Statistical Zero Knowledge Problems for the Quantum Support Vector Machine

Saleh Naghdi, Casey Myers, and Michelle Simmons

In a seminal result, Liu *et al.* recently showed that the Quantum Kernel Estimation Support Vector Machine (`QKE-SVM`) provably outperforms all classical models on a classification task that reduces to the famous discrete logarithm problem [15]. This raises an important question: are there other such learning problems? To this end, we present a new classification task reducible to order-finding that is able to be learned efficiently by the `QKE-SVM`. We additionally outline the conditions under which learning this task would be classically intractable. Our approach relies on a three-part framework that is centered on the class of Statistical Zero Knowledge (SZK) problems whose instances have the curious property of being separable by a hyperplane within some quantum Hilbert space. We show that if a quantum computer can efficiently map into this space for a given SZK-based classification task, then that task is efficiently learnable by a `QKE-SVM`. Such an efficient mapping typically requires efficiently uncomputing the index of a one-way function in a procedure known as index erasure. Notably, our order-finding classification task is able to bypass the requirement for index erasure as it belongs to a version of SZK where the verifier has access to a quantum computer.

1 Introduction

Quantum Machine Learning (QML) for classical data has emerged as a cutting-edge field, promising to redefine the capabilities of classical machine learning algorithms by harnessing the power of quantum computers [3]. The field saw a significant advancement in 2009, when Lloyd *et al.* unveiled an algorithm for quantum principal component analysis, demonstrating an exponential speedup compared to the best-known classical models [16]. Their breakthrough algorithm led to the proliferation of QML models that promised quantum speedup, many of which adopted similar matrix-exponentiation techniques employed in the original paper [4, 11, 17, 25]. All of these algorithm share a dependency on quantum RAM, which, analogous to classical RAM, enables classical data to be efficiently loaded in superposition into a quantum computer [5].

These promising results were disturbed when it was later proved that an exponential speedup for these models can be obtained even on a classical computer equipped with data-loading capabilities similar to QRAM [26]. This raises the suspicion that the source of the speedups lies not in the internal workings of the QML model but rather the powerful way in which the data is loaded.

To address the impracticality of quantum RAM, Variational Quantum Circuits (VQCs) arose as a viable alternative [23]. Whereas QRAM-based QML models require data to be loaded in a highly specific way, VQCs relaxe this data-loading assumption by encoding a classical data point $\mathbf{x} \in \mathbb{R}^n$ as the parameters to a parameterized quantum circuit $U(\mathbf{x})$. This circuit is called a *quantum feature map* which transforms the original data point to a quantum feature state $|\phi(\mathbf{x})\rangle := U(\mathbf{x})\,|0\rangle$. Leveraging VQCs, the Quantum Kernel Estimation Support Vector Machine (`QKE-SVM`) was introduced in 2018 as one of the first quantum models that can be implemented on intermediate-scale quantum computers without requiring QRAM [9]. It leverages the fact that inner products between quantum feature states $K(x_1, x_2) := |\langle\phi(x_1)|\phi(x_2)\rangle|^2$ can be efficiently calculated with a quantum computer. Since the inner product is a measure of distance between quantum states, for a judiciously selected quantum feature map that brings together intraclass

Figure 1: **Complexity relations of classification tasks that are classically intractable to learn, but quantumly learnable by a `QKE-SVM`.** We begin at the orange circle $\mathcal{C}$ which is the set of concept classes that have efficient training set generation. Consider a concept class $C \in \mathcal{C}$. In the left half of the diagram, we establish that PAC learning C is classically difficult. This is satisfied if (but not necessarily only if) there exists an Abelian Hidden Subgroup Problem (AHSP) ξ that reduces to PAC learning C, noting that the set AHSP is famously conjectured to be classically difficult (**C.1**). Often this requirement means that the promise problem to which the concept class reduces satisfies some *aggregation property*, such as in the case of $C^{\text{DLP}} \leq \Pi^{\text{DLP}}$. Conversely, the right half outlines conditions for there existing a quantum learner that efficiently PAC learns C. With our quantum learner being the `QKE-SVM`, it suffices to show that the instance of C admit a quantum feature space in which they become linearly separable. This occurs if we require that PAC learning C reduces to a promise problem Π belonging to SZK (**C.2**). As indicated by the inset, $\Pi \in SZK$ reduces to the SZK-complete problem $\text{SD}_{\alpha\beta}$, denoted here by Γ [20], which further reduces to the problem of Quantum Sampling QS_δ, as indicated by the dotted arrow. The QS instance of Π produces our desired quantum feature map. If the QS reduction of Π additionally reduces to a problem ζ within BQP, then we are guaranteed that the quantum feature map above has an efficient implementation on a quantum computer and can furthermore be used to approximate the kernel matrix required for training the `QKE-SVM` (**C.3**).

instances while pushing interclass instances further apart, the kernel matrix K can contain a lot of information about the instances. As a result, passing it to a linear model like the Support Vector Machine (SVM) can make for an accurate predictor.

Since its formulation, there have been various empirical demonstrations of the `QKE-SVM`'s performance on a variety of classification tasks [1, 21, 22]. The first provable demonstration of quantum advantage occurred in 2021 when Liu *et al.* showcased a classification task C^{DLP} on which the `QKE-SVM` admits a provable quantum speedup over all classical counterparts [15]. Their result follows from the fact that learning C^{DLP} is equivalent to the discrete logarithm problem (DLP) which is widely believed to be classically difficult but quantumly tractable.

Liu *et al.*'s classically intractable quantumly tractable (CIQT) classification task raises an important question: are there any other such CIQT classification tasks that can enjoy a similar quantum speedup? A lot of exciting work has been dedicated to answering this question in different ways, including group theory [6], complexity theory [12], and computational learning theory [8] approaches.

In this paper, we answer this question by looking at the class of Statistical Zero Knowledge (SZK) problems as a potential source of CIQT classification tasks. Our primary result introduces a novel classification task named C^{OFP}, which we prove belongs to SZK and by virtue of which we show is able to be efficiently learned by the `QKE-SVM`. To build up to this result, we first provide a comprehensive review of Liu *et al.*'s classification task and examine the unique mathematical properties that make it classically intractable while efficiently learnable by a `QKE-SVM`. We encapsulate these mathematical properties in a three-part framework that is centred on SZK. To demonstrate the effectiveness of this framework in describing CIQT classification tasks, we provide two illustrative examples. In the first

instance, we deliberately choose a counterexample involving a classification task within the SZK complexity class that is based on the famous quadratic residuosity problem. This task fails to meet all the criteria in our proposed framework since it depends on a subroutine called index erasure. We show that the requirement for index erasure can be foregone if we consider a quantum SZK in which the verifier has access to a quantum computer. Under this modification, we are positioned to introduce our classification task C^{OFP} that is based on order-finding. Alongside Liu *et al.*'s findings, this classification task demonstrates that SZK problems can be a rich source of CIQT classification tasks learnable by a `QKE-SVM`.

2 Preliminaries

2.1 Elementary complexity theory

2.1.1 Promise problems

In the context of complexity theory, a promise problem is a type of computational problem that involves deciding whether a given input satisfying a specific promise is a YES or NO instance. Formally, a promise problem is defined by two disjoint sets of inputs: the "yes" instances, denoted by Π_{yes}, and the "no" instances, denoted by Π_{no}. Importantly, if the input does not satisfy the promise, the algorithm is not constrained to produce any specific output. The goal of solving a promise problem is to design an algorithm that correctly distinguishes between instances in Π_{yes} and Π_{no} while potentially providing any output (or no output) for instances outside this promise.

Promise problems are particularly useful in complexity theory for studying classes of problems, such as the class P and NP. Moreover, it provides a way of discussing when one problem is *as difficult* as another.

2.2 Elementary number theory

2.2.1 The Discrete Logarithm Problem

With origins in number theory, the discrete logarithm problem (DLP) has served as the basis for many well-known cryptography protocols conjectured to be classically intractable [10].

Let p be a prime number and $\mathbb{Z}_p^* = \{1, ..., p-1\}$ the integers under multiplication modulo p. If g is a generator of $\mathbb{Z}_p^*$, then given $y \in \mathbb{Z}_p^*$, the DLP asks for $x \in \mathbb{Z}_p^*$ such that $y = g^x$. In analogy to the real-valued logarithm, such a quantity x is called the discrete logarithm $\log_g(y) = x$. To summarise,

Discrete Logarithm Problem

Input: A prime number p, a generator g of the set $\mathbb{Z}_p^* = \{1, 2, \cdots, p-1\}$, and an element $y \in \mathbb{Z}_p^*$

Question: What is $\log_g(y)$?

It is widely conjectured that there is no classical algorithm capable of efficiently evaluating $\log_g(y)$ for large p. However, a bounded-error (and in fact, zero-error) polynomial-time quantum algorithm for the DLP does exist based on Shor's algorithm [18]. This places the DLP within BQP.

2.2.2 The $\text{DLP}_{\frac{1}{2}}$ Promise Problem

We now discuss our first promise problem. In the following, p is a prime and g a generator of the group $\mathbb{Z}_p^*$:

$$\begin{aligned} \Pi^{DLP_{\frac{1}{2}}} &= (\Pi_Y^{DLP_{\frac{1}{2}}}, \Pi_N^{DLP_{\frac{1}{2}}}) \\ \Pi_Y^{DLP_{\frac{1}{2}}} &= \left\{ y \ : \ \log_g(y) \in [1, \frac{p-3}{2}] \right\} \\ \Pi_N^{DLP_{\frac{1}{2}}} &= \{ y \ : \ \text{else} \} \ , \end{aligned} \tag{1}$$

In words, all the YES (NO) instances are integers whose discrete logarithm has a most-significant bit of $'0'$ ($'1'$). That is, an algorithm deciding $\Pi^{DLP_{\frac{1}{2}}}$ solves the most significant bit of the discrete logarithm for any $y \in \mathbb{Z}_p^*$.

Remarkably, it can be shown that the DLP reduces to $\Pi^{DLP_{\frac{1}{2}}}$, so that an algorithm solving the latter problem can be used to calculate the discrete logarithm of any $y \in \mathbb{Z}_p^*$ [13].

2.3 Probably Approximately Correct (PAC) Learning Theory

In the context of statistical learning theory, classification tasks are formally defined as concept classes over an instance space. The instance space $\mathcal{X}$ is the space of objects that are to be classified, such as the space of all RGB photos of a fixed size. Without loss of generality, we will claim that $\mathcal{X}$ is isomorphic to $\{0,1\}^n$. Along with this designation, we will also add a subscript denoting

the dimension of the instance space, $\mathcal{X}_n$. The instance space is also endowed with a distribution D.

A concept over the instance space is a binary-valued function $c : \mathcal{X}_n \to \{0, 1\}$ that assigns one of two labels to every element of the instance space. It can be thought of as an indicator function. A concept class $C_n = \{c_i\}$ is a family of concepts over $\mathcal{X}_n$. Finally, we define the set of all concept classes of all dimensions as $C = \cup_n C_n$ over the instant space $\mathcal{X} = \cup_n \mathcal{X}_n$.

We now introduce the concept of Probably Approximately Correct (PAC) learning a concept class. This formalizes the notion of learning a classification task and is defined as follows:

Definition 2.1. PAC Learning a concept class. Consider an instance space $\mathcal{X}_n$ endowed with the distribution D. Suppose C_n is a concept class over the instance space $\mathcal{X}_n$. An algorithm $\mathcal{A}$ is said to PAC learn the concept class C_n if $\forall c \in C_n$, given a labeled set $\{(x, c(x)\}_{i=1}^{poly(n)}$ of points randomly sampled from the distribution D, and $\epsilon > 0, \delta < 1/2$, it outputs an efficiently computable hypothesis h that satisfies

$$Pr_{x \in \mathcal{X}}(h(x) \neq c(x)) \leq \epsilon \tag{2}$$

with a probability of at least $1-\delta$ over the choice of the random sample and internal randomization of the algorithm. If the algorithm returns h in time $\mathcal{O}\left(\text{poly}\left(\frac{1}{\delta}, \frac{1}{\epsilon}, n\right)\right)$ then C is *efficiently* PAC learnable.

We refer the interested reader to Ref [13] for further details on PAC learning theory.

3 Review of Liu *et al.*'s classification task based on the discrete logarithm problem

As the only instance of end-to-end quantum machine learning achieving quantum advantage in the literature, the work of Liu *et al.* deserves a closer exploration. Liu *et al.*'s classification task is equivalent to the famous discrete logarithm problem (DLP) which is widely conjectured to be classically intractable. In this section, we will review Liu *et al.*'s result by describing the classification task C^{DLP} and the promise problem Π^{DLP} which it reduces to. We show a quantum feature map that maps instances in Π^{DLP} to a Hilbert space in which they become separated by a hyperplane based on their label. Finally, we present the algorithm for efficiently implementing the quantum feature map which crucially makes use of the quantum algorithm for efficiently solving the DLP.

3.1 The DLP concept class C^{DLP}

We now describe the discrete-logarithm-based concept class C^{DLP} introduced by Liu *et al.*. For a fixed prime p of bit-size $n = \lceil \log p \rceil$, we consider the uniform distribution over the instance space $\mathbb{Z}_p^*$. Our concept class is the set of all concepts of the form c_s^{DLP}

$$C_n^{DLP} = \{c_s^{DLP} | \forall s \in \mathbb{Z}_p^*\} \tag{3}$$

with the concepts defined as

$$c_s^{DLP}(y) = \begin{cases} +1, & \text{if } \log_g(y) \in [s, s + \frac{p-3}{2}] \\ -1, & \text{else.} \end{cases} \tag{4}$$

Note the parameter s which separates the ring representing $\mathbb{Z}_p^*$ into two semicircles. A given concept c_s^{DLP} labels an input $y \in \mathbb{Z}_p^*$ based on which of the two semicircles its discrete logarithm lies[1].

To illustrate what the labelling rule in Equation 4 corresponds to, it is instructive to first imagine the $p-1$ many elements of $\mathbb{Z}_p^*$ as being distributed evenly along a circle. While in this representation the YES and NO class labels seem to be distributed randomly (Figure 2a) – and indeed, they are to a classical classifier – mapping each point to the position of its discrete logarithm $y \mapsto \log_g(y)$ reveals the underlying classification rule. In this logarithm space $\log_g(\mathbb{Z}_p^*)$ the two labels are separated by a diameter line intercepting the point s, with YES (NO) points lying clockwise (anticlockwise) from s (Figure 2b). The existence of this diameter partition should already allude to there being a hyperplane that separates between the two classes of instances.

Classical hardness of learning C^{DLP} . It is important to mention that Liu *et al.* prove that solving the DLP reduces to the task of *PAC learning* C^{DLP}. This means that PAC learning C^{DLP} must be at the very least *as difficult* as the DLP.

[1]The addition in the interval is a slight abuse of notation as addition is not a native operation of $\mathbb{Z}_p^*$, so we have (p-1) + 1 = 1

Since the DLP is widely believed to be classically difficult, it may be strongly conjectured that no classical model can efficiently learn C^{DLP}.

3.2 The discrete logarithm promise problem Π_{DLP}

Having introduced the concept class C^{DLP}, we can also define the natural promise problem Π^{DLP}:

$$\begin{aligned} \Pi^{DLP} &= (\Pi_Y^{DLP}, \Pi_N^{DLP}) \\ \Pi_Y^{DLP} &= \left\{ y \ : \ c_s^{DLP}(y) = +1 \right\} \\ \Pi_N^{DLP} &= \left\{ y \ : \ c_s^{DLP}(y) = -1 \right\} , \end{aligned} \tag{5}$$

Where Π_Y^{DLP} represents all the green instances in 2b and Π_N^{DLP} all the red. This promise problem parameterised by s is actually identical to $\Pi^{DLP_{\frac{1}{2}}}$ for $s = 1$!

3.3 A quantum feature map for separating Π^{DLP}

Let $n = \lceil \log_2(p) \rceil$ be the minimum number of bits required to represent prime number p and define (y, g, s) as before. We will show that the quantum feature space defined by the map

$$|y\rangle \mapsto \sum_{i=0}^{2^k-1} |y \cdot g^i \mod p\rangle := |C_{y,k}\rangle . \tag{6}$$

for a choice of a constant $k \leq n$ admits a hyperplane that separates the data points based on their classes. The data points y are represented as feature states $|C_{y,k}\rangle$ in this feature space.

Consider the state

$$|\phi_s\rangle = \sum_{i=0}^{(p-1)/2-1} |g^s \cdot g^i \mod p\rangle , \tag{7}$$

which is a superposition of all $y \in \Pi_Y^{DLP}$. In Figure 2b, $|\phi_s\rangle$ is depicted as the blue arc sweeping a semicircle across all the YES instances which are depicted in green. The inner product of this state and a given feature state $|C_{y,k}\rangle$ largely depends on the label of y:

$$\begin{aligned} |\langle \phi_s | C_{y,k} \rangle| &= \sum_{i=0}^{2^k-1} \sum_{j=0}^{(p-1)/2-1} \langle y \cdot g^i | g^s \cdot g^j \rangle \\ &= \sum_{i=0}^{2^k-1} \sum_{j=0}^{(p-1)/2-1} \delta_{g^{i+\log_g(y)}, g^{j+s}} \\ &= \sum_{i=0}^{2^k-1} \sum_{j=0}^{(p-1)/2-1} \delta_{i+\log_g(y), j+s} \end{aligned}$$

Note that this expression corresponds to the length of the overlap between the arcs swept by $|\phi_s\rangle$ and feature state $|C_{y,k}\rangle$:

$$\begin{aligned} &= |[\log_g(y), \log_g(y) + 2^k - 1] \cap [s, s + \frac{p-1}{2} - 1]| \\ &= |[\max(\log_g(y), s), \min(\log_g(y) + 2^k - 1, s + \frac{p-1}{2} - 1)]| \end{aligned}$$

which as illustrated in Figure 2b produces a different value depending on the class label of y:

$$= \begin{cases} 2^k & \text{for } (1-\Delta)\% \text{ of } \Pi_Y^{\text{DLP}} \\ 0 & \text{for } (1-\Delta)\% \text{ of } \Pi_N^{\text{DLP}} \end{cases}$$

where $\Delta = \frac{2^k}{(\frac{(p-1)}{2})} = \frac{2^{k+1}}{p-1}$ is the ratio of the arc length spanned by the feature state to that of $|\phi_s\rangle$. By including the normalisation factor and squaring the entire quantity, we get the Hilbert-Schmidt inner product:

$$|\langle \phi_s | C_{y,k} \rangle|^2 = \begin{cases} \Delta & \text{for } (1-\Delta)\% \text{ of } \Pi_Y^{\text{DLP}} \\ 0 & \text{for } (1-\Delta)\% \text{ of } \Pi_N^{\text{DLP}} \end{cases} \tag{8}$$

The fact that taking the inner product with $|\phi_s\rangle$ is able to separate YES and NO establishes the following remark:

Remark. The state $|\phi_s\rangle$ is the normal to a hyperplane that separates the YES and NO instances of Π^{DLP} when mapped to the feature space $\mathcal{V} = \text{span}\{|C_{y,k}\rangle \ \forall y\}$.

To demonstrate the sense in which $|\phi_s\rangle\langle\phi_s|$ (i.e. $|\phi_s\rangle$ as a density matrix) is the normal vector to a differentiating hyperplane, we can alternatively represent both it and additionally any $|C_{y,k}\rangle\langle C_{y,k}|$ with respect to the 4^n-dimensional Pauli basis $\{P_\alpha\}$. Using the HS inner product, we identify these density matrices with their Fourier coefficients, $\vec{w}_s = (w_{s,\alpha})$ and $\vec{y} = (y_\alpha)$, respectively. In this representation, the Hilbert-Schmidt inner

(a) (b)

Figure 2: **Use of the quantum feature map described in Equation 6 to separate the yes/no (green/red) instances of Π^{DLP} for the special case of $(p, g, s, k) = (29, 11, 11, 3)$.** In 2a, the labeled points lying along the circumference are integers $y \in \mathbb{Z}_{29}^*$ with labels assigned according to the labeling rule $c_1^{DLP}1$. For example, the data point $y = 28$ is labelled green because $log_{11}(28) = 14 \in= [s, s + (p-1)/2 - 1] = [11, 24] \implies c_1^{DLP}1(x) = +1$. These points form our data space. In 2b, the data points have been mapped to the position of their discrete logarithm action. For example, since $y = 28 \mapsto log_{11}(28) = 14$, $x = 29$ is mapped to the position of 14. The concentric arcs represent the feature states $|C_{y,k}\rangle$ for $y = 24 \in \Pi_Y$ (green) and $y = 6 \in \Pi_N$ (red), as well as the normal to the hyperplane $|\phi_1 1\rangle$ defined in 7 which a quantum computer is efficiently able to map the data points to. These arcs are shown to sweep whichever integer they contain within their superposition. For example, the superposition obtained from $|6\rangle \mapsto |C_{6,k}\rangle$ (shown as the red arc) sweeps the first $2^k - 1 = 7$ integers whose discrete logarithm come after $log_{11}(6) = 26 \mod p$. Note that the green arc (YES feature state) overlaps much more with the blue arc (normal to the hyperplane) than does the red arc (NO feature state), highlighting the role of $|\phi_s\rangle$ as defining a wide-margin hyperplane in this space.

product of the space $\mathbb{C}^{2^n \times 2^n}$ reduces to the familiar Euclidean inner product in $\mathbb{R}^{4^n}$:

$$\langle |\phi_s\rangle\langle\phi_s|, |C_{y,k}\rangle\langle C_{y,k}|\rangle_{HS} = |\langle\phi_s|C_{y,k}\rangle|^2 = \sum_{\alpha=0}^{4^n-1} w_{s,\alpha} y_\alpha = \langle \vec{w}_s, \vec{y}\rangle_{\mathbb{R}} .$$

Taking $b = \frac{\Delta}{2}$, the inner products in Equation 8 can be recast as Euclidean vectors:

- $\vec{w}_s \cdot \vec{y} - b = \frac{\Delta}{2}$ for $(1-\Delta)\%$ of Π_Y^{DLP}
- $\vec{w}_s \cdot \vec{y} - b = -\frac{\Delta}{2}$ for $(1-\Delta)\%$ of Π_Y^{DLP} .

where $\vec{w}_s$ and b can naturally be interpreted as the normal vector to a hyperplane and offset term respectively. At least $(1-\Delta)\%$ of the YES (NO) data points lie above (below) the hyperplane. Thus, this hyperplane separates most instances based on class with a margin of $\Delta/\|w_s\|$.

3.3.1 Implementing the quantum feature map $|y\rangle \mapsto |C_{y,k}\rangle$

As a first step towards building the desired quantum feature map in Equation 6, consider the following classical operation

$$C_{y,k} : \{0,1\}^k \to \{0,1\}^n$$
$$C_{y,k}(i) = y \cdot g^i \mod p .$$

By using elementary multiplication modulo p, the quantum circuit for which is well understood and efficient[2], it is easy to implement the reversible operator

$$\hat{C}_{y,k} : \mathcal{H}^{\otimes 2n} \to \mathcal{H}^{\otimes 2n}$$
$$\hat{C}_{y,k} |i\rangle |0\rangle^{\otimes n} = |i\rangle |C_{y,k}(i)\rangle$$

[2]Using the multiplication operator $U_{k,p}|l\rangle = |k \cdot l \mod p\rangle$ and exponentiation operator $V_{k,p}|l\rangle = |k^l \mod p\rangle$, one can at once see that $|C_{y,k}(i)\rangle = U_{y,p}^i V_{g,p} |i\rangle$[19]

where $|C_{y,k}(i)\rangle$ in the second register acts on the k least significant bits from the first register. Now using the exact quantum algorithm for solving the DLP, the following operator can be reproduced [18]:

$$U_y \, |C_{y,k}(i)\rangle \, |0\rangle^{\otimes k} = |i\rangle \, |C_{y,k}(i)\rangle \tag{9}$$

Finally, this operator enables the desired superposition state to be obtained up to the addition and removal of auxiliary qubits:

$$|0\rangle \xrightarrow{H^{\otimes k}} \sum_{i=0}^{2^k-1} |i\rangle \xrightarrow{\hat{C}_{y,k}} \sum_{i=0}^{2^k-1} |i\rangle \, |C_{y,k}(i)\rangle \tag{10}$$

$$\xrightarrow{U_y^\dagger} \sum_{i=0}^{2^k-1} |C_{y,k}(i)\rangle := |C_{y,k}\rangle \tag{11}$$

We have thus provided an efficient algorithm for the desired quantum feature map mapping an input y to the quantum feature state $|C_{y,k}\rangle$. From here, we proceed by calculating inner-products between these feature states in order to estimate the kernel matrix K for the dataset. K will then be passed to a classical SVM to predict the instances of our concept class C^{DLP} with high accuracy.

4 When is learning a concept class classically intractable but quantumly tractable?

In the previous section, we examined a concept class C^{DLP} that is classically intractable but quantumly tractable (CIQT) to learn. To claim that it is classically intractable, it was shown that the task of probabilistically solving the discrete logarithm problem (DLP) reduces to that of efficiently PAC learning C^{DLP}. And since the discrete logarithm problem is strongly conjectured to be classically difficult, this establishes that PAC learning C^{DLP} is at least as difficult.

On the other hand, we saw that learning C^{DLP} itself reduces to a promise problem whose instances have the remarkable property of being separated by a soft wide-margin hyperplane in a Hilbert space. It was then shown that the map into this space is efficiently implementable on a quantum computer, allowing it to be used as a *quantum* feature map for learning by the quantum kernel estimation support vector machine (`QKE-SVM`). This demonstrates that learning C^{DLP} is quantumly tractable.

Hence, the fact that C^{DLP} is a CIQT classification task can be summarized in terms of three characteristics:

C.1 Efficiently PAC Learning C^{DLP} is classically intractable.

C.2 There exists a Hilbert space $\mathcal{V}$ in which most instances of C^{DLP} become separated with a wide-margin hyperplane.

C.3 Inner products within $\mathcal{V}$ can be estimated efficiently using a quantum computer.

C.2 and **C.3** together imply that the C^{DLP} can be efficiently learned using the `QKE-SVM`. Subsequently, **C.1** suggests that no classical model exists that could learn the C^{DLP} better than just randomly guessing, therefore establishing

Drawing inspiration from C^{DLP}, this section introduces three mathematical properties that together form a sufficient condition for making a concept class C CIQT:

O.1 There exists an Abelian Hidden Subgroup Problem (AHSP) that reduces to PAC learning C, provided that randomly sampling instance-label pairs from the distribution D over the instance space is efficient.

O.2 PAC learning C reduces to the problem of solving a promise problem Π that is contained in the class of Statistical Zero Knowledge (SZK) problems. [2].

O.3 The reduction of Π as a Quantum Sampling problem is in BQP.

It is important to emphasize that the three properties above form *sufficient* but not necessary criteria for a concept class to be CIQT. In fact, Liu *et al.*'s C^{DLP} itself does not satisfy all of these three criteria. Namely, the promise problem Π^{DLP} which the task of efficiently learning C^{DLP} reduces to is not technically in SZK. However, Π^{DLP} *is* an extension of a related problem $DLP_{\frac{1}{6}}$ which does belong to SZK.

4.1 When is a concept class classically difficult to learn?

4.1.1 Why is learning C^{DLP} classically difficult?

In the previous section, we briefly mentioned that probabilistically solving the classically difficult

Discrete Logarithm Problem reduces to learning C^{DLP}:

Theorem 1. *C^{DLP} is classically efficiently PAC learnable $\implies DLP \in BPP$*

Since it is strongly conjectured that DLP $\notin BPP$, we may conclude that C^{DLP} is not classically efficiently learnable by contrapositive. This fact follows from the DLP belonging to the class of Abelian Hidden Subgroup Problems which are famously considered to be classically difficult. We may conclude that for a given concept class to be classically difficult to learn, it is sufficient for an Abelian Hidden Subgroup problem to be reducible to it.

Note that PAC learnability only implies the existence of a learning algorithm $\mathcal{L}$ - not a hypothesis h that approximates a concept in the concept class. To obtain h, the learner additionally requires polynomially many labeled input-output pairs. It is implicitly assumed in Theorem 1 that generating a training set for the learner of C^{DLP} is efficient.

Lemma 2. *Generating a sample $\{(y_i, c_s^{DLP}(y_i)\}_{i=1}^{\mathsf{poly}(n)}$ for y_i uniformly distributed over $\mathbb{Z}_p^*$ is efficient.*

Proof. Randomly sample $x_i \sim U(\mathbb{Z}_p^*)$ and note that $y_i \leftarrow g^{x_i} \sim U(\mathbb{Z}_p^*)$ as well. If $x_i \in [s, s+(p-1)/3]$ return $(y_i, +1)$. Otherwise return $(y_i, -1)$. Repeat $\mathsf{poly}(n)$ times to generate a set of labeled instances. □

It is instructive to provide a sketch of the proof shown in [15] for Theorem 1 to identify what underlying properties render a concept class CI.

Proof. First, consider the simpler concept class $C_s^{DLP} = \{c_s^{DLP}\}$ consisting of a single concept with a known s. It is clear that C_s^{DLP} reduces to learning C^{DLP}, which corresponds to the more difficult task of approximating c_s for an *unknown* s:

Lemma 3. *PAC learning $C_s^{DLP} = \{c_s^{DLP}\} \leq$ PAC learning C^{DLP}*

Thus, it is sufficient to show that learning C_s^{DLP} is classically difficult. Since s is known, we remark that learning $C_{\frac{1}{2}}^{DLP}$ reduces to C_s^{DLP} as we may simply bijectively transform the instances of the former to that of the latter in the manner of $y \mapsto y \cdot g^{s-1}$ all while preserving the uniform distribution over the instance space.

Lemma 4. *PAC learning $C_{\frac{1}{2}}^{DLP} \leq$ PAC learning C_s^{DLP}*

From this lemma, it follows that if C_s^{DLP} is efficiently PAC learnable, then there exists an efficient learner $\mathcal{L}$ that with high probability outputs an h that is an ϵ-close approximation of $c_{\frac{1}{2}}^{DLP}$. Since our distribution over the instance space $\mathbb{Z}_p^*$ is uniform, the error of h is simply the proportion of instances that it misclassifies. By selecting $\epsilon = 1/2 - \epsilon'$ for some $\epsilon' = \mathcal{O}\left(\frac{1}{\mathsf{poly}(n)}\right)$, with high probability h correctly decides at least $1/2 + \epsilon'\%$ of the instances.

Lemma 5. *Efficiently PAC learning C_s^{DLP} given an oracle for training set generation $\implies \exists$ an efficient algorithm that correctly decides on at least $1/2 + \epsilon'\%$ of the instances of $\mathbb{Z}_p^*$ for the $DLP_{\frac{1}{2}}$ promise problem.*

We now summon a lemma proved in [7] which establishes the nontrivial result that knowing $c_{\frac{1}{2}}(x)$ just over half of the instances is enough to know $c_{\frac{1}{2}}(x)$ of any instance in $\mathbb{Z}_p^*$ with high probability:

Lemma 6. ***Aggregation property*** *Given an oracle that correctly decides $DLP_{\frac{1}{2}}$ on at least $1/2 + \epsilon'\%$, there exists an efficient algorithm that correctly decides $DLP_{\frac{1}{2}}$ for all instances with high probability in time $\mathcal{O}\left(\frac{1}{\epsilon'}\right)$*

A unique aspect of the discrete logarithm, the aggregation property represents the nexus between PAC learning a concept class and solving the promise problem which the concept class reduces to with high probability. It is premised on the fact that if a random variable r follows a uniform distribution over $\mathbb{Z}_p^*$ then so does yr for an arbitrary $y \in \mathbb{Z}_p^*$. Hence, we may conclude a property of y with high probability by calling the oracle on random inputs yr_i for sufficiently many randomly sampled r_i.

Finally, the lemma below affirms that deciding $DLP_{\frac{1}{2}}$ with high probability is sufficient for solving the DLP with high probability.

Lemma 7. *$DLP_{\frac{1}{2}} \in BPP \implies DLP \in BPP$*

□

4.1.2 Sufficient criteria for a general concept class to be classically difficult to learn

The concept class C^{DLP} is classically intractable to learn efficiently because of three reasons:

1. The training sample over the uniform distribution is efficient to generate.

2. The discrete logarithm problem reduces to PAC learning the concept class C^{DLP} with efficient generation of training set

3. The discrete logarithm problem belongs to the class of Abelian Hidden Subgroup Problems famously conjectured to be classically intractable but quantumly tractable.

4.2 When is a concept class quantumly easy to learn?

Consider just the singleton concept class C_s^{DLP} for a fixed $s \in \mathbb{Z}_p^*$. PAC learning this concept class reduces to deciding the Π^{DLP} promise problem as defined in Equation5. An important step in establishing that a promise problem can be PAC learned by a support vector machine is showing that its instances can be mapped into a space where they become linearly separable. For the case of C^{DLP}, this criterion of separability is captured as

$$Pr_{y\sim\mathbb{Z}_p^*}\left[h_s^{DLP}(y)(|\langle C_s|C_y\rangle|^2+b)>\Delta/2\right]=1-\Delta \tag{12}$$

Equation 12 conveys that there exists a hyperplane $(|C_s\rangle, b)$ that separates most instances of Π^{DLP} with a wide margin. Note that the parameter $\Delta = \mathcal{O}(\frac{1}{\mathsf{poly}(n)})$ decreases only polynomially in the size of $n = |p|$. This is an important property of the C^{DLP} as it ensures that the margin of the separating hyperplane remains wide *enough* with increasing n for the `QKE-SVM` to be able to learn it with polynomially many training instances.

By the end of this section, we aim to characterize when a promise problem admits a hyperplane $(\vec{w}, b)$ separating the instances. To this end, we focus our attention on another way of expressing the inner product $\langle C_s|C_y\rangle$ which is the central quantity in establishing separability.

4.2.1 SZK for linear separability

We begin by rewriting our feature states from Equation 6 in the following way:

$$|C_{y,k}\rangle = \sum_{x\in\{0,1\}^n} \sqrt{D_{C_{y,k}}(x)}\,|x\rangle \tag{13}$$

we can identify $|C_{y,k}\rangle$ with a corresponding probability distribution $D_{C_{y,k}}$ defined as

$$D_{C_{y,k}} : \{0,1\}^n \to \{0,1\}, \quad x \mapsto \frac{|C_{y,k}^{-1}(\{x\})|}{2^k} \tag{14}$$

In words, $D_{C_{y,k}}(x)$ refers to the proportion of inputs i in the domain of $C_{y,k}$ whose image is $C_{y,k}(i) = x$. Since $C_{y,k}$ is a bijection, there is a unique input corresponding to each output which implies $D_{C_{y,k}}(x) = \frac{1}{2^k}$ over its support. In its most general form, the distribution D_f of an arbitrary function f can be encoded in the quantum state:

$$|f\rangle = \sum_{x\in\text{Im}(f)} \sqrt{D_f(x)}\,|x\rangle \tag{15}$$

Within this representation, inner products between quantum states correspond to the fidelity between their probability distributions, a popular measure of distance between distributions:

$$\langle f|g\rangle = \sum_{x\in\text{Im}(f)} \sqrt{D_f(x)D_g(x)} = \mathcal{F}(D_f, D_g) \tag{16}$$

Another measure of difference between probability distributions is the variational difference, defined as

$$|p-q| = \frac{1}{2}\sum_x |p(x)-q(x)| \tag{17}$$

It is a well-known result in probability theory that the fidelity is tightly bounded by the variational difference:

$$1-|p-q| \le \mathcal{F}(p,q) \le \sqrt{1-|p-q|^2} \tag{18}$$

As a result, the inner product between two states can also be equally understood in terms of the variational difference of their underlying probability distributions. If instead of using the quantum states $|C_{y,k}\rangle$ and $|\phi_s\rangle$ we worked with their probability distribution representations, $D_{C_{y,k}}$ and $D_{C_{g^s,\lfloor\log_2(p)\rfloor-1}}$, then the analogue of the differential overlap we saw in (8) is the difference

between the two probability distributions:

$$|D_{C_{y,k}} - D_{C_{g^s,\lfloor\log_2(p)\rfloor-1}}| \begin{cases} \geq \alpha & \text{for } (1-\Delta)\% \text{ of } \Pi_Y^{\text{DLP}} \\ \leq \beta & \text{for } (1-\Delta)\% \text{ of } \Pi_N^{\text{DLP}} \end{cases} \tag{19}$$

Because of the tight bound between fidelity and variational distance, it is easy to see that the general promise problem

$$\begin{aligned} \Gamma &= (\Gamma_{\text{Y}}, \Gamma_{\text{N}}) \\ \Gamma_{\text{Y}} &= \{(f,g) \ : \ |D_f - D_g| > \alpha\} \\ \Gamma_{\text{N}} &= \{(f,g) \ : \ |D_f - D_g| < \beta\} \end{aligned} \tag{20}$$

consequently reduces to the problem

$$\begin{aligned} \mathcal{Q} &= (\mathcal{Q}_{\text{Y}}, \mathcal{Q}_{\text{N}}) \\ \mathcal{Q}_{\text{Y}} &= \left\{(|f\rangle, |g\rangle) \ : \ |\langle f|g\rangle|^2 > \alpha'\right\} \\ \mathcal{Q}_{\text{N}} &= \left\{(|f\rangle, |g\rangle) \ : \ |\langle f|g\rangle|^2 < \beta'\right\} \end{aligned}$$

If the problem of `Quantum Sampling` $QS_{\delta=0}$ can be solved:

Quantum Sampling QS_δ
Input: A function f and $\delta > 0$
Question: Can the state ρ be efficiently product in time $\mathcal{O}(1/\delta)$ such that $||\rho - |f\rangle\langle f|\,|| \leq \delta$

Assuming this, then for a promise problem Π that reduces to Γ, instances $x \in \Pi_{\text{Y}}$ can efficiently be mapped to quantum states $(|f_x\rangle, |g_x\rangle)$ with large inner product. If it happens to be that $|g_x\rangle = |g_{x'}\rangle \, \forall x, x' \in \Pi_{\text{Y}} \cup \Pi_{\text{N}}$, then $|g_x\rangle \equiv |g\rangle$ is a constant and can be understood as a normal to a hyperplane. Note that this was precisely the case for C^{DLP}, as $|C_{g^s,\lfloor\log_2(p)\rfloor-1}\rangle$ is the same across all instances. Furthermore, $x \mapsto |f_x\rangle$ can be recognized as the relevant quantum feature map that linearly separates the instances of Π, in analogy to $y \mapsto |C_{y,k}\rangle$,

What is the relevance of Γ in this discussion? In the special case where $0 \leq \beta \leq \alpha^2 \leq 1$, the promise problem in Equation **??** corresponds to the well-studied Statistical Difference $\text{SD}_{\alpha\beta}$ promise problem [20]. The upshot of this formulation is that $\text{SD}_{\alpha\beta}$ is complete for an entire class of problems known as *statistical zero-knowledge* (SZK).

Remark. $\text{SD}_{\alpha\beta}$ is SZK-complete

Informally, a SZK problem consists of an interactive proof system whereby a *verifier* tries to verify that an element x satisfies some condition P via a two-way exchange of messages between itself and an all-knowing *prover*. For a problem to be categorised as SZK, however, the verifier must not learn anything during its exchange with the prover other than the fact that x satisfies P [**?**]. SZK contains other problems including Quadratic Residuosity (QR) and the Closest Vector Problem (CVP) often used in cryptographic contexts.

Besides separability, in the Appendix we additionally discuss how QS_δ reductions of SZK problems yield hyperplanes that satisfy other properties like boundedness which are necessary for learnability by a `QKE-SVM`.

We have shown that SZK problems admit a QS_δ reduction that implies their instances are linearly separable in some quantum Hilbert space. In the next section we investigate the conditions under which the map into that space is efficiently implementable using quantum circuits.

4.2.2 Efficient Quantum Sampling for efficient implementation of the feature map

So far we have recognized that SZK problems are a class of problems whose instances can become linearly separated with a wide-margin hyperplane in some Hilbert space. The map into this Hilbert space is defined by the QS_δ reduction of the SZK problem. If it furthermore reduces to a problem in BQP, then the desired quantum feature map can be efficiently implemented using a quantum computer. In the Appendix, we furthermore show that noise robustness of the `QKE-SVM` is preserved even if the state preparation is approximate, i.e. $\delta \neq 0$.

For the QS_δ reduction of an SZK problem to be in BQP, we often requires an important step known as index erasure. This can be seen even in Liu *et al.*'s special case. The series of operations from Eq. (11) used to obtain the feature state $|C_{y,k}\rangle$ can be divided into two main components:

$$\underbrace{|0\rangle \xrightarrow{H^{\otimes k}} \sum_{i=0}^{2^k-1} |i\rangle \xrightarrow{U_{\text{mod}}} \sum_{i=0}^{2^k-1} |i\rangle\,|C_{y,k}(i)\rangle}_{\text{Initial state preparation}}$$

$$\underbrace{\xrightarrow{U_y^\dagger} \sum_{i=0}^{2^k-1} |C_{y,k}(i)\rangle}_{\text{Index erasure}} := |C_{y,k}\rangle \ .$$

Going from the penultimate to the last line re-

quires the following unitary

$$|C_{y,k}(i)\rangle \xrightarrow{U_y} |i\rangle\,|C_{y,k}(i)\rangle\ ,$$
$$|g^{\log_g(y)+i}\rangle \xrightarrow{U_y} |i\rangle\,|g^{\log_g(y)+i}\rangle\ ,$$

Observe that U_y computes the discrete logarithm[3] which quantum computers are known to perform efficiently [18]. The DLP is a member of the Abelian hidden subgroup problems (AHSP), all of which belong to BQP [19].

In summary, it is sufficient for the quantum feature map of an SZK promise problem to be efficiently implemented if the QS_δ reduction of the SZK promise problem – which may be harder than the promise problem itself – reduces to a problem in BQP. This satisfies criteria **C.3**. It is important not to confuse **C.3** and **C.1**. In **C.1**, the fact that the original SZK promise problem Π^{DLP} reduces to AHSP renders Π^{DLP} classically intractable. In **C.3**, the fact that the QS_δ *reduction* of Π^{DLP} reduces to a problem in BQP (which may or may not be AHSP) ensures the efficient implementation of the quantum feature map on a quantum computer.

In the next section, we will identify and investigate two new concept classes based on SZK promise problems. The first promise problem lacks a known implementation for its induced quantum feature map. In contrast, the second promise problem does have efficient QS_δ, making it learnable by a `QKE-SVM`.

5 A classification task based on quadratic residuosity with unknown index erasure

Guided by the framework we proposed in the previous section, we will now investigate one famous SZK problem called the Quadratic Residuosity problem as the basis for a new potentially CIQT classification task to be learned by the `QKE-SVM` [2]. We will introduce the notion of quadratic residues from number theory first and then the SZK promise problem Π^{QRP} that is based directly on it. The promise problem will motivate a natural definition for a concept class C^{QRP}. We will examine a quantum feature map through which Π^{QRP} is made linearly separable with a wide-margin, following from **O.2**. On the other hand, it remains unknown whether C^{QRP} satisfies either **O.1** or **O.3**, as there is no known efficient implementation into the quantum feature space nor is it known to be classically intractable. This leaves the status of C^{QRP} as a CIQT classification task unresolved.

5.1 A promise problem and concept class based on quadratic residues in $\mathbb{Z}_N^*$

Let our instance space be $\mathbb{Z}_N^*$, the multiplicative group of integers modulo $N = p \times q$, where p and q are odd primes and consider the uniform distribution over the instance space. The number of elements in $\mathbb{Z}_N^*$ is given by Euler's totient function $\varphi(N) = (p-1)(q-1)$ which counts the number of integers less than N that are relatively prime to N[4].

In number theory, an element $x \in \mathbb{Z}_N^*$ is called a *quadratic residue* or perfect square if there exists a $y \in \mathbb{Z}_N^*$ such that $x = y^2 \mod N$. We say that xRN if x is a quadratic residue. The task of identifying whether a given x is a quadratic residue is strongly conjectured to be classically difficult and forms the basis of very many cryptographic protocols. To date, there is also no known quantum algorithm that can solve it with exponential speedup [27]. From the Chinese remainder theorem, it is known a priori that of the $\phi(N)$ elements of $\mathbb{Z}_N^*$, exactly 25% are quadratic residues.

Naturally, the promise problem Π^{QRP} consists of the language of integers that are quadratic residues of N.

$$\begin{aligned}\Pi^{\text{QRP}} &= (\Pi_Y^{\text{QRP}}, \Pi_N^{\text{QRP}}) \\ \Pi_Y^{\text{QRP}} &= \{x :\ xRN\} \\ \Pi_N^{\text{QRP}} &= \{x :\ \neg xRN\}\ ,\end{aligned} \tag{21}$$

Figure 3a displays a fully labelled set of $\mathbb{Z}_N^*$ for $N = 35 = 5 \cdot 7$ according to the rule above. Note that the occasional gap indicates an integer that is not coprime with N and is thus not a member of the set. Also notice that exactly 25% of the data points are labelled as quadratic residues, reflecting the proportion of quadratic residues in $\mathbb{Z}_N^*$ derived earlier.

We furthermore use the definition of Π^{QRP} in Equation 21 to define the concept class $C^{\text{QRP}} =$

[3] up to division by y

[4] for example, $N = 15$ is coprime with $1, 2, 4, 7, 8, 11, 13, 14$, so $\varphi(15) = (3-1)(5-1) = 8$

Figure 3: **Use of the quantum feature map 22 to separate the yes/no (green/red) instances of Π^{OFP} for the special case of $N = 35$.** In dataset 3b, the superposition (shown in blue) obtained as $1 \mapsto |C_1\rangle$ sweeps all quadratic residues (*c.f.* Eq (22)), which corresponds to all the green dots $\in \Pi_Y^{\mathsf{QRP}}$. Analogous to 2b, the blue, green and red arcs indicate the normal to the hyperplane, the feature state of $4 \in \Pi_Y^{\mathsf{QRP}}$, and the feature state of $33 \in \Pi_N^{\mathsf{QRP}}$ respectively. Note that in all cases, the green arcs (feature states) overlap much more with the hyperplane than do the red arcs, indicating that the separating hyperplane has a wide margin.

$\{c^{\mathrm{QRP}}\}$ which comprises of the single concept:

$$c^{\mathrm{QRP}}(x) = [x \in \Pi_Y^{\mathrm{QRP}}]$$

This concept labels all inputs that are quadratic residues $'1'$ and otherwise $'0'$.

5.2 A quantum feature map for separating Π^{QRP}.

It is a well-known finding that Π^{QRP} belongs to SZK. As per **O.2**, we recall that there exists a reduction of Π^{QRP} to an instance of QS_δ. This reduction is derived in [2]:

$$x \mapsto (|C_1\rangle, |C_x\rangle) \quad \text{where} \sum_{r \in \mathbb{Z}_N^*} |r^2 x \mod N\rangle := |C_x\rangle \,. \tag{22}$$

Note that $|C_1\rangle$ remains is not dependent on x in this reduction and serves as a ground-truth hyperplane. However, compared to the feature map in Π^{DLP}, it seems less clear why $|C_1\rangle$ is a wide-margin hyperplane in the space $x \mapsto |C_x\rangle$. The next two theorems which are also proved in Ref [2] will illustrate this result:

Theorem 8. $x \in \Pi_Y^{QRP} \iff \forall r \in \mathbb{Z}_N^*, C_x(r) = r^2 x \in \Pi_Y^{QRP}$.

Theorem 9. $x \in \Pi_Y^{QRP} \implies \forall z \in \Pi_Y^{QRP} \exists r \in \mathbb{Z}_N^*$ *such that* $z = r^2 x \mod p$. *That is,* C_x *maps surjectively to the set of all quadratic residues* mod N.

From Theorem 8, it follows that if x is a quadratic residue, then so are all the basis states of $|C_x\rangle$. Additionally, from Theorem 9 we know that for every quadratic residue $z \in \Pi_Y^{\mathrm{QRP}}$, there will be a corresponding basis state $|C_x(r)\rangle = |z\rangle$. Therefore, $|C_x\rangle$ is a uniform superposition of all quadratic residues in $\mathbb{Z}_N^*$, $\sum_{x \in \Pi_Y^{\mathrm{QRP}}} |x\rangle$. But this is exactly the definition of $|C_1\rangle$:

$$|C_1\rangle = \sum_{r \in \mathbb{Z}_N^*} |r^2\rangle = \sum_{z \in \Pi_Y^{\mathrm{QRP}}} |z\rangle = |C_x\rangle \tag{23}$$

So, $|\langle C_1 | C_x\rangle|^2 = 1$ and the states overlap completely.

Alternatively, if x is a quadratic nonresidue, then by Theorem 8, neither is $r^2 x \, \forall r$, so that $|C_1\rangle$ and $|C_x\rangle$ share no overlap and $\langle C_1 | C_x\rangle = 0$. To summarise:

$$|\langle C_1 | C_x\rangle|^2 - \frac{1}{2} = \begin{cases} \frac{1}{2} & \forall x \in \Pi_Y^{\mathrm{QRP}} \\ -\frac{1}{2} & \forall x \in \Pi_N^{\mathrm{QRP}} \end{cases}. \tag{24}$$

It is now apparent that $(|C_1\rangle, -\frac{1}{2})$ forms a hyperplane with a wide margin of $\Delta = 1 = \mathcal{O}(1)$ that is constant in $n = \log N$.

5.2.1 Unresolved issues with C^{QRP}

O.3: Can the quantum feature map $x \mapsto |C_x\rangle$ be efficiently implemented? As in the case of Π^{DLP}, one approach to preparing the feature map $|C_x\rangle$ in Equation (22) relies on index erasure:

$$\sum_{i\in\mathbb{Z}_N^*} |i\rangle\,|i^2 \mod N\rangle \xrightarrow{U_{\text{uncomp}}} \sum_{i\in\mathbb{Z}_N^*} |i^2 \mod N\rangle \xrightarrow{U_\times} \sum_{i\in X} |i^2\cdot x \mod N\rangle = |C_x\rangle$$

While the modular multiplication step, $U_\times$, has a well-known efficient implementation, it is not clear how one would obtain U_{uncomp}. If we had an efficient modular root operator,

$$|0\rangle\,|i^2 \mod N\rangle \xrightarrow{U_\surd} (|i\rangle + |N-i\rangle)\,|i^2 \mod N\rangle\,, \tag{25}$$

then we could easily recover $U_{uncomp} = U_\surd^\dagger$. Indeed, an efficient operator $U_\surd$ would be able to solve the quadratic residue problem, the conjectured difficulty of which has served it well as a robust encryption protocol. Indeed, the QRP does not belong to the AHSP either, so finding an efficient quantum feature map for the state above on that basis is difficult. If we are willing to surrender exponential speedup, we may still obtain a quadratic improvement by using the general-purpose index erasure algorithm in algorithm 2. We propose algorithm 3 to compute U_{uncomp} with quadratic speedup $O(\sqrt{|\mathbb{Z}_n^*|})$ [24, 27].

O.1: Is PAC learning C^{QRP} classically difficult? First and foremost, it is uncertain whether $C^{\text{QRP}} \in \mathcal{C}$, i.e. the class of concept classes with efficient training set generation. For that to be the case, we require the following conjecture to be true

Conjecture 10. *An efficient algorithm exists for randomly generating labeled quadratic (non)residues $(x, c^{QRP}(x))$ where $x \sim U(\mathbb{Z}_n^*)$.*

Other than efficient training set generation, O.1 also demands that PAC learning C^{QRP} reduce to an AHSP or otherwise classically intractable problem. To this end, we once again require an aggregation property:

Conjecture 11. ***Π^{QRP} satisfies an aggregation property.*** *Given an oracle that correctly decides Π^{QRP} on at least $3/4 + \epsilon'\%$, there exists an efficient algorithm that correctly decides Π^{QRP} for all instances with high probability in time $\mathcal{O}\left(\frac{1}{\epsilon'}\right)$*

Note that the quantity $\frac{3}{4}$ appearing in the conjecture above is the best that a random guesser can achieve with only au priori knowledge of the $1:3$ distribution of quadratic residues to non-residues.

If similar to CITE, our approach cannot rely on randomly sampling $r_i \sim U(\mathbb{Z}_n^*)$ and then querying the oracle on the quantity yr_i^2 sufficiently many times. This is because the induced distribution is not uniform; If y is a quadratic residue, then not only is yr_i^2 also one by 8, but by 9 it is also uniformly distributed across all of them, that is, $yr_i^2 \sim U(\Pi_Y^{\text{QRP}})$, while for y that is not a quadratic residue, neither is yr_i^2 (note the distribution over Π_N^{QRP} may not be uniform). Hence, the outcome of the voting will differ based on which of the $3/4 + \epsilon'\%$ elements it correctly classifies.

In summary, Π^{QRP} admits a quantum feature space wherein it become linearly separable, but has no known efficient quantum feature map into said space. As for classical intractability, it is not known whether the training instances of C^{QRP} are efficiently generatable, nor whether efficiently PAC learning C^{QRP} is the reduction of an AHSP problem, leaving

6 A new classically intractable classification task based on order finding that circumvents index erasure

In the previous section, we introduced a concept class that reduced to a promise problem in SZK. By satisfying O.2, it was demonstrated that there exists a Hilbert space in which the instances of the concept class become separated by a wide-margin hyperplane. However, it is unknown whether there exists an efficient mapping into this space, or one that does not require index erasure. In fact, it is highly likely that such a mapping is difficult even for quantum computers as it would imply that the task of quadratic residuosity is in BQP, which is not known to be true. This raises a question: are there concept classes that are CIQT but importantly do not require index erasure?

In this section, we present a novel concept class C^{OFP} that is based on the famous problem of order finding and satisfies criteria **O.2**–**O.3** without relying on index erasure. We discuss the conditions under which C^{OFP} satisfies **O.1** and consequently can be said to be classically difficult to learn. Our main result will be proving that the promise problem Π^{OFP} which C^{OFP} reduces to belongs to SZK. As a consequence, there exists a quantum feature space in which the instances of C^{OFP} becomes linearly separable. Unlike Π^{QRP} in the previous section, we will discover that the quantum feature map for Π^{OFP} will be efficient to implement chiefly because it circumvents index erasure. Our promise problem is able to avoid index erasure because it belongs to a version of SZK with a quantum verifier.

In what follows, we introduce some elementary results from number theory that are necessary to define Π^{OFP}. Here, we also prove a new result about the distribution of odd and even orders of large semiprimes. After laying this foundation, we present the promise problem Π^{OFP} and show that it satisfies criteria **C.2** - **C.3** and possibly **C.1** which constitute our framework. We present a quantum feature map which separates the YES and NO instances of Π^{OFP} with a large margin and furthermore boasts an efficient implementation. To support our technical findings, we conduct simulations illustrating that various classical models struggle to learn the dataset induced by Π^{OFP} as effectively as the `QKE-SVM`.

6.1 Preliminaries

The Π^{OFP} promise problem will be largely based on the properties of multiplicative orders which we will now briefly define in this section. In the following, $N = pq$ is a product of two odd primes, also known as a semiprime.

Definition 6.1 (multiplicative order modulo N)**.** The multiplicative order (or simply the order) of an element $a \in \mathbb{Z}_N^*$ is defined as the smallest positive integer r satisfying $a^r = 1 \mod N$. We denote the multiplicative order by $r = ord_n(a)$.

We also state a well-known property of orders without proof

Theorem 12. *For all $a \in \mathbb{Z}_N^*$, the order $ord_N(a)$ divides the totient $\varphi(N) = (p-1)(q-1)$.*

Our promise problem will also make use of a property which we call the evenness:

Definition 6.2 (Evenness $\mathcal{E}$)**.** The *evenness* of an integer w, denoted by $\mathcal{E}(w)$, is the power of 2 that occurs in the factorisation of w. If $w = 2^a m$ is the partial factorisation of w, where $m \geq 1$ is the largest odd factor and $a \geq 0$, then by definition $\mathcal{E}(k) = a$.

We now state the following theorem which will serve as the main foundation for our concept class reducing to an SZK promise problem. The proof of this theorem can be found in section B of the Appendix.

Theorem 13. *If $N = pq$ is a semiprime with $\mathcal{E}(\varphi(N)) = a$, then the number of elements in $\mathbb{Z}_N^*$ with odd multiplicative orders is $\frac{\varphi(N)}{2^a}$.*

Corollary 13.1. *Fermat semiprimes which are defined as $S = p'q'$ where $p' = 2^{2^{k_1}} + 1$ and $q' = 2^{2^{k_2}} + 1$ are Fermat primes have no elements with odd orders except the trivial element 1.*

Proof. For a Fermat prime

$$\varphi(S) = (2^{2^{k_1}})(2^{2^{k_2}}) = 2^{2^{k_1}+2^{k_2}} = 2^{a'} \tag{26}$$

Thus, by Theorem 13, the number of elements with odd orders is $\varphi(S)\left(\frac{1}{2^{a'}}\right) = 1$ which can only be the trivial element 1. □

6.2 The C^{OFP} Concept Class

We consider an instance space consisting of all semiprimes with evenness either

Our concept class will contain a single concept $C^{\text{OFP}} = \{c^{\text{OFP}}\}$. For a fixed integer n, the instance space $\mathcal{X}_n$ will be all n-digit semiprimes satisfying either $\mathcal{E}(\phi(N)) = 2$ or $\mathcal{E}(\phi(N)) \geq 6$. Heuristic tests reveal that the ratio of these two classes of semiprimes is fixed at approximately $1 : 2.58$. We conjecture that just as there are infinitely many semiprimes, so are there infinitely many semiprimes of such variety. From here onwards, we will suppress the subscript n.

The concept $c^{OFP} : \mathcal{X} \mapsto \{\pm 1\}$ is defined as follows

$$c^{OFP}(N) = \begin{cases} +1, & \mathcal{E}(\phi(N)) = 2 \\ -1, & \mathcal{E}(\phi(N)) \geq 6 \end{cases} \tag{27}$$

PAC learning C^{OFP} reduces to solving the following promise problem defined as :

$$\Pi^{\text{OFP}} = (\Pi_Y^{\text{OFP}}, \Pi_N^{\text{OFP}}) \tag{28}$$
$$\Pi_Y^{\text{OFP}} = \left\{N \ : \ c^{OFP}(N) = +1\right\}$$
$$\Pi_N^{\text{OFP}} = \left\{N \ : \ c^{OFP}(N) = -1\right\}.$$

The labeling rule c^{OFP} assigns to an odd semiprime N a label based on the evenness of its totient $\varphi(N)$. Specifically, if $\varphi(N)$ is promised to have an evenness of exactly 2, we assign the label $+1$. Alternatively, if $\varphi(N)$ has an evenness of ≥ 6, we assign the label -1. It is worth noting that 2 represents the minimum evenness that $\varphi(N)$ can attain for a semiprime N.

6.3 Π^{OFP} is a potentially classically intractable promise problem learnable by a QKE-SVM

In this section, we examine Π^{OFP} in terms of the criteria **O.1**-**O.3** which comprise our framework for identifying CIQT concept classes. Specifically

6.3.1 **O.2**: Π^{OFP} belongs to SZK.

The formulation of Π^{OFP} might at first glance seem contrived: what is the significance of the evenness of the totient of a semiprime N? However, a look at theorem 13 quickly addresses this ambiguity: the value of $\mathcal{E}(\varphi(N))$ directly the proportions of elements with even or odd orders in $\mathbb{Z}_N^*$. In particular, the higher the evenness[5], the higher the proportion of elements with even orders. This relationship is summarised in the image of the following function that is parameterised by N:

$$\bar{C}_N : [0, N] \to [0, N] \times \{-1, 0, +1\},$$
$$i \mapsto (i, P(ord_N(i)))$$

$$\bar{C}_N(i) = \begin{cases} (i, \ \ 0) & \text{for } G := N - \varphi(N) \quad \text{elements} \\ (i, +1) & \text{for } B := \varphi(N)(1 - \frac{1}{2^a}) \quad \text{elements} \\ (i, -1) & \text{for } R := \varphi(N)(\frac{1}{2^a}) \quad \text{elements} \end{cases} \tag{29}$$

The proportion of blue (even), red (odd) and green (noncoprime) elements has been depicted in

[5]unless stated otherwise, by evenness we mean the evenness *of the totient of a semiprime*, which we omit from here onwards for brevity

Figure 4: **The distribution of elements which are noncoprime, or otherwise have either odd or even multiplicative orders modulo two semiprimes of evenness 2 and 4.** As proven in Theorem 13, the proportion of even orders is greater for the semiprime with the larger evenness. Note that the semiprimes were chosen to have similar proportion of noncoprimes to highlight theorem 13.

the bar graph in Figure 4 for two semiprimes with differing evenness. Observe that a semiprime of evenness 4 clearly has a greater proportion of even orders (in blue) than a semiprime with evenness 2.

By extension, the YES and NO instances of Π^{OFP} have an even greater difference in the distribution of their order parities. This property should sound familiar. We recall that all SZK problems reduce to STATISTICAL DIFFERENCE, in which a YES (NO) instance is a pair of functions whose induced distributions are largely different (similar). Suppose S is a NO instance. Then drawing from Figure 4, we might be tempted to claim that $\bar{C}_S$ and $\bar{C}_N$ form one such pair of functions whose induced distributions are largely dissimilar when N is YES and vice-a-versa:

$$|D_{\bar{C}_N} - D_{\bar{C}_S}| \overset{?}{>} \alpha \quad \forall N \in \Pi_Y^{\text{OFP}} \tag{30}$$
$$|D_{\bar{C}_N} - D_{\bar{C}_S}| \overset{?}{<} \beta \quad \forall N \in \Pi_N^{\text{OFP}}$$

for some $0 \leq \beta < \alpha^2 \leq 1$.

However, we must take precautions as the induced distribution $D_{\bar{C}_N}$ has values that depend on N and not solely the evenness. For example, note that

$$D_{\bar{C}_N}((i, j)) = \frac{1}{N} \text{ or } 0 \quad \forall i \tag{31}$$

This dependency makes the relations 30 also depend on the relative sizes of N and S. One

Figure 5: $|C_S\rangle$ **for any** $S \in \Pi_N^{\text{OFP}}$ **is the normal to a hyperplane that separates the instances of** Π^{OFP}. Each cell corresponds to the value of $C_{(\cdot)}(i)$ as represented in terms of RBG colours according to the same colour scheme used in Figure 4. Because N_{NO} and S have a higher evenness than N_{YES}, they have a higher proportion of blue values (even orders) as indicated by the histograms on the left. A given row may be interpreted as the superposition $|C\rangle$ produced by the feature map 33. In this interpretation, the inner product between any two feature states (rows) corresponds to the overlap of the colours. Here, the black lines represent colour mismatches between such pairs. Since $\langle C_{N_{\text{YES}}}|C_S\rangle$ has a greater proportion of mismatches than $\langle C_{N_{\text{NO}}}|C_S\rangle$, it has less overlap and thus a smaller inner product. In section C we show that the statistical difference in 34 are actually directly related to the inner products $\langle C_N|C_S\rangle$.

could imagine that for $S \gg N$, the differences become almost indiscernible between YES and NO instances because the support of $D_{\bar{C}_N}$ is much less than the support of $D_{\bar{C}_S}$. We fix this problem by making $\bar{C}_N$ periodic over an extended domain M. Since all the semiprimes in our instance space have n-digits, we take $M = k10^{n+1} > N$ which is parametrized in terms of a quantity k to be chosen later. This defines a periodic function related to $\bar{C}_N$:

$$\begin{aligned} C_N : [0, M] &\to [0, M] \times \{-1, 0, +1\}, \\ i &\mapsto (i, P(ord_N(i \mod N))) \end{aligned}$$

However, standardizing the domain from $[0, N]$ to $[0, M]$ comes at a cost: If N does not divide M, the proportions in Equation 55 of blue, red and green elements will be different from what they should be in the aperiodic case $\bar{C}_N$. Nevertheless, with sufficiently large k, the proportions of the periodic case can be brought arbitrarily close to those of the aperiodic case.

Now using C_N in place of $\bar{C}_N$, it can indeed be shown that for sufficiently large N.

$$\begin{aligned} |D_{C_N} - D_{C_S}| &\overset{!}{>} \alpha \quad \forall N \in \Pi_Y^{\text{OFP}} \\ |D_{C_N} - D_{C_S}| &\overset{!}{<} \beta \quad \forall N \in \Pi_N^{\text{OFP}} \end{aligned} \tag{32}$$

for some $0 \leq \beta < \alpha^2 \leq 1$. Thus,

Theorem 14. *The order-finding promise problem Π^{OFP} outlined in Equation 51 reduces to $SD_{\alpha\beta}$ and thus belongs to SZK with a quantum verifier for all semiprimes N with $\frac{\varphi(N)}{N} > 99\%$.*

We refer the reader to section C of the Appendix for the proof of the above theorem, among other relevant details and statistical insights into Π^{OFP}.

The quantum feature map separating Π^{OFP} Given that Π^{OFP} belongs to SZK and satisfies **O.2**, by our framework it follows that there exists a quantum feature space in which the instances become linearly separable (c.f. **C.2**). To identify this map, it is sufficient to derive the QUANTUM SAMPLING reduction of Π^{OFP}. Drawing on the $\text{SD}_{\alpha\beta}$ reduction of Π^{OFP} from Equation 32, the QUANTUM SAMPLING reductions asks for C_N as a quantum state:

$$\begin{aligned} N &\mapsto \sum_{i \in [0,M]} |i\rangle \, |P(ord_N(i \mod N))\rangle \\ &:= |C_N\rangle \ , \end{aligned} \tag{33}$$

From our discussion in the previous section, we saw that $|C_S\rangle$ where $h(S) = -1$ separated the YES and NO instances in terms of the statistical difference. Naturally, this gives rise to $|C_S\rangle$ as the canonical normal to the hyperplane. This is visualised for a small example in Figure 5. In section C, it is shown that the statistical differences in 32 are actually directly related to the inner products between the hyperplane and the feature states, making

$$|\langle C_N|C_S\rangle|^2 = (1 - |D_{C_N} - D_{C_S}|)^2 \begin{cases} \leq \alpha' := (1-\alpha)^2 & \forall x \in \\ \geq \beta' := (1-\beta)^2 & \forall x \in \end{cases} \tag{34}$$

In the appendix, we show how this margin can be widened further using a phase encoding instead of the basis encoding presented in 33.

6.3.2 **O.3** The quantum feature map separating Π^{OFP} is efficiently implementable

The hallmark of the feature map in Equation 33 compared to previous ones we have seen in this paper is its ability to retain the index. Since there exists an exact algorithm for order-finding, preparation of the state $|C_N\rangle\langle C_N|$ up to δ-error falls in BQP. This ensures that Π^{OFP} satisfies **O.3** and yields an efficient feature map implementation which we have described in Algorithm 1.

6.3.3 **O.1** PAC learning C^{OFP} is potentially classically intractable

It's uncertain whether PAC learning C^{OFP} is classically intractable. As with C^{QRP}, we require efficient training set generation to ensure $C^{\text{OFP}} \in \mathcal{C}$:

Conjecture 15. *An efficient algorithm exists for randomly generating ± 1 semiprimes $(N, c^{OFP}(N))$ where $N \sim U(\mathcal{X}_n)$.*

Other than efficient training set generation, **O.1** also demands that PAC learning C^{QRP} is the reduction of an AHSP or otherwise classically intractable problem. If it is true that

Conjecture 16. $\Pi^{OFP} \notin BPP$.

The in this case, an aggregation property would suffice:

Conjecture 17. Π^{OFP} ***satisfies an aggregation property.*** *Given an oracle that correctly decides Π^{OFP} on at least $72 + \epsilon'\%$, there exists an efficient algorithm that correctly decides Π^{OFP} for all instances with high probability in time $\mathcal{O}\left(\frac{1}{\epsilon'}\right)$*

Note that the quantity 72% appearing in the conjecture above is the best that a random guesser can achieve with only a priori knowledge of the $1 : 2.58$ ratio of $+1$ to -1 semiprimes.

Note that Π^{OFP} can be solved if we can efficiently compute the evenness of the totient $\varphi(N)$. If we have prior knowledge of $\varphi(N)$, it is possible to efficiently determine the evenness of $\varphi(N)$ in $\mathcal{O}(\log N)$ by repeatedly halving $\varphi(N)$ until we obtain a non-integer result. The calculation of $\varphi(N)$ itself is known to be equivalent to the factorization of N. Therefore, the task of computing the evenness of $\varphi(N)$ can be reduced to the factorization of N. This reduction implies that determining the evenness of $\varphi(N)$ is, at most, as difficult as factorizing integers, which is a challenging problem in classical computation and furthermore belongs to AHSP. Though it is not sufficient for PAC learning C^{OFP} to reduce to an AHSP problem. For **O.1** to be claimed, we require

---

Algorithm 1 Quantum feature map for Π^{OFP}

Input Working register initalised to $|0\rangle$. A large odd semiprime N. Also provided as input is the oracle $U_{\mathcal{P}}$ for checking parity

Output The quantum feature state $|C_N\rangle$

1: Generate a uniform superposition over the interval $[0, M]$:

$$|0\rangle \xrightarrow{H^{\otimes \lceil \log(M+1) \rceil}} \sum_{i=0}^{2^{\lceil \log(M+1) \rceil}} |i\rangle \xrightarrow{U_{AA}} \sum_{i=0}^{M} |i\rangle \tag{35}$$

2: Mark integers coprime with N using a similar method to Ref. [14] for generating a prime state:

$$\sum_{i=0}^{M} |i\rangle \xrightarrow{U_{\text{gcd}}} \sum_{i=0}^{M} |i\rangle \overbrace{|gcd(i \mod N, N)\rangle}^{d_i}$$
$$\xrightarrow{U_{=1?}} \sum_{i=0}^{M} |i\rangle |d_i\rangle |\delta_{d_i,1}\rangle$$

3: Apply the derandomised algorithm for order-finding conditioned on the third register. Note that this step has a δ chance of failure.

$$\xrightarrow{C-U_{\text{ord}}} \sum_{i=0}^{M} |i\rangle |d_i\rangle |\delta_{d_i,1}\rangle \overbrace{|ord_N(i \mod N)\rangle}^{o_i} \tag{36}$$

4: Apply $U_{\mathcal{P}}$ to the last register o_i which calculates the parity of the orders:

$$\xrightarrow{U_{\mathcal{P}}} \sum_{i=0}^{M} |i\rangle |d_i\rangle |\delta_{d_i,1}\rangle |o_i\rangle |C_N(i)\rangle \tag{37}$$

5: Uncompute all but the first and last registers:

$$\xrightarrow{U^\dagger_{\text{gcd}} U^\dagger_{=1?} C-U^\dagger_{\text{ord}}} \sum_{i=0}^{M} |i\rangle |C_N(i)\rangle = |C_N\rangle \tag{38}$$

---

the converse to be true: that an AHSP problem reduces to PAC learning C^{OFP}.

If determining the evenness of $\varphi(N)$ is indeed as difficult as factorization, then it is strongly conjectured to be classically intractable. However, it is likely easier since calculating $\mathcal{E}(\varphi(N))$ only provides information about the exponent of 2 in the prime factorization of $\varphi(N)$, and not the exponents of other prime factors which would be required to fully reconstruct $\varphi(N)$. From theorem 13, we know that $\mathcal{E}(\varphi(N))$ directly determines the relative proportion of elements with odd or even orders. If there was a way to randomly sample multiplicative orders modulo N, then we could with high probability decide Π^{OFP}. Still, this would not be an exact method.

Ultimately, whether the evenness of $\varphi(N)$ can be accurately calculated using classical methods without explicitly computing $\varphi(N)$ itself remains an unsolved problem in the literature. Based on this, we conjecture that determining the evenness of $\varphi(N)$ is classically intractable. By the same token, we make the assumption that Π^{OFP}, which involves the even easier task of determining whether the evenness of $\varphi(N)$ is greater than or equal to 6 or equal to 2 is also classically intractable.

6.4 Results and simulations

6.5 Discussion

In Figure 6a, we see the distribution of accuracies obtained by classical machine learning models compared to that of the `QKE-SVM` in classifying data points from the order-finding promise problem Π^{OFP}. Observe that the classical models collectively achieve an average accuracy of $\mu_c = 56\%$. Since the dataset used for training had 65% YES instances, this indicates that the classical models performed no better than mere random guessing, suggesting the classical intractability of the Π^{OFP}. In stark contrast to this, the `QKE-SVM` attains a perfect average accuracy of $\mu_c = 100\%$ and with much lower variance than the classical models. The fact that the `QKE-SVM` has been able to efficiently solve the Π^{OFP} to such high accuracy demonstrates quantum speedup as one could imagine the classical models requiring exponentially more training instances to be able to perform comparably well.

To see why there is such a wide gap in performance, it is instructive to compare the `QKE-SVM` with its classical counterpart, the `Linear-SVM`. Figures 6b and 6c are the kernel matrices used by the `Linear-SVM` and `QKE-SVM` respectively. The instances are sorted with respect to labels, so that the first 100 instances are labelled NO, and the remainder are labelled YES‘. Recall that each entry corresponds to an inner product $K_{\mathcal{V}}(N_1, N_2)$ between two semprimes N_1 and N_2 as represented in some feature space $\mathcal{V}$. For the linear SVM, this feature space is merely the original space, and $K_{\text{classical}}(N_1, N_2) = N_1 N_2$. The corresponding kernel matrix 6b reveals no structure that is correlated with the class labels as the cross-hatching patterns appear randomly distributed. From this we can infer that this simple feature map used by the linear SVM cannot linearly separate between the YES and NO instances. Indeed, it is a feature of the Π^{OFP} that no feature map efficiently implementable by a classical computer can linearly separate the dataset (*c.f.* the condition of classical intractability **O.1**). On the other hand, the quantum kernel matrix 6c used by the `QKE-SVM` reveals entry values that are clearly correlated with their class labels. The bright square occupying the entire fourth quadrant in contrast to the relatively darker patch of the first and third quadrants illustrates that in the quantum feature space (**??**), NO instances are more similar than to each other than they are to NO instances. Thus, there exists a wide-margin hyperplane (*c.f.* Equation (**??**)) that linearly separates the dataset. The difference between these two kernel matrices explains why the `QKE-SVM` is able to drastically outperform `Linear-SVM`.

6.6 Summary

We have designed a new classically intractable promise problem based on order finding called Π^{OFP} which admits quantum speedup. We argued that the Π^{OFP} is classically intractable as it is unlikely to be solvable without first performing the classically intractable task of integer factorisation which is equivalent to order finding. We then showed that the Π^{OFP} belongs to SZK and thereby admits a linearly separable representation in some quantum feature space. Then, we implemented the quantum feature map into this space, which was made efficient due to its not requiring index erasure. Through simulations, we evaluated the performance of classical models on

Figure 6: **Smoothed histograms displaying several classical machine learning models being drastically outperformed by the QKE-SVM at the task of Π^{OFP} (6a). 6b and 6c respectively show kernel matrices corresponding to the classical and quantum feature maps that were used, revealing that the latter clearly separates instances based on their binary label unlike the former.** The bottom panel of 6a summarises the average accuracy of each model flanked by a σ error bar. The dataset on which the models were trained had a 65% YES instances, thus the classical models on average performed no better than random guessing ($\mu_c = 56\%$) suggesting the classical intractability of the Π^{OFP}.

the Π^{OFP} against that of QKE-SVM and saw that while classical models can perform no better than random guessing, the QKE-SVM achieves a near perfect accuracy.

7 Conclusion and outlook

We have provided the only other instance of a classically intractable promise problem, the Π^{OFP}, which can be efficiently solved by a quantum machine learning model, the QKE-SVM. More broadly, we have provided a general framework by which new classically intractable promise prob-

lems admitting quantum speedup can be constructed.

We obtained this framework by deconstructing Liu *et al.'s* Π^{DLP} and identifying three sufficient properties that render a promise problem Π classically intractable all while solvable by the `QKE-SVM`:

1. Π belongs to the class of Statistical Zero Knowledge (SZK) problems. This induces a quantum feature space in which the promise problem becomes linearly separable.

2. The Quantum Sampling reduction of Π belongs to the class of Exact Quantum Polynomial-time (EQP) algorithms. This ensures the quantum feature map can be implemented efficiently and exactly using a quantum computer.

3. Π reduces to an Abelian Hidden Subgroup Problem (AHSP) which confers the promise problem potential classical intractability[6].

In arriving at Π^{OFP}, we investigated a counterexample based on quadratic residuosity, Π^{QRP}, which despite belonging to SZK and being strongly conjectured to be classically intractable, failed to admit an efficient exact implementation of its corresponding quantum feature map. Along the way, we also provided a quadratic quantum speedup for the Π^{QRP} by designing an algorithm for the task of index erasure.

Future research can be based on constructing more novel promise problems using the framework above. The task of 'engineering' promise problems is admittedly highly contrived, especially when the field of machine learning concerns practical promise problems obtained from the real world, such as tumour classification or image recognition. Therefore, future work should be dedicated to using the framework above as a diagnostic tool for identifying *real world* — and not merely mathematical — datasets that can uniquely be solved by quantum computers.

8 Acknowledgements

We extend our heartfelt gratitude to Prof. Charles Hill and Sam Sutherland for their unwavering support and relentless dedication in meticulously verifying the proofs of the theorems presented in Section C and the intricate statistical analysis of Π^{OFP}. Their invaluable contributions have significantly enhanced the quality and rigor of this work.

[6] unless it reduces to a problem in **P**

References

[1] Experimental quantum kernel trick with nuclear spins in a solid | npj Quantum Information. URL https://www.nature.com/articles/s41534-021-00423-0.

[2] Dorit Aharonov and Amnon Ta-Shma. Adiabatic Quantum State Generation and Statistical Zero Knowledge, January 2003. URL http://arxiv.org/abs/quant-ph/0301023. arXiv:quant-ph/0301023.

[3] M. Cerezo, Guillaume Verdon, Hsin-Yuan Huang, Lukasz Cincio, and Patrick J. Coles. Challenges and opportunities in quantum machine learning. *Nature Computational Science*, 2(9):567–576, September 2022. ISSN 2662-8457. DOI: 10.1038/s43588-022-00311-3. URL https://www.nature.com/articles/s43588-022-00311-3. Number: 9 Publisher: Nature Publishing Group.

[4] Iris Cong and Luming Duan. Quantum Discriminant Analysis for Dimensionality Reduction and Classification. *New Journal of Physics*, 18(7):073011, July 2016. ISSN 1367-2630. DOI: 10.1088/1367-2630/18/7/073011. URL http://arxiv.org/abs/1510.00113. arXiv:1510.00113 [quant-ph].

[5] Vittorio Giovannetti, Seth Lloyd, and Lorenzo Maccone. Quantum random access memory. *Physical Review Letters*, 100(16):160501, April 2008. ISSN 0031-9007, 1079-7114. DOI: 10.1103/PhysRevLett.100.160501. URL http://arxiv.org/abs/0708.1879. arXiv:0708.1879 [quant-ph].

[6] Jennifer R. Glick, Tanvi P. Gujarati, Antonio D. Corcoles, Youngseok Kim, Abhinav Kandala, Jay M. Gambetta, and Kristan Temme. Covariant quantum kernels for data with group structure, March 2022. URL http://arxiv.org/abs/2105.03406. arXiv:2105.03406 [quant-ph].

[7] Oded Goldreich. *Providing sound founda-*

tions for cryptography: on the work of Shafi Goldwasser and Silvio Micali. ACM, 2019.

[8] Casper Gyurik and Vedran Dunjko. Exponential separations between classical and quantum learners, June 2023. URL http://arxiv.org/abs/2306.16028. arXiv:2306.16028 [quant-ph].

[9] Vojtěch Havlíček, Antonio D. Córcoles, Kristan Temme, Aram W. Harrow, Abhinav Kandala, Jerry M. Chow, and Jay M. Gambetta. Supervised learning with quantum-enhanced feature spaces. *Nature*, 567(7747): 209–212, March 2019. ISSN 0028-0836, 1476-4687. DOI: 10.1038/s41586-019-0980-2. URL http://www.nature.com/articles/s41586-019-0980-2.

[10] R. Jozsa. Quantum factoring, discrete logarithms, and the hidden subgroup problem. *Computing in Science amp; Engineering*, 3(2):34–43, 2001. ISSN 1521-9615. DOI: 10.1109/5992.909000. URL http://dx.doi.org/10.1109/5992.909000.

[11] Jonas Jäger and Roman V. Krems. Universal expressiveness of variational quantum classifiers and quantum kernels for support vector machines, July 2022. URL http://arxiv.org/abs/2207.05865. arXiv:2207.05865 [quant-ph].

[12] Jonas Jäger and Roman V. Krems. Universal expressiveness of variational quantum classifiers and quantum kernels for support vector machines. *Nature Communications*, 14(1):576, February 2023. ISSN 2041-1723. DOI: 10.1038/s41467-023-36144-5. URL https://www.nature.com/articles/s41467-023-36144-5. Number: 1 Publisher: Nature Publishing Group.

[13] Michael J Kearns and Umesh Vazirani. *An introduction to computational learning theory*. MIT press, 1994.

[14] Jose I Latorre and German Sierra. Quantum computation of prime number functions. *arXiv preprint arXiv:1302.6245*, 2013.

[15] Yunchao Liu, Srinivasan Arunachalam, and Kristan Temme. A rigorous and robust quantum speed-up in supervised machine learning. *Nature Physics*, 17(9):1013–1017, September 2021. ISSN 1745-2473, 1745-2481. DOI: 10.1038/s41567-021-01287-z. URL https://www.nature.com/articles/s41567-021-01287-z.

[16] Seth Lloyd, Masoud Mohseni, and Patrick Rebentrost. Quantum principal component analysis. *Nature Physics*, 10(9): 631–633, September 2014. ISSN 1745-2473, 1745-2481. DOI: 10.1038/nphys3029. URL http://arxiv.org/abs/1307.0401. arXiv:1307.0401 [quant-ph].

[17] Seth Lloyd, Maria Schuld, Aroosa Ijaz, Josh Izaac, and Nathan Killoran. Quantum embeddings for machine learning, February 2020. URL http://arxiv.org/abs/2001.03622. arXiv:2001.03622 [quant-ph].

[18] Michele Mosca and Christof Zalka. Exact quantum Fourier transforms and discrete logarithm algorithms, January 2003. URL http://arxiv.org/abs/quant-ph/0301093. arXiv:quant-ph/0301093.

[19] M.A. Nielsen and I.L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. Cambridge University Press, 2010. DOI: 10.1017/CBO9780511976667.

[20] Amit Sahai and Salil Vadhan. A complete problem for statistical zero knowledge. *Journal of the ACM (JACM)*, 50(2):196–249, 2003.

[21] Teresa Sancho-Lorente, Juan Román-Roche, and David Zueco. Quantum kernels to learn the phases of quantum matter. *Physical Review A*, 105(4):042432, April 2022. DOI: 10.1103/PhysRevA.105.042432. URL https://link.aps.org/doi/10.1103/PhysRevA.105.042432. Publisher: American Physical Society.

[22] Nisheeth Saxena and Akriti Nigam. Utilizing non linear functions based quantum kernels to categorize small datasets. In *2023 IEEE 3rd International Conference on Sustainable Energy and Future Electric Transportation (SEFET)*, pages 1–6, 2023. DOI: 10.1109/SeFeT57834.2023.10245485.

[23] Maria Schuld. Supervised quantum machine learning models are kernel methods, January 2021. URL https://arxiv.org/abs/2101.11020v2.

[24] Yaoyun Shi. Quantum lower bounds for the collision and the element distinctness problems. In *The 43rd Annual IEEE Symposium on Foundations of Computer Science, 2002. Proceedings.*, pages 513–519, 2002. DOI:

10.1109/SFCS.2002.1181975. URL http://arxiv.org/abs/quant-ph/0112086. arXiv:quant-ph/0112086.

[25] Apimuk Sornsaeng, Ninnat Dangniam, Pantita Palittapongarnpim, and Thiparat Chotibut. Quantum diffusion map for nonlinear dimensionality reduction. *Physical Review A*, 104(5):052410, November 2021. ISSN 2469-9926, 2469-9934. DOI: 10.1103/PhysRevA.104.052410. URL http://arxiv.org/abs/2106.07302. arXiv:2106.07302 [quant-ph].

[26] Ewin Tang. Quantum principal component analysis only achieves an exponential speedup because of its state preparation assumptions. *Physical Review Letters*, 127(6): 060503, 2021.

[27] Hong-Fu Wang, Shou Zhang, Yong-Fang Zhao, and Kyu-Hwang Yeon. Quantum Mechanical Algorithm for Solving Quadratic Residue Equation. *International Journal of Theoretical Physics*, 48(11):3262–3267, November 2009. ISSN 0020-7748, 1572-9575. DOI: 10.1007/s10773-009-0115-x. URL http://link.springer.com/10.1007/s10773-009-0115-x.

A Algorithm for index erasure with quadratic speedup

Algorithm 2 Index Erasure

Input The initial state $\sum_{i\in X} |i\rangle f(i)$. An oracle U_f with action $|i\rangle |0\rangle \xrightarrow{U_f} |i\rangle |f(i)\rangle$ implementing a one-way function $f : X \to Y$ that is M-to-one. An oracle $|x\rangle |y\rangle |q\rangle \xrightarrow{U_g} |x\rangle |y\rangle |q \oplus g_f(x, y)\rangle$ implementing the verifier $g_f(x, y) = [y = f(x)]$. A unitary operator U_X producing a uniform superposition over the input states.

Output The state $\sum_{i\in X} |f(i)\rangle$.

Note For clarity, the algorithm will begin with the output first from which the initial state will be reversibly reconstructed.

1:
$$\sum_{i\in X} |f(i)\rangle |0\rangle \xrightarrow{U_X} \sum_{i\in X}\sum_{j\in X} |f(i)\rangle |j\rangle$$

2: Up to the addition and removal of a third register set to $|+\rangle$, obtain:
$$\xrightarrow{U_g} \sum_{i\in X}\sum_{j\in X} (-1)^{g_f(f(i),j)} |f(i)\rangle |j\rangle$$

Note that out of the $|X||Y|$ states, $|X|$ will be 'marked' with a negative phase, so the success probability of their measurement is $1/|Y| = M/|X|$.

3: Perform amplitude amplification on the previous state:
$$\xrightarrow{U_{AA}} \sum_{i\in X} |f(i)\rangle |i\rangle$$

4: Note that following the steps (1)-(3) but in reverse yields the desired operation $U_{UNCOMP} := (U_X U_g U_{AA})^\dagger$.

A.1 Index erasure for implementing the quantum feature map in the quadratic residuosity promise problem Π^{OF}

We can use Algorithm 2 to solve the state generation problem for the quadratic residuosity promise problem with quadratic speedup, as shown in 3.

B Proof of Theorem 13

Proof. We begin by first highlighting that $\mathbb{Z}_n^*$ is

- Finite as $|\mathbb{Z}_n^*| = \varphi(n)$ which is well-defined; and
- Abelian, as modular multiplication is commutative: $a \times b \mod n = b \times a \mod n$.

Since $\mathbb{Z}_n^*$ is a finite abelian group, by the Converse Lagrange Theorem (CLT), for all $k \,|\, \varphi(n)$, there exists at least one subgroup $H \leq \mathbb{Z}_n^*$ whose order is exactly k. In particular, since $m|\varphi(n) = 2^a m$, then there exists a subgroup H of order m. Now suffice it to prove that

$$x \in \mathbb{Z}_n^* \text{ has an odd order } \iff x \in H \leq \mathbb{Z}_n^* \text{ where } |H| = m, \text{ which exists by the CLT} \tag{43}$$

$\Leftarrow$ If $x \in H$, then the subgroup $\langle x\rangle$ generated by x must also be contained in H:

$$x \in H \implies \langle x\rangle \leq H \tag{44}$$

Algorithm 3 Quadratic Residuosity state generation

Input Working register initalised to $|0\rangle$, an element $x \in \mathbb{Z}_N^*$, and the quantum oracle U_f implementing $f(x) = x^2 \mod N$.

Output The state $|C_x\rangle$.

1: Generate a uniform superposition over the elements of $\mathbb{Z}_N^*$ using the method of Ref. [?] for generating a prime state:

$$|0\rangle \xrightarrow{H^{\otimes n}} \sum_{i=0}^{N-1} |i\rangle \xrightarrow{U_{\text{gcd}}} \sum_{i=0}^{N-1} |i\rangle\, |gcd(i,N)\rangle \xrightarrow{U_{\text{AA}}} \sum_{i\in\mathbb{Z}_N^*} |i\rangle\, |1\rangle \equiv \sum_{i\in\mathbb{Z}_N^*} |i\rangle \tag{39}$$

Note that amplitude amplification is possible because the success probability of i and N being coprime is known a priori to be $\phi(N)/N = (p-1)(q-1)/pq \approx 1$ for large p, q [18].

2:

$$\xrightarrow{U_f} \sum_{i\in\mathbb{Z}_N^*} |i\rangle\, |i^2 \mod N\rangle \tag{40}$$

3: Using the technique of Algorithm 2

$$\xrightarrow{U_{\text{UNCOMP}}} \sum_{i\in X} |i^2 \mod N\rangle \tag{41}$$

4: Modular multiplication given by

$$\xrightarrow{U_\times} \sum_{i\in X} |i^2 \cdot x \mod N\rangle = |C_x\rangle \tag{42}$$

The number of elements in the $\langle x\rangle$ is precisely the order of x, $ord_n(x)$. By Lagrange's theorem, this generated subgroup must divide the group H

$$|\langle x\rangle| = ord_n(x) \mid |H| = m \tag{45}$$

But since m is odd, so must $ord_n(x)$ [?].

$\Rightarrow$ From the Chinese Remainder Theorem,

$$\mathbb{Z}_n^* \cong \mathbb{Z}_p^* \times \mathbb{Z}_q^*$$

Since p and q are prime numbers,

$$\cong C_{\varphi(p)} \times C_{\varphi(q)} \cong C_{p-1} \times C_{q-1}$$

By the fundamental theorem of arithmetic, rewrite $p-1$ and $q-1$ in terms of their prime factorisations $p-1 = 2^{a_1} p_1^{k_1} \dots p_i^{k_i}$, $q-1 = 2^{a_2} q_1^{l_1} \dots q_j^{l_j}$ so that

$$\varphi(n) = (p-1)(q-1) = (2^{a_1} p_1^{k_1} \dots p_i^{k_i})(2^{a_2} q_1^{l_1} \dots q_j^{l_j}) \tag{46}$$

$$= 2^{a_1} 2^{a_2} p_1^{k_1} \dots p_i^{k_i} q_1^{l_1} \dots q_j^{l_j} \overset{!}{=} 2^a m \tag{47}$$

It is a well-known fact that $C_{rs} \cong C_r \times C_s$ for coprime r and s. Using this result along with the factorisation in 47, we can further simplify 46 as

$$\begin{aligned} C_{p-1} \times C_{q-1} &\cong C_{2^{a_1} p_1^{k_1} \dots p_i^{k_i}} \times C_{2^{a_2} q_1^{l_1} \dots q_j^{l_j}} \\ &\cong C_{2^{a_1}} \times C_{p_1^{k_1}} \dots \times \dots C_{p_i^{k_i}} \times C_{2^{a_2}} \times C_{q_1^{l_1}} \dots \times \dots C_{q_j^{l_j}} \end{aligned}$$

which up to reordering is

$$\cong C_{2^{a_1}} \times C_{2^{a_2}} \times C_{p_1^{k_1}} ... \times ... C_{p_i^{k_i}} \times C_{q_1^{l_1}} ... \times ... C_{q_j^{l_j}} \,.$$

Note that a cyclic group C_{2^k} only has elements whose orders divide 2^k and thus are even, except for the trivial element $1 \in C_1 \leq C_{2^k}$:

$$x \in C_{2^k} \text{ has odd order } \iff x = 1 \in C_1 \leq C_{2^k} \tag{48}$$

So if $x \in \mathbb{Z}_n^*$ has odd order, then

$$x \overset{!}{\in} C_1 \times C_1 \times C_{p_1^{k_1}} ... \times ... C_{p_i^{k_i}} \times C_{q_1^{l_1}} ... \times ... C_{q_j^{l_j}} \leq \mathbb{Z}_n^* \,. \tag{49}$$

But observe that the second term is a subgroup of order $1 \cdot 1 \cdot p_1^{k_1} ... p_i^{k_i} q_1^{l_1} ... q_j^{l_j} = m$, which is precisely the subgroup H

From the above proof, it follows that of the $\varphi(n)$ elements in $\mathbb{Z}_n^*$ there are $m-1$ elements with odd orders, excluding the trivial element 1 which has order 1. Thus, the proportion of such elements is

$$\frac{m-1}{\varphi(n)} = \frac{m-1}{2^a m} = \frac{1}{2^a}(1 - \frac{1}{m}) \,. \tag{50}$$

□

C Mathematical properties of OF_P^2

Suppose $N = pq \in \mathbb{Z}$ is a large semiprime. In OF_P^2 , an instance N is labeled according to the rule:

$$h(N) = \begin{cases} +1, & \mathcal{E}(\phi(N)) = 2 \\ -1, & \mathcal{E}(\phi(N)) \geq 9 \end{cases} \tag{51}$$

Put simply, h labels all odd semiprimes N whose corresponding totient $\varphi(N)$ is promised to attain an evenness of exactly 2, in which case it is labelled $+1$, or an evenness of ≥ 9, in which case it is labelled -1. Note that 2 is the minimum evenness that $\varphi(N)$ can attain for a semiprime N since

$$N = pq \implies \varphi(N) = (p-1)(q-1)$$

since N is odd, so are p, q and thus $p-1$, $q-1$ are even

$$\begin{aligned} &\implies \exists k_1, k_2 \in \mathbb{Z} \; \varphi(N) = (2k_1)(2k_2) = 2^2(k_1 k_2) = 2^{2+a} m, \; a \in \mathbb{Z}_{\geq 0} \text{ and } m \text{ odd} \\ &\implies \mathcal{E}(\varphi(N)) = 2 + a \geq 2 \end{aligned}$$

So, the $+1$-labelled semiprimes are those whose $\varphi(N)$ has a minimum evenness (of 2).

As a promise problem, the OF_P^2 is formulated as the language Π:

$$\begin{aligned} \Pi &= (\Pi_Y, \Pi_N) \\ \Pi_Y &= \{N \; : \; h(N) = +1\} \\ \Pi_N &= \{N \; : \; h(N) = -1\} \end{aligned} \tag{51}$$

C.1 Conjectured classical intractability

If $\varphi(N)$ is known a priori then $\mathcal{E}(\varphi(N))$ can be efficiently determined in $\mathcal{O}(\log N)$ steps by repeatedly halving $\varphi(N)$ until a non-integer result is obtained. It is also well known that computing $\varphi(N)$ given N is equivalent to factorising N itself. As such, the task of computing the evenness of $\varphi(N)$ has the following reduction:

$$\begin{aligned}\text{Compute } \mathcal{E}(\varphi(N)) &\xrightarrow{\text{Reduces to}} \text{Compute } \varphi(N) = (p-1)(q-1) \\ &\xrightarrow{\text{Equivalent to}} \text{Factorise } N = pq.\end{aligned}$$

From this we can conclude that OF_P^2 is *at most* as hard as factorising integers, which is classically difficult. If it is in fact *as hard*, then OF_P^2 is also classically intractable. However, it is much likelier that it is *easier* as $\mathcal{E}(\varphi(N))$ only yields the exponent of 2 in the prime factorisation of $\varphi(N)$ and not that of any other prime factors. Still, it is not known whether the evenness of $\varphi(N)$ can be classically calculated without explicitly requiring $\varphi(N)$ itself. In light of this, we conjecture as to the classically intractability of the OF_P^2.

C.2 OF_P^2 is an instance of SZK

This promise problem over semiprimes is actually a specific instance of Statistical Difference or $\mathrm{SD}_{\alpha\beta}$ which you will recall belongs to SZK. We now provide some groundwork before we prove this claim.

C.2.1 Preliminaries

Fix a semiprime integer N whose totient has the partial factorisation $\phi(N) = 2^a m$ where m is an odd integer and $a \geq 2$. To show that $OF_P^2 \in \mathrm{SD}_{\alpha\beta}$, we will show that there exists a function parameterised by semiprimes whose image for a fixed semiprime N varies drastically depending on whether that semiprime is a $+1$ or -1 instance.

This function will make use of an important quantity, $P\big(ord_N(i)\big)$ which calculates the parity of the order of a number i with respect to a semiprime $N \geq i$. If i is not coprime with N, we use the convention that $ord_N(i) = P(ord_N(i)) = 0$. Given that this quantity can only take three possible values, we will at times use a colour analogy denoting $\{-1, 0, +1\}$ with the colours Red, Green and Blue respectively.

Having defined $P\big(ord_N(i)\big)$, we now introduce the function $\bar{C}_N$ which will bring us one step closer to the final function C_N:

$$\bar{C}_N : [1, N] \to [1, N] \times \{-1, 0, +1\}, \quad i \mapsto \big(i, P\big(ord_N(i)\big)\big) \tag{52}$$

In our proof, we will ultimately care about the global properties of $\bar{C}_N$. In particular, we will require the number of elements in the domain of $\bar{C}_N$ that map to the same value. This can be calculated using our earlier analysis in **??**:

$$\bar{C}_N(i) = \begin{cases} (i, \quad 0) & \text{for } G := N - \varphi(N) \quad \text{elements} \\ (i, +1) & \text{for } B := \varphi(N)(1 - \frac{1}{2^a}) \quad \text{elements} \\ (i, -1) & \text{for } R := \varphi(N)(\frac{1}{2^a}) \quad \text{elements} \end{cases} \tag{53}$$

For reasons that will be explained later, we will make a slight change to $\bar{C}_N$ by extending its domain. Suppose it is known that the largest semiprime in the domain of our labeling rule h is $N_{\max}$ and fix a large multiple $M = kN_{\max} > N$ for some large positive integer k, the choice of which will be discussed later. Now consider the function

$$C_N : [1, M] \to [1, M] \times \{-1, 0, +1\}, \quad i \mapsto \big(i, P\big(ord_N(i \mod N)\big)\big) \tag{54}$$

Figure 7: **Convergence of dashed proportions c' of marble colours to their undashed counterparts c with increasing repetitions k.**

which is essentially a periodic version of $\bar{C}_N$. Specifically, if $M = m_N N + r$ for positive integers m_N and r_N, the image of C_N is that of $\bar{C}_N$ repeated in full m_N times across the interval $[1, m_N N]$ as well as one more round extending only over the first r_N elements. It will be important to track how this change will affect the global properties of the function. It is evident the distribution of elements calculated for the aperiodic version in Equation 53 will only be multiplied m_N times along with some remainder:

$$C_N(i) = \begin{cases} (i, \quad 0) & \text{for } G' := m_N G + r_G \quad \text{elements} \\ (i, +1) & \text{for } B' := m_N B + r_B \quad \text{elements} \\ (i, -1) & \text{for } R' := m_N R + r_R \quad \text{elements} \end{cases} \tag{55}$$

where $r_N = r_R + r_B + r_G < R + B + G = N$. While it may seem like the global properties of the two functions differ greatly, because we will care most about their relative proportions we find that k and by extension m_N can be chosen such that the differences become asymptotically small. For example, the proportion of green (0) instances in C_N and $\bar{C}_N$ differs by

$$\begin{aligned} \left| \frac{G}{N} - \frac{m_N G + r_G}{m_N N + r_N} \right| &= \left| \frac{G}{N} - \left(\frac{G}{N} - \frac{r_N G/N - r_G}{m_N N + r_N} \right) \right| \\ &= \left| \frac{r_N G/N - r_G}{m_N N + r_N} \right| \\ &\leq \begin{cases} \frac{r_N G/N}{m_N N} \leq \frac{r_N}{m_N N} & r_G < 2 r_N G/N \\ \frac{r_G}{m_N N} & r_G > 2 r_N G/N \end{cases} \\ &\leq \frac{1}{m_N} \leq \frac{1}{k} \end{aligned}$$

Thus, given ϵ error, it suffices to select $k > \frac{1}{\epsilon}$ and $k = \mathcal{O}(\frac{1}{\epsilon})$. From now on, we will assume that the proportion of the domain of C_N mapping to the same colour is at most ϵ away from that of $\bar{C}_N$: for example, $\left| \frac{G}{N} - \frac{G'}{M} \right| = \epsilon$. This relationship is demonstrated for a specific instance if Figure 8.

In our discussion so far of proportions of domains that map to the same value we have implicitly made references to the concept of the probability distribution D_f induced by a function f, which is defined as

$$D_f : \mathrm{Im}(f) \to [0, 1], \quad x \mapsto \frac{|f^{-1}(\{x\})|}{|\mathrm{dom}(f)|} \tag{56}$$

In words, $D_f(x)$ refers to the proportion of inputs i in the domain of f whose image is $f(i) = x$. $\mathrm{SD}_{\alpha\beta}$ is the promise problem defined as

$$\begin{aligned} \Gamma &= (\Gamma_{\mathrm{Y}}, \Gamma_{\mathrm{N}}) \\ \Gamma_{\mathrm{Y}} &= \{(f, g) \ : \ |D_f - D_g| > \alpha\} \\ \Gamma_{\mathrm{N}} &= \{(f, g) \ : \ |D_f - D_g| < \beta\} \end{aligned} \tag{57}$$

												Total
G	0	0	0	0	0	0	1	0	0	0	0	1
R	0	0	0	0	1	1	0	1	0	0	0	3
B	0	0	0	0	1	1	1	1	0	0	0	4

$\#\ mismatches = \frac{1}{2}(R+B+G)$	4
$\mathcal{S} = 1 - \lvert\langle C_S \rvert C_N\rangle\rvert = \frac{\#\ mismatches}{M}$	0.4

Figure 8: **Calculation of the statistical difference between the order spectrum of two semiprimes literally corresponds to counting the number of colour mismatches between those two arrays.**

where $0 \leq \beta \leq \alpha \leq 1$ satisfying $\alpha^2 \geq \beta$.

Here, the statistical difference is defined as:

$$|D_f - D_g| = \frac{1}{2} \sum_{i \in \mathrm{Im}(f)} |D_f(i) - D_g(i)| \tag{58}$$

Our general approach will consist of proving that the OF_P^2 Karp-reduces to an instance of $\mathrm{SD}_{\alpha\beta}$, which is an SZK-complete problem and thus belongs to SZKitself. We now apply the definition of $\mathrm{SD}_{\alpha\beta}$ to our function C_N which was introduced in Equation 54. Suppose we have two semiprimes N and S belonging to the domain of h. The statistical difference between C_N and C_S can be written in the following way:

$$\begin{aligned}
|D_{C_N} - D_{C_S}| &= \frac{1}{2} \sum_{i=1}^{M} \sum_{j \in \{-1,0,+1\}} |D_{C_N}((i,j)) - D_{C_S}((i,j))| \\
&= \frac{1}{2} \sum_{i=1}^{M} \sum_{j \in \{-1,0,+1\}} \left| \frac{|C_N^{-1}(\{(i,j)\})|}{|\mathrm{dom}(C_N)|} - \frac{|C_S^{-1}(\{(i,j)\})|}{|\mathrm{dom}(C_S)|} \right| \\
&= \frac{1}{2M} \sum_{j \in \{-1,0,+1\}} \underbrace{\sum_{i=1}^{M} \left| |C_N^{-1}(\{(i,j)\})| - |C_S^{-1}(\{(i,j)\})| \right|}_{X(j)} \\
\mathcal{S} &= \frac{1}{2M} \big(\underbrace{X(0)}_{:=\mathcal{G}} + \underbrace{X(+1)}_{:=\mathcal{B}} + \underbrace{X(-1)}_{:=\mathcal{R}} \big)
\end{aligned}$$

The variables $X(j)$ have quite a visual interpretation. Consider M marbles stacked in a column with the marble in the ith position labelled either red, green, or blue according to the value of $P(ord_N(i \mod N))$. We arrange two of these columns abreast, the marbles of one coloured according to $P(ord_N(i \mod N))$ and those of the other $P(ord_S(i \mod S))$. In this setup, $X(j)$ represents the number of pairs of adjacent marbles across the two columns where only one is coloured j. This is depicted in Figure **??**. In fact, together with the factor of $1/2M$, $\mathcal{S}$ represents the total number of pairs of adjacent marbles with mismatched colours. This interpretation is depicted in figure **??**.

For the proof, we will obtain upper and lower bounds for $\mathcal{S}$ that will end up satisfying the definition of $\mathrm{SD}_{\alpha\beta}$. After the proof, we also adopt a probabilistic approach to give us further insights into the value of $\mathcal{S}$ on average.

We are now ready to begin the proof.

C.2.2 The proof

Definition C.1 (C-(mis)match)**.** A pair of coloured marbles is called a C-mismatch if only one marble from the pair is coloured C. Otherwise if both marbles are coloured C then the pair is called a C-match.

The proof will make use of the following lemma which places bounds on the total number of C-mismatches between two rows of marbles.

Lemma 18. *Consider two rows of M coloured marbles that contain C_1 and C_2 C-coloured marbles respectively. The total number of C-mismatches between the two rows is bounded by the range*

$$[|C_1 - C_2|, M - |C_1 + C_2 - M|] \tag{59}$$

Proof. To obtain the lower bound, it is clear that the case where there is a minimum number of C-mismatches corresponds to the one where there is a maximum number of C-matches. There can at most be $\min(C_1, C_2)$ C-matches between the two rows, in which case there is a remaining $\max(C_1, C_2) - \min(C_1, C_2) = |C_1 - C_2|$ C-mismatches left behind.

For the upper bound, we need to minimise the number of C-matches. One way of doing this is to place the C_1 C-coloured marbles at the left end of the first row, and the C_2 C-coloured marbles at the right end of the second row. In the case where $C_1 + C_2 \leq N$, the tails of the two chains will never overlap, creating a maximum of $C_1 + C_2$ C-mismatches. However, if $C_1 + C_2 > N$, there will be an overlap giving rise to $C_1 - (M - C_2) = C_1 + C_2 - M$ C-matches. We must deduct twice this value from $C_1 + C_2$ to get the correct number of C-mismatches. We can combine both of these cases in a single expression:

$$\begin{aligned} C_1 + C_2 - 2\max(C_1 + C_2 - M, 0) &= \min(2N - (C_1 + C_2), C_1 + C_2) \\ &= M - |C_1 + C_2 - M| \end{aligned}$$

where the factor of two accounts for the double-counting of C-mismatches in the $C_1 + C_2$ term. □

Theorem 19. *The order-finding promise problem OF_P^2 outlined in Equation 51 reduces to $SD_{\alpha\beta}$ for all semiprimes N with $\frac{\varphi(N)}{N} > 99.6\%$.*

Proof. To present a Karp reduction, we have to show that there exists a polynomial-time, computable function f mapping from OF_P^2 to $\mathrm{SD}_{\alpha\beta}$ as the promise problems that we defined them to be in 51 and 57, such that

$$\begin{aligned} \forall N \in \Pi_{\mathrm{Y}} &\implies f(N) \in \Gamma_{\mathrm{Y}} \\ \forall N \in \Pi_{\mathrm{N}} &\implies f(N) \in \Gamma_{\mathrm{N}} \end{aligned}$$

The above condition means that under an input N, f must efficiently produce two functions g_1 and g_2 which have the required statistical differences of $\geq \alpha$ and $\leq \beta$ for when $h(N) = +1$ and -1 respectively for some $0 \leq \beta \leq \alpha \leq 1$. As also stipulated in definition 57 of $\mathrm{SD}_{\alpha\beta}$, we must additionally show that $\alpha^2 \geq \beta$. We will show that the function $f(N) = (C_N, C_S)$ with C_i defined in Equation 55 both satisfies this condition and is efficiently computable, provided S is any fixed semiprime with $h(S) = -1$. In our proof, we additionally assume that we are dealing with sufficiently large semiprimes N, S such that $\frac{\varphi(N)}{N}, \frac{\varphi(S)}{S} > \frac{2^\beta - 1}{2^\beta} > 99.6\%$ where $\beta \geq 9$.

Determining α, β satisfying $\alpha^2 > \beta$. As per the definition of $\mathrm{SD}_{\alpha\beta}$, it is clear that the tightest candidates for α and β are

$$\begin{aligned} \alpha &:= \mathcal{S}_+ = \inf\{|D_{C_N} - D_{C_S}| \ : \ h(N) = +1\} \\ \beta &:= \mathcal{S}_- = \sup\{|D_{C_N} - D_{C_S}| \ : \ h(N) = -1\} \end{aligned}$$

Figure 9: **Visual intuition for the selection of ϵ in the proof of Theorem 19.**

Within this notation, our task translates to showing $\mathcal{S}_+^2 > \mathcal{S}_-$. While it may be difficult to evaluate these two values, we note that it suffices to identify bounds $L_+ < \mathcal{S}_+$ and $U_- > S_-$, and show that $L_+^2 > U_-$.

The expression in Equation **??** corresponds to the sum of C-mismatches across the two columns for each of the three different colours of marbles: $\mathcal{R}$, $\mathcal{B}$, and $\mathcal{G}$. Using Lemma 59, the statistical difference can thus be bounded as

$$\mathcal{S} \in \frac{1}{2M}\left[\sum_{C\in\{R,G,B\}} |C'_1 - C'_2|\,, \sum_{C\in\{R,G,B\}} M - |C'_1 + C'_2 - M|\right] \tag{60}$$

$$= \frac{1}{2}\left[\sum_{c\in\{r,g,b\}} |c'_1 - c'_2|\,, \sum_{c\in\{r,g,b\}} 1 - |c'_1 + c'_2 - 1|\right] \tag{61}$$

where in the second line we have replaced total marble count C by proportions thereof in lower case $c' = \frac{C'}{M}$. We remind the reader that the dashed notation highlights quantities describing the population statistics of C_N as introduced in 55.

We can simplify this expression by considering the cases $h(N) = \pm 1$ separately and constraining the relative proportions of the colours.

The lower bound L_+. Consider the following inequality

$$\begin{aligned} |r'_1 - r'_2| + |g'_1 - g'_2| &= |g'_1 - g'_2 + b'_1 - b'_2| + |g'_1 - g'_2| \\ &\geq \big||g'_1 - g'_2| - |b'_1 - b'_2|\big| + |g'_1 - g'_2| \\ &\geq |b'_1 - b'_2| \end{aligned}$$

From this result, the lower bound in 61 can be reduced further in the following way:

$$L_+ = \frac{1}{2}\sum_{c\in\{r,g,b\}} |c'_1 - c'_2| \geq |b'_N - b'_S| \tag{62}$$

While the value of $|b'_N - b'_S|$ may vary depending on the choice of N and S, we will show that we can find yet another lower bound which is only a function of the evenness of N and S. To this end,

we first focus on the undashed quantity $|b_N - b_S|$. We know from Equation 53 that $b_S - b_N$ can be expressed in terms of the evenness:

$$\begin{aligned}
b_S - b_N &= \frac{\varphi(S)}{S}\left(1 - \frac{1}{2^{\mathcal{E}(S)}}\right) - \frac{\varphi(N)}{N}\left(1 - \frac{1}{2^{\mathcal{E}(N)}}\right) \\
&\geq \frac{2^\beta - 1}{2^\beta}\left(1 - \frac{1}{2^{\mathcal{E}(S)}}\right) - \left(1 - \frac{1}{2^{\mathcal{E}(N)}}\right) \quad \text{by minimising } \frac{\varphi(S)}{S} \text{ and maximising } \frac{\varphi(N)}{N} \\
&= \frac{1}{2^{\mathcal{E}(N)}} - \frac{2^\beta - 1}{2^\beta}\frac{1}{2^{\mathcal{E}(S)}} - \frac{1}{2^\beta} \\
&> 0.244 \qquad\qquad \text{where } \beta \geq 7,\ \mathcal{E}(S) = 6,\ \mathcal{E}(N) = 2
\end{aligned}$$

Now we relate this quantity to the dashed quantity $|b'_N - b'_S|$. We will make use of the fact that dashed proportions are at most ϵ away from their undashed counterparts, $|b_i - b'_i| \leq \epsilon$. Let $0 < \epsilon < \frac{b_S - b_N}{q_+}$ for a choice of $q_+ > 2$, then

$$\epsilon < \frac{b_S - b_N}{2} \implies b'_N < b_N + \epsilon < b_S - \epsilon < b'_S$$

Using this fact, we can drop the absolute value in the following calculation:

$$\begin{aligned}
|b'_S - b'_N| = b'_S - b'_N &> b_S - \epsilon - (b_N + \epsilon) > b_S - b_N - 2\epsilon \\
&> b_S - b_N - \frac{2(b_S - b_N)}{q_+} = (b_S - b_N)(1 - 2/q_+) > 0.24(1 - 2/q_+)\,.
\end{aligned}$$

Now we can derive a lower bound for L_+:

$$L_+ \geq |b'_N - b'_S| \geq 0.244(1 - 2/q_+) > 0.12$$

for $q_+ > 10^4$. Note that constraining q_+ as such places the following bound on ϵ.

$$\frac{b_S - b_N}{q_+} > \frac{0.244}{q_+} = 2.44 \times 10^{-4} > \epsilon$$

Upper bound Similar to the previous section, we need only consider the $h(N) = -1$ instances when evaluating the upper bound U_-. The upper bound in 61 can be further bounded above if we minimise the negative terms $|c'_1 + c'_2 - 1|$. First, we focus on blue marbles $c = b$. Let $|b - b'| < \epsilon = \frac{|b_N + b_S - 1|}{2q_-}$ for some $q_- > 1$. Then by the triangle inequality:

$$|b'_N + b'_S - (b_N + b_S)| \leq |b'_N - b_N| + |b'_S - b_S| \leq \frac{|b_N + b_S - 1|}{q_-} \leq |b_N + b_S - 1|$$

We use this result to advance from the first line to the second of the following derivation:

$$\begin{aligned}
|b'_N + b'_S - 1| &= |b'_N + b'_S - (b_N + b_S) + b_N + b_S - 1| \geq \big||b'_N + b'_S - (b_N + b_S)| - |b_N + b_S - 1|\big| \\
&= |b_N + b_S - 1| - |b'_N + b'_S - (b_N + b_S)| \geq |b_N + b_S - 1| - \frac{|b_N + b_S - 1|}{q_-} \\
&= |b_N + b_S - 1|(1 - 1/q_-)
\end{aligned}$$

By symmetry, note that the identical inequalities hold for $c = g$ and $c = r$ provided $\epsilon < \min_{c \in \{r,g,b\}} |c_N + c_S - 1|/2q_-$. In fact, since $\mathcal{E}(N), \mathcal{E}(S) \geq 9$ ensures that

$$b_S + b_N > 2 \times \frac{2^\beta - 1}{2^\beta}\left(1 - \frac{1}{2^9}\right) > 1.98$$

the following is guaranteed:

$$\begin{aligned}
&0 < b_N + b_S - 1 \leq 1 - r_N - g_N - r_S - g_S \leq \min(1 - (r_N + r_S), 1 - (g_N + g_S)) \\
\implies &|b_N + b_S - 1| < |r_N + r_S - 1|,\ |g_N + g_S - 1|
\end{aligned}$$

And thus epsilon is bounded above by

$$\min_{c \in \{r,g,b\}} \frac{|c_N + c_S - 1|}{2q_-} = \frac{|b_N + b_S - 1|}{2q_-} \geq \frac{0.98}{2q_-} = 4.9 \times 10^{-4} \geq \epsilon \tag{63}$$

With ϵ constrained as above, U_- is upper bounded by

$$\begin{aligned}
U_- &= \frac{1}{2} \sum_{c \in \{r,g,b\}} 1 - |c'_N + c'_S - 1| \\
&< \frac{1}{2} \left(3 - (1 - 1/q_-) \sum_{c \in \{r,g,b\}} |c_N + c_S - 1| \right) \\
&= \frac{1}{2} \left(3 - (1 - 1/q_-) \left(1 - g_N - r_N + b_N - g_S - r_S + b_S \right) \right) \\
&= \left(\frac{3}{2} - \left(1 - \frac{1}{q_-} \right) \left((b_N + b_S) - \frac{1}{2} \right) \right) \\
&< \left(\frac{3}{2} - 1.48 \left(1 - \frac{1}{q_-} \right) \right) \leq 0.0132
\end{aligned}$$

for $q_- > 10^4$.

Finally, we note that

$$\alpha^2 \geq L_+^2 \geq 0.12^2 \geq 0.014 > 0.0132 \geq U_- \geq \beta \tag{64}$$

Efficiency of f. There are two underlying tasks to $f(N) = (C_N, C_S)$ which we must show are efficient under the input N: generating a semiprime S such taht $h(S) = -1$ and computing $C_{(\cdot)(i)} \forall i \in [1, M]$. For the former task, observe that a sufficiently small -1 semiprime can be precomputed efficiently and generated in $\mathcal{O}(1)$ time. One such semiprime is the product of the two largest Fermat primes $F_3 = 2^{2^3} + 1$ and $F_4 = 2^{2^4} + 1$ whose totient is

$$\varphi(F_3 F_4) = (2^{2^3})(2^{2^4}) = 2^{24} \implies h(F_3 F_4) = -1 \tag{65}$$

Furthermore, note that $\frac{\varphi(F_3 F_4)}{F_3 F_4} > 99.6\%$, as required by our constraint on sufficiently large semiprimes. Thus, $S \leftarrow F_3 F_4$ is a sufficient candidate that is efficiently implementable.

As for computing $C_N(i) = (i, P(ord_N(i \mod N)))$ given N, note that the calculation consists of modular arithmetic and parity checking which are both efficiently done. The only nontrivial computation is order finding $ord_N(\cdot)$. If we assume a quantum Turing machine as our model of computation, then order-finding can be done in exact polynomial time using the derandomised order-finding algorithm introduced in REF. This means that the f represents a Karp reduction on a quantum computer. □

Corollary 19.1. *OF_P^2 has a SZK proof where the verifier has access to a quantum Turing Machine.*

Proof. Since $\mathrm{SD}_{\alpha\beta}$ is a complete problem for SZK, $\mathrm{SD}_{\alpha\beta} \in$ SZK. By the reduction in Theorem 19, it follows that

$$OF_P^2 \leq_K \mathrm{SD}_{\alpha\beta} \in \text{SZK} \tag{66}$$

□

To further consolidate the corollary above, it is instructive to see OF_P^2 expressed as an SZK interactive proof. We claim that the following interactive protocol is a statistical zero-knowledge interactive proof:

An interactive proof for OF_P^2
Input: Semiprime $N \in \Pi_Y \cup \Pi_N$

Verifier		**Prover**
$S \leftarrow F_3 \times F_4$		
	$\xrightarrow{\quad S \quad}$	
$i \leftarrow\$ [1, M]$ $b \leftarrow\$ \{N, S\}$ $c \leftarrow \big(i, P\big(ord_b(i \mod b)\big)\big)$		
	$\xrightarrow{\quad c \quad}$	
		$d \leftarrow \big(i, P\big(ord_S(i \mod S)\big)\big)$ **if** $d = c$ **then** $e \leftarrow S$ **else** $e \leftarrow N$
	$\xleftarrow{\quad e \quad}$	
if $e = b$ **then** Accept **else** Reject		

First, we show that the protocol above represents an interactive proof system for the language Π. To do so, we must show that the completeness error ϵ_C (false Reject) and the soundness error (false Accept) satisfy $1 - \epsilon_C > \epsilon_S + \frac{1}{p(k)}$ for some polynomial $p(k)$ where k is a security parameter.

We can easily obtain these errors through a tree diagram:

Note that only the branches leading to an Accept response by the verifier are shown above. Importantly, the first and third branches refer to outcomes where a mismatch between the value of C_N and C_S has occurred for a randomly chosen $i \leftarrow\$ [1, M]$. The probability of this occurring simply corresponds to the proportion of mismatches among the M pairs of marbles. Indeed, this evaluates exactly to the statistical difference $\mathcal{S}$. This is why the lower and upper bounds derived in the proof of theorem 19 bound the probabilities of the first and third branches respectively. Putting it all together, we get:

$$1 - \epsilon_c = \mathbb{P}(\text{Accept } | h(N) = +1) \geq \frac{1}{2}(1 + L_+) \geq 0.56$$
$$\epsilon_s = \mathbb{P}(\text{Accept } | h(N) = -1) \leq \frac{1}{2}(1 + U_-) \leq 0.51$$

Thus,

$$\implies 1 - \epsilon_C > \epsilon_s + 1/p(k)$$

for a sufficiently large constant polynomial such as $p(k) = 100$.

This shows that the protocol above is an interactive proof system for OF_P^2 . It now remains to show that the proof is also statistical zero knowledge.

An interactive proof is considered statistical zero knowledge if the verifier learns nothing else in their interaction with the prover beyond whether N is a ± 1 instance. This concept is made formal by requiring that the verifier's view of the interaction (i.e. the messages it sends to and receives from the prover) be efficiently and accurately simulated by a simulator that only takes the input string as input. In other words, the simulator must produce a distribution over the message transcripts that is arbitrarily close to that which is obtained in the real interactive proof. Vadhin shows that this requirement is satisfied for all interactive proofs where $L_+^2 > U_-$ (which we showed in theorem 19) by proving a polarization lemma. We refer the interested reader to Vadhin's thesis for more information REF.

C.2.3 Statistical insights

To gain further insight into the value of the statistical difference from Equation **??**, we may reinterpret $\mathcal{S}$ as a probabilistic quantity where $X(j)$ represents a random variable. The justification for this approach is that calculating the preimage $C_N^{-1}(\{(i,j)\})$ requires knowing if i is not coprime with N, or otherwise whether it has an even or odd order. The lack of a closed-form expression for deciding these properties along with our knowledge of the global proportions of C_N is what makes the probabilistic approach preferable.

With $\mathcal{S}$ now a random variable, we will rely on the expectancy $\mathbb{E}[\mathcal{S}]$ to derive characterise the average value of α and β satisfying $\alpha^2 \geq \beta$:

$$\mathbb{E}[\mathcal{S}] = \frac{1}{2M}(\mathbb{E}[\mathcal{R}] + \mathbb{E}[\mathcal{G}] + \mathbb{E}[\mathcal{B}])$$

Focusing just on the Red mismatches as the others follow by symmetry,

$$\mathbb{E}[\mathcal{R}] = \mathbb{E}[\sum_{i=1}^{M} I_i]$$

I_i is an indicator random variable denoting whether the pair of marbles at position i in the column is a $\{R, \neg R\}$ mismatch.

$$\begin{aligned}
&= \sum_{i=1}^{M} \mathbb{E}[I_i] = \sum_{i=1}^{M} \mathbb{P}(I_i = 1) \\
&= \#\ \{R, \neg R\} \text{ mismatches} = M \frac{\#\ \{R, \neg R\} \text{ mismatches}}{M}
\end{aligned}$$

The fraction on the right is exactly the probability of getting a mismatch on the σth randomly sampled marble:

$$\begin{aligned}
&= M\mathbb{P}(I_\sigma = 1) \\
&= M(\mathbb{P}(R_1 \wedge \neg R_2) + \mathbb{P}(\neg R_1 \wedge R_2)) \\
&= M(\mathbb{P}(R_1)\mathbb{P}(\neg R_2) + \mathbb{P}(\neg R_1)\mathbb{P}(R_2)) \\
&= M(\mathbb{P}(R_1) + \mathbb{P}(R_2) - 2\mathbb{P}(R_1)\mathbb{P}(R_2))
\end{aligned} \tag{67}$$

In this derivation, we have assumed that the colours of pairs of marbles are independent of each other. While this is ultimately a conjecture, we justify this assumption by claiming that a number having an odd order is unlikely to be dependent on another number not having an odd order. Our

heuristics indicate that if there is some dependency, then it would have to be marginal and decreasing for increasing semiprimes. Incorporating the result from Equation 67 we get the expectation value:

$$\begin{aligned}
\mathbb{E}[\mathcal{S}] &= \frac{1}{2}\Big(\mathbb{P}(R_1)+\mathbb{P}(B_1)+\mathbb{P}(G_1)+\mathbb{P}(R_2)+\mathbb{P}(B_2)+\mathbb{P}(G_2)-2\Big(\mathbb{P}(R_1)\mathbb{P}(R_2)+\mathbb{P}(G_1)\mathbb{P}(G_2)+\mathbb{P}(B_1)\mathbb{P}(B_2)\Big)\Big)\\
&= 1-\mathbb{P}(R_1)\mathbb{P}(R_2)-\mathbb{P}(G_1)\mathbb{P}(G_2)-\mathbb{P}(B_1)\mathbb{P}(B_2)\\
&= 1-\frac{R_1R_2+G_1G_2+B_1B_2}{NS}-\mathcal{O}(\epsilon)
\end{aligned}$$

Recalling the proportions derived in Equation 53, we have

$$\begin{aligned}
&= 1-\left(\left(1-\frac{\varphi(N)}{N}\right)\left(1-\frac{\varphi(S)}{S}\right)+\frac{\varphi(N)}{N}\frac{\varphi(S)}{S}\left(1-\frac{1}{2^{\mathcal{E}(N)}}\right)\left(1-\frac{1}{2^{\mathcal{E}(S)}}\right)+\frac{\varphi(N)}{N}\frac{\varphi(S)}{S}\left(\frac{1}{2^{\mathcal{E}(S)+\mathcal{E}(N)}})\right)\right)-\mathcal{O}(\epsilon)\\
&= \frac{\varphi(N)}{N}+\frac{\varphi(S)}{S}-\frac{\varphi(S)}{S}\frac{\varphi(N)}{N}\left(2-\frac{1}{2^{\mathcal{E}(N)}}-\frac{1}{2^{\mathcal{E}(S)}}+\frac{1}{2^{\mathcal{E}(N)+\mathcal{E}(S)+1}}\right)-\mathcal{O}(\epsilon)\\
&\overset{S,\,N\to\infty}{=} \frac{1}{2^{\mathcal{E}(N)}}+\frac{1}{2^{\mathcal{E}(S)}}-\frac{1}{2^{\mathcal{E}(N)+\mathcal{E}(S)+1}}-\mathcal{O}(\epsilon)\in\left[\frac{1}{2^{\mathcal{E}(N)}},\ \frac{1}{2^{\mathcal{E}(N)}}+\frac{1}{2^{\mathcal{E}(S)}}\right]
\end{aligned}$$

The bounds follow if we ensure that the $\mathcal{O}(\epsilon) < \frac{1}{2^{\mathcal{E}(S)}} - \frac{1}{2^{\mathcal{E}(N)+\mathcal{E}(S)+1}}$ which we can arrange by suitably increasing k in $M = kN_{max}$. If we now fix S such that $h(S) = -1$, then we get

$$\begin{cases} = \left[\frac{1}{4}, \frac{9}{32}\right] & h(N) = +1 \\ = \left[0, \frac{1}{16}\right] & h(N) = -1 \end{cases}$$

Put differently, if we take $\alpha = 1/4$ and $\beta = 1/16$, the expected statistical difference of -1 and $+1$ semiprimes with respect to D_{C_S} has the following bounds:

$$\begin{aligned}
h(N) = +1 &\implies \mathbb{E}[\mathcal{S}] \geq \alpha\\
h(N) = -1 &\implies \mathbb{E}[\mathcal{S}] \leq \beta\,.
\end{aligned}$$

By observing that $0 \leq \beta \leq \alpha^2 \leq 1$, we see that the promise problem defined by h is by definition an instance of $\mathrm{SD}_{\alpha\beta}$ and consequently in SZK*on average*.

This is insufficient for the proof however, as we need to show that we can assign a probability of success of δ to the class membership above that is asymptotically unity. We begin by calculating the variance of the statistical difference:

$$\mathrm{Var}[\mathcal{S}] = \frac{1}{4M^2}(\mathrm{Var}[\mathcal{R}]) + \mathrm{Var}[\mathcal{G}] + \mathrm{Var}[\mathcal{B}] + 2(\mathrm{Cov}[\mathcal{R},\mathcal{G}] + \mathrm{Cov}[\mathcal{G},\mathcal{B}] + \mathrm{Cov}[\mathcal{B},\mathcal{R}])) \tag{68}$$

Let us focus only on one pair of random variables, as the others can be iterated by symmetry

$$\begin{aligned}
\mathrm{Var}[\mathcal{B}+\mathcal{R}] &= \mathrm{Var}[\mathcal{R}]) + \mathrm{Var}[\mathcal{B}] + 2\mathrm{Cov}[\mathcal{B},\mathcal{R}])\\
&= \mathbb{E}[\mathcal{R}^2] - \mathbb{E}[\mathcal{R}]^2 + \mathbb{E}[\mathcal{B}^2] - \mathbb{E}[\mathcal{B}]^2 + 2\mathbb{E}[\mathcal{R}\mathcal{B}] - 2\mathbb{E}[\mathcal{R}]\mathbb{E}[\mathcal{B}]
\end{aligned}$$

In this expression there are two types of quantities that we cannot derive by just recycling the expected values derived from the previous step: $\mathbb{E}[\mathcal{R}^2]$ and $\mathbb{E}[\mathcal{R}\mathcal{B}]$. The manner of deriving these quantities will be similar to calculating expected values using indicator random variables

$$\begin{aligned}
\mathbb{E}[\mathcal{R}^2] &= \mathbb{E}[\sum_{i,j} I_iI_j] = \sum_{i,j}\mathbb{E}[I_iI_j] = \sum_{i,j}\mathbb{P}(I_i = 1 \wedge I_j = 1 \mid i^{\text{th}} \text{ and } j^{\text{th}} \text{ pair of marbles})\\
&= \sum_{i\neq j}\mathbb{P}(I_i = 1 \wedge I_j = 1) + \sum_i \mathbb{P}(I_i = 1) = \sum_{i\neq j}\mathbb{P}(I_i = 1 \wedge I_j = 1) + \mathbb{E}[\mathcal{R}]
\end{aligned}$$

The first term counts the number of distinct pairs of $\{R, \neg R\}$ mismatches along the two columns:

$$= \#\ \{R, \neg R\}\text{—}\{R, \neg R\} \text{ pairs} + \mathbb{E}[\mathcal{R}] = M(M-1)\frac{\#\ \{R, \neg R\}\text{—}\{R, \neg R\} \text{ pairs}}{M(M-1)} + \mathbb{E}[\mathcal{R}]$$

As seen earlier, the fraction is precisely the probability of obtaining two consecutive $\{R, \neg R\}$ mismatches after random sampling without replacement:

$$= M(M-1)\mathbb{P}(I_\sigma = 1 \wedge I_{\sigma+1} = 1 \mid \sigma^{\text{th}} \text{ and } (\sigma+1)^{\text{th}} \text{ pairs randomly sampled}) + \mathbb{E}[\mathcal{R}]$$

There are four combinations of marbles that contribute to this probability term

$$\mathbb{P}(I_\sigma = 1 \wedge I_{\sigma+1} = 1) = \mathbb{P}\begin{pmatrix} \{R \quad \neg R\}_\sigma \\ \{R \quad \neg R\}_{\sigma+1} \end{pmatrix}$$

For clarity, we will suppress the indices but it should be understood that the bottom row is sampled after the top row.

$$\begin{aligned}
&= \mathbb{P}\begin{pmatrix} (R \quad \neg R) \\ (R \quad \neg R) \end{pmatrix} + \mathbb{P}\begin{pmatrix} (\neg R \quad R) \\ (\neg R \quad R) \end{pmatrix} + \mathbb{P}\begin{pmatrix} (\neg R \quad R) \\ (R \quad \neg R) \end{pmatrix} + \mathbb{P}\begin{pmatrix} (R \quad \neg R) \\ (\neg R \quad R) \end{pmatrix} \\
&= \mathbb{P}\begin{pmatrix} R \\ R \end{pmatrix}\mathbb{P}\begin{pmatrix} \neg R \\ \neg R \end{pmatrix} + \mathbb{P}\begin{pmatrix} \neg R \\ \neg R \end{pmatrix}\mathbb{P}\begin{pmatrix} R \\ R \end{pmatrix} + \mathbb{P}\begin{pmatrix} \neg R \\ R \end{pmatrix}\mathbb{P}\begin{pmatrix} R \\ \neg R \end{pmatrix} + \mathbb{P}\begin{pmatrix} R \\ \neg R \end{pmatrix}\mathbb{P}\begin{pmatrix} \neg R \\ R \end{pmatrix}
\end{aligned}$$

The probabilities above fall into two main categories:

$$\begin{aligned}
\mathbb{P}\begin{pmatrix} R \\ R \end{pmatrix}\mathbb{P}\begin{pmatrix} \neg R \\ \neg R \end{pmatrix} &= \left(\frac{R_1'}{M}\frac{R_1'-1}{M-1}\right)\left(\frac{M-R_2'}{M}\frac{M-R_2'-1}{M-1}\right) \\
\mathbb{P}\begin{pmatrix} \neg R \\ R \end{pmatrix}\mathbb{P}\begin{pmatrix} R \\ \neg R \end{pmatrix} &= \left(\frac{M-R_1'}{M}\frac{R_1'}{M-1}\right)\left(\frac{R_2'}{M}\frac{M-R_2'}{M-1}\right)
\end{aligned}$$

Incorporating these probabilities into the final sum, we get

$$\mathbb{P}\begin{pmatrix} \{R \quad \neg R\}_\sigma \\ \{R \quad \neg R\}_{\sigma+1} \end{pmatrix} = \frac{\left(R_1'\left(M-R_2'\right)\left(-2R_1'R_2' + R_2'\left(M+1\right) + \overline{M}\left(R_1'-1\right)\right) + R_2'\left(M-R_1'\right)\left(-2R_1'R_2' + R_1'\left(M+1\right) + \overline{M}\left(R_2'-1\right)\right)\right)}{M^2\overline{M}^2}$$

where $\overline{M} = M - 1$. The cross term $\mathbb{E}[\mathcal{B}, \mathcal{R}]$ can similarly be obtained as:

$$\begin{aligned}
\mathbb{E}[\mathcal{B}\mathcal{R}] &= \sum_{i \neq j}\mathbb{P}(I_i^{\mathcal{R}} = 1 \wedge I_j^{\mathcal{B}} = 1) + \sum_i \mathbb{P}(I_i^{\mathcal{R}} = 1 \wedge I_i^{\mathcal{B}} = 1) \\
&= M\overline{M}\mathbb{P}\begin{pmatrix} \{B \quad \neg B\}_\sigma \\ \{R \quad \neg R\}_{\sigma+1} \end{pmatrix} + M\mathbb{P}\left(\{B \quad R\}_\sigma\right) \\
&= \frac{(B_1'R_2' + B_2'R_1')}{M} + \frac{\left(8B_1'B_2'R_1'R_2' - \overline{M}\left(2B_1R_2' + 2B_2R_1' + (B_1'+B_2')\left(-M\left(R_1'+R_2'\right) + 4R_1'R_2'\right) + \left(R_1'+R_2'\right)\left(4B_1'B_2' - M\left(B_1'+B_2'\right)\right)\right)\right)}{2M\overline{M}}
\end{aligned}$$

Substituting these quantities into Equation 68 and assuming $\overline{M} \approx M$ in the limit of large M, we have

for the variance:

$$
\begin{aligned}
\mathrm{Var}[\mathcal{S}] = & -\frac{{B_1'}^2{B_2'}^2}{M^5} + \frac{3{B_1'}^2B_2'}{4M^4} + \frac{{B_1'}^2B_2'}{4M^5} - \frac{{B_1'}^2}{4M^3} + \frac{3B_1'{B_2'}^2}{4M^4} + \frac{B_1'{B_2'}^2}{4M^5} - \frac{2B_1'B_2'G_1'G_2'}{M^5} + \frac{B_1'B_2'G_1'}{M^4} + \frac{B_1'B_2'G_2'}{M^4} + \\
& \frac{B_1'B_2'R_1'}{M^4} + \frac{B_1'B_2'R_2'}{M^4} - \frac{B_1'B_2'}{M^4} - \frac{2B_1'B_2'R_1'R_2'}{M^5} + \frac{B_1'G_1'G_2'}{M^4} - \frac{B_1'G_1'}{2M^3} - \frac{B_1'G_2'}{2M^3} + \frac{B_1'G_2'}{2M^4} - \frac{B_1'R_1'}{2M^3} - \frac{B_1'R_2'}{2M^3} + \\
& \frac{B_1'}{4M^3} + \frac{B_1'R_1'R_2'}{M^4} + \frac{B_1'R_2'}{2M^4} - \frac{{B_2'}^2}{4M^3} + \frac{B_2'G_1'G_2'}{M^4} - \frac{B_2'G_1'}{2M^3} + \frac{B_2'G_1'}{2M^4} - \frac{B_2'G_2'}{2M^3} - \frac{B_2'R_1'}{2M^3} - \frac{B_2'R_2'}{2M^3} + \frac{B_2'}{4M^3} + \\
& \frac{B_2'R_1'R_2'}{M^4} + \frac{B_2'R_1'}{2M^4} - \frac{{G_1'}^2{G_2'}^2}{M^5} + \frac{3{G_1'}^2G_2'}{4M^4} + \frac{{G_1'}^2G_2'}{4M^5} - \frac{{G_1'}^2}{4M^3} + \frac{3G_1'{G_2'}^2}{4M^4} + \frac{G_1'{G_2'}^2}{4M^5} + \frac{G_1'G_2'R_1'}{M^4} + \frac{G_1'G_2'R_2'}{M^4} - \\
& \frac{G_1'G_2'}{M^4} - \frac{2G_1'G_2'R_1'R_2'}{M^5} - \frac{G_1'R_1'}{2M^3} - \frac{G_1'R_2'}{2M^3} + \frac{G_1'}{4M^3} + \frac{G_1'R_1'R_2'}{M^4} + \frac{G_1'R_2'}{2M^4} - \frac{{G_2'}^2}{4M^3} - \frac{G_2'R_1'}{2M^3} - \frac{G_2'R_2'}{2M^3} + \frac{G_2'}{4M^3} + \\
& \frac{G_2'R_1'R_2'}{M^4} + \frac{G_2'R_1'}{2M^4} - \frac{{R_1'}^2}{4M^3} + \frac{R_1'}{4M^3} - \frac{{R_2'}^2}{4M^3} + \frac{R_2'}{4M^3} + \frac{3{R_1'}^2R_2'}{4M^4} + \frac{3R_1'{R_2'}^2}{4M^4} - \frac{R_1'R_2'}{M^4} - \frac{{R_1'}^2{R_2'}^2}{M^5} + \frac{{R_1'}^2R_2'}{4M^5} + \\
& \frac{R_1'{R_2'}^2}{4M^5} \\
& \leq \frac{9}{M^2} + \frac{33}{M} \leq \frac{42}{M} = \mathcal{O}\left(\frac{1}{M}\right) \implies \sigma\left[\mathcal{S}\right] \leq \sqrt{\frac{42}{M}}
\end{aligned}
\tag{69}
$$

From $\sigma\left[\mathcal{S}\right]$, we can obtain the success probability δ. Let $\mathcal{S}_\pm$ be the probabilistic statistical difference for $C_{N_\pm}$ where $N_\pm$ are randomly sampled semiprimes wth label ± 1 respectively.

$$
\delta = \mathbb{P}\left(\mathcal{S}_- < \mathcal{S}_+^2\right) = \sum_{\substack{s_+, s_- \\ s_+^2 > s_-}} \mathbb{P}\left(\mathcal{S}_- = s_- \wedge \mathcal{S}_+^2 = s_+^2\right)
$$

Since N_+ and N_- are randomly sampled, $\mathcal{S}_-$ and $\mathcal{S}_+$ are independent:

$$
\begin{aligned}
&= \sum_{\substack{s_+, s_- \\ s_+^2 > s_-}} \mathbb{P}\left(\mathcal{S}_- = s_-\right)\mathbb{P}\left(\mathcal{S}_+^2 = s_+^2\right) \geq \sum_{\substack{s_+^2 > \lambda \\ s_- < \lambda}} \mathbb{P}\left(\mathcal{S}_- = s_-\right)\mathbb{P}\left(\mathcal{S}_+^2 = s_+^2\right) \qquad \forall \lambda \\
&= \mathbb{P}\left(\mathcal{S}_- < \lambda\right)\mathbb{P}\left(\mathcal{S}_+^2 > \lambda\right) = \mathbb{P}\left(\mathcal{S}_- < \lambda\right)\mathbb{P}\left(\left|\mathcal{S}_+\right| > \sqrt{\lambda}\right)
\end{aligned}
$$

Let $\mu_\pm = \mathbb{E}[\mathcal{S}_\pm]$ and fix λ so that it satisfies $\lambda > \mu_-$ and $\sqrt{\lambda} < \mu_+$:

$$
\begin{aligned}
&\geq \mathbb{P}\left(2\mu_- - \lambda < \mathcal{S}_- < \lambda\right)\mathbb{P}\left(\sqrt{\lambda} < \mathcal{S}_+ < 2\mu_+ - \sqrt{\lambda}\right) \\
&= \mathbb{P}\left(\left|\mathcal{S}_- - \mu_-\right| < \lambda - \mu_-\right)\mathbb{P}\left(\left|\mathcal{S}_+ - \mu_+\right| < \mu_+ - \sqrt{\lambda}\right)
\end{aligned}
$$

Let $\tilde{\lambda} \coloneqq \min\left(\lambda - \mu_-, \mu_+ - \sqrt{\lambda})\right)$

$$
\geq \mathbb{P}\left(\left|\mathcal{S}_- - \mu_-\right| < \tilde{\lambda}\right)\mathbb{P}\left(\left|\mathcal{S}_+ - \mu_+\right| < \tilde{\lambda}\right)
$$

By Chebyshev's inequality

$$
\begin{aligned}
&\geq \left(1 - \frac{\sigma_+^2}{\tilde{\lambda}^2}\right)\left(1 - \frac{\sigma_-^2}{\tilde{\lambda}^2}\right) \\
1 \geq \delta &\geq \left(1 - \frac{42}{M\tilde{\lambda}}\right)^2 = 1 - \mathcal{O}\left(\frac{1}{M}\right) \to 1 \text{ as } M \to \infty \\
&\therefore \delta \to 1 \text{ as } M \to \infty
\end{aligned}
$$

Thus, the probability of success approaches unity as M increases, which may be arranged for a large choice of k or otherwise a large maximum semiprime N_{max} in the dataset.

C.3 The Concept Class based on OF_P^2 is Quantumly Learnable

In this section we provide a proof for the fact that the concept class based on OF_P^2 is able to be learned by a `QKE-SVM` in a PAC learning sense REF. A brief outline of the proof will be given, followed by a series of Lemmas that are relevant for proving quantum tractability. The proof will follow very similarly to the section IV of the paper by Liu *et al.*.

Recall that the OF_P^2 promise problem is defined by the labeling rule $h : X \to \{+1, -1\}$ introduced in Equation 51. The domain X is referred to as the instance space, which in our case consists of the set of all semiprimes. To introduce a notion of size to the situation, we will consider a separate labeling rule for each X_n, which we define to be the set of all $n-$digit semiprimes. It can be shown that the size of this set grows superpolynomially with n CAN IT? We will now fix some n and limit our attention to h_n suppressing the subscript for clarity.

In the language of Probably Approximately Correct (PAC) learning theory, the labeling rule h_n is called a *concept* over the instance space $\mathcal{X}_N$. The singleton set $C_n = \{h\}$ is called the *concept class*. In PAC learning, an algorithm A is said to have efficiently learned the concept class $C = \cup \mathcal{C}_N$ if $\forall h \in C_n$ and distributions $\mathcal{D}$ over $\mathcal{X}_N$, provided access to an oracle $\mathrm{EX}\langle h, D\rangle$ capable of producing samples $\{N_i, h(N_i)\}$ over the distribution $\mathcal{D}$ in unit time, as well as $\epsilon > 0, \delta < 1/2$, in time $\mathsf{poly}(n)(n, 1/\epsilon, 1/\delta)$ it is able to output a hypothesis function h^* satisfying $acc_D(h, h^*) < 1 - \epsilon$ with probabiliy of success at least $1 - \delta$ over the choice of a random sampling set and internal randomness of the algorithm.

We will show that the `QKE-SVM` can efficiently PAC learn the concept class $C = \cup C_n$ where $C_n = \{h_n\}$.

C.3.1 The ground-truth hyperplane

In the previous section we demonstrated a quantum feature space in which $+1$ and -1 labelled states of OF_P^2 produced different inner products with respect to a state $|C_S\rangle$. This state can be considered to be a separating hyperplane within this space. By representing these states in as density matrices, $|C_S\rangle\langle C_S|$ and $|C_N\rangle\langle C_N|$ can be represented with respect to the 4^n-dimensional Pauli basis $\{P_\alpha\}$. In this basis, they can be identified with their Fourier coefficients, $\vec{w} = (w_\alpha)$ and $\vec{x} = (x_\alpha)$ respectively defined using the trace inner product from **??**. Note that the Euclidean inner product within this $\mathbb{R}^{4^n}$ space evaluates to the standard Hilbert-Schmidt inner product within the space $\mathbb{C}^{2^n \times 2^n}$:

$$\langle |C_S\rangle\langle C_S|, |C_N\rangle\langle C_N|\rangle_{HS} = |\langle C_S|C_N\rangle|^2 = \sum_{\alpha=0}^{4^n-1} w_\alpha x_\alpha = \langle \vec{w}, \vec{x}\rangle_{\mathbb{R}}.$$

In this notation, we can rewrite Equation 34 as

$$\vec{w} \cdot \vec{x} = (1 - \mathcal{S})^2 \begin{cases} \leq (1-\alpha)^2 := a & \forall x \in \Pi_Y^{\mathrm{OFP}} \\ \geq (1-\beta)^2 := b & \forall x \in \Pi_N^{\mathrm{OFP}}. \end{cases} \tag{70}$$

where $\mathcal{S}$ is the statistical difference. Subtracting $\frac{a+b}{2}$ from both sides of the equation above gives:

$$\vec{w} \cdot \vec{x} - c = \begin{cases} \leq -\frac{\Delta}{2} & \forall x \in \Pi_Y^{\mathrm{OFP}} \\ \geq \frac{\Delta}{2} & \forall x \in \Pi_N^{\mathrm{OFP}}. \end{cases} \tag{71}$$

where $c = \Delta/2 + a$ and $\Delta := b - a$.

This equation clearly characterized $\vec{w}$ as a hyperplane separating all Yes and No instances with a hard margin of Δ. It is also easy to see that all instances are bounded within a margin of c from the hyperplane. Both these properties can be summarized in the following way:

$$Pr_{x\sim D}\left[c \geq -y\left(\vec{w} \cdot \vec{x} + c\right) \geq \frac{\Delta}{2}\right] = 1$$

The fact that the ground-truth hyperplane has a hard margin means that the slack variables in the primal program of the soft-margin support vector machine are all identically zero: $||\vec{\xi}||^2 = 0$. Two

more steps can be undertaken to make the equation above easier to work with. First, the bias term c can be absorbed into $\vec{w}$ by making the transformation $\vec{w} \mapsto \frac{(\vec{w},c)}{\sqrt{w^2+c^2}}$ and $\vec{x} \mapsto \frac{(\vec{x},1)}{\sqrt{2}}$. In this case,

$$Pr_{x\sim D}\left[\frac{c}{\sqrt{2+2c^2}} \geq -y\left(\vec{w}\cdot\vec{x}\right) \geq \frac{\Delta}{2(\sqrt{2+2c^2})}\right] = 1$$

Making the transformation $\vec{w} \mapsto \frac{\vec{w}}{\frac{\Delta}{2(\sqrt{2+2c^2})}}$

$$Pr_{x\sim D}\left[\frac{2c}{\Delta} \geq -y\left(\vec{w}\cdot\vec{x}\right) \geq 1\right] = 1$$

Importantly, note that all of a, b and Δ are constants no matter the dimensionality of $\vec{x}$ and $\vec{w}$, making $||\vec{w}|| = \mathcal{O}\left(1\right)$. In [15], Liu *et al.* show that their ground truth hyperplane normal vector satisfies $||\vec{w}|| = \mathcal{O}\left(\mathsf{poly}(n)\right)$. Together with the fact that $||\vec{\xi}||^2 = 0 = \mathcal{O}(1)$, our concept class admits the same proof for quantum learnability as in [15]. Markedly, the constancy of these quantities means that for our concept class, the number of instances m is independent of the size of the instance space n, unlike in the case of the DLP concept class where the number of instances must be polynomial in n, i.e. $m = \mathcal{O}(n^c)$.