Quantum Generative Model with Optimal Transport

Hiroyuki Tezuka[1,2,3] Shumpei Uno[4] Naoki Yamamoto[2,5,a)]

Abstract: Generative model is an unsupervised machine learning framework, that exhibits strong performance in imaging or anomaly detection in classical machine learning regime. Recently we find several quantum version of generative model, some of which are even proven to have quantum advantage. However, those proposals have a strict limitation; that is, the quantum state to be learned (i.e., the quantum state that the model produces) is limited to a single quantum state, and thus those methods are not applicable to a set of quantum states. In this paper, we propose a quantum generative model that can learn a set of quantum state, in an unsupervised machine learning framework. The key idea is to introduce a loss function calculated based on optimal transport distance, i.e. Wasserstein distance. We then apply the proposed method to an anomaly detection task, that cannot be handled via existing methods. The proposed model paves the way for a wide application such as the health check of quantum devices and efficient initialization of quantum computation.

Keywords: quantum machine learning, quantum generative model, optimal transport, anomaly detection

1. Introduction

In the recent great progress of quantum algorithms for both noisy near-term and future fault-tolerant quantum devices, particularly the quantum machine learning (QML) attracts huge attention. QML is largely categorised into two regimes in view of the type of data, which can be roughly called classical data and quantum data. The former has a conventional meaning used in the classical case; for the supervised learning scenario, e.g., a quantum system is trained to give a prediction for a given classical data such as an image. As for the latter, on the other hand, the task is to predict some property for a given quantum state drawn from a set of states, e.g. the phase of a many-body quantum state, again in the supervised learning scenario. Thanks to the obvious difficulty to directly represent a huge quantum state classically, some quantum advantage have been proven in QML for quantum data [1–3].

In the above paragraph we used the supervised learning setting to explain the difference of classical and quantum data. But the success of unsupervised learning, particularly the generative modeling, in classical machine learning is of course notable; actually a variety of algorithms have demonstrated strong performance in several applications, such as image generation [4–6], molecular design [7], and anomaly detection [8]. Hence, it is quite reasonable that several quantum unsupervised learning algorithms have been actively developed, such as quantum circuit born machine (QCBM) [9, 10], quantum generative adversarial network (QGAN) [11, 12], and quantum autoencoder (QAE) [13, 14]. Also, Refs. [15, 16] studied the generative modeling problem for quantum data. That is, the task is to construct a model quantum system producing a set of quantum states that approximates a given quantum dataset. The model quantum system needs to contain latent variables, the change of which corresponds to the change of output quantum state of the system. In classical case, such generative model characterized by latent variables is called the implicit model. Thus, training an implicit model is executed by minimizing a cost that measures a distance between the model dataset and training dataset. The transportation distance, typically the Wasserstein distance, suffices this purpose for measuring the distance of two ensembles; actually the quantum version of Wasserstein distance was proposed in [17] and was applied to construct a generative model for quantum data in QGAN framework [18].

Along this line of research, in this paper we also focus on the generative modeling problem for quantum data. We are motivated from the fact that the Wasserstein distance employed in the above-mentioned existing works compresses each element of the quantum dataset to make a single mixed state (density matrix), and measure the distance between two mixed states corresponding to the training and model dataset. This is clearly problematic, because in general there may be a lot of information loss when converting the dataset to a single mixed state; for instance, single qubit pure states uniformly distributed on the equator of the Bloch sphere are compressed to a maximally mixed state. It is obvious that learning a single mixed state state does not mean learning the original dataset. In this paper, hence, we propose a new quantum Wasserstein distance, which directly measures the distance be-

[1] Sony Group Corporation, 1-7-1 Konan, Minato-ku, Tokyo, 108-0075, Japan
[2] Quantum Computing Center, Keio University, Hiyoshi 3-14-1, Kohoku-ku, Yokohama 223-8522, Japan
[3] Graduate School of Science and Technology, Keio University, 3-14-1 Hiyoshi, Kohoku-ku, Yokohama, Kanagawa, 223- 8522, Japan
[4] Mizuho Research & Technologies, Ltd., 2-3 Kanda-Nishikicho, Chiyoda-ku, Tokyo, 101-8443, Japan
[5] Department of Applied Physics and Physico-Informatics, Keio University, Hiyoshi 3-14-1, Kohoku-ku, Yokohama 223-8522, Japan
[a)] yamamoto@appi.keio.ac.jp

ⓒ 2022 Information Processing Society of Japan

tween quantum datasets. The generative modeling problem can then be executed by minimizing this distance.

Based on such concept, we propose the generative modeling algorithm with parameterized quantum circuit (PQC). One of the crucial problem of QML with PQC is the curse of dimensionality due to the vanishing gradient problem, called barren plateau (BP), under the training process. To avoid that problem, we introduce a local cost into the ground cost of Wasserstein distance. We clarify several key properties, and verify the model with theoretical analysis and thorough numerical simulation. Finally, we demonstrate an application for anomaly detection of quantum data by numerical simulation.

2. Preliminaries

Here, we first describe the implicit generative model of classical machine learning in Sec. 2.1. Next, in Sec. 2.2, we introduce the Optimal Transport Loss, which is one of the promising candidates for the cost function of generative models. Finally, in Sec. 2.3, we briefly describe the vanishing gradient problem of variational quantum algorithms and one of the solutions proposed in [19].

2.1 Implicit Generative Model

The goal of the generative model is to approximate the distribution behind the training data. More concretely, with the probability distribution $\alpha(\boldsymbol{x})$ behind the training data $\{\boldsymbol{x}_i\}_{i=1}^{N_r} \in \mathcal{X}^{N_r}$ and the parameterized probability model $\beta_{\boldsymbol{\theta}}(\boldsymbol{x})$, the goal is to learn the parameters $\boldsymbol{\theta}$ that minimize the appropriate loss function based on training data. In particular, implicit generative models usually assume that the distribution behind the training data resides on a relatively low-dimensional manifold, i.e., implicit generative models are expressed through maps of random latent variables $\boldsymbol{z} = \{z_i\}_{i=1}^{N_z}$ onto the sample space $\mathcal{X}$, where the latent variables reside in a latent space $\mathcal{Z}$ whose dimension N_z is significantly smaller than the that of sample space N_x. The latent variables $\boldsymbol{z}$ are random variables which usually assumed to follow a well-known distribution $\gamma(\boldsymbol{z})$, such as a uniform distribution or Gaussian distribution. The implicit generative models are trained so that the samples generated from the model distribution are close to the training data, by adjusting the parameters $\boldsymbol{\theta}$ to minimize some appropriate cost function $\mathcal{L}$:

$$\theta^\star = \arg\min_{\boldsymbol{\theta}} \mathcal{L}(\hat{\alpha}_{N_r}, \hat{\beta}_{\boldsymbol{\theta},N_g}), \tag{2.1}$$

where $\hat{\beta}_{\boldsymbol{\theta},N_g} = G_{\boldsymbol{\theta}}\#\hat{\gamma}_{N_g}$ is a probability distribution induced by the *push-forward operator*# [20], which is intuitively a probability distribution on $\mathcal{X}$ moved from the distribution $\hat{\gamma}_{N_g}$ on $\mathcal{Z}$ through the map $G_{\boldsymbol{\theta}}$. $\hat{\alpha}_{N_r}(\boldsymbol{x})$ and $\hat{\gamma}_{N_g}(\boldsymbol{z})$ denote empirical distributions defined by using sampled data $\{\boldsymbol{x}_i\}_{i=1}^{N_r}$ and $\{\boldsymbol{z}_i\}_{i=1}^{N_g}$, which follow the probability distribution $\alpha(\boldsymbol{x})$ and $\gamma(\boldsymbol{z})$, respectively:

$$\begin{aligned}\hat{\alpha}_{N_r}(\boldsymbol{x}) &= \frac{1}{N_r}\sum_{i=1}^{N_r}\delta(\boldsymbol{x}-\boldsymbol{x}_i),\\ \hat{\gamma}_{N_g}(\boldsymbol{z}) &= \frac{1}{N_g}\sum_{i=1}^{N_g}\delta(\boldsymbol{z}-\boldsymbol{z}_i),\end{aligned} \tag{2.2}$$

2.2 Optimal Transport Loss

Optimal Transport Loss has recently attracted attention in various fields such as image analysis, natural language processing, and finance [20–23]. In particular, Optimal Transport Loss is widely used as a loss function of generative models, because it can be applicable when the support of probability distributions do not match and it can naturally incorporate the "distance" in the sample space $\mathcal{X}$ [24–27].Optimal Transport Loss is defined as the minimum cost of moving a probability distribution α to another distribution β:

Definition 1 (Optimal Transport Loss [28])**.**

$$\begin{aligned}&\mathcal{L}_c(\alpha,\beta) = \min_{\pi}\int c(\boldsymbol{x},\boldsymbol{y})d\pi(\boldsymbol{x},\boldsymbol{y}),\\ &\text{subject to}\quad \int\pi(\boldsymbol{x},\boldsymbol{y})d\boldsymbol{x} = \beta(y),\ \int\pi(\boldsymbol{x},\boldsymbol{y})d\boldsymbol{y} = \alpha(\boldsymbol{x}),\\ &\qquad\qquad \pi(\boldsymbol{x},\boldsymbol{y}) \geq 0,\end{aligned} \tag{2.3}$$

where $c(\boldsymbol{x},\boldsymbol{y}) \geq 0$ is a non-negative function on $\mathcal{X}\times\mathcal{X}$ that represents the transport cost from $\boldsymbol{x}$ to $\boldsymbol{y}$, and is called *ground cost*. Also, we call the set of couplings π which minimizes Eq.(2.3) as *optimal transport plan*. In general, Optimal Transport Loss does not meet the axioms of metric between the probability distributions, but it is known to meet the axioms when the ground cost is related to metric functions as follows:

Definition 2 (p-Wasserstein distance [29])**.** *When the ground cost $c(\boldsymbol{x},\boldsymbol{y})$ is expressed as $c(\boldsymbol{x},\boldsymbol{y}) = d(\boldsymbol{x},\boldsymbol{y})^p$ with a metric function $d(\boldsymbol{x},\boldsymbol{y})$, the p-Wasserstein distance is defined as*

$$\mathcal{W}_p(\alpha,\beta) = \mathcal{L}_{d^p}(\alpha,\beta)^{1/p}. \tag{2.4}$$

p-Wasserstein distance satisfies the properties of metric between probability distributions, i.e., for any probability distributions α,β,γ, p-Wasserstein distance $\mathcal{W}_p$ satisfies positivity: $\mathcal{W}_p(\alpha,\beta) \geq 0$, symmetricity: $\mathcal{W}_p(\alpha,\beta) = \mathcal{W}_p(\beta,\alpha)$, non-degeneracy: $\mathcal{W}_p(\alpha,\beta) = 0 \Leftrightarrow \alpha = \beta$, and triangle inequality: $\mathcal{W}_p(\alpha,\gamma) \leq \mathcal{W}_p(\alpha,\beta) + \mathcal{W}_p(\beta,\gamma)$.

For the learning of generative models with Optimal Transport Loss, it is usually hard to directly handle the probability distribution behind the training data α or that of the generative models $\beta_{\boldsymbol{\theta}}$. Instead, we can only use the sampled data from those distributions, and usually approximate the Optimal Transport Loss by using the empirical distribution Eq. (2.2):

Definition 3 (Empirical estimator for Optimal Transport Loss [29])**.**

$$\begin{aligned}\mathcal{L}_c(\alpha,\beta_{\boldsymbol{\theta}}) &\simeq \mathcal{L}_c\left(\hat{\alpha}_n, \hat{\beta}_{\boldsymbol{\theta},N_g}\right)\\ &= \min_{\{\pi_{i,j}\}_{i,j=1}^{N_r,N_g}}\sum_{i,j=1}^{N_r,N_g} c(\boldsymbol{x}_i, G_{\boldsymbol{\theta}}(\boldsymbol{z}_j))\pi_{i,j},\\ \text{subject to}\quad &\sum_{i=1}^{N_r}\pi_{i,j} = \frac{1}{N_g},\ \sum_{j=1}^{N_g}\pi_{i,j} = \frac{1}{N_r},\pi_{i,j} \geq 0.\end{aligned} \tag{2.5}$$

This empirical estimator converges as $\mathcal{L}_c(\hat{\alpha}_{N_r}, \hat{\beta}_{\boldsymbol{\theta},N_r}) \to \mathcal{L}_c(\alpha,\beta_{\boldsymbol{\theta}})$ in the limit $N_r = N_g \to \infty$. In general, the speed of convergence of the empirical optimal transport loss is very slow ($O(n^{-1/N_x})$ with the dimension of the sample space N_x [30]), but

p-Wasserstein distance have the following convergence law [31].

Theorem 4 (Convergence rate of p-Wasserstein distance)**.** *For the upper Wasserstein dimension $d_p^*(\alpha)$(Definition 4 of [31]) of the probability distribution α, the following expression holds when s is larger than $d_p^*(\alpha)$:*

$$\mathbb{E}\left[\mathcal{W}_p(\alpha, \hat{\alpha}_{N_r})\right] \lesssim O(N_r{}^{-1/s}), \tag{2.6}$$

where the expectation $\mathbb{E}$ is taken with respect to the random sample data within the empirical distribution $\hat{\alpha}_{N_r}$. Intuitively, the upper Wasserstein dimension $d_p^*(\alpha)$ is the support dimension of the probability distribution α, which corresponds to the dimension of the latent space N_z in the implicit generative model.

Exploiting the metric properties of the p-Wasserstein distance, the following corollaries are immediately derived from Theorem 4:

Corollary 5 (Convergence rate of p-Wasserstein distance between identical empirical distributions)**.** *Let $\hat{\alpha}_{1,N_r}$ and $\hat{\alpha}_{2,N_r}$ be empirical distributions of α with different N_r random samples. Then the following expression holds for $s > d_p^*(\alpha)$.*

$$\mathbb{E}\left[\mathcal{W}_p(\hat{\alpha}_{1,N_r}, \hat{\alpha}_{2,N_r})\right] \lesssim O(N_r{}^{-1/s}). \tag{2.7}$$

Corollary 6 (Convergence rate of p-Wasserstein distance between different empirical distributions)**.** *Suppose that the larger of the upper Wasserstein dimension of the probability distributions α and $\beta_{\boldsymbol{\theta}}$ is d_p^*, then the following expression holds for $s > d_p^*$.*

$$\mathbb{E}\left[\left|\mathcal{W}_p(\alpha,\beta_{\boldsymbol{\theta}}) - \mathcal{W}_p(\hat{\alpha}_{N_r},\hat{\beta}_{\boldsymbol{\theta},N_r})\right|\right] \lesssim O(N_r{}^{-1/s}). \tag{2.8}$$

These corollaries indicates that empirical estimator of Eq. (2.5) is a good estimator if the intrinsic dimension of the training data and the dimension of the latent space N_z are sufficiently small. In Sec. 3.2.1, we numerically confirm that these convergence law hold true even in the case of the proposed loss described below, which is not the p-Wasserstein distance.

2.3 Vanishing Gradient Problem of Variational Quantum Algorithms

For the quantum case, it is necessary to consider the transportation cost from a state $|\psi\rangle$ to another state $|\phi\rangle$ in order to employ the Optimal Transport Loss as the loss function of generative model. One of the candidates for such a ground cost is the trace distance:

Definition 7 (Trace distance for pure states [32])**.**

$$c_{\mathrm{tr}}(|\psi\rangle, |\phi\rangle) = \sqrt{1 - |\langle\psi|\phi\rangle|^2}. \tag{2.9}$$

Since the trace distance satisfies the axioms of metric, it is possible to define the p-Wasserstein distances from this ground cost, which allow us to use various useful properties, such as convergence speed described in the previous subsection. Further, it is relatively easy to obtain the trace distance with quantum computer, such as by swap test [33] or inversion test [34].

However, because the trace distance is a global observable, learning with the trace distance is known to suffer from the curse of dimensionality due to the vanishing gradient problem [35, 36]. Namely, when learning the quantum state with a global cost function using a hardware efficient ansatz, the expected magnitude of the gradient with respect to the parameters decreases exponentially with the number of bits n. This indicates that the number of measurements(shots) N_s required to estimate the gradient increases exponentially with respect to the number of bits n. The numerical simulation of Sec. 3.2.3 shows that this situation is also carried over to the Wasserstein distance when using the trace distance as ground cost.

One possible solution to avoid such a vanishing gradient problem is to build the cost function only from local measurements [35, 36]. The following cost function is proposed for the task of learning the state $|\phi_{\boldsymbol{\theta}}\rangle = U(\boldsymbol{\theta})|0\rangle$ with parameterized circuit $U(\boldsymbol{\theta})$.

Definition 8 (Cost for quantum state only with local measurements [19, 37])**.**

$$\begin{aligned} c_{\mathrm{local}}(|\psi\rangle, \phi_{\boldsymbol{\theta}}) &= \sqrt{\frac{1}{n}\sum_{j=1}^{n}(1 - p^{(k)})}, \\ p^{(k)} &= \mathrm{Tr}\left[P_0^k U^\dagger(\boldsymbol{\theta})|\psi\rangle\langle\psi| U(\boldsymbol{\theta})\right], \\ P_0^k &= \mathbb{I}_1 \otimes \mathbb{I}_2 \otimes \cdots \otimes \overbrace{|0\rangle\langle 0|_k}^{k\text{-th bit}} \otimes \cdots \otimes \mathbb{I}_n, \end{aligned} \tag{2.10}$$

where n is the number of qubits, and $\mathbb{I}_i$ and $|0\rangle\langle 0|_i$ denote the identity operator and the projection operator that act on i-th qubit, respectively. $p^{(k)}$ denote the probability of getting 0 when observing the k-th qubit. The trace distance of Eq. (2.9) satisfies the axiom of metric, but not the local cost Eq. (2.10) (not satisfies symmetricity and triangle inequality). However, the following proposition tells us that the Optimal Transport Loss $\mathcal{L}_{c_{\mathrm{local}}}$ with local cost c_{local} satisfies the properties of divergence.

Proposition 9. *When the ground cost $c(\boldsymbol{x}, \boldsymbol{y})$ satisfies*

$$\begin{aligned} &c(\boldsymbol{x}, \boldsymbol{y}) \geq 0, \\ &c(\boldsymbol{x}, \boldsymbol{y}) = 0 \textit{ iff } \boldsymbol{x} = \boldsymbol{y}, \end{aligned} \tag{2.11}$$

the Optimal Transport Loss $\mathcal{L}_c(\alpha,\beta)$ with $c(\boldsymbol{x}, \boldsymbol{y})$ as ground cost satisfies the following properties for any probability distributions α and β.

$$\begin{aligned} &\mathcal{L}_c(\alpha,\beta) \geq 0, \\ &\mathcal{L}_c(\alpha,\beta) = 0 \textit{ iff } \alpha = \beta. \end{aligned} \tag{2.12}$$

In the case of quantum generative model, random variables $\boldsymbol{x}$ and $\boldsymbol{y}$ correspond to state vectors. Since the Optimal Transport Loss with local ground cost Eq.(2.10) satisfies the properties of divergence, it would be suitable for use in comparing the probability distributions. Further, the numerical simulation of Sec. 2.3 shows that the Optimal Transport Loss with local ground cost may avoid the vanishing gradient problem. Therefore, we employ the Optimal Transport Loss with local ground cost throughout this paper. The next section shows the learning algorithm and performance evaluation of this loss function, and then Sec. 4 shows the demonstration.

3. Proposed Algorithm

Here, we first introduce the learning algorithm for the generative model which uses the Optimal Transport Loss with the

ⓒ 2022 Information Processing Society of Japan

ground cost Eq. (2.10) in Sec. 3.1. Then we analyze the performance of the loss function in Sec. 3.2 from both numerical simulations and analytical calculations.

Algorithm 1 Learning Algorithm with Quantum Optimal Transport Loss Eq.(3.2)

Input: Quantum circuit model $U(\boldsymbol{z},\boldsymbol{\theta})$ with initial parameters $\boldsymbol{\theta}$, learning rate ε, data samples $\{|\psi_i\rangle\}$

Output: A Quantum circuit which represent the distribution of input data

1: **repeat**
2: Generate latent variables $\{\boldsymbol{z}_j\}_{j=1}^{N_g}$ from the latent distribution.
3: Estimate the ground costs $\left\{\tilde{c}^{(N_s)}_{\mathrm{local},i,j}\right\}_{i,j=1}^{N_r,N_g}$ from $\{|\psi_i\rangle\}_{i=1}^{N_r}$ and $\{U(\boldsymbol{z}_j,\boldsymbol{\theta})\}_{j=1}^{N_g}$ with N_s shots as in Eq.(2.10).
4: Calculate the optimal transport plan $\pi_{i,j}$ by Eq.(3.2).
5: Calculate the gradients $\left\{\frac{\partial}{\partial\theta_k}\mathcal{L}_{c_{\mathrm{local}}}\right\}_{k=1}^{N_p}$ from $\pi_{i,j}$ and $\left\{\frac{\partial}{\partial\theta_k}\tilde{c}^{(N_s)}_{\mathrm{local},i,j}\right\}_{i,j,k=1}^{N_r,N_g,N_p}$ with parameter shift rule [38, 39].
6: Update $\{\theta_k\}_{k=1}^{N_p}$ by using the gradients $\left\{\frac{\partial W}{\partial\theta_k}\right\}_{k=1}^{N_p}$ with learning rate ε
7: **until** convergence

3.1 Learning Algorithm

The target of this paper is to learn a generative model that represents the underlying distribution from a quantum data set $\{|\psi_i\rangle\}_{i=1}^{N_r}$. Hereafter, we assume that we can prepare a finite number of copies for each quantum state, e.g., k copies $\{|\psi_i\rangle^{\otimes k}\}_{i=1}^{N_r}$. Note that the learning problem with infinite number of copies becomes classical machine learning problem, because in such a case we can determine all the components by performing full tomography [40].

We employ the implicit generative model described in Sec. 2.1. In the quantum case, one candidate for the implicit model will take the following form:

Definition 10 (Implicit generative model with quantum circuit)**.** *Using the initial state $|0\rangle$ and the parameterized quantum circuit $U(\boldsymbol{z},\boldsymbol{\theta})$, the implicit generative model with quantum circuit is defined as*

$$|\phi_{\boldsymbol{\theta}}(\boldsymbol{z})\rangle = U(\boldsymbol{z},\boldsymbol{\theta})\,|0\rangle\,, \tag{3.1}$$

where $\boldsymbol{\theta}$ are trainable parameters and $\boldsymbol{z}$ are latent variables that follow a known probability distribution.

The similar circuit model is also found in meta-VQE [41], which uses physical parameters instead of random latent variables $\boldsymbol{z}$. Note that the model putting the latent variables $\boldsymbol{z}$ into the initial state ($|\phi_{\boldsymbol{\theta}}(\boldsymbol{z})\rangle = U(\boldsymbol{\theta})\,|\boldsymbol{z}\rangle$) described in [16] does not match the current purpose, because in such a model, states with different latent variables are always orthogonal to each other, making it difficult to capture small changes in Hilbert space as small changes of the latent variables. In addition, it may be advantageous for the proposed implicit model that the analytical derivative can be calculated by the parameter shift rule [38, 39] not only for the parameters but also for the latent variables $\boldsymbol{z}$. In fact, anomaly detection in Sec. 4 exploits derivatives of the latent variables.

The empirical estimator of Optimal Transport Loss calculated from training data $\{|\psi_i\rangle\}_{i=1}^{N_r}$ and samples of the latent variables $\{\boldsymbol{z}_j\}_{j=1}^{N_g}$ becomes,

$$\mathcal{L}_{\tilde{c}^{(N_s)}_{\mathrm{local}}}\left(\{|\psi_i\rangle\}_{i=1}^{N_r},\{|\phi_{\boldsymbol{\theta}}(\boldsymbol{z}_j)\rangle\}_{j=1}^{N_g}\right) = \min_{\{\pi_{i,j}\}_{i,j=1}^{N_r,N_g}} \sum_{i,j=1}^{N_r,N_g} \tilde{c}^{(N_s)}_{\mathrm{local},i,j}\pi_{i,j},$$
$$\text{subject to} \quad \sum_{i=1}^{N_r}\pi_{i,j} = \frac{1}{N_g}, \sum_{j=1}^{N_g}\pi_{i,j} = \frac{1}{N_r}, \pi_{i,j} \geq 0. \tag{3.2}$$

Here, $\tilde{c}^{(N_s)}_{\mathrm{local},i,j}$ is the estimator of ground cost of Eq.(2.10) with N_s shots, i.e., by using the random variables $X^{(s)}_{i,j,k}$ ($s = 1, 2, \ldots, N_s$), which follows Bernoulli distribution with probability distribution $1 - p^{(k)}_{i,j} = 1 - \mathrm{Tr}\left[P^k_0 U^\dagger(\boldsymbol{z}_j,\boldsymbol{\theta})\,|\psi_i\rangle\langle\psi_i|\,U(\boldsymbol{z}_j,\boldsymbol{\theta})\right]$, $\tilde{c}^{(N_s)}_{\mathrm{local},i,j}$ is defined as

$$\tilde{c}^{(N_s)}_{\mathrm{local},i,j} = \sqrt{\frac{1}{n}\sum_{k=1}^{n}\frac{1}{N_s}\sum_{s=1}^{N_s}X^{(s)}_{i,j,k}} \tag{3.3}$$

For the minimization of the loss function of Eq. (3.2), the parameters $\boldsymbol{\theta} = \{\theta_k\}_{k=1}^{N_p}$ are updated by the following two steps:

(1) With N_r training data and N_g generated latent variables, estimate the ground costs $\left\{\tilde{c}^{(N_s)}_{\mathrm{local},i,j}\right\}_{i,j=1}^{N_r,N_g}$ by N_s shots. Then solve the linear programming of Eq.(3.2) to obtain the optimal couplings $\{\pi_{i,j}\}_{i,j=1}^{N_r,N_g}$.

(2) Calculate the gradient $\left\{\frac{\partial}{\partial\theta_k}\tilde{c}^{(N_s)}_{\mathrm{local},i,j}\right\}_{i,j,k=1}^{N_r,N_g,N_p}$ with N_s shots by using parameter shift rule [38, 39] (In practice, only calculate for about $O(\max(N_r,N_g))$ components which satisfy $\pi_{i,j} > 0$). With obtained gradients $\left\{\frac{\partial}{\partial\theta_k}\mathcal{L}_{c_{\mathrm{local}}}\right\}_{k=1}^{N_p}$ and optimal couplings $\pi_{i,j}$, update the parameters $\boldsymbol{\theta} = \{\theta_k\}_{k=1}^{N_p}$ by techniques such as stochastic gradient.

The pseudo-code of this parameter update is shown in the Algorithm 1. With this update method, the total number of training quantum states $|\psi_i\rangle$ required for each update is about $O(N_rN_gN_s)$ in step 1 and $O(\max(N_r,N_g)N_sN_p)$ in step 2.

3.2 Performance Analysis

Here, we analyze the performance of the proposed empirical loss of Eq. (3.2). First, we numerically show that the approximation error of the loss due to the finiteness of training data depends on the intrinsic dimension of data, and does not depend on the number of bits in Sec. 3.2.1. Then, in Sec. 3.2.2, we show the dependence of the approximation error with respect to the number of shots by analytical calculations and numerical simulations. Finally, in Sec.3.2.3, we numerically shows that the vanishing gradient problem may be avoided by using the proposed loss function.

In the following, we employ the structure of the parameterized unitary matrix $U(\boldsymbol{z},\boldsymbol{\theta})$ shown in Fig. 1, which is similar to [42] except for the latent variables $\boldsymbol{z}$, i.e, the structure with the following N_L repeated layers:

$$U_{N_L,\boldsymbol{\xi},\boldsymbol{\eta}}(\boldsymbol{z},\boldsymbol{\theta}) = \prod_{\ell=1}^{N_L} WV_{\boldsymbol{\xi}_\ell,\boldsymbol{\eta}_\ell}(\boldsymbol{z},\boldsymbol{\theta}_\ell), \tag{3.4}$$

where $\boldsymbol{\theta}_\ell = \{\theta_{\ell,j}\}_{j=1}^{n}, \boldsymbol{\xi}_\ell = \{\xi_{\ell,j}\}_{j=1}^{n}$ and $\boldsymbol{\eta}_\ell = \{\eta_{\ell,j}\}_{j=1}^{n}$ are n-dimensional parameters in the ℓ-th layer, and $\boldsymbol{\theta} = \{\boldsymbol{\theta}_\ell\}_{\ell=1}^{N_L}, \boldsymbol{\xi} = \{\boldsymbol{\xi}_\ell\}_{\ell=1}^{N_L}$ and $\boldsymbol{\eta} = \{\boldsymbol{\eta}_\ell\}_{\ell=1}^{N_L}$. $\boldsymbol{\theta}$ are trainable parameters and $\boldsymbol{z}$ are latent variables. W is an entangler of ladder controlled-Z gate which acts controlled-Z gates on all adjacent bits:

ⓒ 2022 Information Processing Society of Japan

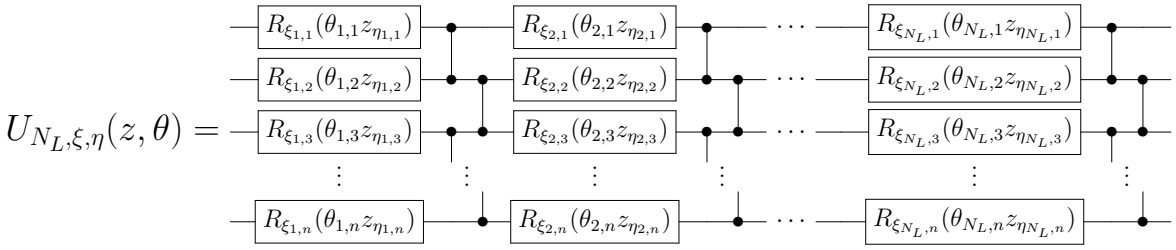


Fig. 1 The structure of parameterized quantum circuit (ansatz) used in the performance analysis of this section and the demonstration of Sec. 4. This ansatz consists of the repeated layers with a similar structure. In the ℓ-th layer, single qubit Pauli rotations with angles $\{\theta_{\ell,j} \times z_{\eta_{\ell,j}}\}_{j=1}^{n}$ and directions $\{\xi_{\ell,j}\}_j = 1^n$ are applied to each qubit followed by a ladder controlled-Z gate. The rotation angles $\boldsymbol{\xi} = \{\xi_{\ell,j}\}_{\ell,j=1}^{N_L,n}$ and the components of the latent variables $\boldsymbol{\eta} = \{\eta_{\ell,j}\}_{\ell,j=1}^{N_L,n}$ are randomly chosen at the beginning of the analysis and never changed during the analysis.

$$W = \prod_{i=1}^{n-1} CZ_{i,i+1}, \tag{3.5}$$

where $CZ_{i,i+1}$ is a controlled-Z gate acting on i-th and $(i+1)$-th bit. The operator $V_{\boldsymbol{\xi}_\ell,\boldsymbol{\eta}_\ell}(\boldsymbol{z},\boldsymbol{\theta}_\ell)$ consists of the random Pauli rotations applied to each qubits:

$$V_{\boldsymbol{\xi}_\ell,\boldsymbol{\eta}_\ell}(\boldsymbol{z},\boldsymbol{\theta}_\ell) = \prod_{i=1}^{n} R_{\xi_{\ell,i}}(\theta_{\ell,i} z_{\eta_{\ell,i}}), \tag{3.6}$$

where $R_{\xi_{\ell,i}}(\theta_{\ell,i} z_{\eta_{\ell,i}})$ is a Pauli ratation with angle $\theta_{\ell,i} z_{\eta_{\ell,i}}$ and direction $\xi_{\ell,i}$. The direction $\xi_{\ell,i} \in \{X, Y, Z\}$, and the component of the latent variables $\eta_{\ell,i} \in \{0, 1, 2, \ldots, N_z\}$ are randomly chosen at the beginning of learning and never changed during learning. Here, for convenience, we have added a cosntant bias term z_0 to the latent variable $\boldsymbol{z}$.

We use Qiskit [43] for analyzing quantum circuits throughout this paper.

3.2.1 Approximation error due to the finiteness of training data

As explained in Sec. 2.2, it is known for the p-Wasserstein distance that the convergence rate of the approximation error of the loss function depends mainly on the intrinsic dimension of data and the dimension N_z of the latent space $\mathcal{Z}$, and not on the dimension of the sample space $\mathcal{X}$. On the other hand, it is not clear for the proposed loss function of Eq.(3.2) whether similar error dependence on the dimension holds, because the ground cost of the proposed loss does not satisfy the axioms of metric and the loss does not meet the definition of p-Wasserstein distance. Here, we numerically confirm that this error dependence also holds for the proposed loss Eq.(3.2).

We present the following two types of numerical simulations; The first one (**Experiment A**) is about Eq.(2.7), which describes the behavior of the approximation error of the loss between two identical distributions, which corresponds to the situation near the end of learning. The second one (**Experiment B**) is about Eq.(2.8), which describes the behavior of the approximation error among two different distributions. In the case of p-Wasserstein distance, Eq. (2.7) and Eq. (2.8) can be easily derived from Eq.(2.6) due to the properties of metric, but in the case of proposed loss, this derivation does not hold. Hence, we here individually confirm each of them by numerical simulations.

In this subsection, we use the statevector simulator [43], i.e., the result with infinite number of shots is presented here. The influence from the finiteness of the number of shots is shown in Sec. 3.2.2.

3.2.1.1 Experiment A

In this experiment, we confirm that the similar dependence as Eq.(2.7) also holds for the proposed loss. Specifically, we numerically show the dependence of the following expected value on the number of training data M.

$$\mathbb{E}_{\tilde{\boldsymbol{z}},\boldsymbol{z}\sim U(0,1)^{N_z}}\left[J^{c_{\text{local}}}_{\boldsymbol{\xi},\boldsymbol{\eta};\boldsymbol{\xi},\boldsymbol{\eta}}(\tilde{\boldsymbol{z}},\boldsymbol{\theta} : \boldsymbol{z},\boldsymbol{\theta}; M)\right], \tag{3.7}$$

where, for convenience, the empirical loss is denoted as

$$J^{c_{\text{local}}}_{\tilde{\boldsymbol{\xi}},\tilde{\boldsymbol{\eta}};\boldsymbol{\xi},\boldsymbol{\eta}}(\tilde{\boldsymbol{z}},\tilde{\boldsymbol{\theta}} : \boldsymbol{z},\boldsymbol{\theta}; M) = \mathcal{L}_{c_{\text{local}}}(\{U_{N_L,\tilde{\boldsymbol{\xi}},\tilde{\boldsymbol{\eta}}}(\tilde{\boldsymbol{z}}_i,\tilde{\boldsymbol{\theta}})\,|0\rangle\}_{i=1}^{M}, \{U_{N_L,\boldsymbol{\xi},\boldsymbol{\eta}}(\boldsymbol{z}_j,\boldsymbol{\theta})\,|0\rangle\}_{j=1}^{M}), \tag{3.8}$$

where $\mathcal{L}_{c_{\text{local}}}$ is the empirical loss function defined in Eq.(3.2). In Eq.(3.7), we set common fixed parameters for the two unitary operators that appear in the argument of $\mathcal{L}_{c_{\text{local}}}$ in Eq.(3.8). This indicates that $J^{c_{\text{local}}}_{\boldsymbol{\xi},\boldsymbol{\eta};\boldsymbol{\xi},\boldsymbol{\eta}}(\tilde{\boldsymbol{z}},\boldsymbol{\theta} : \boldsymbol{z},\boldsymbol{\theta}; M)$ in Eq.(3.7) would vanish in the limit of infinite number of training data ($M \to \infty$). The expectation in the equation is taken with respect to latent variables $\tilde{\boldsymbol{z}}_i$, $\boldsymbol{z}_j$ with N_z-dimensional uniform distribution $U(0,1)^{N_z}$, but we numerically approximate it by N_{Monte} Monte Carlo samplings.

The typical result of the numerical simulation is shown in Fig. 2. In the figures, the points represent the numerical results and the dotted lines represent the scaling curve M^{-1/N_z}. Each figure shows the result of a different number of bits n, and each figure contains results of multiple latent dimensions N_z. In the range with a large number of training data, the points and dotted lines show almost the same trend regardless of the number qubits, which imply that the approximation error of the loss Eq.(3.7) is almost independent of the number of qubits n and depends mainly on the latent dimension N_z.

3.2.1.2 Experiment B

We turn to the second experiment to confirm that the approximation error of the proposed loss scales similar as Eq.(2.8), the case of the loss between different distributions. Specifically, we numerically show the dependence of the following expectation on

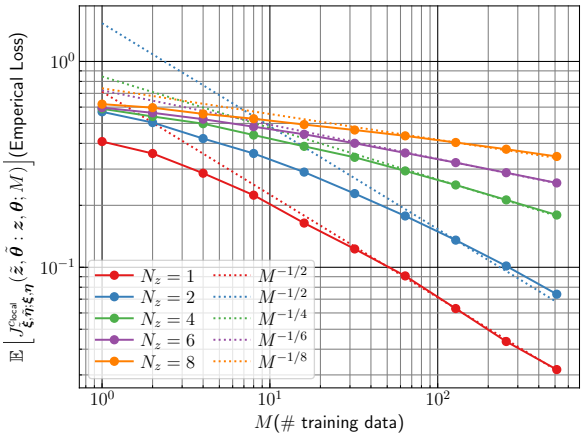


Fig. 2 Typical result of numerical simulations on the relationship between the number of training data and approximation error of the empirical loss Eq.(3.7). Each curve indicates the results of various latent dimensions N_z with the number of qubits $n = 10$. For reference, the scaling curves M^{-1/N_z} are added as dotted lines, and the color of each line are the same as that of the corresponding points. These graph show that the approximation error of the loss Eq.(3.7) mainly scale as M^{-1/N_z}, and almost independent of the number of qubits n.

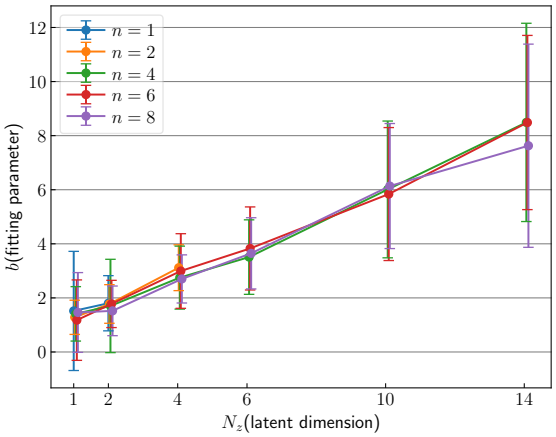


Fig. 3 The simulation results on the dependence of the fitting parameter b on (a) the number of bits n and (b) the latent dimension N_z. The fitting parameter b is obtained by fitting the empirical loss Eq.(3.9) by Eq.(3.10). The subfigure (a) shows that the fitting parameter b is almost independent of the number of bits n, and the subgraph (b) shows that b is linearly dependent on the latent dimension N_z.

the number of training data M.

$$\mathbb{E}_{\tilde{\boldsymbol{z}},\boldsymbol{z}\sim U(0,1)^{N_z}}\left[\lim_{K\to\infty} J^{c_{\rm local}}_{\tilde{\boldsymbol{\xi}},\tilde{\boldsymbol{\eta}};\boldsymbol{\xi},\boldsymbol{\eta}}(\tilde{\boldsymbol{z}},\tilde{\boldsymbol{\theta}}:\boldsymbol{z},\boldsymbol{\theta};K) - J^{c_{\rm local}}_{\tilde{\boldsymbol{\xi}},\tilde{\boldsymbol{\eta}};\boldsymbol{\xi},\boldsymbol{\eta}}(\tilde{\boldsymbol{z}},\tilde{\boldsymbol{\theta}}:\boldsymbol{z},\boldsymbol{\theta};M)\right], \tag{3.9}$$

where $J^{c_{\rm local}}$ is defined in Eq.(3.8). In this case, we set different fixed parameters for the two unitary operators in Eq.(3.8).

The first term of Eq.(3.9) is difficult to calculate numerically, because it contains an limit on the number of training data. Since the first term is independent of the number of training data, the first term is expected to take the following form from Eq.(2.8):

$$\mathbb{E}_{\tilde{\boldsymbol{z}},\boldsymbol{z}\sim U(0,1)^{N_z}}\left[J^{c_{\rm local}}_{\tilde{\boldsymbol{\xi}},\tilde{\boldsymbol{\eta}};\boldsymbol{\xi},\boldsymbol{\eta}}(\tilde{\boldsymbol{z}},\tilde{\boldsymbol{\theta}}:\boldsymbol{z},\boldsymbol{\theta};M)\right] = aM^{-1/b} + c, \tag{3.10}$$

Therefore, in the following, we fit the Monte Carlo results of the first term with Eq.(3.10), and show the dependence of the fitting parameter b on the latent dimension N_z.

The results of the numerical simulation is depicted in Fig.3, which shows the dependence of the fitting parameter b on the number of bits n and latent dimension N_z. Fig.3 indicates that the fitting parameter b is almost independent of the number of bits n, and is almost linearly dependent on the latent space N_z.

The above Experiments A and B suggest the following observation:

Observation 11. *The scaling of the approximation error of the loss Eq.*(3.2) *with respect to the number of training data M is independent of the number of bits, but determined by the latent dimension N_z:*

$$\mathcal{L}_{c_{\rm local}}\left(\{|\psi\rangle_i\}_{i=1}^M, \{U(\boldsymbol{z}_j,\boldsymbol{\theta})|0\rangle\}_{j=1}^M\right) \lesssim O(M^{-1/N_z}). \tag{3.11}$$

Proving this observation would be very challenging and is the subject of future work. This observation shows that the proposed loss is efficient when the intrinsic dimension of the data and the latent dimension are sufficiently low, at least for the approximate error due to the finiteness of training data.

3.2.2 Approximation error due to finite number of shots

The error analysis in Sec.3.2.1 assumes that the nunber of shot is infinite, i.e., the ground cost between the quantum states can be determined perfectly. Here, we analyze the effect of the finiteness of the number of shots on the approximation error.

The following proposition shows the upper bound of the difference between the loss from infinite and finite number of shots.

Proposition 12. *Let $\tilde{c}^{(N_s)}_{\rm local}$ be an estimator of the ground cost $c_{\rm local}$ of Eq.*(2.10) *using N_s samples. Suppose the support of two different probability distributions are sufficiently separated, i.e., there exists a lower bound $g > 0$ to the ground cost for any $i, j \in \{1, 2, \ldots, M\}$ $((c_{\rm local}(|\psi_i\rangle, U(\boldsymbol{z}_j,\boldsymbol{\theta})|0\rangle) > g,\ \forall i, j))$. Then, with any real constant δ, the following inequality holds*

$$P\Big(|\mathcal{L}_{c_{\rm local}}\left(\{|\psi\rangle_i\}_{i=1}^M, \{U(\boldsymbol{z}_j,\boldsymbol{\theta})|0\rangle\}_{j=1}^M\right) - \mathcal{L}_{\tilde{c}^{(N_s)}_{\rm local}}(\{|\psi\rangle_i\}_{i=1}^M, \{U(\boldsymbol{z}_j,\boldsymbol{\theta})|0\rangle\}_{j=1}^M)| \geq \sqrt{\frac{2M}{\delta}}\sqrt{\frac{1-g}{N_s} + \frac{(1-g)^2}{4N_s^2 g}} + \frac{1-g}{2N_s\sqrt{g}}\Big) \leq \delta. \tag{3.12}$$

Proposition 12 suggests that the approximation error is bounded above by $O(\sqrt{M/N_s})$ under the condition $M \gg 1$ and $N_s \gg 1$. Combining this with Observation 11, it is implied that the approximation error due to the finite number of shots N_s and training data M is bounded above by about $O(M^{-1/N_z}) + O\left(\sqrt{M/N_s}\right)$, where N_z is a latent dimension.

Then, we consider the average approximation error due to the number of shots by numerical simulations. Again, we employ the hardware efficient ansatz shown in Fig. 1 of Sec. 3.2.1. The purpose of the numerical simulation is to check the dependence of the following expectation value on the number of shots N_s and training data M.

$$\mathbb{E}_{\tilde{\boldsymbol{z}},\boldsymbol{z}\sim U(0,1)^{N_z}}\left[\left|J^{\tilde{c}^{(N_s)}_{\rm local}}_{\tilde{\boldsymbol{\xi}},\tilde{\boldsymbol{\eta}};\boldsymbol{\xi},\boldsymbol{\eta}}(\tilde{\boldsymbol{z}},\tilde{\boldsymbol{\theta}}:\boldsymbol{z},\boldsymbol{\theta};M) - J^{c_{\rm local}}_{\tilde{\boldsymbol{\xi}},\tilde{\boldsymbol{\eta}};\boldsymbol{\xi},\boldsymbol{\eta}}(\tilde{\boldsymbol{z}},\tilde{\boldsymbol{\theta}}:\boldsymbol{z},\boldsymbol{\theta};M)\right|\right], \tag{3.13}$$

where $J^{\tilde{c}^{(N_s)}_{\rm local}}$ denotes the proposed loss defined in Eq.(3.8). As in

Section 3.2.1, we approximate the expectation with respect to the latent variables $\boldsymbol{z}, \tilde{\boldsymbol{z}}$ by Monte Carlo calculations. Also, we randomly choose fixed parameters $\boldsymbol{\xi}, \boldsymbol{\eta}, \boldsymbol{\theta}, \tilde{\boldsymbol{\xi}}, \tilde{\boldsymbol{\eta}}, \tilde{\boldsymbol{\theta}}$ prior to the simulation.

Simulation results are depicted in Fig. 4, which shows the following phenomena.

- In the range of small training data M, the approximation error is roughly proportional to $M^{-1/2}$.
- In the range of big training data M, the approximation error takes $\sqrt{c_1 \ln M + c_2}$ with constants c_1 and c_2.
- The dependence on the number of shots N_s is roughly proportional to $N_s^{-1/2}$.

Of these, the dependence on the number of shots N_s would be due to the central limit theorem. We also roughly explain the dependence on the number of training data in Appendix A.1.

The analysis here indicates that it is necessary to properly balance the number of shots and the samples to reduce the approximation error.

3.2.3 Qubit and training data number dependence of gradient

In this section, we numerically confirm that the proposed algorithm avoids the vanishing gradient problem. As mentioned in Sec.2.3, the cost function built from the global measurements suffer from the curse of dimendionality, that is known as barren plateaus(BPs) phenomenon for variational quantum algorithms, i.e., the gradients of the cost function vanish exponentially with the number of qubits n. To avoid it, we built the cost function from local measurements in the proposed loss. Here, we see the qubit number dependence of the variance in the partial derivative of the cost function, which is common way to characterize the BPs. We calculated the expectation of the variance of the proposed loss, Eq.(2.10), based on the training data $\{|\psi_i\rangle\}_{i=1}^{N_r}$ and the sampled data from the generative model $\{U(z_j,\theta)|0\rangle\}_{j=1}^{N_r}$. The structure of ansatz of the generative model $\{U(z_j,\theta)|0\rangle\}_{j=1}^{N_r}$ was same as Fig. 1, and we numerically evaluated the following value:

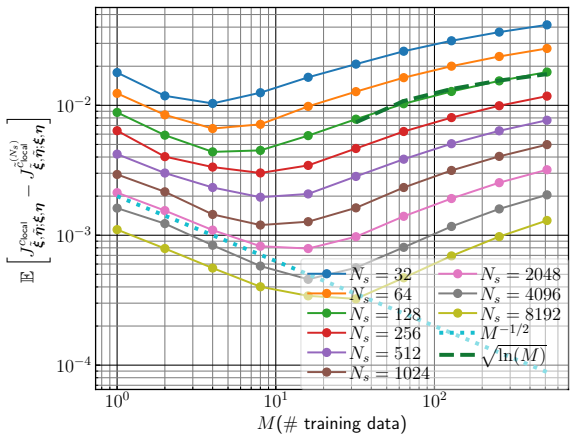


Fig. 4 Typical simulation result of the approximation error of the proposed loss due to the number of shots defined in Eq.(3.13). The dependence on the number of training data M with various latent dimensions are shown. For reference, we add the curve of $M^{-1/2}$ as dotted line. The fitting result of the points $N_s = 128$ with the curve is added as dashed line.

$$\mathbb{V}_{\boldsymbol{\xi},\boldsymbol{\eta},\boldsymbol{\theta},\boldsymbol{z}}\left[\frac{\partial}{\partial\theta}\mathcal{L}_{c_{\text{local}}}\left(\{|\psi\rangle_i\}_{i=1}^{M}, \{U_{N_L,\boldsymbol{\xi},\boldsymbol{\eta}}(\boldsymbol{z}_j,\boldsymbol{\theta})|0\rangle\}_{j=1}^{M}\right)\right], \quad (3.14)$$

where $\boldsymbol{z}$ is sampled from the uniform distribution $U(0,1)$ and parameters $\boldsymbol{\xi}, \boldsymbol{\eta}, \boldsymbol{\theta}$ are randomly chosen. More concretely, the optimal transport loss is calculated by the ground cost with statevector simulator. The derivative of the optimal transport loss is calculated based on the derivative of the ground cost, which is obtained by parameter shift rule [38]. The expectation of the variance is approximated by Monte Carlo calculations with randomly chosen $\boldsymbol{z}$,$\boldsymbol{\xi}$, $\boldsymbol{\eta}$ and $\boldsymbol{\theta}$. The derivatives is taken with respect to $\theta_{1,1}$. We show the numerical simulation result based on i) Global measurements represented by Eq.(2.9) and ii) Local measurements represented by Eq.(2.10).

The fixed training data sets $\{|\psi_i\rangle\}_{i=1}^{N_r}$ are prepared as follows;

$$|\psi\rangle_i = W' V_2'(\boldsymbol{\zeta}_2^i) W' V_1'(\boldsymbol{\zeta}_1^i)|0\rangle^{\otimes n}$$

where $\boldsymbol{\zeta}_1^i = \{\zeta_{1,j}^i\}_{j=1}^n$ and $\boldsymbol{\zeta}_2^i = \{\zeta_{2,j}^i\}_{j=1}^n$ are n-dimensional parameters. These parameters $\zeta_{\ell,j}^i \in [0, 2\pi]$ are randomly chosen from the uniform distribution and fixed during Monte Carlo calculation. The operators W', V_1' and V_2' are defined as follows:

$$W' = \prod_{i=1}^{n-1} CX_{j,j+1},\ V_1'(\zeta_1^i) = \prod_{j=1}^{n} R_{j,Y}(\zeta_{1,j}^i),\ V_2'(\boldsymbol{\zeta}_2^i) = \prod_{j=1}^{n} R_{j,Z}(\zeta_{2,j}^i), \quad (3.15)$$

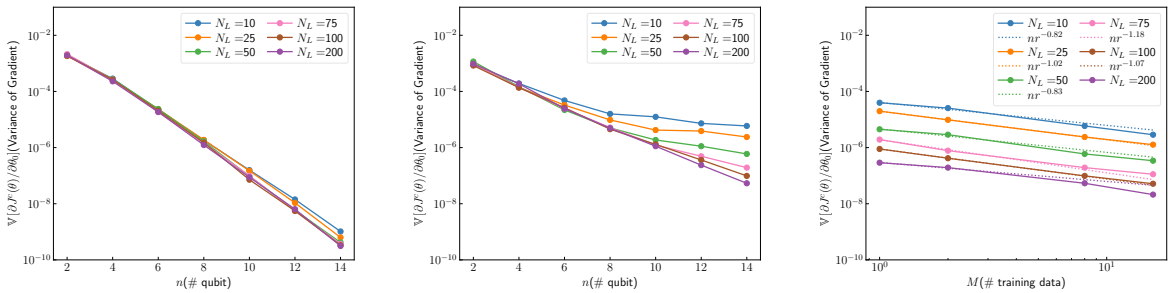
where $CX_{j,k}$ denote a controlled X gate, which act X gate on k-th bit with j-th bit as the control bit. $R_{j,Y}$ and $R_{j,Z}$ denote single qubit Pauli rotations around x and y axes, respectively.

The results of the numerical simulation in the case of $M = 8$ are shown in Fig. 5(a) and (b). The clear exponential decays in variance of gradient are observed for Global cost regardless of N_L. In contrast, for Local cost shallow circuits exhibit approximately constant scaling for $n \geq 10$, and deep circuits also exhibit slower scaling than Global cost and keep larger variance even in $n \geq 8$. These observations coincide with previous report [36] that analyses the commonly used cost function. It is reasonable since the cost function used in [36] is equivalent to single ground cost of our framework. The result implies that the ground cost built from local measurement avoids gradient vanishing even for quantum optimal transport loss.

In addition to the qubit number dependence, the training data number dependence may also be critical for proposed algorithm. The simulation results of the training data number dependence in the case of $n = 14$ are shown in Fig. 5(c). In the figure, the points represent the numerical results and the dotted lines represent the scaling curve M^{-x}. Each curves are well fitted by M^{-x} where x is around 1. It implies that the training number dependence almost obey the simple statistical scaling. Therefore, the proposed algorithm would be scaled though the training sample number should be appropriate for efficient learning.

4. Demonstration

In this section, we present anomaly detection based on the cost function defined in Eq.(2.3) as a proof-of-concept of the proposed loss function. Anomaly detection is a task that one judges whether test data $\boldsymbol{x}^{(t)}$ is anomalous(rare) data or not based on the



(a) Global cost (b) Local cost (c) Training number dependence(Local cost)

Fig. 5 The simulation results on the variance in the gradient of the cost function. The scaling in the qubit number n of (a) Global cost defined in Eq.(2.9) and (b) Local cost defined in Eq.(2.10), and (c) in the training data number M of Local cost. Each curve in the figures represents different number of layers N_L. The clear exponential decay is observed in (a), but is avoided in (b). The polynomial decay ($\simeq M^{-1}$) is observed in (c), and it implies simple statistical scaling.

knowledge learned from past training data $\boldsymbol{x}_i, (i = 1, 2, \ldots, M)$. The feature of this task compared to typical classification is the large bias between the number of normal and anomalous(rare) data, i.e. almost all past data is normal. Therefore, typical classification schemes are not suitable to solve this task, and other schemes have been widely studied [44].

In the field of quantum technology, such as quantum computation, quantum communication and quantum metrology, it is required to control quantum states accurately. However, quantum states are quite fragile and easily disturbed even by trivial environmental fluctuation. To detect such noise quickly and remove damaged state is key for the practical application of quantum technology. Previous anomaly detection schemes rely on the classical data obtained by quantum state measurement [45, 46]. To naively identify a quantum state, which is known as quantum tomography, requires exponentially large measurement in the number of qubit. In contrast, we propose an anomaly detection scheme with direct processing of quantum states and the measurement number is much reduced. More concretely, we use the generative model that learns the quantum data source from input quantum dataset. In other words, we can generate arbitrary quantum circuit with a fixed depth circuit in principle, although off course the precision heavily depends on the expressibility of the circuit. A that imitates a deep circuit

In general, anomaly detection is performed in the following three steps [47].

(1) (*Distribution estimation*): Construct a probability distribution model of normal data based on past data, a large majority of which is normal.

(2) (*Anomaly score design*): Define an Anomaly Score (AS) based on the probability distribution model of normal data.

(3) (*Threshold determination*): Set a threshold of AS for judging whether anomaly or not.

Of these steps, the probability distribution model in Step 1 is estimated by the learning algorithm presented in Sec. 3.1. Anomaly score design in Step 2 is processed by the similar way as AnoGAN [48] in classical machine learning. Namely, we define a loss function $\mathcal{L}(G(\boldsymbol{z}, \boldsymbol{\theta}) |0\rangle, |\psi^{(t)}\rangle)$ calculated by the test data $|\psi^{(t)}\rangle$ and the sample $U(\boldsymbol{z}, \bar{\boldsymbol{\theta}})$ generated by the learned probability distribution model, and take the minimum value in latent variable $\boldsymbol{z}$ as AS:

$$(\text{Anomaly Score}) = \min_{\boldsymbol{z}} \mathcal{L}(U(\boldsymbol{z}, \bar{\boldsymbol{\theta}}) |0\rangle, |\psi^{(t)}\rangle). \tag{4.1}$$

As the loss function $\mathcal{L}$, we use the local ground cost $c_{\text{local}}(|\psi^{(t)}\rangle, U(\boldsymbol{z}, \bar{\boldsymbol{\theta}}))$ defined in Eq.(2.10) below, for simplicity. The derivative in $\boldsymbol{z}$ is calculated for minimizing the loss function with respect to $\boldsymbol{z}$, by employing the parameter shift rule similarly as the derivative in θ.

Algorithm 2 Algorithm for Anomaly Detection

Input: A trained quantum circuit $U(\boldsymbol{z}, \bar{\boldsymbol{\theta}})$, test data $\{|\psi^{(t)}\rangle\}$
Output: Anomaly Score
1: Initialize $\boldsymbol{z}$
2: **repeat**
3: Calculate the ground cost $\mathcal{L}$ from $\{U(\boldsymbol{z}, \bar{\boldsymbol{\theta}})\}$ and $|\psi^{(t)}\rangle$ as in Eq.(2.10)
4: Calculate the gradients $\left\{\frac{\partial L}{\partial z_k}\right\}_{k=1}^{N_z}$ with parameter shift rule [38, 39].
5: Update $\boldsymbol{z}$ by using the gradients $\left\{\frac{\partial L}{\partial z_k}\right\}_{k=1}^{N_z}$
6: **until** convergence

The numerical experiments of Algorithm 2 are shown in Fig. 6. The concept of the model used here is same as previous sections, but we introduce so-called alternated layered ansataz (ALT) into ansatz, that is more favorable than hardware efficient ansatz (HEA) (shown in Fig. 1) from the aspect of the gradient vanishing phenomenon [35]. Fig. 6 depicts the Bloch sphere spanned by $|0\rangle^{\otimes 10}$ and $|1\rangle^{\otimes 10}$, and θ (ϕ) indicate the angle from z (x)-axis in the Bloch sphere. Training data $\{|\psi_j\rangle\}_{j=1}^{N_r}$, that corresponds to the normal data, is depicted in (a). It is represented as follows;

$$|\psi_j\rangle = \cos(\frac{\pi}{2}\Delta\theta_j) |0\rangle + e^{2\pi i \Delta\phi_j} \sin(\frac{\pi}{2}\Delta\theta_j) |2^n - 1\rangle, \tag{4.2}$$

where n is the number of qubit, $\Delta\theta_j$ and $\Delta\phi_j$ are the deviation of θ and ϕ. $\Delta\theta_j$ and $\Delta\phi_j$ are sampled from the normal distribution $N(\mu, \sigma)$ and the uniform distribution $U(a, b)$, where μ and σ is the average and the variance, respectively. We selected $(\mu, \sigma, a, b) = (0, 0.02, 0, 0.2)$ for $n = 10$. This corresponds to the two dimensional distribution, hence we set the dimension of latent variable as $N_z = 2$ for our model. After the training phase, we input various test data $\{|\psi^{(t)}\rangle\}$, shown in (b), that is represented

ⓒ 2022 Information Processing Society of Japan

as follows;

$$|\psi^{(t)}\rangle = \cos(\frac{\pi}{2}\theta^{(t)})\,|0\rangle + e^{2\pi i\phi^{(t)}}\sin(\frac{\pi}{2}\theta^{(t)})\,|2^n-1\rangle\,, \qquad (4.3)$$

where $\theta^{(t)} \in \{-0.5, -0.4, \cdots, 1.5\}$, and $\phi^{(t)} \in \{0, \pm 0.1, \pm 0.2, \cdots, \pm 1\}$.

The resultant anomaly score (AS) for each test data is shown in (c). We see intuitively reasonable results, i.e. AS clearly depends on the distance between the training data and the test data. Therefore, we can perform anomaly detection as follows; if we set the threshold as AS = 0.4, the data corresponding to $0.35 \le \theta^{(t)}/\pi \le 0.7$ and $-0.15 \le \phi^{(t)}/\pi \le 0.35$ in Fig. 6 is normal, and others are anomaly. Note that this dataset cannot be learned by conventional anomaly detection scheme [45, 46] since the ensemble average of the data becomes single mixed state, not the set of pure states, and it is necessary to directly process the quantum state for appropriate learning, not via classical data.

In addition to that, the number of shot for this experiment is also notable. In this section, we use QASM simulator for the numerical simulation. The shot number is $N_s = 1000$ for each measurement in training phase, and $N_s = 50$ for anomaly detection task, even the case of $n = 10$. The dimensionality of $n = 10$ system's Hilbert space is 1024. It indicates that small number of shot is enough to perform anomaly detection, compared to the dimensionality of the Hilbert space, once we have model learn normal state appropriately. This is a strong advantage for practical situation.

Finally, we compare the learning curve with different settings. Typical learning curves are shown in Fig. 7. The blue plots, which are labeled "local", represents the case that "the cost function is calculated based on the local observable (Eq. 2.10), and the ansatz is ALT (L-ALT)". On the other hands, the orange plots, which are labeled "global", represents the case that "the cost function is calculated based on the global observable (Eq. 2.9), and the ansatz is HEA (G-HEA)". Note that both plotted costs are calculated based on global cost. Concretely, a plot of "local" indicates the cost calculated based on global cost with the parameter at each iteration, that is trained with the cost calculated based on local cost, hence we can directly compare them. We observe that L-ALT has clear advantage over G-HEA in terms of fast convergence. This result coincides with that of Sec. 3.2.3, indicating the advantage of the local cost. In addition to the convergence speed, the final cost of L-ALT is lower than that of G-HEA. However, whether training is success or not heavily depends on the random seed, which determines the arrangement of rotation gates in ansatz, and we could not observe the successful training with all cases, even with local cost. Investigating efficient way to train the model is crucial, and we leave it as future work.

5. Conclusion

In classical machine learning, many generative models are vigorously studied, but there are few studies on quantum generative models for quantum data. This paper would be one of the first step for building such a quantum generative model. In this paper, we proposed a loss function for such a generative model by employing the optimal transport, which have the properties of statistical divergence. Also, we numerically and analytically investigated the properties of the proposed loss, and confirm that the approximation error of the proposed loss is almost independent of the number of qubit, the error can be reduced by increasing the number of shots and training data, and the vanishing gradient problem can be avoided. In addition, we performed the demonstration of anomaly detection as a proof-of-concept.

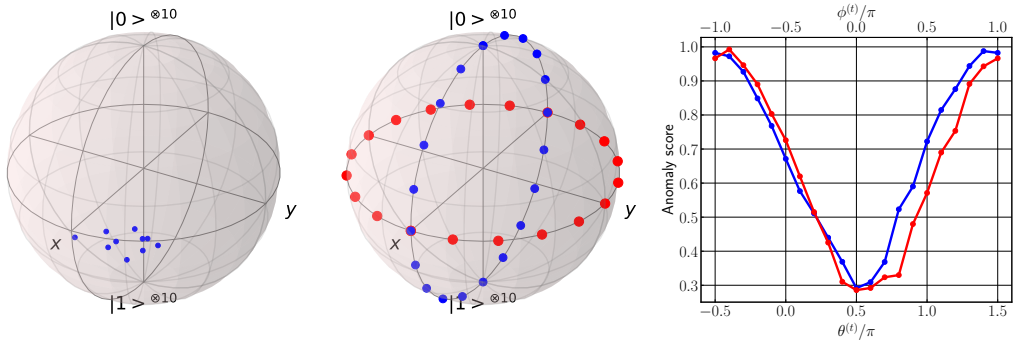
Acknowledgement

This work was supported by MEXT Quantum Leap Flagship Program Grant Number JPMXS0118067285 and JPMXS0120319794

References

[1] Aharonov, D., Cotler, J. and Qi, X.-L.: Quantum algorithmic measurement, *Nature Communications*, Vol. 13, No. 1, pp. 1–9 (2022).
[2] Wu, Y., Wu, B., Wang, J. and Yuan, X.: Provable Advantage in Quantum Phase Learning via Quantum Kernel Alphatron, *arXiv preprint arXiv:2111.07553* (2021).
[3] Huang, H.-Y., Broughton, M., Cotler, J., Chen, S., Li, J., Mohseni, M., Neven, H., Babbush, R., Kueng, R., Preskill, J. et al.: Quantum advantage in learning from experiments, *arXiv preprint arXiv:2112.00778* (2021).
[4] Bao, J., Chen, D., Wen, F., Li, H. and Hua, G.: CVAE-GAN: fine-grained image generation through asymmetric training, *Proceedings of the IEEE international conference on computer vision*, pp. 2745–2754 (2017).
[5] Brock, A., Donahue, J. and Simonyan, K.: Large scale GAN training for high fidelity natural image synthesis. CoRR abs/1809.11096 (2018), *arXiv preprint arXiv:1809.11096* (1809).
[6] Kulkarni, T. D., Whitney, W. F., Kohli, P. and Tenenbaum, J.: Deep convolutional inverse graphics network, *Advances in neural information processing systems*, Vol. 28 (2015).
[7] Gómez-Bombarelli, R., Wei, J. N., Duvenaud, D., Hernández-Lobato, J. M., Sánchez-Lengeling, B., Sheberla, D., Aguilera-Iparraguirre, J., Hirzel, T. D., Adams, R. P. and Aspuru-Guzik, A.: Automatic chemical design using a data-driven continuous representation of molecules, *ACS central science*, Vol. 4, No. 2, pp. 268–276 (2018).
[8] Zhou, C. and Paffenroth, R. C.: Anomaly detection with robust deep autoencoders, *Proceedings of the 23rd ACM SIGKDD international conference on knowledge discovery and data mining*, pp. 665–674 (2017).
[9] Benedetti, M., Garcia-Pintos, D., Perdomo, O., Leyton-Ortega, V., Nam, Y. and Perdomo-Ortiz, A.: A generative modeling approach for benchmarking and training shallow quantum circuits, *npj Quantum Information*, Vol. 5, No. 1, pp. 1–9 (2019).
[10] Coyle, B., Mills, D., Danos, V. and Kashefi, E.: The Born supremacy: quantum advantage and training of an Ising Born machine, *npj Quantum Information*, Vol. 6, No. 1, pp. 1–11 (2020).
[11] Zoufal, C., Lucchi, A. and Woerner, S.: Quantum generative adversarial networks for learning and loading random distributions, *npj Quantum Information*, Vol. 5, No. 1, pp. 1–9 (2019).
[12] Huang, H.-L., Du, Y., Gong, M., Zhao, Y., Wu, Y., Wang, C., Li, S., Liang, F., Lin, J., Xu, Y. et al.: Experimental quantum generative adversarial networks for image generation, *Physical Review Applied*, Vol. 16, No. 2, p. 024051 (2021).
[13] Romero, J., Olson, J. P. and Aspuru-Guzik, A.: Quantum autoencoders for efficient compression of quantum data, *Quantum Science and Technology*, Vol. 2, No. 4, p. 045001 (2017).
[14] Wan, K. H., Dahlsten, O., Kristjánsson, H., Gardner, R. and Kim, M.: Quantum generalisation of feedforward neural networks, *npj Quantum information*, Vol. 3, No. 1, pp. 1–8 (2017).
[15] Lloyd, S. and Weedbrook, C.: Quantum generative adversarial learning, *Physical review letters*, Vol. 121, No. 4, p. 040502 (2018).
[16] Dallaire-Demers, P.-L. and Killoran, N.: Quantum generative adversarial networks, *Physical Review A*, Vol. 98, No. 1, p. 012324 (2018).
[17] Kiani, B. T., De Palma, G., Marvian, M., Liu, Z.-W. and Lloyd, S.: Learning quantum data with the quantum earth mover's distance, *Quantum Science and Technology*, Vol. 7, No. 4, p. 045002 (2022).
[18] Chakrabarti, S., Yiming, H., Li, T., Feizi, S. and Wu, X.: Quantum wasserstein generative adversarial networks, *Advances in Neural Information Processing Systems*, Vol. 32 (2019).
[19] Khatri, S., LaRose, R., Poremba, A., Cincio, L., Sornborger, A. T. and Coles, P. J.: Quantum-assisted quantum compiling, *Quantum*, Vol. 3, p. 140 (2019).
[20] Peyre, G. and Cuturi, M.: *Computational Optimal Transport: With*

ⓒ 2022 Information Processing Society of Japan



(a) Training data (b) Test data (c) Anomaly score

Fig. 6 The simulation results on the anomaly detection at $n = 10$. (a) Training data. (b) The overview of test data. (c) Calculated anomaly scores for different test data. Blue and red line correspond to test data of the same color in (b). Lower axis and upper axis correspond to blue and red line, respectively.

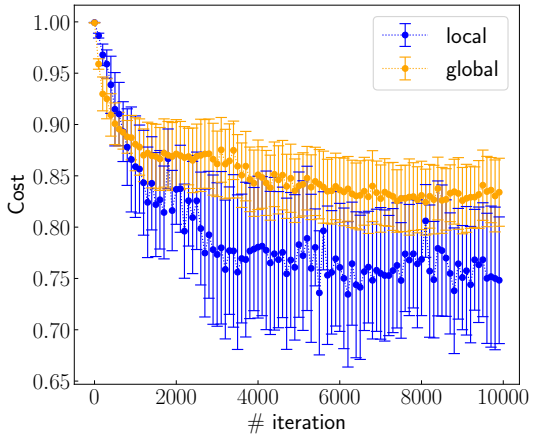


Fig. 7 Typical learning curves with different ground cost; $n = 10$. The plotted "Cost" is calculated with the parameter at each iteration, that is optimized with "local observable" (Eq. 2.10) or "global observable" (Eq. 2.9). Note that both costs are calculated based on the trace distance (Eq. 2.9) to directly compare each other. The error bar indicates the range of 1 sigma.

Applications to Data Science (Foundations and Trends in Machine Learning), Now Publishers, paperback edition (2019).
[21] Ollivier, Y., Hervé, P. and Villani, C.(eds.): *Optimal Transport: Theory and Applications*, London Mathematical Society Lecture Note Series, Cambridge University Press (2014).
[22] Santambrogio, F.: Optimal transport for applied mathematicians, *Birkäuser, NY*, Vol. 55, No. 58-63, p. 94 (2015).
[23] Peyre, G. and Cuturi, M.: Editorial IMA IAI - Information and Inference special issue on optimal transport in data sciences, *Information and Inference: A Journal of the IMA*, Vol. 8, No. 4, pp. 655–656 (online), DOI: 10.1093/imaiai/iaz032 (2019).
[24] Montavon, G., Müller, K.-R. and Cuturi, M.: Wasserstein training of restricted Boltzmann machines, *Advances in Neural Information Processing Systems*, Vol. 29 (2016).
[25] Bernton, E., Jacob, P. E., Gerber, M. and Robert, C. P.: Inference in generative models using the Wasserstein distance, *arXiv preprint arXiv:1701.05146*, Vol. 1, No. 8, p. 9 (2017).
[26] Arjovsky, M., Chintala, S. and Bottou, L.: Wasserstein generative adversarial networks, *International conference on machine learning*, PMLR, pp. 214–223 (2017).
[27] Tolstikhin, I., Bousquet, O., Gelly, S. and Schoelkopf, B.: Wasserstein auto-encoders, *arXiv preprint arXiv:1711.01558* (2017).
[28] Kantorovich, L. V.: On the translocation of masses, *Dokl. Akad. Nauk. USSR (NS)*, Vol. 37, pp. 199–201 (1942).
[29] Villani, C.: *Optimal transport: old and new*, Vol. 338, Springer (2009).
[30] Dudley, R. M.: The speed of mean Glivenko-Cantelli convergence, *The Annals of Mathematical Statistics*, Vol. 40, No. 1, pp. 40–50 (1969).
[31] Weed, J. and Bach, F.: Sharp asymptotic and finite-sample rates of convergence of empirical measures in Wasserstein distance, *Bernoulli*, Vol. 25, No. 4A, pp. 2620–2648 (2019).
[32] Nielsen, M. A. and Chuang, I. L.: *Quantum Computation and Quantum Information (Cambridge Series on Information and the Natural Sciences)*, Cambridge University Press, paperback edition (2000).
[33] Buhrman, H., Cleve, R., Watrous, J. and De Wolf, R.: Quantum fingerprinting, *Physical Review Letters*, Vol. 87, No. 16, p. 167902 (2001).
[34] Havlíček, V., Córcoles, A. D., Temme, K., Harrow, A. W., Kandala, A., Chow, J. M. and Gambetta, J. M.: Supervised learning with quantum-enhanced feature spaces, *Nature*, Vol. 567, No. 7747, pp. 209–212 (2019).
[35] Nakaji, K. and Yamamoto, N.: Expressibility of the alternating layered ansatz for quantum computation, *Quantum*, Vol. 5, p. 434 (2021).
[36] Cerezo, M., Sone, A., Volkoff, T., Cincio, L. and Coles, P. J.: Cost function dependent barren plateaus in shallow parametrized quantum circuits, *Nature communications*, Vol. 12, No. 1, pp. 1–12 (2021).
[37] Sharma, K., Khatri, S., Cerezo, M. and Coles, P. J.: Noise resilience of variational quantum compiling, *New Journal of Physics*, Vol. 22, No. 4, p. 043006 (2020).
[38] Mitarai, K., Negoro, M., Kitagawa, M. and Fujii, K.: Quantum circuit learning, *Physical Review A*, Vol. 98, No. 3, p. 032309 (2018).
[39] Schuld, M., Bergholm, V., Gogolin, C., Izaac, J. and Killoran, N.: Evaluating analytic gradients on quantum hardware, *Physical Review A*, Vol. 99, No. 3, p. 032331 (2019).
[40] Aïmeur, E., Brassard, G. and Gambs, S.: Machine learning in a quantum world, *Conference of the Canadian Society for Computational Studies of Intelligence*, Springer, pp. 431–442 (2006).
[41] Cervera-Lierta, A., Kottmann, J. S. and Aspuru-Guzik, A.: Meta-variational quantum eigensolver: Learning energy profiles of parameterized hamiltonians for quantum simulation, *PRX Quantum*, Vol. 2, No. 2, p. 020329 (2021).
[42] McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R. and Neven, H.: Barren plateaus in quantum neural network training landscapes, *Nature communications*, Vol. 9, No. 1, pp. 1–6 (2018).
[43] ANIS, M. S. et al.: Qiskit: An Open-source Framework for Quantum Computing (2021).
[44] Chandola, V., Banerjee, A. and Kumar, V.: Anomaly detection: A survey, *ACM computing surveys (CSUR)*, Vol. 41, No. 3, pp. 1–58 (2009).
[45] Hara, S., Ono, T., Okamoto, R., Washio, T. and Takeuchi, S.: Anomaly detection in reconstructed quantum states using a machine-learning technique, *Physical Review A*, Vol. 89, No. 2, p. 022104 (2014).
[46] Hara, S., Ono, T., Okamoto, R., Washio, T. and Takeuchi, S.: Quantum-state anomaly detection for arbitrary errors using a machine-learning technique, *Physical Review A*, Vol. 94, No. 4, p. 042341 (2016).
[47] Ide, T.: Introduction to Anomaly Detection Using Machine Learning–a Practical Guide With R (in Japanese), *Corona Publishing*, pp. 132–139 (2015).
[48] Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U. and Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery, *International conference on information processing in medical imaging*, Springer, pp. 146–157 (2017).
[49] De Haan, L., Ferreira, A. and Ferreira, A.: *Extreme value theory: an introduction*, Vol. 21, Springer (2006).

ⓒ 2022 Information Processing Society of Japan

Appendix

A.1 Rough explanation on the shape of Fig.4

Here, we give the rough explanation of the dependence of the mean approximation error on the number of training data, presented in Fig.4 of Sec.3.2.2. Throughout this section, we assume that the number of shots N_s is sufficiently large to hold the asymptotic theory.

We first focus on the case of small number of training data M, where the approximation error behaves like $M^{-1/2}$. In this case, the training data $\{\boldsymbol{x}_i\}_{i=1}^M$ would be well separated from each other and the optimal transport plan $\{\pi_{i,j}\}_{i,j=1}^M$ is not expected to be affected by the number of shots N_s. In most cases, the number of non-zero elements of optimal transport plan $A = \{(i,j)|\pi_{i,j} > 0\}$ is M and the value of those are $1/M$.

The estimated value of the ground cost $c_{\mathrm{local},i,j}$ of Eq.(3.3) with N_s shots is given by $\tilde{c}^{(N_s)}_{\mathrm{local},i,j} = \sqrt{\frac{1}{n}\sum_{k=1}^n \frac{1}{N_s}\sum_{s=1}^{N_s} X^{(s)}_{i,j,k}}$, where $X^{(s)}_{i,j,k}$ are random variables following the Bernoulli distribution with probability $1 - p^{(k)}_{i,j}$. Due to the central limit theorem, the inside of the root $Y^{(s)}_{i,j} = \frac{1}{n}\sum_{k=1}^n X^{(s)}_{i,j,k}$ asymptotically converge to a normal distribution $\sqrt{N_s}(\sum_{s=1}^{N_s} Y^{(s)}_{i,j}/N_s - \mu_{i,j}) \sim \mathcal{N}(0, \sigma^2_{i,j})$ with mean $\mu_{i,j} = \sum_{k=1}^n (1 - p^{(k)}_{i,j})$ and variance $\sigma^2_{i,j} = \sum_{k=1}^n (1 - p^{(k)}_{i,j}) p^{(k)}_{i,j}$. Thus, delta method tells us that the approximation error of the ground cost follows $\sqrt{N_s}\left(\tilde{c}^{(N_s)}_{\mathrm{local},i,j} - c_{\mathrm{local},i,j}\right) \sim \mathcal{N}\left(0, \frac{\sigma^2_{i,j}}{4\mu_{i,j}}\right)$.

Assume that the means and variances are almost the same for all the components, i.e., $\mu_{i,j} \approx \mu$, $\sigma_{i,j} \approx \sigma$ $\forall i, j$, the approximation error of the optimal transport loss due to the number of shots can be written as

$$\begin{aligned}
\mathcal{L}_{\tilde{c}^{(N_s)}_{\mathrm{local}}} - \mathcal{L}_{c_{\mathrm{local}}} &= \frac{1}{M}\sum_{(i,j)\in A}\left(\tilde{c}^{(N_s)}_{\mathrm{local},i,j} - c_{\mathrm{local},i,j}\right) \\
&\sim \mathcal{N}\left(0, \sum_{(i,j)\in A}\frac{\sigma^2_{i,j}}{4N_s\mu_{i,j}M^2}\right) \\
&\approx \mathcal{N}\left(0, \frac{\sigma^2}{4N_s\mu M}\right).
\end{aligned} \tag{A.1}$$

Thus the approximation error with small number of training data behaves like $(N_s M)^{-1/2}$ under the condition shown here.

On the other hand, behavior of the approximation error in the range of large number of training data M is explained by the theory of the extreme value distribution [49]. In the case of large number of training data, it would be expected that there are many ground costs $c_{\mathrm{local},i,j}$ with almost the same value. As an extreme case, consider the case where all ground costs have a common constant value, $c_{\mathrm{local},i,j} = c$, $\forall i, j$. Again assume that the number of shots N_s is sufficiently large, then the ground cost $\tilde{c}^{(N_s)}_{\mathrm{local},i,j}$ follows a normal distribution, which we denote as $\mathcal{N}\left(c, \frac{\sigma^2}{N_s}\right)$. Then, using i.i.d random numbers $\{X_{i,j}\}_{i,j=1}^M$ which follow a normal distribution $\mathcal{N}\left(0, \frac{\sigma^2}{N_s}\right)$, the approximation error can be written as

$$\begin{aligned}
&\mathcal{L}_{\tilde{c}^{(N_s)}_{\mathrm{local}}} - \mathcal{L}_{c_{\mathrm{local}}} \approx \min_{\{\pi_{i,j}\}_{i,j=1}^M} \sum_{i,j=1}^M X_{i,j}\pi_{i,j}, \\
&\text{subject to} \quad \sum_{i=1}^M \pi_{i,j} = \frac{1}{M}, \sum_{j=1}^{N_g}\pi_{i,j} = \frac{1}{N_r}, \pi_{i,j} \geq 0.
\end{aligned} \tag{A.2}$$

Now we approximate this minimization by greedy algorithm, i.e., consider first obtaining the minimum value X_{i_1,j_1} from the M^2 components, and then the second minimum value X_{i_2,j_2} from the rest $(M-1)^2$ components other than i-th row and j-th column, and so on. Denoting the cumulative distribution function of a random variable $X_{i,j}$ as $F(x)$, the distribution of the minimum value of the k data can be written as

$$\begin{aligned}
G(x,k) &= 1 - (1 - F(x))^k, \\
p(x,k) &= \frac{dG(x,k)}{dx} = k\frac{dF(x)}{dx}(1 - F(x))^{k-1}.
\end{aligned} \tag{A.3}$$

Then the probability density at which $x_1, x_2, x_3, \ldots, x_M$ are obtained from the greedy algorithm is given as

$$\begin{aligned}
&p(x_1, x_2, \ldots, x_M) \\
&= p(x_1, M^2)\frac{p(x_2,(M-1)^2)\theta(x_2 - x_1)}{1 - G(x_1,(M-1)^2)}\frac{p(x_3,(M-2)^2)\theta(x_3 - x_2)}{1 - G(x_2,(M-2)^2)} \\
&\qquad \times \cdots \times \frac{p(x_M, 1^2)\theta(x_M - x_{M-1})}{1 - G(x_{M-1}, 1^2)} \\
&= \prod_{k=1}^M \frac{k^2}{2k-1} p(x_k, 2k-1)\theta(x_k - x_{k-1}),
\end{aligned} \tag{A.4}$$

where $\theta(x)$ denotes a step function, and we set $x_0 = -\infty$ in the last expression. Finally, we approximate this expression by the mode. Then, from the theory of the extreme value distribution, the mode of $p(x,k)$ can be written as $x_{mode} \approx -\sigma\sqrt{2\ln M/N_s}$ and we reach

$$\begin{aligned}
\mathcal{L}_{\tilde{c}^{(N_s)}_{\mathrm{local}}} - \mathcal{L}_{c_{\mathrm{local}}} &\approx \frac{\sigma}{\sqrt{N_s}}\left(\frac{1}{M}\sum_{k=1}^M \sqrt{2\ln(2k-1)}\right) \\
&\approx \frac{\sigma}{\sqrt{N_s}}\sqrt{2\ln(2M-1)}.
\end{aligned} \tag{A.5}$$

Thus we can roughly understand that the approximation error with large number of training data behaves like $N_s^{-1/2}\sqrt{\ln(M)}$.

ⓒ 2022 Information Processing Society of Japan