Permutation-Accelerated Distribution Learning With Quantum Generative Adversarial Networks

Allen Mi
Yale University
New Haven, CT
allen.mi@yale.edu

Abstract

Efficient preparation of input states is the prerequisite of many exponential-speedup quantum algorithms. The quantum generative adversarial network (qGAN) model was proposed to facilitate fast approximate loading of quantum states from discrete distributions. In this paper, we propose and identify phantom modes as a phenomenon unique to multimodal qGAN training. We provide an information-theoretic characterization of this effect, and propose descending-order permutation as a means of mitigation. We show that this method is highly effective in improving accuracy and convergence behavior via empirical evaluation on the MNIST dataset. Finally, we suggest circuit-based methods for implementing permutations.

1. Introduction

In recent years, the field of quantum computation has seen rapid development in both theoretical and experimental fronts. A main factor that motivates this development is the prospect that quantum algorithms may offer exponential speedup over their classical counterparts. Examples of these quantum algorithms include Grover's algorithm for database search [3], Shor's algorithm for integer factorization [11], and HHL algorithm for solving linear systems of equations [5]. In practice, however, there lies a significant hurdle in the application of these results: Many exponential-speedup algorithms (e.g. HHL) admit arbitrary quantum states as input, and the exact preparation of such a quantum state on n qubits requires $\mathcal{O}(2^n)$ gates [4]. In the exact case, long preparation circuit that grows exponentially is required as a preamble, effectively nullifying the above mentioned theoretical speedup while rendering scalable experimental implementations infeasible.

Recently, quantum generative adversarial networks (qGANs) have been proposed as a means to efficiently generate a quantum state that approximates the target in $\mathcal{O}(\text{poly}(n))$ gates [10, 2]. qGANs share the same working principles with their classical (i.e., non-quantum) counterparts: A generator and a discriminator are trained with input data and compete in the form of a zero-sum game. The optimal result is achieved when the system reaches Nash equilibrium.

Developing a robust and efficient means of generating quantum states from arbitrary discrete distributions is critical for practical implementation of exponential-speedup quantum algorithms. This is a prerequisite for scalable and economical quantum computation, and the use of qGANs is a promising direction towards achieving this objective. However, as with the classical case [9], learning multimodal distributions is challenging for qGANs [13, 8]. We discuss some recent progress regarding qGAN training in then next section.

2. Background

In this section, we discuss some recent progress on qGANs and review four important concepts in information theory necessary for understanding the training of qGANs.

2.1. Prior Work

In [7], Hu et al. produced the first physical implementation of qGANs. This paper serves as a proof-of-principle of the qGAN approach in a superconducting quantum electrodynamics architecture. The authors successfully demonstrated the viability of qGAN and produced detailed information regarding convergence behavior of the quantum state on the Bloch sphere. However, the results were limited to the single-qubit scenario.

In [13], Zoufal et al. produced a standard implementation of qGANs following [2]. The generator structure consists of layers of n rotation gates along the Y axis, as well as a ring of CZ gates for entanglement. The discriminator is a heuristically-derived 3-layer classical neural network. In simulation, the paper achieved good estimation of unimodal (log-normal, triangular) and bimodal (2-

gaussian mixture) distribution loading in small state spaces (size 8 with 3 qubits). These results were successfully reproduced on physical quantum devices at IBM Q. Additionally, the authors investigated the convergence behavior with different initialization patterns, and explored applications of qGAN distribution loading with quantum amplitude estimation (QAE) [4] in a computational finance scenario. The paper omitted evaluations of the authors' approach on a larger number of qubits, with a more complex distribution, or in higher dimensions.

In [8], Huang et al. demonstrated complex distribution learning with qGANs with MNIST image generation. The sample images are downscaled to 8×8 and normalized to resemble a discrete probability distribution on a state space of size 64. The samples are then partitioned vertically into patches. Each patch is renormalized and serialized into a simpler multimodal distribution. A separate generator is assigned to each patch, and the generated image is reconstructed from the patches before entering the discriminator as input. The simulation and experimental results following this approach are comparable and both promising. However, partitioning the image into patches significantly oversimplifies the problem of distribution loading, as it fails to recreate the complexity of large superpositions (over 2^n basis states on n qubits).

2.2. Theoretical preliminaries

This section is devoted to the review of four important concepts in information theory. Given a pair of continuous random variables (RVs) (X, Y) with values over the space $\mathcal{X} \times \mathcal{Y}$, we define the mutual information between them as [1]

$$I(X;Y) = D_{KL}(P_{(X,Y)}||P_X \otimes P_Y) \tag{1}$$

where D_{KL} is the Kullback-Leibler divergence (or relative entropy) and $P_{(X,Y)}$, P_X, P_Y are their joint distribution, two marginal distributions respectively. For two continuous probability distributions P, Q over the probability space $\mathcal{Z}$,

$$D_{KL}(P||Q) = \int_{\mathcal{Z}} dz P(z) \log \frac{P(z)}{Q(z)}. \tag{2}$$

Mutual information provides a measure of mutual dependence between two RVs, in the case where two RVs are independent, $I(X;Y) = 0$ from the property of D_{KL}.

Another concept related to RVs is the joint entropy. Using the same RVs above, the joint entropy is defined as [1]

$$H(X,Y) = -\int_{\mathcal{X}} dx \int_{\mathcal{Y}} dy P(x,y) \log_2[P(x,y)] \tag{3}$$

where $P(x, y)$ is the joint probability distribution. Joint entropy measures the uncertainty between two RVs, if $P(x, y) = 0$, $H(X, Y)$ is defined to be 0.

Next, we cross entropy and the Wasserstein metric for probability distributions. For two continuous probability distributions p, q over the same support $\mathcal{X}$, their cross entropy is

$$H(p,q) = -\int_{\mathcal{X}} dx P(x) \log Q(x) \tag{4}$$

where P, Q are their probability density functions. The cross entropy is motivated by the Kraft-McMillan theorem, and represents the average message length for distinguishing an event drawn from an estimated distribution instead of the true distribution.

The Wasserstein metric evaluates the distance between two distributions over a given feature space. We can formalize the Wasserstein metric as the following. Given two signatures P, Q with m, n clusters respectively,

$$\begin{aligned} P &= (p_1, w_{p_1}), \ldots, (p_m, w_{p_m}) \\ Q &= (q_1, w_{q_1}), \ldots, (q_n, w_{q_n}) \end{aligned} \tag{5}$$

where p_i, q_j are the cluster representatives and w_{p_i}, w_{q_j} are the weights of the clusters. Define the ground distance between p_i, q_j as $D = [d_{i,j}]$ and the flow between them as $F = [f_{i,j}]$. The problem is to find the minimal work cost

$$C(P, Q, \mathbf{F}) = \sum_{i=1}^{m} \sum_{j=1}^{n} f_{i,j} d_{i,j}. \tag{6}$$

Once it is solved by evoking a series of constraints related to weights and flows, we can formulate the Wasserstein metric as the following:

$$\mathcal{W}(P,Q) = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{i,j} d_{i,j}}{\sum_{i=1}^{m} \sum_{j=1}^{n} f_{i,j}}. \tag{7}$$

3. Approach

In this section, we first describe the general construction of qGANS. Then, we discuss the implementation details of the qGAN developed in this work. Finally, we describe and produce an information-theoretic characterization of the phenomenon of phantom modes. Based on this characterization, we propose the *descending-order permutation* on the target distribution $\mathcal{D}$ as a means of mitigation. We show that under a 2-gaussian mixture, the permuted model significantly outperforms the original in the worst-case estimate.

3.1. General Construction of qGANs

Very similar to its classical counterparts, the qGAN model is partitioned into a quantum generator network G and a classical discriminator network D. For a system on n qubits, G usually consists of a quantum circuit with k layers, each containing n rotation gates and n controlled $-$

$\{X, Y, Z\}$ gates to facilitate entanglement [13]. The generator parameters are therefore kn reals. D consists of a classical classifier with an unspecified number of parameters. In the training phase, G attempts to generate samples that emulate target distribution $\mathcal{D}$, a discrete probability distribution on 2^n basis states, while D attempts to discern samples drawn from G and $\mathcal{D}$. As with the classical case, the cross-entropy loss is often used, and gradient-based loss optimization is performed on the generator and discriminator parameters. The generator and discriminator losses converge when the system reaches Nash equilibrium.

3.2. Implementation-Specific Information

For the quantum generator G, we adopt a parametrized variational quantum circuit inspired by the work of Zoufal et al. [13]. The circuit is initialized by performing a Hadamard gate on each qubit i, where $0 < i \leq n$. Then, to implement each variational parameter θ_j $(0 < j \leq k)$, we perform single-qubit Pauli-Y rotations (R_Y) on all qubits, totaling n gate operations. Since R_Y can be decomposed into Pauli-X-rotations (R_X) and Pauli-Z-rotations (R_Z), this gate choice help us reduce the circuit depth by around $\frac{1}{3}$ compared to other quantum generators consisting of R_X and R_Z gates [2]. Correlation is implemented among qubits via two-qubit controlled-Z gates (CZ). We entangle all qubit-pairs after each R_Y rotation gate layer with n CZ gates except for the last one. At the end of the circuit, qubits are measured in the Z basis, and the classical readout is then fed into the discriminator D for comparison with input data sets.

For the classical discriminator D, we employ a fixed 3-layer dense neural network. The layers consist of 50, 10, and 1 node(s) respectively. The activation function for the first two layers is leaky ReLU with parameter $\alpha = 0.01$. The third layer adopts a sigmoid activation function.

4. Mitigating "Phantom Modes"

In this section, we describe and characterize the phenomenon of "phantom modes", a common pitfall unique to qGAN training. Based on this characterization, we propose the *descending-order permutation* on the target distribution $\mathcal{D}$ as a means of mitigation. We propose circuit-based implementations of the permutation that preserve the structure of learned probability distributions.

4.1. Description of Phenomenon

The phenomenon of "phantom modes" occurs when the generated distribution contains modes that are nonexistent in the target distribution. This effect can usually be observed when qGAN training is performed on a target distribution $\mathcal{D}$ with $m \geq 2$ distinct modes, as shown in Figure 1. At about epoch 150, the other two cells on the diagonal becomes modes in the output distribution. These modes are not present in the target distribution.

This phenomenon is often accompanied by abnormalities in the generator and discriminator loss functions. This is demonstrated in Figure 2.

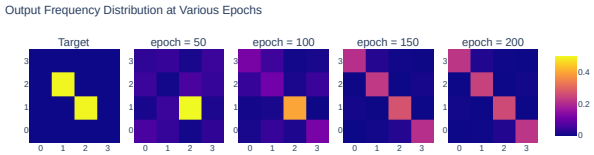

Figure 1. The phantom modes effect is observed when $\mathcal{D}$ is a simple bimodal distribution on two cells of the 4×4 lattice diagonal. Observe that mode collapse occurs first.

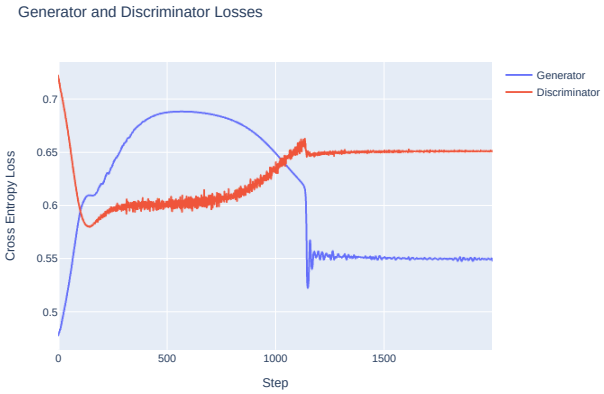

Figure 2. Abnormalities in the generator and discriminator loss functions can be seen between steps 1000 and 1500. Since 10 steps are performed per epoch, this abnormally likely corresponds to the emergence of phantom nodes in Figure 1.

4.2. Characterization

Consider a system of n qubits. The results from circuit measurements are labeled by length-n binary strings, forming a output space of dimension 2^n. Within this space, each combination of measurement frequency distributions is represented by a nonnegative unit vector with respect to the l_1 norm. Recall that the quantum generator is specified by $\mathcal{O}(n)$ parameters. Since the output space is exponential in size, the system is severely underdetermined. We conjecture that this is a main reason behind the phantom modes phenomenon, as well as a reason why this effect is not observed in classical GAN training.

We wish to characterize the correlation between the labels in frequency distribution due to this effect. For arbi-

trary bitstrings a and b, let X_a and X_b denote the random variables that correspond to the frequencies of a and b in the generated distribution respectively. When the system is underdetermined, the effect of single-qubit frequency distribution takes precedence in the joint distribution of the bitstring frequencies. Hence we make the following assumption:

Assumption 4.1. *The mutual information $I(X_a; X_b)$ between X_a and X_b is positively correlated with the number of coordinates i where $a_i = b_i$.*

Under this assumption, we may use the Hamming distance d_H as a well-defined metric for the mutual dependence between bitstring frequencies. Recall that $\forall a, b \in \{0,1\}^n$,

$$d_H(a,b) = \#\{i \in \{0,1,\ldots,n-1\} \mid a_i \neq b_i\}. \tag{8}$$

Thus, $d_H(a,b)$ is negatively correlated with $I(X_a; X_b)$. Under d_H, the set S of 2^n bitstrings form an n-dimensional hypercube graph Q_n, with each vertex $v \in V(Q_n)$ labeled by $v \in S$, and each edge $e = (v,w) \in E(Q_n)$ iff $d_H(v,w) = 1$.

We investigate the phantom modes phenomenon within this framework. Given $v, w \in V(Q_n)$ and some arbitrary $u \neq v, w \in V(Q_n)$, we estimate the total influence X_v and X_w exert on X_u by utilizing the variation of information d_V. Recall that

$$d_V(X_a, X_b) := H(X_a, X_b) - I(X_a; X_b) \tag{9}$$

is a well-defined metric, where $H(X_a, X_b)$ denotes the joint entropy between X_a and X_b. Since $I(X_a; X_b) \sim -d_H(a,b)$, we have the following approximation:

$$d_V(X_v, X_u) + d_V(X_w, X_u) \sim -d_H(v,u) - d_H(w,u). \tag{10}$$

It is then manifest that given v, w, the vertices that are most influenced by X_v and X_w are situated on a shortest path between v and w. Concretely, let $\mathrm{SP}_{v,w}$ denote the set of shortest paths between v and w in Q_n, where $\forall P \in \mathrm{SP}_{v,w}$, P is a subgraph of Q_n. Then we define

$$\mathfrak{P}_{v,w} := \bigcup_{P \in \mathrm{SP}_{v,w}} V(P) \setminus v, w, \tag{11}$$

where each $u \in \mathfrak{P}_{v,w}$ is a vertex distinct from v or w on a shortest path between v and w. Utilizing $\mathfrak{P}_{v,w}$, we produce the following conjecture that characterizes the phantom modes.

Conjecture 4.2. *Let $u \in V(Q_n)$ be a phantom mode, in that $X_u \gg \hat{X}_u$, where $\hat{X}_u$ is the frequency of u under the target distribution. Then $\exists v, w \in V(Q_n)$ such that $\hat{X}_v, \hat{X}_w \gg \hat{X}_u$, and $u \in \mathfrak{P}_{v,w}$.*

In order to mitigate occurrences of phantom modes, we wish to reduce the maximum discrepancy between $\hat{X}_{v,w}$ and $\hat{X}_u$ across all choices of u, v, w. Concretely, given $v, w \in V(Q_n)$, we define a quadratic cost function for $u \in \mathfrak{P}_{v,w}$

$$\phi_{v,w}(u) := (\hat{X}_v - \hat{X}_u)^2 + (\hat{X}_w - \hat{X}_u)^2, \tag{12}$$

and the associated worst-case cost for target distribution $\mathcal{D}$

$$\Phi(\mathcal{D}) := \max_{v,w \in V(Q_n), u \in \mathfrak{P}_{v,w}, \hat{X}_u < \hat{X}_{v,w}} \phi_{v,w}(u). \tag{13}$$

To perform mitigation, we wish to find a target distribution $\mathcal{D}'$ that is *congruent* to $\mathcal{D}$, such that

$$\Phi(\mathcal{D}') < \Phi(\mathcal{D}). \tag{14}$$

The congruence condition is given as follows:

$$\mathcal{D} \equiv \mathcal{D}' \text{ iff } \{\mathcal{D}\} = \{\mathcal{D}'\}, \tag{15}$$

where $\{\mathcal{D}\}$ and $\{\mathcal{D}'\}$ denote the multisets containing all 2^n frequencies within each distribution respectively. In other words, we consider two distributions congruent if and only if they contain the same multiset of frequencies.

4.3. Proposed Mitigation

Given the characterization above, we propose the *descending-order permutation* routine that derives a congruent distribution $\mathcal{D}' \equiv \mathcal{D}$ given target distribution $\mathcal{D}$. The procedure is as follows:

1. Under $\mathcal{D}$, sort the label strings into list L via descending order in frequency. Hence

   $$\forall u, v \in V(Q_n), \hat{X}_u \geq \hat{X}_v \text{ iff } u \preccurlyeq_L v, \tag{16}$$

   where $\preccurlyeq_L$ denotes the precedence relation in L. Tie-breaking is arbitrary.

2. Define a permutation $f : V(Q_n) \to V(Q_n)$ via $f : u \mapsto L[\mathrm{decimal}(u)]$. Then D' is given as

   $$\hat{X}_u^{\mathcal{D}'} = \hat{X}_{f(u)}^{\mathcal{D}}, \forall u \in V(Q_n), \tag{17}$$

   where $X_w^{\mathcal{D}}$, $X_w^{\mathcal{D}'}$ denote the frequency of some w in D and D' respectively. Since f is bijective, we also have

   $$\hat{X}_v^{\mathcal{D}} = \hat{X}_{f^{-1}(v)}^{\mathcal{D}'}, \forall v \in V(Q_n). \tag{18}$$

The bijectivity of f directly implies that $\mathcal{D}' \equiv \mathcal{D}$. Additionally, $\forall u, v \in V(Q_n)$, $\hat{X}_u^{\mathcal{D}'} \geq \hat{X}_v^{\mathcal{D}'}$ iff $u \leq v$ in the lexicographic order. After $\mathcal{D}'$ is constructed, training is performed with $\mathcal{D}'$, and the resulting distribution is mapped back to the original labeling via f^{-1}.

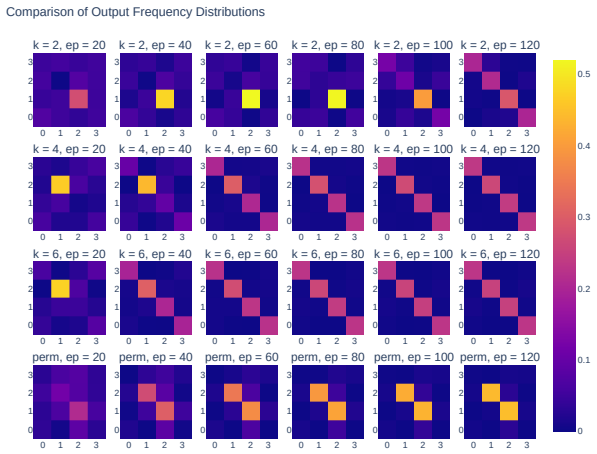

Figure 3. Comparison of output frequency distributions with various layer depths k and at various epochs in the training process. The first 3 rows correspond to the original model, and the last row corresponds to the permuted model with layer depth $k = 2$.

4.4. Circuit-Based Implementations

Instead of recording f^{-1} and applying it to the qGAN output, it is possible to integrate it into the generator circuit itself. Recall that f can be expressed as a 2^n-by-2^n permutation matrix M, with its inverse M^{-1} corresponding to f^{-1}. Since $f^{-1} \circ f$ equals identity, we have $M^{-1}M = I$, implying that M^{-1} is unitary. In light of this, M^{-1} can be viewed as a unitary gate that acts on all n qubits. The effect of M^{-1} resembles a permutation on the joint basis states of all qubits. In practical applications, M^{-1} can be decomposed to $2n - 1$ single-target gates [12]. Each single-target gate can further be represented by $O(n^2)$ single- and two-qubit gates, resulting in an $O(n^3)$ cost in circuit depth [6].

5. Empirical Evaluation

In this section, we evaluate the empirical performance of qGAN training, with a focus on comparing the cases with and without descending-order permutation.

5.1. Setup

Throughout this section, we perform a total of 10^4 random samples, evenly separated into 10 batches, on both $\mathcal{D}$ and G. Each epoch iterates through the batches in 10 steps, and the generator G and discriminator D parameters are updated in lockstep.

5.2. Diagonal Bimodal Distribution

Recall the diagonal bimodal distribution presented in Figure 1. Figure 3 provides a comparison of the training outcome within 120 epochs for layer depths $k = 2, 4, 6$ in the original model, as well as $k = 2$ in the permuted model. Observe that in the original model, as the number of layers increases, the transition point between mode collapse and phantom modes moves earlier in the training process. Also observe that with the original model, increasing the number of layers linearly does not resolve the issue of phantom modes. With $k = 2$ and at epoch 100, the original model begins its transition to the phantom modes state, while the permuted model has largely converged to the target distribution.

5.3. MNIST Example

We evaluate the training performance of the models on the 16×16 MNIST dataset, obtained from a cubic downsampling of the original 28×28 dataset. Each greyscale image is converted to a discrete probability distribution via normalization. Due to limitations in computational power, we take a random sample of size 36 from the MNIST dataset for evaluation.

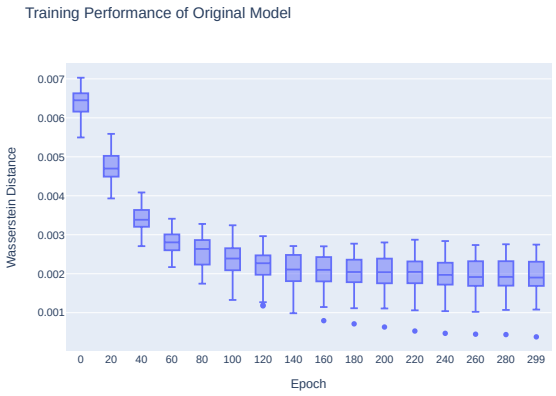

Figure 4. MNIST training performance of the original model. The Wasserstein distance converges at around 2×10^{-3} with large variations across images.

To support an output space of size 256, we employ 8 qubits. Both models are trained with $k = 2$ layers for 300 epochs. For each model, we measure its performance by taking the Wasserstein distance between the output distribution and the target distribution at various epochs. The results are presented in Figure 4 and 5. Observe that the Wasserstein distance of both models decreases rapidly at the start of the training. The two models share similar convergence rates, but the permuted model significantly outperforms the vanilla in convergence value.

Figures 6 and 7 at the end of the report depict the frequency distributions of the two models. The original model is able to capture the rough position of the modes in the

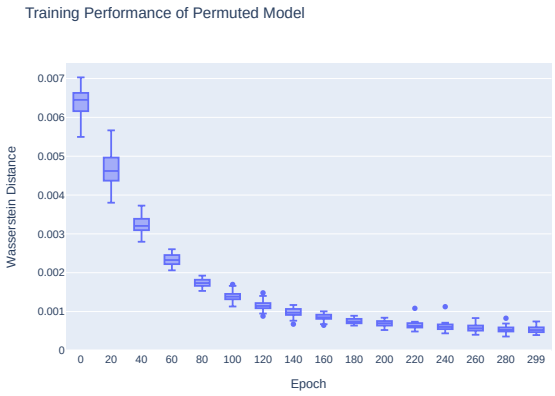

Figure 5. MNIST training performance of the permuted model. The Wasserstein distance converges at around 5×10^{-4} with very small variations across images.

target distributions, but the phantom modes effect prevents it from acquiring the correct distributions. The permuted model is able to acquire the target distributions with good accuracy.

6. Conclusion

In this project, we identified the "phantom modes" phenomenon as a training artifact in multimodal distribution loading via qGAN. We attributed the phenomenon to underdeterminism in the quantum generator, which is further due to the exponential—linear mismatch in the dimensionality of the output space and the available number of parameters. This mismatch is intrinsic to the design of qGAN, and is essential to its efficiency in gate depth. We characterized the phenomenon by localizing its effect to shortest path vertices between real modes on the hypercube graph. Based on this characterization, we proposed descending-order permutation as a method for mitigating phantom modes. We validated its effectiveness by applying the permuted model to the MNIST dataset, from which we concluded that the model is an effective mitigation of the phantom modes effect, and it achieves significant improvement in training accuracy over the original model. In comparison to previous works, the performance demonstrated by descending-order permutation model is at or exceeds the state-of-the-art.

Future work is concentrated in two fronts. Theoretically, we would like to characterize classes of permutations with simple circuit representations. Utilizing such permutations may help reduce the $O(n^3)$ circuit depth required for reconstructing the generator output. Empirically, we hope to evaluate the qGAN training process on IBM's superconducting devices and examine the effectiveness of descending-order permutation on noisy hardware.

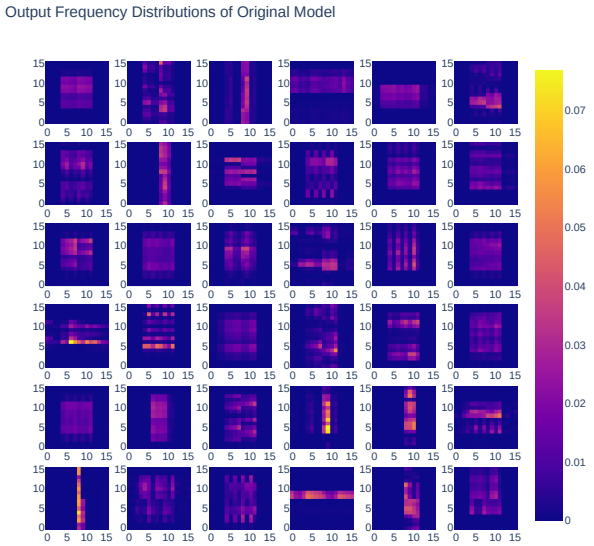

Figure 6. MNIST output frequency distributions of the original model. Note that the phenomenon of phantom modes is pronounced, as the output distributions contains high-frequency "chunks" that surround the modes of the target distribution.

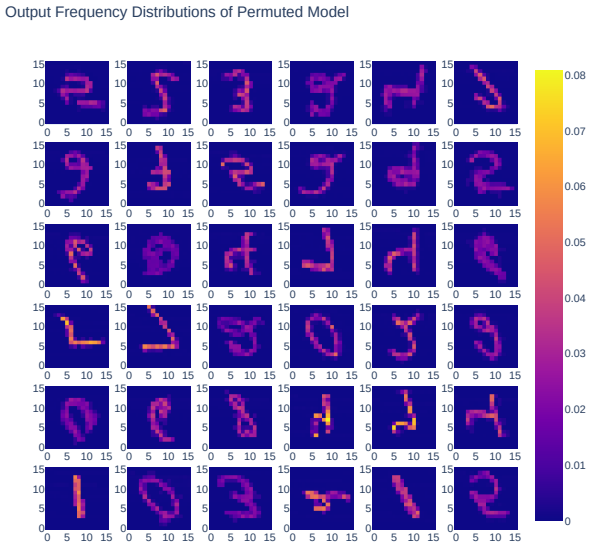

Figure 7. MNIST output frequency distributions of the permuted model. Observe that the output distributions largely resemble the target distributions in the MNIST dataset.

References

[1] T. M. Cover and J. A. Thomas. *Elements of Information Theory 2nd Edition (Wiley Series in Telecommunications and Signal Processing)*. Wiley-Interscience, 2006.

[2] P.-L. Dallaire-Demers and N. Killoran. Quantum generative adversarial networks. *Phys. Rev. A*, 98:012324, Jul 2018.

[3] L. K. Grover. A fast quantum mechanical algorithm for database search. In *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing*, STOC '96, page 212–219, New York, NY, USA, 1996. Association for Computing Machinery.

[4] L. K. Grover. Synthesis of quantum superpositions by quantum computation. *Phys. Rev. Lett.*, 85:1334–1337, Aug 2000.

[5] A. W. Harrow, A. Hassidim, and S. Lloyd. Quantum algorithm for linear systems of equations. *Phys. Rev. Lett.*, 103:150502, Oct 2009.

[6] Y. He, M.-X. Luo, E. Zhang, H.-K. Wang, and X.-F. Wang. Decompositions of n-qubit toffoli gates with linear circuit complexity. *International Journal of Theoretical Physics*, 56(7):2350–2361, Jul 2017.

[7] L. Hu, S.-H. Wu, W. Cai, Y. Ma, X. Mu, Y. Xu, H. Wang, Y. Song, D.-L. Deng, C.-L. Zou, and L. Sun. Quantum generative adversarial learning in a superconducting quantum circuit. *Science Advances*, 5(1), 2019.

[8] H.-L. Huang, Y. Du, M. Gong, Y. Zhao, Y. Wu, C. Wang, S. Li, F. Liang, J. Lin, Y. Xu, R. Yang, T. Liu, M.-H. Hsieh, H. Deng, H. Rong, C.-Z. Peng, C.-Y. Lu, Y.-A. Chen, D. Tao, X. Zhu, and J.-W. Pan. Experimental Quantum Generative Adversarial Networks for Image Generation. *arXiv e-prints*, page arXiv:2010.06201, Oct. 2020.

[9] V. Keswani, O. Mangoubi, S. Sachdeva, and N. K. Vishnoi. GANs with First-Order Greedy Discriminators. *arXiv e-prints*, page arXiv:2006.12376, June 2020.

[10] S. Lloyd and C. Weedbrook. Quantum generative adversarial learning. *Phys. Rev. Lett.*, 121:040502, Jul 2018.

[11] P. W. Shor. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. Comput.*, 26(5):1484–1509, Oct. 1997.

[12] M. Soeken, F. Mozafari, B. Schmitt, and G. D. Micheli. Compiling permutations for superconducting qpus. In *2019 Design, Automation Test in Europe Conference Exhibition (DATE)*, pages 1349–1354, 2019.

[13] C. Zoufal, A. Lucchi, and S. Woerner. Quantum generative adversarial networks for learning and loading random distributions. *npj Quantum Information*, 5(1):103, Nov 2019.