Quantum mean embedding of probability distributions

Jonas M. Kübler,* Krikamol Muandet,† and Bernhard Schölkopf‡
Max Planck Institute for Intelligent Systems, 72076 Tübingen, Germany

(Received 15 June 2019; published 9 December 2019)

The kernel mean embedding of probability distributions is commonly used in machine learning as an injective mapping from distributions to functions in an infinite-dimensional Hilbert space. It allows us, for example, to define a distance measure between probability distributions, called the maximum mean discrepancy. In this work, we propose to represent probability distributions in a pure quantum state of a system that is described by an infinite-dimensional Hilbert space and prove that the representation is unique if the corresponding kernel function is c_0 universal. This enables us to work with an explicit representation of the mean embedding, whereas classically one can only work implicitly with an infinite-dimensional Hilbert space through the use of the kernel trick. We show how this explicit representation can speed up methods that rely on inner products of mean embeddings and discuss the theoretical and experimental challenges that need to be solved in order to achieve these speedups.

DOI: 10.1103/PhysRevResearch.1.033159

I. INTRODUCTION

In machine learning, kernel methods are used to implicitly evaluate inner products in high-dimensional feature spaces. Popular linear algorithms, such as the support vector machine [1,2] or principal component analysis [3], become more expressive if the data are first mapped onto a high-dimensional feature space. Instead of evaluating inner products explicitly in the feature space, a more efficient evaluation can be done implicitly in the original space using a positive-definite kernel function. This is known as the *kernel trick* [4]. The kernel trick does not require an explicit feature map, and hence allows us to work with infinite-dimensional feature spaces, e.g., using a Gaussian kernel. Nevertheless, most kernel-based methods scale polynomially with the size of the data sets. This problem has been tackled in the realm of quantum computation and exponential speedups have been conjectured [5,6]. However, such speedups are still highly controversial [7,8].

Recently, the cost of a single kernel evaluation was addressed by quantum computing research [9–11]. Speedups might be possible since the cost of explicitly evaluating inner products of quantum states only grows logarithmically with the system size [12], as opposed to linear on a classical computer. Schuld and Killoran further conjecture the usage of continuous-variable quantum systems for working with classically intractable, i.e., hard to compute, kernels in infinite dimensions [10], but it is unclear whether problems exist for which such kernels are beneficial. Furthermore, the recent suggestions do not address the polynomial scaling of kernel methods with the sample size, leaving the application of quantum computing in large-scale kernel methods a challenging problem.

The idea of explicitly representing an infinite-dimensional feature vector as a quantum state opens a way to tackle this problem. While it is impossible classically to sum two infinite-dimensional vectors, a quantum mechanical *superposition* of two states can be constructed explicitly, even for infinite-dimensional systems; see, e.g., [13]. On the other hand, *the evaluation of inner products in an infinite-dimensional quantum Hilbert space is independent of the number of states in a superposition*. We identify methods involving the *kernel mean embedding* [14–16] as a branch of machine-learning techniques that suffer from the fact that on a classical computer, the cost of the evaluation of inner products of sums of feature maps is not independent of the number of data points involved.

This paper is organized as follows. We start by introducing the kernel mean embedding from a classical perspective, point out the main problem it has in big data applications, and present its relevance in current machine-learning research through some real-world applications. We then define the *quantum mean embedding* as a modified version of the kernel mean embedding, which makes it suitable for investigation in the context of quantum computation, and show that this modification still allows for the usage in conventional applications. We present how the quantum mean embedding can be used, in principle, to overcome the problems faced classically and discuss the challenges left to achieve this. Finally, we sum up with a discussion of our results.

II. KERNEL MEAN EMBEDDING

Let $\mathcal{X}$ be a locally compact and Hausdorff space. A function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{C}$ is called a positive-definite kernel function, or kernel function for brevity, if, for all

*jmkuebler@tuebingen.mpg.de
†krikamol@tuebingen.mpg.de
‡bs@tuebingen.mpg.de

$n \in \mathbb{N}$, $x_1, \ldots, x_n \in \mathcal{X}$, and $c_1, \ldots, c_n \in \mathbb{C}$, it holds that $\sum_{i,j=1}^{n} c_i^* c_j k(x_i, x_j) \geqslant 0$ [4]. For every kernel function, there exists a unique reproducing kernel Hilbert space (RKHS) $\mathcal{H}_k$ such that $k(\cdot, x) \in \mathcal{H}_k$ for all $x \in \mathcal{X}$ and the *reproducing property* $f(x) = \langle f, k(\cdot, x)\rangle_{\mathcal{H}_k}$ holds for all $f \in \mathcal{H}_k$ and $x \in \mathcal{X}$. We call the mapping $\phi : \mathcal{X} \to \mathcal{H}_k$ given by $\phi(x) := k(\cdot, x)$ the *canonical* feature map of k, i.e., $k(x, y) = \langle \phi(y), \phi(x)\rangle$ [17].

Let $\mathbb{P}$ be a probability measure over $\mathcal{X}$. The kernel mean embedding (KME) of $\mathbb{P}$ is defined as [14,15]

$$\mu_{\mathbb{P}} := \int_{\mathcal{X}} k(\cdot, x) d\mathbb{P}(x) = \int_{\mathcal{X}} \phi(x) d\mathbb{P}(x). \tag{1}$$

The embedding $\mu_{\mathbb{P}}$ exists and is a function in $\mathcal{H}_k$ if $\mathbb{E}_{X\sim\mathbb{P}}[k(X, X)] < \infty$ [15]. Based on a sample $X = \{x_1, \ldots, x_n\}$ drawn from $\mathbb{P}$, an empirical estimate of $\mu_{\mathbb{P}}$ is given by the KME of the empirical distribution $\hat{\mathbb{P}} = \frac{1}{n}\sum_{i=1}^{n} \delta_{x_i}$,

$$\mu_X := \frac{1}{n}\sum_{i=1}^{n} \phi(x_i). \tag{2}$$

The kernel function k is said to be *characteristic* if the map $\mu : \mathbb{P} \mapsto \mu_{\mathbb{P}}$ is injective [18,19]. Thus, the corresponding KMEs represent all properties of a probability distribution by a function in the RKHS. The notion of characteristic kernels is closely related to the notion of universal kernels [20]. Here we call a kernel c_0 *universal* if the corresponding RKHS is dense in the space of continuous functions over $\mathcal{X}$ that vanish at infinity [21]. For c_0-universal kernels, the KME is injective even for finite-signed measures [21]. Popular universal kernels include the Gaussian kernel $k(x, y) = \exp(-\|x - y\|^2/2\sigma^2)$ and Laplacian kernel $k(x, y) = \exp(-\|x - y\|_1/\sigma)$, where σ is a bandwidth parameter [19,22].

The expressiveness of characteristic kernels comes at a price. Since there exist distributions with infinite moments, the corresponding RKHS must have infinite dimensions to prevent information loss. Consequently, it is classically impossible to represent and manipulate μ_X directly. However, if we only care about inner products of mean embeddings, which is usually the case in most algorithms, we can resort to the "kernel trick" and replace inner products with kernel evaluations [4]. That is, given independent and identically distributed (i.i.d.) samples $X = \{x_1, \ldots, x_n\}$ from $\mathbb{P}$ and $Y = \{y_1, \ldots, y_n\}$ from $\mathbb{Q}$ [23], we can evaluate

$$\langle \mu_X, \mu_Y\rangle = \frac{1}{n^2}\sum_{i,j=1}^{n} \langle \phi(x_i), \phi(y_j)\rangle = \frac{1}{n^2}\sum_{i,j=1}^{n} k(x_i, y_j)$$
$$=: K(X, Y). \tag{3}$$

The inevitable drawback of this trick is that algorithms based on $K(X, Y)$ have a runtime complexity that scales at least quadratically with the number of data points n.

In the following, we present essential applications of the KME, which suffer from the above limitation.

Learning on probability distributions. Classical machine-learning algorithms were originally developed for training data consisting of *points* in some vector space. In several domains such as astronomy and high-energy physics, however, data are represented naturally as probability distributions, e.g., clusters of galaxies and groups of collision events. The KME (1) allows us to generalize algorithms to the space of probability distributions [24–27] through the *distributional* kernel function,

$$K(\mathbb{P}, \mathbb{Q}) = \langle \mu_{\mathbb{P}}, \mu_{\mathbb{Q}}\rangle_{\mathcal{H}_k} = \iint_{\mathcal{X}} k(x, y) d\mathbb{P}(x) d\mathbb{Q}(y). \tag{4}$$

Given i.i.d. samples $X = \{x_1, \ldots, x_n\}$ from $\mathbb{P}$ and $Y = \{y_1, \ldots, y_n\}$ from $\mathbb{Q}$, $K(\mathbb{P}, \mathbb{Q})$ can be approximated by

$$K(\mathbb{P}, \mathbb{Q}) \approx \frac{1}{n^2}\sum_{i,j=1}^{n} k(x_i, y_j) = K(X, Y). \tag{5}$$

Maximum mean discrepancy (MMD). The MMD is a discrepancy measure between any two distributions $\mathbb{P}$ and $\mathbb{Q}$ [28,29]. It is given by the distance of the corresponding mean embeddings of the distributions [29, Lemma 4] and can be expressed solely in terms of inner products of mean embeddings (assuming a real kernel),

$$\mathrm{MMD}[\mathcal{H}_k, \mathbb{P}, \mathbb{Q}]^2 = \|\mu_{\mathbb{P}} - \mu_{\mathbb{Q}}\|^2$$
$$= \langle \mu_{\mathbb{P}}, \mu_{\mathbb{P}}\rangle - 2\langle \mu_{\mathbb{P}}, \mu_{\mathbb{Q}}\rangle + \langle \mu_{\mathbb{Q}}, \mu_{\mathbb{Q}}\rangle. \tag{6}$$

For characteristic kernels, $\mathrm{MMD}[\mathcal{H}_k, \mathbb{P}, \mathbb{Q}] = 0 \Leftrightarrow \mathbb{P} = \mathbb{Q}$ [29, Theorem 5]. With the samples X and Y, it is possible to estimate the MMD by evaluating (6) with the embeddings μ_X and μ_Y [29, Eq. (5)]:

$$\mathrm{MMD}[\mathcal{H}_k, X, Y]^2 = \|\mu_X - \mu_Y\|^2$$
$$= K(X, X) - 2K(X, Y) + K(Y, Y). \tag{7}$$

Deep learning. The applications of KMEs in deep learning have gained a lot of attention in the past few years. Notably, the MMD has been used as an objective function for training deep generative models [30–32]. For a deep generative model G_θ, parametrized by a parameter vector θ, the idea is to learn θ by minimizing the $\mathrm{MMD}[\mathcal{H}_k, \mathbb{P}, \mathbb{Q}_\theta]^2$, where $\mathbb{P}$ is the data distribution and $\mathbb{Q}_\theta$ is the distribution induced by the generative model G_θ. In this area, we usually deal with a huge amount of data [33].

All of the above applications require the estimation of terms such as $K(X, Y)$, which scale quadratically with the sample size n, and hence become prohibitive for large n. To enable large-scale learning with KMEs, a common approach is to approximate μ_X by a finite-dimensional representation, e.g., using random Fourier features [34] or the Nyström method [35], after which it can be manipulated explicitly. For a d-dimensional approximation, the cost drops to $O(n + d)$, which is linear in n. The downside is that the embedding defined in terms of this representation can no longer be injective, which is an essential requirement in most applications of the KME.

Recent work [10,11] showed how one can, in principle, evaluate a d-dimensional approximation of the kernel function using only $O(\log_2 d)$ qubits. Furthermore, Ref. [36] has investigated quantum kernels in the context of the MMD. Reference [37] formulates quantum graphical models in terms of the kernel mean embedding and uses a density matrix as a mean map. On the contrary, we focus on the quadratic scaling when using an infinite-dimensional feature map.

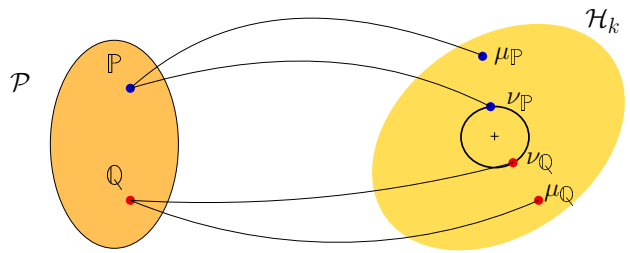

FIG. 1. Schematic comparison of the classical KME and the QME: The KME maps probability distributions $\mathbb{P}$ onto functions in the RKHS $\mathcal{H}_k$. The QME additionally enforces that the mapping is onto the unit ball (denoted by the circle) in the RKHS. Theorem 1 shows the injectivity of the QME for c_0-universal kernels. For visualization, we choose $\mathcal{H} = \mathcal{H}_k$.

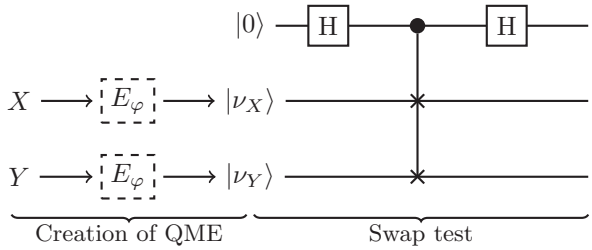

FIG. 2. The quantum approach separates the creation of the QME from the inner-product estimation. It requires two subroutines: an experimental setup E_φ that creates the QME efficiently (left), and the swap test (right), which uses an ancillary qubit to estimate inner products of arbitrary states. This approach detaches the estimation of the inner product from the sample size.

III. QUANTUM MEAN EMBEDDING

Let $\mathcal{H}$ be the Hilbert space of a quantum system and $\varphi : \mathcal{X} \to \mathcal{H}, x \mapsto |\varphi(x)\rangle$ be a quantum feature map that assigns a quantum state $|\varphi(x)\rangle$, i.e., a normalized function in $\mathcal{H}$, to each point in the input domain $x \in \mathcal{X}$ [38]. This defines a kernel $k(x, x') = \langle\varphi(x)|\varphi(x')\rangle$ [10,11] with the constraint $k(x, x) = 1$ for all $x \in \mathcal{X}$, due to the normalization of quantum states [39].

Let $\mathbb{P}$ be a probability distribution over the input domain. We define the *quantum mean embedding* (QME),

$$|\nu_{\mathbb{P}}\rangle := \frac{1}{\mathcal{N}_{\mathbb{P}}} \int_{\mathcal{X}} |\varphi(x)\rangle \, d\mathbb{P}(x), \tag{8}$$

where the normalization $\mathcal{N}_{\mathbb{P}}$ ensures the physicality of the state and is given by the norm of the corresponding KME (1), i.e., $\mathcal{N}_{\mathbb{P}} := \|\mu_{\mathbb{P}}\|_{\mathcal{H}_k}$.

The QME exists for all probability distributions due to the constraint $k(x, x) = 1$. *A subtle difference between the KME and the QME are the spaces in which the embeddings live.* While the KME is a function in the RKHS $\mathcal{H}_k$ and uniquely defined by the kernel k, the QME depends on the quantum system's Hilbert space $\mathcal{H}$ and the choice of the feature map φ.

Even though the embeddings live in different spaces, for any two probability distributions $\mathbb{P}$ and $\mathbb{Q}$, we have

$$\langle\mu_{\mathbb{P}}, \mu_{\mathbb{Q}}\rangle_{\mathcal{H}_k} = \mathcal{N}_{\mathbb{P}} \cdot \mathcal{N}_{\mathbb{Q}} \, \langle\nu_{\mathbb{P}}|\nu_{\mathbb{Q}}\rangle_{\mathcal{H}} \, . \tag{9}$$

That is, their inner products have a fixed relation independent of $\mathcal{H}$. Hence, the important difference is that the QME maps every probability distribution on the unit sphere in a Hilbert space, whereas the KME does not enforce this; see Fig. 1. In the following theorem, we show that if the kernel is c_0 universal, we do not lose information about a probability measure when using the QME.

Theorem 1. Injectivity of the QME. Let $\mathcal{P}$ be the space of Borel probability measures over the measurable space $(\mathcal{X}, \mathcal{A})$, where $\mathcal{A}$ denotes the Borel σ algebra. Let $\varphi : \mathcal{X} \to \mathcal{H}, x \mapsto |\varphi(x)\rangle$ be a mapping such that $k(x, y) = \langle\varphi(x)|\varphi(y)\rangle$. If k is a c_0-universal kernel, the QME (8) is injective over $\mathcal{P}$, i.e., $|\nu_{\mathbb{P}}\rangle = |\nu_{\mathbb{Q}}\rangle \Leftrightarrow \mathbb{P} = \mathbb{Q}$ for any $\mathbb{P}, \mathbb{Q} \in \mathcal{P}$.

The proof is included in the Appendix.

For a finite sample X, we define an empirical QME as

$$|\nu_X\rangle := \frac{1}{\mathcal{N}_X} \frac{1}{n} \sum_{i=1}^{n} |\varphi(x_i)\rangle \, , \tag{10}$$

with the normalization constant

$$\mathcal{N}_X = \|\mu_X\|_{\mathcal{H}_k} = \sqrt{\frac{1}{n^2} \sum_{i,j=1}^{n} k(x_i, x_j)}. \tag{11}$$

As discussed before, for infinite-dimensional feature maps, the KME cannot be described explicitly and only used via inner products. The advantage of the QME is that it is possible, in principle, to explicitly create $|\nu_X\rangle$ in the laboratory, even for infinite-dimensional cases. Here it is important that an experimenter only needs to create a state that is proportional to $\sum_{i=1}^{n} |\varphi(x_i)\rangle$. The prefactor (11) is enforced by the laws of physics and is not required for the state preparation. Given this explicit representation, it allows us to decouple the cost of the inner-product evaluation from the sample size n; see Fig. 2.

Conjecture 1. Suppose we are given a routine that prepares states of the form (10) with cost $O(n)$ for a feature map φ. In addition, we are given a routine that can evaluate inner products of arbitrary states in $\mathcal{H}$ in constant time. Then, for two samples $X = \{x_1, \ldots, x_n\}$ and $Y = \{y_1, \ldots, y_n\}$, one can evaluate $K(X, Y)$, defined in Eq. (3), with cost $O(n)$, whereas a classical computer scales with $O(n^2)$.

Proof. By assumption, we can prepare $|\nu_X\rangle$ and $|\nu_Y\rangle$ with linear cost in n. Furthermore, we can evaluate $\langle\nu_X|\nu_Y\rangle$ in constant time, given the individual states. Together the cost of evaluating the term $\langle\nu_X|\nu_Y\rangle$ scales, at most, with $O(n)$. The normalizations $\mathcal{N}_X$ and $\mathcal{N}_Y$ can also be estimated with cost $O(n)$; see Sec. IV. Using relation (9), we obtain

$$K(X, Y) = \langle\mu_X, \mu_Y\rangle_{\mathcal{H}_k} = \mathcal{N}_X \mathcal{N}_Y \, \langle\nu_X|\nu_Y\rangle_{\mathcal{H}} \, . \tag{12}$$

■

Compared to the classical KME, this conjecture implies that under the stated assumptions, it is possible to simultaneously reduce the cost of the QME while preserving its expressibility guarantee given in Theorem 1.

Given an efficient evaluation of $K(X, Y)$, it is possible to speed up the methods presented earlier, which rely on inner products of the KMEs. In Sec. IV, we discuss the assumptions of Conjecture 1.

Apart from using the QME to speed up the evaluation of inner products of the KMEs, it follows from the proof of Theorem 1 that the QME is also important on its own, as it can uniquely represent probability distributions. However, it is unclear to what extent the applications of the KME could

be rephrased solely in terms of inner products of the QME instead of taking the detour over $K(X, Y)$, where we need to determine the normalizations.

IV. CHALLENGES

In order to harvest a potential quantum speedup, it is necessary to create the QME efficiently, i.e., with resources and time linear in the sample size. We phrase this as the first challenge:

Given a quantum feature map φ, find an experimental strategy, denoted E_φ, such that for an arbitrary input sample $X = \{x_1, \ldots, x_n\}$, with $n \in \mathbb{N}$, it creates $|\nu_X\rangle$, using resources that scale, at most, linear in n.

In the case of coherent states as the feature map (see the Appendix), superpositions similar to $|\nu_X\rangle$ have already been experimentally realized for specific cases and are known as "cat states" [13,40,41]. However, it is an open question how these approaches scale, even theoretically, for superposing a large number of states; see [42] for an overview on similar experimental approaches. In general, the rigorous study of resources required to construct superpositions of quantum states and the connections to entanglement are the subject of current research [43,44]. Particularly for the case of superpositions of nonorthogonal states, as is the case for our proposed embedding, the theory becomes more involved; see Sec. III.K.4 of Ref. [44].

Note that we explicitly allow for an experimental setup E_φ that is specific to the given quantum feature map φ, i.e., a specific kernel function. This is necessary because a universal machine that builds a superposition of completely arbitrary and unknown quantum states cannot exist [45,46]. Furthermore, we emphasize that this work does not require a qRAM [47].

Given the QMEs, at the core of our approach lies the estimation of the inner product of two arbitrary quantum states in $\mathcal{H}$. Formally, this can be done by using the *swap test* [48]; see right side of Fig. 2. The swap test works independently of the input states, which for our purpose we denote by $|\nu_X\rangle, |\nu_Y\rangle \in \mathcal{H}$. These inputs are each in one register and a single ancilla qubit in the state $|0\rangle$ in an additional register. The test itself consists of a Hadamard transformation H on the qubit, followed by a controlled swap of the two states conditioned on the state of the qubit, and another Hadamard transformation on the qubit. This circuit maps the initial state $|0\rangle\,|\nu_X\rangle\,|\nu_Y\rangle$ onto

$$\frac{|0\rangle\,(|\nu_Y\rangle\,|\nu_X\rangle + |\nu_X\rangle\,|\nu_Y\rangle) + |1\rangle\,(|\nu_Y\rangle\,|\nu_X\rangle - |\nu_X\rangle\,|\nu_Y\rangle)}{2};$$

see [48, Eq. (4)]. At the end, the qubit is measured in the computational basis. This results in outcome 0 with probability $p_0 = (1 + |\langle\nu_X|\nu_Y\rangle|^2)/2$ and outcome 1 with probability $p_1 = 1 - p_0$. Repetitive application of this routine allows for an estimation of p_0 and p_1, from which one can infer $|\langle\nu_X|\nu_Y\rangle|^2 = 2p_0 - 1$. When using a Gaussian kernel, we know *a priori* that $\langle\nu_X|\nu_Y\rangle > 0$, and thus $\langle\nu_X|\nu_Y\rangle = \sqrt{2p_0 - 1}$. If we cannot guarantee the positivity of $\langle\nu_X|\nu_Y\rangle$, we need a phase-sensitive estimation of inner products, as discussed in the supplemental material of [10]. Crucially, the swap test works independently of the size of the samples X and Y.

For finite-dimensional systems, Ref. [12] recently proposed an implementation that scales logarithmically with the dimension of the Hilbert space. But this approach does not translate to systems of infinite dimension. The infinite-dimensional case has been studied in Refs. [49–51]. However, they do not give an explicit solution and we are not aware of any experimental realization of a universal swap test for the infinite-dimensional case. This marks the second challenge arising from this paper.

At the stage of preparing superpositions in the form of (10) on a quantum device, it is not necessary to know the value of the normalization $\mathcal{N}_X$. However, if the goal is to estimate $K(X, Y)$ with the help of a quantum device, then knowledge of the normalizations is needed; see (12). The naive approach, i.e., using its definition (11), takes $O(n^2)$ operations and would prohibit the polynomial advantage. In the Appendix, we show how one can estimate $\mathcal{N}_X$. The suggested strategy only relies on the previous two challenges and hence does not pose a difficulty by itself.

V. CONCLUSION

In this work, we adapted the concept of kernel mean embeddings to quantum mechanics, by defining the quantum mean embedding. While the kernel mean embedding maps a probability distribution to a function in a reproducing kernel Hilbert space, the quantum mean embedding can only map onto the unit sphere of a Hilbert space, a necessity that arises due to the normalization of quantum states. Despite this additional constraint, we showed that the quantum mean embedding is still injective if the induced kernel is c_0 universal. Since the quantum mean embedding can, in principle, be created in the laboratory, it allows for a polynomial speedup when computing inner products between mean embeddings of empirical distributions. We highlighted the relevance of this task by describing use cases in recent machine-learning applications. We made explicit which requirements need to be fulfilled by the quantum hardware in order to harvest the polynomial advantage.

This work opens multiple paths for further research; for example, on the quantum side, the experimental creation of superpositions of a large number of states and the estimation of inner products thereof. Furthermore, the quantum mean embedding is a way of encoding probability distributions in quantum states, which allows us to use the results known from the kernel theory. For machine-learning research, it is an open question what the possible applications of the embedding of probability distributions onto the unit sphere in the reproducing kernel Hilbert space could be.

ACKNOWLEDGMENT

We would like to thank C. J. Simon-Gabriel for his advice on universal and characteristic kernels.

APPENDIX

Proof of Theorem 1. We make the proof in terms of the canonical feature map ϕ, which maps into the RKHS. The

validity for any mapping $\varphi : \mathcal{X} \to \mathcal{H}$ that leads to the same kernel function is then trivial.

Let $\mathcal{M}(\mathcal{X}, \mathcal{A})$ denote the set of finite non-negative measures on the measurable space $(\mathcal{X}, \mathcal{A})$, i.e., $\xi(\mathcal{X}) < \infty$ for all $\xi \in \mathcal{M}(\mathcal{X}, \mathcal{A})$. We can extend the definition of the kernel mean embedding (1) to $\mathcal{M}(\mathcal{X}, \mathcal{A})$ by defining

$$\mu_\xi = \int_{\mathcal{X}} k(\cdot, x) d\xi(x) = \int_{\mathcal{X}} \phi(x) d\xi(x), \tag{A1}$$

for any $\xi \in \mathcal{M}(\mathcal{X}, \mathcal{A})$ that fulfills $\int_{\mathcal{X}} k(x, x) d\xi(x) < \infty$. Let ξ_1 and ξ_2 be arbitrary measures in $\mathcal{M}(\mathcal{X}, \mathcal{A})$. By assumption, k is universal over $\mathcal{C}_0(\mathcal{X})$ and thus characteristic over $\mathcal{M}(\mathcal{X}, \mathcal{A})$, i.e., $\mu_{\xi_1} = \mu_{\xi_2} \Leftrightarrow \xi_1 = \xi_2$; see Theorem 6 in Ref. [21].

Define $\nu_{\mathbb{P}}$ as the mean embedding onto the unit sphere of the RKHS,

$$\nu_{\mathbb{P}} := \frac{1}{\mathcal{N}_{\mathbb{P}}} \mu_{\mathbb{P}}, \tag{A2}$$

with $\mathcal{N}_{\mathbb{P}} \in \mathbb{R}^+$ such that $\|\nu_{\mathbb{P}}\|_{\mathcal{H}_k} = 1$. Let $\mathbb{P}$ and $\mathbb{Q}$ be probability measures for which the embedding onto the unit sphere (A2) coincide, i.e., $\nu_{\mathbb{P}} = \nu_{\mathbb{Q}}$. We can relate this to the kernel mean embeddings as

$$\mu_{\mathbb{P}} = \mathcal{N}_{\mathbb{P}} \, \nu_{\mathbb{Q}} = \frac{\mathcal{N}_{\mathbb{P}}}{\mathcal{N}_{\mathbb{Q}}} \mu_{\mathbb{Q}} = \mu_\xi, \tag{A3}$$

where we defined the finite non-negative measure $\xi = \frac{\mathcal{N}_{\mathbb{P}}}{\mathcal{N}_{\mathbb{Q}}} \mathbb{Q}$, using the linearity of (A1). With the injectivity of the embedding (A1), this implies $\mathbb{P} = \xi = \frac{\mathcal{N}_{\mathbb{P}}}{\mathcal{N}_{\mathbb{Q}}} \mathbb{Q}$. By assumption, $\mathbb{P}$ and $\mathbb{Q}$ are probability measures and fulfill $\mathbb{P}(\mathcal{X}) = \mathbb{Q}(\mathcal{X}) = 1$. This implies $\frac{\mathcal{N}_{\mathbb{P}}}{\mathcal{N}_{\mathbb{Q}}} = 1$ and thus $\mathbb{P} = \mathbb{Q}$, which proves the injectivity of ν for the set of probability distributions. ■

1. Coherent states and Gaussian kernel

In this section, we consider an explicit example, previously reported in Ref. [9]. Let $\mathcal{H}$ be an infinite-dimensional (complex) Hilbert space, with orthonormal basis $\{|n\rangle\}_{n \in \mathbb{N}_0}$. This could, for example, be the space corresponding to a single mode of the electromagnetic field [52]. For simplicity, we consider $\mathcal{X} = \mathbb{R}$ and define the feature map $\varphi : \mathbb{R} \to \mathcal{H}$ as

$$|\varphi(x)\rangle = e^{-\frac{1}{2}x^2} \sum_{n=0}^{\infty} \frac{x^n}{\sqrt{n!}} |n\rangle \, . \tag{A4}$$

In quantum optics, the states $|n\rangle$ are called Fock states. States of the form (A4) are called coherent states and are well studied [53]. In the context of this paper, however, the nature of the basis and hence the exact form of the Hilbert space are unimportant. The important part is the orthonormality of the basis states, which implies

$$\langle \varphi(x) | \varphi(x') \rangle = e^{-\frac{1}{2}(x-x')^2} =: k(x, x'), \tag{A5}$$

for arbitrary $x, x' \in \mathbb{R}$, and defines the popular Gaussian kernel [4]. By composing the mapping (A4) with the mapping $x \mapsto \frac{x}{\sigma}$, for some $\sigma > 0$, it is also possible to include a bandwidth parameter σ. The Gaussian kernel fulfills the requirements of Theorem 1 (see [21, theorem 17]). Therefore, it is possible to construct an injective embedding of probability distributions over the real numbers in a superposition of coherent states.

Coherent states are commonly considered the *most classical* states in quantum optics and are easy to simulate on a classical device. Working with a quantum device becomes interesting when the states become *nonclassical* [52]. When using the coherent feature map (A4), the embedding of a sample (10) corresponds to the *cat states* [13,40,41]. Cat states are considered nonclassical, as their Wigner function attains negative values. From a quantum perspective, this already hints at the difficulties encountered when working with such states on classical devices.

2. Estimation of $\mathcal{N}_X$

In order to obtain $\mathcal{N}_X$ without explicitly calculating (11), we can evaluate $\mathcal{N}_X$ by estimating the inner product with a reference state $|\psi_{\text{ref}}\rangle = |\varphi(x_{\text{ref}})\rangle$ for some reference value $x_{\text{ref}} \in \mathcal{X}$. To this end, we analytically calculate

$$c := \frac{1}{n} \sum_{i=1}^{n} \langle \psi_{\text{ref}} | \varphi(x_i) \rangle = \frac{1}{n} \sum_{i=1}^{n} k(x_{\text{ref}}, x_i), \tag{A6}$$

using $O(n)$ operations. Now given the preparation of $|\nu_X\rangle$ and of $|\psi_{\text{ref}}\rangle$, we can experimentally evaluate the inner product $\langle \psi_{\text{ref}} | \nu_X \rangle$ and from this obtain the normalization $\mathcal{N}_X = c \langle \psi_{\text{ref}} | \nu_X \rangle^{-1}$. Obviously, in order to make this well defined, we need to choose the reference function such that $\langle \psi_{\text{ref}} | \nu_X \rangle \neq 0$. This strategy relies on the two challenges phrased in the main text, i.e., the preparation of $|\nu_X\rangle$ and the estimation of inner products, but apart from this does not pose an extra difficulty by itself.

We emphasize again that due to Theorem 1, it should be possible to come up with algorithms that directly work with the QME and hence make the estimation of the normalization superfluous.

[1] C. Cortes and V. Vapnik, Mach. Learn. **20**, 273 (1995).

[2] I. Steinwart and A. Christmann, *Support Vector Machines*, 1st ed. (Springer, New York, 2008).

[3] H. Hotelling, J. Educ. Psychol. **24**, 417 (1933).

[4] B. Schölkopf and A. J. Smola, *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond* (MIT Press, Cambridge, 2001).

[5] P. Rebentrost, M. Mohseni, and S. Lloyd, Phys. Rev. Lett. **113**, 130503 (2014).

[6] S. Lloyd, M. Mohseni, and P. Rebentrost, Nat. Phys. **10**, 631 (2014).

[7] S. Aaronson, Nat. Phys. **11**, 291 (2015).

[8] C. Ciliberto, M. Herbster, A. D. Ialongo, M. Pontil, A. Rocchetto, S. Severini, and L. Wossnig, Proc. R. Soc. A **474**, 20170551 (2018).

[9] R. Chatterjee and T. Yu, Quantum Inf. Comput. **17**, 1292 (2017).

[10] M. Schuld and N. Killoran, Phys. Rev. Lett. **122**, 040504 (2019).

[11] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, Nature (London) **567**, 209 (2019).

[12] L. Cincio, Y. Subaşı, A. T. Sornborger, and P. J. Coles, New J. Phys. **20**, 113022 (2018).

[13] B. Vlastakis, G. Kirchmair, Z. Leghtas, S. E. Nigg, L. Frunzio, S. M. Girvin, M. Mirrahimi, M. H. Devoret, and R. J. Schoelkopf, Science **342**, 607 (2013).

[14] A. Berlinet and C. Thomas-Agnan, *Reproducing Kernel Hilbert Spaces in Probability and Statistics* (Springer, Boston, 2004).

[15] A. Smola, A. Gretton, L. Song, and B. Schölkopf, in *Algorithmic Learning Theory*, edited by M. Hutter, R. A. Servedio, and E. Takimoto (Springer, Berlin, 2007), pp. 13–31.

[16] K. Muandet, K. Fukumizu, B. Sriperumbudur, and B. Schölkopf, Found. Trends Mach. Learn. **10**, 1 (2017).

[17] N. Aronszajn, Trans. Am. Math. Soc. **68**, 337 (1950).

[18] K. Fukumizu, A. Gretton, X. Sun, and B. Schölkopf, in *Advances in Neural Information Processing Systems 20*, edited by J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis (Curran Associates, 2008), pp. 489–496.

[19] B. K. Sriperumbudur, A. Gretton, K. Fukumizu, B. Schölkopf, and G. R. Lanckriet, J. Mach. Learn. Res. **11**, 1517 (2010).

[20] I. Steinwart, J. Mach. Learn. Res. **2**, 67 (2001).

[21] C.-J. Simon-Gabriel and B. Schölkopf, J. Mach. Learn. Res. **19**, 1 (2018).

[22] K. Fukumizu, F. R. Bach, and M. I. Jordan, J. Mach. Learn. Res. **5**, 73 (2004).

[23] For simplicity, we assume the sample sizes to be equal.

[24] K. Muandet, K. Fukumizu, F. Dinuzzo, and B. Schölkopf, in *Advances in Neural Information Processing Systems 25*, edited by F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger (Curran Associates, 2012), pp. 10–18.

[25] K. Muandet and B. Schölkopf, in *Proceedings 29th Conference on Uncertainty in Artificial Intelligence (UAI)*, edited by A. Nicholson and P. Smyth (AUAI Press, Corvallis, OR, 2013), pp. 449–458.

[26] D. Lopez-Paz, K. Muandet, B. Schölkopf, and I. Tolstikhin, in *Proceedings of the 32nd International Conference on Machine Learning*, Vol. 37, edited by F. Bach and D. Blei (PMLR, 2015), pp. 1452–1461.

[27] Z. Szabó, B. K. Sriperumbudur, B. Póczos, and A. Gretton, J. Mach. Learn. Res. **17**, 1 (2016).

[28] K. Borgwardt, A. Gretton, M. Rasch, H.-P. Kriegel, B. Schölkopf, and A. Smola, Bioinformatics **22**, 49 (2006).

[29] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, and A. Smola, J. Mach. Learn. Res. **13**, 723 (2012).

[30] G. K. Dziugaite, D. M. Roy, and Z. Ghahramani, in *Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence*, edited by M. Meila and T. Heskes (AUAI Press, Arlington, VA, 2015), pp. 258–267.

[31] Y. Li, K. Swersky, and R. Zemel, in *Proceedings of the 32nd International Conference on Machine Learning*, edited by F. Bach and D. Blei (Lille, France, 2015), pp. 1718–1727.

[32] C.-L. Li, W.-C. Chang, Y. Cheng, Y. Yang, and B. Póczos, in *Advances in Neural Information Processing Systems*, edited by I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett (Curran Associates, Inc., 2017), pp. 2203–2213.

[33] Y. LeCun, Y. Bengio, and G. Hinton, Nature (London) **521**, 436 (2015).

[34] A. Rahimi and B. Recht, in *Advances in Neural Information Processing Systems 20*, edited by J. C. Platt, D. Koller, Y. Singer, and S. T. Roweis (Curran Associates, 2008), pp. 1177–1184.

[35] C. K. I. Williams and M. Seeger, in *Advances in Neural Information Processing Systems 13*, edited by T. K. Leen, T. G. Dietterich, and V. Tresp (MIT Press, Cambridge, 2001), pp. 682–688.

[36] B. Coyle, D. Mills, V. Danos, and E. Kashefi, arXiv:1904.02214v2.

[37] S. Srinivasan, C. Downey, and B. Boots, in *Advances in Neural Information Processing Systems 31*, edited by S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett (Curran Associates, 2018), pp. 10338–10347.

[38] In order to emphasize that we deal with a quantum state, we shall abuse notation by denoting the image of a point x under the mapping φ as $|\varphi(x)\rangle$ instead of $\varphi(x)$.

[39] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th ed. (Cambridge University Press, Cambridge, 2010).

[40] S. Deléglise, I. Dotsenko, C. Sayrin, J. Bernu, M. Brune, J.-M. Raimond, and S. Haroche, Nature (London) **455**, 510 (2008).

[41] A. Ourjoumtsev, H. Jeong, R. Tualle-Brouri, and P. Grangier, Nature (London) **448**, 784 (2007).

[42] U. L. Andersen, J. S. Neergaard-Nielsen, P. van Loock, and A. Furusawa, Nat. Phys. **11**, 713 (2015).

[43] T. Theurer, N. Killoran, D. Egloff, and M. B. Plenio, Phys. Rev. Lett. **119**, 230401 (2017).

[44] A. Streltsov, G. Adesso, and M. B. Plenio, Rev. Mod. Phys. **89**, 041003 (2017).

[45] U. Alvarez-Rodriguez, M. Sanz, L. Lamata, and E. Solano, Sci. Rep. **5**, 11983 (2015).

[46] M. Oszmaniec, A. Grudka, M. Horodecki, and A. Wójcik, Phys. Rev. Lett. **116**, 110403 (2016).

[47] V. Giovannetti, S. Lloyd, and L. Maccone, Phys. Rev. Lett. **100**, 160501 (2008).

[48] H. Buhrman, R. Cleve, J. Watrous, and R. de Wolf, Phys. Rev. Lett. **87**, 167902 (2001).

[49] R. Filip, Phys. Rev. A **65**, 062320 (2002).

[50] K. L. Pregnell, Phys. Rev. Lett. **96**, 060501 (2006).

[51] H. Jeong, C. Noh, S. Bae, D. G. Angelakis, and T. C. Ralph, J. Opt. Soc. Am. B **31**, 3057 (2014).

[52] D. V. Strekalov and G. Leuchs, in *Quantum Photonics: Pioneering Advances and Emerging Applications*, edited by R. Boyd, S. Lukishova, and V. Zadkov (Springer Nature, Switzerland, 2019).

[53] G. S. Agarwal, *Quantum Optics* (Cambridge University Press, Cambridge, 2012).