (C.2) are vacuous; the quantum-amplitude-estimation variants instead go through Lemma C.5 and acquire the polylogarithmic factor recorded there.

**Corollary C.2** (Fixed budget, sampling)**.** *Consider Algorithm C.3 and the problem set-up in Definition 1. To estimate the quantum kernel model prediction in Eq. (1) to additive precision $\varepsilon$ with constant success probability, the runtime scaling of Algorithm C.3 is dominated by the query complexity*

$$M \in \mathcal{O}\left(\frac{N \|\alpha\|_2^2}{\varepsilon^2}\right). \tag{C.20}$$

**Algorithm C.2** List-and-sum algorithm template.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Input:** $\varepsilon > 0$ ▷ Required additive precision.
**Input:** $b \geq 0$ ▷ Bias parameter; $b = 0$ for $r = 1$, Eq. (C.24) for $r = 2$.
**Input:** $M_1, \ldots, M_N$ with $M = \sum_{i=1}^N M_i \geq N$ ▷ Per-term query budgets.
**Output:** $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ to $\varepsilon + B$ additive precision, $B := b\|\alpha\|_1$ ▷ Inference, Eq. (1).

1: **for** $i \in \{1, \ldots, N\}$ **do**:
2: Reading oracle $O_R(\alpha, i)$: $\alpha_i$
3: Estimate $k(x, x_i)$ using $M_i$ queries to single-input oracle: $\hat{k}^{(M_i)}(x, x_i)$
▷ Bias $\leq b$, variance $\mathcal{O}(1/M_i^r + b)$, with $r \in \{1, 2\}$.
4: Multiply $\hat{k}^{(M_i)}(x, x_i)$ with the coefficient $\alpha_i$ ▷ Bias $\leq |\alpha_i| b$, variance $\mathcal{O}\left(\alpha_i^2 \left(\frac{1}{M_i^r} + b\right)\right)$.
5: **end for** ▷ Total number of queries is $M = \sum_{i=1}^N M_i$.
6: **return** $\hat{f}_\alpha^{(M)}(x) := \sum_{i=1}^N \alpha_i \hat{k}^{(M_i)}(x, x_i)$
▷ Total bias $B = b\|\alpha\|_1$, $\mathcal{O}\left(\sum_{i=1}^N \frac{\alpha_i^2}{M_i^r} + b\|\alpha\|_2^2\right)$.
▷ Sufficient: $b\|\alpha\|_2^2 \leq \varepsilon^2/48$ and $M \in \mathcal{O}(\varepsilon^{-2/r}\|\alpha\|_{2/(r+1)}^{2/r})$.

**Algorithm C.3** List-and-sum with fixed budget, via sampling.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Input:** $\varepsilon > 0$ ▷ Required additive precision.
**Input:** $M \geq N$ ▷ Total shot budget, split evenly.
**Output:** $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ to $\varepsilon$ additive precision. ▷ Inference, Eq. (1).

1: **for** $i \in \{1, \ldots, N\}$ **do**:
2: Reading oracle $O_R(\alpha, i)$: $\alpha_i$
3: Single-input oracle for $U(x)$ and $U^\dagger(x_i)$: estimate $k(x, x_i)$ using $M/N$ shots: $\hat{k}^{(M/N)}(x, x_i)$.
▷ Unbiased, variance $\mathcal{O}(N/M)$.
4: Multiply $\hat{k}^{(M/N)}(x, x_i)$ with the coefficient $\alpha_i$ ▷ Variance $\mathcal{O}\left(\frac{N\alpha_i^2}{M}\right)$.
5: **end for** ▷ Total number of shots is $M$.
6: **return** $\hat{f}_\alpha^{(M)}(x) = \sum_{i=1}^N \alpha_i \hat{k}^{(M/N)}(x, x_i)$
▷ Bias $B = 0$, total variance $\mathcal{O}\left(\frac{N\|\alpha\|_2^2}{M}\right)$
Sufficient: $M \in \mathcal{O}(\varepsilon^{-2} N \|\alpha\|_2^2)$.

*Proof.* Apply Theorem C.1 with $X_i = k(x, x_i) \in [0, 1]$ and the estimators of Theorem A.1, which are exactly unbiased with $\mathrm{Var} \leq 1/4M_i$: the hypotheses of Eq. (C.1) thus hold with $r = 1$, $\beta_i = 0$, $\gamma = 0$ and $c = 1/4$. The floor condition Eq. (C.2) is satisfied trivially and the accumulated bias vanishes, $B = 0$, so the fixed-budget case of the theorem gives the claim. □

**Corollary C.3** (Adaptive budget, sampling)**.** *Consider Algorithm C.4 and the problem set-up in Definition 1. To estimate the quantum kernel model prediction in Eq. (1) to additive precision $\varepsilon$ with constant success probability, the runtime scaling of Algorithm C.4 is dominated by the query complexity*

$$M \in \mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right). \tag{C.21}$$

*The optimal allocation of shots is*

$$M_i \in \mathcal{O}\left(\frac{|\alpha_i|\|\alpha\|_1}{\varepsilon^2}\right). \tag{C.22}$$

*Proof.* As in Corollary C.2, the estimators of Theorem A.1 satisfy Eq. (C.1) with $r = 1$, $\beta_i = 0$, $\gamma = 0$ and $c = 1/4$, so that Eq. (C.2) holds trivially and $B = 0$. The claim is the optimal-allocation case of Theorem C.1, where $2/(r+1) = 1$ and $2/(r(r+1)) = 1$ at $r = 1$. □

**Algorithm C.4** List-and-sum with adaptive budget, via sampling.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Input:** $\varepsilon > 0$ ▷ Required additive precision.
**Input:** $M_1, \dots, M_N$ with $M = \sum_{i=1}^N M_i \geq N$ ▷ Per-term shot budgets.
**Output:** $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ to $\varepsilon$ additive precision. ▷ Inference, Eq. (1).

1: **for** $i \in \{1, \dots, N\}$ **do**:
2: Reading oracle $O_R(\alpha, i)$: $\alpha_i$
3: Single-input oracle for $U(x)$ and $U^\dagger(x_i)$: estimate $k(x, x_i)$ using $M_i$ shots: $\hat{k}^{(M_i)}(x, x_i)$ ▷ Unbiased, variance $\mathcal{O}(1/M_i)$.
4: Multiply $\hat{k}^{(M_i)}(x, x_i)$ with the coefficient $\alpha_i$ ▷ Variance $\mathcal{O}(\alpha_i^2/M_i)$.
5: **end for** ▷ Total number of shots is $M = \sum_{i=1}^N M_i$.
6: **return** $\hat{f}_\alpha^{(M)}(x) := \sum_{i=1}^N \alpha_i \hat{k}^{(M_i)}(x, x_i)$
▷ Bias $B = 0$, total variance $\mathcal{O}(\sum_{i=1}^N \alpha_i^2/M_i)$, Sufficient: $M \in \mathcal{O}(\varepsilon^{-2}\|\alpha\|_1^2)$.

For the variants using quantum amplitude estimation, we first give the nearly unbiased estimator that results from instantiating Cornelissen and Hamoudi [2023] in our problem set-up.

**Lemma C.4** (Nearly unbiased kernel-value estimator)**.** *Let $x, x_i \in \mathcal{X}$ and $b \in (0, 1)$. In the access model of Definition 4, for every $M \geq C_0 \log\log(M) \log(\frac{M}{b})$ there is an estimator $\hat{k}_i$ of $k(x, x_i)$ that uses at most $M$ queries in expectation to the single-input oracle (counting $U(\cdot)$, $U^\dagger(\cdot)$, and controlled versions thereof), satisfies $|\hat{k}_i| \leq 2\pi$, and obeys*

$$\left|\mathbb{E}[\hat{k}_i] - k(x, x_i)\right| \leq b, \qquad \mathrm{Var}[\hat{k}_i] \leq \frac{c \log\log^2(M) \log^2(\frac{M}{b})}{M^2} + b, \tag{C.23}$$

*for universal constants $C_0, c > 0$.*

*Proof.* Instantiate Theorem A.6 with $|\psi\rangle = U(x)|0\rangle^{\otimes n}$ and $\Pi = U(x_i)|0\rangle^{\otimes n}\langle 0|^{\otimes n} U^\dagger(x_i)$, so that $p = |\langle\psi(x_i)|\psi(x)\rangle|^2 = k(x, x_i)$. The reflection $\mathbb{I} - 2|\psi\rangle\langle\psi| = U(x)\big(\mathbb{I} - 2|0\rangle\langle 0|^{\otimes n}\big)U^\dagger(x)$ costs two single-input queries plus oracle-independent gates, and likewise for $\mathbb{I} - 2\Pi$; controlled reflections decompose into controlled single-input queries, which Definition 4 grants, and preparing $|\psi\rangle$ costs one further query. The query count of Theorem A.6 is stated in expectation; we keep it so and defer the conversion to a worst-case bound to the point of use (Lemma C.5), where a single truncation over the whole algorithm suffices. Choosing $t = \Theta\big(M/(\log\log(M)\log(\frac{M}{b}))\big)$ and invoking Theorem A.6 with bias parameter $b$ yields the claim, using $p \leq 1$ in the variance bound; the bias and variance guarantees of Theorem A.6 are unconditional, so no interaction with the state-restoration branch needs to be tracked. □

Next, we spell out how the poly-logarithmic factors affect the scaling of the complete estimator:

**Lemma C.5** (Instantiating the template with quantum amplitude estimation)**.** *Consider the problem set-up in Definition 1 and let $\varepsilon > 0$. The case $\alpha = 0$ is solved exactly by returning zero. Assume henceforth that $\alpha \neq 0$. Set the uniform bias parameter*

$$b := \min\left\{\frac{1}{4}, \frac{\varepsilon}{4\|\alpha\|_1}, \frac{\varepsilon^2}{192\|\alpha\|_2^2}\right\}, \tag{C.24}$$

*and let $\tilde{M}_1, \dots, \tilde{M}_N$ be the budgets provided by Theorem C.1 applied with $r = 2$, variance floor $\gamma = b$, precision $\varepsilon/2$, and either of its two allocation rules, writing $\tilde{M} := \sum_{i=1}^N \tilde{M}_i$. Run the estimator of Lemma C.4 for each $i$ at bias parameter $b$ with the inflated budget $M_i := \lceil \Lambda \tilde{M}_i \rceil$, where*

$$\Lambda := C_1 \log\log(\tilde{M}) \log\left(\frac{\tilde{M}}{b}\right), \tag{C.25}$$

*for a suitable universal constant $C_1$. Then $|\hat{f}_\alpha(x) - f_\alpha(x)| < \varepsilon$ with probability at least $2/3$, at a total query cost $M = \sum_{i=1}^N M_i \in \mathcal{O}(\Lambda \tilde{M})$.*

*Proof.* The case $\alpha = 0$ is immediate, so assume $\alpha \neq 0$. Let $\tilde{M}_1, \ldots, \tilde{M}_N \geq 1$ be the template budgets, and write $\tilde{M} = \sum_{i=1}^N \tilde{M}_i$. We may assume $\tilde{M} \geq 16$ by increasing $\tilde{M}_1$ by a constant if necessary. This preserves the allocation constraint and ensures that the iterated logarithms below are positive.

By Eq. (C.5), applied at precision $\varepsilon/2$, the budgets satisfy

$$\sum_{i=1}^N \frac{\alpha_i^2}{\tilde{M}_i^2} \leq \frac{\varepsilon^2}{16c}. \tag{C.26}$$

Hence, by Cauchy–Schwarz,

$$\|\alpha\|_1 \leq \left(\sum_{i=1}^N \frac{\alpha_i^2}{\tilde{M}_i^2}\right)^{1/2} \left(\sum_{i=1}^N \tilde{M}_i^2\right)^{1/2} \tag{C.27}$$

$$\leq \frac{\varepsilon \tilde{M}}{4\sqrt{c}}. \tag{C.28}$$

Since $\|\alpha\|_2 \leq \|\alpha\|_1$, Eq. (C.24) implies

$$\frac{1}{b} = \max\left\{4, \frac{4\|\alpha\|_1}{\varepsilon}, \frac{192\|\alpha\|_2^2}{\varepsilon^2}\right\} \in \mathcal{O}(\tilde{M}^2). \tag{C.29}$$

Consequently, for a fixed universal constant $C_1$,

$$\Lambda = C_1 \log\log(\tilde{M}) \log(\tilde{M}/b) \in \mathcal{O}\Big(\log\log(\tilde{M}) \log(\tilde{M})\Big). \tag{C.30}$$

Set $M_i = \lceil \Lambda \tilde{M}_i \rceil$. Since $M_i \leq \Lambda \tilde{M} + 1$ and $b^{-1} \in \mathcal{O}(\tilde{M}^2)$, we can choose $C_1$ sufficiently large that $\Lambda \geq 1$ and, for every $i$,

$$\max\{1, C_0\} \log\log(M_i) \log(M_i/b) \leq \Lambda. \tag{C.31}$$

Indeed, increasing $C_1$ increases $\Lambda$ linearly, whereas its contribution to the logarithmic factors on the left grows only polylogarithmically. Because $\tilde{M}_i \geq 1$, this choice ensures

$$M_i \geq C_0 \log\log(M_i) \log(M_i/b), \tag{C.32}$$

as required by Lemma C.4.

We first analyze the algorithm without truncation. Apply Lemma C.4 independently for each $i$, using bias parameter $b$ and budget $M_i$. The resulting estimators $\hat{k}_i$ satisfy

$$\left|\mathbb{E}[\hat{k}_i] - k(x, x_i)\right| \leq b \tag{C.33}$$

and

$$\mathrm{Var}[\hat{k}_i] \leq \frac{c\big(\log\log(M_i)\big)^2 \big(\log(M_i/b)\big)^2}{M_i^2} + b \tag{C.34}$$

$$\leq \frac{c}{\tilde{M}_i^2} + b. \tag{C.35}$$

The second inequality follows from $M_i \geq \Lambda \tilde{M}_i$ and the choice of $C_1$.

Define the untruncated prediction estimator

$$\widehat{F} := \sum_{i=1}^N \alpha_i \hat{k}_i. \tag{C.36}$$

Its accumulated bias obeys

$$\left|\mathbb{E}[\widehat{F}] - f_\alpha(x)\right| \leq b\|\alpha\|_1 \leq \frac{\varepsilon}{4}. \tag{C.37}$$

By independence, the allocation constraint, and Eq. (C.24),

$$\mathrm{Var}[\widehat{F}] = \sum_{i=1}^{N} \alpha_i^2 \,\mathrm{Var}[\hat{k}_i] \tag{C.38}$$

$$\leq c \sum_{i=1}^{N} \frac{\alpha_i^2}{\tilde{M}_i^2} + b\|\alpha\|_2^2 \tag{C.39}$$

$$\leq \frac{\varepsilon^2}{16} + \frac{\varepsilon^2}{192} = \frac{13\varepsilon^2}{192}. \tag{C.40}$$

Chebyshev's inequality therefore gives

$$\mathbb{P}\Big(\big|\widehat{F} - \mathbb{E}[\widehat{F}]\big| \geq \frac{\varepsilon}{2}\Big) \leq \frac{13}{48}. \tag{C.41}$$

Thus, with probability at least $35/48$,

$$\big|\widehat{F} - f_\alpha(x)\big| < \frac{\varepsilon}{2} + \frac{\varepsilon}{4} < \varepsilon. \tag{C.42}$$

It remains to bound the query cost. Let $Q$ be the total number of queries used by the untruncated algorithm. Lemma C.4 bounds each estimator's query count in expectation, so

$$\mathbb{E}[Q] \leq \sum_{i=1}^{N} M_i \leq \Lambda\tilde{M} + N \leq 2\Lambda\tilde{M}, \tag{C.43}$$

where we used $\tilde{M} \geq N$ and $\Lambda \geq 1$.

Set

$$Q_{\max} \coloneqq \lceil 48\Lambda\tilde{M} \rceil. \tag{C.44}$$

Truncate the algorithm if it would require more than $Q_{\max}$ queries, returning zero on truncation. Otherwise, return $\widehat{F}$. By Markov's inequality,

$$\mathbb{P}(Q > Q_{\max}) \leq \frac{\mathbb{E}[Q]}{Q_{\max}} \leq \frac{1}{24}. \tag{C.45}$$

Whenever truncation does not occur, the output agrees with the untruncated estimator. Therefore, by a union bound, the truncated algorithm succeeds with probability at least

$$\frac{35}{48} - \frac{1}{24} = \frac{11}{16} > \frac{2}{3}. \tag{C.46}$$

Its worst-case query cost is $Q_{\max} \in \mathcal{O}(\Lambda\tilde{M})$. □

We can finally give the concise results that characterize the query-complexity scaling of both *list-and-sum* strategies based on quantum amplitude estimation.

**Corollary C.6** (Fixed budget, quantum amplitude estimation)**.** *Consider Algorithm C.5 and the problem set-up in Definition 1. To estimate the quantum kernel model prediction in Eq. (1) to additive precision $\varepsilon$ with constant success probability, the runtime scaling of Algorithm C.5 is dominated by the query complexity*

$$M \in \tilde{\mathcal{O}}\left(\frac{N\|\alpha\|_2}{\varepsilon}\right). \tag{C.47}$$

*Proof.* Theorem C.1 with $r = 2$ and the fixed-budget allocation gives $\tilde{M} \in \mathcal{O}(\varepsilon^{-1}N\|\alpha\|_2)$, and Lemma C.5 yields $M \in \mathcal{O}(\Lambda\tilde{M})$. The $\tilde{\mathcal{O}}$ suppresses the polylogarithmic factor $\Lambda$. □

**Algorithm C.5** List-and-sum with fixed budget, via quantum amplitude estimation.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Input:** $\varepsilon > 0$ ▷ Required additive precision.
**Input:** $b$ as in Eq. (C.24) ▷ Bias parameter and variance floor.
**Input:** $M \geq N$, with $M/N$ admissible for Lemma C.4
**Output:** $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ to $\varepsilon$ additive precision. ▷ Inference, Eq. (1).

1: **for** $i \in \{1, \ldots, N\}$ **do**:
2: Reading oracle $O_R(\alpha, i)$: $\alpha_i$
3: Single-input oracle for $U(x)$ and $U^\dagger(x_i)$: estimate $k(x, x_i)$ using $M/N$ queries via Lemma C.4 at bias parameter $b$: $\tilde{k}^{(M/N)}(x, x_i)$ ▷ Bias $\leq b$, variance $\tilde{\mathcal{O}}\left(\frac{N^2}{M^2} + b\right)$.
4: Multiply $\tilde{k}^{(M/N)}(x, x_i)$ with the coefficient $\alpha_i$ ▷ Bias $\leq |\alpha_i| b$, variance $\tilde{\mathcal{O}}\left(\alpha_i^2 \left(\frac{N^2}{M^2} + b\right)\right)$.
5: **end for** ▷ Total number of queries is $M$.
6: **return** $\tilde{f}_\alpha^{(M)}(x) = \sum_{i=1}^N \alpha_i \tilde{k}^{(M/N)}(x, x_i)$
▷ Bias $B = b\|\alpha\|_1 \leq \varepsilon/4$, variance $\tilde{\mathcal{O}}\left(\frac{N^2\|\alpha\|_2^2}{M^2} + b\|\alpha\|_2^2\right)$.
▷ Sufficient: $M \in \tilde{\mathcal{O}}(\varepsilon^{-1} N \|\alpha\|_2)$.

**Corollary C.7** (Adaptive budget, quantum amplitude estimation)**.** *Consider Algorithm C.6 and the problem set-up in Definition 1. To estimate the quantum kernel model prediction in Eq. (1) to additive precision $\varepsilon$ with constant success probability, the runtime scaling of Algorithm C.6 is dominated by the query complexity*

$$M \in \tilde{\mathcal{O}}\left(\frac{\|\alpha\|_{2/3}}{\varepsilon}\right). \tag{C.48}$$

*The optimal allocation of queries is*

$$M_i \in \tilde{\mathcal{O}}\left(\frac{|\alpha_i|^{2/3}\|\alpha\|_{2/3}^{1/3}}{\varepsilon}\right). \tag{C.49}$$

*Proof.* Theorem C.1 with $r = 2$ and the optimal query allocation gives $\tilde{M}_i \in \mathcal{O}(\varepsilon^{-1}|\alpha_i|^{2/3}\|\alpha\|_{2/3}^{1/3})$ and hence $\tilde{M} \in \mathcal{O}(\varepsilon^{-1}\|\alpha\|_{2/3})$, and Lemma C.5 inflates each budget to $M_i = \lceil \Lambda \tilde{M}_i \rceil$, for a total of $M \in \mathcal{O}(\Lambda \tilde{M})$. The $\tilde{\mathcal{O}}$ suppresses the polylogarithmic factor $\Lambda$. □

### C.2 All-at-once

We next provide the algorithms and analysis for the *all-at-once* approaches. Unlike for the case of list-and-sum, there is no single template from which all results follow, but rather we present each scenario individually. Several relevant definitions are in Section 3.1 in the main text.

#### C.2.1 Via sampling

We now analyze the query complexity of estimating the *all-at-once* observable $\mathcal{M}(\alpha, S)$ introduced in Theorem 2 directly via sampling, using the circuit from Algorithm B.1. The resulting algorithm is specified in Algorithm C.7 and the resulting runtime in Theorem C.8.

**Theorem C.8** (Query complexity of all-at-once via sampling)**.** *Consider Algorithm C.7 and the problem set-up in Definition 1 for estimating the kernel model prediction in Eq. (1) to additive precision $\varepsilon$ with constant success probability. The algorithm uses a total number of queries*

$$M \in \mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right). \tag{C.50}$$

**Algorithm C.6** List-and-sum with adaptive budget, via quantum amplitude estimation.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Input:** $\varepsilon > 0$ ▷ Required additive precision.
**Input:** $b$ as in Eq. (C.24) ▷ Bias parameter and variance floor.
**Input:** $M_1, \ldots, M_N$ admissible for Lemma C.4, $M = \sum_{i=1}^N M_i \geq N$ ▷ Per-term query budgets.
**Output:** $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ to $\varepsilon$ additive precision. ▷ Inference, Eq. (1).

1: **for** $i \in \{1, \ldots, N\}$ **do**:
2: Reading oracle $O_R(\alpha, i)$: $\alpha_i$.
3: Single-input oracle for $U(x)$ and $U^\dagger(x_i)$: estimate $k(x, x_i)$ using $M_i$ queries via Lemma C.4 at bias parameter $b$: $\tilde{k}^{(M_i)}(x, x_i)$ ▷ Bias $\leq b$, variance $\tilde{\mathcal{O}}\left(\frac{1}{M_i^2} + b\right)$.
4: Multiply $\tilde{k}^{(M_i)}(x, x_i)$ with the coefficient $\alpha_i$ ▷ Bias $\leq |\alpha_i| b$, variance $\tilde{\mathcal{O}}\left(\alpha_i^2 \left(\frac{1}{M_i^2} + b\right)\right)$.
5: **end for** ▷ Total number of queries is $M = \sum_{i=1}^N M_i$.
6: **return** $\tilde{f}_\alpha^{(M)}(x) = \sum_{i=1}^N \alpha_i \tilde{k}^{(M_i)}(x, x_i)$ ▷
▷ Bias $B = b\|\alpha\|_1 \leq \varepsilon/4$, variance $\tilde{\mathcal{O}}\left(\sum_{i=1}^N \frac{\alpha_i^2}{M_i^2} + b\|\alpha\|_2^2\right)$.
▷ Sufficient: $M \in \tilde{\mathcal{O}}(\varepsilon^{-1}\|\alpha\|_{2/3})$.

**Algorithm C.7** All-at-once via sampling.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Input:** $\varepsilon > 0$ ▷ Required additive precision.
**Output:** $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ to $\varepsilon$ additive precision. ▷ Inference, Eq. (1).

1: Single-input oracle $U(x)$: $|\phi(x)\rangle$
2: Estimate $\langle\phi(x)|\tilde{\mathcal{M}}(\alpha, S)|\phi(x)\rangle$ via sampling (see Algorithm B.1), using $M$ shots: $\hat{f}_\alpha^{(M)}(x)$ ▷ Achieve precision $\mathcal{O}(\|\tilde{\mathcal{M}}(\alpha, S)\|_\infty/\sqrt{M})$.
3: **return** $\hat{f}_\alpha^{(M)}(x)$ ▷ Total additive precision is in $\mathcal{O}\left(\|\alpha\|_1/\sqrt{M}\right)$.

*Proof.* Following Algorithms C.7 and B.1, as well as Lemma B.1, using $M$ shots produces an estimate $\hat{f}_\alpha^{(M)}(x)$ such that, with probability at least $2/3$,

$$\left|\hat{f}_\alpha^{(M)}(x) - f_\alpha(x)\right| \in \mathcal{O}\left(\frac{\|\alpha\|_1}{\sqrt{M}}\right). \tag{C.51}$$

To achieve $\varepsilon$ additive precision, it is thus sufficient to pick

$$M \in \mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right). \tag{C.52}$$

Since each shot constitutes three oracle queries (one execution of the circuit in Algorithm B.1), the total query complexity is $3M \in \mathcal{O}\left(\|\alpha\|_1^2/\varepsilon^2\right)$. □

### C.2.2 Via quantum amplitude estimation

We next provide a detailed description of the algorithm used to implement the amplitude-encoding unitary $V$ from Definition 7, with accompanying analysis in Lemma 3. Below we analyze the query complexity of the *all-at-once* algorithm using quantum amplitude estimation. The procedure is specified in Algorithm C.9 and the resulting runtime in Theorem C.9.

**Lemma 3** (Single-observable encoded in amplitude)**.** *Under the conditions of Theorem 2, there exists a quantum circuit using $n + \lceil \log_2(N) \rceil + 3$ qubits which implements the amplitude-encoding unitary in Definition 7.*

*Proof.* We refer to Fig. 4 as visual proof. The quantum circuit follows the principles from Lemma B.1, though replacing the observables $|0\rangle\langle 0|_{\text{data}}$ and $Z_{\text{sign}}$ with controlled bitflips on auxiliary qubits.

**Algorithm C.8** Amplitude-encoding unitary.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Output:** Amplitude-encoding unitary $V$ ▷ From Definition 7.

1: Initialize $|0\rangle_{\text{data}}|0\rangle_{\text{idx}}|0\rangle_{\text{sign}}|0,0\rangle_{\text{QP}}$. ▷ $n + \lceil \log_2(N) \rceil + 3$ qubits.

2: Single-input oracle $U(x)_{\text{data}}$: $|\tilde{\phi}(x)\rangle = |\phi(x)\rangle_{\text{data}}|0\rangle_{\text{idx}}|0\rangle_{\text{sign}}$ ▷ $G$ gates.

3: Quantum-state coefficient oracle $W(\alpha)_{\text{idx,sign}}$: ▷ $N$ gates.

$$|\phi(x)\rangle_{\text{data}}|0\rangle_{\text{idx}}|0\rangle_{\text{sign}} \mapsto \sum_{i=1}^{N} \sqrt{p_i}|\phi(x)\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}$$

4: Training-set oracle $O^\dagger(S)$: ▷ $\mathcal{O}(N(G + \log N))$ gates.

$$\sum_{i=1}^{N} \sqrt{p_i}|\phi(x)\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}} \mapsto \sum_{i=1}^{N} \sqrt{p_i}U^\dagger(x_i)|\phi(x)\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}$$

5: Coherently flag $|0\rangle_{\text{data}}$ on the $|0,0\rangle_{\text{QP}}$ branch with a multi-control $X$: ▷ $\mathcal{O}(n)$ elementary gates

$$\begin{aligned}
&\sum_{i=1}^{N} \sqrt{p_i}U^\dagger(x_i)|\phi(x)\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}|0,0\rangle_{\text{QP}} \\
&\mapsto \sum_{i=1}^{N} \sqrt{p_i}|0\rangle\langle 0|U^\dagger(x_i)|\phi(x)\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}|0,0\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}} \\
&= \sum_{i=1}^{N} \sqrt{p_i}\langle\phi(x_i)|\phi(x)\rangle|0\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}|0,0\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}}
\end{aligned}$$

6: Separate the positive and negative terms in the sum with a CNOT ▷ $\mathcal{O}(1)$ elementary gates.

$$\begin{aligned}
&\sum_{i=1}^{N} \sqrt{p_i}\langle\phi(x_i)|\phi(x)\rangle|0\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}|0,0\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}} \\
&\mapsto \sum_{i=1}^{N} \sqrt{p_i}\langle\phi(x_i)|\phi(x)\rangle|0\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}|0,\frac{1-s_i}{2}\rangle_{\text{QP}} \\
&\qquad + \cdots |1,0\rangle_{\text{QP}} + \cdots |1,1\rangle_{\text{QP}}
\end{aligned}$$

7: **return** $\sum_{i=1}^{N} \sqrt{p_i}\langle\phi(x_i)|\phi(x)\rangle|0\rangle_{\text{data}}|i\rangle_{\text{idx}}|s_i\rangle_{\text{sign}}|0,\frac{1-s_i}{2}\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}} + \cdots |1,1\rangle_{\text{QP}}$
▷ Number of gates $\mathcal{O}(N(G + \log N) + n)$.

We rigorously prove the statement, by following the steps of Algorithm C.8. Using the same notation as above for the different oracles, we now introduce two controlled operations as

$$C_0X_{\text{data, QP,trash}} = |0\rangle\langle 0|_{\text{data}} \otimes \mathbb{I}_{\text{QP,trash}} + (\mathbb{I} - |0\rangle\langle 0|)_{\text{data}} \otimes X_{\text{QP,trash}}, \tag{C.53}$$

$$\text{CNOT}_{\text{sign, QP,}\pm} = |1\rangle\langle 1|_{\text{sign}} \otimes \mathbb{I}_{\text{QP,}\pm} + |-1\rangle\langle -1|_{\text{sign}} \otimes X_{\text{QP,}\pm}. \tag{C.54}$$

Recall we defined the sign register as the eigenbasis of the Pauli $Z$ operator $Z|s_i\rangle = s_i|s_i\rangle$. We claim that $V$ is of the form

$$V = \text{CNOT}_{\text{sign, QP,}\pm} \cdot C_0X_{\text{data, QP,trash}} \cdot O^\dagger(S)_{\text{data,idx,sign}} \cdot W(\alpha)_{\text{idx,sign}} \cdot U(x)_{\text{data}}, \tag{C.55}$$

as depicted in Fig. 4. We confirm this by applying a sequence of matrix-vector operations starting from the all-0 state, by performing the calculation

$$
\begin{aligned}
V|0\rangle &= \mathrm{CNOT}_{\text{sign, QP},\pm} \cdot C_0 X_{\text{data, QP,trash}} \cdot O^\dagger(S)_{\text{data,idx,sign}} \cdot W(\alpha)_{\text{idx,sign}} \cdot U(x)_{\text{data}} |0\rangle_{\text{data,idx,sign,QP}} \qquad \text{(C.56)} \\
&= \mathrm{CNOT}_{\text{sign, QP},\pm} \cdot C_0 X_{\text{data, QP,trash}} \cdot O^\dagger(S)_{\text{data,idx,sign}} \left( \sum_{i=1}^{N} \sqrt{p_i} |\phi(x)\rangle_{\text{data}} |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0,0\rangle_{\text{QP}} \right) \\
&= \mathrm{CNOT}_{\text{sign, QP},\pm} \cdot C_0 X_{\text{data, QP,trash}} \left( \sum_{i=1}^{N} \sqrt{p_i} \left( U^\dagger(x_i) |\phi(x)\rangle_{\text{data}} \right) |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0,0\rangle_{\text{QP}} \right) \\
&= \mathrm{CNOT}_{\text{sign, QP},\pm} \left( |0\rangle\langle 0|_{\text{data}} \otimes \mathbb{I}_{\text{QP,trash}} + \left( \mathbb{I} - |0\rangle\langle 0| \right)_{\text{data}} \otimes X_{\text{QP,trash}} \right) \\
&\quad \times \left( \sum_{i=1}^{N} \sqrt{p_i} \left( U^\dagger(x_i) |\phi(x)\rangle_{\text{data}} \right) |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0,0\rangle_{\text{QP}} \right) \\
&= \mathrm{CNOT}_{\text{sign, QP},\pm} \left( |0\rangle\langle 0|_{\text{data}} \otimes \mathbb{I}_{\text{QP,trash}} \left( \sum_{i=1}^{N} \sqrt{p_i} \left( U^\dagger(x_i) |\phi(x)\rangle_{\text{data}} \right) |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0,0\rangle_{\text{QP}} \right) \right. \\
&\quad \left. + \left( \mathbb{I} - |0\rangle\langle 0| \right)_{\text{data}} \otimes X_{\text{QP,trash}} \left( \sum_{i=1}^{N} \sqrt{p_i} \left( U^\dagger(x_i) |\phi(x)\rangle_{\text{data}} \right) |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0,0\rangle_{\text{QP}} \right) \right) \\
&= \mathrm{CNOT}_{\text{sign, QP},\pm} \left( \left( \sum_{i=1}^{N} \sqrt{p_i} \underbrace{\langle 0|U^\dagger(x_i)}_{\langle \phi(x_i)|} |\phi(x)\rangle |0\rangle_{\text{data}} |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} \right) |0,0\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}} \right) \\
&= \left( |1\rangle\langle 1|_{\text{sign}} \otimes \mathbb{I}_{\text{QP},\pm} + |-1\rangle\langle -1|_{\text{sign}} \otimes X_{\text{QP},\pm} \right) \\
&\quad \times \left( \left( \sum_{i=1}^{N} \sqrt{p_i} \underbrace{\langle 0|U^\dagger(x_i)}_{\langle \phi(x_i)|} |\phi(x)\rangle |0\rangle_{\text{data}} |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} \right) |0,0\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}} \right) \\
&= \sum_{i=1}^{N} \sqrt{p_i} \langle \phi(x_i)|\phi(x)\rangle |0\rangle_{\text{data}} |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0, \frac{1-s_i}{2}\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}} + \cdots |1,1\rangle_{\text{QP}}.
\end{aligned}
$$

According to our convention for the definition of the sign register, we have that $\frac{1-s_i}{2}$ is 0 if $s_i = 1$, and 1 in the case $s_i = -1$. For us to confirm that the last line corresponds to the r.h.s. of Eq. (6), we must only compute its $|0,0\rangle_{\text{QP}}$ and $|0,1\rangle_{\text{QP}}$ components. We denote the projectors as $P_+ = \mathbb{I}_{\text{data,idx,sign}} \otimes |0,0\rangle\langle 0,0|_{\text{QP}}$, and $P_- = \mathbb{I}_{\text{data,idx,sign}} \otimes |0,1\rangle\langle 0,1|_{\text{QP}}$. Then, we find

$$
\begin{aligned}
&P_+ \left( \sum_{i=1}^{N} \sqrt{p_i} \langle \phi(x_i)|\phi(x)\rangle |0\rangle_{\text{data}} |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0, \frac{1-s_i}{2}\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}} + \cdots |1,1\rangle_{\text{QP}} \right) \qquad \text{(C.57)} \\
&= \sum_{i=1}^{N} \delta_{s_i,1} \sqrt{p_i} \langle \phi(x_i)|\phi(x)\rangle |0\rangle_{\text{data}} |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0,0\rangle_{\text{QP}} = |\tilde{\psi}_+\rangle_{\text{data,idx,sign}}, \\
&P_- \left( \sum_{i=1}^{N} \sqrt{p_i} \langle \phi(x_i)|\phi(x)\rangle |0\rangle_{\text{data}} |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0, \frac{1-s_i}{2}\rangle_{\text{QP}} + \cdots |1,0\rangle_{\text{QP}} + \cdots |1,1\rangle_{\text{QP}} \right) \qquad \text{(C.58)} \\
&= \sum_{i=1}^{N} \delta_{s_i,-1} \sqrt{p_i} \langle \phi(x_i)|\phi(x)\rangle |0\rangle_{\text{data}} |i\rangle_{\text{idx}} |s_i\rangle_{\text{sign}} |0,1\rangle_{\text{QP}} = |\tilde{\psi}_-\rangle_{\text{data,idx,sign}},
\end{aligned}
$$

**Algorithm C.9** All-at-once via quantum amplitude estimation.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Input:** $\varepsilon > 0$ ▷ Required additive precision.
**Output:** $f_\alpha(x) = \sum_{i=1}^N \alpha_i k(x, x_i)$ to $\varepsilon$ additive precision ▷ Inference, Eq. (1).

1: Initialize $|0\rangle_{\text{data}}|0\rangle_{\text{idx}}|0\rangle_{\text{sign}}|0,0\rangle_{\text{QP}}$. ▷ $n + \lceil \log_2(N) \rceil + 3$ qubits.
2: $\tilde{a}_+^{(M)} \leftarrow M$-query-quantum amplitude estimation on $V$ from Algorithm C.8 with $|\psi_{\text{good}}\rangle = |\psi_+\rangle$ ▷ Estimate $\sqrt{f_+}$ to precision $\mathcal{O}(1/M)$.
3: $\tilde{a}_-^{(M)} \leftarrow M$-query-quantum amplitude estimation on $V$ from Algorithm C.8 with $|\psi_{\text{good}}\rangle = |\psi_-\rangle$ ▷ Estimate $\sqrt{f_-}$ to precision $\mathcal{O}(1/M)$.
4: Square estimates $\tilde{f}_\pm^{(M)} = \left(\tilde{a}_\pm^{(M)}\right)^2$. ▷ Estimate $f_\pm$ to precision $\mathcal{O}(1/M)$.
5: Call the norm oracle $O_N(\alpha)$ to obtain $\|\alpha\|_1$.
6: **return** $\tilde{f}_\alpha^{(2M)}(x) = \|\alpha\|_1 \left(\tilde{f}_+^{(M)} - \tilde{f}_-^{(M)}\right)$
▷ Combined additive precision in $\mathcal{O}\left(\frac{\|\alpha\|_1}{M}\right)$. The total number of queries is $2M$.

(note that $|\tilde{\psi}_\pm\rangle$ are unnormalized) we also compute the measurement probability for each component

$$\begin{aligned}
&\langle\tilde{\psi}_+|P_+|\tilde{\psi}_+\rangle \qquad \text{(C.59)}\\
&= \sum_{i,j=1}^N \delta_{s_i,1}\delta_{s_j,1}\sqrt{p_i p_j}\langle\phi(x_i)|\phi(x)\rangle\langle\phi(x)|\phi(x_j)\rangle\langle 0|0\rangle_{\text{data}} \underbrace{\langle j|i\rangle_{\text{idx}}}_{\delta_{i,j}}\langle s_j|s_i\rangle_{\text{sign}}\langle 0,0|0,0\rangle_{\text{QP}}\\
&= \sum_{i=1}^N \delta_{s_i,1} p_i \underbrace{|\langle\phi(x_i)|\phi(x)\rangle|^2}_{k(x,x_i)}.
\end{aligned}$$

We notice that the last line corresponds exactly to the definition of $f_+$ from Definition 3. The same is true for $f_-$:

$$\begin{aligned}
&\langle\tilde{\psi}_-|P_-|\tilde{\psi}_-\rangle \qquad \text{(C.60)}\\
&= \sum_{i,j=1}^N \delta_{s_i,-1}\delta_{s_j,-1}\sqrt{p_i p_j}\langle\phi(x_i)|\phi(x)\rangle\langle\phi(x)|\phi(x_j)\rangle\langle 0|0\rangle_{\text{data}} \underbrace{\langle j|i\rangle_{\text{idx}}}_{\delta_{i,j}}\langle s_j|s_i\rangle_{\text{sign}}\langle 01|01\rangle_{\text{QP}}\\
&= \sum_{i=1}^N \delta_{s_i,-1} p_i \underbrace{|\langle\phi(x_i)|\phi(x)\rangle|^2}_{k(x,x_i)} = f_-.
\end{aligned}$$

This completes the proof. □

**Remark** Under the assumption that multi-controlled $X$ gates with $n$ controls require $\mathcal{O}(n)$ elementary gates, Algorithm C.8 uses $\mathcal{O}(N(G + \log N) + n)$ gates and 2 auxiliary qubits.

Our main contribution is the following result, which underpins the complexity of inference in quantum kernels via quantum amplitude estimation.

**Theorem C.9** (Query complexity of all-at-once via quantum amplitude estimation)**.** *Consider Algorithm C.9. To estimate the quantum kernel model prediction in Eq.* (1) *to additive precision $\varepsilon$ with constant success probability, it suffices to choose $M = \Theta(\|\alpha\|_1/\varepsilon)$. The algorithm uses $2M$ applications of $V$ and $V^\dagger$ in total.*

*Proof.* Algorithm C.9 evaluates Eq. (1) by using quantum amplitude estimation twice as a subroutine, once for the positive terms in the sum $f_+$, and once for the negative terms $f_-$. Recall we defined these terms as $f_\alpha(x) = \|\alpha\|_1 (f_+ - f_-)$, so that $f_\pm \in [0,1]$. Each of the two quantum amplitude estimation subroutines uses $M$ applications of $V$ and $V^\dagger$, for a total query count of $2M$. By

Theorem A.3, $M$ queries yield additive error $\mathcal{O}(1/M)$ on each of the amplitudes $\sqrt{f_\pm}$, so that

$$|\tilde{a}_\pm^{(M)} - \sqrt{f_\pm}| \in \mathcal{O}\left(\frac{1}{M}\right). \tag{C.61}$$

Here we choose the failure probability $\delta = 1/6$ in Theorem A.3 for each subroutine, with the constant confidence overhead included in the budget $M$ of each subroutine. By the union bound, both amplitude-error bounds hold simultaneously with probability at least $2/3$. The remainder of the argument is on this joint success event. We recover estimates of $f_\pm$ by squaring: $\hat{f}_\pm^{(M)} := \left(\tilde{a}_\pm^{(M)}\right)^2$. Since $\sqrt{f_\pm} \in [0,1]$ (as $f_\pm$ is a convex combination of kernel values in $[0,1]$) and $\tilde{a}_\pm^{(M)} \in [0,1]$, we have that the factor $|\tilde{a}_\pm^{(M)} + \sqrt{f_\pm}|$ is upper bounded by 2. Therefore,

$$|\hat{f}_\pm^{(M)} - f_\pm| = |\tilde{a}_\pm^{(M)} - \sqrt{f_\pm}| \cdot \underbrace{|\tilde{a}_\pm^{(M)} + \sqrt{f_\pm}|}_{\leq 2} \leq 2|\tilde{a}_\pm^{(M)} - \sqrt{f_\pm}| \in \mathcal{O}\left(\frac{1}{M}\right). \tag{C.62}$$

Taking the difference and applying the triangle inequality, we get

$$\left|\left(\hat{f}_+^{(M)} - \hat{f}_-^{(M)}\right) - (f_+ - f_-)\right| \leq |\hat{f}_+^{(M)} - f_+| + |\hat{f}_-^{(M)} - f_-| \in \mathcal{O}\left(\frac{1}{M}\right). \tag{C.63}$$

Multiplying by $\|\alpha\|_1$ to recover an estimate $\hat{f}_\alpha^{(2M)}(x) = \|\alpha\|_1 \left(\hat{f}_+^{(M)} - \hat{f}_-^{(M)}\right)$ for our target function $f_\alpha$, we obtain

$$|\hat{f}_\alpha^{(2M)}(x) - f_\alpha(x)| \in \mathcal{O}\left(\frac{\|\alpha\|_1}{M}\right). \tag{C.64}$$

To achieve a final precision $\varepsilon$, we, therefore, set

$$M \in \Theta\left(\frac{\|\alpha\|_1}{\varepsilon}\right), \tag{C.65}$$

which gives a total of $2M \in \Theta(\|\alpha\|_1/\varepsilon)$ applications of $V$ and $V^\dagger$. □

**Remark** Algorithm C.9 does not require reading access to $\alpha$ as the positive and negative parts are separated implicitly inside the algorithm.

# D Matching lower bound: statements and proofs

This appendix contains the formal statement and proof of the query lower bound announced in Section 3.4, together with the remarks delineating its scope. The bound holds for any algorithm equipped with the access model specified in Definition 4. The mechanism is to place the hardness inside the feature-map oracle, so that the only way to learn $f_\alpha(x)$ is to query $U(\cdot)$, and to reduce to the problem of approximately counting the ones of a Boolean function rather than to amplitude estimation.

**Theorem D.1** (Query lower bound)**.** *Let $\varepsilon, A > 0$ and let $n \in \mathbb{N}$ be such that*

$$64 \leq \frac{A}{\varepsilon} \leq 2^{n-1}. \tag{D.1}$$

*Then there exists a family of oracle instances satisfying Definitions 1 and 4, with $\|\alpha\|_1 = A$, on which any quantum algorithm that estimates $f_\alpha(x)$ to additive precision $\varepsilon$ with constant success probability must make*

$$\Omega\left(\frac{\|\alpha\|_1}{\varepsilon}\right) \tag{D.2}$$

*queries to the single-input oracle. Here, each application of $U(x')$, $U^\dagger(x')$, or a controlled version thereof, at any input $x'$, counts as one query; for $N = 1$ this includes the training-set oracle $O^{(\dagger)}(S) = U^{(\dagger)}(x_1)$, where the notation $O^{(\dagger)}$ covers the oracle and its inverse; all remaining oracles of Definition 4 may be called free of charge.*

*Proof.* Throughout, write $D := 2^n$, so that Eq. (D.1) reads $64 \le \frac{A}{\varepsilon} \le \frac{D}{2}$; in particular $D \ge 128$. Without loss of generality, the success probability is at least $2/3$.

*Step 1: The hard family.* Set $N = 1$ and $\alpha_1 = A$, so that $\|\alpha\|_1 = A$ and $f_\alpha(x) = Ak(x, x_1)$, and fix any query point $x \neq x_1$. For $g\colon \{0,1\}^n \to \{0,1\}$, let $O_g|z\rangle = (-1)^{g(z)}|z\rangle$ denote the phase oracle of $g$, and define the family

$$U_g(x_1) = O_g H^{\otimes n}, \qquad U_g(x') = H^{\otimes n} \quad \text{for all } x' \neq x_1, \tag{D.3}$$

in particular at the query point $x$. Each $U_g$ is a valid input-dependent unitary quantum feature map in the sense of Definition 1: writing $g^{-1}(1) = \{z_1, \dots, z_t\}$, we have $O_g = \prod_{\ell=1}^{t} (\mathbb{I} - 2|z_\ell\rangle\langle z_\ell|)$, and each such reflection is exactly realizable with $\mathcal{O}(n)$ gates from a fixed finite gate set containing the Toffoli gate, using one auxiliary qubit [Nielsen and Chuang, 2000]; padding with identity gates, the gate count can be taken as $G \in \mathcal{O}(2^n n)$, uniformly over the family. The members of the family differ only at the single training input $x_1$, and since Definition 4 grants only black-box access to the map $x' \mapsto U(x')$, these are legitimate oracle instances for every $g$. Uniformity of $G$ across the family is essential, since $G$ is visible to the algorithm; padding makes all instances indistinguishable in gate count.

*Step 2: Kernel values.* Write $t := |g^{-1}(1)|$ and $w := t/D$. Then

$$\langle\psi(x)|\psi_g(x_1)\rangle = \langle 0|H^{\otimes n} O_g H^{\otimes n}|0\rangle = \frac{1}{D}\sum_z (-1)^{g(z)} = 1 - 2w, \tag{D.4}$$

and hence

$$k(x, x_1) = (1-2w)^2, \qquad f_\alpha(x) = A(1-2w)^2. \tag{D.5}$$

We restrict the family to those $g$ with $w \in [1/8, 3/8]$. On this window, the map $\varphi(w) := (1-2w)^2$ is strictly decreasing and maps $[1/8, 3/8]$ bijectively onto $[1/16, 9/16]$, and its inverse $\varphi^{-1}(k) = (1 - \sqrt{k})/2$ satisfies $|(\varphi^{-1})'(k)| = 1/(4\sqrt{k}) \le 1$ for $k \ge 1/16$, i.e., $\varphi^{-1}$ is 1-Lipschitz on the window. In particular, all kernel values of the restricted family lie in $[1/16, 9/16]$, bounded away from 0 and 1.

*Step 3: From estimating $f_\alpha(x)$ to distinguishing two Hamming weights.* Suppose an algorithm outputs $\hat{f}$ with $|\hat{f} - f_\alpha(x)| \le \varepsilon$ with probability at least $2/3$ on every instance of the restricted family. Post-process $\hat{f}$ as follows: clip $\hat{k} := \min\{\max\{\hat{f}/A, 1/16\}, 9/16\}$, invert $\hat{w} := \varphi^{-1}(\hat{k})$, and round to $\hat{t}$, the nearest integer to $D\hat{w}$. Whenever $|\hat{f} - f_\alpha(x)| \le \varepsilon$: clipping cannot increase the distance to $k(x, x_1) \in [1/16, 9/16]$, so $|\hat{k} - k(x, x_1)| \le \varepsilon/A$; the 1-Lipschitz property gives $|\hat{w} - w| \le \varepsilon/A$; and rounding gives $|\hat{t} - t| \le \frac{\varepsilon D}{A} + \frac{1}{2}$. Hence, with probability at least $2/3$, the algorithm computes $t$ to additive accuracy

$$\Delta := \left\lceil \frac{\varepsilon D}{A} + \frac{1}{2} \right\rceil, \qquad 1 \le \Delta \le 2\frac{\varepsilon D}{A}, \tag{D.6}$$

where the upper bound uses $\varepsilon D/A \ge 2$, guaranteed by the right end of Eq. (D.1). Now set $t_0 := \lceil D/4 \rceil$ and $t_1 := t_0 + 2\Delta + 1$. Both weights lie in the window: $t_0 \ge D/8$ is clear, and using $\Delta \le 2\varepsilon D/A \le D/32$ from the left end of Eq. (D.1),

$$t_1 \le \left(\frac{D}{4} + 1\right) + \left(\frac{D}{16} + 1\right) \le \frac{3D}{8} \qquad \text{for } D \ge 32. \tag{D.7}$$

A $\Delta$-accurate estimate distinguishes the two: given $\hat{t}$, declare weight $t_0$ if $\hat{t} \le t_0 + \Delta$ and weight $t_1$ otherwise; since $t_1 - t_0 = 2\Delta + 1$, this decision is correct with probability at least $2/3$ on every instance of either weight. The estimation algorithm therefore yields a bounded-error algorithm that, promised $|g^{-1}(1)| \in \{t_0, t_1\}$, decides which case holds, at the same query cost.

*Step 4: The counting lower bound.* For integers $0 \le \ell \neq \ell' \le D$, let $f_{\ell,\ell'}$ denote the partial Boolean function on $\{0,1\}^D$ that evaluates to 1 on inputs of Hamming weight $\ell$, to 0 on inputs of Hamming weight $\ell'$, and is undefined otherwise. Identifying a $D$-bit input string with the truth table of a function $g\colon \{0,1\}^n \to \{0,1\}$, Nayak and Wu [1999] prove that any quantum algorithm which computes $f_{\ell,\ell'}$ with bounded error, given access to the input via the standard bit oracle

$$B_g\colon |z\rangle|b\rangle \mapsto |z\rangle|b \oplus g(z)\rangle, \tag{D.8}$$

must make

$$\Omega\left(\sqrt{\frac{D}{\Delta'}}+\frac{\sqrt{m(D-m)}}{\Delta'}\right) \tag{D.9}$$

queries, where $\Delta' = |\ell - \ell'|$ and $m \in \{\ell, \ell'\}$ is the weight farther from $D/2$ [Nayak and Wu, 1999, Corollary 1.2]. The distinguishing task of Step 3 is exactly the computation of $f_{t_1,t_0}$, with $\Delta' = t_1 - t_0 = 2\Delta + 1 \le 5\Delta$ and, since $t_0 < t_1 \le 3D/8 < D/2$, with $m = t_0 = \lceil D/4 \rceil$. There, $\sqrt{t_0(D-t_0)} \ge \sqrt{\frac{D}{4}\cdot\frac{D}{2}} \ge \frac{D}{3}$; since Eq. (D.9) is a sum of non-negative terms, it is bounded below by its second term alone, which is at least $D/(3\Delta') \ge D/(15\Delta)$. Hence the distinguishing task, and by Step 3 the estimation task, requires

$$\Omega\left(\frac{D}{\Delta}\right) = \Omega\left(\frac{A}{\varepsilon}\right) \tag{D.10}$$

queries to $B_g$, where the last equality uses $\Delta \le 2\varepsilon D/A$.

*Step 5: Back to the single-input oracle.* It remains to check that the single-input oracle is the only carrier of information about $g$, and that it carries it at unit rate. The classical oracles $O_R, O_N, O_C$ and the coefficient quantum-state oracle $W(\alpha)$ act on the fixed coefficient vector $\alpha = (A)$ and are therefore $g$-independent, as is $U_g(x') = H^{\otimes n}$ for every $x' \ne x_1$; for $N = 1$ the training-set oracle is $O^\dagger(S) = U_g^\dagger(x_1)$, i.e. a single-input query by the convention of the theorem. The remaining calls are $U_g(x_1) = O_g H^{\otimes n}$, its inverse $U_g^\dagger(x_1) = H^{\otimes n} O_g$ (using $O_g^\dagger = O_g$), and controlled versions of either; we show that each can be simulated with exactly one query to $B_g$, together with $\mathcal{O}(n)$ $g$-independent gates. For the uncontrolled calls this is standard phase kickback: on an auxiliary qubit prepared in $|-\rangle$, one query to $B_g$ acts as $B_g|z\rangle|-\rangle = (-1)^{g(z)}|z\rangle|-\rangle$ and thereby implements $O_g$ on the data register; the preparation and uncomputation of the auxiliary qubit are $g$-independent. For the controlled calls, write $c$-$W$ for the version of a unitary $W$ controlled on a qubit $c$, and decompose $c$-$U_g(x_1) = (c\text{-}O_g)(c\text{-}H^{\otimes n})$ and $c$-$U_g^\dagger(x_1) = (c\text{-}H^{\otimes n})(c\text{-}O_g)$, where $c$-$H^{\otimes n}$ is $g$-independent and costs $\mathcal{O}(n)$ gates. The controlled phase oracle $c$-$O_g$ is realized with one query to $B_g$ by the following gadget on an auxiliary qubit $a$ initialized in $|0\rangle$: apply $\mathrm{CNOT}_{c\to a}$ followed by a Hadamard gate on $a$, so that $a$ carries $|+\rangle$ on the control-$|0\rangle$ branch and $|-\rangle$ on the control-$|1\rangle$ branch; query $B_g$ with target $a$, which, by

$$B_g|z\rangle|+\rangle = |z\rangle|+\rangle, \qquad B_g|z\rangle|-\rangle = (-1)^{g(z)}|z\rangle|-\rangle, \tag{D.11}$$

acts as the identity on the first branch and imprints the phase on the second; then undo the Hadamard gate and the $\mathrm{CNOT}_{c\to a}$. The auxiliary qubit returns to $|0\rangle$ coherently on both branches, so the net action is

$$|c\rangle|z\rangle|0\rangle_a \mapsto (-1)^{cg(z)}|c\rangle|z\rangle|0\rangle_a, \tag{D.12}$$

i.e. exactly $c$-$O_g$, at the cost of one query and $\mathcal{O}(1)$ $g$-independent gates; multiply controlled versions are handled identically, replacing $\mathrm{CNOT}_{c\to a}$ by the corresponding multi-controlled bit flip. Hence an algorithm making $q$ queries to the single-input oracle induces a bounded-error algorithm for the distinguishing task of Step 3 making $q$ queries to $B_g$, and $q \in \Omega(A/\varepsilon) = \Omega(\|\alpha\|_1/\varepsilon)$ follows from Step 4. □

**Remark** Neither end of the regime in Eq. (D.1) weakens the optimality conclusion. Some cutoff at the lower end is necessary: for $A \le \varepsilon$ the constant estimate 0 is already $\varepsilon$-accurate, since $|f_\alpha(x)| \le \|\alpha\|_1$, and for $A/\varepsilon \le 64$ Theorem C.9 solves the task with $\mathcal{O}(1)$ queries, so no super-constant bound can hold there. The upper end, in turn, is an artifact of the hard family used in the proof, whose attainable values of $f_\alpha(x)$ are $\Theta(A/2^n)$-spaced. Since the theorem asserts the existence of hard instances and the number of feature qubits $n$ is a parameter of the construction, hard instances exist for every ratio $A/\varepsilon \ge 64$ upon choosing $n$ with $2^{n-1} \ge A/\varepsilon$. Whether the full-access-model bound persists for fixed $n$ beyond the upper cutoff is a question we do not pursue.

**Corollary D.2** (Optimality of Algorithm C.9)**.** *Consider any algorithm that accesses the problem instance only through applications of the amplitude-encoding unitary $V$ from Definition 7, of its inverse $V^\dagger$, and of controlled versions thereof, as is the case for Algorithm C.9 and for any quantum-amplitude-estimation routine run on $V$. In the regime of Eq. (D.1), if such an algorithm estimates $f_\alpha(x)$ to additive precision $\varepsilon$ with constant success probability, it must apply $V$, $V^\dagger$, or controlled versions thereof $\Omega\left(\|\alpha\|_1/\varepsilon\right)$ times. Together with the matching upper bound of Theorem C.9, Algorithm C.9 is query-optimal up to constant factors.*

*Proof.* Instantiate the hard family of Theorem D.1. By construction (Algorithm C.8), each application of $V$ makes one call to the single-input oracle at the query point $x$ and one call to the training-set oracle $O^\dagger(S)$, which for $N = 1$ is a single call to $U^\dagger(x_1)$; all further operations inside $V$ are calls to $W(\alpha)$ or fixed $g$-independent gates. Each application of $V^\dagger$ likewise makes one call to $U^\dagger(x)$ and one to $O(S) = U(x_1)$, and controlled applications decompose into controlled versions of the same oracle calls, each of which Theorem D.1 counts as a single query. A total of $q$ applications therefore induces at most $2q$ queries to the single-input oracle, and Theorem D.1 yields $2q \in \Omega(\|\alpha\|_1/\varepsilon)$, hence $q \in \Omega(\|\alpha\|_1/\varepsilon)$. □

Three comments delimit the scope of Theorem D.1. First, the bound is unconditional within the access model of Definition 4: it constrains every algorithm equipped with those oracles, with no requirement that the computation be routed through the amplitude-encoding unitary $V$. This is possible because the hardness is confined to the feature-map oracle at the single training input $x_1$, while the classical oracles $O_R$, $O_N$, $O_C$ and the coefficient quantum-state oracle $W(\alpha)$ act on the fixed vector $\alpha = (A)$ and are therefore instance-independent, as established in Step 5. It says nothing, however, about procedures that exploit a known structure of the feature-map unitaries: the family of Step 1 is hard only insofar as the map $x' \mapsto U(x')$ is accessible as a black box. Second, the hard family has $N = 1$, where $\|\alpha\|_1 = \|\alpha\|_2 = \|\alpha\|_{2/3} = \|\alpha\|_\infty$. What the theorem certifies is therefore the optimality of the rate of Theorem C.9 in the two parameters $\|\alpha\|_1$ and $\varepsilon$; it does not rule out algorithms whose cost is governed by a finer feature of the coefficient vector, and we leave the tight rate for general $N$ open. Third, quantum amplitude estimation is optimal for each individual kernel term: applying Theorem D.1 with $N = 1$ and $A = 1$ shows that, for any target precision $\varepsilon_i$ with $\Omega(2^{-n}) \le \varepsilon_i \le \mathcal{O}(1)$, estimating $k(x, x_i)$ to additive error $\varepsilon_i$ requires $\Omega(\varepsilon_i^{-1})$ queries to the single-input oracle. Per-term estimation in *list-and-sum* strategies is therefore already optimal, and any improvement must arise from changing how the sum is approximated.

# E Sample-and-average algorithms

We introduce one final algorithmic choice along the axis of how to approximate the sum in Eq. (1), which we refer to as *sample-and-average*. Having identified that we can express the sum as the expectation value of a quantum observable, we take a step back and notice we could have expressed it as the expectation value of a *classical* observable. Following the notation in Definition 3, we could re-write inference as

$$f_\alpha(x) = \sum_{i=1}^{N} \alpha_i k(x, x_i) = \|\alpha\|_1 \sum_{i=1}^{N} p_i s_i k(x, x_i) = \|\alpha\|_1 \mathbb{E}_{i\sim p}[s_i k(x, x_i)]. \tag{E.1}$$

Intuitively, we could now decide to approximate the expectation value on the right-most part of the equation via its finite-sample mean. In doing this, we recall that we defined $k(x, x') = |\langle\phi(x)|\phi(x')\rangle|^2$ as a (quantum) expectation value itself, and hence this formula involves two sources of randomness: an *inner source of randomness* associated to estimating $k(x, x_i)$ for a given $i$, and an *outer source of randomness* in selecting $i$ from the distribution $p$. In the following, we give an algorithmic template as Algorithm E.1, where the *inner* source of randomness (evaluating the kernel, either via sampling or via quantum amplitude estimation), is wrapped by the *outer* source of randomness (evaluating the sum, via sampling). The template is given in terms of a free parameter $r \in \{1, 2\}$ which can be further specialized to $r = 1$ for sampling and $r = 2$ for quantum amplitude estimation. We analyze the sample complexity of Algorithm E.1 generically in Theorem E.1, and further provide Corollaries E.2 and E.3 for the specific runtimes using sampling and quantum amplitude estimation, respectively.

For the following result, we assume we have an *unbiased* estimator for quantum amplitude estimation, as for instance the one given by Ambainis et al. [2026].

**Theorem E.1** (Sample-and-average with fixed budget algorithm template analysis)**.** *Let $X_1, \dots, X_N \in [0, 1]$ be fixed values, let $a \in \mathbb{R}$ be a normalization factor, let $s = (s_1, \dots, s_N) \in \{\pm 1\}^N$ be a vector of signs, and let $p = (p_1, \dots, p_N)$ be an $N$-outcome probability distribution. Define $F \coloneqq a\mathbb{E}_{i\sim p}[s_i X_i]$. Assume we have $T$-query independent estimators $\hat{X}_i^{(T)}$ with $\mathbb{E}_{\hat{X}}[\hat{X}_i^{(T)}] = X_i$ and $\mathrm{Var}_{\hat{X}}[\hat{X}_i^{(T)}] \le c/T^r$, for some constant $c > 0$, and for $r \in \{1, 2\}$. For $L \in \mathbb{N}$, consider i.i.d. samples $i_1, \dots, i_L \sim p$, and define the intermediate $T$-query estimator $\hat{Y}_\ell^{(T)}$ corresponding to the $\ell^{th}$*

*sample:* $\hat{Y}_\ell^{(T)} \coloneqq s_{i_\ell} \hat{X}_{i_\ell}^{(T)}$. *Finally, define the* $M$*-query estimator* $\hat{F}^{(M)} = a \sum_{\ell=1}^{L} \hat{Y}_\ell^{(T)} / L$, *where* $M = LT$. *Then, for any* $\varepsilon > 0$, *to ensure* $|\hat{F}^{(M)} - F| < \varepsilon$ *with constant probability, it suffices to choose* $M \in \mathcal{O}(a^2 \varepsilon^{-2})$ *total queries, with* $T \in \mathcal{O}(1)$ *inner queries per sample. For any* $\delta \in (0, 1)$, *by a median-of-means amplification (repeating the estimator* $\mathcal{O}(\log(\delta^{-1}))$ *times and taking the median), the failure probability can be reduced to* $\delta$ *without changing the* $\varepsilon$*-dependence of the query complexity.*

*Proof.* We prove the statement directly by first deriving the variance of the estimator $\hat{F}^{(M)}$ and then optimizing over the possible values of $T$ and $L$, with $M = LT$. We define the intermediate quantity $\hat{Y}^{(M)} = \sum_{\ell=1}^{L} \hat{Y}_\ell^{(T)}$. From the independence of the estimators, it follows that

$$\mathrm{Var}\left(\hat{F}^{(M)}\right) = \frac{a^2}{L^2} \mathrm{Var}\left(\hat{Y}^{(M)}\right) = \frac{a^2}{L^2} \sum_{\ell=1}^{L} \mathrm{Var}\left(\hat{Y}_\ell^{(T)}\right) = \frac{a^2}{L^2} \cdot L \cdot \mathrm{Var}\left(\hat{Y}_\ell^{(T)}\right) = \frac{a^2}{L} \mathrm{Var}\left(\hat{Y}_\ell^{(T)}\right). \tag{E.2}$$

By Chebyshev's inequality, this implies that $|\hat{F}^{(M)} - F| < \varepsilon$ with constant probability whenever $\mathrm{Var}(\hat{F}^{(M)}) \in \mathcal{O}(\varepsilon^2)$. We next invoke the law of total variance, to obtain

$$\mathrm{Var}\left(\hat{Y}_\ell^{(T)}\right) = \mathbb{E}_{i\sim p}\left[\mathrm{Var}_{\hat{X}}\left(\hat{Y}_i^{(T)} \middle| i\right)\right] + \mathrm{Var}_{i\sim p}\left(\mathbb{E}_{\hat{X}}\left[\hat{Y}_i^{(T)} \middle| i\right]\right). \tag{E.3}$$

We can relate each term to the estimators for the initial quantities as

$$\mathbb{E}_{i\sim p}\left[\mathrm{Var}_{\hat{X}}\left(\hat{Y}_i^{(T)} \middle| i\right)\right] = \mathbb{E}_{i\sim p}\left[\mathrm{Var}_{\hat{X}}\left(\hat{X}_i^{(T)}\right)\right] \leq \mathbb{E}_{i\sim p}\left[\frac{c}{T^r}\right] = \frac{c}{T^r}, \tag{E.4}$$

$$\mathrm{Var}_{i\sim p}\left(\mathbb{E}_{\hat{X}}\left[\hat{Y}_i^{(T)} \middle| i\right]\right) = \mathrm{Var}_{i\sim p}\left(s_i X_i\right) \leq 1. \tag{E.5}$$

Plugging these back in the previous formulas, we obtain

$$\mathrm{Var}\left(\hat{F}^{(M)}\right) \leq \frac{a^2}{L}\left(1 + \frac{c}{T^r}\right). \tag{E.6}$$

This formula holds in general, and we can exploit it to bound the required number of queries. By enforcing $M = LT$, we would like to minimize $\frac{a^2}{L}\left(1 + \frac{c}{T^r}\right)$. We start by substituting $L = M/T$, to get

$$\frac{a^2 T}{M}\left(1 + \frac{c}{T^r}\right) = \frac{a^2}{M}\left(T + \frac{c}{T^{r-1}}\right). \tag{E.7}$$

To find the minimum with respect to $T$, we take the derivative

$$\frac{\mathrm{d}}{\mathrm{d}T}\left(\frac{a^2}{M}\left(T + \frac{c}{T^{r-1}}\right)\right) = \begin{cases} \frac{a^2}{M} & \text{if } r = 1, \\ \frac{a^2}{M}\left(1 - \frac{c}{T^2}\right) & \text{if } r = 2. \end{cases} \tag{E.8}$$

In the case $r = 1$, it follows that there are no local extrema, and the first derivative is always positive, hence the minimum is achieved at the lower limit of the domain, $T = 1$. In the case $r = 2$, the derivative nullifies when $T = \sqrt{c}$, which is a constant. Therefore, in both cases $T \in \mathcal{O}(1)$ achieves the optimal balance, for a total number of queries $M \in \mathcal{O}(a^2 \varepsilon^{-2})$. □

**Algorithm E.1** Sample-and-average with fixed budget, template.

**Input:** $x \in \mathcal{X}$ ▷ New input.
**Input:** $\varepsilon > 0$ ▷ Required additive precision.
**Output:** $f_\alpha(x) = \sum_{i=1}^{N} \alpha_i k(x, x_i)$ to $\varepsilon$ additive precision. ▷ Inference, Eq. (1).

1: **for** $\ell \in \{1, \ldots, L\}$ **do**:
2: Sampling oracle $O_C(\alpha)$: $i_\ell \sim p$
3: Estimate $k(x, x_{i_\ell})$ using $T$ queries to the single-input oracle: $\hat{k}^{(T)}(x, x_{i_\ell})$
▷ Variance $\mathcal{O}(\frac{1}{T^r})$, with $r \in \{1, 2\}$.
4: Multiply $\hat{k}^{(T)}(x, x_{i_\ell})$ with the sign $s_{i_\ell}$: $\hat{Y}_\ell^{(T)} = s_{i_\ell} \hat{k}^{(T)}(x, x_{i_\ell})$. ▷ Variance remains $\mathcal{O}(\frac{1}{T^r})$.
5: **end for** ▷ Total number of queries is $M = LT$.
6: Sum $\hat{Y}^{(M)} = \sum_{\ell=1}^{L} \hat{Y}_\ell^{(T)}$ ▷ Variance $\mathcal{O}(1 + \frac{c}{T^r})$.
7: Norm oracle $O_N(\alpha)$: $\|\alpha\|_1$
8: Multiply $\hat{Y}^{(M)}$ by $\|\alpha\|_1 / L$: obtain $\hat{f}_\alpha^{(M)}(x) = \frac{\|\alpha\|_1}{L} \hat{Y}^{(M)}$.
9: **return** $\hat{f}_\alpha^{(M)}(x)$
▷ Total variance $\mathcal{O}(\frac{\|\alpha\|_1^2}{L}(1 + \frac{c}{T^r}))$, sufficient $T \in \mathcal{O}(1)$ and $M \in \mathcal{O}(\varepsilon^{-2}\|\alpha\|_1^2)$.

**Corollary E.2** (Sampling)**.** *Consider Algorithm E.1 with $r = 1$ and the problem set-up in Definition 1. To estimate the quantum kernel model prediction in Eq. (1) to additive precision $\varepsilon$ with constant success probability, the runtime scaling is dominated by the query complexity*

$$M \in \mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right). \tag{E.9}$$

*The optimal allocation of queries is $T = 1$, $L = M$.*

*Proof.* Immediate from Theorem E.1. □

**Corollary E.3** (Quantum amplitude estimation)**.** *Consider Algorithm E.1 with $r = 2$ and the problem set-up in Definition 1. Assuming that quantum amplitude estimation provides an unbiased estimator: to estimate the quantum kernel model prediction in Eq. (1) to additive precision $\varepsilon$ with constant success probability, the runtime scaling is dominated by the query complexity*

$$M \in \mathcal{O}\left(\frac{\|\alpha\|_1^2}{\varepsilon^2}\right). \tag{E.10}$$

*The optimal allocation of queries is $T \in \mathcal{O}(1)$, $L \in \mathcal{O}(M)$.*

*Proof.* Immediate from Theorem E.1. □

Corollaries E.3 and E.2 therefore establish the same query-complexity upper bound for sampling and quantum amplitude estimation.

**Sample-and-average recovers previous algorithms.** The proof of Theorem E.1 confirms that the optimal balance between the inner and outer numbers of samples, $T$ and $L$ respectively, is struck by $T \in \mathcal{O}(1)$ (and even $T = 1$ for $r = 1$). Explicitly, this is independent of whether the inner estimator is based on sampling or quantum amplitude estimation $r \in \{1, 2\}$. The resulting estimator achieves the same query complexity scaling as both *all-at-once* and *list-and-sum* with adaptive budget, via sampling.

On the one hand, we can interpret Algorithm E.1 as the *all-at-once* observable where the linear combination over indices is instantiated as a classical expectation value, instead of a quantum superposition. On the other hand, we can also interpret it as the *list-and-sum* algorithm with adaptive budget, where the number of classical samples $M_i$ for the $i^{\text{th}}$ entry is automatically selected by the probability distribution $i \sim p$.

## NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The abstract and introduction clearly state the main contributions: (1) an improved query complexity of $\mathcal{O}(\|\alpha\|_1/\varepsilon)$ removing $N$-dependence, (2) a matching lower bound establishing optimality, (3) gate complexity analysis showing the query-optimal strategy is not always gate-optimal, and (4) a complete landscape of intermediate algorithms. All claims are supported by Theorems 4, 5, and Corollary D.2. Assumptions (black-box oracle access, pure-state feature maps) are stated explicitly in Definitions 1 and 4.

2. **Limitations**

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

   Answer: [Yes]

   Justification: The paper discusses several limitations: the restriction to pure-state (unitary) quantum feature maps, leaving mixed-state generalization as an open question (Section 4); the gap between query-optimality and gate-optimality (Theorem 5 and the accompanying remark); the assumption of black-box oracle access, which precludes structure-exploiting speedups; and the open question of whether the lower bound extends beyond the $V$-query model to the full access model (Section 3.4).

3. **Theory assumptions and proofs**

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

   Answer: [Yes]

   Justification: All theorems, lemmas, corollaries, and definitions are numbered and cross-referenced. The access model and problem set-up are formally stated in Definitions 1 and 4. Proof sketches are given in the main text (e.g., Theorems 2 and Lemma 3), with complete proofs in the supplementary material (Appendices B and C).

4. **Experimental result reproducibility**

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

   Answer: [N/A]

   Justification: This is a purely theoretical paper with no experimental results.

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

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

   Answer: [N/A]

   Justification: This is a purely theoretical paper; no code or data were produced.

6. **Experimental setting/details**

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

   Answer: [N/A]

   Justification: The paper does not include experiments.

7. **Experiment statistical significance**

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

   Answer: [N/A]

Justification: The paper does not include experiments.

8. **Experiments compute resources**

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

   Answer: [N/A]

   Justification: The paper does not include experiments.

9. **Code of ethics**

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

   Answer: [N/A]

   Justification: The research is purely theoretical work on quantum algorithms and conforms with the NeurIPS Code of Ethics. No human subjects, sensitive data, or dual-use concerns are involved.

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: This is foundational theoretical research on the complexity of quantum kernel inference. There is no direct path to negative societal applications; the work characterizes fundamental algorithmic trade-offs rather than enabling specific deployments.

11. **Safeguards**

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

    Answer: [N/A]

    Justification: This is foundational theoretical research on the complexity of quantum kernel inference. There is no direct path to negative societal applications; the work characterizes fundamental algorithmic trade-offs rather than enabling specific deployments.

12. **Licenses for existing assets**

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

    Answer: [N/A]

    Justification: The paper does not use existing code, data, or model assets.

13. **New assets**

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

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

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

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human subjects.

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

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

Answer: [N/A]

Justification: The paper does not involve human subjects research.

16. **Declaration of LLM usage**

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

Answer: [N/A]

Justification: LLMs were not used as a component of the core methodology in this research.