The intrinsic probabilistic nature of quantum mechanics invokes endeavors of designing quantum generative learning models (QGLMs) with computational advantages over classical ones. To date, two prototypical QGLMs are quantum circuit Born machines (QCBMs) and quantum generative adversarial networks (QGANs), which approximate the target distribution in explicit and implicit ways, respectively. Despite the empirical achievements, the fundamental theory of these models remains largely obscure. To narrow this knowledge gap, here we explore the learnability of QCBMs and QGANs from the perspective of generalization when their loss is specified to be the maximum mean discrepancy. Particularly, we first analyze the generalization ability of QCBMs and identify their superiorities when the quantum devices can directly access the target distribution and the quantum kernels are employed. Next, we prove how the generalization error bound of QGANs depends on the employed Ansatz, the number of qudits, and input states. This bound can be further employed to seek potential quantum advantages in Hamiltonian learning tasks. Numerical results of QGLMs in approximating quantum states, Gaussian distribution, and ground states of parameterized Hamiltonians accord with the theoretical analysis. Our work opens the avenue for quantitatively understanding the power of quantum generative learning models.

I. INTRODUCTION

Learning is a generative activity that constructs its own interpretations of information and draws inferences on them [1]. This comprehensive philosophy sculpts a substantial subject in artificial intelligence, which is designing powerful generative learning models (GLMs) to capture the distribution $Q$ describing the real-world data shown in Fig. 1(a). Concisely, a fundamental concept behind GLMs is estimating $Q$ by a tunable probability distribution $P_{\theta}$. In the past decades, a plethora of GLMs, e.g., the Helmholtz machine [4], variational auto-encoders [5], and generative adversarial networks (GANs) [6, 7], have been proposed. Attributed to the efficacy and flexibility of handling $P_{\theta}$, these GLMs have been broadly applied to myriad scientific domains and gained remarkable success, including image synthesis and editing [8–10], medical imaging [11], molecule optimization [12, 13], and quantum computing [14–19]. Despite the wide success, their limitations have recently been recognized from different perspectives. Concretely, energy-based GLMs suffer from the expensive runtime of estimating and sampling the partition function [20]; variational auto-encoders tend to produce unrealistic and blurry samples when applied to complex datasets [21]; GANs encounter the issues of model collapse, divergence, and inferior performance of simulating discrete distributions [22–25].

Envisioned by the intrinsic probabilistic nature of quantum mechanics and the superior power of quantum computers [20, 28], quantum generative learning models (QGLMs) are widely believed to further enhance the ability of GLMs. Concrete evidence has been provided by Refs. [29, 30], showing that when the fault-tolerant quantum computers are available, QGLMs could surpass GLMs with provable quantum advantages such as stronger model expressivity and exponential speedups. Since fault-tolerant quantum computing is still in absence, attention has recently shifted to design QGLMs that can be efficiently carried out on noisy intermediate-scale quantum (NISQ) machines [31–33] with computational advantages on certain tasks [34–36]. Toward this goal, a leading strategy is constructing QGLMs through variational quantum algorithms [38, 39]. These QGLMs can be mainly divided into two categories, depending on whether the probability distribution $P_{\theta}$ is explicitly formulated or not. For the explicit QGLMs, a variational quantum Ansatz $U(\theta)$ (a.k.a., parameterized quantum circuit [10]) forms the distribution $P_{\theta}$. Primary protocols belonging to this class are quantum circuit Born machines (QCBMs) [41–44], quantum variational auto-encoders [45], and quantum Boltzmann machines [46, 47]. As for the implicit QGLMs, a mainstream protocol is quantum generative adversarial networks (QGANs) [48–57]. Different from QCBMs, the Ansatz $U(\theta)$ in QGANs implicitly constructs $P_{\theta}$ in the sense that the output of the quantum circuit amounts to an example sampled from $P_{\theta}$ [58]. Extensive experimental studies have demonstrated the feasibility of QGLMs for different learning tasks, e.g., image generation [50, 59], state approximation [54, 60], and drug design [61, 62].

A crucial vein in quantum machine learning [27] is understanding the learnability of a given quantum learning model from the perspective of generalization [63, 64]. A QGLM with good generalization means that the population distance between $P_{\theta}$ and $Q$ is closed to the empirical distance between the empirical distributions of $P_{\theta}$ and $Q$ [65–69]. In this respect, generalization can
QGLMs may directly access it without sampling. (c) The paradigm of QCBMs with MMD loss. Through the lens of the statistical learning theory, we separately unveil the power of QGLMs towards discrete and continuous distributions. Through the lens of the statistical learning theory, we separately unveil the power of QGLMs towards discrete and continuous distributions. (d) The scheme of QGANs with MMD loss for the continuous distribution. (e) Ansatzes. The key challenges in demystifying the generalization of QGLMs are imposed by two factors: as depicted in Fig. 1(a)-(f), the distribution $Q$ for QGLMs, the way of implementation of QGLMs, and the selection of the objective function and the employed Ansatz are diverse; the evaluation of the distance between $P_{\theta}$ and $Q$ is intricate due to the curse of dimensionality.

To shrink the above knowledge gap, here we understand the learnability of QGLMs with the maximum mean discrepancy (MMD) \cite{78}. The attention on MMD loss originates from the fact that many QGLMs employ it as the loss function to measure the difference of two distributions \cite{42-44, 51}. Through the lens of the statistical learning theory, \cite{70}, we separately unveil the power of QGLMs towards discrete and continuous distributions. That is, when $Q$ is discrete and can be efficiently accessed by quantum machines, we prove that quantum kernels can greatly benefit the generalization ability of QCBMs over their classical counterparts. Meanwhile, to attain the similar generalization error, the required computational overhead for QCBM with classical kernels is significantly larger than that of QCBM with quantum kernels. This separation advocates to use quantum kernels to underscore potential quantum advantages of QGLMs. When $Q$ is continuous, we connect the generalization with model expressivity and then leverage a statistical tool—covering number, to quantify the generalization of QGANs. Concisely, we prove that the generalization error of QGAN is upper bounded by $O(1/n + 1/m + Nd^2\sqrt{N_{gt}N_{ge}/n})$, where $n$, $m$, $d$, $k$, $N_{ge}$, and $N_{gt}$ refer to the number of referenced samples, the number of training examples, the number of encoding gates, and the number of trainable parameters, respectively. This bound explicitly exhibits how the encoding strategy and the adopted Ansatz affect the generalization error, which can not only provide practical guidance in designing advanced QGANs, but also contributes to discover potential advantages of QGLMs.

II. MAIN RESULTS

To better present our main results, let us first recap QCBMs and QGANs, introduce the maximum mean discrepancy loss and the measure of generalization, and exhibit some properties of QGLMs with MMD loss.
QCBMs. The paradigm of QCBM is shown in Fig. 1(c). Specifically, an $N$-qubit Ansatz $\hat{U}(\theta)$ with $\theta \in \Theta$ is applied to a fixed input state $\rho_0 = (|0\rangle^\otimes N)/N$ to form the parameterized distribution $P_{\theta} \in P_{\Theta}$. The probability of sampling $i \in [2^N]$ over the distribution $P_{\theta}$ yields

$$P_{\theta}(i) = \text{Tr}(\Pi_i \hat{U}(\theta) \rho_0 \hat{U}(\theta)^\dagger), \quad (1)$$

where $\Pi_i = |i\rangle \langle i|$ refers to the projector of the computational basis $i$. Given $n$ examples $\{x^{(j)}\}_{j=1}^n$ sampled from $P_{\theta}$, its empirical distribution is defined as $P_{\theta}^{(n)}(i) = \frac{1}{n} \sum_{j=1}^n \delta_{x^{(j)}(i)}$ with $\delta_{x^{(j)}}(\cdot)$ being the indicator.

QGANs. The schematic of QGANs is shown in Fig. 1(d). An $N$-qubit Ansatz $\hat{U}(\theta)$ is used to realize the generator $G_{\theta}(\cdot)$, which maps an example $z$ sampled from a prior distribution $P_z$ to the generated example $x$, i.e., $x := G_{\theta}(z) \in \mathbb{R}^{2N}$. Formally, the $j$-th component of $\mathbf{x}$ yields

$$x_j = \text{Tr}(\Pi_j \hat{U}(\theta) \rho_z \hat{U}(\theta)^\dagger), \quad \forall j \in [2^N], \quad (2)$$

where $\rho_z$ refers to the encoded quantum state of $z$. Given $n$ examples $\{x^{(j)}\}_{j=1}^n$ produced by $\{G_{\theta}(z^{(i)})\}_{i=1}^n$, its empirical distribution is

$$P_{\theta}^{(n)}(dx) = \frac{1}{n} \sum_{i=1}^n \delta_{G_{\theta}(z^{(i)})}(dx)/n.$$  

Notably, when $Q$ is discrete, the mechanism of QGANs is equivalent to QCBMs (refer to SM A for details). Due to this reason, in this study we only focus on applying QGANs to estimate the continuous distribution $Q$.

Throughout the whole study, the Ansatz employed in both QCBMs and QGANs takes the generic form

$$\hat{U}(\theta) = \prod_{l=1}^{N_{a}} \hat{U}_l(\theta), \quad (3)$$

where $\theta$ are trainable parameters living in the parameter space $\Theta$, $\hat{U}_l(\theta) \in U(d^k)$ refers to the $l$-th quantum gate operated with at most $k$-qubits with $k \leq N$, and $U(d^k)$ denotes the unitary group in dimension $d^k$ ($d = 2$ for qubits) [82]. The form of $\hat{U}(\theta)$ covers almost all Ansätze in VQAs and some constructions are given in Fig. 1(e).

Maximum Mean Discrepancy. Suppose that an unknown distribution $Q$ and a parametrized family of model distributions $P_{\Theta}$ on the same space. The maximum mean discrepancy (MMD) loss [78], which measures the difference between $P_{\theta} \in P_{\Theta}$ and $Q$, is $\text{MMD}^2(P_{\theta} || Q) = \mathbb{E}(k(x, x')) + \mathbb{E}(k(y, y')) - 2 \mathbb{E}(k(x, y))$, where the expectations are taken over the randomness of $x, x'$ $\sim P_{\theta}$ and $y, y' \sim Q$, and $k(\cdot, \cdot)$ denotes a predefined kernel (e.g., linear, radial basis function (RBF) kernels). Without loss of generality, we assume $\max_{\theta \in \Theta} \text{MMD}^2(P_{\theta} || Q) \leq C_1$ with $C_1$ being a constant. QGLMs aim to find an estimator minimizing MMD loss,

$$\hat{\theta} = \arg \min_{\theta \in \Theta} \text{MMD}^2(P_{\theta} || Q). \quad (4)$$

If the distribution $P_{\theta}$ and $Q$ cannot be accessed directly, the evaluation of expectation becomes intractable and we instead consider the empirical MMD loss, as an unbiased estimator of the MMD loss proposed by [78], i.e., $\text{MMD}^2_\delta(P_{\theta} || Q^m) := \frac{1}{m(m-1)} \sum_{i \neq j}^m k(x^{(i)}, x^{(j)}) + \frac{1}{m(m-1)} \sum_{i \neq j}^m k(y^{(i)}, y^{(j)}) - \frac{1}{2m} \sum_{i,j} k(x^{(i)}, y^{(j)})$. The minimizer of $\text{MMD}^2_\delta$ yields

$$\hat{\theta}^{(m,m)} = \arg \min_{\theta \in \Theta} \text{MMD}^2_\delta(P_{\theta} || Q^m),  \quad (5)$$

where $P_{\theta}$ and $Q^m$ separately refers to the empirical distribution of $P_{\theta}$ and $Q$ defined above. See SM B for optimizing QCBMs and QGANs using MMD loss.

We follow the classical routine [81] to define the generalization error of QGLMs as follows. When either the kernel or the target distribution $Q$ is classical, the generalization error of QGLMs yields

$$\mathcal{R}^C = \text{MMD}^2(P_{\theta}^{(m,m)} || Q) - \inf_{\theta \in \Theta} \text{MMD}^2(P_{\theta} || Q). \quad (6)$$

When the kernel $k(\cdot, \cdot)$ is quantum [82] and the distribution $Q$ can be efficiently accessed by quantum machines, the generalization error of QGLMs yields

$$\mathcal{R}^Q = \text{MMD}^2(P_{\theta} || Q) - \inf_{\theta \in \Theta} \text{MMD}^2(P_{\theta} || Q). \quad (7)$$

Intuitively, both $\mathcal{R}^C$ and $\mathcal{R}^Q$ evaluate the divergence of the estimated and the optimal MMD loss, where a lower $\mathcal{R}^C$ or $\mathcal{R}^Q$ implies a better learning performance.

Quantum kernel in MMD loss. The choice of the kernel $k(\cdot, \cdot)$ in Eq. (4) is flexible. As shown in the right panel of Fig. 1(c), when it is specified to be the quantum kernel (e.g., the linear kernel) and $Q$ can be directly accessed by QGLMs (i.e., $Q$ can be efficiently prepared by a quantum state), the corresponding MMD loss can be efficiently calculated.

Lemma 1. Suppose the distribution $Q$ can be directly accessed by QGLMs. When the quantum kernel is adopted, the MMD loss in Eq. (4) can be estimated within an error $\epsilon$ in $O(1/\epsilon^2)$ sample complexity.

The proof of Lemma 1 is provided in SM C. This lemma delivers a crucial message such that when both $k(\cdot, \cdot)$ and $Q$ are quantum, $\text{MMD}^2(P_{\theta} || Q)$ can be efficiently estimated by QGLMs in which the runtime cost is independent of the dimension of data space. In contrast, for GLMs, the runtime cost of calculating the MMD loss polynomially scales with the sample size $n$ and $m$. Such runtime discrepancy warrants the good performance of QGLMs explained in the subsequent context.

A. Generalization of QCBMs

A central topic in QCBMs is understanding whether quantum kernels can provide generalization advantages over classical ones. The following theorem provides a positive affirmation whose proof is postponed to SM D.
Theorem 1. Following the settings in Lemma 2 when the employed kernel \( k(\cdot, \cdot) \) can either be realized by quantum or classical machines, with probability at least \( 1 - \delta \), the generalization error of QCBSMs yields

\[
\mathbb{R}^Q \leq \mathbb{R}^C \leq C_1 \frac{8}{n} \frac{1}{m} \sqrt{C_2(2 + \sqrt{\log \frac{1}{\delta}})},
\]

where \( C_1 = \max_{\theta} \text{MMD}^2(\mathbb{P}_\theta \| \mathbb{Q}) \) and \( C_2 = \sup_x k(x, x) \).

It indicates that when \( Q \) is quantum, the generalization error of QCBSMs with the quantum kernel is strictly lower than its classical counterparts. Remarkably, most tasks in quantum many body physics and quantum information processing satisfy the requirement of directly accessing the target distribution \( Q \). In consequence, QCBSMs may achieve generalization advantages in these regimes. In addition, the upper bound in Eq. (8) underlies that the decisive factor to improve generalization of QCBSMs is simultaneously increasing \( n \) and \( m \). As such, for QCBSMs with classical kernels, there exists a tradeoff between the generalization and the runtime complexity, which is not the case for quantum kernels warranted by Lemma 1. Besides, the factor \( C_2 \) suggests that the choice of kernels also affects the generalization of QCBSMs. As with quantum discriminative learning models [70], the factor \( C_1 \) connects the expressivity of QCBBM with its generalization in the sense that Ansatz with an overwhelmed expressivity may degrade the generalization. All of these observations provide a practical guidance of designing QCBSMs.

Remark. In SM [52], we partially address another long standing problem in the quantum generative learning theory, i.e., whether QCBSMs are superior to classical GLMs with better performance. Concisely, we show that for certain \( Q \), QCBSMs can attain a lower \( \inf_{\theta \in \Theta} \text{MMD}^2(\mathbb{P}_\theta \| \mathbb{Q}) \) over a typical GLM–restricted Boltzmann machine [82], which may lead to a better generalization ability.

We conduct numerical simulations to examine the potential advantages of QCBSMs as claimed in Theorem 1. The first task is applying QCBSMs to estimate the discrete Gaussian distribution \( N(\mu, \sigma) \). In consequence, QCBSMs may achieve generalization advantages in these regimes. In addition, the upper bound in Eq. (8) underlies that the decisive factor to improve generalization of QCBSMs is simultaneously increasing \( n \) and \( m \). As such, for QCBSMs with classical kernels, there exists a tradeoff between the generalization and the runtime complexity, which is not the case for quantum kernels warranted by Lemma 1. Besides, the factor \( C_2 \) suggests that the choice of kernels also affects the generalization of QCBSMs. As with quantum discriminative learning models [70], the factor \( C_1 \) connects the expressivity of QCBBM with its generalization

\[
\text{KLDivergence} = 
\begin{array}{cccccc}
N = 8 & N = 12 & N = 4 & N = 6 & N = 8 & N = 10 \\
KLDivergence & 1.28 & 1.24 & 0.83 & 0.39 & 0.18 & 0.07 \\
1.21 & 0.83 & 0.59 & 0.18 & 0.07 & 0.07 & 0.07 \\
0.83 & 0.39 & 0.18 & 0.07 & 0.07 & 0.07 & 0.07 \\
0.59 & 0.18 & 0.07 & 0.07 & 0.07 & 0.07 & 0.07 \\
0.18 & 0.07 & 0.07 & 0.07 & 0.07 & 0.07 & 0.07 \\
0.07 & 0.07 & 0.07 & 0.07 & 0.07 & 0.07 & 0.07 \\
\end{array}
\]

FIG. 2: Simulation results of QCBSMs. (a) The implementation of QCBSMs when \( N = 6 \). The label ‘\( L_i \)’ refers to repeating the architecture highlighted by the brown color \( L_i \) times. The gate \( U(\theta_i) \) refers to the Ry gate applied on the \( i \)-th qubit in the \( l \)-th layer of Ansatz \( U(\theta_l) \). (b) The visualization of a two-qubit GHZ state. (c) The upper and lower panels separately show the simulation results of QCBSMs in the task of estimating the discrete Gaussian distribution when \( N = 8 \) and \( N = 12 \). The labels ‘\( Q^0 \), \( R^0 \), \( R^2 \), \( n \)’, stand for the target distribution, the output of QCBBM with the quantum kernel, and the output of QCBBM with the RBF kernel and \( n \) samples, respectively. The inner plots evaluate the statistical performance of QCBSM through KL divergence, where the x-axis labels the number of examples \( n \). (d) The four box-plots separately show the simulation results of QCBBM in the task of approximating \( N \)-qubit GHZ state with \( N = 4, 6, 8, 10 \). The y-axis refers to the fidelity. The x-axis refers to the applied kernels in QCBBM, where the label ‘\( Q \)’ represents the quantum kernel and the rest four labels refers to the RBF kernel with \( n \) samples.
setting, we repeat the training 5 times to get a better understanding of the robustness of the results.

The simulation results of QCBMs are illustrated in Fig. 2(c). The two outer plots exhibit the approximated distributions under different settings. In particular, for both $N = 8, 12$, the approximated distribution generated by QCBM with the quantum kernel well approximates $Q$. In the measure of KL divergence, the similarity of these two distributions is 0.18 and 1.6 for $N = 8, 12$, respectively. In contrast, when the adopted kernels are classical and the number of measurements is finite, QCBMs encounter the inferior performance. Namely, by increasing $n$ and $m$ from 50 to 1000, the KL divergence between the approximated distribution and the target distribution only decreases from 1.28 to 0.59 in the case of $N = 8$. Moreover, under the same setting, the KL divergence does not manifestly decrease when $N = 12$, which requires a larger $n$ and $m$ to attain a good approximation as suggested by Theorem 1. This argument is warranted by the numerical results with the setting $n = m \to \infty$, where the achieved KL divergence is comparable with QCBM with the quantum kernel. Nevertheless, the runtime complexity of QCBMs with the classical kernel polynomially scales with $n$ and $m$. According to Lemma 1 under this scenario, QCBMs with the quantum kernel embraces the runtime advantages.

We next follow Ref. [1] to apply QCBMs to accomplish the task of preparing GHZ states, a.k.a., “cat states” $|\Psi\rangle$. An intuition is depicted in Fig. 2(b). The choice of GHZ states is motivated by their importance in quantum information. The formal expression of an $N$-qubit GHZ state is $|\text{GHZ}\rangle = (|0\rangle^{\otimes N} + |1\rangle^{\otimes N})/\sqrt{2}$. The hyper-parameter settings are as follows. The number of qubits is set as $N = 4, 6, 8, 10$ and the corresponding depth is $L_1 = 4, 6, 8, 10$. The other settings are identical to those used in the task of discrete Gaussian approximation.

The simulation results, as illustrated in Fig. 2(d), indicate that QCBMs with quantum kernels outperform RBF kernels when $n$ and $m$ are finite. This observation becomes apparent with an increased $N$. For all settings of $N$, the averaged fidelity between the generated states of QCBMs with the quantum kernel and the target $|\text{GHZ}\rangle$ is above 0.99, whereas the obtained averaged fidelity for QCBMs with the RBF kernel is 0.46 for $N = 10$ and $n = m = 100$. Meanwhile, as with the prior task, RBF kernel attain a competitive performance with the quantum kernel unless $n = m \to \infty$, while the price to pay is an affordable computational overhead.

**B. Generalization of QGANs**

QGANs formulated in Eq. (2) are specified to be a class of learning protocols estimating the continuous distributions $P$ and thus only concern $R^C$ in Eq. (6). In this scenario, it is of paramount importance of unveiling how the generalization of QGANs depends on uploading methods and the structure information Ansatz. The following theorem makes a concrete step toward this goal, where the proof is deferred to SM E.

**Theorem 2.** Assume the kernel $k(\cdot, \cdot)$ is $C_3$-Lipschitz. Suppose that the employed quantum circuit $\hat{U}(z)$ to prepare $\rho_z$ containing in total $N_{ge}$ parameterized gates and each gate acting on at most $k$ qubits. Following notations in Eqs. (7) and (8), with probability at least $1 - \delta$, the generalization error of QGANs, $R^C$ in Eq. (2) is upper bounded by

$$
8 \sqrt{\frac{8C_2^2(n + m)}{nm}} \ln \frac{1}{\delta} + \frac{48}{n - 1} + \frac{144d^k \sqrt{N_{gt} + N_{ge}}}{n - 1}C_4,
$$

where $C_2 = \sup_x k(x, x)$, $C_4 = N \ln(1764C_3^2 n N_{ge} N_{gt}) + 1$.

The results of Theorem 2 describe how different components in QGANs effect the generalization bound and convey four-fold implications. First, according to the first term in the right-hand side of Eq. (C2), to achieve a tight upper bound of $R^C$, the ratio between the number of examples sampled from $Q$ and $P$ should satisfy $m/n = 1$. In this case, the generalization error linearly decreases and finally converges to zero with the increased $n$ or $m$. Second, $R^C$ linearly depends on the kernel term $C_2$, exponentially depends on $k$ in Eq. (3), and sublinearly depends on the number of trainable quantum gates $N_{gt}$. These observations underpin the importance of controlling the expressivity of the adopted Ansatz and selecting proper kernels to ensure both the good learning performance and generalization of QGANs. Third, the way of preparing $\rho_z$ is a determinant factor in generalization of QGANs. In other words, to improve the performance of QGANs, the prior distribution $P_Z$ and the number of encoding gates $N_{ge}$ should be carefully designed. Last, the explicit dependence on the architecture of Ansatz connects the generalization of QGANs with their trainability. Concretely, a large $N_{gt}$ or $k$ may induce barren plateaus in training QGLMs [73, 75] and results in an inferior learning performance. Meanwhile, it also leads to a degraded generalization error bound. This inherent connection queries a unified framework to simultaneously enhance the trainability and generalization of QGANs.

**Remark.** We emphasize that most kernels such as RBF, linear kernels, and Matérn kernels satisfy the Lipchitz.

$$
(9)
$$

$$
\text{Suppose that the employed quantum circuit } \hat{U}(z) \text{ to prepare } \rho_z \text{ containing in total } N_{ge} \text{ parameterized gates and each gate acting on at most } k \text{ qubits. Following notations in Eqs. (7) and (8), with probability at least } 1 - \delta, \text{ the generalization error of QGANs, } R^C \text{ in Eq. (2) is upper bounded by}
$$

$$
8 \sqrt{\frac{8C_2^2(n + m)}{nm}} \ln \frac{1}{\delta} + \frac{48}{n - 1} + \frac{144d^k \sqrt{N_{gt} + N_{ge}}}{n - 1}C_4,
$$

where $C_2 = \sup_x k(x, x)$, $C_4 = N \ln(1764C_3^2 n N_{ge} N_{gt}) + 1$.

The results of Theorem 2 describe how different components in QGANs effect the generalization bound and convey four-fold implications. First, according to the first term in the right-hand side of Eq. (C2), to achieve a tight upper bound of $R^C$, the ratio between the number of examples sampled from $Q$ and $P$ should satisfy $m/n = 1$. In this case, the generalization error linearly decreases and finally converges to zero with the increased $n$ or $m$. Second, $R^C$ linearly depends on the kernel term $C_2$, exponentially depends on $k$ in Eq. (3), and sublinearly depends on the number of trainable quantum gates $N_{gt}$. These observations underpin the importance of controlling the expressivity of the adopted Ansatz and selecting proper kernels to ensure both the good learning performance and generalization of QGANs. Third, the way of preparing $\rho_z$ is a determinant factor in generalization of QGANs. In other words, to improve the performance of QGANs, the prior distribution $P_Z$ and the number of encoding gates $N_{ge}$ should be carefully designed. Last, the explicit dependence on the architecture of Ansatz connects the generalization of QGANs with their trainability. Concretely, a large $N_{gt}$ or $k$ may induce barren plateaus in training QGLMs [73, 75] and results in an inferior learning performance. Meanwhile, it also leads to a degraded generalization error bound. This inherent connection queries a unified framework to simultaneously enhance the trainability and generalization of QGANs.

**Remark.** We emphasize that most kernels such as RBF, linear kernels, and Matérn kernels satisfy the Lipchitz.

$$
(9)
$$

$$
\text{Suppose that the employed quantum circuit } \hat{U}(z) \text{ to prepare } \rho_z \text{ containing in total } N_{ge} \text{ parameterized gates and each gate acting on at most } k \text{ qubits. Following notations in Eqs. (7) and (8), with probability at least } 1 - \delta, \text{ the generalization error of QGANs, } R^C \text{ in Eq. (2) is upper bounded by}
$$

$$
8 \sqrt{\frac{8C_2^2(n + m)}{nm}} \ln \frac{1}{\delta} + \frac{48}{n - 1} + \frac{144d^k \sqrt{N_{gt} + N_{ge}}}{n - 1}C_4,
$$

where $C_2 = \sup_x k(x, x)$, $C_4 = N \ln(1764C_3^2 n N_{ge} N_{gt}) + 1$.

The results of Theorem 2 describe how different components in QGANs effect the generalization bound and convey four-fold implications. First, according to the first term in the right-hand side of Eq. (C2), to achieve a tight upper bound of $R^C$, the ratio between the number of examples sampled from $Q$ and $P$ should satisfy $m/n = 1$. In this case, the generalization error linearly decreases and finally converges to zero with the increased $n$ or $m$. Second, $R^C$ linearly depends on the kernel term $C_2$, exponentially depends on $k$ in Eq. (3), and sublinearly depends on the number of trainable quantum gates $N_{gt}$. These observations underpin the importance of controlling the expressivity of the adopted Ansatz and selecting proper kernels to ensure both the good learning performance and generalization of QGANs. Third, the way of preparing $\rho_z$ is a determinant factor in generalization of QGANs. In other words, to improve the performance of QGANs, the prior distribution $P_Z$ and the number of encoding gates $N_{ge}$ should be carefully designed. Last, the explicit dependence on the architecture of Ansatz connects the generalization of QGANs with their trainability. Concretely, a large $N_{gt}$ or $k$ may induce barren plateaus in training QGLMs [73, 75] and results in an inferior learning performance. Meanwhile, it also leads to a degraded generalization error bound. This inherent connection queries a unified framework to simultaneously enhance the trainability and generalization of QGANs.

**Remark.** We emphasize that most kernels such as RBF, linear kernels, and Matérn kernels satisfy the Lipchitz.
construction details. The target distribution $Q$ is a 3D correlated Gaussian distribution centered at $\mu = (0, 0, 0)$ with covariance matrix $\sigma = \begin{pmatrix} 0.5 & 0.25 & 0.1 \\ 0.25 & 0.1 & 0.4 \\ 0.1 & 0.4 & 0.5 \end{pmatrix}$. The sampled examples from $Q$ are visualized in Fig. 3(c). The hyperparameter settings employed in the training procedure is as follows. The reference samples $n$ ranges from 200 to 10000 and we keep $n = m$. The layer depth of $G(\theta)$ is set as $L \in \{2, 4, 6, 8\}$. Each setting is repeated with 5 times to collect the statistical results.

The simulation results are exhibited in Figs. 3(b)-(d). The outer plot in Fig. 3(b) shows that for all settings of $m$, the empirical MMD loss, i.e., $\text{MMD}_U(P_{\theta(n,m)} \mid \mid Q^m)$, converges after 60 iterations, where the averaged loss is 0.0114, 0.0077, and 0.0054 for $m = 2, 10, 200$, respectively. The inner plot measures the expected MMD loss, i.e., the trained QGANs are employed to generate new 10000 examples and then evaluate $\text{MMD}_U$ to estimate MMD. The averaged expected MMD loss for $m = 2, 10, 200$ is 0.1178, 0.0122, and 0.0041 respectively. Since the generalization error of QGANs $\mathcal{E}_C$ is proportional to $|\text{MMD}_U - \text{MMD}|$, an important observation is that QGANs with a large $m$ can obtain a better generalization ability, which echoes with Theorem 2. For illustration, we depicts the generated distribution of the trained QGANs in Fig. 3(d). With increasing $m$, the learned distribution is close to the real distribution in Fig. 3(c). See SM [7] for more simulations.

### III. DISCUSSIONS

We conduct a comprehensive study to quantify the generalization ability of QGLMs, including QCBMs and QGANs, with MMD loss. Although the attained theoretical results do not exhibit the generic exponential advantages, we clearly show that under certain tasks and model settings, QCBMs and QGANs can surpass classical learning models. Moreover, we provide a succinct and direct way to compare generalization of QGLMs with different Ansätze. Extensive numerical simulations have been conducted to support our theoretical statements. These theoretical and empirical observations deepen our understanding about the capabilities of QGLMs and benefit the design of advanced QGLMs.

The developed techniques in this study are general and provide a novel approach to theoretically investigate the power of QGLMs. For instance, a promising direction is uncovering the generalization property of other QGLMs such as quantum auto-encoder, quantum Boltzmann machines, etc. Furthermore, our work mainly concentrates on QGLMs with MMD loss, whereas a promising research direction is to derive the generalization of QGLMs with other loss functions such as Sinkhorn divergence, Stein discrepancy $\mathcal{S}_7$, and Wasserstein distance $\mathcal{S}_8$.

For QCBMs, Theorem 1 unveils that quantum kernels can greatly benefit their generalization and reduce computational overhead over classical kernels when the target distribution is quantum. These results underline that an important future direction will be identifying how to use QGLMs to gain substantial quantum advantages for practical applications, e.g., quantum many body physics, quantum sensing, and quantum information processing.

For QGANs, Theorem 2 hints that their generalization error has the explicit dependence on the qudits count, the structure information of the employed Ansätze, the adopted encoding method, and the choice of prior distribution. These results enable us to theoretically understand the power of QGLMs and provide practical guidance to
devise novel Ansätze to enhance the learning performance of QGLMs. Besides, since Ansatz with too much expressivity may degrade the generalization ability of QGANs, it is necessary to integrate various advanced techniques such as quantum circuit architecture design techniques to boost QGLMs performance. From the theoretical perspective, the entangled relation between expressivity and generalization in QGLMs queries a deeper understanding from each side.

[1] Merlin C Wittrock. Generative processes of comprehension. Educational psychologist, 24(4):345–376, 1989.
[2] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. nature, 521(7553):436, 2015.
[3] Jürgen Schmidhuber. Deep learning in neural networks: An overview. Neural Networks, 61:85–117, 2015.
[4] Peter Dayan, Geoffrey E Hinton, Radford M Neal, and Richard S Zemel. The helmholtz machine. Neural computation, 7(5):889–904, 1995.
[5] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, 2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings, 2014.
[6] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In Advances in neural information processing systems, pages 2672–2680, 2014.
[7] Junbo Zhao, Michael Mathieu, and Yann LeCun. Energy-based generative adversarial networks. In 5th International Conference on Learning Representations, ICLR 2017, 2017.
[8] Ran Yi, Yong-Jin Liu, Yu-Kun Lai, and Paul L Rosin. APDrawingGAN: Generating artistic portrait drawings from face photos with hierarchical gans. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR ’19), pages 10743–10752, 2019.
[9] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4401–4410, 2019.
[10] Taesung Park, Ming-Yu Liu, Ting-Chun Wang, and Jun-Yan Zhu. Semantic image synthesis with spatially-adaptive normalization. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 2337–2346, 2019.
[11] Karim Armanious, Chenming Jiang, Marc Fischer, Thomas Küstner, Tobias Hepp, Konstantin Nikolaou, Sergios Gatidis, and Bin Yang. Medgan: Medical image translation using gans. Computerized Medical Imaging and Graphics, 79:101684, 2020.
[12] Łukasz Maziarka, Agnieszka Pocha, Jan Kaczmarczyk, Krzysztof Rataf, Tomasz Danel, and Michał Warchol. Mol-cyclegan: a generative model for molecular optimization. Journal of Cheminformatics, 12(1):1–18, 2020.
[13] Oscar Méndez-Lucio, Benoît Baillif, Djork-Arné Clevert, David Rouqué, and Joerg Wichard. De novo generation of hit-like molecules from gene expression signatures using artificial intelligence. Nature communications, 11(1):1–10, 2020.
[14] Shahnavaz Ahmed, Carlos Sánchez Muñoz, Franco Nori, and Anton Frisk Kockum. Quantum state tomography and Anton Frisk Kockum. Quantum state tomography of hit-like molecules from gene expression signatures using artificial intelligence. Nature communications, 11(1):1–10, 2020.
ervative adversarial learning. *Physical review letters*, 121(4):040502, 2018.

[31] John Preskill. Quantum computing in the nisq era and Beyond. *Quantum*, 2:7, 2019.

[32] Frank Arute, Kunal Arya, Ryan Babbush, Dave Bacon, Joseph C Bardin, Rami Barends, Rupak Biswas, Sergio Boixo, Fernando G A Brandsen, David B Blatt, et al. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

[33] Yuling Wu, Wu-San Bao, Siuru Cao, Min-Hsiu Hsieh, Tongliang Liu, Min-Hsiu Hsieh, and Dacheng Tao. Quantum advantage in learning from experiments. *arXiv preprint arXiv:2112.00778*, 2021.

[34] Hsin-Yuan Huang, Michael Broughton, Jordan Cotler, Sitan Chen, Jerry Li, Masoud Mohseni, Hartmut Neven, Ryan Babbush, Richard Kueng, John Preskill, et al. Quantum advantage in learning from experiments. *arXiv preprint arXiv:2106.15432*, 2021.

[35] Hsin-Yuan Huang, Michael Broughton, Masoud Mohseni, Ryan Babbush, Sergio Boixo, Hartmut Neven, and Jarrod R McClean. Power of data in quantum machine learning. *Nature communications*, 12(1):1–9, 2021.

[36] Xinhao Wang, Yuxuan Du, Yong Luo, and Dacheng Tao. Towards understanding the power of quantum kernels in the NISQ era. *Quantum*, 5:531, August 2021.

[37] Yuxuan Du and Dacheng Tao. On exploring practical potentials of quantum auto-encoder with advantages. *arXiv preprint arXiv:2106.15432*, 2021.

[38] Marco Cerezo, Andrew Arrasmith, Ryan Babbush, Simon C Benjamin, Suguru Endo, Keisuke Fujii, Jarrod R McClean, Kosuke Mitarai, Xiao Yuan, Lukasz Cincio, et al. Variational quantum algorithms. *Nature Reviews Physics*, 3(9):625–644, 2021.

[39] Kishor Bharti, Alba Cervera-Lierta, Thi Ha Kyaw, Tobias Haug, Sumner Alperin-Lea, Abhinav Anand, Matthias Degroote, Hermann Heimonen, Jakob S Kottmann, Tim Menke, et al. Noisy intermediate-scale quantum algorithms. *Reviews of Modern Physics*, 94(1):015004, 2022.

[40] Marcello Benedetti, Erika Lloyd, Stefan Sank, and Mattia Fiorentini. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001, 2019.

[41] Marcello Benedetti, Delfina Garcia-Pintos, Oscar Perdomo, Vicente Leyton-Ortega, Yunseong Nam, and Alejandro Perdomo-Ortiz. A generative modeling approach for benchmarking and training shallow quantum circuits. *npj Quantum Information*, 5(1):1–9, 2019.

[42] Jing-Guo Liu and Lei Wang. Differentiable learning of quantum circuit born machines. *Physical Review A*, 98(6):062324, 2018.

[43] Brian Coyle, Maxwell Henderson, Justin Chan Jin Le, Niraj Kumar, Marco Paini, and Elham Kashefi. Quantum versus classical generative modelling in finance. *Quantum Science and Technology*, 6(2):024013, 2021.

[44] Yuxuan Du, Min-Hsiu Hsieh, Tongliang Liu, and Dacheng Tao. Expressive power of parameterized quantum circuits. *Phys. Rev. Research*, 2:033125, Jul 2020.

[45] Amir Khoshaman, Walter Vinci, Brandon Denis, Evgeny Andriyash, Hossein Sadeghi, and Mohammad H Amin. Quantum variational autoencoder. *Quantum Science and Technology*, 4(1):014001, 2018.

[46] Yuta Shingu, Yuuya Seki, Shohei Watabe, Suguru Endo, Yuichiro Matsuzaki, Shiro Kawabata, Tetsuro Nikuni, and Hideaki Hakoshima. Boltzmann machine learning with a variational quantum algorithm. *Physical Review A*, 104(3):032413, 2021.

[47] Chang Yu Hsieh, Qiming Sun, Shengyu Zhang, and Chee Kong Lee. Unitary-coupled restricted boltzmann machine ansatz for quantum simulations. *npj Quantum Information*, 7(1):1–10, 2021.

[48] Christa Zoufal, Aurélien Lucchi, and Stefan Woerner. Quantum generative adversarial networks for learning and loading random distributions. *npj Quantum Information*, 5(1):1–9, 2019.

[49] Jinfeng Zeng, Yufeng Wu, Jin-Guo Liu, Lei Wang, and Jiaping Hu. Learning and inference on generative adversarial quantum circuits. *Physical Review A*, 99(5):052306, 2019.

[50] He-Liang Huang, Yuxuan Du, Ming Gong, Youwei Zhao, Yulin Wu, Chaoyue Wang, Shaowei Li, Futian Liang, Jin Lin, Yu Xu, et al. Experimental quantum generative adversarial networks for image generation. *Physical Review Applied*, 16(2):024051, 2021.

[51] Yiming Huang, Hang Lei, Xiaoyu Li, and Guowu Yang. Quantum maximum mean discrepancy gan. *Neurocomputing*, 454:88–100, 2021.

[52] Jonathan Romero and Alán Aspuru-Guzik. Variational quantum generators: Generative adversarial quantum machine learning for continuous distributions. *Advanced Quantum Technologies*, 4(1):2000003, 2021.

[53] Kaitlin Gili, Marta Mauri, and Alejandro Perdomo-Ortiz. Evaluating generalization in classical and quantum generative models. *arXiv preprint arXiv:2201.08770*, 2022.

[54] Kouhei Nakaji and Naoki Yamamoto. Quantum semi-supervised generative adversarial network for enhanced data classification. *Scientific reports*, 11(1):1–10, 2021.

[55] Paolo Braccia, Leonardo Banchi, and Filippo Caruso. Quantum noise sensing by generating fake noise. *Physical Review Applied*, 17(2):024002, 2022.

[56] Abhinav Anand, Jonathan Romero, Matthias Degroote, and Alán Aspuru-Guzik. Noise robustness and experimental demonstration of a quantum generative adversarial network for continuous distributions. *Advanced Quantum Technologies*, 4(5):2000069, 2021.

[57] Xu-Fei Yin, Yuxuan Du, Yue-Yang Fei, Rui Zhang, Li-Zheng Liu, Yingqiu Mao, Tongliang Liu, Min-Hsiu Hsieh, Li Li, Nai-Le Liu, Dacheng Tao, Yu-Ao Chen, and Jian-Wei Pan. Efficient Bipartite Entanglement Detection Scheme with a Quantum Adversarial Solver. *Phys. Rev. Lett.*, 128, 110501 (2022), 2022. arXiv:2203.07749v1.

[58] Implicit probabilistic models do not specify the distribution of the data itself, but rather define a stochastic process that, after training, aims to draw samples from the underlying data distribution.

[59] Daiwei Zhu, Norbert M Linke, Marcello Benedetti, Kevin A Landsman, Nhung H Nguyen, C Huerta Alderete, Alejandro Perdomo-Ortiz, Nathan Korda, A Garfoot, Charles Brecque, et al. Training of quantum circuits on a hybrid quantum computer. *Science advances*, 5(10):eaaw9918, 2019.

[60] Ling Hu, Shu-Hao Wu, Weizhou Cai, Yuwei Ma, Xiang-hao Mu, Yuan Xu, Haiyang Wang, Yipu Song, Dong-Ling Deng, Chang-Ling Zou, et al. Quantum generative adversarial learning in a superconducting quantum circuit. *Science advances*, 5(1):eaav2761, 2019.

[61] Junde Li, Rasit O Topaloglu, and Swaroop Ghosh. Quan-
tum generative models for small molecule drug discovery. *IEEE Transactions on Quantum Engineering*, 2:1–8, 2021.

[62] Yu-Xin Jin, Jun-Jie Hu, Qi Li, Zhi-Cheng Luo, Fang-Yan Zhang, Hao Tang, Kun Qian, and Xian-Min Jin. Quantum Deep Learning for Mutant COVID-19 Strain Prediction, 2022. arXiv:2203.03556v1.

[63] Zeyuan Allen-Zhu, Yuanzhi Li, and Yingyu Liang. Learning and generalization in overparameterized neural networks, going beyond two layers. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, pages 6158–6169, 2019.

[64] Yuxuan Du, Min-Hsiu Hsieh, Tongliang Liu, Shan You, and Dacheng Tao. Learnability of quantum neural networks. *PRX Quantum*, 2(4):040337, 2021.

[65] Sanjeev Arora, Rong Ge, Yingyu Liang, Tengyu Ma, and Yi Zhang. Generalization and equilibrium in generative adversarial nets (gans). arXiv preprint arXiv:1703.00573, 2017.

[66] Shengjia Zhao, Hongyu Ren, Arianna Yuan, Jiaming Song, Noah Goodman, and Stefano Ermon. Bias and generalization in deep generative models: An empirical study. *Advances in Neural Information Processing Systems*, 31, 2018.

[67] Amira Abbas, David Sutter, Christa Zoufal, Aurélien Lucchi, Alessio Figalli, and Stefan Woerner. The power of quantum neural networks. Nature *Computational Science*, 1(6):403–409, 2021.

[68] Leonardo Banchi, Jason Pereira, and Stefano Pirandola. Generalization in quantum machine learning: A quantum information standpoint. *PRX Quantum*, 2(4):040321, 2021.

[69] Matthias C Caro, Hsin-Yuan Huang, M Cerezo, Kunal Sharma, Andrew Sornborger, Lukasz Cincio, and Patrick J Coles. Generalization in quantum machine learning from few training data. arXiv preprint arXiv:2111.05922, 2021.

[70] Yuxuan Du, Zhuozhuo Tu, Xiao Yuan, and Dacheng Tao. Efficient measure for the expressivity of variational quantum algorithms. *Physical Review Letters*, 128(8):080506, 2022.

[71] Hsin-Yuan Huang, Richard Kueng, and John Preskill. Information-theoretic bounds on quantum advantage in machine learning. *Physical Review Letters*, 126(19):190505, 2021.

[72] Junyu Liu, Khadjeh Najafi, Kunal Sharma, Francesco Tacchino, Liang Jiang, and Antonio Mezzacapo. An analytic theory for the dynamics of wide quantum neural networks, 2022. arXiv:2203.10671v1.

[73] Yang Qian, Xinbiao Wang, Yuxuan Du, Xingyao Wu, and Dacheng Tao. The dilemma of quantum neural networks. arXiv preprint arXiv:2106.04975, 2021.

[74] Maria Kieferova, Ortiz Marrero Carlos, and Nathan Wiebe. Quantum generative training using $r$\setminus enyi divergences. arXiv preprint arXiv:2106.09567, 2021.

[75] Jarrod R McClean, Sergio Boixo, Vadim N Smelyanskiy, Ryan Babbush, and Hartmut Neven. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9(1):1–6, 2018.

[76] Kuining Zhang, Min-Hsiu Hsieh, Liu Liu, and Dacheng Tao. Toward trainability of deep quantum neural networks. arXiv preprint arXiv:2112.15002, 2021.

[77] Xin Gao, Eric R Anschuetz, Sheng-Tao Wang, J Ignacio Cirac, and Mikhail D Lukin. Enhancing generative models via quantum correlations. arXiv preprint arXiv:2101.08334, 2021.

[78] Arthur Gretton, Karsten M. Borgwardt, Malte J. Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *J. Mach. Learn. Res.*, 13(null):723–773, mar 2012.

[79] Vladimir Vapnik. *The nature of statistical learning theory*. Springer science & business media, 1999.

[80] For simplicity, the definition of N-qubit Ansatz $\hat{U}(\theta)$ in Eq. (3) omits some identity operators. The complete description for the l-th layer is $\mathbb{I}_{N-k} \otimes \hat{U}(\theta)$.

[81] Gorka Dziugaite, DM Roy, and Z Ghahramani. Training generative neural networks via maximum mean discrepancy optimization. In *Uncertainty in Artificial Intelligence-Proceedings of the 31st Conference, UAI 2015*, pages 258–267, 2015.

[82] $k(\cdot, \cdot)$ is quantum means that the specified kernel can be efficiently computed by quantum algorithms. Representative examples are linear kernel and polynomial kernels.

[83] Geoffrey E Hinton. A practical guide to training restricted boltzmann machines. In *Neural networks: Tricks of the trade*, pages 599–619. Springer, 2012.

[84] Michael A Nielsen and Isaac L Chuang. *Quantum computation and quantum information*. Cambridge University Press, 2010.

[85] Armin Lederer, Jonas Umlauf, and Sandra Hirche. Uniform Error Bounds for Gaussian Process Regression with Application to Safe Control. Curran Associates Inc., Red Hook, NY, USA, 2019.

[86] Carlos Bravo-Prieto, Julien Baglio, Marco Cè, Anthony Francis, Dorota M Grabowska, and Stefano Carrazza. Style-based quantum generative adversarial networks for monte carlo events. arXiv preprint arXiv:2110.09933, 2021.

[87] Francois-Xavier Briol, Alessandro Barp, Andrew B Duncan, and Mark Girolami. Statistical inference for generative models with maximum mean discrepancy. arXiv preprint arXiv:1906.05944, 2019.

[88] Shouvanik Chakrabarti, Huang Yiming, Tongyang Li, Soheil Feizi, and Xiaodi Wu. Quantum wasserstein generative adversarial networks. *Advances in Neural Information Processing Systems*, 32, 2019.

[89] David Amaro, Carlo Modica, Matthias Rosenkranz, Mattia Fiorentini, Marcello Benedetti, and Michael Lubasch. Filtering variational quantum algorithms for combinatorial optimization. *Quantum Science and Technology*, 7(1):015021, 2022.

[90] M Bilik, M Cerezo, Guillaume Verdon, Patrick J Coles, and Lukasz Cincio. A semi-agnostic ansatz with variable structure for quantum machine learning. arXiv preprint arXiv:2103.06712, 2021.

[91] Yuxuan Du, Tao Huang, Shan You, Min-Hsiu Hsieh, and Dacheng Tao. Quantum circuit architecture search: error mitigation and trainability enhancement for variational quantum solvers. arXiv preprint arXiv:2010.10217, 2020.

[92] Kejun Linghu, Yang Qian, Ruixia Wang, Meng-Jun Hu, Zhiyuan Li, Xuegang Li, Huikai Xu, Jingning Zhang, Teng Ma, Peng Zhao, et al. Quantum circuit architecture search on a superconducting processor. arXiv preprint arXiv:2201.00933, 2022.

[93] En-Jui Kuo, Yao-Lung L Fang, and Samuel Yen-Chi Chen. Quantum architecture search via deep reinforce-
Differentiable quantum architecture search. arXiv preprint arXiv:2010.08561, 2020.

Maria Schuld and NathanKilloran. Quantum machine learning in feature hilbert spaces. Physical review letters, 122(4):040504, 2019.

Marcello Benedetti, Edward Grant, Leonard Wossnig, and Simone Severini. Adversarial quantum circuit learning for pure state approximation. New Journal of Physics, 21(4):043023, 2019.

Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In International Conference on Machine Learning, pages 214–223, 2017.

Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. arXiv preprint arXiv:1411.1784, 2014.

Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In Proceedings of the IEEE International Conference on Computer Vision, pages 5907–5915, 2017.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. Deep learning. MIT press, 2016.

Stephen Boyd, Stephen P. Boyd, and Lieven Vandenberghe. Convex optimization. Cambridge university press, 2004.

Maria Schuld, Ville Bergholm, Christian Gogolin, Josh Izaac, and Nathan Killoran. Evaluating analytic gradients on quantum hardware. Physical Review A, 99(3):032331, 2019.

Harry Buhrman, Richard Cleve, John Watrous, and Ronald De Wolf. Quantum fingerprinting. Physical Review Letters, 87(16):167902, 2001.

Hirotada Kobayashi, Keiji Matsumoto, and Tomoyuki Yamakami. Quantum merlin-arthur proof systems: Are multiple merlins more helpful to arthur? In International Symposium on Algorithms and Computation, pages 189–198. Springer, 2003.

Jacob Biamonte. Universal variational quantum computation. Physical Review A, 103(3):L030401, 2021.

Nicolas Le Roux and Yoshua Bengio. Representational power of restricted boltzmann machines and deep belief networks. Neural computation, 20(6):1631–1649, 2008.

Xun Gao and Lu-Ming Duan. Efficient representation of quantum many-body states with deep neural networks. Nature communications, 8(1):662, 2017.

Shahar Mendelson. A few notes on statistical learning theory. In Advanced lectures on machine learning, pages 1–40. Springer, 2003.

Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. Foundations of Machine Learning. Adaptive Computation and Machine Learning. MIT Press, Cambridge, MA, 2 edition, 2018.

Richard M Dudley. The sizes of compact subsets of hilbert space and continuity of gaussian processes. Journal of Functional Analysis, 1(3):290–330, 1967.

Thomas Barthel and Jianfeng Lu. Fundamental limitations for measurements in quantum many-body systems. Phys. Rev. Lett., 121:080406, Aug 2018.

Sergio Boixo, Sergei V Isakov, Vadim N Smelyanskiy, Ryan Babbush, Nan Ding, Zhang Jiang, Michael J Bremner, John M Martinis, and Hartmut Neven. Characterizing quantum supremacy in near-term devices. Nature Physics, 14(6):595, 2018.

Qingling Zhu, Sirui Cao, Fusiheng Chen, Ming-Cheng Chen, Xueiwei Chen, Tung-Hsun Chung, Hui Deng, Yajie Du, Daojin Fan, Ming Gong, Cheng Guo, Chu Guo, Shaojun Guo, Lianchan Han, Linyin Hong, He-Liang Huang, Yong-Heng Huo, Liping Li, Na Li, Shaowei Li, Yuan Li, Futian Liang, Chun Lin, Jin Lin, Haoran Qian, Dan Qiao, Hao Rong, Hong Su, Lihua Sun, Liangyuan Wang, Shiyu Wang, Dachao Wu, Yulin Wu, Yu Xu, Kai Yan, Weifeng Yang, Yang Yang, Yangsen Ye, Jianshan Yin, Chong Ying, Jiale Yu, Chen Zha, Cha Zhang, Habin Zhang, Kaiili Zhang, Yiming Zhang, Han Zhao, Youwei Zhao, Liang Zhou, Chao-Yang Lu, Cheng-Zhi Peng, Xiaobo Zhu, and Jian-Wei Pan. Quantum computational advantage via 60-qubit 24-cycle random circuit sampling, 2021.

Scott Aaronson and Lijie Chen. Complexity-theoretic foundations of quantum supremacy experiments. In 32nd Computational Complexity Conference, page 1, 2017.

Julia Kempe, Alexei Kitaev, and Oded Regev. The complexity of the local hamiltonian problem. SIAM Journal on Computing, 35(5):1070–1097, 2006.
Supplementary Material: “Theory of Quantum Generative Learning Models with Maximum Mean Discrepancy”

SM A: Schematic of QGANs in the discrete and continuous settings

For the purpose of elucidating, in this section, we first demonstrate the equivalence of QCBMs and QGANs in the discrete setting when the loss function is specified to be MMD, and then introduce the basic theory of GANs and QGANs, especially for QGANs with MMD loss.

**Equivalence between QCBMs and QGANs when \( Q \) is discrete.** In accordance with the explanations in Refs. [48, 96], when QGAN is applied to estimate a discrete distribution \( Q \) (e.g., quantum state approximation), the quantum generator aims to directly capture the distribution of the data itself. This violates the criteria of implicit generative models, where a stochastic process is employed to draw samples from the underlying data distribution after training. More specifically, when \( Q \) is discrete, the output of the quantum circuit for both QCBM and QGAN takes the form \( P_{\theta}(i) = \text{Tr}(U(\theta)\rho_i U(\theta)^\dagger) \) in Eq. (1). The concept ‘adversarial’ originates from the way of optimizing \( \theta \). Instead of using a deterministic distance measure (e.g., KL divergence) as in QCBMs, QGANs utilize a discriminator \( D_\gamma \), implemented by either trainable parameterized quantum circuit or a neural network, to maximally separate \( \mathbb{P}_Q \) from \( \mathbb{P}_G \). The behavior of simultaneously update \( \theta \) (to minimize the loss) and \( \gamma \) (to maximize the loss) is termed as quantum generative adversarial learning. With this regard, when we replace the trainable \( D_\gamma \) by the deterministic measure MMD, QGAN takes an equivalent mathematical form with QCBM.

**Basic theory of (classical) GANs and QGANs when \( Q \) is continuous.** The fundamental mechanism of GAN [6] and its variations [87, 100] is as follows. GAN sets up a two-players game: the generator \( G \) creates data that pretends to come from the real data distribution \( Q \) to fool the discriminator \( D \), while \( D \) tries to distinguish the fake generated data from the real training data. Mathematically, \( G \) and \( D \) corresponds to two a differentiable functions. In particular, the input of \( G \) is a latent variable \( z \) and its output is \( x \), i.e., \( G : [z, \theta] \rightarrow x \) with \( \theta \) being trainable parameters for \( G \). The role of the latent variable \( z \) is ensuring GAN to be a structured probabilistic model [101]. The input of \( D \) can either be the generated data \( x \) or the real data \( y \sim Q \) and its output corresponds to the binary classification result (real or fake), respectively. The mathematical expression of \( D \) yields \( D : (x, y, \gamma) \rightarrow (0, 1) \) with \( \gamma \) being trainable parameters for \( D \). If the distribution learned by \( G \) equals to the real data distribution, i.e., \( \mathbb{P}_G = \mathbb{P}_Q \), then \( D \) can never discriminate between the generated data and the real data and this unique solution is called Nash equilibrium [6]. To reach the Nash equilibrium, the training process of GANs corresponds to the minimax optimization. Namely, the discriminator \( D \) updates \( \gamma \) to maximize the classification accuracy, while the generator \( G \) updates \( \theta \) to minimize the classification accuracy. With this regard, the optimization of GAN follows

\[
\min_{\theta} \max_{\gamma} \mathcal{L}(D_\gamma(G_\theta(z)), D_\gamma(x)) := \mathbb{E}_{x \sim Q}[D_\gamma(x)] + \mathbb{E}_{z \sim P_Z}[(1 - D_\gamma(G_\theta(z)))] ,
\]

where \( Q \) is the distribution of training dataset, and \( P_Z \) is the probability distribution of the latent variable \( z \). In general, \( G_\theta \) and \( D_\gamma \) are constructed by deep neural networks, and their parameters are updated iteratively using gradient descent methods [102].

The key difference between GANs and QGANs is the way of implementing \( G_\theta \) and \( D_\gamma \). Particularly, in QGANs, either \( G \), \( D \), or both can be realized by variational quantum circuits instead of deep neural networks. The training strategy of QGANs is similar to classical GANs. In this study, we focus on QGANs with MMD loss, which can be treated as the quantum extension of MMD-GAN [51]. Unlike conventional GANs and QGANs, MMD-GAN and QGAN replace a trainable discriminator with MMD. In this way, the family of discriminators is substituted with a family \( \mathcal{H} \) of test functions, closed under negation, where the optimization of \( D \) can be completed with the analytical form. Therefore, the goal of QGANs is finding an estimator minimizing an unbiased MMD loss, i.e.,

\[
\text{MMD}^2_U(\mathbb{P}_G \| Q) := \frac{1}{n(n-1)} \sum_{i \neq i'} k(x^{(i)}, x^{(i')}) + \frac{1}{m(m-1)} \sum_{j \neq j'} k(y^{(j)}, y^{(j')}) - \frac{2}{nm} \sum_{i,j} k(x^{(i)}, y^{(j)})
\]

where \( x^{(i)} \sim \mathbb{P}_G \) and \( y^{(j)} \sim Q \).

SM B: Optimization of QGANs with MMD loss

For self-consistency, in this section, we introduce the elementary backgrounds of the optimization of QGLMs with MMD loss. See Ref. [78] for elaborations.
MMD loss. Let $k: \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ be a Borel measurable kernel on $\mathcal{X}$, and consider the reproducing kernel Hilbert space $\mathcal{H}_k$ associated with $k$ (see Berlìnét and Thomas-Agnan [2004]), equipped with inner product $\langle \cdot, \cdot \rangle_{\mathcal{H}_k}$. Let $\mathcal{P}_k(\mu)$ be the set of Borel probability measures $\mu$ such that $\int_X k(x, x') \mu(dx') < \infty$. The kernel mean embedding $\Pi_k(\mu) = \int k(\cdot, y) \mu(dy)$, interpreted as a Bochner integral, defines a continuous embedding from $\mathcal{P}_k(\mathcal{X})$ into $\mathcal{H}_k$. The mean embedding pulls-back the metric on $\mathcal{H}_k$ generated by the inner product to define a pseudo-metric on $\mathcal{P}_k(\mathcal{X})$ called the maximum mean discrepancy MMD: $\mathcal{P}_k(\mathcal{X}) \times \mathcal{P}_k(\mathcal{X}) \to \mathbb{R}_+$, i.e.,

$$\text{MMD}(P_1 \| P_2) = \| \Pi_k(P_1) - \Pi_k(P_2) \|_{\mathcal{H}_k}.$$  

(B1)

The MMD loss has a particularly simple expression that can be derived through an application of the reproducing property $(f(x) = (f, k(\cdot, x))_{\mathcal{H}_k})$, i.e.,

$$\text{MMD}^2(P_1 \| P_2) := \left\| \int_X k(\cdot, x) P_1(dx) - \int_X k(\cdot, x) P_2(dx) \right\|_{\mathcal{H}_k}^2 = \int_X \int_X k(x, y) P_1(dx) P_2(dy) - 2 \int_X k(x, y) P_1(dx) P_2(dy) + \int_X \int_X k(x, y) P_2(dx) P_2(dy) = \mathbb{E}_{x, y \sim P_1}(k(x, y)) - 2 \mathbb{E}_{x \sim P_1, y \sim P_2}(k(x, y)) + \mathbb{E}_{x, y \sim P_2}(k(x, y)),$$

(B2)

which provides a closed form expression up to calculation of expectations.

Optimization of QCBMs with MMD loss. The goal of QCBMs is finding an estimator minimizing the loss function $\text{MMD}^2(P_{\theta} \| Q)$ in Eq. (B2), where $P_{\theta}$ is defined in Eq. (1). The optimization is completed by the gradient based descent optimizer. The updating rule satisfies $\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_\theta \text{MMD}^2(P_{\theta} \| Q)$ and $\eta$ is the learning rate. Concretely, the partial derivative of the $j$-th entry satisfies

$$\frac{\partial}{\partial \theta_j} \text{MMD}^2(P_{\theta} \| Q) = \partial \mathbb{E}_{x, y \sim P_{\theta}}(k(x, y)) - 2 \mathbb{E}_{x \sim P_{\theta}, y \sim Q}(k(x, y)) + \mathbb{E}_{x, y \sim Q}(k(x, y))$$

$$= \sum_{x, y} k(x, y) \left( \mathbb{E}_{\theta}(y) \frac{\partial \mathbb{E}_{\theta}(x)}{\partial \theta_j} + \mathbb{E}_{\theta}(x) \frac{\partial \mathbb{E}_{\theta}(y)}{\partial \theta_j} \right) - 2 \sum_{x, y} k(x, y) \frac{\partial \mathbb{E}_{\theta}(x)}{\partial \theta_j} \mathbb{E}_{\theta}(y)$$

$$= \sum_{x, y} k(x, y) \left( \mathbb{E}_{\theta}(y + \varepsilon_j(x)) - \mathbb{E}_{\theta - \varepsilon_j(x)}(y) \right) + \mathbb{E}_{\theta}(x) \left( \mathbb{E}_{\theta + \varepsilon_j(y)} - \mathbb{E}_{\theta - \varepsilon_j(y)} \right)$$

$$- 2 \sum_{x, y} k(x, y) \left( \mathbb{E}_{\theta + \varepsilon_j(x)} - \mathbb{E}_{\theta - \varepsilon_j(x)} \right)$$

$$= \mathbb{E}_{x \sim P_{\theta} + \varepsilon_j}(k(x, y)) - \mathbb{E}_{x \sim P_{\theta} - \varepsilon_j}(k(x, y)) + \mathbb{E}_{x \sim P_{\theta}, y \sim Q}(k(x, y)) - \mathbb{E}_{x \sim P_{\theta}, y \sim Q}(k(x, y))$$

$$- 2 \mathbb{E}_{x \sim P_{\theta} + \varepsilon_j, y \sim Q}(k(x, y)) + 2 \mathbb{E}_{x \sim P_{\theta} - \varepsilon_j, y \sim Q}(k(x, y)),$$

(B3)

where the last second equality employs the parameter shift rule [103] to calculate the partial derivative $\partial \mathbb{E}_{\theta}(y)/\partial \theta_j$ and $\partial \mathbb{E}_{\theta}(x)/\partial \theta_j$. According to Lemma 1, the six expectation terms in Eq. (B3) can be analytically and efficiently calculated when $k(\cdot, \cdot)$ is quantum. In the case of classical kernels, the six expectation terms in Eq. (B3) are estimated by the sample mean.

Optimization of QGANs with MMD loss. We next derive the gradients of QGANs with respect to the $t$-th entry. Since the evaluation of expectation is runtime expensive when $Q$ is continuous, QGANs employ an unbiased estimator of the MMD loss in Eq. (A2) to update $\theta$. The updating rule at the $t$-th iteration is $\theta^{(t+1)} = \theta^{(t)} - \eta \nabla_\theta \text{MMD}^2(P_{\theta} \| Q)$. According to the chain rule, we have

$$\frac{\partial}{\partial \theta_t} \text{MMD}^2(P_{\theta} \| Q) = \frac{1}{n(n-1)} \sum_{i \neq j} \frac{\partial}{\partial \theta_t} k(G_{\theta}(z^{(i)}), G_{\theta}(z^{(j)})) - \frac{2}{nm} \sum_{i, j} \frac{\partial}{\partial \theta_t} k(G_{\theta}(z^{(i)}), y^{(j)})$$

$$= \frac{1}{n(n-1)} \sum_{i \neq j} \frac{\partial}{\partial \theta_t} k(G_{\theta}(z^{(i)}), G_{\theta}(z^{(j)})) \frac{\partial G_{\theta}(z^{(i)})}{\partial \theta_t} + \frac{\partial k(G_{\theta}(z^{(i)}), G_{\theta}(z^{(j)}))}{\partial \theta_t} \frac{\partial G_{\theta}(z^{(i)})}{\partial \theta_t} - \frac{2}{nm} \sum_{i,j} \frac{\partial}{\partial \theta_t} k(G_{\theta}(z^{(i)}), y^{(j)}) \frac{\partial G_{\theta}(z^{(i)})}{\partial \theta_t},$$

(B4)

$$= \frac{2}{nm} \sum_{i, j} \frac{\partial}{\partial \theta_t} k(G_{\theta}(z^{(i)}), y^{(j)}) \frac{\partial G_{\theta}(z^{(i)})}{\partial \theta_t}.$$  

(B5)
where the first equality uses \( \frac{1}{m(m-1)} \sum_{j \neq i}^m k(y^{(j)}, y^{(j)})/\partial \theta_i = 0 \), each derivative \( \partial k(G_\theta(z^{(i)}), G_\theta(z^{(i)}))/\partial G_\theta(z^{(i)}) \) can be easily computed for standard kernels, and the derivative \( \partial G_\theta(z^{(i)})/\partial \theta_i \) for \( \forall i \in n, j \in [m] \) can be computed via the parameter shift rule. Therefore, the gradients of QGANs with MMD loss can be achieved.

**SM C: Proof of Lemma [1]**

We first introduce the definition of kernels.

**Definition 1** (Definition 2, [95]). Let \( \mathcal{X} \) be a nonempty set, called the input set. A function \( k : \mathcal{X} \times \mathcal{X} \to \mathbb{C} \) is called kernel if the Gram matrix \( K \) with entries \( K_{m,m'} = k(x^m, x^{m'}) \) is positive semi-definite.

**Proof of Lemma [1]** Here we separately elaborate on the calculation of MMD loss when the target distribution \( \mathbb{Q} \) can be efficiently prepared by a diagonalized mixed state \( \sigma = \sum_y \sqrt{Q(y)} |y \rangle \langle y | \) or a pure quantum state \( |\Psi\rangle = \sum_y \sqrt{Q(y)} |y \rangle |y \rangle \).

**Diagonalized mixed states.** In this setting, the quantum kernel corresponds to the linear kernel, i.e., \( k(a, a') = \langle a, a' \rangle \).

Recall the definition of the MMD loss is \( \text{MMD}^2(\mathbb{P}_\theta || \mathbb{Q}) = E(k(x, x')) - 2 E(k(x, y)) + E(k(y, y')) \). The first term equals to

\[
E(k(x, x')) = E_{\mathbb{P}_\theta, \mathbb{P}_\theta} (\langle x, x' \rangle) = \sum_x \mathbb{P}_\theta^2(x). \tag{C1}
\]

Similarly, the second term equals to

\[
E(k(x, y)) = \sum_x \mathbb{P}_\theta(x) Q(x). \tag{C2}
\]

And the third term equals to

\[
E(k(y, y')) = \sum_y Q^2(y). \tag{C3}
\]

The above three terms can be efficiently and analytically evaluated by quantum Swap test when the input state of QCBM in Eq. (1) is a full rank mixed state, e.g., \( \rho_0 = \frac{I_{2^n}}{2^n} \). Denote the output state of QCBM as \( \rho = U(\theta) \rho_0 U(\theta)^\dagger \). This state is also diagonalized and its diagonalized entry records \( \mathbb{P}_\theta \), i.e.,

\[
\rho = \sum_x \mathbb{P}_\theta(x) |x \rangle \langle x |. \tag{C4}
\]

According to [105], given two mixed states \( \rho_1 \) and \( \rho_2 \), the output of Swap test is \( 1/2 + \text{Tr}(\rho_1 \rho_2)/2 \) with an additive error \( \epsilon \) in \( O(1/\epsilon^2) \) runtime. As such, when \( \rho_1 = \rho_2 = \rho \), the first term \( E(k(x, x')) \) can be calculated by Swap test, because \( \text{Tr}(\rho^2) = \sum_x \mathbb{P}_\theta^2(x) \). Likewise, through setting \( \rho_1 = \rho, \) and \( \rho_2 = \sigma \) \( (\rho_1 = \rho_2 = \sigma) \), the second (third) term can be efficiently evaluated by Swap test with an additive error \( \epsilon \). In other words, by leveraging Swap test, we can estimate MMD loss with an additive error \( \epsilon \) in \( O(1/\epsilon^2) \) runtime cost.

**Pure states.** In this setting, the quantum kernel corresponds to a nonlinear kernel, i.e., \( k(a, a') = \langle a, \sqrt{Q(a')} \rangle \), where \( \mathbb{P}(a) \) stands for the probability of sampling \( a \) and \( \sum_a \mathbb{P}(a) = 1 \). With this regard, the explicit form of MMD loss yields

\[
\text{MMD}(\mathbb{P}_\theta || \mathbb{Q}) = E(k(x, x')) - 2 E(k(x, y)) + E(k(y, y'))
\]

\[
= \sum_x \sum_{x'} \mathbb{P}_\theta(x) \mathbb{P}_\theta(x') \left( \frac{x}{\sqrt{\mathbb{P}_\theta(x)}} \right) \left( \frac{x'}{\sqrt{\mathbb{P}_\theta(x')}} \right) - 2 \sum_x \sum_y \mathbb{P}_\theta(x) Q(y) \left( \frac{x}{\sqrt{\mathbb{P}_\theta(x)}} \right) \left( \frac{y}{\sqrt{Q(y)}} \right) + \sum_y \sum_{y'} \mathbb{Q}(y) \mathbb{Q}(y') \left( \frac{y}{\sqrt{Q(y)}} \right) \left( \frac{y'}{\sqrt{Q(y')}} \right)
\]

\[
= \sum_x \mathbb{P}_\theta(x) + \sum_y \mathbb{Q}(y) - 2 \sum_x \sqrt{\mathbb{P}_\theta(x) \mathbb{Q}(x)}
\]

\[
= 2 - 2 \sum_x \sqrt{\mathbb{P}_\theta(x) \mathbb{Q}(x)}. \tag{C5}
\]

The above results indicate that the evaluation of MMD loss amounts to calculating \( \sum_x \sqrt{\mathbb{P}_\theta(x) \mathbb{Q}(x)} \). Denote the generated state of QCBM as \( |\Phi(\theta)\rangle = U(\theta) |0\rangle^\otimes n = e^{i\phi} \sum_x \mathbb{P}_\theta(x) |x \rangle \), where \( \rho_0 = (|0\rangle \langle 0|)^\otimes n \). When the target
distribution $Q$ refers to a pure quantum state $|\Psi\rangle = \sum_y \sqrt{Q(y)} |y\rangle$, the term $\sum_x \sqrt{P_\theta(x) Q(x)}$ can be evaluated by Swap test \[104\], i.e.,

$$| \langle \Phi(\theta) | \Psi \rangle |^2 = \left( \sum_{i=1}^{2N} \sum_{j=1}^{2N} \sqrt{P_\theta(x) Q(y)} \langle x | y \rangle \right)^2 = \left( \sum_{i=1}^{2N} P_\theta(x) Q(x) \right)^2.$$  \hspace{1cm} (C6)

According to \[104\], taking account of the sample error, this term can be estimated within an additive error $\epsilon$ in $O(1/c^2)$ runtime complexity.

\[\square\]

**SM D: Proof of Theorem 1 (generalization of QCBMs)**

The proof of Theorem 1 adopts the following lemma.

**Lemma 2** (Adapted from Theorem 1, [87]). Suppose that the kernel $k(\cdot, \cdot)$ is bounded. Following the notations in Theorem 1 when the number of examples sampled from $P_{\delta(n,m)}$ and $Q$ is $n$ and $m$, with probability $1 - \delta$,

$$\text{MMD}(P_\theta^{(n,m)} || Q) \leq \inf_{\hat{\theta} \in \Theta} \text{MMD}(P_\theta || Q) + 2 \sqrt{\frac{2}{n} \frac{2}{m}} \sup_{x \in X} k(x, x) \left( 2 + \frac{2}{\delta} \right).$$  \hspace{1cm} (D1)

**Proof of Theorem 1** To prove Theorem 1, we first derive the upper bound of the generalization error $R_C$ under the generic setting. Then, we analyze $R_Q$ under the specific setting where the quantum kernel is employed and the target distribution $Q$ can be directly accessed by quantum machines.

**The calculation of $R_C$.** Recall the definition of $\hat{\theta}^{(n,m)}$ in Eq. (4). Let us first rewrite the generalization error as

$$R_C = \text{MMD}^2(P_{\hat{\theta}^{(n,m)}} || Q) - \inf_{\hat{\theta} \in \Theta} \text{MMD}^2(P_\theta || Q)$$

$$= \left( \text{MMD}(P_{\hat{\theta}^{(n,m)}} || Q) - \inf_{\hat{\theta} \in \Theta} \text{MMD}(P_\theta || Q) \right) \left( \text{MMD}(P_{\hat{\theta}^{(n,m)}} || Q) + \inf_{\hat{\theta} \in \Theta} \text{MMD}(P_\theta || Q) \right)$$

$$\leq 2C_1 \left( \text{MMD}(P_{\hat{\theta}^{(n,m)}} || Q) - \inf_{\hat{\theta} \in \Theta} \text{MMD}(P_\theta || Q) \right),$$  \hspace{1cm} (D2)

where the second equality uses $\inf_{\hat{\theta} \in \Theta} \text{MMD}^2(P_\theta || Q) = (\inf_{\hat{\theta} \in \Theta} \text{MMD}(P_\theta || Q))^2$ and the inequality employs the $\text{MMD}(P_{\hat{\theta}^{(n,m)}} || Q) + \inf_{\hat{\theta} \in \Theta} \text{MMD}(P_\theta || Q) \leq 2 \text{MMD}(P_{\hat{\theta}^{(n,m)}} || Q) \leq 2C_1$.

In conjunction with the above equation with the results of Lemma 2 we obtain that with probability at least $1 - \delta$, the upper bound of the generalization error of QGCM yields

$$R_C \leq 4C_1 \left( \frac{2}{n} \frac{2}{m} \sup_{x \in X} k(x, x) \left( 2 + \sqrt{\frac{2}{\delta}} \right) \right).$$  \hspace{1cm} (D3)

**The calculation of $R_Q$.** Recall the definition of $\hat{\theta}$ in Eq. (4). When QCBM adopts the quantum kernel and the target distribution $Q$ can be directly accessed by quantum machines, the minimum argument of the loss function yields $\hat{\theta} = \arg \min_{\theta \in \Theta} \text{MMD}^2(P_\theta || Q)$. Following the definition of the generalization error, we obtain

$$R_Q = \text{MMD}^2(P_{\hat{\theta}} || Q) - \inf_{\hat{\theta} \in \Theta} \text{MMD}^2(P_\theta || Q)$$

$$\leq \text{MMD}^2(P_{\hat{\theta}^{(n,m)}} || Q) - \inf_{\hat{\theta} \in \Theta} \text{MMD}^2(P_\theta || Q)$$

$$= R_C,$$  \hspace{1cm} (D4)

where the inequality is supported by the definition of $\hat{\theta}$, i.e, $\text{MMD}^2(P_{\hat{\theta}} || Q) = \min_{\theta \in \Theta} \text{MMD}^2(P_\theta || Q) \leq \text{MMD}^2(P_{\hat{\theta}^{(n,m)}} || Q)$.

Combining the results of $R_C$ and $R_Q$ in Eqs. (D3) and (D4), we obtain that with probability at least $1 - \delta$,

$$R_Q \leq R_C \leq 4C_1 \left( \frac{2}{n} \frac{2}{m} \sup_{x \in X} k(x, x) \left( 2 + \sqrt{\frac{2}{\delta}} \right) \right).$$  \hspace{1cm} (D5)

\[\square\]
In this section, we emphasize a central question in QGLMs, i.e., when both variational quantum circuits and neural networks are used to implement $P_\theta$, which one can attain a lower $\inf_{\Theta \in \Theta} \text{MMD}^2(P_\theta || Q)$. The importance of this issue comes from Eq. (4), where the generalization error bound becomes meaningful when $\inf_{\Theta \in \Theta} \text{MMD}^2(P_\theta || Q)$ is small. In this respect, it is necessary to understand whether QCBMs allow a lower $\inf_{\Theta \in \Theta} \text{MMD}^2(P_\theta || Q)$ over (classical) GLMs.

In what follows, we analyze when QCBMs promise a lower $\inf_{\Theta \in \Theta} \text{MMD}^2(P_\theta || Q)$ over a typical GLM—restricted Boltzmann machine (RBM) [83]. To be more specific, consider that both QCBMs and RBMs are universal approximators with an exponential number of trainable parameters [106, 107] with $\inf_{\Theta \in \Theta}$ Lemma 4. Following the notations in Theorem 2, define Lemma 3 (McDiarmid’s inequality, [109]), examples include projected entangled pair states and ground states of $\text{MMD}^2(P_\theta)$. To be more specific, consider that both QCBMs and RBMs are universal approximators

In this respect, it is necessary to understand whether QCBMs allow a lower $\inf_{\Theta \in \Theta} \text{MMD}^2(P_\theta || Q)$ over (classical) GLMs.

We are now ready to prove Theorem 2.

**SM F: Proof of Theorem 2** (generalization of QGANs)

The proof of Theorem 2 utilizes the following three lemmas. For clearness, we defer the proof of Lemmas 4 and 5 to SM F1 and F2 respectively.

**Lemma 3** (McDiarmid’s inequality, [109]). Let $f : \mathcal{X}_1 \times \mathcal{X}_2 \times \ldots \times \mathcal{X}_N \to \mathbb{R}$ and assume there exists $c_1, \ldots, c_2 \geq 0$ such that, for all $k \in \{1, \ldots, N\}$, we have

$$\sup_{x_1, \ldots, x_k, x_{k+1}, \ldots, x_N} |f(x_1, \ldots, x_k, \ldots, x_N) - f(x_1, \ldots, x_k, \ldots, x_N)| \leq c_k. \quad (F1)$$

Then for all $\epsilon \geq 0$ and independent random variables $\xi_1, \ldots, \xi_N$ in $\mathcal{X}$,

$$\Pr(|f(\xi_1, \ldots, \xi_N) - \mathbb{E}(f(\xi_1, \ldots, \xi_N))| \geq \epsilon) \leq \exp\left(\frac{-2\epsilon^2}{\sum_{n=1}^{N} c_n^2}\right). \quad (F2)$$

**Lemma 4.** Following the notations in Theorem 2 define $\mathcal{G} = \{k(G_\theta(z), G_\theta(z')) | \theta \in \Theta \}$ and $\mathcal{G}^+ = \{k(G_\theta(z), G_\theta(z')) | \theta \in \Theta, z \in \mathcal{Z} \}$. Given the set $\mathcal{S} = \{z^{(i)}_1\}_{i=1}^n$, we have

$$\mathbb{E}(\sup_{\theta \in \Theta} |\mathbb{E}_{z,z'}(k(G_\theta(z), G_\theta(z'))) - \frac{1}{n(n-1)} \sum_{i \neq j} k(G_\theta(z^{(i)}), G_\theta(z^{(j)})))| \leq \frac{8}{n-1} + \frac{24\sqrt{d^2_n(N_{ge} + N_{gt})}}{n-1} (1 + N \ln(1764C^2_3(n-1)N_{ge}N_{gt})). \quad (F3)$$

**Lemma 5.** Following the notations in Theorem 2 define $\mathcal{W} = \{k(G_\theta(\cdot), \cdot) | \theta \in \Theta \}$ and $\mathcal{W}^+ = \{k(G_\theta(\cdot), y) | \theta \in \Theta, y \in \mathcal{Y} \}$. Given the set $\mathcal{S} = \{z^{(i)}_1\}_{i=1}^n$ and the set $\{y^{(j)}_1\}_{j=1}^m$, we have

$$\mathbb{E}\left(\sup_{\theta \in \Theta} |\mathbb{E}_{z,y}(k(G_\theta(z), y)) - \frac{1}{mn} \sum_{i \in [n], j \in [m]} k(G_\theta(z^{(i)}), y^{(j)}))| \right) \leq \frac{8}{n} + \frac{24\sqrt{d^2_n(N_{ge} + N_{gt})}}{n} (1 + N \ln(1764C^2_3nN_{ge}N_{gt})). \quad (F4)$$

We are now ready to prove Theorem 2.
Proof of Theorem 2. Let $\mathcal{E}(\theta) = \text{MMD}^2_\nu(P_\theta \mid \mid Q^m)$ and $\mathcal{T}(\theta) = \text{MMD}^2(P_\theta \mid \mid Q)$. Note that the generalization error equals to

$$R^C = \mathcal{T}(\hat{\theta}^{(n,m)}) - \mathcal{T}(\theta^*).$$

(F5)

In the remainder of the proof, when no confusion occurs, we abbreviate $\hat{\theta}^{(n,m)}$ as $\hat{\theta}$ for clearness. The above equation is upper bounded by

$$R^C = \mathcal{E}(\hat{\theta}) - \mathcal{E}(\hat{\theta}) + \mathcal{T}(\hat{\theta}) - \mathcal{T}(\theta^*) \leq \mathcal{E}(\theta^*) - \mathcal{E}(\hat{\theta}) + \mathcal{T}(\hat{\theta}) - \mathcal{T}(\theta^*) \leq |\mathcal{T}(\hat{\theta}) - \mathcal{E}(\hat{\theta})| + |\mathcal{T}(\theta^*) - \mathcal{E}(\theta^*)|,$$

(F6)

where the first inequality employs the definition of $\hat{\theta}$ with $\mathcal{E}(\theta^*) \geq \mathcal{E}(\hat{\theta})$ and the second inequality uses the property of absolute value function. In the following, we derive the probability for $|\mathcal{T}(\hat{\theta}) - \mathcal{E}(\hat{\theta})| < \epsilon$, which in turn can achieve the upper bound of $R^C$.

According to the explicit form of the MMD loss, $\sup_{\theta \in \Theta} |\mathcal{E}(\theta) - \mathcal{T}(\theta)|$ satisfies

$$\sup_{\theta \in \Theta} |\text{MMD}^2_\nu(P_\theta \mid \mid Q^m) - \text{MMD}^2(P_\theta \mid \mid Q)|$$

$$= \sup_{\theta \in \Theta} \left| \mathbb{E}_{z,z'}(k(G_\theta(z), G_\theta(z'))) - 2 \mathbb{E}_{z,y}(k(G_\theta(z), y)) + \mathbb{E}_{y,y'}(k(y, y')) - \frac{1}{m(m-1)} \sum_{j \neq j'} k(G_\theta(z^{(j)}), G_\theta(z^{(j')})) \right|$$

$$+ \frac{2}{mn} \sum_{i \in [n], j \in [m]} k(G_\theta(z^{(i)}), y^{(j)}) - \frac{1}{m(m-1)} \sum_{j \neq j'} k(y^{(j)}, y^{(j')})$$

$$\leq \sup_{\theta \in \Theta} \left| \mathbb{E}_{y,y'}(k(y, y')) - \frac{1}{m(m-1)} \sum_{j \neq j'} k(y^{(j)}, y^{(j')}) \right| + \sup_{\theta \in \Theta} \left| \mathbb{E}_{z,z'}(k(G_\theta(z), G_\theta(z'))) - \frac{1}{n(n-1)} \sum_{i \neq i'} k(G_\theta(z^{(i)}), G_\theta(z^{(i')})) \right|$$

$$+ 2 \sup_{\theta \in \Theta} \left| \mathbb{E}_{z,y}(k(G_\theta(z), y)) - \frac{1}{mn} \sum_{i \in [n], j \in [m]} k(G_\theta(z^{(i)}), y^{(j)}) \right|$$

$$= T_1 + T_2 + T_3$$

(F7)

where the inequality comes from the Jensen inequality.

We next separately derive the upper bounds of the terms $T_1, T_2, T_3$ in Eq. (F7).

Upper bound of $T_1$. The upper bound of $T_1$ only depends on the examples sampled from the target distribution $Q$, which is independent of $\Theta$. With this regard, $T_1$ can be taken out of the supremum and we apply the concentration inequality in Lemma 3 to derive the upper bound of $|\mathbb{E}_{y,y'}(k(y, y')) - \frac{1}{m(m-1)} \sum_{j \neq j'} k(y^{(j)}, y^{(j')})|$. Recall the precondition of employing Lemma 3 is finding the upper bound on $|\mathbb{E} - \mathbb{E}|$. For each $\ell \in \{1, \ldots, m\}$, the desired upper bound yields

$$\left| - \frac{1}{m(m-1)} \left( \sum_{j \neq j', j \neq \ell} k(y^{(j)}, y^{(j')}) + \sum_{j' \neq \ell} k(y^{(\ell)}, y^{(j')}) \right) + \frac{1}{m(m-1)} \left( \sum_{j \neq j', j \neq \ell} k(y^{(j)}, y^{(j')}) + \sum_{j' \neq \ell} k(y^{(\ell)}, y^{(j')}) \right) \right|$$

$$= \left| \frac{1}{m(m-1)} \sum_{j' \neq \ell} \left( k(y^{(\ell)}, y^{(j')}) - k(y^{(j)}, y^{(j')}) \right) \right|$$

$$\leq \frac{2C_2}{m},$$

(F8)

where the inequality leverages the assumption that the kernel $k(\cdot, \cdot)$ is upper bounded by $C_2$.

Given this upper bound, we obtain

$$\Pr(T_1 \geq \epsilon)$$

$$= \Pr \left( \left| \mathbb{E}_{y,y'}(k(y, y')) - \frac{1}{m(m-1)} \sum_{j \neq j'} k(y^{(j)}, y^{(j')}) \right| \geq \epsilon \right)$$

$$\leq \exp \left( - \frac{\epsilon^2}{8C_2^2 m} \right) = \delta_{T_1},$$

(F9)
where the inequality exploits the results in Lemma 3. 

**Upper bound of $T_2$.** We next use the concentration inequality to quantify the upper bound of $T_2$. The derivation is similar to that of $T_1$. In particular, supported by the results of Lemma 3, we have

$$\Pr(|T_2 - \mathbb{E}(T_2)| \geq \epsilon) \leq \exp\left(-\frac{\epsilon^2}{8C_2^2 n}\right) = \delta_{T_2}. \quad (F10)$$

Suppose that $\mathbb{E}(T_2) \leq \epsilon_1$, an immediate observation is that

$$\Pr(T_2 \geq \epsilon_1 + \epsilon) \leq \exp\left(-\frac{\epsilon^2}{8C_2^2 n}\right) = \delta_{T_2}. \quad (F11)$$

In other words, the derivation of the upper bound of $T_2$ amounts to analyzing the upper bound $\epsilon_1$.

**Upper bound of $T_3$.** Following the same routine with the derivation of the upper bound of $T_2$, we obtain

$$\Pr(|T_3 - \mathbb{E}(T_3)| \geq \epsilon) \leq \exp\left(-\frac{\epsilon^2 \cdot nm}{8C_2^2 (n + m)}\right) = \delta_{T_3}. \quad (F12)$$

Suppose that $\mathbb{E}(T_3) \leq \epsilon_2$. The above result hints that

$$\Pr(T_3 \geq \epsilon_2 + \epsilon) \leq \exp\left(-\frac{\epsilon^2 \cdot nm}{8C_2^2 (n + m)}\right) = \delta_{T_3}. \quad (F13)$$

Summing up Eqs. (F9), (F11), and (F13), the union bound gives

\[
\Pr \left( \sup_{\theta \in \Theta} |\mathcal{E}(\theta) - T(\theta)| \geq \epsilon_1 + 2\epsilon_2 + 4\epsilon \right) \leq \delta_{T_1} + \delta_{T_2} + \delta_{T_3} \\
\Rightarrow \Pr \left( \sup_{\theta \in \Theta} |\mathcal{E}(\theta) - T(\theta)| \geq \epsilon_1 + 2\epsilon_2 + 4\epsilon \right) \leq 3\delta_{T_3} \\
\Rightarrow \Pr \left( |\mathcal{E}(\hat{\theta}) - T(\hat{\theta})| \geq \epsilon_1 + 2\epsilon_2 + 4\epsilon \right) \leq 3\delta_{T_3}. \quad (F14)
\]

This yields that with probability at least $1 - 3\delta_{T_3}$,

$$2(\epsilon_1 + 2\epsilon_2 + 4\epsilon) \geq |\mathcal{E}(\hat{\theta}) - T(\hat{\theta})| + |\mathcal{E}(\theta^*) - T(\theta^*)| \geq |\mathcal{E}(\hat{\theta}) - T(\hat{\theta}) - \mathcal{E}(\theta^*) + T(\theta^*)| \geq |T(\hat{\theta}) - T(\theta^*)| = \mathcal{R}^C. \quad (F15)$$

According to the explicit forms of $\epsilon_1$ and $\epsilon_2$ achieved in Lemmas 4 and 5, with probability $1 - 3\delta_{T_3}$, the generalization error of QGANs is upper bounded by

$$\mathcal{R}^C \leq 8\epsilon + 2 \left( \frac{8}{n - 1} + \frac{24\sqrt{d_k(N_{ge} + N_{gt})}}{n - 1} \left( 1 + N \ln(1764C_2^2(n - 1)N_{ge}N_{gt}) \right) \right) + 4 \left( \frac{8}{n} + \frac{24\sqrt{d_k(N_{ge} + N_{gt})}}{n} \left( 1 + N \ln(1764C_2^2nN_{ge}N_{gt}) \right) \right) \leq 8\sqrt{C_2^2(n + m) \frac{1}{nm}} \ln\left( \frac{1}{3\delta_{T_3}} \right) + 6 \left( \frac{8}{n - 1} + \frac{24\sqrt{d_k(N_{ge} + N_{gt})}}{n - 1} \left( 1 + N \ln(1764C_2^2nN_{ge}N_{gt}) \right) \right). \quad (F16)$$

where the last inequality uses the relation between $\delta_{T_3}$ and $\epsilon$ in Eq. (F13) with $\epsilon = \sqrt{8C_2^2(n + m) \ln(1/3\delta_{T_3})/(nm)}$, $1/n < 1/(n - 1)$, and $n - 1 < n$.

\[\square\]

1. **Proof of Lemma 4**

Recall Lemma 4 aims to derive the upper bound of $\mathbb{E}(T_2)$ in Eq. (F7). In Ref. [81], the authors utilize a statistical measure named Rademacher complexity to quantify these two terms. Different from the classical counterpart, here we adopt another statistical measure, i.e., covering number, to derive the upper bound of $\mathbb{E}(T_2)$. This measure allows us to identify how $\mathbb{E}(T_2)$ scales with the qubit count $N$ and the architecture of the employed Ansatz such as the trainable parameters $N_{gt}$ and the types of the quantum gates. For self-consistency, we provide the formal definition of covering number and Rademacher as follows.
Definition 2 (Covering number, [110]). The covering number $\mathcal{N}(\mathcal{U}, \varepsilon, \| \cdot \|)$ denotes the least cardinality of any subset $\mathcal{V} \subset \mathcal{U}$ that covers $\mathcal{U}$ at scale $\varepsilon$ with a norm $\| \cdot \|$, i.e.,

$$\sup_{A \in \mathcal{U}} \min_{B \in \mathcal{V}} \| A - B \| \leq \varepsilon.$$  

(\text{F17})

Definition 3 (Rademacher, [110]). Let $\mu$ be a probability measure on $\mathcal{X}$, and let $\mathcal{F}$ be a class of uniformly bounded functions on $\mathcal{X}$. Then the Rademacher complexity of $\mathcal{F}$ is

$$\mathcal{R}_n(\mathcal{F}) = \mathbb{E}_\mu \mathbb{E}_{\sigma_1, \ldots, \sigma_n} \left( \frac{1}{\sqrt{n}} \sup_{f \in \mathcal{F}} \left| \sum_{i=1}^n \sigma_i f(x^{(i)}) \right| \right),$$

(\text{F18})

where $\sigma = (\sigma_1, \ldots, \sigma_n)$ is a sequence of independent Rademacher variables taking values in $\{-1, 1\}$ and each with probability $1/2$, and $x_1, \ldots, x_n \in \mathcal{X}$ are independent, $\mu$-distributed random variables.

Intuitively, the covering number concerns the minimum number of spherical balls with radius $\varepsilon$ that occupies the whole space; the Rademacher complexity measures the ability of functions from $\mathcal{F}$ to fit random noise. The relation between Rademacher complexity and covering number is established by the following Dudley entropy integral bound.

Lemma 6 (Adapted from [81, 111]). Let $\mathcal{F} = \{ f : \mathcal{X} \times \mathcal{X} \to \mathbb{R} \}$ and $\mathcal{F}_+ = \{ h = f(x, \cdot) : f \in \mathcal{F}, x \in \mathcal{X} \}$ and $\mathcal{F}_+ \subset B(L_\infty(\mathcal{X}))$. Given the set $\mathcal{S} = \{ x^{(1)}, \ldots, x^{(n)} \} \subset \mathcal{X}$, denote the Rademacher complexity of $\mathcal{F}_+$ as $\mathcal{R}_n(\mathcal{F}_+)$, it satisfies

$$\mathcal{R}_n(\mathcal{F}_+) \leq \inf_{\alpha > 0} \left( 4\alpha + \frac{12}{\sqrt{n}} \int_0^\infty \frac{\sqrt{\ln(\mathcal{N}(\mathcal{F}_+|_{\mathcal{S}}, \varepsilon, \| \cdot \|_2))}}{\varepsilon} d\varepsilon \right),$$

(\text{F19})

where $(\mathcal{F}_+|_{\mathcal{S}}) = \{ \{f(x, x^{(i)})\}_{i=1}^n : f \in \mathcal{F}, x \in \mathcal{X} \}$ denotes the set of vectors formed by the hypothesis with $\mathcal{S}$.

Ref. [81] hinges on the term $\mathbb{E}(T2)$ with the Rademacher complexity, as stated in the following lemma.

Lemma 7 (Adapted from Lemma 1, [81]). Following notations in Theorem 2 and Lemma 6 define $\mathcal{G}_+ = \{ k(G_\theta(\cdot), G_\theta(\cdot))|\theta \in \Theta \}$ and $\mathcal{G}_+ = \{ k(G_\theta(x), G_\theta(z))|\theta \in \Theta, x, z \in \mathbb{Z} \}$. Given the set $\mathcal{S} = \{ z^{(i)} \}_{i=1}^n$, we have

$$\mathbb{E}(T2) \leq \frac{2}{\sqrt{n-1}} \mathcal{R}_{n-1}(\mathcal{G}_+),$$

where $\mathcal{R}_{n-1}(\mathcal{G}_+)$ refers to the Rademacher's complexity of $\mathcal{G}_+$.

In conjunction with the above two lemmas, the term $\mathbb{E}(T2)$ is upper bounded by the covering number of $\mathcal{G}_+$. As such, the proof of Lemma 3 utilizes the following three lemmas, which are used to formalize the relation of covering number of two metric spaces, quantify the covering number of variational quantum circuits, and evaluate the covering number of the space living in N-qubit quantum states, respectively.

Lemma 8 (Lemma 5, [112]). Let $\langle \mathcal{H}_1, d_1 \rangle$ and $\langle \mathcal{H}_2, d_2 \rangle$ be metric spaces and $f : \mathcal{H}_1 \to \mathcal{H}_2$ be bi-Lipschitz such that

$$d_2(f(x), f(y)) \leq Kd_1(x, y), \ \forall x, y \in \mathcal{H}_1,$$

(\text{F20})

and

$$d_2(f(x), f(y)) \geq kd_1(x, y), \ \forall x, y \in \mathcal{H}_1 \text{ with } d_1(x, y) \leq r.$$  

(\text{F21})

Then their covering numbers obey

$$\mathcal{N}(\mathcal{H}_1, 2\varepsilon/k, d_1) \leq \mathcal{N}(\mathcal{H}_2, \varepsilon, d_2) \leq \mathcal{N}(\mathcal{H}_1, \varepsilon/K, d_1),$$

(\text{F22})

where the left inequality requires $\varepsilon \leq kr/2$.

Lemma 9 (Lemma 2, [70]). Define the operator group as

$$\mathcal{H}_{\text{circ}} := \{ \tilde{U}(\theta) \Pi_j \tilde{U}(\theta) | \theta \in \Theta \}.$$  

(\text{F23})

Suppose that the employed encoding Ansatz $\tilde{U}(\theta)$ containing in total $N_g$ gates, each gate $\tilde{u}_i(\theta)$ acting on at most $k$ qudits, and $N_g \leq N_q$ gates in $\tilde{U}(\theta)$ are trainable. The $\varepsilon$-covering number for the operator group $\mathcal{H}_{\text{circ}}$ with respect to the operator-norm distance obeys

$$\mathcal{N}(\mathcal{H}_{\text{circ}}, \varepsilon, \| \cdot \|) \leq \left( \frac{T N_g \| \Pi_j \|}{\varepsilon} \right)^{d^2 K N_g},$$

(\text{F24})

where $\| \Pi_j \|$ denotes the operator norm of $\Pi_j$. 

Lemma 10. Define the input state group as $\mathcal{B} := \{ \rho_z := \hat{U}(z)^\dagger|\{0\}\rangle\langle 0|\hat{U}(z)|z \in \mathcal{Z} \}$. Suppose that the employed quantum circuit $\hat{U}(z)$ containing in total $N_{ge}$ parameterized gates to load $z$ and each gate $\hat{u}_i(z)$ acting on at most $k$ qudits. The $\epsilon$-covering number for $\mathcal{B}$ with respect to the operator-norm distance obeys

$$N(\mathcal{B}, \epsilon, \| \cdot \|) \leq \left( \frac{7N_{ge}}{\epsilon} \right)^{d_{2k}N_{ge}}.$$ (F25)

Proof of Lemma 10. The proof is identical to that presented in Lemma 2 of Ref. [70].

We are now ready to prove Lemma 4.

Proof of Lemma 4. Recall the aim of Lemma 4 is to obtain the upper bound of $\mathbb{E}(T2)$. In conjunction with Lemmas 6 and 7, we obtain

$$\mathbb{E}(T2) \leq \mathbb{E} \frac{2}{\sqrt{n-1}} \inf_{\delta > 0} \left( 4\alpha + \frac{12}{\sqrt{n-1}} \int_0^1 \sqrt{\ln(N(\mathcal{G}_+|S, \epsilon, \| \cdot \|_2))} \, d\epsilon \right),$$ (F26)

where $\mathcal{G}_+ = \{ k(G_\theta(z), G_\theta(z)) \theta \in \Theta, z \in \mathcal{Z} \}$ and $S$ denotes the set $\{ z(1), ..., z(n-1) \}$ sampled from the prior distribution $P_Z$, and $(\mathcal{G}_+)_S = \{ \| k(G_\theta(z), G_\theta(z)) \|_{i=1:n-1} : \theta \in \Theta, z \in \mathcal{Z} \}$ denotes the set of vectors formed by the hypothesis with $S$. In other words, the upper bound of $\mathbb{E}(T2)$ is quantified by the covering number $N((\mathcal{G}_+)_S, \epsilon, \| \cdot \|_2)$.

We next follow the definition of covering number to quantify how $N((\mathcal{G}_+)_S, \epsilon, \| \cdot \|_2)$ depends on the structure information of the employed Ansatz and the input quantum states. Denote $\mathcal{G}_{\epsilon_1}$ as an $\epsilon_1$-covering of the set $\mathcal{Q}_1 = \{ G_\theta(z) \theta \in \Theta \}$ and $\mathcal{G}_{\epsilon_3}$ as an $\epsilon_3$-covering of the set $\mathcal{Q}_3 = \{ G_{\theta}(z) | z \in \mathcal{Z} \}$. Then, the covering number $N((\mathcal{G}_+)_S, \epsilon, \| \cdot \|_2)$ can be upper bounded by $N((\mathcal{Q}_1|S, \epsilon_1, \| \cdot \|_2)$ and $N((\mathcal{Q}_3|S, \epsilon_3, \| \cdot \|_3)$. Mathematically, according to the explicit expression of $(\mathcal{G}_+)_S$, we have for any $(\theta, z)$ and $(\theta', z')$

$$\| k(G_\theta(z), G_\theta(z)) \|_{i=1:n-1} - \| k(G_{\theta'}(z'), G_{\theta'}(z')) \|_{i=1:n-1}$$

$$\leq \| k(G_\theta(z), G_\theta(z)) \|_{i=1:n-1} - \| k(G_{\theta}(z), G_{\theta}(z)) \|_{i=1:n-1} + \| k(G_{\theta'}(z), G_{\theta'}(z')) \|_{i=1:n-1}$$

$$\| k(G_{\theta'}(z), G_{\theta'}(z')) \|_{i=1:n-1} - \| k(G_{\theta}(z), G_{\theta}(z)) \|_{i=1:n-1} - \| k(G_{\theta'}(z'), G_{\theta'}(z')) \|_{i=1:n-1}$$

$$\leq \| k(G_\theta(z), G_\theta(z)) \|_{i=1:n-1} - \| k(G_{\theta}(z), G_{\theta}(z)) \|_{i=1:n-1} + \| k(G_{\theta'}(z), G_{\theta'}(z')) \|_{i=1:n-1}$$

$$\leq \| k(G_{\theta}(z), G_{\theta}(z)) \|_{i=1:n-1} - \| k(G_{\theta'}(z'), G_{\theta'}(z')) \|_{i=1:n-1}$$

$$\leq C_3 \left( \sqrt{n-1} \| G_\theta(z) - G_{\theta'}(z') \| + \| G_{\theta'}(z) - G_{\theta'}(z') \| \right),$$ (F27)

where the first inequality uses the triangle inequality and the last inequality exploits $C_3$-Lipschitz property of the kernel. Following the definition of covering number, the above relationship indicates that if for any $\theta$ there exists $\theta'$ such that $\| G_{\theta}(z) - G_{\theta'}(z') \| \leq \epsilon_1$ holds for every $z$, and for any $z$ there exists $z'$ such that $\| G_{\theta}(z) - G_{\theta'}(z') \| \leq \epsilon_3$ holds for every $\theta$, the composition of the covering sets $\mathcal{Q}_{\epsilon_1}$ and $\mathcal{Q}_{\epsilon_3}$ forms the covering set of $(\mathcal{G}_+)_S$. That is, the covering number of $(\mathcal{G}_+)_S$ is upper bounded by

$$N((\mathcal{G}_+)_S, C_3\sqrt{n-1} - 1(2\epsilon_1 + \epsilon_3), \| \cdot \|_2) \leq N((\mathcal{Q}_1, \epsilon_1, \| \cdot \|_2) \times N((\mathcal{Q}_3, \epsilon_3, \| \cdot \|_3).$$ (F28)

In other words, to quantify the $\epsilon$-covering of $(\mathcal{G}_+)_S$, it is equivalent to deriving the upper bound of $N((\mathcal{Q}_1, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2)$ and $N((\mathcal{Q}_3, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2)$, respectively. We next separately derive these two quantities.

The upper bound of $N((\mathcal{Q}_1, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2)$, Let $\mathcal{Q}_4$ be an $\epsilon/(3C_3\sqrt{n-1})$-cover of $\mathcal{H}_{circ}$ in Eq. (F23). Then, for any $\theta$, there exists $\theta'$ such that $\| \hat{U}(\theta)P_j\hat{U}(\theta) - \hat{U}(\theta')P_j\hat{U}(\theta') \| \leq \frac{\epsilon}{3C_32^{\sqrt{n-1}}}$ for every $j$ with $\hat{U}(\theta)P_j\hat{U}(\theta') \in \mathcal{Q}_4$. 

This leads that for any $z$, we have
\[
\|G_{\theta}(z) - G_{\theta'}(z)\|_2 \\
= \left\| \left( \text{Tr}(\hat{U}(\theta)\Pi_j\hat{U}(\theta')^\dagger)\rho_z - \text{Tr}(\hat{U}(\theta')\Pi_j\hat{U}(\theta)^\dagger)\rho_z \right)_{j=1:2^N} \right\|_2 \\
\leq \left\| \left( \|U(\theta)\Pi_j\hat{U}(\theta')^\dagger - \hat{U}(\theta')\Pi_j\hat{U}(\theta)^\dagger \| \right)_{j=1:2^N} \right\|_2 \\
\leq \frac{2^N}{3C_32^N}\sqrt{n-1},
\]

where the first inequality uses the Cauchy-Schwartz inequality and the second inequality follows the definition of covering number.

The above observation means that the covering set of $\mathcal{N}(Q_{1}, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2)$ is independent with $z$ and its covering number is upper bound by $\mathcal{N}(\mathcal{H}_{circ}, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2)$. Then, by leveraging the results in Lemma 10, we obtain
\[
\mathcal{N}(Q_{1}, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2) \leq \mathcal{N}(\mathcal{H}_{circ}, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2) \leq \left( \frac{21C_32^N\sqrt{n-1}TN_{ge}}{\epsilon} \right)^{d_{2k}N_{ge}}.
\]

The upper bound of $\mathcal{N}(Q_{3}, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2)$ is $\mathcal{N}(\mathcal{B}, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2)$. Let $Q_{5}$ be an $\frac{\epsilon}{3C_32^N\sqrt{n-1}}$-cover of $B$ in Eq. [F25]. Then, for any encoding state $\rho_z \in B$, there exists $\rho_{z'} \in Q_{5}$ with $\|\rho_z - \rho_{z'}\| \leq \frac{\epsilon}{3C_32^N\sqrt{n-1}}$. By expanding the term $\|G_{\theta'}(z) - G_{\theta'}(z')\|$, we obtain the following result, i.e., for any $\theta'$,
\[
\left\| G_{\theta}(z) - G_{\theta'}(z') \right\| \\
= \left\| \left( \text{Tr}(\Pi_j\hat{U}(\theta')\rho_z\hat{U}(\theta')^\dagger) - \text{Tr}(\Pi_j\hat{U}(\theta')\rho_{z'}\hat{U}(\theta')^\dagger) \right)_{j=1:2^N} \right\| \\
= \left\| \left( \|\rho_z - \rho_{z'}\| \right)_{j=1:2^N} \right\| \\
\leq \frac{2^N}{3C_32^N}\sqrt{n-1},
\]

where the first inequality uses $\text{Tr}(AB) \leq \text{Tr}(A)\|B\|$ when $0 \leq A$ and $\text{Tr}(\hat{U}(\theta')\Pi_j\hat{U}(\theta')^\dagger) = \text{Tr}(\Pi_j) = 1$ for all $j \in [2^N]$, and the last inequality follows the definition of covering number. The achieved relation means that the covering set of $\mathcal{N}(Q_{3}, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2)$ does not depend on $\theta$ and its covering number is upper bounded by $\mathcal{N}(\mathcal{B}, \epsilon/(3C_32^N\sqrt{n-1}), \| \cdot \|_2)$.

Then, based on the results in Lemma 10, we have
\[
\mathcal{N}(Q_{3}, \epsilon/(3C_3\sqrt{n-1}), \| \cdot \|_2) \leq \mathcal{N}(\mathcal{B}, \epsilon/(3C_32^N\sqrt{n-1}), \| \cdot \|_2) \leq \left( \frac{21C_32^N\sqrt{n-1}TN_{ge}}{\epsilon} \right)^{d_{2k}N_{ge}}.
\]

Combining Eqs. [F30] and [F32], the covering number $\mathcal{N}((G_{+})_{\|S\|}, \epsilon, \| \cdot \|_2)$ in Eqn. [F28] is upper bounded by
\[
\mathcal{N}((G_{+})_{\|S\|}, \epsilon, \| \cdot \|_2) \\
\leq \mathcal{N}(\mathcal{H}_{circ}, \epsilon/(3C_32^N\sqrt{n-1}), \| \cdot \|_2) \times \mathcal{N}(\mathcal{B}, \epsilon/(3C_32^N\sqrt{n-1}), \| \cdot \|_2) \\
\leq \left( \frac{21C_32^N\sqrt{n-1}TN_{ge}}{\epsilon} \right)^{d_{2k}N_{ge}} \left( \frac{21C_32^N\sqrt{n-1}TN_{ge}}{\epsilon} \right)^{d_{2k}N_{ge}} \\
= (21C_32^N\sqrt{n-1}TN_{ge})^{d_{2k}N_{ge}} (21C_32^N\sqrt{n-1}TN_{ge})^{d_{2k}N_{ge}} \left( \frac{1}{\epsilon} \right)^{d_{2k}N_{ge}N_{ge}}.
\]
Denote \( C_5 = (21C_32^N \sqrt{n - 1}N_{gt})^{2^{k(N_{ge} + N_{gt})}} (21C_32^N \sqrt{n - 1}N_{ge})^{2^{k(N_{ge} + N_{gt})}} \). Using Lemma 6 we obtain

\[
E(T2) \leq \frac{2}{\sqrt{n - 1}} \inf_{\alpha > 0} \left( 4\alpha + \frac{12}{\sqrt{n - 1}} \int_{\alpha}^{1} \ln \left( \left( \frac{C_5}{\epsilon} \right)^{d^{2k(N_{ge} + N_{gt})}} \right) \, d\epsilon \right)
\]

\[
= \frac{2}{\sqrt{n - 1}} \inf_{\alpha > 0} \left( 4\alpha + \frac{12\sqrt{d^{2k(N_{ge} + N_{gt})}}}{\sqrt{n - 1}} \int_{\alpha}^{1} \sqrt{\ln \left( \left( \frac{C_5}{\epsilon} \right)^{d^{2k(N_{ge} + N_{gt})}} \right)} \, d\epsilon \right)
\]

\[
\leq \frac{2}{\sqrt{n - 1}} \inf_{\alpha > 0} \left( 4\alpha + \frac{12\sqrt{N_{ge} + N_{gt}}^{d^{2k(N_{ge} + N_{gt})}}}{\sqrt{n - 1}} \left( \epsilon + \epsilon \ln \left( \left( \frac{C_5}{\epsilon} \right)^{d^{2k(N_{ge} + N_{gt})}} \right) \right) \right)_{\epsilon = \alpha}.
\]  

(F34)

For simplicity, we set \( \alpha = 1/\sqrt{n - 1} \) in Eq. (F34) and then \( E(T2) \) is upper bounded by

\[
E(T2) \leq \frac{2}{n - 1} \left( \frac{4}{\sqrt{n - 1}} + \frac{12\sqrt{N_{ge} + N_{gt}}^{d^{2k(N_{ge} + N_{gt})}}}{\sqrt{n - 1}} \left( \epsilon + \epsilon \ln \left( \left( \frac{C_5}{\epsilon} \right)^{d^{2k(N_{ge} + N_{gt})}} \right) \right) \right)_{\epsilon = \alpha}
\]

\[
\leq \frac{8}{n - 1} + \frac{24\sqrt{N_{ge} + N_{gt}}^{d^{2k(N_{ge} + N_{gt})}}}{n - 1} \left( 1 + \ln C_5 \right).
\]  

(F35)

Since the two exponent terms in \( C_5 \) are no larger than 1, we have \( C_5 \leq (21C_32^N \sqrt{n - 1}N_{gt})(21C_32^N \sqrt{n - 1}N_{ge}) \). This relation further simplifies the upper bound \( E(T2) \) as

\[
E(T2) \leq \frac{8}{n - 1} + \frac{24\sqrt{N_{ge} + N_{gt}}^{d^{2k(N_{ge} + N_{gt})}}}{n - 1} \left( 1 + N \ln(1764C_3^2(n - 1)N_{ge}N_{gt}) \right).
\]  

(F36)

\[ \square \]

2. Proof of Lemma 5

Proof of Lemma 5. The proof of Lemma 5 is very similar to the one of Lemma 4 and thus we skip it here. \[ \square \]

SM G: Generalization of QGANs with varied Ansätze

The derived generalization error bound in Theorem 2 is succinct, which can be directly employed to quantify the generalization ability of QGANs with the specified Ansätze. For concreteness, here we separately analyze the generalization ability of QGANs with two typical classes of Ansätze, i.e., the hardware-efficient Ansätze and the quantum approximation optimization Ansätze.

Hardware-efficient Ansätze. An \( N \)-qubits hardware-efficient Ansatz is composed of \( L \) layers, i.e., \( U(\theta) = \prod_{l=1}^LU(\theta^l) \) with \( L \sim \text{poly}(N) \), where \( U(\theta^l) \) is composed of parameterized single-qubit gates and fixed two-qubit gates. In general, the topology of \( U(\theta^l) \) for any \( l \in [L] \) is the same and each qubit interacts with at least one parameterized single-qubit gate and two qubits gate. Mathematically, we have \( U(\theta^l) = (\otimes_{i=1}^NU)U_{eng} \) with \( U_s = \mathcal{R}_z(\beta)\mathcal{R}_y(\gamma)\mathcal{R}_z(\nu) \) being realized by three rotational qubit gates and \( \gamma, \beta, \nu \in [0, 2\pi] \). The number of two-qubit gates in each layer is set as \( N \) and the connectively of two-qubit gates aims to adapt to the topology restriction of quantum hardware. The entangled layer \( U_{eng} \) contains two-qubit gates, i.e., CNOT gates, whose connectivity adapts the topology of the quantum hardware.

An example of 4-qubit QGAN with hardware-efficient Ansatz is illustrated in the left panel of Fig. 2(a). Under this setting, when both the encoding unitary and the trainable unitary adopt the hardware-efficient Ansatz, we have \( k = 2, d = 2, N_{ge} = L_EN, N_{gt} = L(3N) \), and \( N_g = L(3N + N) = 4L \). Based on the above settings, we achieve the generalization error of ab \( N \)-qubit QGAN with the hardware-efficient Ansätze, supported by Theorem 2, i.e., with probability at least \( 1 - \delta \),

\[
\mathcal{R}^C \leq 8\sqrt{\frac{8C_2^2(n + m)}{nm}} \ln \frac{1}{\delta} + \frac{48}{n - 1} + \frac{576}{n - 1} \sqrt{N(L_E + 3L)}(N \ln(5292C_3^2n2^L_E + 1)) \]

\[ (G1) \]

where \( C_2 = \sup_{\mathfrak{x}, \mathfrak{x}'} k(\mathfrak{x}, \mathfrak{x}) \).
Quantum approximate optimization Ansätze. Fig. 4 (b) plots the quantum approximate optimization Ansätze. The mathematical expression of this Ansatz takes the form $U(\theta^i) = \prod_{l=1}^{L} U(\theta^i)$, where the $l$-th layer $U(\theta^i) = U_B(\theta^i)U_C(\theta^i)$ is implemented by the driven Hamiltonian $U_B(\theta^i) = \otimes_{i=1}^{N} R_X(\theta^i)$ and the target Hamiltonian $U_C(\theta^i) = \exp(-i \theta_{l+1} H_C)$ with $H_C$ being a specified Hamiltonian. Under this setting, when the encoding unitary is constructed by the hardware-efficient Ansatz and the trainable unitary is realized by the quantum approximate optimization Ansätze, we have $k = N$, $d = 2$, $N_{ge} = L_{E}N$, and $N_{gt} = L(N + 1)$. Based on the above settings, we achieve the generalization error of an $N$-qubit QGAN with the quantum approximate optimization Ansätze, supported by Theorem 2, i.e., with probability at least $1 - \delta$,

$$\mathcal{R}^C \leq \sqrt{8C_2^2(n + m) \ln \frac{1}{\delta} + \frac{\delta}{n - 1} + \frac{144 \times 2^N \sqrt{(N + 1)(L + L_E)}}{n - 1} (N \ln(1764C_2^2 n LL_{E}N(N + 1)) + 1),}$$

where $C_2 = \sup_{\sigma}\langle \kappa(x, x) \rangle$.

SM H: More details of numerical simulations

1. Hyper-para and metrics

RBF kernel. The explicit expression of the radial basis function (RBF) kernel is $k(x, y) = \exp(-\frac{||x - y||^2}{\sigma^2})$, where $\sigma$ refers to the bandwidth. In all simulations, we set $\sigma^{-2}$ as $\{0.25, 4\}$ for QCBMs and $\{-0.001, 1, 10\}$ for QGANs, respectively.

KL divergence. We use the KL divergence to measure the similarity between the generated distribution $P_\theta$ and true distribution $Q$. In the discrete setting, its mathematical expression is $KL(P_\theta \| Q) = \sum_i P_\theta(i) \log(Q(i)/P_\theta(i)) \in [0, \infty)$. In the continuous setting, $KL(P_\theta \| Q) = \int Q(dx) \log(Q(dx)/P_\theta(dx)) dx \in [0, \infty)$. When the two distributions are exactly matched with $P = Q$, we have $KL(P \| Q) = 0$.

State fidelity. Suppose that the generated state is $|\Psi(\theta)\rangle$ and the target state is $|\Psi^*\rangle$. The state fidelity $\mathcal{F}$ for pure states is defined as $F = |\langle \Psi(\theta)|\Psi^*\rangle|^2$.

Optimizer. For QCBMs, the classical optimizer is assigned as L-BFGS-B algorithm and the tolerance for termination is set as $10^{-12}$. For QGANs, the classical optimizer is assigned as Adam with default parameters. The source code. The QCBM is realized by Python, Numpy, Scipy, Tensorflow, Keras, and QIBO. We release the source code in Github repository.

Hardware parameters. All simulation results in this study are completed by the classical device with Intel(R) Xeon(R) Gold 6267C CPU @ 2.60GHz and 128 GB memory.
2. More simulation results related to the task of discrete Gaussian approximation

Training loss of QCBMs. Fig. 5(a) plots the last iteration training loss of QCBMs. All hyper-parameter settings are identical to those introduced in the main text. The x-axis stands for the setting of \( n \) used to compute \( \text{MMD}_Y \) in Eq. (A2). The simulation results indicate that the performance QCBM with RBM kernel is steadily enhanced with the increased \( n \). When \( n \to \infty \), its performance approaches to the QCBM with quantum kernel. These phenomenons accord with Theorem 1.

Effect of circuit depth. We explore the performance of QCBMs with quantum kernels by varying the employed Ansatz. Specifically, we consider the case of \( N = 12 \) and set \( L_1 \) in Fig. 1(a) as 4, 6, 8, 10. The collected simulation results are shown in Fig. 5(b). In conduction with Fig. 1(c), QCBM with \( L_1 = 6 \) attains the best performance over all settings, where the achieved MMD loss is \( 7 \times 10^{-5} \) and the KL divergence is 0.03. This observation implies that properly controlling the expressivity of Ansatz, which effects the term \( C_1 \) in Theorem 1, contributes to improve the learning performance of QCBM.

3. More simulation results related to the task of GHZ state approximation

Approximated states. The approximated GHZ states of QGANs with different random seeds discussed in Fig. 3(d) are depicted in Fig. 6. Specifically, the difference between the approximated state and the target GHZ state becomes apparent with the decreased number of examples \( n \) and the increased number of qubits \( N \). These observations echo with the statement of Theorem 1.

4. More simulation results related to the task of 3D Gaussian approximation

Implementation of the modified style-QGANs. Recall the quantum generator adopted \( G_\theta(z) \) in the modified style-QGANs is exhibited in Fig. 3(a). Different from the original proposal applying the re-uploading method, the modified quantum generator first uploads the prior example \( z \) using \( U(\gamma) \) followed by the Ansatz \( U(\theta) \). Such a modification facilitates the analysis of the generalization behavior of QGANs as claimed in Theorem 2.

The construction details of \( U(\gamma) \) and \( U(\theta) \) are illustrated in Fig. 7. Particularly, the circuit layout of \( U(z) \) and the \( l \)-th layer of \( U(\theta) \) is the same. Mathematically, \( U(z) = U_E(\gamma_1)(\otimes_{i=1}^{3} U(\gamma_i)) \), where \( U(\gamma_i) \) = \( R_Z(z_i) R_Y(z_i) R_Z(z_i) R_Y(z_i), \forall i \in [3] \) and \( U_E(\gamma_i) = (I_2 \otimes CR_Y(z_i))(CR_Y(z_i) \otimes I_2) \) refers to the entanglement layer. Similarly, for the \( l \)-th layer of \( U(\theta) \), its mathematical expression is \( \bar{U}_l(\theta) = U_E(\gamma_1)(\otimes_{i=1}^{3} U(\gamma_i)) \), where \( U(\gamma_i) = R_Z(\theta_3) R_Y(\theta_2) R_Z(\theta_2) R_Y(\theta_1), \forall i \in [3] \). When \( l \) is odd, the entanglement layer takes the form \( U_E(\gamma_i) = (I_2 \otimes CR_Y(\theta_2))(CR_Y(\theta_1) \otimes I_2) \); otherwise, its implementation is shown in the lower right panel of Fig. 7.

The optimization of the modified style-QGAN follows an iterative manner. At each iteration, a classical optimizer leverages the batch gradient descent method to update the trainable parameters \( \theta \) minimizing MMD loss. After \( T \)
FIG. 6: The approximated GHZ state with the varied number of qubits. The label follows the same meaning explained in Fig. 3.

FIG. 7: The implementation detail of the modified style-QGANs. The circuit architecture of $U(z)$ and the $l$-th layer of $\hat{U}(\theta)$ is identical, which is depicted in the upper right panel. The lower panel plots the construction of the entanglement layer $U_{E}(\gamma^l)$.

iterations, the optimized parameters are output as the estimation of the optimal results. The Pseudo code of the
Algorithm 1: The modified style-QGAN

| Data: | Training set \( \{ \mathbf{y}^{(i)} \}_{i=1}^{m} \), number of examples \( n \), learning rate \( \eta \), iterations \( T \), MMD loss; |
|-------|----------------------------------------------------------|
| Result: | Output the optimized parameters. |

1. Randomly divide \( \{ \mathbf{y}^{(i)} \}_{i=1}^{m} \) into \( m_{\text{mini}} \) mini batches with batch size \( b \);
2. Initialize parameters \( \theta \);
3. while \( T > 0 \) do
   4. Regenerate noise inputs \( \{ \mathbf{z}^{(i)} \}_{i=1}^{n} \) every \( r \) iterations;
   5. for \( j \leftarrow 1, m_{\text{mini}} \) do
      6. Generate \( \{ \mathbf{x}^{(i)} \}_{i=1}^{n} \) with \( \mathbf{x}^{(i)} = G_{\theta}(\mathbf{z}^{(i)}) \);
      7. Compute the \( b' \)th minibatch’s gradient \( \nabla \text{MMD}^2(P_{\theta} || Q_{b'}) \);
      8. \( \theta \leftarrow \theta - \eta \nabla \text{MMD}^2(P_{\theta} || Q_{b'}) \);
   9. end
10. \( T \leftarrow T - 1 \);
11. end

**Code availability.** The source code for conducting all numerical simulations will be available in Github repository [https://github.com/yuxuan-du/QGLM-Theory](https://github.com/yuxuan-du/QGLM-Theory).

**Simulation results.** Here we examine how the number of examples \( m \) and the number of trainable gates \( N_g \) effect the generalization of QGANs. The experimental setup is identical to those introduced in the main text. To attain a varied number of \( N_g \), the circuit depth of Ansatz in Fig. 3(a) is set as \( L = 2, 4, 6, 8 \). Other hyper-parameters are fixed with \( T = 800, n = m = 5000 \), batch size \( b = 64 \). We repeat each setting with 5 times to collect the simulation results.

**Effect of the number of examples.** Let us first focus on the setting \( L = 2 \) and \( m = 5000 \). In conjunction with the simulation results in the main text (i.e., \( L = 2, n = 1000, 2000 \), the simulation results in Fig. 8 indicate that an increased number of \( n \) and \( m \) contribute to a better generalization property. Specifically, at \( t = 120 \), the averaged empirical MMD loss \( \text{MMD}_{\text{emp}}(P_{\theta}) \) of QGANs is 0.0086, which is comparable with other settings discussed in the main text. The averaged expected MMD loss is 0.0045, which is similar to the setting with \( m = 200 \). In other words, when the number of examples \( m \) and \( n \) exceeds a certain threshold, the generalization error of QGANs is dominated by other factors instead of \( m \) and \( n \).

**Effect of the number of trainable gates.** We next study how the number of trainable gates affects the generalization error of QGANs. Following the structure of the employed Ansatz shown in Fig. 3(a), varying the number of trainable gates amounts to varying the number of blocks \( L_1 \). The results of QGANs with varied \( L_1 \) are illustrated in Fig. 8. For all setting of QGANs, their empirical MMD loss fast converges after 40 iterations. Nevertheless, their expected MMD loss is distinct, where a larger \( L_1 \) (or equivalently, a larger number of trainable gates \( N_g \)) implies a higher expected MMD loss and leads to a worse generalization. These observations accord with the result of Theorem 2 in the sense that an Ansatz with the overwhelming expressivity may incur a poor generalization ability of QGANs.

**SM I: Implications of Theorem 2 from the perspective of potential advantages**

Here we explore how the results of Theorem 2 contribute to see potential advantages of QGANs. To do so, we first theoretically formalize the task of parameterized Hamiltonian learning and prove this task is computationally hard for
classical computers. Then we conduct numerical simulations to apply QGANs to tackle parameterized Hamiltonian learning problems.

1. Theoretical analysis

Let us first introduce the parameterized Hamiltonian learning problem. Define an $N$-qubit parameterized Hamiltonian as $H(a)$, where $a$ refers to the interaction parameter sampled from a prior continuous distribution $\mathbb{D}$. For instance, the parameter $a$ can be specified to the strength of the transverse magnetic field and $\mathbb{D}$ can be set as the uniform distribution. Define $|\phi(a)\rangle$ as the ground state of $H(a)$. The aim of the parameterized Hamiltonian learning is using $m$ training samples $\{a^{(i)}, |\phi(a^{(i)})\rangle\}_{i=1}^{m}$ to approximate the distribution of the ground states for $H(a)$ with $a \sim \mathbb{D}$, i.e., $|\phi(a)\rangle \sim \mathbb{Q}$. If a trained QGLM can well approximate $\mathbb{Q}$, then it can prepare the ground state of $H(a')$ for an unseen parameter $a' \sim \mathbb{D}$. This property may contribute to investigating many crucial behaviors in condensed-matter systems.

We note that using QGLM instead of GLM to approximate $\mathbb{Q}$ allows certain computational advantages, warranted by the following lemma.

**Lemma 11.** Suppose that $\mathbb{Q}$ refers to the distribution of the ground states for parameterized Hamiltonians $H(a)$ with $a \sim \mathbb{D}$. Under the quantum threshold assumption, there exists a distribution $\mathbb{Q}$ that can be efficiently represented by QGLMs but is computationally hard for GLMs.

The results of Lemma 11 indicate the superiority of QGLMs over GLMs, i.e., $\mathbb{Q} \in \mathbb{P}^{\mathbb{Q}}_0$ and $\mathbb{Q} /\not\in \mathbb{P}^{\mathbb{C}}_0$. In conjunction this salient observation with the definition of generalization, we can conclude that when the number of parameters scales with $O(\text{poly}(N))$, there may exist a certain kernel leading to $\inf_{\theta \in \Theta} \text{MMD}^2(\mathbb{P}_\theta \| \mathbb{Q}) \rightarrow 0$ and $\inf_{\theta \in \Theta} \text{MMD}^2(\mathbb{P}_\theta \| \mathbb{Q}) > 0$. This separation is crucial in quantum machine learning. In particular, although both the generalization error $\mathcal{R}$ and $9^{\mathcal{R}}$ are continuously decreased by increasing $n$ and $m$, the separated expressive power between GLMs and QGMLs means that the MMD loss for GLMs can not converge to zero and fail to exactly learn the target distribution $\mathbb{Q}$.

2. Proof of Lemma 11

The proof of Lemma 11 is established on the results of quantum random circuits, which is widely believed to be classically computationally hard and in turn can be used to demonstrate quantum advantages on NISQ devices [113, 114]. The construction of a random quantum circuit is as follows. Denote $\mathbb{H}(N, s)$ as the distribution over the quantum circuit $\mathcal{C}$ under 2D-lattice $(\sqrt{N} \times \sqrt{N})$ structure, where $\mathcal{C}$ is composed of $s$ two-qubit gates, each of them is drawn from the 2-qubit Haar random distribution, and $s$ is required to be greater than the number of qubits $N$. For simplicity, here we choose $s = 2N^2$ to guarantee the hardness of simulating the distribution $\mathbb{H}(N, s)$. The operating rule for the $i$-th quantum gate satisfies:

- If $i \leq N$, the first qubit of the $i$-th gate is specified to be the $i$-th qubit and the second is selected randomly from its neighbors;
- If $i > N$, the first qubit is randomly selected from $\{1, 2, ..., N\}$ and randomly select the second qubit from its neighbors.

Following the same routine, Ref. [115] proposed a Heavy Output Generation (HOG) problem detailed below to separate the power between classical and quantum computers on the distribution of the output quantum state after performing a circuit $\mathcal{C}$ sampled from $\mathbb{H}(N, s)$ on the initial state $|0^N\rangle$.

**Definition 4 (HOG, [115]).** Given a random quantum circuit $\mathcal{C} \sim \mathbb{H}(N, s)$ for $s \geq N^2$, generate $k$ binary strings $z_1, z_2, \cdots, z_k$ in $\{0, 1\}^N$ such that at least $2/3$ fraction of $z_i$’s are greater than the median of all probabilities of the circuit outputs $\mathcal{C} |0^N\rangle$.

Concisely, under the quantum threshold assumption, there do not exist classical samples that can spoof the $k$ samples output by a random quantum circuit with success probability at least 0.99 [115]. In addition, they prove that quantum computers can solve the HOG problem with high success probability. For completeness, we introduce the quantum threshold assumption as follows.

**Assumption 1** (quantum threshold assumption, [115]). There is no polynomial-time classical algorithm that takes a random quantum circuit $\mathcal{C}$ as input with $s \geq N^2$ and decides whether $0^N$ is greater than the median of all probabilities of $\mathcal{C}$ with success probability $1/2 + \Omega(2^{-N})$. 
To achieve this goal, we connect the core idea of the proof is to show that there exists a ground state output by QGANs and the exact ground state is above 0.97. Moreover, when the fidelity with the exact ground states.

Proof of Lemma 11. The core idea of the proof is to show that there exists a ground state $|\phi(\theta)\rangle$ of a Hamiltonian $H(a)$, which can be efficiently prepared by quantum computers but is computationally hard for classical algorithms. To achieve this goal, we connect $|\phi(\theta)\rangle$ with the output state of random quantum circuits.

With the quantum threshold assumption, Aaronson and Chen [115] proved that there exists a quantum state $|\Phi\rangle$ generated from a random circuit $C$ sampling from $\mathbb{H}(N,2N^2)$ such that HOG problem on this instance is classically hard, with success probability at least 0.99. Conversely, $|\Phi\rangle$ can be efficiently prepared by the parameterized quantum circuit $\hat{U}(\theta^*)$ whose topology is identical to $C$. Moreover, due to the fact that quantum adiabatic algorithms with 2-local Hamiltonians can implement universal quantum computational tasks [116], the quantum state $|\Phi\rangle \equiv |\phi(a^*)\rangle = \hat{U}(\theta^*) |0\rangle^\otimes N$ must correspond to the ground state of a certain Hamiltonian $H(a^*)$.

We now leverage the above result to design a parameterized Hamiltonian learning task that separates the power of classical and quantum machines. In particular, we restrict the target distribution $Q$ as the delta distribution, where the probability of sampling $|\Phi\rangle \equiv |\phi(a^*)\rangle = \hat{U}(\theta^*) |0\rangle^\otimes N$ equals to one and the probability of sampling other ground states of $H(a)'$ with $a' \neq a^*$ and $a' \sim D$ is zero. In this way, the Hamiltonian learning task is reduced to using QGLM or GLM to prepare the quantum state $|\Phi\rangle$. This task can be efficiently achieved by QGML but is computationally hard for GLMs.

3. Numerical simulations

We apply QGANs introduced in the main text to study the parameterized Hamiltonian learning problem formulated in Section 11. In particular, the parameterized Hamiltonian is specified as the XXZ spin chain, i.e.,

$$H(a) = \sum_{i=1}^{N} (X_iX_{i+1} + Y_iY_{i+1} + aZ_iZ_{i+1}) + \eta \sum_{i=1}^{N} Z_i.$$  \hspace{1cm} (11)

In all numerical simulations, we set $N = 2$ and $\eta = 0.25$. The distribution $D$ for the parameter $a$ is uniform ranging from $-0.2$ to $0.2$. In the preprocessing stage, to collect $m$ referenced samples $\{a^{(j)}, |\phi(a^{(j)})\rangle\}_{j=1}^{m}$, we first uniformly sample $\{a^{(j)}\}_{j=1}^{m}$ points from $D$ and calculate the corresponding eigenstates of $\{H(a^{(j)})\}_{j=1}^{m}$ via the exact diagonalization.

The setup of QGAN is as follows. The prior distribution $P_Z$ is set as $D$. The encoding unitary is $U(a) = \text{CNOT}(\text{RZ}(a)\text{RY}(a)) \otimes (\text{RZ}(a)\text{RY}(a))$. The hardware-efficient Ansatz is used to implement $U(\theta) = \prod_{l=1}^{L} U_l(\theta)$ with $U_l(\theta) = \text{CNOT}(\text{RZ}(\theta_{l,1})\text{RY}(\theta_{l,2})) \otimes (\text{RZ}(\theta_{l,3})\text{RY}(\theta_{l,4}))$. The number of blocks is $L = 4$. The number of training and referenced examples is set as $n = m = 9$. The Adagrad optimizer is used to update $\theta$. The total number of iterations is set as $T = 80$. We employ 5 different random seeds to collect the statistical results. To evaluate the performance of the trained QGAN, we apply it to generate in total 41 ground states of $H(a)$ with $a \in [-0.2, 0.2]$ and compute the fidelity with the exact ground states.

The simulation results are shown in Fig. 9. Specifically, for all settings of $a$, the fidelity between the approximated ground state output by QGANs and the exact ground state is above 0.97. Moreover, when $a \in [-0.06, 0.06]$, the

![Simulation results of QGANs in parameterized Hamiltonian learning.](image-url)
fidelity is near to 1. The right panel depicts the estimated ground energy using the output of QGANs, where the maximum estimation error is 0.125 when $a = 0.2$. These observations verify the ability of QGANs in estimating the ground states of parameterized Hamiltonians.