Exponentially Many Local Minima in Quantum Neural Networks

Xuchen You[1 2] **Xiaodi Wu**[1 2]

Abstract

Quantum Neural Networks (QNNs), or the so-called variational quantum circuits, are important quantum applications both because of their similar promises as classical neural networks and because of the feasibility of their implementation on near-term intermediate-size noisy quantum machines (NISQ). However, the training task of QNNs is challenging and much less understood. We conduct a quantitative investigation on the landscape of loss functions of QNNs and identify a class of simple yet extremely hard QNN instances for training. Specifically, we show for typical under-parameterized QNNs, there exists a dataset that induces a loss function with the number of spurious local minima depending exponentially on the number of parameters. Moreover, we show the optimality of our construction by providing an almost matching upper bound on such dependence. While local minima in classical neural networks are due to non-linear activations, in quantum neural networks local minima appear as a result of the quantum interference phenomenon. Finally, we empirically confirm that our constructions can indeed be hard instances in practice with typical gradient-based optimizers, which demonstrates the practical value of our findings.

1. Introduction

Motivations. With the recent establishment of quantum supremacy (Arute et al., 2019; Zhong et al., 2020), the research of quantum computing has entered a new stage where near-term Noisy Intermediate-Scale Quantum (NISQ) computers (Preskill, 2018) become the important platform for demonstrating quantum applications. *Quantum Neural Networks (QNNs)* (e.g., Farhi et al. (2020; 2014)), or the so-called variational quantum method (e.g., Peruzzo et al. (2014)), are the major candidates of applications that can be implemented on NISQ machines.

Typical QNNs replace classical neural networks (ClaNNs), which are just parameterized classical circuits, by quantum circuits with *classically parameterized unitary gates*. Instead of a classical mapping in ClaNNs from input to output, QNNs use a *quantum* one which could be very hard for classical computation to simulate (e.g., Harrow & Montanaro (2017)) and hence provide potential quantum speedups for machine learning tasks (e.g., see the survey by Biamonte et al. (2017) and by Harrow & Montanaro (2017) and examples in Schuld & Killoran (2019) and in Havlíček et al. (2019)). Moreover, given their quantum-mechanical nature, QNNs (or the variational quantum method) have also demonstrated huge promises in attacking problems in quantum chemistry and material science. Contrary to quantum supremacy tasks which serve only as a way to separate quantum and classical computational power but are not necessarily useful, Google has recently used the same machine to demonstrate the variational quantum method in calculating accurate electronic structures – an important task in quantum chemistry (Arute et al., 2020). Please see the survey (Benedetti et al., 2019) for more recent exciting developments of QNNs.

Similar to the classical case, the success of QNN applications will critically depend on the effectiveness of the training procedure which optimizes a *loss function* in terms of the *read-outs* and the *parameters* of QNNs for specific applications. The design of effective training methods has been under intense investigation both empirically and theoretically for ClaNNs. Moreover, understanding the landscape of the loss functions (e.g., Sagun et al. (2015); Choromanska et al. (2015b;a); Baity-Jesi et al. (2018)) and designing corresponding training/optimization methods have recently emerged as a principled approach to tackle this problem: (Auer et al., 1996; Safran & Shamir, 2018; Yun et al., 2018; Ding et al., 2019; Venturi et al., 2018) showed the existence of spurious local minima for ClaNNs; In turn, (Kawaguchi, 2016; Du & Lee, 2018; Soudry & Carmon, 2016; Nguyen & Hein, 2017; Li et al., 2018) characterized conditions for benign landscapes in terms of choice of activation, loss function and (over)-parameterization, providing insights on the design of ClaNNs and motivating explanations to the

[1] Joint Center for Quantum Information and Computer Science, University of Maryland [2] Department of Computer Science and Institute for Advanced Computer Studies, University of Maryland. Correspondence to: X.You <xyou@umd.edu>, X.Wu <xwu@cs.umd.edu>.

Proceedings of the 38^{th} *International Conference on Machine Learning*, PMLR 139, 2021. Copyright 2021 by the author(s).

success of gradient descent in training ClaNNs in certain scenarios (Jacot et al., 2018; Arora et al., 2019; Du et al., 2019); And training methods beyond simple variants of gradient descent have been devised for training with guarantees (Goel et al., 2017; Goel & Klivans, 2019; Zhong et al., 2017; Du & Goel, 2018).

Much less has been understood for QNNs. Most of the study of QNNs takes a trial-and-error approach by empirically comparing the performance of standard classical optimizers on training QNNs' loss functions (Benedetti et al., 2019). It has been observed empirically that training QNNs could be very challenging due to the *non-convex* nature of the corresponding loss functions (e.g., Wang et al. (2020; 2018)). However, these empirical studies are unfortunately restricted to small cases due to the limited access to quantum machines of reasonable sizes and the exponential cost in simulating them classically.

A theoretical study on the training of QNNs would be more *favorable* and *scalable* given the limit on empirical study. Indeed, a handful of such theoretical progress has been made. One prominent result is that random initialization of parameters will lead to vanishing gradients for much smaller size QNNs than ClaNNs (McClean et al., 2018) and hence pose one unique training difficulty for QNNs. Most of the remaining theoretical results are about special cases of QNNs such as *quantum approximate optimization algorithms* (QAOA) (e.g., Farhi et al. (2014); Farhi et al. (2019)) and extremely over-parameterized cases (e.g., Rabitz et al. (2004); Russell et al. (2016); Kiani et al. (2020)).

In this paper, we conduct a quantitative investigation on the landscape of loss functions for QNNs as a way to study their training issue. In particular, we are interested in understanding the properties of local minima of loss functions, such as, (1) the number of local minima depending on the architecture of QNNs, and (2) whether these local minima are *benign* or *spurious* ones, meaning that they are either close to the global minima or saddle points that can be escaped, or they are truly bad local minima that will hinder the training procedure. We are also motivated by the observation that QNNs share some similarity with linear neural networks without non-linear activation layers (Kawaguchi, 2016) or one-hidden layer neural networks with quadratic activation (Du & Lee, 2018) that are both known to have only benign local minima. The similarity is due to the fact that quantum mechanics underlying QNNs has a linear algebraic formulation similar to the linear part of ClaNNs. (Details in Section 2.) It is hence natural to wonder whether the local minima of QNNs could share these nice properties.

Contributions. Contrary to our original hope, we turn out to identify a class of *simple yet extremely hard* instances of QNNs for the training. Despite the similarity between QNNs and linear classical neural networks, we demonstrate that *spurious* (or *sub-optimal*) local minima do appear in QNNs and provide a quantitative characterization of the possible number of them. We focus on QNNs with the commonly used *square loss* function under a practical range of the number of parameters (or gates). Specifically, we identify a general condition of under-parameterized QNNs, which we refer to as QNNs *with linear independence*. We show for such QNNs, a dataset can be constructed such that the number of spurious local minima scales *exponentially* with the number of parameters. It demonstrates that QNNs behave quite differently from linear neural networks (e.g., Kawaguchi (2016)) but share the feature of neurons with *non-linear* activation functions (e.g., Auer et al. (1996)). This conceptual paradox could be explained by one central phenomenon of quantum mechanics behind QNNs called *interference*. We observe that interference replaces the role of non-linear activation in creating bad local minima for QNNs. (Section 3)

We investigate further and prove that typical under-parameterized QNNs are indeed *with linear independence*. This indicates that for almost all under-parameterized QNNs, there is a dataset where training with simple variants of gradient-based methods is hard. (Section 4)

Moreover, we show our construction is almost *optimal* in terms of the dependence of the number of local minima on the number of parameters, by developing an almost matching upper bound. This upper bound also demonstrates a sharp separation between QNNs and ClaNNs: For ClaNNs, provided an arbitrary number of training samples, the number of local minima could be unbounded, and hence won't be upper bounded by any function of the number of parameters (Auer et al., 1996). (Section 5)

Finally, we perform numerical experiments on concrete QNN instances with typical optimizers, and empirically confirm that our constructions can indeed be hard instances in practice. These experiments strengthen the value of our theoretical findings on the practical end. (Section 6)

It is worthwhile mentioning that our investigation on the landscape of loss functions has a direct implication on the hardness of gradient-based methods. While it does not rule out the possibility of efficient non-gradient-based training, there are no obvious solutions to the efficient training for our constructions. Identifying such training methods would be very interesting.

Related work. There are only a few previous studies on the training of QNNs, each of which has targeted at some specific parameter range for QNNs. The observation of vanishing gradients for random initialization of QNNs (McClean et al., 2018) provides hard QNN instances for training, which, however, still require many layers to demonstrate the difficulty of training in practice. Our constructions are based on a general condition which includes simple special cases like 1-layer QNNs that are already able to demonstrate QNNs' training difficulty.

Another line of work (Rabitz et al., 2004; Russell et al., 2016; Kiani et al., 2020) considers the extremely over-parameterized QNN cases. Specifically, when the number of parameters is comparable to the dimension of the underlying quantum system and the quantum *controllability* condition can be established, all local minima of QNNs' loss functions will become global (Rabitz et al., 2004; Russell et al., 2016). This theoretical prediction has also been observed empirically (Kiani et al., 2020). However, as the dimension of quantum systems grows *exponentially* with the number of qubits, this over-parameterized case can hardly be realistic for any QNN of reasonable size.

2. Preliminaries

Supervised learning. The goal of supervised learning is to identify a mapping from the feature space $\mathcal{X}$ to the label space $\mathcal{Y}$, given a training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^m \subset (\mathcal{X} \times \mathcal{Y})^m$ of m samples of *feature vectors* and *labels*. A common practice to find a mapping based on a training set is through empirical risk minimization (ERM), finding a mapping that best align with the training sample with respect to a specific loss function $l : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}$. Let $\hat{y}_i$ be the prediction of a certain mapping given $\mathbf{x}_i$. The goal of ERM is to find the mapping that minimizes the average loss $\frac{1}{m}\sum_{i=1}^m l(\hat{y}_i, y_i)$. Throughout this paper we will consider square loss $l(\hat{y}, y) = (\hat{y} - y)^2$.

Classical neural networks (ClaNNs). Neural networks are parameterized families of mappings, widely considered for practical problems. Typical feed-forward neural networks are parameterized by a sequence of matrices $\{\mathbf{W}_i\}_{i=1}^t$, such that $\mathbf{W}_i \in \mathbb{R}^{d_i \times d_{i-1}}$, with $d_t = 1$ and d_0 is the same as the dimension of the feature space $\mathcal{X}$. For feature vector $\mathbf{x}$, the output $\hat{y}$ of the neural network is

$$\hat{y} = \mathbf{W}_t \sigma(\mathbf{W}_{t-1} \sigma(\cdots \sigma(\mathbf{W}_1 \mathbf{x}) \cdots)), \tag{1}$$

where $\sigma(\cdot)$ denotes an element-wise activation on the output of each layer. (See Figure 1.) Linear neural networks (Kawaguchi, 2016) is one special example where σ is the identity mapping $\sigma(w) = w$: $\hat{y} = \mathbf{W}_t \mathbf{W}_{t-1} \cdots \mathbf{W}_1 \mathbf{x}$. Another example is one-hidden layer neural networks with quadratic activation $\sigma(w) = w^2$ (Du & Lee, 2018), where the output $\hat{y} = \mathbf{x}^T \mathbf{W}_1^T \mathbf{W}_1 \mathbf{x}$. Given the training set $\{(\mathbf{x}_i, y_i)\}_{i=1}^m$, the empirical risk minimization with square loss solves the optimization problem:

$$\min_{\mathbf{W}_1} \frac{1}{m} \sum_{i=1}^m \left(\mathbf{x}_i^T \mathbf{W}_1^T \mathbf{W}_1 \mathbf{x}_i - y_i\right)^2 \tag{2}$$

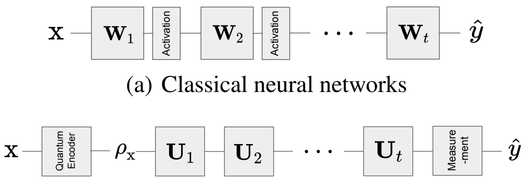
A common choice of $\sigma(\cdot)$ is non-linear activation such as Relu or Sigmoid. These activations introduce *non-linearity* which is the source of spurious local minima in neural networks (Kawaguchi, 2016; Auer et al., 1996).

Quantum neural networks. QNNs share the layered structure (Figure 1) where a linear transformation $\mathbf{U}_i$ is applied on the output of the previous layer, however, with the following differences:

(1) *Input*. The inputs to ClaNNs are feature vectors. Yet for QNNs, a feature vector $\mathbf{x}$ is first encoded into a quantum state $\boldsymbol{\rho}_{\mathbf{x}}$ then fed to the quantum circuits. We are not restricted to specific encoding scheme (e.g., (Mitarai et al., 2018; Benedetti et al., 2019; Lloyd et al., 2020)). For technical convenience, we will directly work with a set of m samples of *quantum encoding* and *labels* $\mathcal{S} = \{(\boldsymbol{\rho}_i, y_i)\}_{i=1}^m$ where $\boldsymbol{\rho}_i$ encodes the information of $\mathbf{x}_i$.

(2) *Linear Transformation & Parameterization*. The linear transformations $\{\mathbf{W}_i\}_{i=1}^t$ in ClaNNs could be general matrices, whereas the corresponding $\{\mathbf{U}_i\}_{i=1}^t$ in QNNs must be unitaries. Moreover, although $\{\mathbf{U}_i\}_{i=1}^t$ can be efficiently implemented by quantum machines, their classical representations are matrices of exponential dimension in terms of the system size (e.g., the number of qubits in QNNs). This makes classical simulation of QNNs extremely expensive and also makes the parameterizations of $\{\mathbf{U}_i\}_{i=1}^t$ different from the straightforward parameterizations of $\{\mathbf{W}_i\}_{i=1}^t$ (explained below).

(3) *Output*. Contrary to ClaNNs, one needs to make a quantum *measurement* to read information from QNNs (explained below). While there exist more advanced models of QNNs with additional nonlinearity, we consider the most basic QNNs, where the measurements are the only source of slight non-linearity allowed by quantum mechanics, which as we will see won't necessarily create bad local minima for the training. Note further there is no direct counterpart of classical non-linear activation $\sigma(\cdot)$ in QNNs of our consideration.



(a) Classical neural networks

(b) Quantum neural networks

Figure 1. An illustration of layer-structured classical and quantum neural networks.

Mathematical formulation of quantum states. A general quantum state with dimension d can be represented by a *density operator* that is a positive semidefinite (PSD) Hermitian matrix $\boldsymbol{\rho} \in \mathbb{C}^{d \times d}$ with $\mathrm{tr}(\boldsymbol{\rho}) = 1$. A quantum state $\boldsymbol{\rho}$ is *pure* if $\boldsymbol{\rho} = \mathbf{v}\mathbf{v}^\dagger$ for a ℓ_2 unit vector $\mathbf{v}$. A two-dimensional quantum state $\boldsymbol{\rho} \in \mathbb{C}^{2 \times 2}$ is usually referred as a *qubit*, the quantum generalization of the classical binary bit. The state of n qubits lies in $\otimes_{i=1}^{n}\mathbb{C}^{2\times 2}$ following the tensor product of spaces for single qubits, and is a linear operator on a Hilbert space with dimension $d = 2^n$, i.e., scales *exponentially* with the number of qubits n.

Parameterization of quantum transformations. Instead of directly parameterized matrices W_i, QNNs typically consist of *classically parameterized* quantum gates. A general form of these gates is $\exp(-i\theta\mathbf{H})$, where θ is the parameter, $\mathbf{H}$ the Hamiltonian (i.e., a Hermitian matrix) , and the exponential is a *matrix* exponential. For example, a commonly used gate set, called the Pauli rotation gate (e.g., Farhi et al. (2020); Li et al. (2017); Ostaszewski et al. (2019)), can be expressed as $\exp(-i\theta\mathbf{P}_c)$ (on c-th qubit) or $\exp(-i\theta\mathbf{P}_c \otimes \mathbf{P}_{c'})$ (on c-th and c'-th qubits), where $\mathbf{P}_c$ refers to Pauli $\mathbf{X}, \mathbf{Y}, \mathbf{Z}$ matrices.[1] We can also group gates in QNNs with respect to the layer structure in Figure 1 by putting gates that can be executed in parallel in the same layer. For example, let $\mathbf{V}_{i,j}(\theta_{i,j}) = \exp(-i\theta_{i,j}\mathbf{H}_{i,j})$ be the jth gate in the ith layer. Then $\mathbf{U}_i(\boldsymbol{\theta}) = \prod_j \mathbf{V}_{i,j}(\theta_{i,j})$ and

$$\mathbf{U}(\boldsymbol{\theta}) = \mathbf{U}_t(\boldsymbol{\theta})\mathbf{U}_{t-1}(\boldsymbol{\theta})\cdots\mathbf{U}_1(\boldsymbol{\theta}), \tag{3}$$

where $\mathbf{U}(\boldsymbol{\theta})$ refers to the unitary transformation of the entire QNN with parameters $\boldsymbol{\theta}$. For technical convenience and to highlight the dependence on the number of parameters p, we can also write

$$\mathbf{U}(\boldsymbol{\theta}) = \mathbf{V}_p(\theta_p)\mathbf{V}_{p-1}(\theta_{p-1})\cdots\mathbf{V}_1(\theta_1), \tag{4}$$

with $\mathbf{V}_l(\theta_l) = e^{-i\theta_l\mathbf{H}_l}$ for Hamiltonian $\mathbf{H}_l$ and $l \in [p]$.

Quantum measurements and observables. Quantum *observables*, mathematically formulated as Hermitian matrices $\mathbf{M} \in \mathbb{C}^{d\times d}$, are used in quantum mechanics to encode the information of the classical random outcomes generated by quantum *measurements* on quantum states. The expected outcome $\hat{y}$ of observable M on the output state $\mathbf{U}(\boldsymbol{\theta})\boldsymbol{\rho}\mathbf{U}^\dagger(\boldsymbol{\theta})$ of any QNN $\mathbf{U}(\theta)$ is given by

$$\begin{aligned}\hat{y} =& f(\boldsymbol{\rho}, \boldsymbol{\theta}) = \mathrm{tr}(\mathbf{U}(\boldsymbol{\theta})\boldsymbol{\rho}\mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}) \\ \text{or} \quad & \mathrm{tr}(\mathbf{v}^\dagger\mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}\mathbf{U}(\boldsymbol{\theta})\mathbf{v}) \text{ when } \boldsymbol{\rho} = \mathbf{v}\mathbf{v}^\dagger.\end{aligned}$$

A more complete introduction to quantum mechanics and QNNs can be found in S.M. Sect. A.

Given a quantum training set $\mathcal{S} = \{(\boldsymbol{\rho}_i, y_i)\}_{i=1}^m$ and a QNN with output $\hat{y} = f(\boldsymbol{\rho}, \boldsymbol{\theta})$, the empirical risk minimization with square loss optimizes the following loss function:

$$L(\boldsymbol{\theta}; \mathcal{S}) = \frac{1}{m}\sum_{i=1}^{m}\Big(\mathrm{tr}(\mathbf{U}(\boldsymbol{\theta})\boldsymbol{\rho}_i\mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}) - y_i\Big)^2. \tag{5}$$

When quantum encoding states are pure, namely $\boldsymbol{\rho}_i = \mathbf{v}_i\mathbf{v}_i^\dagger$ for $i \in [m]$, the loss function becomes

$$L(\boldsymbol{\theta}; \mathcal{S}) = \frac{1}{m}\sum_{j=1}^{m}\Big(\mathbf{v}_j^\dagger\mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}\mathbf{U}(\boldsymbol{\theta})\mathbf{v}_j - y_j\Big)^2 \tag{6}$$

which resembles Eqn.(2) from one-hidden layer neural networks with quadratic activation except for unitary transformations. It is known in (Du & Lee, 2018) that such neural networks do not possess spurious local minima almost certainly, whereas we establish a completely different behavior for QNNs.

Characterization of the landscape. For a differentiable function F defined on an unconstrained domain, $\boldsymbol{\theta}^*$ is a *critical* point if and only if the gradient vanishes at the point: $\nabla F(\boldsymbol{\theta}^*) = \mathbf{0}$. $\boldsymbol{\theta}$ is a local minimum if and only if there is an open set U containing $\boldsymbol{\theta}^*$ such that $F(\boldsymbol{\theta}^*) \leq F(\boldsymbol{\theta})$ for all $\boldsymbol{\theta} \in U$. A local minimum is global if the minimum value of F is attained at θ^*. For twice-differentiable function over an unconstrained domain, $\boldsymbol{\theta}^*$ is a local minimum if the Hessian is positive definite at $\boldsymbol{\theta}^*$ (sufficient condition) and only if $\boldsymbol{\theta}^*$ is a critical point (necessary condition).

Note further that the form of quantum gates $\exp(-i\theta\mathbf{H})$ will be *periodic* in θ for $\mathbf{H}$ with rational eigenvalues, which is typically true for commonly used $\mathbf{H}$ (e.g., Pauli matrices). It hence suffices to study the number of (spurious) local minima of the loss function within one period.

3. Exponentially Many Spurious Local Minima for Under-parameterized QNNs

In this section, we present our main result on the constructions of datasets for p-parameter quantum neural network instances with $\Omega(2^p)$ spurious local minima. We consider QNNs defined in Eqn. (4), with parameterized gates $\mathbf{V}_l(\theta_l)$ generated by $\mathbf{H}_l$ with eigenvalues ± 1. This is the case for single-qubit parameterized gates and two-qubit gates generated by Kronecker products of Pauli matrices.

Shifting $\mathbf{H}_l$ by $\lambda\mathbf{I}$ for any $\lambda \in \mathbb{R}$ introduces a global phase factor to the output state and does not change the output $f(\boldsymbol{\rho}, \boldsymbol{\theta})$. Also, shifting the observable $\mathbf{M}$ by $\lambda\mathbf{I}$ is equivalent to shifting the labels in the dataset by $-\lambda$. Without loss of generality, we assume $\mathrm{tr}(\mathbf{H}_l) = 0$ and $\mathrm{tr}(\mathbf{M}) = 0$.

We start by characterizing the output $f(\boldsymbol{\rho}, \boldsymbol{\theta})$. For any $l \in$

[1] $\mathbf{X} = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$, $\mathbf{Y} = \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$, $\mathbf{Z} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$. For Pauli matrix $\mathbf{Z}$, $\exp(-i\theta\mathbf{Z}) = \begin{bmatrix} e^{-i\theta} & 0 \\ 0 & e^{i\theta} \end{bmatrix}$.

$[p]$, define linear maps $\Phi_l^{(0)}(\cdot)$, $\Phi_l^{(1)}(\cdot)$ and $\Phi_l^{(2)}(\cdot)$ such that

$$\Phi_l^{(0)}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} + \mathbf{H}_l\mathbf{A}\mathbf{H}_l) \tag{7}$$

$$\Phi_l^{(1)}(\mathbf{A}) = \frac{1}{2}(\mathbf{A} - \mathbf{H}_l\mathbf{A}\mathbf{H}_l) \tag{8}$$

$$\Phi_l^{(2)}(\mathbf{A}) = \frac{i}{2}[\mathbf{H}_l, \mathbf{A}] \tag{9}$$

Here $[\cdot,\cdot]$ is the commutator of two matrices. For any Hermitian $\mathbf{A}$, $\Phi_l^{(0)}(\mathbf{A})$ commutes with $\mathbf{H}_l$, and the output of $\Phi_l^{(1)}$ and $\Phi_l^{(2)}$ anti-commute with $\mathbf{H}_l$. For any vector $\boldsymbol{\xi} \in \{0,1,2\}^p$, define:

$$\Phi_{\boldsymbol{\xi}}(\mathbf{A}) = \Phi_1^{(\xi_1)} \circ \Phi_2^{(\xi_2)} \circ \cdots \circ \Phi_p^{(\xi_p)}(\mathbf{A}) \tag{10}$$

with $\circ$ denoting the composition of mappings.

The observable in Heisenberg picture $\mathbf{M}(\boldsymbol{\theta}) := \mathbf{U}^\dagger(\boldsymbol{\theta})\mathbf{M}\mathbf{U}(\boldsymbol{\theta})$ can be expanded as:

$$\sum_{\boldsymbol{\xi}\in\{0,1,2\}^p} \Phi_{\boldsymbol{\xi}}(\mathbf{M}) \prod_{l:\xi_l=1} \cos 2\theta_l \prod_{l':\xi_{l'}=2} \sin 2\theta_{l'} \tag{11}$$

The QNN output $f(\boldsymbol{\rho},\boldsymbol{\theta}) = \mathrm{tr}(\boldsymbol{\rho}\mathbf{M}(\boldsymbol{\theta}))$ can be expressed as the following trigonometric polynomial:

$$\sum_{\boldsymbol{\xi}\in\{0,1,2\}^p} \mathrm{tr}(\boldsymbol{\rho}\Phi_{\boldsymbol{\xi}}(\mathbf{M})) \prod_{l:\xi_l=1} \cos 2\theta_l \prod_{l':\xi_{l'}=2} \sin 2\theta_{l'} \tag{12}$$

As shown in S.M. Sect. B, the loss function remains invariant under the joint transformation $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ and

$$\Phi_l^{(0)}(\cdot) \mapsto \mathbf{H}_l\Phi_l^{(0)}(\cdot)\mathbf{H}_l = \Phi_l^{(0)}(\cdot) \tag{13}$$

$$\Phi_l^{(1)}(\cdot) \mapsto \mathbf{H}_l\Phi_l^{(1)}(\cdot)\mathbf{H}_l = -\Phi_l^{(1)}(\cdot) \tag{14}$$

$$\Phi_l^{(2)}(\cdot) \mapsto \mathbf{H}_l\Phi_l^{(2)}(\cdot)\mathbf{H}_l = -\Phi_l^{(2)}(\cdot) \tag{15}$$

Under the transformation $\theta_l \mapsto \theta_l + \frac{\pi}{2}$, terms in Eqn. (12) associated with $\boldsymbol{\xi} : \xi_l = 0$ are invariant, while terms associated with $\boldsymbol{\xi} : \xi_l = 1, 2$ flip signs.

From an alternative perspective, $L(\boldsymbol{\theta};\mathcal{S})$ contains *oscillating wave* components proportional to $\cos 4\theta_l$, $\sin 4\theta_l$, $\cos 2\theta_l$ and $\sin 2\theta_l$, hence periodic with π on each coordinate. However, due the existence of lower frequency, the periodicity with $\frac{\pi}{2}$ does not always hold for all datasets. Our construction utilizes the presence and absence of this $\frac{\pi}{2}$-translational symmetry.

We will focus on a general class of QNN, which we call QNN *with linear independence*:

Definition 1 (QNN with linear independence)**.** A QNN is said to be with linear independence, if the associated set of $3^p - 1$ operators $\{\Phi_{\boldsymbol{\xi}}(\mathbf{M})\}_{\boldsymbol{\xi}\in\{0,1,2\}^p, \boldsymbol{\xi}\neq\mathbf{0}}$ forms a linearly independent set.

Note that for the linear independence condition to hold, the dimension of the QNN $d \geq 3^{p/2}$. Namely, it is a under-parameterized case, which differentiates us from the over-parameterized ones (Rabitz et al., 2004; Russell et al., 2016; Kiani et al., 2020). Our main result states:

Theorem 2 (Construction: exponentially many local minima)**.** *Consider QNNs composed of unitaries generated by two-level Hamiltonians, parameterized by $\boldsymbol{\theta} \in \mathbb{R}^p$. If the QNN is with linear independence, a dataset $\mathcal{S}$ can be constructed to induce a loss function $L(\boldsymbol{\theta};\mathcal{S})$ with 2^p local minima within each period, and $2^p - 1$ of these minima are spurious with positive suboptimality gap.*

Proof of Theorem 2. The dataset we construct is composed of two parts $\mathcal{S}_0$ and $\mathcal{S}_1$. The first component of the loss function $L(\boldsymbol{\theta};\mathcal{S}_0)$ is constructed with 2^p local minima using the $\frac{\pi}{2}$-translational symmetry:

Lemma 3 (Creating symmetry)**.** *For QNNs with linear independence as mentioned in Theorem 2, a dataset $\mathcal{S}_0$ can be constructed to induce a loss function $L(\boldsymbol{\theta};\mathcal{S}_0)$ that (1) has a local minimum at some $\boldsymbol{\theta}^\star$, and (2) is invariant under translation $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ for all $l \in [p]$.*

Due to the translational invariance, for any $\boldsymbol{\zeta} \in \{0,1\}^p$, $\boldsymbol{\theta}^\star + \frac{\pi}{2}\boldsymbol{\zeta}$ is a local minimum for $L(\boldsymbol{\theta};\mathcal{S}_0)$, forming a total of 2^p local minima. A second dataset $\mathcal{S}_1$ is introduced to break this symmetry, creating spurious local minima:

Lemma 4 (Breaking symmetry)**.** *Consider the QNN, dataset $\mathcal{S}_0$ and local minimum $\boldsymbol{\theta}^\star$ defined in Lemma 3. Let Θ denote the set of 2^p local minima due to the translational invariance. There exists a dataset $\mathcal{S}_1$ such that*

$$\inf_{\boldsymbol{\theta}\in\mathcal{N}(\boldsymbol{\theta}^\star)} L(\boldsymbol{\theta};\mathcal{S}_0) + L(\boldsymbol{\theta};\mathcal{S}_1) < \inf_{\boldsymbol{\theta}\in\mathcal{N}(\boldsymbol{\theta}')} L(\boldsymbol{\theta};\mathcal{S}_0) + L(\boldsymbol{\theta};\mathcal{S}_1) \tag{16}$$

for all $\boldsymbol{\theta}' \in \Theta/\{\boldsymbol{\theta}^\star\}$, and that

$$L(\boldsymbol{\theta};\mathcal{S}_0) + L(\boldsymbol{\theta};\mathcal{S}_1) > L(\boldsymbol{\theta}';\mathcal{S}_0) + L(\boldsymbol{\theta}';\mathcal{S}_1) \tag{17}$$

for all $\boldsymbol{\theta}' \in \Theta$ and all $\boldsymbol{\theta} \in \partial\mathcal{N}(\boldsymbol{\theta}')$. Here $\mathcal{N}(\cdot)$ denote a bounded and closed neighbourhood, such that $\mathcal{N}(\boldsymbol{\theta}) \cap \mathcal{N}(\boldsymbol{\theta}') = \emptyset$ for any $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$. And let $\partial\mathcal{N}$ denote its boundary.

Eqn. (17) in Lemma 4 ensures the existence of a local minimum within $\mathcal{N}(\boldsymbol{\theta})$ for each $\boldsymbol{\theta} \in \Theta$, and Eqn. (16) promises that only the local minimum within $\mathcal{N}(\boldsymbol{\theta}^\star)$ achieves the global optimal value. Combining $\mathcal{S}_0$ and $\mathcal{S}_1$ finishes the proof for Theorem 2. □

We give proof sketches for Lemma 3 and 4. The full proofs are postponed to S.M. Sect. B.

Proof sketch for Lemma 3. It suffices to construct a dataset $\mathcal{S}_0 = \{(\boldsymbol{\rho}_k, y_k)\}_{k=1}^{m_0}$, such that (1) for all $k \in [p]$, $f_k(\boldsymbol{\theta}) := \langle \boldsymbol{\rho}_k, \mathbf{M}(\boldsymbol{\theta}) \rangle - y_k$ is either symmetric or anti-symmetric under $\theta_l \mapsto \theta_l + \frac{\pi}{2}$ for all $l \in [p]$, and (2) the intersection Θ of the set of roots Θ_k of $f_k(\boldsymbol{\theta}) = 0$ is non-empty and contains at least one isolated point $\boldsymbol{\theta}^\star$. For such $\mathcal{S}_0$, $\boldsymbol{\theta}^\star$ is an isolated root of the non-negative loss function $L(\boldsymbol{\theta}; \mathcal{S}_0) = \sum_{k=1}^{m_0} f_k(\boldsymbol{\theta})^2$.

The existence of such dataset $\mathcal{S}_0$ follows from the linear independence of operators for the QNN. As a result, for any $k \in [m_0]$, the solution to the following linear system for Hermitian $\mathbf{D}_k \in \mathbb{C}^{d \times d}$ is non-empty:

$$\begin{cases} \operatorname{tr}(\mathbf{D}_k \cdot \mathbf{I}) = 0, \\ \operatorname{tr}(\mathbf{D}_k \cdot \Phi_{\boldsymbol{\xi}}(\mathbf{M})) = \hat{f}_{\boldsymbol{\xi},k}, \ \forall \boldsymbol{\xi} \neq \mathbf{0}. \end{cases} \tag{18}$$

Here $\hat{f}_{\boldsymbol{\xi},k}$ is the coefficient corresponding to the term $\prod_{l:\xi_l=1} \cos 2\theta_l \prod_{l':\xi_{l'}=2} \sin 2\theta_{l'}$ in $f_k(\boldsymbol{\theta})$. Given the solution $\{\mathbf{D}_k\}_{k=1}^{m_0}$, $\mathcal{S}_0$ can be constructed by setting $\boldsymbol{\rho}_k := \frac{1}{d}\mathbf{I} + \kappa \mathbf{D}_k$ for a proper scaling factor κ and let $y_k = \operatorname{tr}(\boldsymbol{\rho}_k \Phi_{\mathbf{0}}(\mathbf{M}))$. □

Proof sketch for Lemma 4. Rewrite the loss function as

$$\begin{aligned} L(\boldsymbol{\theta}; \mathcal{S}_1) = &- \frac{2}{m_1} \sum_{k=1}^{m_1} y_k \operatorname{tr}(\boldsymbol{\rho}_k \mathbf{M}(\boldsymbol{\theta})) && (19) \\ &+ \frac{1}{m_1} \sum_{k=1}^{m_1} \operatorname{tr}(\boldsymbol{\rho}_k \mathbf{M}(\boldsymbol{\theta}))^2 + \frac{1}{m_1} \sum_{k=1}^{m_1} y_k^2 && (20) \end{aligned}$$

As will be made clear in S.M. Sect. B, our key observation is that, under a joint scaling of y_k and $\boldsymbol{\rho}_k$, the second term can be arbitrarily suppressed while the first term remains the same. Therefore it suffices to study the first term $L'(\boldsymbol{\theta}; \mathcal{S}_1) := -\frac{2}{m_1} \sum_{k=1}^{m_1} y_k \operatorname{tr}(\boldsymbol{\rho}_k \mathbf{M}(\boldsymbol{\theta}))$. The linear independence allows us to solve a linear system to construct $\mathcal{S}_1$ that satisfies the requirements in Lemma 4. □

Remarks. The statements above involve unitaries generated by two-level Hamiltonians only. For more general local quantum gates, $\{\mathbf{H}_l\}_{l=1}^p$ are allowed to have more than two distinct eigenvalues. We are especially interested in Hamiltonians with eigenvalues $\{E_1, \cdots, E_d\} \subset \mathbb{Z}$, as arbitrary Hamiltonians with rational spectrum can be converted to ones with integral spectrum with proper shifting and scaling. Theorem 2 can be generalized for $\mathbf{H}_l$'s with largest eigengap $\max_{c,c' \in [d]} |E_c - E_{c'}|$ bounded by Δ, with the number of spurious local minima being $\Omega(\Delta^p)$. This observation further supports the intuition of interference as the source of local minima.

1-layer QNN. A simple example of QNNs with linear independence is a one-layer circuit with local $\mathbf{H}_l$ acting on the l-th qubit, and a product operator $\mathbf{M}$ as the observable:

Proposition 5 (One-layer QNNs with product observables)**.** *Consider the family of QNNs composed of unitaries generated by two-level Hamiltonians, parameterized by $\boldsymbol{\theta} \in \mathbb{R}^p$. For all $l \in [p]$, let $\mathbf{H}_l$ be a local Hamiltonian on the l-qubit, taking the form $\mathbf{I} \otimes \cdots \otimes \mathbf{h}_l \otimes \cdots \otimes \mathbf{I}$ for some Hermitian $\mathbf{h}_l$ at the l-th position, and $\mathbf{M} = \mathbf{m}_1 \otimes \cdots \otimes \mathbf{m}_p$ such that $\mathbf{m}_l + \mathbf{h}_l \mathbf{m}_l \mathbf{h}_l$ and $\mathbf{m}_l - \mathbf{h}_l \mathbf{m}_l \mathbf{h}_l$ are non-zero for any l. There exists a dataset that induces a loss function with $2^p - 1$ spurious local minima.*

This follows from the fact that $\operatorname{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M}) \Phi_{\boldsymbol{\xi}'}(\mathbf{M})) = 0$ if and only if $\boldsymbol{\xi} \neq \boldsymbol{\xi}'$. In S.M. Sect. B, we provide proof for Proposition 5 and several concrete example QNNs to demonstrate that our construction can have local minima at arbitrary $\boldsymbol{\theta}$, and does not allow trivial solutions such as coordinate-wise greedy optimization.

4. Typical QNNs are with Linear Independence

In Section 3, we provided a general condition (Definition 1) for QNNs to have exponentially many bad local minima for some datasets. In this section, we show that this condition is met for typical under-parameterized QNNs. To see this, we consider the following measure over instances of QNNs: Let $\mathbf{H}$ be a d-dimensional Hermitian such that $\operatorname{tr}(\mathbf{H}) = 0$ and $\mathbf{H}^2 = \mathbf{I}$. A random circuit $\mathbf{U}(\boldsymbol{\theta})$ is specified as

$$\mathbf{U}(\boldsymbol{\theta}) = e^{-i\theta_p \mathbf{W}_p \mathbf{H} \mathbf{W}_p^\dagger} \cdots e^{-i\theta_1 \mathbf{W}_1 \mathbf{H} \mathbf{W}_1^\dagger} \tag{21}$$

with $\{\mathbf{W}_l\}_{l=1}^p$ independently sampled with respect to the Haar measure on the d-dimensional unitary group $U(d)$.

Up to a unitary transformation, this random model is equivalent to a circuit with p interleaving parameterized gate $\{e^{-i\theta_l \mathbf{H}}\}_{l=1}^p$ and unitary $\{\tilde{\mathbf{W}}_l\}_{l=1}^p$ randomly sampled with respect to the Haar measure:

$$\mathbf{U}(\boldsymbol{\theta}) = \tilde{\mathbf{W}}_p e^{-i\theta_p \mathbf{H}} \tilde{\mathbf{W}}_{p-1} \cdots \tilde{\mathbf{W}}_1 e^{-i\theta_1 \mathbf{H}} \tag{22}$$

The equivalence is due to the left (or right) invariance of the Haar measure. This interleaving nature of fixed and parameterized gates are shared by existing designs of QNNs, and any p-parameter QNN generated by two-level Hamiltonians can be expressed in Eqn. (22). Moreover, applying polynomially many random 2-qubit gates on random pairs of qubits generates a distribution over gates that approximates the Haar measure up to the 4-th moments (Brandao et al., 2016), which is what we require in this section.

The Gram matrix for the set $\{\Phi_{\boldsymbol{\xi}}(\mathbf{M})\}_{\boldsymbol{\xi} \in \{0,1,2\}^p, \boldsymbol{\xi} \neq \mathbf{0}}$ is defined such that the element corresponding to the pair $(\boldsymbol{\xi}, \boldsymbol{\xi}')$ is $\operatorname{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M}) \Phi_{\boldsymbol{\xi}'}(\mathbf{M}))$. The Gram matrix is always positive semidefinite, and a positive definite Gram matrix implies the linear independence of the set.

Using the integral formula with respect to Haar measure on unitary groups (Puchała & Miszczak, 2011), we can estimate the expectations and variances of the diagonal and off-diagonal terms, and upper bound the probability of the event:

$$\exists \boldsymbol{\xi} : \operatorname{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})^2) \leq \sum_{\boldsymbol{\xi}' \neq \boldsymbol{\xi}} |\operatorname{tr}(\Phi_{\boldsymbol{\xi}}(\mathbf{M})\Phi_{\boldsymbol{\xi}'}(\mathbf{M}))| \tag{23}$$

Applying the Gershgorin circle theorem (Golub & Van Loan, 1996), we can lower bound the probability for a random QNN to have linear independent terms:

Theorem 6 (Typical under-parameterized QNNs are with linear independence)**.** *Consider a random p-parameter d-dimensional QNN with two-level Hamiltonians sampled from the model specified in Eqn. (21). Let the observable $\mathbf{M}$ be an arbitrary non-zero trace-0 Hermitian. Such QNN is with linear independence with probability $\geq 1 - O(d^{-1})$ for fixed p, and with probability $\geq 1 - O(e^{-p})$ for dimension $d : \log(d) = \Theta(p)$.*

Please refer to S.M. Sect. C for the full proof.

5. Upper Bound on the Number of Local Minima

Our construction above possesses 2^p local minima for p parameters, whereas the classical work of Auer et al. (1996) demonstrates a construction for a single neuron with $\lfloor m/p \rfloor^p$ local minima for m training samples. Note that the latter could grow unboundedly with m. In this section, we show, however, this classical unbounded growth of local minima does not hold for QNNs. In fact, we could establish an almost matching upper bound for 2^p. All the formal proofs are deferred to S.M. Sect. D.

To that end, let us examine the *Fourier* expansion of the loss function $L(\boldsymbol{\theta}, \mathcal{S})$ (Eqn. (5)). Let T_l be the period of $L(\boldsymbol{\theta}; \mathcal{S})$ corresponding to θ_l, and $\hat{L}(\mathbf{k})$ the Fourier coefficient for $\mathbf{k} = (k_1, \cdots, k_p)^T \in \mathbb{Z}^p$. We have

$$L(\boldsymbol{\theta}; \mathcal{S}) = \sum_{\mathbf{k} \in K} \hat{L}(\mathbf{k}) \prod_{l=1}^{p} \Big(\cos \frac{k_l \theta_l}{T_l} + i \sin \frac{k_l \theta_l}{T_l} \Big) \tag{24}$$

where $K \subseteq \mathbb{Z}^p$ is the support of the Fourier coefficients.

One critical observation is that, for arbitrary choice of two-level $\{\mathbf{H}_l\}_{l=1}^p$, observable $\mathbf{M}$ and training set $\mathcal{S}$, the support K of the Fourier spectrum is bounded in ℓ_1-norm: $\max_{\mathbf{k} \in K} \sum_{l=1}^p |k_l| \leq 2p$, indicating that the Fourier degree of $L(\boldsymbol{\theta}; \mathcal{S})$ is upper bounded by $2p$ (See S.M. Sect. D.1).

By definition, a local minimum must be a critical point, hence it suffices to bound the number of critical points for functions with Fourier spectrum supported on a ℓ_1-bounded set. Define $G_l(\boldsymbol{\theta})$ as $\frac{\partial}{\partial \theta_l} L(\boldsymbol{\theta}; \mathcal{S})$:

$$G_l(\boldsymbol{\theta}) = \sum_{\mathbf{k} \in K} k_l \hat{L}(\mathbf{k}) \big(- \sin \frac{k_l \theta_1}{T_l} + i \cos \frac{k_l \theta_l}{T_l} \big) \tag{25}$$

$$\cdot \prod_{l' \neq l} \big(\cos \frac{k_{l'} \theta_{l'}}{T_{l'}} + i \sin \frac{k_{l'} \theta_{l'}}{T_{l'}} \big) \tag{26}$$

Notice that the Fourier spectrum of G_l is supported on the same set K. A critical point of $L(\boldsymbol{\theta}; \mathcal{S})$ must satisfy that for all $l \in [p]$, $G_l(\boldsymbol{\theta}) = 0$. By basic trigonometry, $\cos k\theta$ can be expressed as a degree-k polynomial of $\cos\theta$ and $\sin k\theta$ as a degree-$(k-1)$ polynomial of $\cos\theta$ multiplied by $\sin\theta$. Consider the change of variable

$$c_l = \cos(\theta_l / T_l), \; s_l = \sin(\theta_l / T_l), \; \forall l \in [p]. \tag{27}$$

Let $g_l(c_1, s_1, \cdots, c_p, s_p)$ be the multivariate polynomial constraints corresponding to $G_l(\boldsymbol{\theta})$ after the change of variable. For each g_l, the sum of degrees of $c_{l'}$ and $s_{l'}$ is bounded by $\max_{\mathbf{k} \in K} |k_{l'}|$, and the degree $\mathsf{deg}(g_l)$ of g_l is bounded by $\max_{\mathbf{k} \in} \sum_{l=1}^p |k_l| \leq 2p$. The change of variable is one-to-one from $\theta_l \in [0, T_l)$ to a pair of $(c_l, s_l) \in \mathbb{R}^2$ under the constraint $c_l^2 + s_l^2 = 1$. Therefore, it suffices to count the number of roots of the polynomial system with $2p$ parameters and $2p$ constraints:

$$g_l(c_1, s_1, \cdots, c_p, s_p) = 0, \; c_l^2 + s_l^2 - 1 = 0 \tag{28}$$

for all $l \in [p]$. Notice that for a general polynomial system, the number of critical points can be unbounded. For example, consider a system composed of constant polynomials, every point in the domain is a critical point. This corresponds to the constant loss function, where the gradients vanish everywhere with positive semidefinite Hessians. For this reason, we will focus on the non-degenerated case with finitely many local minima. Under the premise of non-degeneracy, by Bézout's Theorem (e.g. Section 3.3 in Cox et al. (2006)), the number of roots can be bounded by the product of the degree of polynomial constraints $2^p \mathsf{deg}(g_1) \mathsf{deg}(g_2) \cdots \mathsf{deg}(g_p) \leq (4p)^p$. A formal statement of the above derivation is as follows:

Theorem 7 (Upper bound: the number of local minima)**.** *Consider non-degenerated QNNs composed of unitaries generated by two-level Hamiltonians $\{H_l\}_{l=1}^p$ with p parameters. For training set $\mathcal{S}$, within each period, the loss function $L(\boldsymbol{\theta}; \mathcal{S})$ possesses at most $(4p)^p$ local minima.*

We also prove a similar result for the more general case where the generators are Hamiltonians with integral spectrum: let Δ be the largest eigen-gap for each of the Hamiltonians, the number of local minima within each period is upper bounded by $O((\Delta p)^p)$. Please refer to S.M. Sect. D for details.

6. Experiments

We investigate the practical performance of the common optimizers on our construction in this section. It is well-known in the classical literature that the existence of spurious local minima does not necessarily cause difficulties in optimization (e.g., Ge & Ma (2017)). We show, however, our constructions can indeed be hard instances for training in practice.

To that end, we evaluate a specific construction from Proposition 5 in Section 3 by using the standard optimizers with randomly initialized parameters uniformly sampled from the domain[2], and visualize the distribution of function values at convergence. For p-parameter instances, our construction involves p-qubits. We choose $\mathbf{h}_1 = \cdots = \mathbf{h}_p = \mathbf{Z}$ and $\mathbf{m}_1 = \cdots = \mathbf{m}_p = \mathbf{Y}+\mathbf{I}$. The specific form of the instance and all the training details are provided in S.M. Sect. E.

Implementation The experiments are run on Intel Core i7-7700HQ Processor (2.80GHz) with 16G memory. We classically simulate the training with Pytorch (Paszke et al., 2019), using the analytical form of the objective function for the purpose of efficiency.

Optimizers The QNN instances are trained with three popular optimizers in classical optimization or machine learning: Adam(Kingma & Ba, 2015), RMSProp(Bengio, 2015), and L-BFGS(Liu & Nocedal, 1989). The first two methods (Kingma & Ba, 2015; Bengio, 2015) are variants of vanilla gradient descent with adaptive learning rate. and are widely used for training large-scale deep neural networks as well as for the quantum counterparts (Killoran et al., 2019; Mari et al., 2020; Lloyd et al., 2020; Ostaszewski et al., 2019; Sweke et al., 2020). The last method (Liu & Nocedal, 1989) is an efficient implementation of the approximate Newton method that utilizes the second-order information (i.e. the Hessian). For all instances and optimizers, we use the exact gradient induced by the dataset without stochasticity from the mini-batched gradient descent.

It turns out, for all the examined instances and all three optimizers, under random initialization, the optimizations converge to local minima with non-negligible suboptimality (i.e., different from the global one by a non-negligible amount) with high probability. In Figure 2, we train the 4-parameter construction with RMSProp and repeat for 100 times. Let $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}_f$ denote the parameters at initialization and at convergence. The function values at initialization $L(\boldsymbol{\theta}_i; \mathcal{S})$ are supported on a continuous spectrum as shown in gray. After training and converging with RMSProp, the function values $L(\boldsymbol{\theta}_f; \mathcal{S})$ fall into discretized values as shown in orange. The smallest training loss attainable in our construction is 0, therefore only the leftmost bar (to the left of the dotted **black** vertical line) corresponds to the global minimum. Namely, the success probability of converging to the global minimum is very small. A similar phenomenon persists for instances with more parameters and with different optimizers in Figure 3. As the number of bad local minima grows exponentially in our construction, the success probability should also in theory decay exponentially. This is empirically confirmed in Figure 4, where we illustrate the precise empirical success probability for all these tests. Moreover, as shown in S.M. Sect. E.3, the tendency of exponential decay remains unchanged in the presence of label noises, indicating the robustness of our constructions.

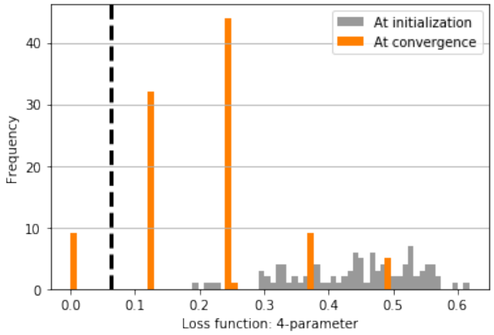


Figure 2. Loss functions at random initialization and at convergence for 4-parameter instances trained with RMSProp, repeated for 100 times. The function values are supported on a continuous spectrum at initialization as plotted in **gray** and converge to discretized values as plotted in **orange**.

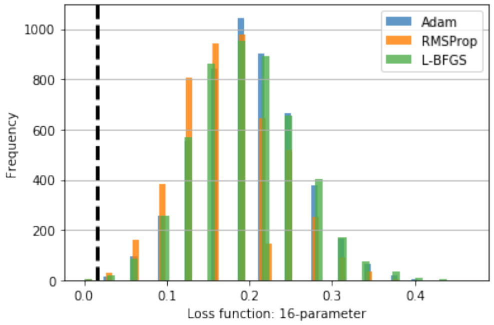


Figure 3. Distributions of loss functions at convergence for instances with 16 parameters trained with Adam, RMSProp and L-BFGS, repeated 5000 times with uniformly random initialization. All methods fail to converge to the global minimum 0.0 with high probability.

Beyond the constructed datasets To demonstrate the generality of our results, we repeat the experiments for datasets with more practical significance: for p-parameter instances, we choose the input state to be a p-qubit encoding of $\mathbf{x} \in [0, 2\pi)^{2p}$ via $\mathbf{X}$- and $\mathbf{Y}$-rotations on each of

[2] For p-parameter instances, we uniformly sample the initial parameters from $[0, 2\pi)^p$.

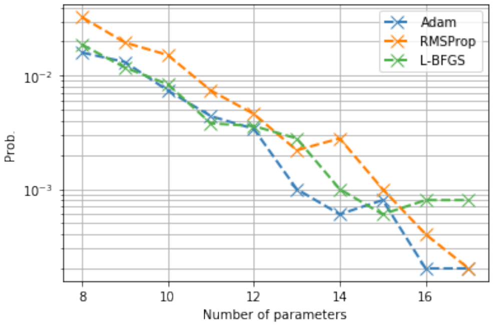


Figure 4. The decay of success rate for finding the global minimum under random initialization with Adam, RMSProp, L-BFGS. For each data point, we repeat the experiments for 5000 times.

the qubits. The associated label is either 1 or 0, depending on the sign of $\mathbf{w}^T\mathbf{x}$, with $\mathbf{w}$ being the normal vector of a hyperplane in $\mathbb{R}^{2p}$. These datasets have the interpretation as an encoding of a linearly separable classical concept. In Figure5, we plot the function values at convergence for an 8-parameter instance: no more than 4 of the 70 random initializations have reached the global minima. This is repeated for instances with $2, 4$ and 6 qubits. While we no longer have a clear exponential dependency in the success rate, the number of local minima increases significantly as the number of parameters increases (see S.M. Sect. E.4). This observation suggests that our theory and experiments on the constructed datasets can capture the practical difficulty in training under-parameterized QNNs with gradient-based methods.

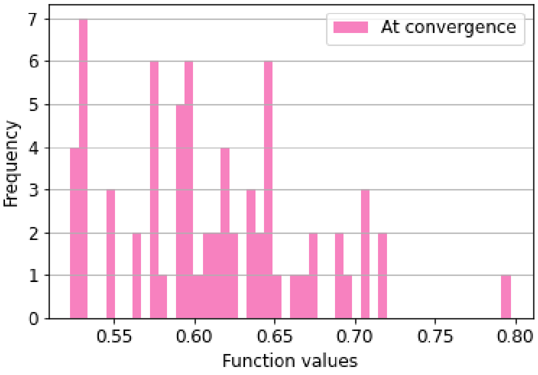


Figure 5. Function values at convergence for training an 8-parameter instance with RMSProp on the linearly-separable classical concept. No more than 4 among the 70 random initializations find the global minima, indicating the existence of many sub-optimal local minma.

7. Conclusion

In this work, we provide a characterization of the landscape for under-parameterized QNNs, by showing that in the worst-case, the number of local minima can increase exponentially with the number of parameters. Supported by numerical simulations, our result suggests when under-parameterized, QNNs may not be efficiently solved by gradient-based black-box methods.

This work leaves several open questions:

- Given the knowledge of the data distribution, can we design a QNN architecture with a benign landscape?
- We know that when sufficiently parameterized (e.g. (Russell et al., 2016)), the landscape for optimizing variational quantum ansatz can be benign. It is therefore natural to ask, fixing the system size, how does the landscape change as the number of parameters increases?
- Classically, despite the provable bad landscape of shallow neural networks(e.g. Safran & Shamir (2018)), Goel & Klivans (2019) came up with algorithms that can minimize the loss with guarantees. Can we design an algorithm (beyond gradient-based method) that can solve the optimization problem efficiently and provably?

Acknowledgements

We thank reviewers for useful comments. This work received support from the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Accelerated Research in Quantum Computing and Quantum Algorithms Team programs, as well as the U.S. National Science Foundation grant CCF-1755800, CCF-1816695, and CCF-1942837 (CAREER).

References

Arora, S., Du, S. S., Hu, W., Li, Z., Salakhutdinov, R., and Wang, R. On exact computation with an infinitely wide neural net. *arXiv preprint arXiv:1904.11955*, 2019.

Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., Biswas, R., Boixo, S., Brandao, F. G. S. L., Buell, D. A., Burkett, B., Chen, Y., Chen, Z., Chiaro, B., Collins, R., Courtney, W., Dunsworth, A., Farhi, E., Foxen, B., Fowler, A., Gidney, C., Giustina, M., Graff, R., Guerin, K., Habegger, S., Harrigan, M. P., Hartmann, M. J., Ho, A., Hoffmann, M., Huang, T., Humble, T. S., Isakov, S. V., Jeffrey, E., Jiang, Z., Kafri, D., Kechedzhi, K., Kelly, J., Klimov, P. V., Knysh, S., Korotkov, A., Kostritsa, F., Landhuis, D., Lindmark, M., Lucero, E., Lyakh, D., Mandrà, S., McClean, J. R., McEwen, M., Megrant, A., Mi, X., Michielsen, K., Mohseni, M., Mutus, J., Naaman, O., Neeley, M., Neill, C., Niu, M. Y., Ostby, E., Petukhov, A., Platt, J. C., Quintana, C., Rieffel, E. G., Roushan, P., Rubin, N. C., Sank, D., Satzinger, K. J., Smelyanskiy, V., Sung, K. J., Trevithick, M. D., Vainsencher, A., Villalonga, B., White, T., Yao, Z. J., Yeh, P., Zalcman, A., Neven, H., and Martinis, J. M. Quantum supremacy using a programmable superconducting processor. *Nature*, 574(7779):505–510, 2019.

Arute, F., Arya, K., Babbush, R., Bacon, D., Bardin, J. C., Barends, R., Boixo, S., Broughton, M., Buckley, B. B., Buell, D. A., Burkett, B., Bushnell, N., Chen, Y., Chen, Z., Chiaro, B., Collins, R., Courtney, W., Demura, S., Dunsworth, A., Farhi, E., Fowler, A., Foxen, B., Gidney, C., Giustina, M., Graff, R., Habegger, S., Harrigan, M. P., Ho, A., Hong, S., Huang, T., Huggins, W. J., Ioffe, L., Isakov, S. V., Jeffrey, E., Jiang, Z., Jones, C., Kafri, D., Kechedzhi, K., Kelly, J., Kim, S., Klimov, P. V., Korotkov, A., Kostritsa, F., Landhuis, D., Laptev, P., Lindmark, M., Lucero, E., Martin, O., Martinis, J. M., McClean, J. R., McEwen, M., Megrant, A., Mi, X., Mohseni, M., Mruczkiewicz, W., Mutus, J., Naaman, O., Neeley, M., Neill, C., Neven, H., Niu, M. Y., O'Brien, T. E., Ostby, E., Petukhov, A., Putterman, H., Quintana, C., Roushan, P., Rubin, N. C., Sank, D., Satzinger, K. J., Smelyanskiy, V., Strain, D., Sung, K. J., Szalay, M., Takeshita, T. Y., Vainsencher, A., White, T., Wiebe, N., Yao, Z. J., Yeh, P., and Zalcman, A. Hartree-fock on a superconducting qubit quantum computer. *Science*, 369(6507):1084–1089, 2020. ISSN 0036-8075. doi: 10.1126/science.abb9811. URL https://science.sciencemag.org/content/369/6507/1084.

Auer, P., Herbster, M., and Warmuth, M. K. Exponentially many local minima for single neurons. In *Advances in neural information processing systems*, pp. 316–322, 1996.

Baity-Jesi, M., Sagun, L., Geiger, M., Spigler, S., Arous, G. B., Cammarota, C., LeCun, Y., Wyart, M., and Biroli, G. Comparing dynamics: Deep neural networks versus glassy systems. In *International Conference on Machine Learning*, pp. 314–323. PMLR, 2018.

Benedetti, M., Lloyd, E., Sack, S., and Fiorentini, M. Parameterized quantum circuits as machine learning models. *Quantum Science and Technology*, 4(4):043001, 2019.

Bengio, Y. Rmsprop and equilibrated adaptive learning rates for nonconvex optimization. *corr abs/1502.04390*, 2015.

Biamonte, J., Wittek, P., Pancotti, N., Rebentrost, P., Wiebe, N., and Lloyd, S. Quantum machine learning. *Nature*, 549(7671):195, 2017.

Brandao, F. G., Harrow, A. W., and Horodecki, M. Local random quantum circuits are approximate polynomial-designs. *Communications in Mathematical Physics*, 346 (2):397–434, 2016.

Choromanska, A., Henaff, M., Mathieu, M., Arous, G. B., and LeCun, Y. The loss surfaces of multilayer networks. In *Artificial intelligence and statistics*, pp. 192–204. PMLR, 2015a.

Choromanska, A., LeCun, Y., and Arous, G. B. Open problem: The landscape of the loss surfaces of multilayer networks. In *Conference on Learning Theory*, pp. 1756–1760. PMLR, 2015b.

Collins, B. and Śniady, P. Integration with respect to the haar measure on unitary, orthogonal and symplectic group. *Communications in Mathematical Physics*, 264(3):773–795, 2006.

Cox, D. A., Little, J., and O'shea, D. *Using algebraic geometry*, volume 185. Springer Science & Business Media, 2006.

Ding, T., Li, D., and Sun, R. Sub-optimal local minima exist for almost all over-parameterized neural networks. *arXiv preprint arXiv:1911.01413*, 2019.

Du, S. and Lee, J. On the power of over-parametrization in neural networks with quadratic activation. In *International Conference on Machine Learning*, pp. 1329–1338. PMLR, 2018.

Du, S., Lee, J., Li, H., Wang, L., and Zhai, X. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pp. 1675–1685. PMLR, 2019.

Du, S. S. and Goel, S. Improved learning of one-hidden-layer convolutional neural networks with overlaps. *arXiv preprint arXiv:1805.07798*, 2018.

Farhi, E., Goldstone, J., and Gutmann, S. A quantum approximate optimization algorithm. *arXiv preprint arXiv:1411.4028*, 2014.

Farhi, E., Goldstone, J., Gutmann, S., and Zhou, L. The Quantum Approximate Optimization Algorithm and the Sherrington-Kirkpatrick Model at Infinite Size. *arXiv e-prints*, art. arXiv:1910.08187, October 2019.

Farhi, E., Neven, H., et al. Classification with quantum neural networks on near term processors. *Quantum Review Letters*, 1(2 (2020)):10–37686, 2020.

Ge, R. and Ma, T. On the optimization landscape of tensor decompositions. In *Advances in Neural Information Processing Systems*, pp. 3653–3663, 2017.

Goel, S. and Klivans, A. R. Learning neural networks with two nonlinear layers in polynomial time. In *Conference on Learning Theory*, pp. 1470–1499. PMLR, 2019.

Goel, S., Kanade, V., Klivans, A., and Thaler, J. Reliably learning the relu in polynomial time. In *Conference on Learning Theory*, pp. 1004–1042. PMLR, 2017.

Golub, G. H. and Van Loan, C. F. *Matrix computations*. JHU press, 1996.

Harrow, A. W. and Montanaro, A. Quantum computational supremacy. *Nature*, 549(7671):203–209, 2017.

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*, 567(7747):209–212, 2019.

Horn, R. A. and Johnson, C. R. *Matrix analysis*. Cambridge university press, 2013.

Jacot, A., Hongler, C., and Gabriel, F. Neural tangent kernel: Convergence and generalization in neural networks. In Bengio, S., Wallach, H. M., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pp. 8580–8589, 2018. URL https://proceedings.neurips.cc/paper/2018/hash/5a4be1fa34e62bb8a6ec6b91d2462f5a-Abstract.html.

Kawaguchi, K. Deep learning without poor local minima. In *Advances in neural information processing systems*, pp. 586–594, 2016.

Kiani, B. T., Lloyd, S., and Maity, R. Learning unitaries by gradient descent, 2020.

Killoran, N., Bromley, T. R., Arrazola, J. M., Schuld, M., Quesada, N., and Lloyd, S. Continuous-variable quantum neural networks. *Physical Review Research*, 1(3):033063, 2019.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6980.

Li, D., Ding, T., and Sun, R. Over-parameterized deep neural networks have no strict local minima for any continuous activations. *arXiv preprint arXiv:1812.11039*, 2018.

Li, J., Yang, X., Peng, X., and Sun, C.-P. Hybrid quantum-classical approach to quantum optimal control. *Physical review letters*, 118(15):150503, 2017.

Liu, D. C. and Nocedal, J. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, 45(1-3):503–528, 1989.

Lloyd, S., Schuld, M., Ijaz, A., Izaac, J., and Killoran, N. Quantum embeddings for machine learning. *arXiv preprint arXiv:2001.03622*, 2020.

Mari, A., Bromley, T. R., Izaac, J., Schuld, M., and Killoran, N. Transfer learning in hybrid classical-quantum neural networks. *Quantum*, 4:340, 2020.

McClean, J. R., Boixo, S., Smelyanskiy, V. N., Babbush, R., and Neven, H. Barren plateaus in quantum neural network training landscapes. *Nature communications*, 9 (1):1–6, 2018.

Mitarai, K., Negoro, M., Kitagawa, M., and Fujii, K. Quantum circuit learning. *Physical Review A*, 98(3):032309, 2018.

Nguyen, Q. and Hein, M. The loss surface of deep and wide neural networks. In *International conference on machine learning*, pp. 2603–2612. PMLR, 2017.

Nielsen, M. A. and Chuang, I. Quantum computation and quantum information, 2002.

Ostaszewski, M., Grant, E., and Benedetti, M. Quantum circuit structure learning. *arXiv preprint arXiv:1905.09692*, 2019.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems*, pp. 8024–8035, 2019.

Peruzzo, A., McClean, J., Shadbolt, P., Yung, M.-H., Zhou, X.-Q., Love, P. J., Aspuru-Guzik, A., and O'brien, J. L. A variational eigenvalue solver on a photonic quantum processor. *Nature Communications*, 5:4213, 2014.

Petz, D. A survey of certain trace inequalities. *Banach Center Publications*, 30(1):287–298, 1994.

Petz, D. and Zemánek, J. Characterizations of the trace. *Linear Algebra and its Applications*, 111:43–52, 1988.

Preskill, J. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, 2018. ISSN 2521-327X.

Puchała, Z. and Miszczak, J. A. Symbolic integration with respect to the haar measure on the unitary group. *arXiv preprint arXiv:1109.4244*, 2011.

Rabitz, H. A., Hsieh, M. M., and Rosenthal, C. M. Quantum optimally controlled transition landscapes. *Science*, 303 (5666):1998–2001, 2004.

Russell, B., Rabitz, H., and Wu, R. Quantum control landscapes are almost always trap free. *arXiv preprint arXiv:1608.06198*, 2016.

Safran, I. and Shamir, O. Spurious local minima are common in two-layer relu neural networks. In *International Conference on Machine Learning*, pp. 4433–4441. PMLR, 2018.

Sagun, L., Güney, V. U., and LeCun, Y. Explorations on high dimensional landscapes. In Bengio, Y. and LeCun, Y. (eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*, 2015. URL http://arxiv.org/abs/1412.6615.

Schuld, M. and Killoran, N. Quantum machine learning in feature hilbert spaces. *Physical review letters*, 122(4): 040504, 2019.

Soudry, D. and Carmon, Y. No bad local minima: Data independent training error guarantees for multilayer neural networks. *arXiv preprint arXiv:1605.08361*, 2016.

Sweke, R., Wilde, F., Meyer, J. J., Schuld, M., Fährmann, P. K., Meynard-Piganeau, B., and Eisert, J. Stochastic gradient descent for hybrid quantum-classical optimization. *Quantum*, 4:314, 2020.

Venturi, L., Bandeira, A. S., and Bruna, J. Spurious valleys in two-layer neural network optimization landscapes. *arXiv preprint arXiv:1802.06384*, 2018.

Wang, Z., Hadfield, S., Jiang, Z., and Rieffel, E. G. Quantum approximate optimization algorithm for maxcut: A fermionic view. *Phys. Rev. A*, 97: 022304, Feb 2018. doi: 10.1103/PhysRevA.97.022304. URL https://link.aps.org/doi/10.1103/PhysRevA.97.022304.

Wang, Z., Rubin, N. C., Dominy, J. M., and Rieffel, E. G. xy mixers: Analytical and numerical results for the quantum alternating operator ansatz. *Phys. Rev. A*, 101:012320, Jan 2020. doi: 10.1103/PhysRevA.101.012320. URL https://link.aps.org/doi/10.1103/PhysRevA.101.012320.

Watrous, J. *The theory of quantum information*. Cambridge University Press, 2018.

Yun, C., Sra, S., and Jadbabaie, A. Small nonlinearities in activation functions create bad local minima in neural networks. *arXiv preprint arXiv:1802.03487*, 2018.

Zhong, H.-S., Wang, H., Deng, Y.-H., Chen, M.-C., Peng, L.-C., Luo, Y.-H., Qin, J., Wu, D., Ding, X., Hu, Y., Hu, P., Yang, X.-Y., Zhang, W.-J., Li, H., Li, Y., Jiang, X., Gan, L., Yang, G., You, L., Wang, Z., Li, L., Liu, N.-L., Lu, C.-Y., and Pan, J.-W. Quantum computational advantage using photons. *Science*, 370(6523):1460–1463, 2020. ISSN 0036-8075. doi: 10.1126/science.abe8770. URL https://science.sciencemag.org/content/370/6523/1460.

Zhong, K., Song, Z., Jain, P., Bartlett, P. L., and Dhillon, I. S. Recovery guarantees for one-hidden-layer neural networks. In *International conference on machine learning*, pp. 4140–4149. PMLR, 2017.