Learning Continuous Probability Distributions with Parameterized Observables

Kevin Shen,[1,2,3,*] Andrii Kurkin,[2,1,3] Hao Wang,[2,3] Adrián Pérez Salinas,[2,4] Elvira Shishenina,[1] and Vedran Dunjko[2,3]

[1] *BMW Group, 80788 München, Germany*
[2] *⟨aQa^L⟩ Applied Quantum Algorithms, Universiteit Leiden*
[3] *Institute LIACS, Universiteit Leiden, the Netherlands*
[4] *Instituut-Lorentz, Universiteit Leiden, the Netherlands*

(Dated: May 23, 2024)

I. OVERVIEW

Generative modelling, i.e., the learning and sampling from probability distributions, has raised increasing attention in academia and industry in recent times. Quantum computation arguably holds a great potential in achieving superclassical performances in certain albeit perhaps very special generative modelling tasks, as evidenced by, e.g., some distributions are efficiently generable quantumly but not classically [1, 2] and some classically efficiently generable distributions are efficiently learnable quantumly but not classically [3].

Motivated by potential industry-relevant generative modelling applications, which often concern with continuous probability distributions, in this work we study a type of quantum models known as Expectation Value Samplers (EVSs) [4, 5], which also appeared as the generators in some Quantum Generative Adversarial Networks (QGANs) [4, 6, 7]. Different from the more commonly known Quantum Circuit Born Machines [8], EVSs involve the computation of expectation values of observables on quantum states prepared by parameterized quantum circuits (PQCs) with random gate parameters. Under common complexity theory assumptions [9], with such computations EVSs should be classically intractable in general cases. In addition, on the face of it, EVSs have the attractive property that the generated distributions can have arbitrarily large dimensions independent of the number of qubits used. EVSs have recently been studied more intensely for these reasons. However, many questions remain, including the design of highly expressive yet trainable models.

In this work, we propose a more general version of EVS where the observables are parameterized and trained as well, with the goal of being expressive, yet trainable (in the sense of sample efficiency). We prove that except for a few edge cases, training the observables strictly increases the models' expressivity. Next, we study in detail the settings that the observables are kept $\mathcal{O}(1)-$local and bounded during training. We derive efficient schemes for computing optimal settings w.r.t the observables which we use to propose a new efficient training algorithm, adapted from the Wasserstein Generative Adversarial Networks (WGANs) [10]. We consider multiple variants of the training algorithm: using conventional sample averaging methods or classical shadow methods [11], updating the observable parameters via gradient methods or semidefinite programming, and compare their performances on different datasets.

II. EXPECTATION VALUE SAMPLER

We formally define EVSs below.

Definition 1. *(Expectation value sampler) An expectation value sampler (EVS) is specified by three components:*

1. *A continuous random vector $\boldsymbol{z}$ of some dimension K, with known distribution and efficient sampling method, e.g. uniform or Gaussian. We write the support of $\boldsymbol{z}$ as $\mathcal{Z} \subset \mathbb{R}^K$.*
2. *A parametric family of quantum circuits $U(\boldsymbol{z}, \boldsymbol{\theta})$ where $\boldsymbol{\theta} \in \Theta \subset \mathbb{R}^d$ is a parameter vector of some dimension d. We denote n as the number of qubits that $U(\boldsymbol{z}, \boldsymbol{\theta})$ acts on, and write $N = 2^n$.*
3. *A tuple of observables $(O_i)_{i=0}^{M-1}$ of size M.*

We can then define a parametric family of deterministic functions $g : \mathcal{Z} \times \Theta \longrightarrow \mathcal{Y} \subset \mathbb{R}^M$, where each coordinate function writes as:

$$g_i(\boldsymbol{z}, \boldsymbol{\theta}) = \langle 0|U^\dagger(\boldsymbol{z}, \boldsymbol{\theta}) O_i U(\boldsymbol{z}, \boldsymbol{\theta})|0\rangle \tag{1}$$

EVS is a model that repeatedly draws samples $\boldsymbol{z}$, process them with g and return the outputs.

* kevin.shen@bmwgroup.com

In the context of generative modelling, one will train the circuit parameters $\boldsymbol{\theta}$ of an EVS such that the generated data follows a target probability distribution approximately. Ref. [5] demonstrates that for any defined probability distribution, there exists an EVS that can approximately generate it to arbitrary precision in the Wasserstein-1 metric.

III. EXPECTATION VALUE SAMPLER WITH PARAMETERIZED OBSERVABLES

We propose that, without specific knowledge about target distributions, one should consider a more general model where the observables are parameteried and trained as well to have richer expressivity. For intuition, consider an EVS with fixed observables, changing the circuit parameters $\boldsymbol{\theta}$ effectively changes the eigenvectors of all observables simultaneously. However, to address the eigenvectors of individual observables, one needs to separately apply change-of-basis unitaries after the circuit $U(\boldsymbol{z}, \boldsymbol{\theta})$. In addition, circuits will never affect the eigenvalues of the observables. Thus, training observables offers extra degrees of freedom beyond the circuits.

Definition 2. *(Expectation value sampler with parameterized observables) An EVS with parameterized observables (EVS-PO) is an EVS in which all observables $(O_i)_{i=0}^{M-1}$ are parameterized.*

We proceed to study the consequences of training observables on the expressivity of an EVS, i.e., "If an EVS can express more probability distributions when its observables become trainable, ceteris paribus". For this, we introduce the concept "relative expressivity".

Definition 3. *(Relative expressivity) Consider a probability space $(\mathbb{R}^K, \mathcal{A}, \mu)$ and a measurable space $(\mathbb{R}^M, \mathcal{B})$, where $\mathcal{B}$ are the Borel sets on $\mathbb{R}^M$. Consider then two $(\mathcal{B}, \mathcal{A})$-measurable functions (random variables) with finite variances, denoted as g and h respectively (with abuse of notation, we will also call the induced EVSs g and h). We denote the set of all random variables expressible by g as $S_g = \{g_{\boldsymbol{\theta}} \colon \mathbb{R}^K \to \mathbb{R}^M | \boldsymbol{\theta} \in [0, 2\pi)^d\} \subseteq L^2(\mathbb{R}^K, \mu)$ and likewise for S_h. We say g is strictly more expressive than h if S_g is dense in S_h and S_h is not dense in S_g, which we will denote as $S_h \prec S_g$. The relation $\prec$ is irreflexitive, asymmetric, and transitive, and hence a strict partial order. We then define the non-strict order "at least as expressive as" ($\preceq$) and the identity relation "as expressive as" ($\cong$) analogously.*

We find that up to location and scale parameters, the following statements hold:

Proposition 1. *(Expressivity never decreases if one trains the observables.) For any EVS g, let h be the corresponding EVS-PO, then $g \preceq h$.*

Proposition 2. *(Expressivity does not increase if one trains the observables of a universal EVS.) For any EVS g, let h be the corresponding EVS-PO, if g is universal (i.e., S_g is dense in $L^2(\mathbb{R}^K, \mu)$), then $g \cong h$.*

Proposition 3. *(There are more edge cases in which expressivity does not improve.) There exists a non-universal EVS g and its corresponding EVS-TO h such that $g \cong h$.*

One example is an EVS in which the circuit $U(\boldsymbol{z})|0\rangle$ has no trainable parameters and follows the one-qubit Haar-random distribution, equipped with one observable. No matter what the observable is, the EVS will express the same random variable up to location and scale parameters. Hence, training the observables will not increase expressivity. If we consider, however, at least two observables, or a Haar-random distribution of at least two qubits, then training the observables generally changes the expressed random variable. We can further prove that except for a few edge cases such as above, training the observables strictly increases expressivity of an EVS in general.

IV. OBSERVABLE PARAMETRIZATION, TRAINING AND EXPERIMENTS

For practical usage, we would like to have at most $poly(n)$ many quantum state preparation and measurements to evaluate an EVS. Therefore, we propose to consider using linear combinations of k-local observables with $k \in \mathcal{O}(1)$. In this case, it is convenient to arrange the Pauli basis according to ascending locality, ascending qubit index and the order of I, X, Y, Z, and use a weight matrix $(\alpha_{i,j})$ to keep track of the observables:

$$\forall 1 \leq i \leq M, \quad O_i = \sum_{j=1}^{\binom{n}{k}4^k} \alpha_{i,j} P_j. \tag{2}$$

We propose to train an EVS adversarially against a neural network discriminator, in a way similar to WGANs [10, 12]. Algorithm 1 shows the vanilla version of our training algorithm. However, instead of updating the circuit

Algorithm 1 Vanilla version of the adapted WGAN Training

Require: η, the learning rate. c, the clipping parameter. m, the batch size. n_{critic}, the number of iterations of the critic per generator iteration. $\boldsymbol{\theta_0}$, initial circuit parameters. $\boldsymbol{w_0}$, initial critic parameters. $\boldsymbol{\alpha_0}$, initial observable parameters.

1: **while** $(\boldsymbol{\theta}, \boldsymbol{\alpha})$ have not converged **do**
2: **for** $t = 0, ..., n_{\text{critic}}$ **do**
3: Sample a batch of real data $\{\boldsymbol{x}^{(i)}\}_{i=1}^{m}$.
4: Sample a batch of prior samples $\{\boldsymbol{z}^{(i)}\}_{i=1}^{m} \sim p(\boldsymbol{z})$.
5: $L_D = \frac{1}{m}\sum_{i=1}^{m} f_{\boldsymbol{w}}(\boldsymbol{x}^{(i)}) - \frac{1}{m}\sum_{i=1}^{m} f_{\boldsymbol{w}}(g_{\boldsymbol{\theta},\boldsymbol{\alpha}}(\boldsymbol{z}^{(i)}))$
6: $\boldsymbol{w} \leftarrow \boldsymbol{w} + \eta \cdot \nabla_{\boldsymbol{w}} L_D$
7: $\boldsymbol{w} \leftarrow \text{clip}(\boldsymbol{w}, -c, c)$
8: **end for**
9: Sample a batch of prior samples $\{\boldsymbol{z}^{(i)}\}_{i=1}^{m} \sim p(\boldsymbol{z})$.
10: $L_G = -\frac{1}{m}\sum_{i=1}^{m} f_{\boldsymbol{w}}(g_{\boldsymbol{\theta},\boldsymbol{\alpha}}(\boldsymbol{z}^{(i)}))$
11: $(\boldsymbol{\theta}, \boldsymbol{\alpha}) \leftarrow (\boldsymbol{\theta}, \boldsymbol{\alpha}) - \eta \cdot \nabla_{(\boldsymbol{\theta},\boldsymbol{\alpha})} L_G$
12: **end while**

parameters $\boldsymbol{\theta}$ and observable parameters $\boldsymbol{\alpha}$ altogether (step 11), we have the choice of treating them separately. In particular, we show that by using semidefinite programming, when $\boldsymbol{w}$ and θ are fixed, we can always find analytically the optimal α that minimizes the loss. Also, we have computed that, instead of estimating the expectation values of the observables with separate state preparation and measurements, in some cases depending on the number and norm of the observables, we can apply the classical shadow method [11] to simultaneously estimate them to the same precision with fewer quantum resources.

(Work in Progress) We aim to address two main questions through the ongoing numerical experiments. First we investigate how the increase in expressivity from training observables is different to that from increasing circuit depth. Second, we implement the different variants of the above training algorithm over a range of datasets to evaluate their empirical performances.

[1] F. Arute, K. Arya, R. Babbush, D. Bacon, J. C. Bardin, R. Barends, R. Biswas, S. Boixo, F. G. S. L. Brandao, D. A. Buell, B. Burkett, Y. Chen, Z. Chen, B. Chiaro, R. Collins, W. Courtney, A. Dunsworth, E. Farhi, B. Foxen, A. Fowler, C. Gidney, M. Giustina, R. Graff, K. Guerin, S. Habegger, M. P. Harrigan, M. J. Hartmann, A. Ho, M. Hoffmann, T. Huang, T. S. Humble, S. V. Isakov, E. Jeffrey, Z. Jiang, D. Kafri, K. Kechedzhi, J. Kelly, P. V. Klimov, S. Knysh, A. Korotkov, F. Kostritsa, D. Landhuis, M. Lindmark, E. Lucero, D. Lyakh, S. Mandrà, J. R. McClean, M. McEwen, A. Megrant, X. Mi, K. Michielsen, M. Mohseni, J. Mutus, O. Naaman, M. Neeley, C. Neill, M. Y. Niu, E. Ostby, A. Petukhov, J. C. Platt, C. Quintana, E. G. Rieffel, P. Roushan, N. C. Rubin, D. Sank, K. J. Satzinger, V. Smelyanskiy, K. J. Sung, M. D. Trevithick, A. Vainsencher, B. Villalonga, T. White, Z. J. Yao, P. Yeh, A. Zalcman, H. Neven, and J. M. Martinis, enQuantum supremacy using a programmable superconducting processor, Nature **574**, 505 (2019).

[2] B. Coyle, D. Mills, V. Danos, and E. Kashefi, The Born Supremacy: Quantum Advantage and Training of an Ising Born Machine, npj Quantum Information **6**, 60 (2020), arXiv:1904.02214 [quant-ph].

[3] R. Sweke, J.-P. Seifert, D. Hangleiter, and J. Eisert, On the Quantum versus Classical Learnability of Discrete Distributions, Quantum **5**, 417 (2021), arXiv:2007.14451 [quant-ph].

[4] J. Romero and A. Aspuru-Guzik, enVariational Quantum Generators: Generative Adversarial Quantum Machine Learning for Continuous Distributions, Advanced Quantum Technologies **4**, 2000003 (2021).

[5] A. Barthe, M. Grossi, S. Vallecorsa, J. Tura, and V. Dunjko, Expressivity of parameterized quantum circuits for generative modeling of continuous multivariate distributions (2024), arXiv:2402.09848 [quant-ph].

[6] S. Lloyd and C. Weedbrook, Quantum generative adversarial learning, Physical Review Letters **121**, 040502 (2018), arXiv:1804.09139 [quant-ph].

[7] P.-L. Dallaire-Demers and N. Killoran, Quantum generative adversarial networks, Physical Review A **98**, 012324 (2018), arXiv:1804.08641 [quant-ph].

[8] M. Benedetti, D. Garcia-Pintos, O. Perdomo, V. Leyton-Ortega, Y. Nam, and A. Perdomo-Ortiz, A generative modeling approach for benchmarking and training shallow quantum circuits, npj Quantum Information **5**, 45 (2019), arXiv:1801.07686 [quant-ph].

[9] D. Aharonov, M. Ben-Or, E. Eban, and U. Mahadev, Interactive Proofs for Quantum Computations (2017), arXiv:1704.04487 [quant-ph].

[10] M. Arjovsky, S. Chintala, and L. Bottou, Wasserstein GAN (2017), arXiv:1701.07875 [cs, stat].

[11] H.-Y. Huang, R. Kueng, and J. Preskill, Predicting Many Properties of a Quantum System from Very Few Measurements, Nature Physics **16**, 1050 (2020), arXiv:2002.08953 [quant-ph].

[12] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin, and A. Courville, Improved Training of Wasserstein GANs (2017), arXiv:1704.00028 [cs, stat].