npj | quantum information

Article

Published in partnership with The University of New South Wales

https://doi.org/10.1038/s41534-024-00902-0

Trainability barriers and opportunities in quantum generative modeling

Check for updates

Manuel S. Rudolph[1,5], Sacha Lerch[1,5], Supanut Thanasilp[1,2,5], Oriel Kiss[3,4], Oxana Shaya[1], Sofia Vallecorsa[3], Michele Grossi[3] & Zoë Holmes[1]

Quantum generative models provide inherently efficient sampling strategies and thus show promise for achieving an advantage using quantum hardware. In this work, we investigate the barriers to the trainability of quantum generative models posed by barren plateaus and exponential loss concentration. We explore the interplay between explicit and implicit models and losses, and show that using quantum generative models with explicit losses such as the KL divergence leads to a new flavor of barren plateaus. In contrast, the implicit Maximum Mean Discrepancy loss can be viewed as the expectation value of an observable that is either low-bodied and provably trainable, or global and untrainable depending on the choice of kernel. In parallel, we find that solely low-bodied implicit losses cannot in general distinguish high-order correlations in the target data, while some quantum loss estimation strategies can. We validate our findings by comparing different loss functions for modeling data from High-Energy-Physics.

The advent of quantum computing has opened up new avenues for solving classically intractable problems[1–4]. Naturally, researchers gravitate towards finding the first high-value applications that could be tackled with near- and mid-term quantum devices[5]. This includes not only speed-ups[3,6–8] but potentially superior memory efficiency[9] or concrete qualitative improvements[10,11]. Quantum machine learning (QML) is one of the domains that attracts this attention[2]. Quantum systems, in being inherently probabilistic, are particularly well suited to generative modeling tasks[12]. Generative models aim to learn the underlying distribution of a dataset and thereby provide a means of generating new data samples that are similar to the original data. As well as providing a naturally efficient means of generating samples, quantum generative models can provably encode probability distributions that are out of reach for classical models[13–15], and have been proposed for various applications, such as handwritten digits[16], finance[17], or High-Energy-Physics (HEP)[18,19].

Despite the excitement surrounding the potential of generative QML, there remain substantial questions concerning its scalability. This is nontrivial to assess since implementations are constrained by hardware limitations to small-scale proof-of-principle problems[16,17,20–22]. Thus analytic results are essential to guide the successful development of this field. Of particular concern is the growing body of literature on cost function concentration and barren plateaus (BPs)[23–30], where loss function values can exponentially concentrate around a fixed value and loss gradients vanish exponentially with growing problem size. This phenomenon, which exponentially increases the resources required for training, originates from different sources[23,25,31–39], and has been studied in a number of architectures[23,25,30,40–45] as well as classes of cost function[32,37,40]. However, its impact on quantum generative modeling thus far has, except for the odd notable exception[46], and very recent developments[47], been largely overlooked.

In this work, we provide a thorough study of trainability barriers and opportunities in quantum generative modeling. Critical to our analysis is the distinction between explicit and implicit models and losses. Explicit models provide efficient access directly to the model probabilities, whereas implicit models only provide samples drawn from their distribution[48]. Quantum circuit Born machines (QCBMs)[49], the focus of this work, encode a probability distribution in an n-qubit pure state and thus are a paradigmatic example of an implicit model. Mirroring the capabilities of the models, explicit losses are those that are formulated explicitly in terms of the model and target probabilities, whereas implicit losses compare samples from the model and the training distribution. The most commonly used explicit loss for quantum generative models is the Kullbach-Leibler (KL) divergence[50]. Other examples include the Jensen-Shannon divergence (JSD), the total variation distance (TVD) and the classical fidelity. The Maximum Mean Discrepancy (MMD)[51] on the other hand is one of the leading examples of an implicit loss.

Here we argue that the tension between using an implicit generative model (providing only samples) with an explicit loss (requiring access to probabilities) leads to a new flavor of BP. This result disqualifies all beforementioned explicit losses, and crucially the KL divergence, for efficient

[1]Institute of Physics, Ecole Polytechnique Fédérale de Lausanne (EPFL), Lausanne, Switzerland. [2]Chula Intelligent and Complex Systems, Department of Physics, Faculty of Science, Chulalongkorn University, Bangkok, Thailand. [3]European Organization for Nuclear Research (CERN), Geneva, Switzerland. [4]Department of Nuclear and Particle Physics, University of Geneva, Geneva, Switzerland. [5]These authors contributed equally: Manuel S. Rudolph, Sacha Lerch, Supanut Thanasilp. ✉e-mail: supanut.thanasilp@gmail.com; zoe.holmes@epfl.ch

Table 1 | Summary of our main results

Circuit depth	Explicit loss (pairwise)		Implicit loss (MMD)
	Conventional strategy	Quantum strategy	
Product	**No** (Corollary 3)	**Yes** (Local Quantum Fidelity[32])	**Yes** ($\sigma \in \Theta(n)$, Theorem 2)
Shallow			**Yes** ($\sigma \in \Theta(n)$, Theorem 3)
Deep		**No**[23,31]	**No**[23,31]

This table summarizes our key analytical results on the trainability of different loss functions in quantum generative modeling tasks. Without a strong inductive bias, pairwise explicit losses are untrainable for all circuit depths with the conventional sampling strategy. A quantum strategy could be utilized to efficiently estimate the local quantum fidelity, Eq. (52), which is trainable for a shallow-depth circuit. The MMD using a classical Gaussian kernel with a linearly-scaled bandwidth ($\sigma \in \Theta(n)$) is expected to be trainable for a shallow-depth circuits. Note that "Yes" here indicates the existence of regimes with trainability guarantees—it does not preclude untrainable regimes including, for example, the use of global quantum fidelity or the MMD with a fixed bandwidth.

training of QCBMs without a strong inductive bias towards the target distribution. In contrast, the MMD as an implicit loss exhibits more nuanced behavior and can be either trainable or untrainable. By viewing the classical MMD loss as the expectation value of a quantum observable, we show that varying the bandwidth parameter of a Gaussian kernel interpolates the MMD loss between a loss composed of predominantly global terms and one composed of low-bodied terms with either exponentially or polynomially decaying loss variances in the number of qubits. In particular, we derive a polynomial lower bound on the loss for a wide family of different classes of structured and unstructured models that depends only on the effective entanglement light cone of the circuit. These results are summarized in Table 1.

In parallel, we provide insights into how the globality of a generative loss affects the types of correlations in a dataset that can reliably be learned. In particular, we show that a k-bodied loss (see Fig. 5) cannot distinguish between distributions that agree on all k-marginals but disagree about higher-order correlations. Hence we argue that in the context of quantum generative modeling, it is advantageous to train on *full-bodied* losses, that is losses containing both low and high-bodied terms, rather than the purely local losses advocated elsewhere in QML. The MMD is then a promising candidate choice for the training of QCBMs as its bodyness can be controlled via the bandwidth parameter.

We additionally expand the pool of viable loss functions by proposing a new local quantum fidelity (LQF)-type loss which leverages what we call a quantum strategy for evaluating losses. This is to be contrasted with the conventional measurement strategy which simply uses samples from the model distribution in the computational basis. We provide an efficient training protocol using the LQF loss with provable trainability guarantees.

Finally, we support our analysis with a comparison of the performance of the KL divergence, MMD, and LQF losses for modeling HEP data. Specifically, we consider electron energy depositions in the electromagnetic calorimeter (ECAL) part of detectors involved in a typical proton-proton collision experiment at the LHC. We learn to generate hits in the detector as black and white images of various sizes, with up to 16 qubits. We confirm that the properly-tuned MMD and the LQF losses remain trainable using a restrictive shot budget, while training with the KL divergence becomes increasingly futile.

Results

Framework

The goal of generative modeling is to use samples from a target distribution $p(\boldsymbol{x})$ to learn a model of $p(\boldsymbol{x})$ which can be used to generate new samples. More concretely, as sketched in Fig. 1, a generative model takes as input a training dataset $\tilde{P}$ consisting of $M = |\tilde{P}|$ samples drawn from the target distribution $p(\boldsymbol{x})$. This training set can be used to construct the empirical probability distribution $\tilde{p}(\boldsymbol{x})$ for all samples $\boldsymbol{x} \in \tilde{P}$. The training dataset, or the training distribution, is then used to train the variational parameters $\boldsymbol{\theta}$ of a parameterized probability distribution $q_{\boldsymbol{\theta}}(\boldsymbol{x})$. If successful, the output of the algorithm is a set of optimized parameters $\boldsymbol{\theta}_{\mathbf{opt}}$ such that the trained model $q_{\boldsymbol{\theta}_{\rm opt}}(\boldsymbol{x})$ well-approximates the unknown target distribution $p(\boldsymbol{x})$. The trained model $q_{\boldsymbol{\theta}_{\rm opt}}(\boldsymbol{x})$ can then be used to generate new and previously unseen data. For compactness, we use the notation p and $q_{\boldsymbol{\theta}}$ to denote the target and model distributions respectively.

The process of training requires a *loss function* $\mathcal{L}(\boldsymbol{\theta})$ which estimates the distance between the model distribution $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ and the training distribution $\tilde{p}(\boldsymbol{x})$. For typical choices in loss function (detailed further in Section "Loss functions"), the loss is minimized when the model parameters $\boldsymbol{\theta}$ are tuned such that the model distribution perfectly matches the empirical distribution obtained from the training data. That is, $\mathcal{L}(\boldsymbol{\theta}) = 0$ if and only if $q_{\boldsymbol{\theta}}(\boldsymbol{x}) = \tilde{p}(\boldsymbol{x})$ over the entire data space $\mathcal{X}$. Thus, by perfectly minimizing the loss, one perfectly learns the empirical distribution $\tilde{p}(\boldsymbol{x})$ but not the true target distribution $p(\boldsymbol{x})$. This scenario is commonly called *overfitting* (In contrast, discriminative machine learning models can be perfectly minimized on the training data and not be overfitted.). To allow for *generalization*[52], whereby the model can generate novel data with similar properties to the training data, one seeks to significantly reduce (but not perfectly minimize) the training loss. While generalization is the end-all goal of generative models, it is not the focus of this work. Instead, we focus on the training component of the generative framework, as failing to train also prohibits generalization.

Quantum circuit models. One prototypical quantum generative model is the *quantum circuit Born machine* (QCBM)[13,49,53,54]. Owing its name to the Born rule of quantum mechanics, a QCBM encodes a probability distribution over discrete data (here bitstrings) in an n-qubit pure quantum state that depends on a parameterized unitary $U(\boldsymbol{\theta})$,

$$q_{\boldsymbol{\theta}}(\boldsymbol{x}) = |\langle \boldsymbol{x}|U(\boldsymbol{\theta})|\boldsymbol{0}\rangle|^2. \tag{1}$$

Here $|\boldsymbol{x}\rangle$ is a computational basis state corresponding to a bitstring $\boldsymbol{x}$ and, without loss of generality, an initial state can be chosen as $|\boldsymbol{0}\rangle = |0\rangle^{\otimes n}$. We note that estimating $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ is equivalent to finding the expectation value of a global projector $|\boldsymbol{x}\rangle\langle\boldsymbol{x}|$. More fundamentally, QCBMs enable the encoded distribution to be efficiently sampled simply by measuring in a chosen computational basis. That is, every measurement of the quantum state provides an unbiased sample from the encoded distribution (in an ideal noise-free setting). This is a very desirable property in generative models that many (classical) generative models do not share with the QCBM. Sampling techniques for classical generative models are often unreliable and may break down for certain distributions, as is the case for *restricted Boltzmann machines* (RBMs)[55,56]. Born machines represent an effort to create a powerful, flexible, and efficient generative model for classical discrete data, and as well as numerous 'standard' digital quantum implemenations[16,17,20–22], they have been widely implemented using tensor networks[57–60], continuous variable hardware[61], in a conditional setting[18,62], with non-linearities[63].

An important, but rather subtle, distinction in generative modeling is that between *explicit and implicit generative models*[48,64]. Explicit generative models are ones that allow efficient access to the model probability $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ for any data sample $\boldsymbol{x}$. Here, "efficient" means that the probabilities can be computed in a time and memory that are polynomial in the size of the data samples, i.e., $\mathcal{O}(\text{poly}(n))$ resources. Explicit (classical) generative models include for example auto-regressive models[65], RNNs[66], tensor networks without loops (which includes tensor network Born machines)[57,58], and

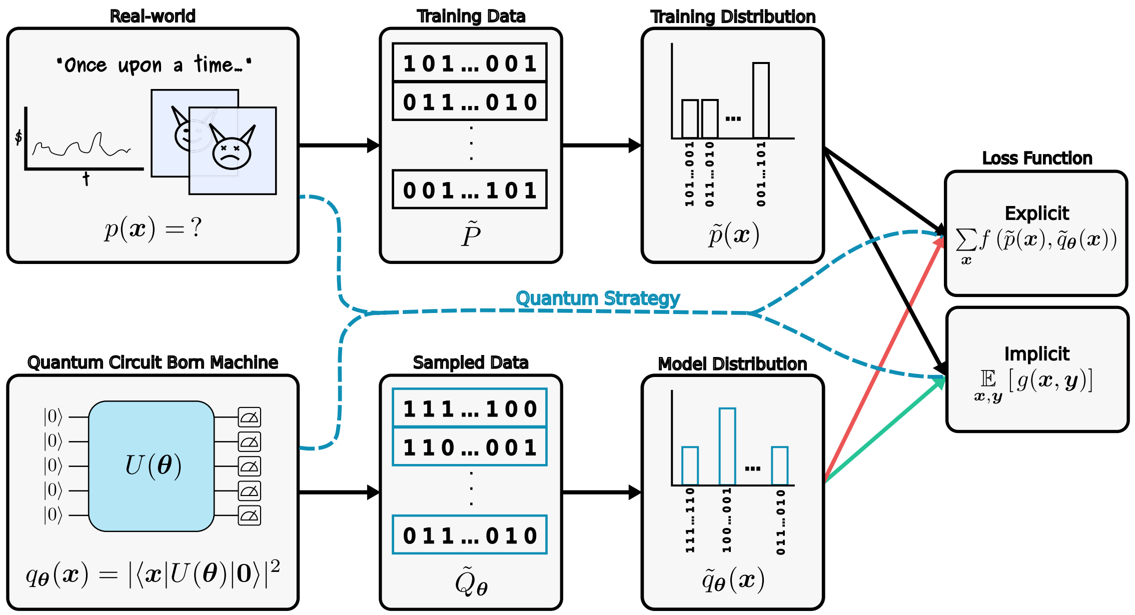
Real-world
'Once upon a time..'
$
t
$p(\boldsymbol{x}) = ?$
Training Data
101...001
011...010
001...101
$\tilde{P}$
Training Distribution
101...001
011...010
001...101
$\tilde{p}(\boldsymbol{x})$
Loss Function
Explicit
$\sum_{\boldsymbol{x}} f(\tilde{p}(\boldsymbol{x}), \tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x}))$
Implicit
$\mathbb{E}_{\boldsymbol{x},\boldsymbol{y}}[g(\boldsymbol{x},\boldsymbol{y})]$
Quantum Strategy
Quantum Circuit Born Machine
$|0\rangle$
$|0\rangle$
$|0\rangle$
$|0\rangle$
$|0\rangle$
$U(\boldsymbol{\theta})$
$q_{\boldsymbol{\theta}}(\boldsymbol{x}) = |\langle \boldsymbol{x}|U(\boldsymbol{\theta})|\boldsymbol{0}\rangle|^2$
Sampled Data
111...100
110...001
011...010
$\tilde{Q}_{\boldsymbol{\theta}}$
Model Distribution
111...110
100...001
011...010
$\tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x})$

Fig. 1 | The generative modeling framework using quantum circuit Born machines. Given a training dataset $\tilde{P}$ with distribution $\tilde{p}(\boldsymbol{x})$ over discrete data samples $\boldsymbol{x}$, the goal of a QCBM is to learn a distribution $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ which models the real-world distribution $p(\boldsymbol{x})$ from which the training data itself was sampled. This is done by tuning the parameters $\boldsymbol{\theta}$ of a parametrized quantum circuit such that the QCBM minimizes a loss function that estimates the distance between the model and the training distribution. The QCBM is an implicit model and can thus in general not be paired with an explicit loss function, but it may be trainable using an implicit loss. In contrast to the conventional loss estimation strategy (solid lines) of generating a set of samples $\tilde{Q}_{\boldsymbol{\theta}}$ and forming an empirical distribution $\tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x})$, strategies that are `more quantum' (dashed lines) can be employed with the aim of allowing QCBMs to be trained with loss functions which conventionally appear explicit.

many forms of density estimators. In contrast, *implicit* models lack this property and instead offer efficient access to samples from $q_{\boldsymbol{\theta}}(\boldsymbol{x})$, which some forms of explicit models may struggle with. A popular example of an implicit generative model is *Generative Adversarial Networks* (GANs)[67] that leverage an implicit training scheme to learn powerful generators.

In the case of QCBMs implemented on quantum devices, it becomes evident that we do not have (efficient) explicit access to $q_{\boldsymbol{\theta}}(\boldsymbol{x})$, but only to samples of the distribution in the computational basis. Consequently, QCBMs can be classified as implicit generative models. In this work, we study the trainability issues that QCBMs suffer from as a result.

Loss functions. Similarly to the distinction between explicit and implicit generative models, we draw a distinction between *explicit and implicit loss functions*. In broad terms, explicit losses are those that can only be formulated explicitly in terms of the target and model *probabilities*, whereas implicit losses are those that can be formulated in terms of an average over model and training data *samples*. This distinction at the level of loss functions thus mirrors the capabilities and limitations of explicit and implicit generative models.

More concretely, we define an *explicit loss* as a loss function $\mathcal{L}$ that can be written solely as a function of the probabilities of the target and model distributions, without any dependence on the data itself. Explicit losses thus take the general form

$$\mathcal{L}_{\text{expl}}(\boldsymbol{\theta}) := \sum_{\boldsymbol{x}_1 \ldots \boldsymbol{x}_r} f\left(p(\boldsymbol{x}_1), \ldots, p(\boldsymbol{x}_r), q_{\boldsymbol{\theta}}(\boldsymbol{x}_1), \ldots, q_{\boldsymbol{\theta}}(\boldsymbol{x}_r)\right), \qquad (2)$$

where $f(\cdot)$ is a function that depends on the target probabilities $p(\boldsymbol{x}_i)$ and model probabilities $q_{\boldsymbol{\theta}}(\boldsymbol{x}_i)$ for data variables $\boldsymbol{x}_i \in \mathcal{X}$ with $i = 1, \ldots, r$. For this loss to be useful, the function f should be chosen such that it measures the distance between the probability distributions p and $q_{\boldsymbol{\theta}}$. Crucially, the function f does not take the data values $\boldsymbol{x}$ themselves as arguments.

While in full generality explicit losses could compare multiple copies of the target and model probabilities (i.e., we can have $r > 1$), in practice, they usually take the simpler form

$$\mathcal{L}(\boldsymbol{\theta}) = \sum_{\boldsymbol{x} \in \mathcal{X}} f(p(\boldsymbol{x}), q_{\boldsymbol{\theta}}(\boldsymbol{x})). \qquad (3)$$

We call such losses *pairwise explicit losses* since they compare the model and target probabilities on the same data samples, or in our case, bitstrings. The pairwise explicit loss covers all so-called f-divergences[68], including the commonly encountered KL divergence (KLD)[69],

$$\mathcal{L}^{\text{KLD}}(\boldsymbol{\theta}) = \sum_{\boldsymbol{x} \in \mathcal{X}} p(\boldsymbol{x}) \log\left(\frac{p(\boldsymbol{x})}{q_{\boldsymbol{\theta}}(\boldsymbol{x})}\right), \qquad (4)$$

the reverse-KLD,

$$\mathcal{L}^{\text{rev}-\text{KLD}}(\boldsymbol{\theta}) = \sum_{\boldsymbol{x} \in \mathcal{X}} q_{\boldsymbol{\theta}}(\boldsymbol{x}) \log\left(\frac{q_{\boldsymbol{\theta}}(\boldsymbol{x})}{p(\boldsymbol{x})}\right), \qquad (5)$$

the Jensen-Shannon divergence (JSD)[70],

$$\begin{aligned}\mathcal{L}^{\text{JSD}}(\boldsymbol{\theta}) = \sum_{\boldsymbol{x} \in \mathcal{X}} \Bigg[& p(\boldsymbol{x}) \log\left(\frac{p(\boldsymbol{x})}{p(\boldsymbol{x}) + q_{\boldsymbol{\theta}}(\boldsymbol{x})}\right) \\ & + q_{\boldsymbol{\theta}}(\boldsymbol{x}) \log\left(\frac{q_{\boldsymbol{\theta}}(\boldsymbol{x})}{p(\boldsymbol{x}) + q_{\boldsymbol{\theta}}(\boldsymbol{x})}\right)\Bigg],\end{aligned} \qquad (6)$$

and the total variation distance (TVD),

$$\mathcal{L}^{\text{TVD}}(\boldsymbol{\theta}) = \sum_{\boldsymbol{x} \in \mathcal{X}} |p(\boldsymbol{x}) - q_{\boldsymbol{\theta}}(\boldsymbol{x})|. \qquad (7)$$

Another example of loss function that can be written in this form is the classical fidelity,

$$\mathcal{L}^{\text{CF}}(\boldsymbol{\theta}) = 1 - \sum_{\boldsymbol{x} \in \mathcal{X}} \sqrt{p(\boldsymbol{x}) q_{\boldsymbol{\theta}}(\boldsymbol{x})}. \qquad (8)$$

Notably, any non-data-dependent post-processing of an explicit loss retains its explicit character. Thus, any non-data-dependent function of an explicit loss (Eq. (2)) may also be considered an explicit loss. For example, the Rényi divergence[71]

$$\mathcal{L}_{R,\alpha}(\boldsymbol{\theta}) = \frac{1}{\alpha - 1}\log\left(\sum_{\boldsymbol{x}} \frac{p^{\alpha}(\boldsymbol{x})}{q_{\boldsymbol{\theta}}^{\alpha-1}(\boldsymbol{x})}\right), \tag{9}$$

with $0 < \alpha < \infty$ and $\alpha \neq 1$ can be classified as an explicit loss function.

On the other hand, we define an *implicit loss* as one that can be written as an average over samples drawn from the target and model distributions. That is, an implicit loss function can be expressed as

$$\mathcal{L}_{\mathrm{impl}}(\boldsymbol{\theta}) := \mathbb{E}_{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_r \sim \{p, q_{\boldsymbol{\theta}}\}}\, g(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_r), \tag{10}$$

where $g(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_r)$ is some function that depends on the data (but not probabilities), and the expectation is taken over data variables $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_r$ sampled either from the data distribution p or the model distribution $q_{\boldsymbol{\theta}}$.

As a key example of an implicit loss, we focus on the commonly used *Maximum Mean Discrepancy* (MMD)[51] loss. The MMD takes the form

$$\begin{aligned}\mathcal{L}_{\mathrm{MMD}}(\boldsymbol{\theta}) = \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim q_{\boldsymbol{\theta}}}[K(\boldsymbol{x},\boldsymbol{y})] - 2\mathbb{E}_{\boldsymbol{x}\sim q_{\boldsymbol{\theta}},\boldsymbol{y}\sim p}[K(\boldsymbol{x},\boldsymbol{y})] \\ + \mathbb{E}_{\boldsymbol{x},\boldsymbol{y}\sim p}[K(\boldsymbol{x},\boldsymbol{y})],\end{aligned} \tag{11}$$

where $K(\boldsymbol{x}, \boldsymbol{y})$ is a freely chosen kernel function. We consider the popular choice of a classical *Gaussian kernel*, which is defined as

$$K_{\sigma}(\boldsymbol{x},\boldsymbol{y}) = e^{-\frac{\|\boldsymbol{x}-\boldsymbol{y}\|_2^2}{2\sigma}} = \prod_{i=1}^{n} e^{-\frac{(x_i-y_i)^2}{2\sigma}}. \tag{12}$$

Here, $\|.\|_2$ is the 2-norm, $\sigma > 0$ is the so-called *bandwidth* parameter, and x_i, y_i are the values of bit i in bitstring $\boldsymbol{x}$, $\boldsymbol{y}$, respectively. This kernel in effect provides a continuous measure of the distance between target and model bitstrings.

Interestingly, an implicit loss can always additionally be expressed in a form where it contains the target and model probabilities. Taking the MMD loss in Eq. (11) as a concrete example, the loss can be re-written as

$$\begin{aligned}\mathcal{L}_{\mathrm{MMD}}(\boldsymbol{\theta}) = &\sum_{\boldsymbol{x},\boldsymbol{y}\in\mathcal{X}} q_{\boldsymbol{\theta}}(\boldsymbol{x})q_{\boldsymbol{\theta}}(\boldsymbol{y})K(\boldsymbol{x},\boldsymbol{y}) \\ &-2\sum_{\boldsymbol{x},\boldsymbol{y}\in\mathcal{X}} q_{\boldsymbol{\theta}}(\boldsymbol{x})p(\boldsymbol{y})K(\boldsymbol{x},\boldsymbol{y}) \\ &+\sum_{\boldsymbol{x},\boldsymbol{y}\in\mathcal{X}} p(\boldsymbol{x})p(\boldsymbol{y})K(\boldsymbol{x},\boldsymbol{y}).\end{aligned} \tag{13}$$

However, we stress that due to the data dependence in the kernel $K(\boldsymbol{x}, \boldsymbol{y})$, the MMD loss function can in general not be classified as an explicit loss.

Nonetheless, this brings us to the subtle point that explicitness and implicitness are in fact not strictly mutually exclusive, i.e., one may be able to find a loss function that satisfies both Eq. (2) and Eq. (10) in specific cases. For example, for the MMD this occurs if the kernel is chosen to be a Kronecker delta function, $K(\boldsymbol{x}, \boldsymbol{y}) = \delta_{\boldsymbol{x}\boldsymbol{y}}$. However, such hybrid losses are very much rare edge cases, and the overwhelming majority of losses are either explicit or implicit. A more detailed discussion of the technical nuances of the explicit and implicit loss distinction is provided in Supplementary Note I.

Loss measurement strategies. Central to the trainability of quantum generative models is the measurement strategy used to estimate the loss. Here we draw a distinction between *conventional and quantum measurement strategies*. For simplicity we now restrict our discussion to implicit quantum generative models such as the QCBM.

The *conventional* measurement strategy, which can be employed by both classical and quantum implicit models, starts by collecting sample data from the target and model distributions in the bases in which the data distribution is modeled, e.g., the computational basis for the case of classical data. For an implicit loss these samples can then be directly used to evaluate the loss function in Eq. (10). For an explicit loss, this is not possible, and instead one needs to use the collected samples to recreate an empirical estimate $\tilde{q}_{\boldsymbol{\theta}}$ of the true model distributions $q_{\boldsymbol{\theta}}$.

More formally, as sketched in Fig. 1, consider the set of bitstrings $\tilde{Q}_{\boldsymbol{\theta}}$ obtained after collecting N samples from the model and the empirical model distribution $\tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x})$ constructed from these samples. Then, the statistical estimate of the pairwise explicit loss function $\tilde{\mathcal{L}}(\boldsymbol{\theta})$ in Eq. (3) can be expressed as

$$\tilde{\mathcal{L}}(\boldsymbol{\theta}) = \sum_{\boldsymbol{x}\in\mathcal{X}} f(\tilde{p}(\boldsymbol{x}), \tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x})). \tag{14}$$

Crucially, since this proxy is all we have access to, the properties of this statistical estimate are what determine the trainability of an explicit loss function when evaluated via the conventional strategy. We note that zero-estimates of the model probabilities with $\tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x}) = 0$ are often "clipped" with a small regularization parameter $\epsilon \ll 1$ in order to avoid numerical instabilities in the loss computation.

This conventional strategy is somewhat classical in the sense that after sampling is performed on the quantum model, the post processing required to compute the cost is entirely classical. However, "more quantum" measurement strategies are also possible. In this case, a quantum circuit is used to compute functions of the probabilities, potentially more directly and/or collectively.

For example, rather than computing the classical fidelity in Eq. (8) by explicitly computing the probabilities $q_{\boldsymbol{\theta}}(\boldsymbol{x})$, one could encode the target distribution in a quantum state $|\phi\rangle = \sum_{\boldsymbol{x}} \sqrt{\tilde{p}(\boldsymbol{x})}|\boldsymbol{x}\rangle$ and compute the quantum fidelity

$$\mathcal{L}_{QF}(\boldsymbol{\theta}) := 1 - |\langle\phi|\psi(\boldsymbol{\theta})\rangle|^2 \tag{15}$$

$$\sim 1 - \left|\sum_{\boldsymbol{x}} \sqrt{\tilde{p}(\boldsymbol{x})q_{\boldsymbol{\theta}}(\boldsymbol{x})}\right|^2. \tag{16}$$

Up to arbitrary global phase factors (and a mod-square) this is equivalent to the classical fidelity. However, it can be computed via coherent strategies—namely a Loschmidt echo circuit[72–75] or a SWAP test[76,77]. We note that in this case quantum generative modeling is equivalent to a state learning problem. While this expression seemingly requires the entire training dataset to be loaded into a wavefunction, we present an approach in Sec. II B 5 to estimate this cost using pairwise Hadamard tests.

More generally, it remains an open question if/when commonly encountered losses for generative modeling can be computed using quantum strategies and whether or not this brings any advantages (Beyond QCBMs, *Quantum Generative Adversarial Networks* (QGANs)[78] trained with classical discriminators[79–81] in effect use a conventional measurement strategy, whereas their variant with quantum discriminators[82] use a quantum strategy.). Nonetheless, we suggest that this is an interesting avenue for future research.

Exponential concentration and barren plateaus. For a quantum generative model to be trained successfully, the loss landscape must be sufficiently featured to enable a solution to be found. There is a growing awareness of the importance of BPs, and its sister phenomenon *exponential concentration*, for QML[23–30]. A BP is a loss landscape where the magnitudes of gradients vanish exponentially with growing problem size[23,25–29,31–37]. Closely related and equally problematic is exponential concentration where the loss is shown to concentrate with high probability to a single fixed value[24]. This, with high probability, results in poorly trained models using a polynomial number of measurement shots (regardless of the optimization method employed)[27]. More precisely, exponential concentration can be formally defined as follows.

Definition 1. (Exponential concentration). Consider a quantity $X(\boldsymbol{\alpha})$ that depends on a set of variables $\boldsymbol{\alpha}$ and can be measured from a quantum computer as the expectation of some observable. $X(\boldsymbol{\alpha})$ is said to be deterministically exponentially concentrated in the number of qubits n towards a certain fixed value μ if

$$|X(\boldsymbol{\alpha}) - \mu| \leqslant \beta \in O(1/b^n), \tag{17}$$

for some $b > 1$ and all $\boldsymbol{\alpha}$. Analogously, $X(\boldsymbol{\alpha})$ is probabilistically exponentially concentrated if

$$\mathrm{Pr}_{\boldsymbol{\alpha}}[|X(\boldsymbol{\alpha}) - \mu| \geqslant \delta] \leqslant \frac{\beta}{\delta^2}\ , \quad \beta \in O(1/b^n), \tag{18}$$

for $b > 1$. That is, the probability that $X(\boldsymbol{\alpha})$ deviates from μ by a small amount δ is exponentially small for all $\boldsymbol{\alpha}$.

A number of causes of exponential concentration and BPs have been identified including using parameterized circuits that are too expressive[23,25,31,43] or too entangling[33,34,44]. Hardware noise[35,36,83] has also been shown to exponentially flatten the loss landscapes, which strongly hinders the potential of current noisy quantum devices. The exponential concentration can also happen due to randomness in the training dataset[37–39]. In addition, there are studies on the exponential concentration in different QML models including dissipative parametrized quantum circuits[44] as well as quantum kernel-based models[30].

Finally, the choice of loss function can also induce these phenomena. Thus far, loss concentration has predominantly been studied in the context of losses of the form

$$C(\boldsymbol{\theta}) = \mathrm{Tr}[OU(\boldsymbol{\theta})\rho U(\boldsymbol{\theta})^\dagger], \tag{19}$$

where ρ is an n-qubit input state and O is a Hermitian operator. In particular, it has been shown that "global"[32] losses, i.e., those where O acts non-trivially on $\mathcal{O}(n)$ qubits, induce loss concentration even for very shallow random circuits. Conversely, local losses where O acts non-trivially on at most *log*(n) *adjacent* qubits (and more generally low-body losses where the adjacency constraint is lifted—see panel a) of (Fig. 5) have been shown to enjoy trainability guarantees[32,40] with shallow unstructured circuits. Furthermore, we note that how BPs affect parametrized quantum circuits with a non-linear loss in the discriminative QML setting has been studied in ref. 37.

Here we study exponential concentration for generative modeling tasks on classical discrete data using implicit quantum generative models, and use our insights to establish guidelines of how best to train such models. Crucially, in this generative modeling context, the fixed points of the model probabilities tend to be exponentially small and the loss function contains the sum over exponentially many terms. These two together render previously used tools not directly applicable for studying the trainability of quantum generative models.

Large gradient variances are not enough. The presence or absence of BPs is usually diagnosed by computing the variance of the loss over a given parameter distribution. Crucially this is usually computed for the *exact loss*, i.e., not including the effect of shot noise. Here we argue that this approach can fail in the context of quantum generative modeling. In particular, if one computes the variance of the KLD loss then the loss variance can be non-exponentially vanishing even for very deep circuits. However, as we will argue in this section, the KLD loss is untrainable for both deep and shallow unstructured circuits if the model is implicit (i.e., only gives efficient access to samples and not to the probabilities).

We now show that the variance of the exact KL divergence depends directly on the support of the target distribution and hence can be polynomially large. This is quantified by the following proposition which we prove in Supplementary Note III.

Proposition 1. Consider the KLD loss as defined in Eq. (4). Assume access to the exact target distribution $p(\boldsymbol{x})$ and the model distribution $q_{\boldsymbol{\theta}}(\boldsymbol{x})$. Then, we have

- For deep (Haar random) parametrized circuit $U(\boldsymbol{\theta})$, the variance of the loss scales asymptotically ($2^n \gg 1$) as

$$\mathrm{Var}_{\boldsymbol{\theta}}[\mathcal{L}^{\mathrm{KLD}}(\boldsymbol{\theta})] = \frac{\pi^2}{6}\sum_{\boldsymbol{x}} p^2(\boldsymbol{x}). \tag{20}$$

- For a random tensor product circuit $U(\boldsymbol{\theta}) = \bigotimes_{i=1}^n U_i(\theta_i)$ where $U_i(\theta_i)$ is a random single-qubit unitary, the variance of the loss scales as

$$\mathrm{Var}_{\boldsymbol{\theta}}[\mathcal{L}^{\mathrm{KLD}}(\boldsymbol{\theta})] = n - \frac{\pi^2}{6}\sum_{\boldsymbol{x},\boldsymbol{x}'} p(\boldsymbol{x})p(\boldsymbol{x}')\left\|\boldsymbol{x} - \boldsymbol{x}'\right\|_H, \tag{21}$$

where $\|\cdot\|_{\mathrm{H}}$ is a Hamming distance.

It follows that the variance of the exact KLD can be non-exponentially vanishing even for a deep circuit, where one would generally expect a BP[23], if the purity $\sum_{\boldsymbol{x}} p^2(\boldsymbol{x})$ of the target distribution is non-exponentially vanishing. For this condition to be met, all we need is that at least one probability $p(\boldsymbol{x})$ of the target distribution is non-exponentially vanishing. This is captured by the following corollary.

Corollary 1. Under the same assumption as in Proposition 1, for the target distribution, at least one probability is at least polynomially large. Then, the variance of the KLD loss function does not vanish exponentially with the system's size. That is, $\exists \boldsymbol{x}:\ p(\boldsymbol{x}) \in \Omega\left(\frac{1}{\mathrm{poly}(n)}\right)$, we have

$$\mathrm{Var}_{\boldsymbol{\theta}}[\mathcal{L}^{\mathrm{KLD}}(\boldsymbol{\theta})] \notin \mathcal{O}\left(\frac{1}{b^n}\right), \tag{22}$$

for some constant $b > 1$.

We note that any distribution with support on at most D bit strings necessarily has at least one probability that is $1/D$ large. Thus the support of a distribution lower bounds the variance of the exact KLD. This is reflected in Fig. 2. For the GHZ dataset, which has $\mathcal{O}(1)$ support, we observe a strong evidence for non-vanishing variance for all circuit depths. For linear and quadratic support datasets, the variances moderately decrease as the number of qubits increases for deep circuits.

Thus we see that for certain target probability distributions, the KLD does not exhibit a BP for *explicit* models. This suggests that quantum-inspired models that can provide direct access to probabilities (e.g., tensor network Born machines[57,58]) might be trainable with the KLD. However, current generative models running on quantum devices only provide access to samples from a distribution via measurements and, as we will argue in the next section, the large variance of the exact loss, in contrast to standard VQE-style losses, does not translate to substantial loss gradients in practise.

Trainability analysis on loss functions

In this section, we analyze the trainability of different loss functions used in quantum generative modeling.

Pairwise explicit losses. Part of the power of *quantum* generative models is that they can be used to continuously parameterize and express distributions over discrete data with exponential support. That is, an n-qubit model can be used to model distributions over 2^n different n-bitstrings. However, while the true target distribution may have exponential support, the amount of training data $\tilde{P}$ is in practise restricted. More precisely, for large n (e.g., $n > 50$), it is reasonable to assume that the number of bitstrings in the training dataset scales at most polynomially in n. Similarly, the number of bitstrings samples obtained from the model must also scale at most polynomially in n. That is, $|\tilde{P}|, |\tilde{Q}_{\boldsymbol{\theta}}| \in \mathcal{O}(\mathrm{poly}(n))$.

This discrepancy between the polynomial support of the training data and the exponential support of the model, can make it highly challenging to

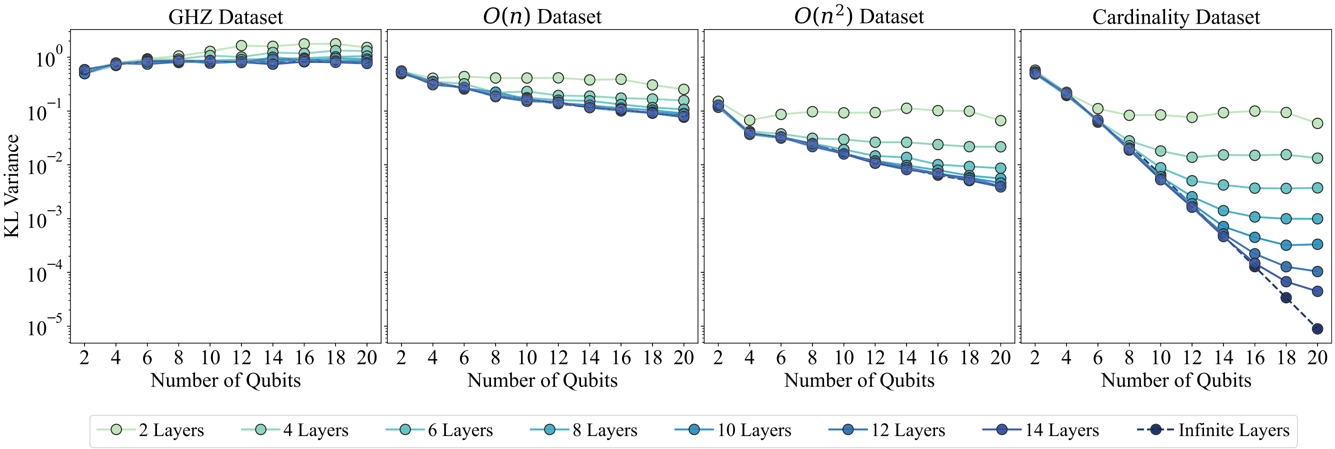

Fig. 2 | Study of loss concentration with the *exact* KLD loss function. Numerical evidence that the *exact* KLD loss can have a non-vanishing loss variance even when model probabilities exhibit exponential concentration. We study the loss concentration in randomly initialized line-topology circuits for various datasets, and increasing the number of qubits n and circuit depth. We emphasize that the model probabilities $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ where evaluated exactly and in the absence of shot noise. We also show the infinite layer results beyond 6 qubits that are generated using Eq. (20). The GHZ dataset consists of the all-0 and all-1 bitstrings ($\mathcal{O}(1)$ support), the $\mathcal{O}(n)$ and $\mathcal{O}(n^2)$ datasets consist of n and n^2 random bitstrings, respectively, and the cardinality dataset contains all bitstrings with $\frac{n}{2}$ cardinality ($\mathcal{O}(2^n)$ support). There appears to be a strong data dependence for the magnitude of the loss variance, which could lead to exponential concentration.

train implicit models using pairwise explicit loss functions. In loose terms, the problem is that the only bitstrings that contribute to the evaluation of a statistical estimate of an explicit cost are those corresponding to bitstrings $\tilde{P}$ in the training data. To estimate the loss one thus needs good estimates of the model distributions over the support of $\tilde{P}$. However, for an implicit model these estimates are obtained via sampling and the set $\tilde{P}$ contains an exponentially small proportion of the total number of bitstrings. As such, *for generic models* (i.e., those using no information about the particular dataset at hand), the probability of measuring any bitstring in the training set will also be exponentially small (as sketched in Fig. 3), leading to a poor statistical estimate of the loss. This observation was in fact one of the original motivations for moving away from the KLD and introducing the MMD loss in a quantum context in ref. 54 or follow-up works such as ref. 13.

Concentration of pairwise explicit losses. To make this line of argument more concrete, the first family of models we will consider are those where the individual model probabilities $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ are exponentially concentrated over different values of $\boldsymbol{\theta}$. This is the case for a large family of unstructured parameterized quantum circuits. Since estimating $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ is equivalent to computing the expectation value of the global projector $|\boldsymbol{x}\rangle\langle\boldsymbol{x}|$, the concentration of $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ can be viewed as resulting from the global-measurement-induced BP phenomenon[32]. In this case, concentration is observed even for an ansatz that is comprised of only a single layer of single-qubit rotations. However, alternative phenomena (e.g., noise[35] or expressibility[31]) can also lead to the exponential concentration of $q_{\boldsymbol{\theta}}(\boldsymbol{x})$. More formally, the following proposition holds.

Proposition 2. (Concentration of model). For all possible bitstrings $\boldsymbol{x} \in \mathcal{X}$, the underlying probability $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ of the quantum model exponentially concentrates towards some exponentially small fixed point $\mu \in O(1/b^n)$ for $b > 1$ if the quantum generative model is constructed with:

- A single layer of random single qubit gates $U(\boldsymbol{\theta})= \bigotimes_{i=1}^{n} U_i(\boldsymbol{\theta}_i)$. Or, more precisely, if $\{U_i(\boldsymbol{\theta}_i)\}_{\boldsymbol{\theta}_i}$ forms a local 2-design on qubit i[32].
- L layers of random k-local 2-designs, i.e., $U(\boldsymbol{\theta}) = \prod_{l=1}^{L} \bigotimes_{j=1}^{n/k} U_{l,j}(\boldsymbol{\theta}_{l,j})$ with each $U_{l,j}(\boldsymbol{\theta}_{l,j})$ acting on k qubits and $\{U_{l,j}(\boldsymbol{\theta}_{l,j})\}_{\boldsymbol{\theta}_{l,j}}$ forming a k-local 2-design over $\boldsymbol{\theta}_{l,j}$[32].
- A parameterized quantum circuit $U(\boldsymbol{\theta})$ such that its ensemble over $\boldsymbol{\theta}$ i.e., $\{U(\boldsymbol{\theta})\}_{\boldsymbol{\theta}}$ forms an approximate 2-design on n qubits[23,31]. This holds even for the problem-inspired circuits[25].
- A linear-depth quantum circuit subject to local Pauli noise between each layer[35].

Proposition 2 provides examples of cases where the model probabilities exponentially concentrate over *all* bitstrings in $\mathcal{X}$. However, we find that in fact trainability difficulties arise even if model probabilities are only exponentially concentrated over the training dataset (but perhaps not on points outside the dataset). That is, all that is required for untrainability is that the probability of measuring a sample that is also in the dataset is practically zero. This is likely to be the case even for highly structured quantum circuits if the generative model is built without a strong inductive bias. We formalize this intuition in Supplementary Note II B.

We now argue that the exponential concentration of probabilities $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ over the dataset causes $\tilde{\mathcal{L}}(\boldsymbol{\theta})$ to also exponentially concentrate. To understand why, let us look at the probability of measuring one specific bitstring (e.g., $\boldsymbol{x}_0$—the all-zero bitstring) and assume that $q_{\boldsymbol{\theta}}(\boldsymbol{x}_0)$ is exponentially concentrated towards some exponentially small value μ. Then, for any given parameter constellation, it is highly likely that $q_{\boldsymbol{\theta}}(\boldsymbol{x}_0)$ is exponentially close to μ. To estimate $q_{\boldsymbol{\theta}}(\boldsymbol{x}_0)$ on a quantum computer we sample N bitstrings from the quantum model and record the observations. The chance that none of the sampled bitstrings are the specific bitstring that we are interested in is $(1 - q_{\boldsymbol{\theta}}(\boldsymbol{x}_0))^N \approx 1 - N\mu$. However, the number of circuits N that can be efficiently run is necessarily limited—here we will assume $N \in \text{poly}(n)$. Thus we have that the probability of not measuring the bitstring we are interested in is exponentially close to 1. That is, the statistical estimate of $\tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x}_0)$ is almost always zero. We can then generalize this intuition for a single bitstring to the estimation of each of the (polynomially many) target bitstrings and therefore the whole loss function. The following theorem formalizes this argument.

Theorem 1. (Concentration of pairwise explicit loss for concentrated models). Consider the loss function of the form in Eq. (3). Assume that for all bitstrings in the training dataset, $\boldsymbol{x} \in \tilde{P}$, the quantum generative model $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ exponentially concentrates towards some exponentially small value (as defined in Definition 1). Suppose that $N \in \mathcal{O}(\text{poly}(n))$ samples are collected from the quantum model corresponding to the set of sampled bitstrings $\tilde{Q}_{\boldsymbol{\theta}}$, and that the training dataset $\tilde{P}$ contains $M \in \mathcal{O}(\text{poly}(n))$ samples. We define the fixed point of the loss as

$$\mathcal{L}_0(\tilde{P}, \tilde{Q}_{\boldsymbol{\theta}}) = \sum_{\boldsymbol{x}\in\mathcal{P}} f(\tilde{p}(\boldsymbol{x}), 0) + \sum_{\boldsymbol{x}\in\mathcal{Q}_{\boldsymbol{\theta}}} f(0, \tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x})), \tag{23}$$

with $\mathcal{P}$ (and $\mathcal{Q}_{\boldsymbol{\theta}}$) being a set of *unique* bitstrings in $\tilde{P}$ (and $\tilde{Q}_{\boldsymbol{\theta}}$). Then, the probability that the estimated value $\tilde{\mathcal{L}}(\boldsymbol{\theta})$ is equal to $\mathcal{L}_0(\tilde{P}, \tilde{Q}_{\boldsymbol{\theta}})$ is

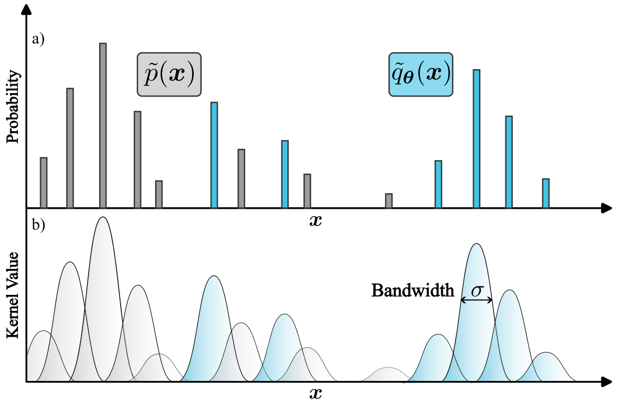

Fig. 3 | The problem with pairwise explicit losses. In the space with 2^n unique n-bit bitstrings, samples $\boldsymbol{x}$ generated from an uninformed model with high probability do not coincide with any of the training bitstrings. In other words, the empirical model distribution $\tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x})$ and the training distribution $\tilde{p}(\boldsymbol{x})$ do not both have non-zero probabilities for any bitstring $\boldsymbol{x}$. On the other hand, an implicit loss function such as the MMD provides a continuous measure of distance between the distributions by use of a Gaussian kernel with bandwidth σ.

exponentially close to 1, i.e.,

$$\Pr_{\tilde{Q}_{\boldsymbol{\theta}},\boldsymbol{\theta}}[\tilde{\mathcal{L}}(\boldsymbol{\theta})=\mathcal{L}_0(\tilde{P},\tilde{Q}_{\boldsymbol{\theta}})]\geqslant 1-\delta, \tag{24}$$

with $\delta\in\mathcal{O}\left(\frac{\mathrm{poly}(n)}{c^n}\right)$ for some $c>1$.

As a direct consequence of Theorem 1, the following corollary gives the concentration points of some specific explicit loss functions mentioned in this work.

Corollary 2. (Concentration points of common explicit loss functions). Under the same conditions as in Theorem 1, the following loss functions concentrate at

- KL-divergence:

$$\mathcal{L}_0^{\mathrm{KLD}}(\tilde{P},\tilde{Q}_{\boldsymbol{\theta}})=\sum_{\boldsymbol{x}\in\mathcal{P}}\tilde{p}(\boldsymbol{x})\log\left(\frac{\tilde{p}(\boldsymbol{x})}{\epsilon}\right). \tag{25}$$

Here $\epsilon\ll 1$ is a clipping value, which is common practice to avoid the singularity of the logarithm at $q_{\boldsymbol{\theta}}(\boldsymbol{x})=0$.

- Classical fidelity:

$$\mathcal{L}_0^{\mathrm{CF}}(\tilde{P},\tilde{Q}_{\boldsymbol{\theta}})=1. \tag{26}$$

- Reverse KL-divergence:

$$\mathcal{L}_0^{\mathrm{rev-KLD}}(\tilde{P},\tilde{Q}_{\boldsymbol{\theta}})=\sum_{\boldsymbol{x}\in\mathcal{Q}_{\boldsymbol{\theta}}}\tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x})\log\left(\frac{\tilde{q}_{\boldsymbol{\theta}}(\boldsymbol{x})}{\epsilon}\right). \tag{27}$$

- Total variation distance:

$$\mathcal{L}_0^{\mathrm{TVD}}(\tilde{P},\tilde{Q}_{\boldsymbol{\theta}})=2. \tag{28}$$

Looking at the expressions for the fixed points given above, in the case of the KL divergence, classical fidelity and total variational distance, the fixed point is independent of $\boldsymbol{\theta}$. Thus it is clear that the costs cannot be used to train the quantum circuit model. In the case of the reverse KL divergence, the fixed point depends on $\boldsymbol{\theta}$ but is independent of the training data and thus the reverse KL also cannot be used to train the model to learn the target distribution.

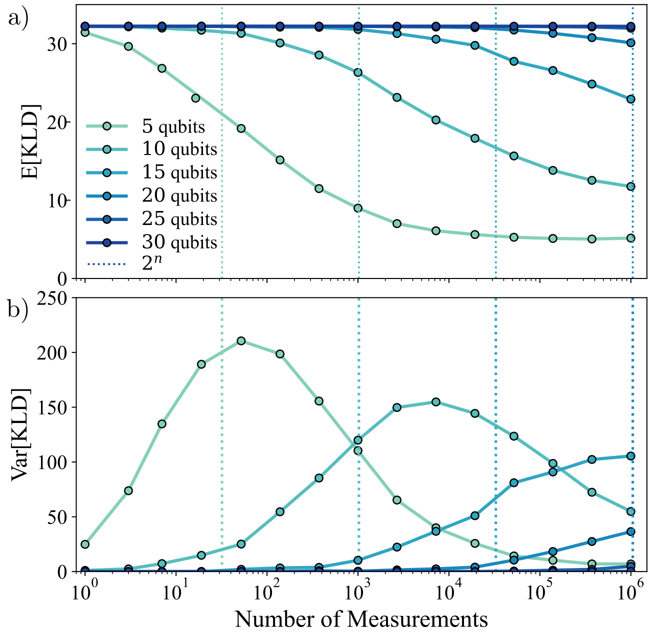

Fig. 4 | Variance of the KL divergence with finite shots. Concentration of the KL divergence loss as a function of the number of measurements and qubits for random product state circuits. Here we take the target distribution to be $\tilde{p}(\boldsymbol{0})=1$ and take the cutoff of the KLD to be $\epsilon=10^{-14}$. Vertical lines indicate where the number of measurements equal 2^n. Thus, we see that the the KLD estimate is biased upwards with any finite number of measurements, and the number of measurements required to achieve a reasonable level of uncertainty increases exponentially with the number of qubits n.

More generally, for all explicit losses of the form Eq. (3), the concentration point $\mathcal{L}_0(\tilde{P},\tilde{Q}_{\boldsymbol{\theta}})$, Eq. (23), can be separated into two terms: (i) the term that involves only $\tilde{P}$ and (ii) the other that involves only $\tilde{Q}_{\boldsymbol{\theta}}$. In other words, the $\boldsymbol{\theta}$ dependence of the estimator of the loss is independent of the target distribution and thus the estimate of the loss is worthless for training the generative model. This no-go result is rigorously established in Corollary 3. Our approach is to show that the loss function at two arbitrary parameter values $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, contains no information about the training distribution.

Corollary 3. (Untrainability of pairwise explicit loss functions). Under the same conditions as in Theorem 1, the probability that the difference between the two statistical estimates of the loss function at $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ does not contain any information about the training distribution is exponentially close to 1. Particularly, we have

$$\Pr_{\tilde{Q}_{\boldsymbol{\theta}},\boldsymbol{\theta}}[\tilde{\mathcal{L}}(\boldsymbol{\theta}_1)-\tilde{\mathcal{L}}(\boldsymbol{\theta}_2)=\Delta\mathcal{L}_0(\tilde{Q}_{\boldsymbol{\theta}_1},\tilde{Q}_{\boldsymbol{\theta}_2})]\geqslant 1-2\delta, \tag{29}$$

with $\delta\in\mathcal{O}\left(\frac{\mathrm{poly}(n)}{c^n}\right)$ for some $c>1$, $\tilde{Q}_{\boldsymbol{\theta}_1}$ (and $\tilde{Q}_{\boldsymbol{\theta}_2}$) is a set of sampling bitstrings obtained from the quantum generative model at the parameter value $\boldsymbol{\theta}_1$ (and $\boldsymbol{\theta}_2$), as well as

$$\Delta\mathcal{L}_0(\tilde{Q}_{\boldsymbol{\theta}_1},\tilde{Q}_{\boldsymbol{\theta}_2})=\sum_{\boldsymbol{x}\in\mathcal{Q}_{\boldsymbol{\theta}_1}}f(0,\tilde{q}_{\boldsymbol{\theta}_1}(\boldsymbol{x}))-\sum_{\boldsymbol{x}\in\mathcal{Q}_{\boldsymbol{\theta}_2}}f(0,\tilde{q}_{\boldsymbol{\theta}_2}(\boldsymbol{x})), \tag{30}$$

with $\mathcal{Q}_{\boldsymbol{\theta}_1}$ (and $\mathcal{Q}_{\boldsymbol{\theta}_2}$) being a set of unique bit-strings in $\tilde{Q}_{\boldsymbol{\theta}_1}$ (and $\tilde{Q}_{\boldsymbol{\theta}_2}$). Crucially, $\Delta\mathcal{L}_0(\tilde{Q}_{\boldsymbol{\theta}_1},\tilde{Q}_{\boldsymbol{\theta}_2})$ does not depend on any $\tilde{p}(\boldsymbol{x})\in\tilde{P}$.

To support our analytic claims we further conducted a numerical study of the exponential concentration of pairwise explicit costs. For concreteness, we here decided to focus on the KL divergence. In Fig. 4, we plot the mean and variance (over $\boldsymbol{\theta}$) of the KL divergence for the target distribution $\tilde{p}(\boldsymbol{0})=1$ as a function of the number of measurement shots and qubits. For simplicity we take our model to be a (Haar) random product state.

We see in (Fig. 4a) that with a polynomial number of measurements, as per Eq. (25), the empirical estimate of the loss concentrates at $\log(1/\epsilon) \approx 32.2$ for $\epsilon = 10^{-14}$. Correspondingly, with a polynomial number of measurements the variance in (Fig. 4b) is exponentially close to zero. Using an exponential number of measurements, the estimate of the KL tends towards its true value and the variance is again small. The transition between these two regimes is marked by a very high variance corresponding to the case where the measurement count is high enough for there to be some overlap between the sampled bits strings and the $\boldsymbol{0}$ bitstring, but not enough overlap to obtain a reliable estimate of $q_{\boldsymbol{\theta}}(\boldsymbol{0})$. This results in the loss estimate to sporadically fluctuate between $\log(1/\epsilon)$ and $\log(1/q_{\boldsymbol{\theta}}(\boldsymbol{x}))$ with $q_{\boldsymbol{\theta}}(\boldsymbol{x}) > 0$. While in Fig. 4 the target dataset consists of a single bitstring, larger datasets only shift the curves to the left by a polynomial amount.

Broader Implications. While our results above are formulated for training QCBMs with pairwise explicit costs, we argue that the underlying problem is more general and immune to simple solutions. One approach, for example, might be to take non-data-dependent functions of pairwise explicit losses, as in the case of the Rényi-divergence in Eq. (9). However, such loss functions exponentially concentrate in the same manner as the explicit losses themselves when employing the conventional measurement strategy. A more promising but challenging approach would be to attempt to measure such losses via quantum strategies. We discuss this further in Section "Quantum strategies: quantum fidelity".

More generally, while we provide strict no-go results only for pairwise explicit losses, we believe that any explicit losses in the general form of Eq. (2) will suffer from concentration or exponential imprecision due to the inherent inability of implicit models to accurately estimate the model probabilities in polynomial time (A possible exception is if a particular implicit model instead allows for efficient estimation of gradients of an explicit loss function, as it is the case for RBMs training on the KL divergence loss function.). We are however not aware of any practical explicit loss function that cannot be brought into the pairwise explicit form.

We further stress that our results hold for unstructured ansätze or ansätze that lack an appropriate inductive initial bias. Thus, while explicit losses such as the KLD will not work at scale with implicit models straight out-of-the-box, our no-go theorems could be side-stepped using clever initialization strategies in conjunction with specialized ansätze. For example, while we argue in Supplementary Note II B that initializing the quantum circuit model on a subset of training states will not alleviate the fundamental issue when using a generic ansatz, this may work if one leverages a quantum circuit that constrains the model to the symmetry sector of the data. Among other hard constraints, this is conceivable if the data consists only of samples with a certain hamming weight or cardinality, as it can be the case in certain financial applications[84,85]. However, many real-world datasets may not contain strong symmetries that one can leverage so straightforwardly. It is therefore critically important to study the effect of strong parameter initializations and inductive biases using explicit losses—both theoretically and experimentally.

Implicit losses: maximum mean discrepancy. In the previous section we saw that an explicit loss function, used in conjunction with an implicit generative model and the conventional sampling strategy, exhibits exponential concentration and hence is untrainable. The root cause was, at least in part, a miss-match between using an explicit loss function with an implicit model. Thus it is natural to ask whether an implicit quantum loss would fare better.

Here we focus on analyzing the MMD loss function (see Eqs. (11) and (13)), which is a commonly-used implicit loss. In contrast to the pairwise explicit losses discussed previously, each bitstring drawn from the model is generally compared with all training bitstrings, with the kernel function $K(\boldsymbol{x}, \boldsymbol{y})$ controlling the contribution of each comparison. With a poor choice in kernel it is clear that the MMD will be susceptible to exponential concentration. For example, the Gaussian kernel with the bandwidth $\sigma \to 0$ is equivalent to a delta function kernel, $K(\boldsymbol{x}, \boldsymbol{y}) = \langle \boldsymbol{x}, \boldsymbol{y}\rangle = \delta_{\boldsymbol{x}\boldsymbol{y}}$. In this case the MMD reduces to the pairwise explicit loss $\sum_{\boldsymbol{x}\in\mathcal{X}}(p(\boldsymbol{x}) - q_{\boldsymbol{\theta}}(\boldsymbol{x}))^2$ (see Supplementary Note I for details), and consequently is subject to our no-go result in Theorem 1. This thus prompts the question of how exactly σ affects trainability.

Properties of the MMD loss. To study the properties of the MMD loss, it is helpful to note that each term in the MMD can be viewed as the expectation value of an observable whose properties depend on the choice of σ. This change in perspective allows us to leverage existing knowledge from the VQA trainability literature. In particular, prior no-go results on VQAs with observable-type loss functions are now directly applicable here, including those on cost function induced[32], expressiblity-induced[23,31], and noise-induced[35] BPs.

Specifically, each term in the MMD can be written as

$$\mathcal{M}(\rho, \rho') = \mathrm{Tr}[O^{(\sigma)}_{\mathrm{MMD}}(\rho \otimes \rho')], \tag{31}$$

where we have defined the MMD observable

$$O^{(\sigma)}_{\mathrm{MMD}} := \sum_{\boldsymbol{x},\boldsymbol{y}} K_\sigma(\boldsymbol{x}, \boldsymbol{y}) |\boldsymbol{x}\rangle\langle\boldsymbol{x}| \otimes |\boldsymbol{y}\rangle\langle\boldsymbol{y}|. \tag{32}$$

This observable acts on $2n$ qubits, namely n qubits corresponding to the QCBM, $\rho_{\boldsymbol{\theta}} = |\psi(\boldsymbol{\theta})\rangle\langle\psi(\boldsymbol{\theta})|$, and n qubits corresponding to the dataset, $\rho_{\tilde{p}} = \sum_{\boldsymbol{y}} \tilde{p}(\boldsymbol{y})|\boldsymbol{y}\rangle\langle\boldsymbol{y}|$. For the first term in the MMD, both $\boldsymbol{x}$ and $\boldsymbol{y}$ are sampled from the QCBM and we have $\rho = \rho' = \rho_{\boldsymbol{\theta}}$. The cross-term instead has $\rho = \rho_{\boldsymbol{\theta}}$ and $\rho' = \rho_{\tilde{p}}$, and the final term has $\rho = \rho' = \rho_{\tilde{p}}$.

In the Pauli basis, the MMD observable $O^{(\sigma)}_{\mathrm{MMD}}$ takes the elegant form

$$O^{(\sigma)}_{\mathrm{MMD}} = \sum_{l=0}^{n} w_\sigma(l) D_{2l}, \tag{33}$$

where D_{2l} are normalized $2l$-body diagonal operators (defined explicitly in Supplementary Note IV A), and

$$w_\sigma(l) = \binom{n}{l}(1 - p_\sigma)^{n-l} p_\sigma^l \tag{34}$$

are Bernoulli-distributed weights with effective *probability*

$$p_\sigma = (1 - e^{-1/2\sigma})/2. \tag{35}$$

Thus estimating the MMD loss function in Eq. (11) using a batch of measurements $\tilde{Q}$ is equivalent to using the same measurements to estimate a weighted expectation of the observables D_{2l}.

The properties of the MMD observable clearly depend on the distribution of the terms of different bodyness through the $w_\sigma(l)$ factor. Figure 5 shows how $w_\sigma(l)$ are distributed for different σ. Owing to the Bernoulli-distributed weights, we can straightforwardly provide the average bodyness of $O^{(\sigma)}_{\mathrm{MMD}}$, which is given by

$$\mathbb{E}_{l\sim\omega_\sigma(l)}[2l] = 2np_\sigma, \tag{36}$$

and the variance in the bodyness, which is

$$\mathrm{Var}_{l\sim\omega_\sigma(l)}[2l] = 4np_\sigma(1 - p_\sigma). \tag{37}$$

From these expressions it follows that the MMD loss is predominantly composed of global operators when $\sigma \in \mathcal{O}(1)$. More concretely the following proposition holds.

Proposition 3. (MMD consists largely of global terms for $\sigma \in \mathcal{O}(1)$). For $\sigma \in \mathcal{O}(1)$, the average bodyness of the MMD operator containing Pauli terms with weight $w_\sigma(l)$ is

$$\mathbb{E}_{l\sim w_\sigma(l)}[2l] \in \Theta(n). \tag{38}$$

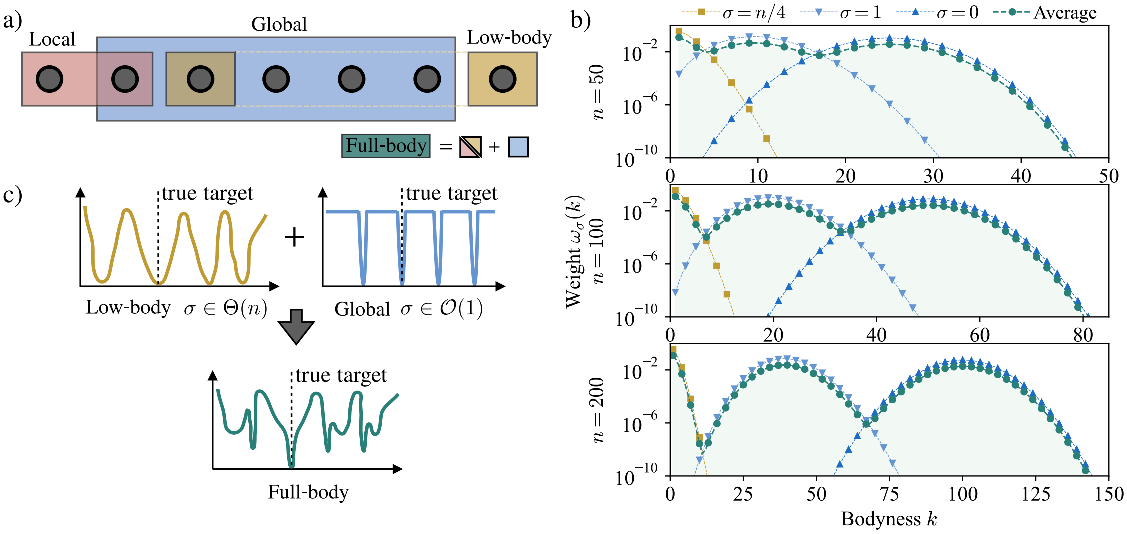

Fig. 5 | Bodyness of the MMD loss. a We illustrate the difference between “low-body”, “local”, “global” and “full-body” operators. An operator O is low-bodied if it acts non-trivially on at most $\mathcal{O}(\log(n))$ qubits. If a low-bodied operator acts on qubits that are adjacent to each other, then O is said to be local. On the other hand, O is global if it acts non-trivially on $\Theta(n)$ qubits. Lastly, a full-body operator consists of the sum of several operators that are low-body/local and global. **b** We depict the weight $w_\sigma(k)$ for the terms in the MMD operator as a function of their bodyness k for $n = 50, 100$, and 200 qubits and the bandwidths $\sigma = 0$, 1, and $n/4$. The average weight over these three σ values is also shown. For small σ, the MMD operator is a sum of predominantly global operators, i.e., with $\sigma \in \mathcal{O}(1)$ the mean bodyness is $\Theta(n)$. In contrast, $\sigma \in \Theta(n)$ results in predominantly low-bodied operators. **c** Sketch of the expected landscapes for low-body, global and full-body losses respectively. Because low-body and global operators are exclusively sensitive to low-body and global features, respectively, their loss landscapes exhibit spurious minima, which don’t coincide with the minimum of the true target distribution. A full-body loss on the other hand should have a single optimal solution solution where all its constituent operator’s minima align.

Similarly, the variance in the bodyness is given by

$$\mathrm{Var}_{l\sim w_\sigma(l)}[2l] \in \Theta(n). \tag{39}$$

This shows that with fixed-size bandwidths σ, as is commonly done (e.g., ref. 54), the MMD suffers from global loss function-induced BPs[32] and hence is untrainable. This practice of using constant bandwidths is carried over from classical ML literature[86–88], but Proposition 3 shows that this is fundamentally incompatible with quantum generative models using unstructured circuits.

In contrast, we show that if the bandwidth scales linearly in the number of qubits, $\sigma \in \Theta(n)$, the MMD loss function is approximately low-bodied. We recall that being low-bodied is more general than being *local*, the latter corresponding to the case where an operator is low-bodied and each term only acts non-trivially on adjacent qubits. The following proposition formalizes this relation by quantifying the error made when truncating the MMD observable after a certain bodyness.

Proposition 4. (MMD consists largely of low-body terms for $\sigma \in \Theta(n)$). Let $\tilde{\mathcal{L}}^{(\sigma,k)}_{\mathrm{MMD}}(\boldsymbol{\theta})$ be a truncated MMD loss with a truncated operator $\tilde{O}^{(\sigma,k)}_{\mathrm{MMD}}$ that contains up to the $2k$-body interactions in $O^{(\sigma)}_{\mathrm{MMD}}$,

$$\tilde{O}^{(\sigma,k)}_{\mathrm{MMD}} := \sum_{l=0}^{k} w_\sigma(l) D_{2l}, \tag{40}$$

where $w_\sigma(l)$ are Bernoulli-distributed weights defined in Eq. (34). For $\sigma \in \Theta(n)$, the difference between the exact and local approximation of the loss is bounded as

$$|\mathcal{L}^{(\sigma)}_{\mathrm{MMD}}(\boldsymbol{\theta}) - \tilde{\mathcal{L}}^{(\sigma,k)}_{\mathrm{MMD}}(\boldsymbol{\theta})| \leqslant \epsilon(k), \tag{41}$$

with

$$\epsilon(k) \in \mathcal{O}\left(n(c/k)^k\right), \tag{42}$$

for some positive constant c.

This implies that one can view the MMD loss with a bandwidth $\sigma \in \Theta(n)$ as composed almost exclusively of low-body contributions. We therefore expect, given the results of refs. 32,40, that the MMD is trainable for $\sigma \in \Theta(n)$ for quantum generative models which employ shallow quantum circuits. We note that there appears to be no merit in increasing σ beyond $\Theta(n)$, as that simply increases the relative weight of the constant $l = 0$ term in Eq. (33). That is, the MMD operator tends towards the trivial identity measurement for $\sigma \longrightarrow \infty$.

To probe this further, and get a better understanding of the effect of σ on the trainability of the MMD loss, we start by considering the case of QCBM with a product ansatz. This allows us to find a closed-form expression of the MMD variance as a function of the circuit parameters (Supplemental Proposition 2) from which we can study the concentration of the MMD for different σ values. Our findings are summarized by the following Theorem (proven in Supplementary Note IV B 3).

Theorem 2. (Product ansatz trainability of MMD, informal). Consider the MMD loss function $\mathcal{L}^{(\sigma)}_{\mathrm{MMD}}(\boldsymbol{\theta})$ as defined in Eq. (11), which uses the classical Gaussian kernel as defined in Eq. (12) with the bandwidth $\sigma > 0$, and a quantum circuit generative model that is comprised of a tensor-product ansatz $U = \bigotimes_i^n U_i(\theta_i)$ with $\{U_i(\theta_i)\}_{\theta_i}$ being single-qubit (Haar) random unitaries. Given a training dataset $\tilde{P}$, the asymptotic scaling of the variance of the MMD loss depends on the value of σ.

For $\sigma \in \mathcal{O}(1)$, we have

$$\mathrm{Var}_{\boldsymbol{\theta}}[\mathcal{L}^{(\sigma)}_{\mathrm{MMD}}(\boldsymbol{\theta})] \in \mathcal{O}(1/b^n), \tag{43}$$

with some $b > 1$.

On the other hand, for $\sigma \in \Theta(n)$, we have

$$\mathrm{Var}_{\boldsymbol{\theta}}[\mathcal{L}^{(\sigma)}_{\mathrm{MMD}}(\boldsymbol{\theta})] \in \Omega(1/n). \tag{44}$$

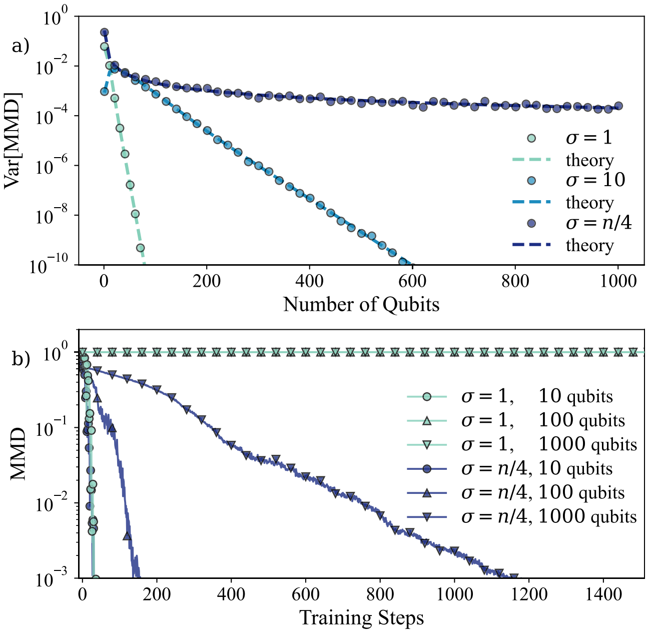

Fig. 6 | σ-dependence of the MMD loss function. a Comparison of the MMD variance between the analytical prediction (see Eq. (182) in Supplementary Information for an exact analytical expression) and empirical variance using 100 measurements from a product state ansatz. **b** Training a product state ansatz on the $|\boldsymbol{0}\rangle$ target state for $\sigma = 1$ and $\sigma = n/4$ using the CMA-ES[113] optimizer and 512 measurements. In both cases the QCBM ansatz consists of a single layer of Ry rotations on each qubit.

We numerically verify Theorem 2 in Fig. 6. In panel (a) we show that the analytical predictions for different bandwidths coincide perfectly with the numerical estimates. The exponentially vanishing loss variances observed for $\sigma \in O(1)$ are expected to render the loss untrainable. This is demonstrated in panel (b), where we further train a QCBM with $\sigma = n/4$ (which approximately maximizes the variance) and $\sigma = 1$. We find that a QCBM with $\sigma = n/4$ can be successfully trained even for $n = 1000$ qubits. In contrast, the training starts to fail to learn the $|\boldsymbol{0}\rangle$ target state after $n \approx 50$ and is fully untrainable at $n = 100$ when $\sigma = 1$ is used.

It is interesting to note that the approximately optimal bandwidth $\sigma \sim \frac{n}{4}$ for the product state ansatz coincides with the so-called *median heuristic*[51] from classical ML literature. For random circuits, the median (hamming) distance between bitstrings is in fact $\frac{n}{2}$, which we satisfy with the factor of 2 in our kernel convention.

To go towards more practical generative modeling, we recall that ref. 40 proves that cost functions of the form of Eq. (19) using $2k$-body observables with $k \in \mathcal{O}(\log(n))$ are trainable using 1D-random $\log(n)$ depth circuits. Since Proposition 4 implies that the MMD is well approximated by a $\log(n)$-body cost, it should follow that the MMD is also trainable at $\log(n)$ depths. There are a few technical caveats associated with constructing a full proof. For example, the first term of the MMD requires working with 4-designs instead of 2-designs, and the second term depends on the target distribution, leading to additional subtleties. However, there is no strong reason to expect that these technicalities make the MMD untrainable.

To extend the trainability result beyond a simple tensor product ansatz, we consider a generic Pauli rotation ansatz of the form $U_{\mathrm{PQC}}(\boldsymbol{\theta}) = U(\boldsymbol{\alpha})\, U_{\mathrm{tensor}}(\boldsymbol{\beta})$ where

$$U(\boldsymbol{\alpha}) = \prod_{k=1}^{M} e^{-i\alpha_k G_k/2} V_k \tag{45}$$

such that the generators $\{G_k\}_{k=1}^{M}$ are some n-qubit Pauli strings $G_k \in \{\mathbb{1}, X, Y, Z\}^n$ and $\{V_k\}_{k=1}^{M}$ is a set of non-parametrized Clifford gates, and $U_{\mathrm{tensor}}(\boldsymbol{\beta}) = \bigotimes_{i=1^n} U_i(\beta_i)$ is a layer of random single qubit rotations $U_i(\beta_i)$. We additionally assume all parameters are uncorrelated. Then, we have the variance of the MMD loss scales as follows.

Theorem 3. (General Pauli rotation ansatz trainability of MMD, informal). Consider a general Pauli rotation ansatz of the form in Eq. (45) and the MMD loss function as defined in Eq. (11) using the Gaussian kernel in Eq. (12) with the bandwidth $\sigma \in \Theta(n)$. As long as the average light cone of the back-propagated MMD observable with $U'(\boldsymbol{\alpha})$ remains in the order of $\log(n)$, then the QCBM is trainable in the sense that

$$\mathrm{Var}_{\boldsymbol{\theta}}[\mathcal{L}^{(\sigma)}_{\mathrm{MMD}}(\boldsymbol{\theta})] \in \Omega(1/\mathrm{poly}(n)). \tag{46}$$

Theorem 3 indicates that in practice one has to only determine the average light cone of the effective MMD observable back-propagated by a *given* ansatz (see Eq. (243) in Supplementary Note IV C for a formal definition) to have a trainability guarantee. In Supplementary Note IV C, we provide further details on how to compute this average light cone for the Pauli rotation ansatz. An example of an ansatz satisfying the few-body light cone condition is a shallow depth circuit with nearest neighbor connectivity (which could be either hardware efficient or problem inspired). Crucially, we emphasize that this light cone argument goes beyond the 2-design assumption and is expected to work even more generally to any ansatz that may not even be in the Pauli rotation form.

It is worth noting that as few mild technical assumptions are required to more formally state Theorem 3. These, along with our proof are provided in Supplementary Note IV C. We further remark that although some proof techniques are similar to ref. 89, the main technical challenges here are to analytically show that the covariance between different terms which involve higher moments vanish and compute the exact form of the variance lower bound of the purity term. We further remark that although some proof techniques are similar to ref. 89, the main technical challenges here have arisen from dealing with the two system registers of the MMD which involves computing the exact form of the higher moments.

Theorem 3 is further supported by our numerical evidence for the trainability of the MMD for deeper circuits and more realistic datasets shown in Fig. 7. Here we plot the loss variance as a function of circuit depth L and the number of qubits n for $\sigma = n/4$ on four datasets from four different target distributions. We observe that the polynomial scaling of the loss variance does in fact extend beyond product states to shallow circuits, i.e., $L \in \mathcal{O}(\log(n))$. However, for sufficiently deep circuits, i.e., $L \in \Omega(n)$, the MMD variance appears to decay exponentially. This aligns with expressibility-induced BPs observed in other VQA applications, which occur even for maximally local loss functions, i.e., $k = 1$.

The role of local minima. Our results so far appear to indicate that picking a single bandwidth $\sigma \in \Theta(n)$ maximizes the trainability of the MMD loss function with a Gaussian kernel. While it is true that this choice maximizes the expected magnitude of initial gradients for a QCBM, non-vanishing gradients are a necessary condition but not sufficient to guarantee reliable training performance. And in fact it turns out that while low-body losses exhibit large gradients they come with other limitations. Particularly, we show that the bodyness of a generative loss function defines the maximal order of marginals of the target distribution that can be distinguished. That is, the model only learns to match the target distribution on subsets of bits, i.e., on its marginals. This introduces a continuous family of minima which are indistinguishable from the true minimum when using a low-bodied loss function, but which are systematically wrong for the purposes of generative modeling. The worry is that the non-vanishing loss gradients in low-bodied losses are predominantly due to the presence of such spurious minima and do not point in the direction of the true global minimum. This is sketched in Fig. 5.

Formally, let $q_{\boldsymbol{\theta}}(\boldsymbol{x}_A)$ denote the marginal model distribution on a subset $A \subseteq \{1, 2, \ldots, n\}$ of qubits, and $\tilde{p}(\boldsymbol{x}_A)$ the marginal target distribution on that same subset. For more details we refer to Eq. (368) and Eq. (370) in

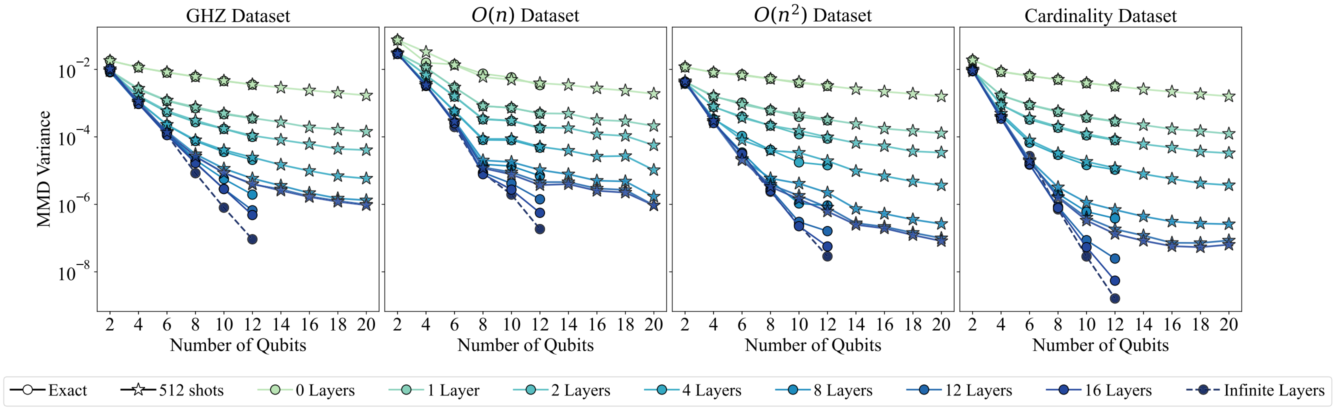

Fig. 7 | Study of loss concentration with the MMD loss function. Numerical evidence that the MMD loss with Gaussian bandwidth parameter $\sigma = n/4$ does not exhibit global or explicit loss function barren plateaus, but does exhibit loss concentration with deep quantum circuits. We study the loss concentration in randomly initialized line-topology circuits for various datasets, and increasing number of qubits n and circuit depth. The infinite layers results were generated by drawing wavefunctions from the Porter-Thomas distribution[114]. The GHZ dataset consists of the all-0 and all-1 bitstrings ($\mathcal{O}(1)$ support), the $\mathcal{O}(n)$ and $\mathcal{O}(n^2)$ datasets consist of n and n^2 random bitstrings, respectively, and the cardinality dataset contains all bitstrings with $\frac{n}{2}$ cardinality ($\mathcal{O}(2^n)$ support). There does not appear to be a strong data-dependence for the magnitude of the loss variance.

Supplementary Note IV D. The connection between the bodyness of the loss operator and the marginals of the model and target distributions is then formalized in the following Proposition.

Proposition 5. (The truncated MMD loss is not faithful). Consider a distribution $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ that agrees with the training distribution $\tilde{p}(\boldsymbol{x})$ on all the marginals up to k bits, but disagrees on higher-order marginals. The distribution $q_{\boldsymbol{\theta}}(\boldsymbol{x})$ minimizes the truncated MMD loss. That is, suppose

$$q_{\boldsymbol{\theta}}(\boldsymbol{x}_A) = \tilde{p}(\boldsymbol{x}_A), \tag{47}$$

for all $A \subseteq \{1, 2, \ldots, n\}$ with $|A| \leqslant k$, then

$$\tilde{\mathcal{L}}^{(\sigma,k)}_{\mathrm{MMD}}(\boldsymbol{\theta}) = 0. \tag{48}$$

Crucially, this is true even if for some $B \subseteq \{1, 2, \ldots, n\}$ with $|B| > k$

$$q_{\boldsymbol{\theta}}(\boldsymbol{x}_B) \neq \tilde{p}(\boldsymbol{x}_B). \tag{49}$$

In other words, if the MMD operator can be approximated well by a truncated operator with at most $2k$-body terms, model distributions that match the target distribution exactly up to k-body marginals or higher cannot be distinguished from ones that match fully. As an example of such distributions, consider the uniform distribution over the bitstrings [001, 011, 101, 110], where the third bit is the bit-wise addition of the previous two bits. Using only second-order marginals, it is not possible to distinguish this correlated distribution from the uniform distribution over all eight possible outcomes.

Notably, long-range correlations in the data can still be learned by the low-bodied MMD loss, just not ones that are particularly high-order (Note that this is in contrast to a loss composed purely of local terms which would be restricted to learning local/short-range correlations.). Not all distributions will however exhibit such higher-order correlations and thus some distributions will be learnable using losses composed of low-body terms. Viewing this result through the lens of generalization to the underlying distribution, there are two opposing consequences that would need to be studied in future works. On the one hand, convergence to the systematically wrong low-order minima is likely to impede generalization. On the other hand, one could also imagine that being ignorant to very high-order marginals of the training set could reduce overfitting and thus enhance generalization.

Proposition 5 thus establishes that to fulfill the promise of quantum generative models, that is to be able to learn both long-range *and* many-body correlations, one cannot use exclusively low-body losses. However, such a requirement is in immediate tension with the low-bodyness required for the trainability guarantees (see Theorem 2). In particular, in Proposition 4 we show that for $\sigma \in \Theta(n)$ the contribution of $k \in \Theta(n)$ terms are exponentially small in n. Thus, although the loss is still strictly faithful given an infinite shot budget, with a reasonable shot budget we will not be able to resolve the contribution from the exponentially small high-body terms. Hence, there can be spurious minima that we cannot resolve from the true minimum and therefore for all practical purposes the loss is effectively not truly faithful.

One approach to resolving this tension would be to adapt the initial value of σ from $\Theta(n)$, where the loss exhibits large gradients but predominantly learns low-order marginals, towards $\mathcal{O}(1)$ to also learn high-order correlations as the model improves. This is in line with studies from the classical ML literature showing that bandwidths for optimal MMD performance are oftentimes smaller than the so-called median heuristic[90–92], which coincides with our result of $\sigma \in \Theta(n)$. Another approach, which is also already employed in classical ML literature, is to use a kernel that averages the effects of several σ[86–88]. That is, the kernel is taken to be

$$K_{\mathbf{c}}(\boldsymbol{x}, \boldsymbol{y}) = \frac{1}{|\mathbf{c}|}\sum_{i\in\mathbf{c}} K_{\sigma_i}(\boldsymbol{x}, \boldsymbol{y}) \equiv \sum_{l=1}^{k} \langle w_\sigma(l)\rangle_{\mathbf{c}} D_{2l} \tag{50}$$

for a set of bandwidths $\mathbf{c} = \{\sigma_1, \sigma_2, \ldots\}$. The resulting weight of each $2k$—body term of the new MMD observable is an average of the weightings corresponding to each σ_i in $\mathbf{c}$ as shown in Fig. 5. Theorem 2 shows that for a QCBM without inductive bias to not fall prey to exponential concentration, at least one of the σ_i needs to be $\Theta(n)$. But the results of Proposition 5 suggest that for data sets exhibiting high-order correlations a small bandwidth $\sigma_i \in \mathcal{O}(1)$ is required for correct convergence. It stands to reason that the optimal set $\mathbf{c}$ contains a spectrum of bandwidths that both enable trainability and faithful convergence to the target distribution (as sketched in Fig. 5c). How successful this strategy is in practice remains to be determined.

Broader Implications. Our work highlights that one can treat classical machine learning losses as quantum observables to study their properties. This implies that our results transfer to other types of quantum generative models beyond the QCBM that will also be affected by the fundamental limitations described by Proposition 5. In fact, we show in Supplementary Note IV E that any generative modeling loss function for classical data that can be brought into the form $\mathcal{L}(\boldsymbol{\theta}) = \mathrm{Tr}[\mathcal{M}\rho_{\boldsymbol{\theta}}]$, with a diagonal measurement operator $\mathcal{M}$, faces the same tension described above. That is, if $\mathcal{M}$ contains at most k-body terms in the Pauli basis representation, then

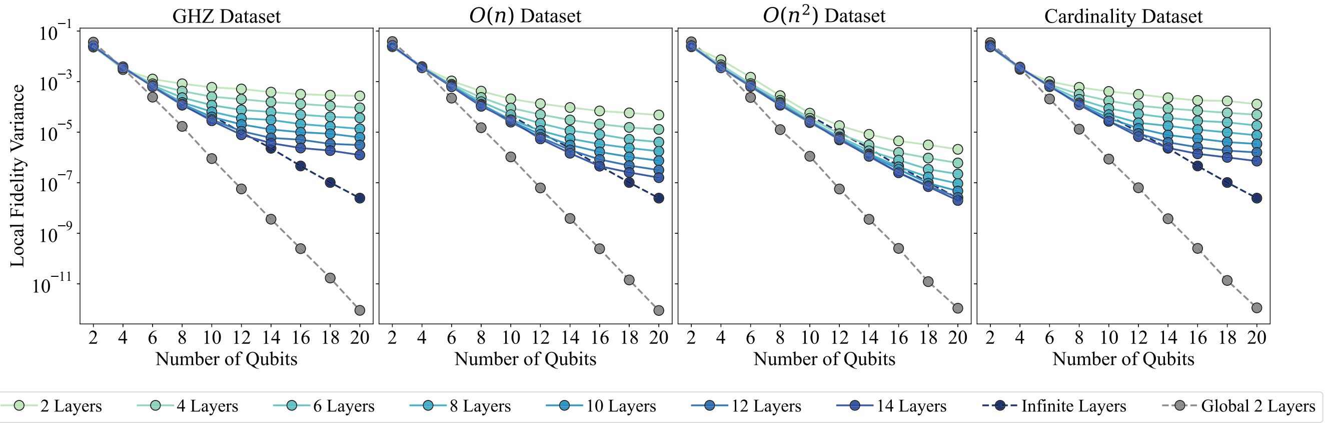

Fig. 8 | Study of loss concentration with the local quantum fidelity loss function. Numerical evidence that the local quantum fidelity loss function does not exhibit global or explicit loss function barren plateaus. It does however exhibit expressivity-induced barren plateaus with deeper and deeper circuits, as it is the case for all generic VQA-type loss functions in the form of Eq. (19). In contrast, the global quantum fidelity variance decays exponentially at all circuit depths. The numerical setup is the same as for the MMD in Fig. 7, and the infinite layers results were generated by drawing wavefunctions from the Porter-Thomas distribution[114].

the loss cannot distinguish two distributions that agree on all k-order marginals but disagree on higher-order marginals. Thus losses composed exclusively of local terms (with the conventional measurement strategy) cannot be used in generative modeling to learn complex higher-order correlations.

With a little thought it becomes clear that an exclusively global loss is also undesirable. Not only do such losses exhibit exponential concentration for unstructured circuits, they will also in general possess spurious minima in virtue of only probing global properties of the distribution (i.e., the average global parity), as shown in Fig. 5. Instead we advocate using *full-body* losses which contain both low and high-body terms, such as those obtained by averaging in Fig. 5. Even then, global contributions cannot be vanishingly small or else they will not be possible to resolve with a realistic shot budget.

For another example, one may aim to train a quantum generative model using a QGAN framework, where a Discriminator D provides a score $D(\boldsymbol{x})$ to every sample. The corresponding operator can then be written as $\mathcal{M} = \sum_{\boldsymbol{x}} D(\boldsymbol{x})|\boldsymbol{x}\rangle\langle\boldsymbol{x}|$. The Discriminator may have to initially implement an effectively low-bodied operator to facilitate initial gradients, but later in training become higher-bodied to learn global features. That is not to say that the Discriminator should only classify marginals of the bitstring such as in ref. 93. Rather, the architecture and initialization should be such that the operator $\mathcal{M}$ in the Pauli basis initially contains low-body terms but can include high-body terms during convergence. Interestingly, the interpolation from trainable to faithful could be naturally full-filled during training when the Generator and Discriminator are optimized in tandem.

Fine-tuning the interplay between the loss function gradients, density of local minima and the faithfulness of a generative loss is beyond the scope of this work, but is an important direction for future research. We especially emphasize the necessity to evaluate the implications of our results on models and datasets of practical relevance. In Section “Training on a HEP dataset” we take steps in this direction by investigating training a QCBM to model real data from the HEP domain.

Quantum strategies: quantum fidelity. While the classical fidelity in Eq. (8) is an explicit cost function, the quantum fidelity, defined in Eq. (15), allows for a simple known quantum estimation strategy. Key to the quantum fidelity loss is to interpret the training distribution as a target state $|\phi\rangle = \sum_{\boldsymbol{x}} \sqrt{\tilde{p}(\boldsymbol{x})}|\boldsymbol{x}\rangle$. The QCBM model loss can then be rewritten as the expectation of an observable, e.g., in the form of Eq. (19), with $\rho = |\phi\rangle\langle\phi|$ and $O = |\boldsymbol{0}\rangle\langle\boldsymbol{0}|$ being the all-zero projective measurement. Crucially, as $O = |\boldsymbol{0}\rangle\langle\boldsymbol{0}|$ is a global projector, the quantum fidelity is subject to a globality-induced BPs[32] and the loss exponentially concentrates towards one[24]. That is, we have

$$\mathrm{Var}_{\boldsymbol{\theta}}[\mathcal{L}_{QF}(\boldsymbol{\theta})] \in \mathcal{O}(1/b^n). \tag{51}$$

This global-measurement-induced BP can however be avoided by localizing $\mathcal{L}_{QF}(\boldsymbol{\theta})$. That is, we replace the global projective measurement $|0\rangle\langle 0|$ with its local version $H_L = \frac{1}{n}\sum_{i=1}^{n} |0_i\rangle\langle 0_i| \otimes \mathbb{1}_{\bar{i}}$, where $\bar{i}$ indicates all qubits except qubit i. The new localized version of the quantum fidelity loss is given by

$$\mathcal{L}_{QF}^{(L)}(\boldsymbol{\theta}) = 1 - \langle\phi|U(\boldsymbol{\theta})H_L U^{\dagger}(\boldsymbol{\theta})|\phi\rangle. \tag{52}$$

This local loss is faithful to its global variant for product state training in the sense that it vanishes under the same conditions[94], i.e., when the QCBM distribution matches the data distribution exactly. However, it enjoys trainability guarantees via the results of ref. 32. This implies that, unlike the MMD and other classical losses that utilize the conventional measurement strategy, the LQF can effectively distinguish between high-order marginals even at $k = 1$ bodyness. However, although the local loss function can evade global measurement-induced BPs, it still suffers under BPs from other sources, such as expressibility or noise. Additionally, it is not yet explored how practical a fidelity loss is for the purposes of generalizing from training data.

Figure 8 depicts numerical variance results for the fidelity loss on a range of datasets, circuit depths, and numbers of qubits. For all datasets, the LQF exhibits only polynomially decaying variance over random parameters when the quantum circuits are not too deep. As a reference, we additionally depict the global quantum fidelity which exponentially decays for all circuit depths.

The challenge now becomes how to estimate $\mathcal{L}_{QF}^{(L)}(\boldsymbol{\theta})$ using measurements from the quantum computer. The seemingly straight-forward approach is to prepare the initial state $|\phi\rangle$, evolve it under $U^{\dagger}(\boldsymbol{\theta})$, and then evaluate the observable defined by H_L through measurements in the computational basis. However, loading classical data into a quantum state $|\phi\rangle$ is not expected to be feasible in general. In Supplementary Note V, we propose an approach that can be used to estimate $\mathcal{L}_{QF}^{(L)}(\boldsymbol{\theta})$ using a series of Hadamard tests without needing to prepare $|\phi\rangle$. We note that, while in theory our approach requires a number of Hadamard tests that scales with the amount training data, we expect stochastic techniques, such as stochastic gradient descent[95], to be sufficient in practice.

Broader Implications. In this section, we have presented one example of a quantum strategy to measure a fidelity-based loss for quantum generative modeling. While this approach puts more load on

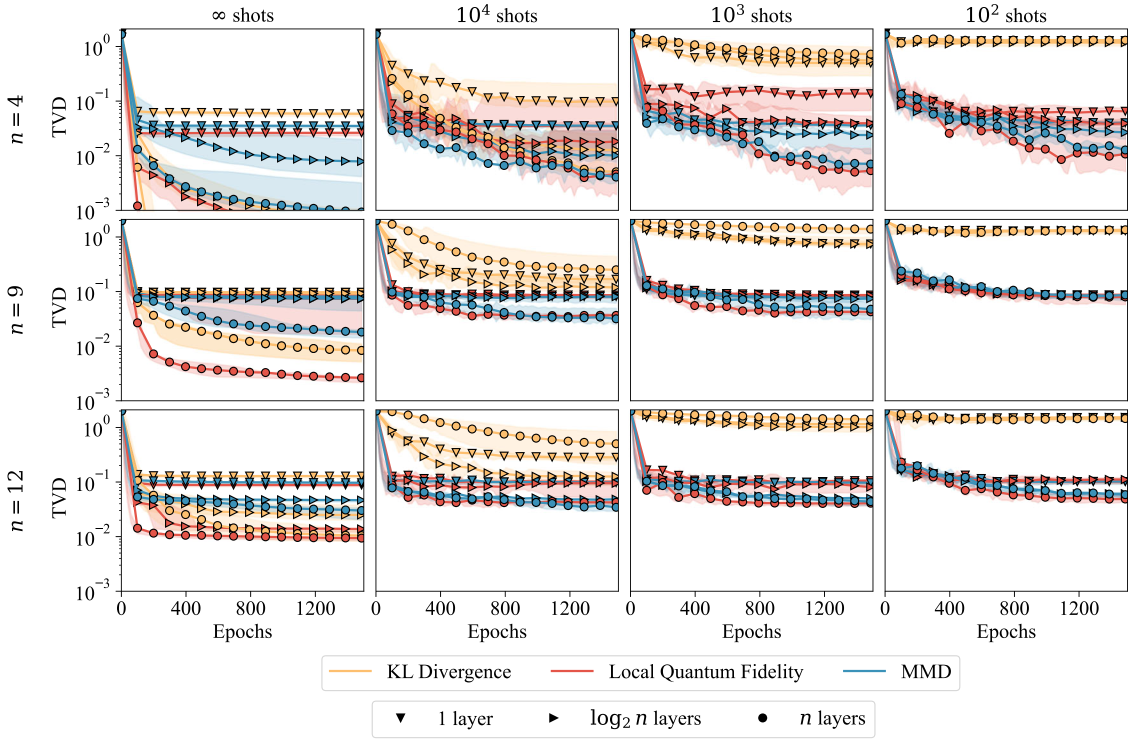

Fig. 9 | Finite-shot comparison of loss functions. TVD, computed with infinite statistics, on the training curve of the QCBMs with varying numbers of qubits $n = 4$, $n = 9$ and $n = 12$ (rows) and layers (symbols), where the gradients are computed with different numbers of shots (columns) for different loss function (colors).

the quantum computer as compared to losses employing the conventional measurement strategy, it enjoys simultaneous trainability and faithfulness to the target distribution.

An interesting extension would be to explore other quantum approaches for efficiently training QML models. One could for example attempt to compute the KL divergence or other explicit losses directly on the quantum computer. Although the implementation of non-linear operations on quantum computers has been demonstrated in refs. 96–98, we are not yet aware of quantum strategies beyond one related demonstration for the Rènyi divergence in ref. 46. One alternative approach would be to attempt to indirectly turn the QCBM into an explicit generative model by estimating its probabilities using amplitude amplification or other techniques. As discussed in Section “Large gradient variances are not enough”, with access to exact probabilities the KL divergence can provably avoid BPs even with unstructured circuits for certain target distributions.

Training on a HEP dataset

In this section, we perform realistic training of QCBMs on a more practical dataset which is derived from HEP colliders experiments. We compare the implicit cost functions MMD and LQF with the explicit KL divergence for an increasing number of circuit depth L and the number of qubits n, and across several measurement budgets. To summarize our results, we observe that the presence of shot noise causes the training with KLD to fail, while the MMD and LQF both hold up significantly better.

Dataset. We consider a dataset consisting of energy depositions in an ECAL[99]. The data was generated using a Monte Carlo approach (theGeant4 toolkit[100]), which accurately describes the ECAL detector behavior under a typically proton- proton collision at a LHC experiment. The dataset consists of the energy deposition on a $25 \times 25 \times 25$ grid, that we downsized to a two-dimensional grid of various sizes. The images are converted to a black and white scale by considering the pixel ‘hit’ if the energy deposition exceeds a certain threshold, which is chosen as one tenth of the mean energy deposit. We map each pixel to a qubit and take the state $|1\rangle$ to represent a hit. This dataset naturally has a polynomial support and thus is precisely the type of dataset that we might hope to learn using QML.

Training. We use a parametrized quantum circuit of the form

$$U(\boldsymbol{\theta}) = \left[\prod_{l=1}^{L} \mathrm{R}(\boldsymbol{\theta}_l) W_l(\boldsymbol{\alpha}_l)\right] \mathrm{R}(\boldsymbol{\theta}_0), \tag{53}$$

where $\mathrm{R}(\boldsymbol{\theta}_l)$ is a layer of arbitrary single qubit unitaries that can be parameterized using $3n$ Euler angles, $W(\boldsymbol{\alpha}_l) := \prod_{i=1}^{n-1} \mathrm{CX}_{i,i+1} \mathrm{RY}_i(\alpha_l^i) \mathrm{CX}_{i,i+1}$ acts as parametrized entangling gate with $\mathrm{CX}_{i,j}$ a CNOT gate between qubits i and j and $\mathrm{RY}_i(\alpha_l^i)$ a single qubit rotation of qubit i around the y-axis, and the parameters $\boldsymbol{\theta} = \{\boldsymbol{\theta}_l, \boldsymbol{\alpha}_l\}$. We use the TVD, see Eq. (7), as a common metric to assess the performance of each loss function. To verify performance accurately, we compute the TVD using exact simulation. The gradients for each loss are computed using the parameter shift rule[101] which provides estimates of the analytical gradient, and the parameters are updated with the ADAM[102] optimizer with a decaying learning rate $\mathrm{lr}(t) = \max(0.01e^{-\beta t}, 10^{-5})$, where t is the optimization step and $\beta = 0.005$ the decaying rate. The computation of the KLD is stabilized using a regulariser of $\epsilon = 10^{-6}$, which has been tuned through trial and error. To follow best-practices with the MMD, we average the gradient estimation over several different bandwidths

$$\boldsymbol{\sigma} = \begin{pmatrix} 0.01, & 0.1, & 0.25, & 0.5, & 1, & 10 \end{pmatrix} n, \tag{54}$$

which incurs no additional quantum resources. This makes the loss full-bodied and thus keeps the model trainable while aiding convergence. We note that these are likely not the optimal bandwidths to average over but it demonstrates the best-practice approach.

Results. Figure 9 shows the TVD, computed with infinite statistics, on the training curve of the QCBMs with varying numbers of qubits $n \in \{4, 9, 12\}$ (rows) and layers (symbols), where the gradients are computed with different numbers of shots (columns) for different loss function (colors). The lines denote the median over ten random

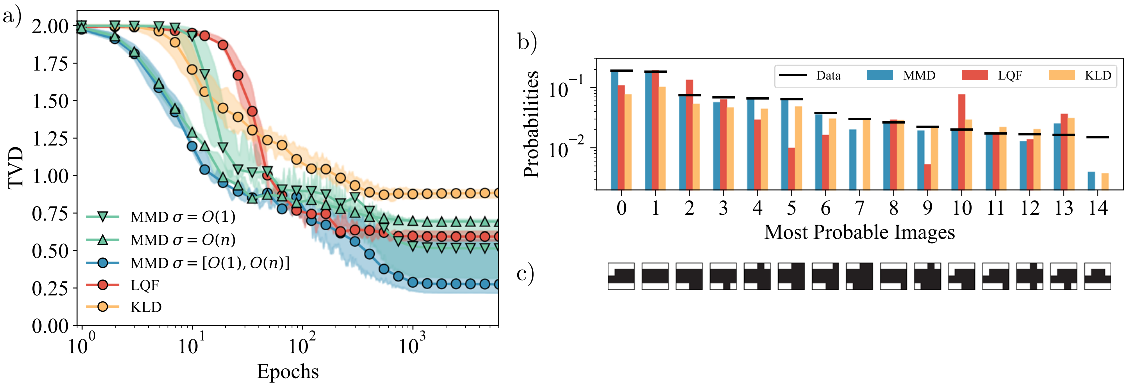

Fig. 10 | 16 qubit QCBM training. a Exact TVD computed during training of $n = 16$ QCBMs with $\log_2 n = 4$ layers and 100 shots per loss evaluation. We report median TVD values and 25% to 75% percentiles for the MMD, LQF and KLD loss functions. For the MMD, bandwidths $\sigma = 0.01$, $n/4$, and n are used to both showcase improved trainability of large σ and improved convergence of small σ. **b** Histograms of the trained QCBMs on the 15 most probable images, which are shown in panel (**c**). The black lines denote the training dataset probabilities.

parameter initialization while the shaded area denotes the 25% to 75% percentile. We observe that the performance of the KLD quickly deteriorates as the number of shots is reduced while the MMD and LQF remain more stable. We further observe that increasing the expressivity of the QCBM from $\log_2 n$ to n layers does not lead to a significant increase in performance for a low number of shots.

To demonstrate the scalability to larger systems, we also train a $n = 16$ QCBM with $\log_2 n = 4$ layers and 100 shots per function evaluation. The quantum circuit has a linear entangling topology and the initial parameters are chosen uniformly at random. We highlight that the downsized data at $n = 16$ is structurally different to the one in Fig. 9, which can yield quantitatively different results. In panel (a) of Fig. 10, we depict the median and 25%–75% percentiles for the KLD and LQF over 50 and 10 random repetitions respectively, whereas for the MMD we use 6 random repetitions per line. We also compare the MMD performance for different bandwidths $\sigma = 0.01$, $\sigma = n/4$, and a kernel averaging $\sigma = [0.01, n/4, n]$. We indeed see that large σ shows improved initial trainability and small σ improved convergence. In panel (b) we show the probability histograms of the 15 most occurring images in the dataset, as well as the final respective model probabilities. The corresponding 4×4 pixel images are displayed at the bottom panel (c).

In this 16-qubit example, it appears that the LQF is no longer performing on-par with the MMD, as was the case in Fig. 9 for smaller system sizes. A possible explanation is that one chooses all relative phases in the data state $|\phi\rangle = \sum_x \sqrt{\tilde{p}(x)}|x\rangle$, which strongly reduces the number of wavefunctions that minimize the LQF loss even though they may produce the desired measurement distribution. This may not only produce less solutions, it also enforces that the ansatz needs to be able to express exactly that state. While this could be leveraged using specialized real-valued ansätze, this is not attempted here. We conclude that the practical properties of the LQF loss as compared to implicit losses using the conventional measurement strategy are still to be studied in more detail.

To emphasize the importance of the size of the support, in Supplementary Note VI we also consider an exponential version of the dataset, by using a negative logarithm transformation. We find in this case that the KLD does not suffer from exponential concentration and can be trained. This explains the successes previously observed for training QCBMs using the KLD for small-scale problems. However, as the amount of classical training data cannot scale exponentially these successes are not relevant to larger, non-classically simulable, problems.

Discussion

In this work, we have introduced the notion of explicit and implicit losses, which broadly reflect the capabilities of explicit and implicit generative models[48]. We argue that these concepts provide a useful framework to understand the trainability of quantum generative models. In particular, we argue that the mismatch between the indirect access to probabilities provided by implicit models with the explicit probabilities required by explicit losses renders implicit models untrainable via explicit losses. More concretely, focusing our attention on QCBMs as a commonly used implicit model, we prove that pairwise explicit losses exponentially concentrate (Theorem 1). This result prohibits efficient training using a large class of commonly-used losses including the KL divergence, JS divergence, and the TVD. Such losses may however be usable with explicit "quantum" generative models such as tensor network Born machines[57–59].

Crucially, our results assume access to a polynomial (in the number of qubits) number of training data samples and measurements from the quantum circuit. With only moderate numbers of qubits, this assumption is unnecessary and explicit losses such as the KL divergence may appear to be trainable (see e.g., refs. 16,17,20–22). More generally, if we restrict the number of qubits used to classically simulable sizes, this assumption can be lifted and one could use quantum generative models purely for their efficient sampling capabilities. However, to harness the full potential of quantum generative modeling one surely wants to push to non-classically tractable problem sizes, at which point this assumption is essential. For example, even with only 50 qubits, access to ~$2^{50} \approx 10^{15}$ training samples or quantum measurements is unrealistic.

While formulated initially for random quantum circuits, the intuition underlying Theorem 1 suggests our no-go result extends to scenarios where the implicit generative model's measurement distribution only has polynomial support (e.g., near-identity initialization of the circuit[103]), as well as beyond the pairwise explicit form of the explicit loss. One exception may be if the quantum generative model has a strong inductive bias. Hence our work further motivates the search for new methods for constructing parameterized circuits with strong inductive biases (e.g., via warm starts[104–108] or incorporating symmetry constraints[28,42,109–112]).

In contrast to explicit losses, implicit losses are naturally suited to training implicit models. Within this line of thought, we have identified the MMD loss with a Gaussian kernel as a promising implicit loss for training QCBMs. We show that this loss can be interpreted as the expectation value of a quantum observable, where crucially the properties of the observable depend on the bandwidth parameter σ. In the common case where σ is independent of the system size, $\sigma \in \mathcal{O}(1)$, the observable becomes predominantly global and thus exponentially concentrates. Conversely, when σ scales linearly with the system size, $\sigma \in \Theta(n)$, the low-body interaction terms in the observable are largely dominant over the global terms and hence exhibit large gradients. We further use these insights to derive a rigorous polynomially lower bound on the MMD loss variance for a wide range of structured and unstructured circuits with small effective light cones.

Our main results for explicit and implicit losses assume the conventional strategy for estimating a generative loss function from an implicit model, where the model provides a set of samples in the computational basis, which are then used to estimate the loss in conjunction with the training data samples. While this is the standard classical strategy, quantum generative models can employ alternative quantum strategies by leveraging quantum computing power. As an example, we propose the LQF as a trainable loss function for generative modeling. Developing alternative quantum strategies for training quantum generative models is an interesting avenue for future research. A natural candidate might be, as suggested in ref. 13, to implement the MMD loss with a *quantum* kernel, where the kernel values themselves are estimated using quantum computers. While one could hope for a potential quantum advantage with this approach (especially when training on quantum data), there is the additional challenge that quantum kernels without inductive bias tend to exponentially concentrate[30].

To put our conclusions to the test, we studied how these loss functions perform in more practical scenarios with data derived from High Energy Physics experiments at the LHC. This dataset naturally satisfies our assumptions of a polynomial number of data samples at all system sizes. Our training results are found to be consistent with our theoretical predictions in which both the MMD and quantum fidelity losses significantly outperform the KLD loss when a strict measurement budget is employed.

Finally, while our work addresses the question of whether a given loss exhibits non-exponentially vanishing gradients, we stress that this is just one ingredient among many to ensure the success of quantum generative modeling. Of particular importance is the observation that models with local losses will generally struggle to learn global correlations due to the function's inability to distinguish high-order features in the data. Hence we advocate using *full*-body losses, which contain both low and high-body terms, for quantum generative modeling. More broadly, the ability of a model to successfully generalize will also presumably depend on the choice in loss, but this is beyond the scope of this work. Nonetheless, ensuring non-vanishing loss gradients and ensuring faithfulness of the loss function are critical steps since failing here precludes the successful training and generalization of quantum generative models. Hence, our work constitutes an important first step to understanding of the barriers that need to be overcome to achieve a quantum advantage in generative modeling.

Data availability

Data generated and analyzed during the current study are available from the corresponding author upon reasonable request.

Code availability

Code used for the current study is available from the corresponding author upon reasonable request.

Received: 28 July 2023; Accepted: 10 October 2024;
Published online: 13 November 2024

References

1. Harrow, A. W. & Montanaro, A. Quantum computational supremacy. *Nature* **549**, 203–209 (2017).
2. Biamonte, J. et al. Quantum machine learning. *Nature* **549**, 195–202 (2017).
3. Huang, H.-Y. et al. Quantum advantage in learning from experiments. *Science* **376**, 1182–1186 (2022).
4. Daley, A. J. et al. Practical quantum advantage in quantum simulation. *Nature* **607**, 667–676 (2022).
5. Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum* **2**, 79 (2018).
6. Harrow, A. W., Hassidim, A. & Lloyd, S. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.* **103**, 150502 (2009).
7. Lloyd, S., Mohseni, M. & Rebentrost, P. Quantum principal component analysis. *Nat. Phys.* **10**, 631–633 (2014).
8. Huang, H.-Y. et al. Power of data in quantum machine learning. *Nat. Commun.* **12**, 1–9 (2021).
9. Anschuetz, E. R., Hu, H.-Y., Huang, J.-L. & Gao, X. Interpretable quantum advantage in neural sequence learning. *PRX Quantum* **4**, 020338 (2023).
10. Alcazar, J., Leyton-Ortega, V. & Perdomo-Ortiz, A. Classical versus quantum models in machine learning: insights from a finance application. *Mach. Learn. Sci. Technol.* **1**, 035003 (2020).
11. Gili, K., Hibat-Allah, M., Mauri, M., Ballance, C. & Perdomo-Ortiz, A. Do quantum circuit born machines generalize? *Quantum Sci. Technol.* **8**, 035021 (2023).
12. Perdomo-Ortiz, A., Benedetti, M., Realpe-Gómez, J. & Biswas, R. Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers. *Quantum Sci. Technol.* **3**, 030502 (2018).
13. Coyle, B., Mills, D., Danos, V. & Kashefi, E. The born supremacy: quantum advantage and training of an ising born machine. *npj Quantum Inf.* **6**, 60 (2020).
14. Sweke, R., Seifert, J.-P., Hangleiter, D. & Eisert, J. On the quantum versus classical learnability of discrete distributions. *Quantum* **5**, 417 (2021).
15. Gao, X., Anschuetz, E. R., Wang, S.-T., Cirac, J. I. & Lukin, M. D. Enhancing generative models via quantum correlations. *Phys. Rev. X* **12**, 021037 (2022).
16. Rudolph, M. S. et al. Generation of high-resolution handwritten digits with an ion-trap quantum computer. *Phys. Rev. X* **12**, 031010 (2022).
17. Coyle, B. et al. Quantum versus classical generative modelling in finance. *Quantum Sci. Technol.* **6**, 024013 (2021).
18. Kiss, O., Grossi, M., Kajomovitz, E. & Vallecorsa, S. Conditional born machine for monte carlo event generation. *Phys. Rev. A* **106**, 022612 (2022).
19. Delgado, A. & Hamilton, K. E. Unsupervised quantum circuit learning in high energy physics. *Phys. Rev. D* **106**, 096006 (2022).
20. Hamilton, K. E., Dumitrescu, E. F. & Pooser, R. C. Generative model benchmarks for superconducting qubits. *Phys. Rev. A* **99**, 062323 (2019).
21. Leyton-Ortega, V., Perdomo-Ortiz, A. & Perdomo, O. Robust implementation of generative modeling with parametrized quantum circuits. *Quantum Mach. Intell.* **3**, 1–10 (2021).
22. Zhu, D. et al. Training of quantum circuits on a hybrid quantum computer, *Sci. Adv.* **5**, https://doi.org/10.1126/sciadv.aaw9918 (2019).
23. McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R. & Neven, H. Barren plateaus in quantum neural network training landscapes. *Nat. Commun.* **9**, 1–6 (2018).
24. Arrasmith, A., Holmes, Z., Cerezo, M. & Coles, P. J. Equivalence of quantum barren plateaus to cost concentration and narrow gorges. *Quantum Sci. Technol.* **7**, 045015 (2022).
25. Larocca, M. et al. Diagnosing barren plateaus with tools from quantum optimal control. *Quantum* **6**, 824 (2022).
26. Cerezo, M. & Coles, P. J. Higher order derivatives of quantum neural networks with barren plateaus. *Quantum Sci. Technol.* **6**, 035006 (2021).
27. Arrasmith, A., Cerezo, M., Czarnik, P., Cincio, L. & Coles, P. J. Effect of barren plateaus on gradient-free optimization. *Quantum* **5**, 558 (2021).
28. Holmes, Z. et al. Barren plateaus preclude learning scramblers. *Phys. Rev. Lett.* **126**, 190501 (2021).

29. Zhao, C. & Gao, X.-S. Analyzing the barren plateau phenomenon in training quantum neural networks with the ZX-calculus. *Quantum* **5**, 466 (2021).
30. Thanasilp, S., Wang, S., Cerezo, M. & Holmes, Z. Exponential concentration in quantum kernel methods. *Nat. Commun.* **15**, 5200 (2024).
31. Holmes, Z., Sharma, K., Cerezo, M. & Coles, P. J. Connecting ansatz expressibility to gradient magnitudes and barren plateaus. *PRX Quantum* **3**, 010313 (2022).
32. Cerezo, M., Sone, A., Volkoff, T., Cincio, L. & Coles, P. J. Cost function dependent barren plateaus in shallow parametrized quantum circuits. *Nat. Commun.* **12**, 1–12 (2021).
33. Marrero, C. O., Kieferová, M. & Wiebe, N. Entanglement-induced barren plateaus. *PRX Quantum* **2**, 040316 (2021).
34. Patti, T. L., Najafi, K., Gao, X. & Yelin, S. F. Entanglement devised barren plateau mitigation. *Phys. Rev. Res.* **3**, 033090 (2021).
35. Wang, S. et al. Noise-induced barren plateaus in variational quantum algorithms. *Nat. Commun.* **12**, 1–11 (2021).
36. Wang, S. et al. Can error mitigation improve trainability of noisy variational quantum algorithms? *Quantum* **8**, 1287 (2024).
37. Thanasilp, S., Wang, S., Nghiem, N. A., Coles, P. & Cerezo, M. Subtleties in the trainability of quantum machine learning models. *Quantum Mach. Intell.* **5**, 21 (2023).
38. Leone, L., Oliviero, S. F. E., Cincio, L. & Cerezo, M. On the practical usefulness of the hardware efficient ansatz. *Quantum* **8**, 1395 (2024).
39. Li, G., Ye, R., Zhao, X. & Wang, X. Concentration of data encoding in parameterized quantum circuits. *Adv. Neural Inf. Process. Syst.* **35**, 19456–19469 (2022).
40. Napp, J. Quantifying the barren plateau phenomenon for a model of unstructured variational ansätze, arXiv preprint arXiv:2203.06174 https://arxiv.org/abs/2203.06174 (2022).
41. Pesah, A. et al. Absence of barren plateaus in quantum convolutional neural networks. *Phys. Rev. X* **11**, 041011 (2021).
42. Larocca, M. et al. Group-invariant quantum machine learning. *PRX Quantum* **3**, 030341 (2022).
43. Tangpanitanon, J., Thanasilp, S., Dangniam, N., Lemonde, M.-A. & Angelakis, D. G. Expressibility and trainability of parametrized analog quantum systems for machine learning applications. *Phys. Rev. Res.* **2**, 043364 (2020).
44. Sharma, K., Cerezo, M., Cincio, L. & Coles, P. J. Trainability of dissipative perceptron-based quantum neural networks. *Phys. Rev. Lett.* **128**, 180505 (2022).
45. Rudolph, M. S. et al. Orqviz: visualizing high-dimensional landscapes in variational quantum algorithms, arXiv preprint arXiv:2111.04695 https://doi.org/10.48550/arXiv.2111.04695 (2021).
46. Kieferova, M., Carlos, O. M. & Wiebe, N. Quantum generative training using rényi divergences, arXiv preprint arXiv:2106.09567 https://arxiv.org/abs/2106.09567 (2021).
47. Coopmans, L. & Benedetti, M. On the sample complexity of quantum Boltzmann machine learning. *Communi. Phys.* **7**, 274 (2024).
48. Mohamed, S. & Lakshminarayanan, B. Learning in implicit generative models, arXiv preprint arXiv:1610.03483 https://arxiv.org/abs/1610.03483 (2016).
49. Benedetti, M. et al. A generative modeling approach for benchmarking and training shallow quantum circuits. *npj Quantum Inf.* **5**, 45 (2019).
50. Csiszar, I. *I*-divergence geometry of probability distributions and minimization problems. *Ann. Probab.* **3**, 146–158 (1975).
51. Gretton, A., Borgwardt, K. M., Rasch, M. J., Schölkopf, B. & Smola, A. A kernel two-sample test. *J. Mach. Learn. Res.* **13**, 723–773 (2012).
52. Gili, K., Mauri, M. & Perdomo-Ortiz, A. Generalization metrics for practical quantum advantage in generative models. *Phys. Rev. Appl.* **21**, 044032 (2022).
53. Cheng, S., Chen, J. & Wang, L. Information perspective to probabilistic modeling: Boltzmann machines versus born machines. *Entropy* **20**, 583 (2018).
54. Liu, J.-G. & Wang, L. Differentiable learning of quantum circuit born machines. *Phys. Rev. A* **98**, 062324 (2018).
55. Smolensky, P. Information processing in dynamical systems: foundations of harmony theory. in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations* 194–281 https://stanford.edu/~jlmcc/papers/PDP/Volume (MIT Press, 1986).
56. Hinton, G. E. A practical guide to training restricted Boltzmann machines. *Neural Networks: Tricks of the Trade* 2nd edn 599–619 https://doi.org/10.1007/978-3-642-35289-8_32 (2012).
57. Han, Z.-Y., Wang, J., Fan, H., Wang, L. & Zhang, P. Unsupervised generative modeling using matrix product states. *Phys. Rev. X* **8**, 031012 (2018).
58. Cheng, S., Wang, L., Xiang, T. & Zhang, P. Tree tensor networks for generative modeling. *Phys. Rev. B* **99**, 155131 (2019).
59. Vieijra, T., Vanderstraeten, L. & Verstraete, F. Generative modeling with projected entangled-pair states, arXiv preprint arXiv:2202.08177 https://doi.org/10.48550/arXiv.2202.08177 (2022).
60. Wall, M. L., Abernathy, M. R. & Quiroz, G. Generative machine learning with tensor networks: benchmarks on near-term quantum computers. *Phys. Rev. Res.* **3**, 023010 (2021).
61. Čepaitė, I., Coyle, B. & Kashefi, E. A continuous variable born machine. *Quantum Mach. Intell.* **4**, 6 (2022).
62. Benedetti, M., Coyle, B., Fiorentini, M., Lubasch, M. & Rosenkranz, M. Variational inference with a quantum computer. *Phys. Rev. Appl.* **16**, 044057 (2021).
63. Gili, K., Sveistrys, M. & Ballance, C. Introducing nonlinear activations into quantum generative models. *Phys. Rev. A* **107**, 012406 (2023).
64. Jerbi, S. et al. Quantum machine learning beyond kernel methods. *Nat. Commun.* **14**, 517 (2023).
65. Van Den Oord, A., Kalchbrenner, N. & Kavukcuoglu, K. Pixel recurrent neural networks. In *Proc. 33rd International Conference on International Conference on Machine Learning* Vol. 48, 1747–1756 https://doi.org/10.5555/3045390.3045575 (2016).
66. Rumelhart, D. E. & McClelland, J. L. Learning internal representations by error propagation. in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition: Foundations* 318–362 https://ieeexplore.ieee.org/document/6302929 (1987).
67. Goodfellow, I. et al. Generative adversarial nets. In *Proc. Advances in Neural Information Processing Systems* Vol. 27, https://proceedings.neurips.cc/paper_files/paper/2014/file/5ca3e9b122f61f8f06494c97b1afccf3-Paper.pdf (2014).
68. Csiszár, I. On information-type measure of difference of probability distributions and indirect observations. *Stud. Sci. Math. Hung.* **2**, 299–318 (1967).
69. Kullback, S. & Leibler, R. A. On information and sufficiency. in *The Annals of Mathematical Statistics* Vol. 22, https://doi.org/10.1214/aoms/1177729694 (1951).
70. Lin, J. Divergence measures based on the Shannon entropy. *IEEE Trans. Inf. theory* **37**, 145–151 (1991).
71. Rényi, A. On measures of entropy and information. In *Proc. Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics* 547–562 https://projecteuclid.org/proceedings/berkeley-symposium-on-mathematical-statistics-and-probability/Proceedings-of-the-Fourth-Berkeley-Symposium-on-

Mathematical-Statistics-and/Chapter/On-Measures-of-Entropy-and-Information/bsmsp/1200512181 (1961).
72. Gibbs, J. et al. Long-time simulations for fixed input states on quantum hardware. *npj Quantum Inf.* **8**, 135 (2022).
73. Gibbs, J. et al. Dynamical simulation via quantum machine learning with provable generalization. *Phys. Rev. Res.* **6**, 013241 (2024).
74. Caro, M. C. et al. Out-of-distribution generalization for learning quantum dynamics. *Nat. Commun.* **14**, 3751 (2023).
75. Volkoff, T., Holmes, Z. & Sornborger, A. Universal compiling and (no-)free-lunch theorems for continuous-variable quantum learning. *PRX Quantum* **2**, 040327 (2021).
76. Barenco, A. et al. Stabilization of quantum computations by symmetrization. *SIAM J. Comput.* **26**, 1541–1557 (1997).
77. Garcia-Escartin, J. C. & Chamorro-Posada, P. Swap test and hong-ou-mandel effect are equivalent. *Phys. Rev. A* **87**, 052330 (2013).
78. Lloyd, S. & Weedbrook, C. Quantum generative adversarial learning. *Phys. Rev. Lett.* **121**, 040502 (2018).
79. Zoufal, C., Lucchi, A. & Woerner, S. Quantum generative adversarial networks for learning and loading random distributions. *npj Quantum Inf.* **5**, 103 (2019).
80. Situ, H., He, Z., Wang, Y., Li, L. & Zheng, S. Quantum generative adversarial network for generating discrete distribution. *Inf. Sci.* **538**, 193–208 (2020).
81. Bravo-Prieto, C. et al. Style-based quantum generative adversarial networks for Monte Carlo events. *Quantum* **6**, 777 (2022).
82. Niu, M. Y. et al. Entangling quantum generative adversarial networks. *Phys. Rev. Lett.* **128**, 220505 (2022).
83. Stilck França, D. & Garcia-Patron, R. Limitations of optimization algorithms on noisy quantum devices. *Nat. Phys.* **17**, 1221–1227 (2021).
84. Chang, T.-J., Meade, N., Beasley, J. E. & Sharaiha, Y. M. Heuristics for cardinality constrained portfolio optimisation. *Comput. Oper. Res.* **27**, 1271–1302 (2000).
85. Alcazar, J., Ghazi Vakili, M., Kalayci, C. B. & Perdomo-Ortiz, A. Enhancing combinatorial optimization with classical and quantum generative models. *Nat. Commun*. **15**, 2761 (2024).
86. Li, C.-L., Chang, W.-C., Cheng, Y., Yang, Y. and Póczos, B. Mmd gan: Towards deeper understanding of moment matching network. In *Proc. 31st International Conference on Neural Information Processing Systems* 2200–2210 https://proceedings.neurips.cc/paper/2017/file/dfd7468ac613286cdbb40872c8ef3b06-Paper.pdf (Red Hook, 2017).
87. Wang, W., Sun, Y. & Halgamuge, S. Improving MMD-GAN training with repulsive loss function, arXiv preprint arXiv:1812.09916 https://arxiv.org/abs/1812.09916 (2018).
88. Li, H., Pan, S. J., Wang, S. & Kot, A. C. Domain generalization with adversarial feature learning, In *Proc. 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition* 5400–5409 https://doi.org/10.1109/CVPR.2018.00566 (2018).
89. Letcher, A., Woerner, S. & Zoufal, C. Tight and efficient gradient bounds for parameterized quantum circuits. *Quantum* **8**, 1484 (2024).
90. Gretton, A. et al. Optimal kernel choice for large-scale two-sample tests. In *Proc. Advances in Neural Information Processing Systems* https://proceedings.neurips.cc/paper_files/paper/2012/file/dbe272bab69f8e13f14b405e038deb64-Paper.pdf (2012).
91. Sutherland, D. J. et al. Generative models and model criticism via optimized maximum mean discrepancy, arXiv preprint arXiv:1611.04488 https://arxiv.org/abs/1611.04488 (2016).
92. Garreau, D., Jitkrittum, W. & Kanagawa, M. Large sample analysis of the median heuristic, arXiv preprint arXiv:1707.07269 https://arxiv.org/abs/1707.07269 (2017).
93. Leadbeater, C., Sharrock, L., Coyle, B. & Benedetti, M. F-divergences and cost function locality in generative modelling with quantum circuits. *Entropy* **23**, 1281 (2021).
94. Khatri, S. et al. Quantum-assisted quantum compiling. *Quantum* **3**, 140 (2019).
95. Sweke, R. et al. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum* **4**, 314 (2020).
96. Holmes, Z., Coble, N. J., Sornborger, A. T. & Subaşı, Y. Nonlinear transformations in quantum computation. *Phys. Rev. Res.* **5**, 013105 (2023).
97. Gilyén, A., Su, Y., Low, G. H. & Wiebe, N. Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics. In *Proc. 51st Annual ACM SIGACT Symposium on Theory of Computing* 193–204 https://dl.acm.org/doi/abs/10.1145/3313276.3316366 (2019).
98. Martyn, J. M., Rossi, Z. M., Tan, A. K. & Chuang, I. L. Grand unification of quantum algorithms. *PRX Quantum* **2**, 040203 (2021).
99. Pierini, M. & Zhang, M. CLIC Calorimeter 3D images: electron showers at fixed angle, https://doi.org/10.5281/zenodo.3603122 (2020).
100. Agostinelli, S. et al. Geant4—a simulation toolkit. *Nucl. Instrum. Methods Phys. Res. Sect. A Accel. Spectrom. Detect. Assoc. Equip.* **506**, 250–303 (2003).
101. Schuld, M., Bergholm, V., Gogolin, C., Izaac, J. & Killoran, N. Evaluating analytic gradients on quantum hardware. *Phys. Rev. A* **99**, 032331 (2019).
102. Kingma, D. P. & Ba, J. Adam: a method for stochastic optimization. In *Proc. 3rd International Conference on Learning Representations, ICLR, Conference Track Proceedings* http://arxiv.org/abs/1412.6980 (2015).
103. Grant, E., Wossnig, L., Ostaszewski, M. & Benedetti, M. An initialization strategy for addressing barren plateaus in parametrized quantum circuits. *Quantum* **3**, 214 (2019).
104. Sauvage, F. et al. Flip: a flexible initializer for arbitrarily-sized parametrized quantum circuits, arXiv preprint arXiv:2103.08572 https://arxiv.org/abs/2103.08572 (2021).
105. Liu, H.-Y., Sun, T.-P., Wu, Y.-C., Han, Y.-J. & Guo, G.-P. Mitigating barren plateaus with transfer-learning-inspired parameter initializations. *N. J. Phys.* **25**, 013039 (2023).
106. Cheng, M. H. et al. Clifford circuit initialisation for variational quantum algorithms, arXiv preprint arXiv:2207.01539 https://arxiv.org/abs/2207.01539 (2022).
107. Mitarai, K., Suzuki, Y., Mizukami, W., Nakagawa, Y. O. & Fujii, K. Quadratic clifford expansion for efficient benchmarking and initialization of variational quantum algorithms. *Phys. Rev. Res.* **4**, 033012 (2022).
108. Rudolph, M. S. et al. Synergistic pretraining of parametrized quantum circuits via tensor networks. *Nat. Commun.* **14**, 8367 (2023).
109. Schatzki, L., Larocca, M., Nguyen, Q. T., Sauvage, F. & Cerezo, M. Theoretical guarantees for permutation-equivariant quantum neural networks. *npj Quantum Inf.* **10**, 12 (2024).
110. Nguyen, Q. T. et al. Theory for equivariant quantum neural networks. *PRX Quantum* **5**, 020328 (2024).
111. Ragone, M. et al. Representation theory for geometric quantum machine learning, arXiv preprint arXiv:2210.07980 https://arxiv.org/abs/2210.07980 (2022).
112. Meyer, J. J. et al. Exploiting symmetry in variational quantum machine learning. *PRX Quantum* **4**, 010328 (2023).
113. Igel, C., Hansen, N. & Roth, S. Covariance matrix adaptation for multi-objective optimization. *Evolut. Comput.* **15**, 1–28 (2007).
114. Porter, C. E. & Thomas, R. G. Fluctuations of nuclear reaction widths. *Phys. Rev.* **104**, 483 (1956).

Acknowledgements

The authors would like to acknowledge Christa Zoufal, Samson Wang, and Marco Cerezo for helpful discussions. S.T. and Z.H. acknowledge support from the Sandoz Family Foundation-Monique de Meuron program for Academic Promotion. O.K., S.V., and M.G. are supported by CERN through the CERN Quantum Technology Initiative.

Author contributions

M.S.R., S.L., and S.T. contributed equally to this work. The project was conceived by M.S.R., S.T., and Z.H. Theoretical results were proved by M.S.R., S.L., S.T., O.S., and Z.H. Numerical implementations were performed by M.S.R. and O.K. The practical application of QCBMs was conceived and guided by S.V. and M.G. All authors contributed to the scientific discussions. The manuscript was written by all authors.

Competing interests

The authors declare no competing interests.

Additional information

Supplementary information The online version contains supplementary material available at https://doi.org/10.1038/s41534-024-00902-0.

Correspondence and requests for materials should be addressed to Supanut Thanasilp or Zoë Holmes.

Reprints and permissions information is available at http://www.nature.com/reprints

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.