Robust data encodings for quantum classifiers

Ryan LaRose [1] and Brian Coyle [2,*]

[1]*Department of Computational Mathematics, Science, and Engineering, Michigan State University, East Lansing, Michigan 48823, USA*
[2]*School of Informatics, University of Edinburgh, 10 Crichton Street, United Kingdom*

Data representation is crucial for the success of machine-learning models. In the context of quantum machine learning with near-term quantum computers, equally important considerations of how to efficiently input (encode) data and effectively deal with noise arise. In this paper, we study data encodings for binary quantum classification and investigate their properties both with and without noise. For the common classifier we consider, we show that encodings determine the classes of learnable decision boundaries as well as the set of points which retain the same classification in the presence of noise. After defining the notion of a robust data encoding, we prove several results on robustness for different channels, discuss the existence of robust encodings, and prove a lower bound on the number of robust points in terms of fidelities between noisy and noiseless states. Numerical results for several example implementations are provided to reinforce our findings.

DOI: 10.1103/PhysRevA.102.032420

I. INTRODUCTION

Fault-tolerant quantum computers which can efficiently simulate physics [1] and factor prime numbers [2] lie on an unclear timeline. Current quantum processors in laboratories or in the cloud [3] contain fewer than 100 qubits with short lifetimes and noisy gate operations. Such devices cannot yet implement fault-tolerant procedures and are said to belong to the noisy, intermediate scale quantum (NISQ) era [4]. While the gap between NISQ capabilities and required resources for, e.g., factoring 2048-bit integers is large [5], recent hardware advancements have led to the first quantum computation which classical computers cannot emulate [6]. This combination of improved hardware and unclear timeline for fault tolerance makes the question "What (useful) applications can NISQ computers implement?" both interesting and important to consider.

A leading candidate for NISQ applications is a class of algorithms known as variational quantum algorithms (VQAs) [7]. VQAs use a NISQ computer to evaluate an objective function and a classical computer to adjust input parameters to optimize the function. VQAs have been proposed or used for many applications including quantum chemistry [8], approximate optimization [9], quantum state diagonalization [10], quantum compilation [11,12], quantum field theory simulation [13,14], linear systems of equations [15–17], and even quantum foundations [18]. More fundamental questions about the computational complexity [7,19], trainability [20–22], and noise resilience [23] of VQAs have also been considered.

Variational quantum algorithms have strong overlap with machine-learning algorithms which seek to train a computer to recognize patterns by designing and minimizing a cost function defined over an input data set [24]. In this context, VQAs can be considered quantum neural networks (QNNs) [25,26], and a multitude of applications from classical machine learning can be realized with (simulated) quantum computers. Such applications include generative modeling [27–31], transfer learning [32], and classification [25,33–38]. These QNN applications, along with additional techniques and applications based on quantum kernel methods [39–42], define the emerging field of quantum machine learning (QML)[1] [34,43].

Despite these applications, several fundamental questions lie at the forefront of QML. Perhaps the most pressing question is whether quantum models can provide any advantages over classical models. While some (generally) negative theoretical results have been shown for sampling complexity [45] and information capacity [46], other (generally) positive results have been shown for expressive power [30,47] and problem-specific sampling complexity [48,49], and the question is largely open. Any practical experiments toward demonstrating advantage give rise to additional crucial questions. One question, general to all NISQ applications, is how to deal with noise present in NISQ computers. A second question, specific to QML applications, is how to (efficiently) input or *encode* data into a quantum state such that it can be processed by a QML model. This encoding encompasses both how to best represent data (a consideration also present in classical machine learning, e.g., in support vector machines) as well as how to practically create the quantum state by means of a state preparation unitary or other procedure.

---

[1]Although quantum machine learning is generally considered an emerging field, foundational ideas were published more than two decades ago [44].

---

*brian.coyle@ed.ac.uk

For the first question, several general-purpose strategies for dealing with noise such as dynamical decoupling [50], probabilistic error cancellation [51], zero-noise extrapolation [51,52], and quantum subspace expansion [53] have been proposed. However, the robustness (resilience) of particular VQAs in the presence of noise has not been thoroughly investigated, with the exception of very recent studies in quantum compiling [11,23] and approximate optimization [54–56]. Understanding robustness properties of VQAs is key to making progress towards practical NISQ implementations. For the second question on inputting data, most studies in QML focus on the design of the QNN while assuming a full wavefunction representation of arbitrary data [25,33,57–61]. This assumption is not suitable for practical implementations as it is well known that preparing an arbitrary quantum state requires a number of gates exponential in the number of qubits [62].

In this paper, we study both of the above questions in the context of binary quantum classification. In particular, we define different methods of encoding data and analyze their properties both with and without noise. For the noiseless case, we demonstrate that different data encodings lead to different sets of learnable decision boundaries for the quantum classifier. For the noisy case, given a quantum channel, we define the notion of a robust point for the quantum classifier—a generalization of fixed points for quantum operations. We completely characterize the set of robust points for example quantum channels and discuss how encoding data into robust points is a type of problem-specific error mitigation for quantum classifiers.

To these ends, the rest of the paper is organized as follows. Section II presents definitions which are used in the remainder to prove our results, including a formal definition of a common binary quantum classifier we consider (Sec. II A), definitions and examples of data encodings (Sec. II B), noise channels we consider (Sec. II C), and definitions of robust points and robust encodings (Sec. II D). After this, we present analytic results and proofs for robustness in Section III. We begin by showing that different encodings lead to different classes of learnable decision boundaries in Sec. III A, then characterize the set of robust points for example quantum channels in Sec. III B. In Sec. III C, we state and prove robustness results, and in Sec. III E we discuss the existence of robust encodings. Finally, we prove an upper bound on the number of robust points in terms of fidelities between noisy and ideal states in Sec. III F. Last, we include several numerical results in Sec. IV that reinforce and extend our findings, and finally conclude in Sec. V.

II. PRELIMINARY DEFINITIONS

A. Quantum classifiers

In classical machine learning, classification problems are a subclass of supervised learning problems in which the computer model is presented with labeled data and asked to learn some pattern. For binary classification, the input is a set of labeled feature vectors

$$\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{M},$$

where $\boldsymbol{x}_i \in \mathcal{X}$ is a feature vector, $\mathcal{X}$ is an arbitrary set [2], and $y_i \in \{0, 1\}$ is a binary label. Given this data, the goal of the learner is to output a rule $f : \mathcal{X} \rightarrow \{0, 1\}$ which accurately classifies the data and can be used to make predictions on new data. In practice, this is accomplished by defining a model (e.g., a neural network) and cost function, then minimizing this cost function by "training" the model over the input data.

In this paper, we restrict to binary (quantum) classifiers, henceforth called simply (quantum) classifiers. We remark that multilabel classification problems can be reduced to binary classification by standard methods.

A quantum classifier is essentially the same as a classical one, the key difference being the model (Ansatz) of the classifier. As mentioned in the Introduction, many QML architectures use a VQA, nominally consisting of parameterized one- and two-qubit gates, which is a QNN in the context of machine learning. QNNs can be considered function approximators analogous to classical neural networks (e.g., feedforward neural networks [63]), and the procedure for training the QNN consists of adjusting gate parameters such that this function approximator outputs good predictions for the input data.

While the use of QNNs as machine-learning models may present the possibility of advantage for particular problems [30,47], QNNs also present key challenges for machine learning. For nearly all QML problems, a pressing challenge is inputting (arbitrary) data to the model such that the QNN can process it. We refer to this input process as *data encoding*, and discuss it in detail in Sec. II B. Another potential challenge with QNNs is outputting information, since the data propagated through the QNN is a quantum state. For machine-learning applications, this means that the output feature vector (amplitudes of the quantum state) cannot be accessed efficiently. Rather, as is usual in quantum mechanics, quantities of the form $\text{Tr}[\rho \hat{O}]$, where ρ is the quantum state and $\hat{O}$ is some Hermitian operator, can be efficiently estimated to a set precision.

For quantum classification, outputting information (predictions) can be done in a relatively straightforward manner. As several authors have noted [25,33–37], it is natural to use the measurement outcome of a single qubit, which produces a binary outcome, as a class prediction. We adopt this strategy in our paper.

Informally, we define a (binary) quantum classifier as a procedure for encoding data into a quantum circuit, processing it through trainable QNN, and outputting a (binary) predicted label. Given a feature vector $\boldsymbol{x} \in \mathcal{X}$, a concise description of such a classifier can be written

$$\boldsymbol{x} \mapsto \rho_{\boldsymbol{x}} \qquad \text{(encoding)}, \qquad (1)$$

$$\mapsto \tilde{\rho}_{\boldsymbol{x}} \qquad \text{(processing)}, \qquad (2)$$

$$\mapsto \hat{y}[\tilde{\rho}_{\boldsymbol{x}}] \qquad \text{(prediction)}. \qquad (3)$$

Several remarks are in order. First, a given data point $\boldsymbol{x}$ in the training set (Sec. II A) is encoded in a quantum state

---

[2]In practice, we typically have $\mathcal{X} = \mathbb{R}^N$, but other sets—e.g., $\mathcal{X} = \mathbb{Z}^N$ or $\mathcal{X} \in \mathbb{Z}_2^N$—are possible, so we write $\mathcal{X}$ for generality.
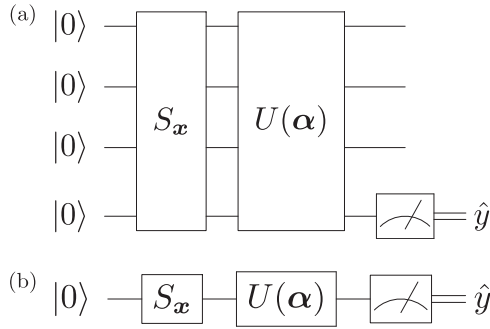
FIG. 1. A common architecture for a binary quantum classifier that we study in this paper. The general circuit structure is shown in (a) and the structure for a single qubit is highlighted in (b). In both, a feature vector x is encoded into a quantum state ρ_x via a "state preparation" unitary S_x. The encoded state ρ_x then evolves to $U\rho_x U^\dagger =: \tilde{\rho}_x$ where $U(\boldsymbol{\alpha})$ is a unitary Ansatz with trainable parameters $\boldsymbol{\alpha}$. A single qubit of the evolved state $\tilde{\rho}_x$ is measured to yield a predicted label $\hat{y}$ for the vector x.

$\rho_x \in \mathcal{D}_n$ via a state preparation unitary S_x (see Fig. 1). Throughout the paper, we use $\mathcal{D}_n \subset \mathbb{C}^{2^n \times 2^n}$ to denote the set of density operators (matrices) on n qubits. We remark that each x in the training set leads to a (unique) S_x, so the state preparation unitary can be considered a parameterized family of unitary Ansätze. We discuss encodings in detail in Sec. II B.

For the processing step (2), there have been many proposed QNN architectures in recent literature, including quantum convolutional neural networks [64,65], strongly entangling Ansätze [33], and more [35,66]. In this paper, we allow for a general unitary evolution $U(\boldsymbol{\alpha})$ such that

$$\tilde{\rho}_x = U(\boldsymbol{\alpha})\rho_x U^\dagger(\boldsymbol{\alpha}). \tag{4}$$

We remark that some QNN architectures involve intermediate measurements and conditional processing (notably, Ref. [64]) and so do not immediately fit into (4). Our techniques for
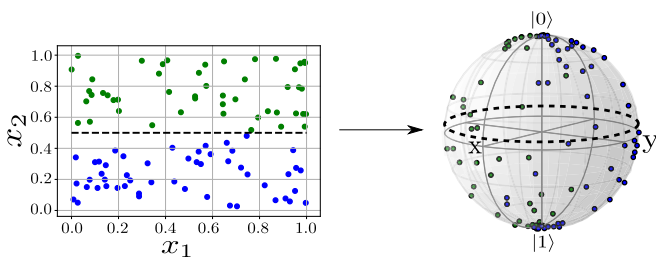

FIG. 2. A visual representation of data encoding (7) for a single qubit. On the left is shown a set of randomly generated points $\{x_i, y_i\}_{i=1}^M$ normalized to lie within the unit square, ($x_i = [x_{1,i}, x_{2,i}]^T$), separated by a true decision boundary shown by the dashed black line. A data encoding maps each $x_i \in \mathbb{R}^2$ to a point on the Bloch sphere $\rho_{x_i} \in \mathbb{C}^2$, here using the dense angle encoding (14). The dashed black line on the Bloch sphere shows the initial decision boundary of the quantum classifier. During the training phase, unitary parameters are adjusted to rotate the dashed black line to correctly classify as many training points as possible. Different data encodings lead to different learnable decision boundaries and different robustness properties, as discussed in the main text.

showing robustness could be naturally extended to such architectures, however, and so we consider (4) as a simple yet general model. We also note that training the classifier via minimization of a well-defined cost function is an important task with interesting questions, but we primarily focus on data encodings and their properties in this paper. For this reason, we often suppress the trainable parameters $\boldsymbol{\alpha}$ and write U for $U(\boldsymbol{\alpha})$.

Finally, the remaining step is to extract information from the state $\tilde{\rho}_x$ to obtain a predicted label. As mentioned, a natural method for doing this is to measure a single qubit which yields a binary outcome 0 or 1 taken as the predicted label $\hat{y}$. Since measurements are probabilistic, we measure N_m times and take a majority vote. That is, if 0 is measured N_0 times and $N_0 \geqslant N_m/2$, we take 0 as the class prediction, otherwise 1. Generalizing the finite statistics,[3] this condition can be expressed analytically as

$$\hat{y}[\tilde{\rho}_x] = \begin{cases} 0 & \text{if } \mathrm{Tr}[\Pi_0^c \tilde{\rho}_x] \geqslant 1/2 \\ 1 & \text{otherwise,} \end{cases} \tag{5}$$

where

$$\Pi_0^c := |0\rangle\langle 0|_c \equiv |0\rangle\langle 0|_c \otimes I_{\bar{c}} \tag{6}$$

is the projector onto the ground state of the classification qubit, labeled c, and the remaining qubits are labeled $\bar{c}$. For brevity, we often omit these labels when it is clear from context. Throughout the paper, we use $\hat{y}$ for predicted labels and y for true labels, and we refer to (5) as the *decision rule* for the classifier. Equation (5) is not the only choice for such a decision rule. In particular, one could choose a different "weight" λ such that $\hat{y} = 0$ if $\mathrm{Tr}[\Pi_0 \tilde{\rho}_x] \geqslant \lambda$ as in Ref. [36], add a bias to the classifier as in Ref. [33], or measure the classification qubit in a different basis (e.g., the Hadamard basis instead of the computational basis).

Our techniques for showing robustness (Sec. III C) could be easily adapted for such alternate decision boundaries.

The preceding discussion is summarized in the following formal definition of a quantum classifier.

Definition 1 (Quantum classifier). A (binary) quantum classifier consists of three well-defined functions:

(i) an encoding function

$$E : \mathcal{X} \to \mathcal{D}_n, \tag{7}$$

$$E(x) = \rho_x, \tag{8}$$

(ii) a function which evolves the state

$$\mathcal{U} : \mathbb{C}^{2^n \times 2^n} \to \mathbb{C}^{2^m \times 2^m}, \tag{9}$$

$$\mathcal{U}(\rho_x) = \tilde{\rho}_x, \tag{10}$$

and (iii) a decision rule:

$$\hat{y} : \mathbb{C}^{2^m \times 2^m} \to \{0, 1\}. \tag{11}$$

---

[3]We discuss details arising from finite statistics in Sec. III D. The results we prove can be easily modified to incorporate finite statistics as shown in this section, but they are simpler to state in terms of probabilities.

Note that we have not included training data or a cost function in this definition, so a quantum classifier can be considered a hypothesis family. We can distinguish between the hypothesis family and the trained model—in which optimization has been performed to minimize a cost function over a specified training data set—by referring to the latter as the *realized* quantum classifier if it is not clear from context.

In the remainder, we study the effects of different encoding functions (7), which we now discuss in more detail.

B. Data encodings

An encoding can be thought of as "loading" a data point $\boldsymbol{x} \in \mathcal{X}$ from memory into a quantum state so it can be processed by a QNN. This loading is accomplished by an encoding function (7) from the data set $\mathcal{X}$ to n-qubit quantum states $\mathcal{D}_n$. As mentioned, many QML papers [57–61] assume a full amplitude encoding with $n = \log_2 N$. This provides an exponential saving in "space" at the cost of an exponential increase in "time." That is, a quantum state of $n = \log_2 N$ qubits can represent a data point with N features, but in general such a quantum state takes time $O(2^n)$ to prepare [62].

In practice, data is encoded via a state preparation circuit (unitary) $S_{\boldsymbol{x}}$—written in terms of one- and two-qubit gates—which acts on an initial state $|\phi\rangle$, nominally the all-zero state $|\phi\rangle = |0\rangle^{\otimes n}$. This realizes the encoding

$$\boldsymbol{x} \mapsto E(\boldsymbol{x}) = S_{\boldsymbol{x}}|\phi\rangle\langle\phi|S_{\boldsymbol{x}}^{\dagger} = |\boldsymbol{x}\rangle\langle\boldsymbol{x}| =: \rho_{\boldsymbol{x}}.$$

For $S_{\boldsymbol{x}}$ to be useful for a data encoding, it should have several desirable properties. First, $S_{\boldsymbol{x}}$ should have a number of gates which is at most polynomial in the number of qubits as well as the size of the data set, dimension of the data to be encoded, and all other relevant input parameters. For machine-learning applications, we want the family of state preparation unitaries to have enough free parameters such that there is a unique quantum state $\rho_{\boldsymbol{x}}$ for each feature vector $\boldsymbol{x}$—i.e., such that the encoding function E is bijective. Additionally, for NISQ applications, subpolynomial depth is even more desirable, and we want $S_{\boldsymbol{x}}$ to be hardware efficient—meaning that the one- and two-qubit gates comprising $S_{\boldsymbol{x}}$ can be realized without too much overhead due to, e.g., compiling into the computer's native gate set and implementing swap gates to connect disjoint qubits.

Motivated by such NISQ limitations, some recent authors [34,35,66,67] have considered a qubit encoding,

$$|\boldsymbol{x}\rangle = \bigotimes_{i=1}^{N} \cos(x_i)|0\rangle + \sin(x_i)|1\rangle, \tag{12}$$

for the feature vector $\boldsymbol{x} = [x_1, ..., x_N]^T \in \mathcal{X}^N$. (Note that for pure state encodings, we often write only the wave function $|\boldsymbol{x}\rangle \in \mathbb{C}^{2^n}$, from which the density matrix $\rho_{\boldsymbol{x}} = |\boldsymbol{x}\rangle\langle\boldsymbol{x}| \in \mathcal{D}_n$ is implicit.) We will also refer to (12) as an angle encoding. The angle encoding uses N qubits with a constant depth quantum circuit and is thus amenable to NISQ computers. The state preparation unitary is $S_{\boldsymbol{x}_j} = \bigotimes_{i=1}^{N} U_i$ where

$$U_i := \begin{bmatrix} \cos(x_j^{(i)}) & -\sin(x_j^{(i)}) \\ \sin(x_j^{(i)}) & \cos(x_j^{(i)}) \end{bmatrix},$$

a strategy which encodes one feature per qubit.

This encoding can be slightly generalized to encode two features per qubit by exploiting the relative phase degree of freedom. We refer to this as the dense angle encoding and include a definition below.

Definition 2 (Ddense angle encoding). Given a feature vector $\boldsymbol{x} = [x_1, ..., x_N]^T \in \mathbb{R}^N$, the dense angle encoding maps $\boldsymbol{x} \mapsto E(\boldsymbol{x})$ given by

$$|\boldsymbol{x}\rangle = \bigotimes_{i=1}^{\lceil N/2 \rceil} \cos(\pi x_{2i-1})|0\rangle + e^{2\pi i x_{2i}} \sin(\pi x_{2i-1})|1\rangle. \tag{13}$$

For some of our analytic and numerical results, we highlight the dense angle encoding for two-dimensional data $\boldsymbol{x} \in \mathbb{R}^2$ with a single qubit given by

$$|\boldsymbol{x}\rangle = \cos(\pi x_1)|0\rangle + e^{2\pi i x_2} \sin(\pi x_1)|1\rangle, \tag{14}$$

which has density matrix

$$\rho_{\boldsymbol{x}} = \begin{bmatrix} \cos^2 \pi x_1 & e^{-2\pi i x_2} \cos \pi x_1 \sin \pi x_1 \\ e^{2\pi i x_2} \cos \pi x_1 \sin \pi x_1 & \sin^2 \pi x_1 \end{bmatrix}.$$

Although the angle encoding (12) and dense angle encoding (13) use sinusoids and exponentials, there is nothing special about these functions (other than, perhaps, they appear in common parametrizations of qubits and unitary matrices [68]). We can easily abstract these to a general class of qubit encodings which use arbitrary functions.

Definition 3 (Ggeneral qubit encoding). Given a feature vector $\boldsymbol{x} = [x_1, ..., x_N]^T \in \mathbb{R}^N$, the general qubit encoding maps $\boldsymbol{x} \mapsto E(\boldsymbol{x})$ given by

$$|\boldsymbol{x}\rangle = \bigotimes_{i=1}^{\lceil N/2 \rceil} f_i(x_{2i-1}, x_{2i})|0\rangle + g_i(x_{2i-1}, x_{2i})|1\rangle. \tag{15}$$

where $f, g : \mathbb{R} \times \mathbb{R} \to \mathbb{C}$ are such that $|f_i|^2 + |g_i|^2 = 1 \ \forall i$.

We remark that a similar type of generalization was used in Ref. [36] with a single qubit classifier that allowed for repeated application of an arbitrary state preparation unitary.

While (15) is the most general description of a qubit encoding—and is the encoding we primarily focus on in this paper—it is of course not the most general encoding (7). The previously mentioned amplitude encoding maps N features into $n = \log_2 N$ qubits as follows.

Definition 4 (Aamplitude encoding). The amplitude encoding of a vector $\boldsymbol{x} \in \mathbb{R}^N$ is

$$|\boldsymbol{x}\rangle := \frac{1}{||\boldsymbol{x}||_2^2} \sum_{i=1}^{N} x_i|i\rangle, \tag{16}$$

where x_i is the ith feature of $\boldsymbol{x}$.

As with the dense angle encoding, we highlight the amplitude encoding for $\boldsymbol{x} = [x_1 \ x_2]^T$ with $||\boldsymbol{x}||_2 = 1$:

$$\rho_{\boldsymbol{x}} = \begin{bmatrix} x_1^2 & x_1 x_2 \\ x_1 x_2 & x_2^2 \end{bmatrix}. \tag{17}$$

We note that the amplitude encoding can be slightly generalized to allow for parametrizations of features (amplitudes).

Definition 5 (Ggeneralized amplitude encoding). For $\boldsymbol{x} \in \mathbb{R}^N$, the generalized amplitude encoding maps $\boldsymbol{x} \mapsto E(\boldsymbol{x})$ is

given by

$$|\boldsymbol{x}\rangle = \sum_{i=1}^{N} f_i(\boldsymbol{x})|i\rangle,$$

where $\sum_i |f_i|^2 = 1$.

The functions f_i could only act on the ith feature, e.g., $f_i(\boldsymbol{x}) = \sin x_i$, or could be more complicated functions of several (or all) features.

Thus far, we have formally defined a data encoding (15) and its role in a quantum classifier (Def. 1), and we have given several examples. While we have discussed different properties of state preparation circuits which implement data encodings (depth, overhead, etc.), we have not yet discussed the two main properties of data encodings we consider in this paper: *learnability* and *robustness*. By *learnability*, we mean the expressive power [69] of a given hypothesis family in its ability to find a set of parameters such that the realized quantum classifier can (optimally) separate the data classes. In other words, learnability measures the extent to which the quantum classifier can represent certain functions. We show in Sec. III A that different data encodings lead to different classes of learnable decision boundaries. For *robustness*, we show that different data encodings lead to different sets of robust points (to be defined) in Sec. III B—Sec. III E. For the latter results which constitute the bulk of our work, we first need to introduce the noise channels we consider and define the notion of a robust point, which we do in the following two sections.

C. Noise in quantum systems

In this section, we introduce our notation for the common quantum channels we use in this paper. While we provide brief exposition on quantum noise, we refer the reader desiring more background to the standard references [68,70,71].

Noise occurs in quantum systems due to interactions with the environment. Letting ρ denote the quantum state of interest and ρ_{env} the environment, noise can be characterized physically by the process

$$\rho \mapsto \mathrm{Tr}_{\mathrm{env}}[U(\rho \otimes \rho_{\mathrm{env}})U^\dagger],$$

where U is a unitary on the composite Hilbert space. This can be written in the equivalent, often more convenient, operator-sum representation

$$\rho \mapsto \sum_{k=1}^{K} E_k \rho E_k^\dagger, \tag{18}$$

where the Kraus operators E_k satisfy the completeness relation

$$\sum_{k=1}^{K} E_k^\dagger E_k = I.$$

Equation (18) is known as a quantum operation or quantum channel. Physically, it can be interpreted as randomly replacing the state ρ by the (properly normalized) state $E_k \rho E_k^\dagger$ with probability $\mathrm{Tr}[E_k \rho E_k^\dagger]$.

We note that random and systematic coherent errors (defined by $K = 1$ unitary Kraus operators) can be accounted for in the training of variational circuits [11,23,72]. For this

reason, we primarily focus on incoherent errors defined below. The quantum channels we study here are standard and often used in theoretical work as reasonable noise models [68]. For readers familiar with these channels, the following definitions are solely to introduce our notation.

A commonly used noise model is the Pauli channel.

Definition 6. The Pauli channel maps a single qubit state ρ to $\mathcal{E}_{\mathbf{p}}^{\mathrm{P}}(\rho)$ defined by

$$\mathcal{E}_{\mathbf{p}}^{\mathrm{P}}(\rho) := p_I \rho + p_X X \rho X + p_Y Y \rho Y + p_Z Z \rho Z, \tag{19}$$

where $p_I + p_X + p_Y + p_Z = 1$.

While the Pauli channel acts on a single qubit, it can be generalized to a d-dimensional Hilbert space via the *Weyl channel*,

$$\mathcal{E}_p^{\mathrm{W}}(\rho) := \sum_{k,l=0}^{d-1} p_{kl} W_{kl} \rho W_{kl}^\dagger, \tag{20}$$

where p_{kl} are probabilities and the Weyl operators are

$$W_{kl} := \sum_{m=0}^{d-1} e^{2\pi imk/d}|m\rangle\langle m+1|.$$

For $d = 2$, Eq. (20) reduces to Eq. (19).

Two special cases of the Pauli channel are the bit-flip and phase-flip (dephasing) channel.

Definition 7. The bit-flip channel maps a single qubit state ρ to $\mathcal{E}_p^{\mathrm{BF}}(\rho)$ defined by

$$\mathcal{E}_p^{\mathrm{BF}}(\rho) := (1-p)\rho + pX\rho X \tag{21}$$

where $0 \leqslant p \leqslant 1$.

While a bit-flip channel flips the computational basis state with probability p, the phase-flip channel introduces a relative phase with probability p.

Definition 8. The phase-flip (dephasing) channel maps a single qubit state ρ to $\mathcal{E}_p^{\mathrm{deph}}(\rho)$ defined by

$$\mathcal{E}_p^{\mathrm{deph}}(\rho) := (1-p)\rho + pZ\rho Z, \tag{22}$$

where $0 \leqslant p \leqslant 1$.

Another special case of the Pauli channel is the depolarizing channel which occurs when each Pauli is equiprobable $p_X = p_Y = p_Z = p$ and $p_I = 1 - 3p$. This channel can be equivalently thought of as replacing the state ρ by the maximally mixed state $I/2$ with probability p.

Definition 9. The depolarizing channel maps a single qubit state ρ to $\mathcal{E}_p^{\mathrm{depo}}(\rho)$ defined by

$$\mathcal{E}_p^{\mathrm{depo}}(\rho) := (1-p)\rho + pI/2, \tag{23}$$

where $0 \leqslant p \leqslant 1$.

The $d = 2^n$-dimensional generalization of Def. 9 is straightforward.

Definition 10. The global depolarizing channel maps an n-qubit state ρ to $\mathcal{E}_p^{\mathrm{GD}}(\rho)$ defined by

$$\mathcal{E}_p^{\mathrm{GD}}(\rho) := (1-p)\rho + pI/d, \tag{24}$$

where $0 \leqslant p \leqslant 1$, $d = 2^n$, and $I \equiv I_d$ is the d-dimensional identity.
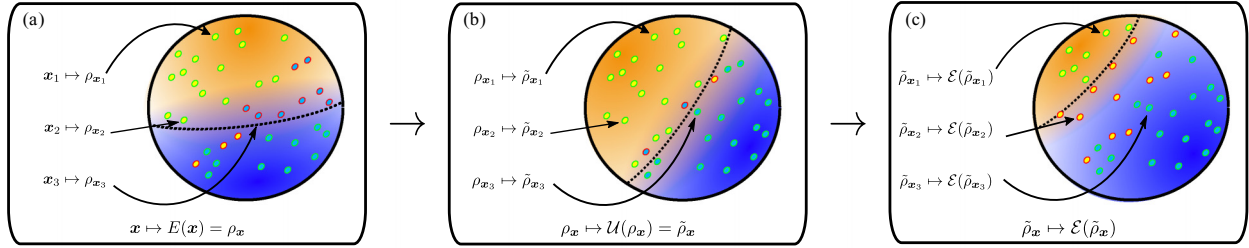
FIG. 3. Cartoon illustration of robust points for a single qubit classifier. In panel (a), input training data points $\boldsymbol{x}_i$ with classes $y_i \in$ {yellow, blue} are mapped into quantum states $\rho_{\boldsymbol{x}_i}$ according to some encoding function E. The dashed line through the Bloch sphere indicates the initial decision boundary. Points with a green outline are classified correctly, while points with a red outline are misclassified. In (b), data points are processed by the QNN with optimal unitary parameters (after minimizing a cost function to find such parameters). For clarity, we keep data points fixed and adjust the location of the decision boundary, which is now rotated to correctly classify more points (fewer points with red outlines). In (c), a noise process $\mathcal{E}$ occurs which shifts the location of the final processed points (or location of decision boundary), causing some points to be misclassified. The set of points which maintain the same classification in (b) and (c) are the robust points. Example points $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are correctly classified in (a) and (b) then misclassified in (c) due to the noise. Example point $\boldsymbol{x}_3$ is incorrectly classified in (a), correctly classified in (b) after propagating through the QNN, and remains correctly classified in (c).

Finally, we consider amplitude damping noise which models decay from the excited state to the ground state via spontaneous emission of a photon.

Definition 11. The amplitude damping channel maps a single qubit state ρ to $\mathcal{E}_p^{\mathrm{AD}}(\rho)$ defined by

$$\mathcal{E}_p^{\mathrm{AD}}(\rho) := \begin{bmatrix} \rho_{00} + p\rho_{11} & \sqrt{1-p}\rho_{01} \\ \sqrt{1-p}\rho_{10} & (1-p)\rho_{11} \end{bmatrix}, \qquad (25)$$

where $0 \leqslant p \leqslant 1$.

Now that we have introduced quantum noise and several common channels, we can define robust data encodings for quantum classifiers. We remark that we also consider measurement noise in Appendix B but omit the definition and results from the main text for brevity.

D. Robust data encodings

In this section, we define robust points and robust data encodings of quantum classifiers. Informally, the intuition is as follows: the quantum classifier with decision rule (5) requires only a "coarse-grained" measurement to extract a predicted label. For example, with a single qubit classifier, all points in the "top" hemisphere of the Bloch sphere are predicted to have label 0, while all points in the "bottom" hemisphere are predicted to have label 1. The effect of noise is to shift points on the Bloch sphere, but certain points can get shifted such that they get assigned the same labels they would without noise. This is the idea of robustness, represented schematically in Fig. 3   . For classification purposes, we do not require completely precise measurements, only that the point remain in the same hemisphere to get the same predicted label.

Formally, we define a robust point as follows.

Definition 12 (Rrobust point). Let $\mathcal{E}$ be a quantum channel, and consider a (binary) quantum classifier with decision rule $\hat{y}$ as defined in (5). We say that the state $\rho_{\boldsymbol{x}} \in \mathcal{D}_n$ encoding a data point $\boldsymbol{x} \in \mathcal{X}$ is a *robust point* of the quantum classifier if and only if

$$\hat{y}[\mathcal{E}(\tilde{\rho}_{\boldsymbol{x}})] = \hat{y}[\tilde{\rho}_{\boldsymbol{x}}], \qquad (26)$$

where $\tilde{\rho}_{\boldsymbol{x}}$ is the processed state via (9).

As mentioned, for the purpose of classification, Eq. (26) is a well-motivated and reasonable definition of robustness. We remark that (26) is expressed in terms of probability; in practice, additional measurements may be required to reliably determine robustness. We discuss this point further in Sec. III D.

Further, we note that (26) assumes that noise occurs only after the evolution $\rho_{\boldsymbol{x}} \mapsto \tilde{\rho}_{\boldsymbol{x}}$. While this may be a useful theoretical assumption, in practice, noise happens throughout a quantum circuit. We can therefore consider robustness for an *ideal data encoding* as in Def. 12, or for a *noisy data encoding* in which some noise process $\mathcal{E}_1$ occurs after encoding and another noise process $\mathcal{E}_2$ occurs after evolution:

$$\hat{y}[\mathcal{E}_2(\mathcal{U}(\mathcal{E}_1(\rho_{\boldsymbol{x}})))] = \hat{y}[\tilde{\rho}_{\boldsymbol{x}}]. \qquad (27)$$

For our results, we primarily consider (26), although we show robustness for (27) in some cases.

Robust points (26) are related but not equivalent to (density operator) fixed points of a quantum channel and can be considered an application-specific generalization of fixed points. In Sec. III B, we characterize the set of robust points for example channels, and in Sec. III E we use this connection to prove the existence of robust data encodings.

For classification, we are concerned with not just one data point, but rather a set of points [e.g., the set $\mathcal{X}$ or training set (Sec. II A)]. We therefore define the set of robust points, or robust set, in the following natural way.

Definition 13 (Rrobust set). Consider a (binary) quantum classifier with encoding $E : \mathcal{X} \to \mathcal{D}_n$ and decision rule $\hat{y}$ as defined in (5). Let $\mathcal{E}$ be a quantum channel. The set of robust points, or simply *robust set*, is

$$\mathcal{R}(\mathcal{E}, E, \hat{y}) := \{\boldsymbol{x} \in \mathcal{X} : \hat{y}[\mathcal{E}(\tilde{\rho}_{\boldsymbol{x}})] = \hat{y}[\tilde{\rho}_{\boldsymbol{x}}]\}, \qquad (28)$$

where $\tilde{\rho}_{\boldsymbol{x}}$ is the processed state via (9) and $\rho_{\boldsymbol{x}} = E(\boldsymbol{x})$.

While the robust set generally depends on the encoding E, there are cases in which $\mathcal{R}$ is independent of E. In this scenario, we say all encodings are robust to this channel. Otherwise, the size of the robust set (i.e., number of robust points) can vary based on the encoding, and we distinguish between two cases. If the robust set is the set of all possible

points, we say that the encoding is *completely robust* to the given noise channel.

Definition 14 (Ccompletely robust data encoding).
Consider a (binary) quantum classifier with encoding E and decision rule $\hat{y}$ as defined in (5). Let $\boldsymbol{x} \in \mathcal{X}$ and let $\mathcal{E}$ be a quantum channel. We say that E is a *completely robust data encoding* for the quantum classifier if and only if

$$\mathcal{R}(\mathcal{E}, E, \hat{y}) = \mathcal{X}. \qquad (29)$$

We note that in practice (e.g., for numerical results), complete robustness is determined relative to the training set (Sec. II A). That is, we empirically observe that E is a completely robust data encoding if and only if

$$\mathcal{R}(\mathcal{E}, E, \hat{y}) = \{\boldsymbol{x}_i\}_{i=1}^M. \qquad (30)$$

Complete robustness can be a strong condition, so we also consider a partially robust data encoding, defined as follows.

Definition 15 (Ppartially robust data encoding). Consider a (binary) quantum classifier with encoding E and decision rule $\hat{y}$ as defined in (5). Let $\boldsymbol{x} \in \mathcal{X}$ and let $\mathcal{E}$ be a quantum channel. We say that E is a *partially robust data encoding* for the quantum classifier if and only if

$$\mathcal{R}(\mathcal{E}, E, \hat{y}) \subsetneq \mathcal{X}. \qquad (31)$$

Similar to complete robustness, partial robustness is determined in practice relative to the training set. For $0 \leqslant \delta \leqslant 1$, we say that E is a δ-robust data encoding if and only if

$$|\mathcal{R}(\mathcal{E}, E, \hat{y})| = \delta M, \qquad (32)$$

where $|\cdot|$ denotes cardinality so that $|\mathcal{R}(\mathcal{E}, E, \hat{y})| \in [M]$.

III. ANALYTIC RESULTS

Using the definitions from Sec. II, we now state and prove results about data encodings. First, we show that different encodings lead to different classes of decision boundaries in Sec. III A. Next, we characterize the set of robust points for example quantum channels in Sec. III B. In Sec. III C, we prove several robustness results for different quantum channels, and in Sec. III E we discuss the existence of robust encodings as well as an observed tradeoff between learnability and robustness. Finally, in Sec. III F, we prove an upper bound on the number of robust points in terms of fidelities between noisy and noiseless states.

A. Classes of learnable decision boundaries

We defined several different encodings in Sec. II B and discussed differences in the state preparation circuits which realize the encodings. Here, we show that different encodings lead to different sets of decision boundaries for the quantum classifier, thereby demonstrating that the success of the quantum classifier in Def. 1 depends crucially on the data encoding (15).

The decision boundary according to the decision rule (5) is implicitly defined by

$$\text{Tr}[\Pi_0 \tilde{\rho}_{\boldsymbol{x}}] = 1/2. \qquad (33)$$

Consider a single qubit encoding (15) so

$$\rho_{\boldsymbol{x}} = \begin{bmatrix} f(x_1, x_2)^2 & f(x_1, x_2)g(x_1, x_2)^* \\ f(x_1, x_2)g(x_1, x_2) & |g(x_1, x_2)|^2 \end{bmatrix},$$

where we assumed without loss of generality that f is real valued. Let the unitary U such that $\tilde{\rho}_{\boldsymbol{x}} = U \rho_{\boldsymbol{x}} U^\dagger$ have matrix elements U_{ij}. Then, one can write the decision boundary (33) as (see Appendix A 1)

$$|U_{00}|^2 f^2 + 2\text{Re}[U_{00}^* U_{01} fg] + |U_{01}|^2 |g|^2 = 1/2, \qquad (34)$$

where we have let $f = f(x_1, x_2)$ and $g = g(x_1, x_2)$ for brevity. Equation (34) implicitly defines the decision boundary in terms of the data encoding f and g. The unitary matrix elements U_{ij} act as hyperparameters to define a hypothesis family.

Equation (34) can be solved numerically for different encodings, and we do so in Sec. IV A (Fig. 4) to visualize decision boundaries for single qubit classifiers. At present, we can proceed further analytically with a few inconsequential assumptions to simplify the equations.

For the amplitude encoding, we have $f(x_1, x_2) = x_1$ and $g(x_1, x_2) = x_2$. Suppose for simplicity that matrix elements $U_{00} \equiv a$ and $U_{01} \equiv b$ are real. Then, Eq. (34) can be written

$$(ax_1 + bx_2)^2 = 1/2, \qquad (35)$$

which defines a line $x_2 = x_2(x_1)$ with slope $-a/b$ and intercept $1/\sqrt{2}b$. Thus, a single qubit classifier in Def. 1 which uses the amplitude encoding (16) can learn decision boundaries that are straight lines.

Now consider the dense angle encoding (13) on a single qubit, for which $f(x_1, x_2) = \cos(\pi x_1)$ and $g(x_1, x_2) = e^{2\pi i x_2} \sin(\pi x_1)$. Supposing again that matrix elements $U_{00} \equiv a$ and $U_{01} \equiv b$ are real, we can write (34) as

$$\begin{aligned} &a^2 \cos^2 \pi x_1 + 2ab \cos \pi x_1 \sin \pi x_1 \cos 2\pi x_2 \\ &+ b^2 \sin^2 \pi x_1 \\ &= 1/2. \end{aligned} \qquad (36)$$

This can be rearranged to

$$\cos 2\pi x_2 = \frac{1 - 2a^2 + (2a^2 - 2b^2)\sin^2 \pi x_1}{ab \sin 2\pi x_1}, \qquad (37)$$

which defines a class of sinusoidal functions $x_2 = x_2(x_1)$. (See Sec. IV A and Fig. 4.)

The different decision boundaries defined by (35) and (37) emphasize the effect that encoding has on learnability. A realized classifier may have poor performance due to its encoding and switching the encoding may lead to better results. We note that a similar phenomenon occurs in classical machine learning—a standard example being that a dot product kernel cannot separate data on a spiral, but a Gaussian kernel can. It may not be clear *a priori* what encoding to use (similarly in classical machine learning with kernels), but different properties of the data may lead to educated guesses. We note that Lloyd *et al.* [73] consider training over hyperparameters to find good encodings, and we introduce a similar idea in Sec. IV C to find good *robust* encodings.

In Sec. IV A, we numerically evaluate decision boundaries for additional single-qubit encodings as well as two-qubit
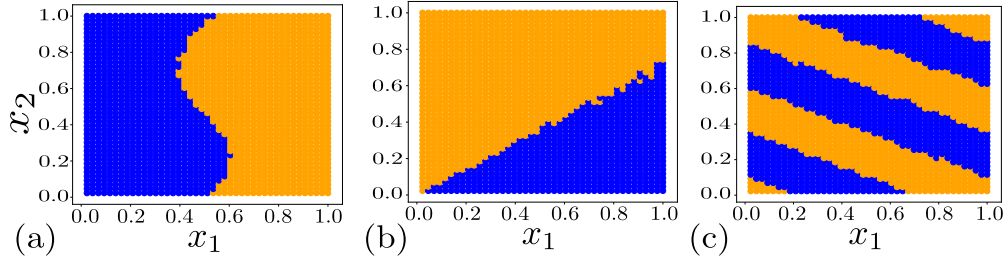
FIG. 4. Examples of learnable decision boundaries for a single qubit classifier with the (a) dense angle encoding, (b) amplitude encoding, and (c) superdense angle encoding where $\theta = \pi$ and $\phi = 2\pi$. Colors denote class labels. The QNN used here consisted of an arbitrary single qubit rotation (see Fig. 13) with random parameters.

encodings to further illustrate the differences that arise from different encodings.

B. Characterization of robust points

For a given quantum channel $\mathcal{E}$, it is a standard exercise to characterize the set of density operator fixed points, i.e., states $\rho \in \mathcal{D}_n$ such that

$$\mathcal{E}(\rho) = \rho. \tag{38}$$

In this section, we characterize the set of robust points, for example, quantum channels. This demonstrates the relationship between robust points and fixed points which we further elaborate on in Sec. III E. We remark that the characterizations similar to the ones in this Section may be of independent interest from a purely theoretical perspective, as robust points can be considered a type of generalized fixed point, or symmetry, of quantum channels.

The pure states which are fixed points of the dephasing channel (22) are $\Pi_0 := |0\rangle\langle 0|$ and $\Pi_1 := |1\rangle\langle 1|$, and

$$\rho = a\Pi_0 + b\Pi_1, \tag{39}$$

with $a + b = 1$ is the general mixed-state density operator fixed point. In contrast, let us now consider the robust points of the same dephasing channel, which satisfy

$$\hat{y}\big[\mathcal{E}_p^{\text{deph}}(\rho)\big] = \hat{y}[\rho] \tag{40}$$

instead of (38). Certainly the state in Eq. (39) will satisfy (40)—i.e., any fixed point is a robust point—but the set of robust points may contain more elements. To completely characterize the robust set, we seek the set of $\rho \in \mathcal{D}_2$ such that

$$\text{Tr}[\Pi_0\rho] \geqslant 1/2 \Rightarrow \text{Tr}\big[\Pi_0\mathcal{E}_p^{\text{deph}}(\rho)\big] \geqslant 1/2 \tag{41}$$

and

$$\text{Tr}[\Pi_0\rho] < 1/2 \Rightarrow \text{Tr}\big[\Pi_0\mathcal{E}_p^{\text{deph}}(\rho)\big] < 1/2. \tag{42}$$

Using simple properties of the trace and Pauli matrices (see Appendix A if desired), we can write

$$\text{Tr}\big[\Pi_0\mathcal{E}_p^{\text{deph}}(\rho)\big] = (1-p)\text{Tr}[\Pi_0\rho] + p\text{Tr}[\Pi_0 Z\rho Z]$$

$$= \text{Tr}[\Pi_0\rho].$$

Thus (41) and (42) are satisfied for all density operators $\rho \in \mathcal{D}_2$. That is, every data point $\boldsymbol{x} \in \mathcal{X}$ is a robust point of the dephasing channel (independent of the encoding) for the quantum classifier in Def. 1.

Consider now an amplitude damping channel (25) with $p = 1$, for which the only fixed point is the pure state Π_0. By evaluating

$$\text{Tr}\big[\Pi_0\mathcal{E}_p^{\text{AD}}(\rho)\big] = (1-p)\text{Tr}[\Pi_0\rho] + p,$$

we see that a robust point σ must satisfy $\text{Tr}[\Pi_0\sigma] = 1$. That is, the only robust point is Π_0, and in this case the set of robust points is identical to the set of fixed points.

The previous two examples illustrate how to find the robust points of a quantum channel, and the relationship between robust points and fixed points for the given channels. As expected from (26) and (38), these examples confirm that

$$\mathcal{F}(\mathcal{E}) \subseteq \mathcal{R}(E, \mathcal{E}, \hat{y}), \tag{43}$$

where $\mathcal{F}(\mathcal{E})$ denotes the set of fixed points of $\mathcal{E}$. In Sec. III E, we use this connection to generalize the above discussion and prove the existence of robust data encodings.

C. Robustness results

In this section, we state and prove results on robust encodings. We first show robustness results for single qubit classifiers with several channels in Sec. III C 1. Then, we state and prove a robustness result for multiqubit classifiers with (global) depolarizing noise in Sec. III C 2.

1. Single qubit classifier

First, we consider when robustness can be achieved for a Pauli channel.

Theorem 1. Let $\mathcal{E}_{\boldsymbol{p}}^{\text{P}}$ be a Pauli channel (19) and consider a quantum classifier on data from the set $\mathcal{X}$. Then, for any encoding $E : \mathcal{X} \to \mathcal{D}_2$, we have complete robustness

$$\mathcal{R}\big(\mathcal{E}_{\boldsymbol{p}}^{\text{P}}, E, \hat{y}\big) = \mathcal{X}$$

if $p_X + p_Y \leqslant 1/2$. (Recall that $\boldsymbol{p} = [p_I, p_X, p_Y, p_Z]$.)

Proof. The predicted label in the noisy case is identical to (5) with $\tilde{\rho}_{\boldsymbol{x}}$ replaced by $\mathcal{E}_{\boldsymbol{p}}^{\text{P}}(\tilde{\rho}_{\boldsymbol{x}})$. That is,

$$\hat{y}\big[\mathcal{E}_{\boldsymbol{p}}^{\text{P}}(\tilde{\rho}_{\boldsymbol{x}})\big] = \begin{cases} 0 & \text{if } \text{Tr}[\Pi_0\mathcal{E}_{\boldsymbol{p}}^{\text{P}}(\tilde{\rho}_{\boldsymbol{x}})] \geqslant 1/2 \\ 1 & \text{otherwise.} \end{cases}$$

By definition (19), we have

$$\text{Tr}\big[\Pi_0\mathcal{E}_{\boldsymbol{p}}^{\text{P}}(\tilde{\rho}_{\boldsymbol{x}})\big] = p_I\text{Tr}[\Pi_0\tilde{\rho}_{\boldsymbol{x}}] + p_X\text{Tr}[\Pi_0 X\tilde{\rho}_{\boldsymbol{x}}X]$$

$$+ p_Y\text{Tr}[\Pi_0 Y\tilde{\rho}_{\boldsymbol{x}}Y] + p_Z\text{Tr}[\Pi_0 Z\tilde{\rho}_{\boldsymbol{x}}Z].$$

Using straightforward substitutions (Appendix A),

we may write Sec. III C 1 as

$$\text{Tr}\big[\Pi_0 \mathcal{E}_{\mathbf{p}}^{\text{P}}(\tilde{\rho}_x)\big] = (p_I + p_Z)\text{Tr}[\Pi_0 \tilde{\rho}_x] + (p_X + p_Y)\text{Tr}[\Pi_1 \tilde{\rho}_x].$$

By resolution of the identity

$$1 = \text{Tr}[\tilde{\rho}_x] = \text{Tr}[\Pi_0 \tilde{\rho}_x] + \text{Tr}[\Pi_1 \tilde{\rho}_x],$$

we come to the simplified expression

$$\text{Tr}\big[\Pi_0 \mathcal{E}_{\mathbf{p}}^{\text{P}}(\tilde{\rho}_x)\big] = [1 - 2\nu]\text{Tr}[\Pi_0 \tilde{\rho}_x] + \nu.$$

where $\nu := p_X + p_Y$.

Suppose the noiseless classification is $\hat{y} = 0$ so $\text{Tr}[\Pi_0 \tilde{\rho}_x] \geqslant 1/2$. Since $\nu \leqslant 1/2$, we have

$$\text{Tr}\big[\Pi_0 \mathcal{E}_{\mathbf{p}}^{\text{P}}(\tilde{\rho}_x)\big] \geqslant [1 - 2\nu]\tfrac{1}{2} + \nu = \tfrac{1}{2}.$$

Hence, classification of data points with label $\hat{y} = 0$ is robust for any encoding. Suppose the noiseless classification is $\hat{y} = 1$ so $\text{Tr}[\Pi_0 \tilde{\rho}_x] < 1/2$. Since $\nu \leqslant 1/2$, we have

$$\text{Tr}\big[\Pi_0 \mathcal{E}_{\mathbf{p}}^{\text{P}}(\tilde{\rho}_x)\big] < [1 - 2\nu]\tfrac{1}{2} + \nu = \tfrac{1}{2}.$$

Hence, classification of data points with label $\hat{y} = 1$ is also robust for any encoding. ∎

Returning to the condition, $p_X + p_Y \leqslant 1/2$, one can imagine a NISQ computer in which either p_X or p_Y were large enough such that this condition is not satisfied. In this regard, we note two things. First, if this condition is *not* satisfied, then not every encoding strategy will be robust to the Pauli channel in this model. In particular, the set of robust points will now be *dependent* on the encoding strategy. This is similar to the behavior of the amplitude damping channel (which we demonstrate shortly), and we illustrate in Sec. IV.

Second, the requirement $p_X + p_Y \leqslant 1/2$ appears because the decision rule uses a measurement in the computational basis. We note that if we measure in the Hadamard basis [by replacing the projector Π_0 in the decision rule (5) with $\Pi_+ := |+\rangle\langle+|$], then we get a modified robustness condition, namely, $p_Y + p_Z \leqslant 1/2$. A similar conclusion can be drawn from measurements in the Pauli-Y basis. These results suggest that device-specific encoding strategies and decision rules may be important for achieving robustness in practice on NISQ computers.

Three robustness results for different channels can be shown as corollaries of Theorem 1 as follows. We include detailed proofs in Appendix B for completeness.

(1) By setting $p_X = p_Y = 0$ in Theorem 1, it follows that $\mathcal{R}(\mathcal{E}_p^{\text{deph}}, E, \hat{y}) = \mathcal{X}$. That is, all encodings are unconditionally robust to dephasing errors.

(2) By setting $p_Z = p_Y = 0$ and using a modified decision rule which measures in the Hadamard basis, it follows that $\mathcal{R}(\mathcal{E}_p^{\text{BF}}, E, \hat{z}) = \mathcal{X}$. That is, all encodings are unconditionally robust to bit-flip errors (using a modified decision rule).

(3) By setting $p_X = p_Y = p_Z = p/4$ and $p_I := (1 - 3p/4)$, it follows that $\mathcal{R}(\mathcal{E}_p^{\text{depo}}, E, \hat{y}) = \mathcal{X}$. That is, all encodings are unconditionally robust to depolarizing noise.

Interestingly, we remark that fact (3) holds with measurements in any basis, not just the computational basis. In Sec. III C 2, we generalize this result to multiqubit classifiers and as well as noisy data encoding (27).

We now consider amplitude damping noise, for which the robust set $\mathcal{R}$ depends on the encoding E. From the channel

definition (25), it is straightforward to see that

$$\text{Tr}\big[\Pi_0 \mathcal{E}_p^{\text{AD}}(\tilde{\rho}_x)\big] = \text{Tr}[\Pi_0 \tilde{\rho}_x] + p\text{Tr}[\Pi_1 \tilde{\rho}_x]. \quad (44)$$

Suppose first that the noiseless prediction is $\hat{y} = 0$ so $\text{Tr}[\Pi_0 \tilde{\rho}_x] \geqslant 1/2$. Then, certainly $\text{Tr}[\Pi_0 \mathcal{E}_p^{\text{AD}}(\tilde{\rho}_x)] \geqslant 1/2$ because $p \geqslant 0$ and $\text{Tr}[\Pi_1 \tilde{\rho}_x] \geqslant 0$. Thus, the noisy prediction is always identical to the noiseless prediction when the noiseless prediction is $\hat{y} = 0$. This can be understood intuitively because an amplitude damping channel models the $|1\rangle \mapsto |0\rangle$ transition [70] which only increases the probability of the ground state.

Suppose now that the noiseless prediction is $\hat{y} = 1$. From (44), we require that

$$\text{Tr}\big[\Pi_0 \mathcal{E}_p^{\text{AD}}(\tilde{\rho}_x)\big] = \text{Tr}[\Pi_0 \tilde{\rho}_x] + p\text{Tr}[\Pi_1 \tilde{\rho}_x] < 1/2$$

to achieve robustness. We use resolution of the identity

$$\text{Tr}[\Pi_1 \tilde{\rho}_x] = 1 - \text{Tr}[\Pi_0 \tilde{\rho}_x]$$

to arrive at the condition

$$\text{Tr}[\Pi_1 \tilde{\rho}_x] > \frac{1}{2(1 - p)}. \quad (45)$$

Let ρ_x be given by the general qubit encoding (15) so (45) can be written (see Appendix A 2)

$$|U_{10}|^2 f^2 + 2\text{Re}[U_{11}^* U_{10} f g^*] + |U_{11}|^2 |g|^2 > \frac{1}{2(1 - p)},$$

where U_{ij} denote the optimal unitary matrix elements.

We have thus shown the following.

Theorem 2. Consider a quantum classifier on data from the set $\mathcal{X}$, and let $\mathcal{E}_p^{\text{AD}}$ denote the amplitude damping channel (25). Then, for any qubit encoding E defined in (15) which satisfies

$$|U_{10}|^2 f^2 + 2\text{Re}[U_{11}^* U_{10} f g^*] + |U_{11}|^2 |g|^2 > \frac{1}{2(1 - p)}, \quad (46)$$

we have

$$\mathcal{R}(\mathcal{E}_p^{\text{AD}}, E, \hat{y}) = \mathcal{X}.$$

If E is not completely robust, the set of points x such that (46) holds define the partially robust set.

We note that (46) depends on the optimal unitary U as well as the encoding E. This is expected as the final state $\tilde{\rho}_x$ has been processed by the QNN. In practice, since we do not know the optimal unitary parameters *a priori*, it remains a question of how large the (partially) robust set will for a given an encoding. To address this point, we discuss in Sec. IV C how training over hyperparameters in the encoding function can help find the robust region even after application of the *a priori* unknown optimal unitary. Additionally, in the next section we discuss whether we can find an encoding which satisfies (46) or, more generally, whether a robust encoding exists for a given channel.

Given the robustness condition (46) for the amplitude damping channel, it is natural to ask whether such an encoding exists. In Sec. III E, we show the answer is yes by demonstrating there always exists a robust encoding for any trace preserving quantum operation. This encoding may be trivial, which leads to the idea of a tradeoff between learnability and robustness. (See Sec. III E.)

2. Multiqubit classifier

We now consider global depolarizing noise on a multiqubit classifier. It turns out that any encoding is completely robust to this channel applied at any point throughout the circuit. To clearly state the theorem, we introduce the following notation. First, let

$$\mathcal{E}_{p_i}(\rho) = p_i \rho + (1 - p_i) I_d / d \qquad (47)$$

be shorthand for a global depolarizing channel with probability p_i. (Note p_i and $1 - p_i$ are intentionally reversed compared to Def. 10 to simplify the proof.) Then, let

$$\tilde{\rho}_{\boldsymbol{x}}^{(m)} \equiv \left[\prod_{i=1}^{m} U_i \circ \mathcal{E}_{p_i} \right] \circ \rho_{\boldsymbol{x}} \qquad (48)$$

denote the state of the encoded point $\rho_{\boldsymbol{x}}$ after m applications of a global depolarizing channel and unitary channel. For instance, $m = 1$ corresponds to

$$U_1 \circ \mathcal{E}_{p_1} \circ \rho_{\boldsymbol{x}} \equiv U_1(\mathcal{E}_{p_1}(\rho_{\boldsymbol{x}}))$$

and $m = 2$ corresponds to

$$U_2 \circ \mathcal{E}_{p_2} \circ U_1 \circ \mathcal{E}_{p_1} \circ \rho_{\boldsymbol{x}} \equiv U_2(\mathcal{E}_{p_2}(U_1(\mathcal{E}_{p_1}(\rho_{\boldsymbol{x}})))).$$

We remark that U_i can denote any unitary in the circuit.

With this notation, we state the theorem as follows.

Theorem 3. Consider a quantum classifier on data from the set $\mathcal{X}$ with decision rule $\hat{y}$ defined in Eq. (B1). Then, for any encoding $E : \mathcal{X} \to \mathcal{D}_n$,

$$\mathcal{R}\big(\mathcal{E}_p^{\mathrm{GD}}, E, \hat{y}\big) = \mathcal{X}.$$

where $\mathcal{E}_p^{\mathrm{GD}}$ denotes the composition of global depolarizing noise acting at any point in the circuit—i.e., such that the final state of the classifier is given by (48).

To prove Theorem 3, we use the following lemma.

Lemma 1. The state in Eq. (48) can be written as (adapted from Ref. [23])

$$\tilde{\rho}_{\boldsymbol{x}}^{(m)} = \prod_{i=1}^{m} p_i U_m \cdots U_1 \rho_{\boldsymbol{x}} U_1^{\dagger} \cdots U_m^{\dagger} + \left(1 - \prod_{i=1}^{m} p_i \right) \frac{I_d}{d}, \quad (49)$$

where $d = 2^n$ is the dimension of the Hilbert space.

Proof. Using the definition of the global depolarizing channel (47), it is straightforward to evaluate

$$\tilde{\rho}_{\boldsymbol{x}}^{(1)} = U_1 \circ \mathcal{E}_{p_1} \circ \rho_{\boldsymbol{x}} = p_1 U_1 \rho_{\boldsymbol{x}} U_1^{\dagger} + (1 - p_1) I_d / d.$$

Thus (49) is true for $m = 1$. Assume (49) holds for $m = k$. Then, for $k + 1$ we have

$$\tilde{\rho}_{\boldsymbol{x}}^{(k+1)} = U_{k+1} \circ \mathcal{E}_{p_{k+1}} \circ \tilde{\rho}_{\boldsymbol{x}}^{(k)}$$
$$= p_{k+1} U_{k+1} \tilde{\rho}_{\boldsymbol{x}}^{(k)} U_{k+1}^{\dagger} + (1 - p_{k+1}) I_d / d.$$

The last line can be simplified to arrive at

$$\tilde{\rho}_{\boldsymbol{x}}^{(k+1)} = \prod_{i=1}^{k+1} p_i U_{k+1} \cdots U_1 \rho_{\boldsymbol{x}} U_1^{\dagger} \cdots U_{k+1}^{\dagger}$$
$$+ \left(1 - \prod_{i=1}^{k+1} p_i \right) I / d,$$

which completes the proof. ∎

We can now prove Theorem 3 as follows. Let l denote the total number of alternating unitary gates with depolarizing noise in the classifier circuit so (49) can be written

$$\tilde{\rho}_{\boldsymbol{x}}^{(l)} = \bar{p} \tilde{\rho}_{\boldsymbol{x}} + (1 - \bar{p}) I / d. \qquad (50)$$

Here, we have let $\bar{p} := \prod_{i=1}^{l} p_i$ and noted that $U_l \cdots U_1 \rho_{\boldsymbol{x}} U_1^{\dagger} \cdots U_l^{\dagger} = \tilde{\rho}_{\boldsymbol{x}}$ is the final state of the noiseless circuit before measuring. Eq. (50) is thus the final state of the noisy circuit before measuring. We can now evaluate

$$\mathrm{Tr}\big[\Pi_0 \tilde{\rho}_{\boldsymbol{x}}^{(l)}\big] = \bar{p} \mathrm{Tr}[\Pi_0 \tilde{\rho}_{\boldsymbol{x}}] + (1 - \bar{p})/2, \qquad (51)$$

where we have used $\mathrm{Tr}[\Pi_0 I_d] = 2^{d-1}$. To prove robustness, suppose that $\hat{y}[\tilde{\rho}_{\boldsymbol{x}}] = 0$ so $\mathrm{Tr}[\Pi_0 \tilde{\rho}_{\boldsymbol{x}}] \geqslant 1/2$. Then,

$$\mathrm{Tr}\big[\Pi_0 \tilde{\rho}_{\boldsymbol{x}}^{(l)}\big] \geqslant \bar{p}/2 + (1 - \bar{p})/2 = 1/2,$$

so $\hat{y}[\tilde{\rho}_{\boldsymbol{x}}^{(l)}] = 0$, similarly for the case $\hat{y}[\tilde{\rho}_{\boldsymbol{x}}] = 1$, which completes the proof of Theorem 3. ∎

Thus, any encoding strategy exhibits complete robustness to global depolarizing noise. We remark again that our definition of robustness (Def. 12) is in terms of probabilities, meaning that more measurements for sampling may be required to reliably evaluate robustness. With this remark, we note an interesting connection to explain a phenomenon observed in recent literature: In Ref. [35], the authors found that classification accuracy decreased under the presence of depolarizing noise. Theorem 3 implies this was exclusively due to the finite shot noise used to obtain the predicted label. We discuss errors due to finite sampling in more detail in Sec. III D.

While global depolarizing noise admits a clean robustness result for an arbitrary d-dimensional circuit, general channels can lead to complicated equations which are best handled numerically. We include several numerical results in Sec. IV, and we discuss avenues for proving more analytical results with certain classes of channels in future work in Sec. V. To close the present discussion, we highlight the special case of multiqubit classifiers with "factorizable noise," for which it is straightforward to apply previous results proved in this section.

In particular, suppose that $\mathcal{E} : \mathcal{D}_n \to \mathcal{D}_n$ is a noise channel which factorizes into single qubit channels, e.g.,

$$\mathcal{E} = \mathcal{E}_1 \otimes \cdots \otimes \mathcal{E}_n, \qquad (52)$$

where $\mathcal{E}_i : \mathcal{D}_2 \to \mathcal{D}_2$ for $i \in [n]$. Without loss of generality, let the classification qubit be the nth qubit. Then, if the processed state of the classification qubit is robust to the channel $\mathcal{E}_n$, the encoded state will be robust to the entire channel $\mathcal{E}$ in (52). This result, which is precisely stated and proved in Appendix B 2, also holds for general $n - 1$ qubit channels which act on every qubit except the classification qubit. Although this is relatively straightforward, the result could be used as a building block to better understand more intricate robustness properties of quantum classifiers.

D. Modifications for finite sampling

As mentioned in Sec. II A, the decision rule (5) is written in terms of probabilities. In practical applications, we do not have access to probabilities but rather samples from the underlying distribution. Specifically, we sample a finite number

N times from the state and get an estimate of the probabilities in terms of recorded frequencies. In this section, we discuss implications of this finite sampling error and show how to modify results from the previous sections to account for finite sampling.

The first modification is to the decision rule (5). Here, we replace the true probability $p_x := \text{Tr}[\Pi_0^c \tilde{\rho}_x]$ with an unbiased estimator $\bar{p}_x(N)$ obtained from N samples. The modified decision rule is then

$$\hat{y}_N[\tilde{\rho}_x] = \begin{cases} 0 & \text{if } \bar{p}_x(N) \geqslant 1/2 \\ 1 & \text{otherwise.} \end{cases} \tag{53}$$

Suppose we find that $\bar{p}_x(N) \geqslant 1/2$. Then, for any $\epsilon > 0$, we know that $p_x \geqslant 1/2 - \epsilon$ with probability at least $1 - \gamma$ if we take

$$N \geqslant N_c(\epsilon, \gamma) := 1/(2\epsilon^2) \ln(1/\gamma) \tag{54}$$

samples from p_x. This follows by Hoeffding's inequality,

$$\Pr\left(|\bar{p}_x(N) - p_x| \leqslant \epsilon\right) \geqslant 1 - \gamma, \tag{55}$$

which we can use since each measurement of the observable gives a Bernoulli random variable with expectation $\text{Tr}[\Pi_0^c \tilde{\rho}_x]$. We refer to $1 - \gamma$ as the confidence.

Before discussing robustness with finite sampling, we note that it is in principle #P-hard to exactly determine the predicted class label since points could be within any $\epsilon > 0$ from the decision boundary. (In other words, there is no promised gap.) In practice, we do not expect this to be a limiting issue aside from pathological examples but it is important to note. One could introduce a penalty term into the cost function such that the distance ϵ between the boundary and the closest point is maximized (as in support vector machines) to mitigate this effect.

Let us now modify the definition of a robust point to account for errors due to finite sampling. As discussed, finite sampling introduces uncertainty into the decision rule of the classifier which carries through to robust points in the following natural definition.

Definition 16 (ϵ, γ-robust point). Let $0 < \epsilon < 1/2$, $0 < \gamma < 1$, and $\mathcal{E}$ be a quantum channel. We say that the state $\rho_x \in \mathcal{D}_n$ encoding a data point $x \in \mathcal{X}$ is an ϵ, γ-robust point of the quantum classifier if and only if

$$\hat{y}_N[\mathcal{E}(\tilde{\rho}_x)] = \hat{y}_N[\tilde{\rho}_x], \tag{56}$$

with confidence $1 - \gamma$. Explicitly, (56) gives two conditions:

$$\text{Tr}[\Pi_0^c \tilde{\rho}_x] \geqslant 1/2 - \epsilon \Rightarrow \text{Tr}[\Pi_0^c \mathcal{E}(\tilde{\rho}_x)] \geqslant 1/2 - \epsilon \tag{57}$$

with confidence $1 - \gamma$ and

$$\text{Tr}[\Pi_0^c \tilde{\rho}_x] < 1/2 + \epsilon \Rightarrow \text{Tr}[\Pi_0^c \mathcal{E}(\tilde{\rho}_x)] < 1/2 + \epsilon \tag{58}$$

with confidence $1 - \gamma$.

This is achieved using $N \geqslant N_c(\epsilon, \gamma) := 1/(2\epsilon^2) \log 1/\gamma$ measurements in the decision rule (53).

Procedurally, we can think of this definition as follows. We are provided with an $0 < \epsilon < 1/2$ which corresponds to how close to the boundary we want to be able to distinguish robust points (the farthest we can be from the boundary is $1/2$), and a $0 < \gamma < 1$ which corresponds to how certain we want to be

about points within this region. Every point σ which satisfies

$$|\text{Tr}[\sigma \Pi_0^c] - 1/2| < \epsilon \tag{59}$$

is classified correctly with confidence $1 - \gamma$ using at least $N_c(\epsilon, \gamma)$ measurements. Here, σ could be either $\tilde{\rho}_x$ or $\mathcal{E}(\tilde{\rho}_x)$ as in (56).

We can incorporate finite sampling considerations into robustness results in the following natural way. For example, consider Corollary (3) of Theorem 1 which states that a single qubit classifier is unconditionally robust to depolarizing noise. The modified statement of this theorem which accounts for finite sampling is as follows.

Theorem 4. Let $\mathcal{E}_p^{\text{depo}}$ be a single qubit depolarizing channel and consider a quantum classifier on the set $\mathcal{X}$. For any encoding $E : \mathcal{X} \to \mathcal{D}_2$, let $x \in \mathcal{X}$ and ϵ, γ be such that

$$\bar{p}_x(N(\epsilon, \gamma)) \geqslant 1/2. \tag{60}$$

Then, $\rho_x = E(x)$ is an ϵ, γ-robust point.

Proof. Given $x \in \mathcal{X}$ and ϵ, γ such that

$$\bar{p}_x(N(\epsilon, \gamma)) \geqslant 1/2, \tag{61}$$

we have $\hat{y}_N[\tilde{\rho}_x] = 0$ with confidence $1 - \gamma$. This means that

$$\text{Tr}[\Pi_0^c \tilde{\rho}_x] \geqslant 1/2 - \epsilon \tag{62}$$

with confidence $1 - \gamma$. It follows that

$$\begin{aligned} \text{Tr}[\Pi_0^c \mathcal{E}_p^{\text{depo}}(\tilde{\rho}_x)] &= p/2 + (1 - p)\text{Tr}[\Pi_0^c \tilde{\rho}_x] \\ &\geqslant p/2 + (1 - p)(1/2 - \epsilon) \\ &\geqslant 1/2 - \epsilon \end{aligned}$$

with confidence $1 - \gamma$. The analogous result can be shown when $\bar{p}_x(N(\epsilon, \gamma)) < 1/2$ which completes the proof. ∎

While the results outside of this section are stated in terms of probabilities (i.e., without finite sampling), they can each be modified to incorporate effects due to finite sampling in the same manner as the above proof.

E. Existence of robust encodings

In Sec. III B, we considered example channels and characterized their robust points and fixed points. We found that the set of fixed points $\mathcal{F}(\mathcal{E})$ is always a subset of the robust set $\mathcal{R}(\mathcal{E}, E, \hat{y})$ in (43). Here, we use this connection to show that there always exists a robust encoding for a trace-preserving channel $\mathcal{E}$ [regardless of optimal unitary parameters which may appear in the robustness condition, e.g. (46)].

Theorem 5. Existence of fixed points [68,74] . Any trace-preserving quantum operation has at least one density operator fixed point (38).

Using this and the observation that $\mathcal{F}(\mathcal{E}) \subset \mathcal{R}(\mathcal{E}, E, \hat{y})$, we have the following existence theorem for robust encodings.

Theorem 6. Given a data point $x \in \mathcal{X}$, a trace-preserving quantum channel $\mathcal{E}$, and decision rule $\hat{y}$ defined in (5), there exists an encoding E such that

$$\hat{y}[\mathcal{E}(E(x))] = \hat{y}[E(x)]. \tag{63}$$

We note that the optimal unitary of the QNN affects the location of the robust set but not the existence. We emphasize that Theorem 6 is with respect to a single data point $x \in \mathcal{X}$.

As mentioned in Sec. II D, it is more relevant for applications to consider the training set (Sec. II A) or entire set $\mathcal{X}$.

Appropriately, one can ask whether a completely robust encoding (Def. 14) exists for a given channel $\mathcal{E}$. This answer also turns out to be yes, but in a potentially trivial way.

In particular, suppose that there is a unique fixed point σ of the channel $\mathcal{E}$, e.g., depolarizing noise or amplitude damping noise with $p = 1$. Then, consider the encoding

$$\mathcal{E}(\boldsymbol{x}) = \sigma$$

for all $\boldsymbol{x} \in \mathcal{X}$. From a robustness perspective, this has the desirable property of complete robustness. From a machine learning perspective, however, this has very few desirable properties: all training data is mapped to the same point so that it is impossible to successfully train a classifier.[4]

The previous example, while extreme, serves to illustrate the trade-off between learnability (expressive power) and robustness. By expressive power, we mean the ability of the classifier (hypothesis family) to predict correct labels without regard to noise. By robustness, we mean the property of retaining the same label (without regard to correctness) in the presence of noise. These two properties can be schematically connected as below:

$$y[\boldsymbol{x}] \quad \overset{\text{Learnability}}{\Longleftrightarrow} \quad \hat{y}[\tilde{\rho}_{\boldsymbol{x}}] \quad \overset{\text{Robustness}}{\Longleftrightarrow} \quad \hat{y}[\mathcal{E}(\tilde{\rho}_{\boldsymbol{x}})].$$

The trade-off we observe is that it is possible to maintain robustness by means of a certain encodings, but these encodings generally reduce the expressive power of the classifier. Succinctly, more robustness leads to less expressive power, and *vice versa*. We discuss this point more in Sec. IV C.

F. Lower bounds on partial robustness

In this section, we consider a slightly modified binary quantum classifier which embeds the cost function in the circuit and computes the cost by measuring expectation values. In contrast to the classifier in Def. 1, the output of this circuit is thus the cost C instead of an individual predicted label $\hat{y}$. Correspondingly, the input to the circuit is all data points in the training set (II A) (using a "mixed state encoding" discussed below) instead of a single data point $\boldsymbol{x}$. Such a classifier was recently introduced by Cao *et al.* in Ref. [67] and presents an interesting framework to analyze in the context of noise. In the remainder of this section, we prove a lower bound on the size of the robust set in terms of fidelities between noisy and noiseless states.

Before precisely stating this theorem, we formally define the mixed state encoding and cost function of this modified classifier.

Definition 17 (mixed state encoding). Let $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{M}$ be a data set and E be an encoding. For each feature vector $\boldsymbol{x}_i$, let

$$\sigma_{\boldsymbol{x}_i} := E(\boldsymbol{x}_i) \otimes |y_i\rangle\langle y_i| = \rho_{\boldsymbol{x}_i} \otimes |y_i\rangle\langle y_i|. \tag{64}$$

---

[4]In principle, one can achieve an encoding which is completely robust and able to correctly classify all data if there are at least two orthogonal fixed points in $\mathcal{F}(\mathcal{E})$. For example, if $\mathcal{E}$ the bit flip channel, the encoding $\boldsymbol{x} \mapsto |0\rangle + (-1)^{y_i}|1\rangle$ is both completely robust and completely learnable (the optimal unitary is a Hadamard gate) but assumes the true labels y_i are known.

The mixed state encoding is then defined by

$$\sigma := \frac{1}{M} \sum_{i=1}^{M} \sigma_{\boldsymbol{x}_i}. \tag{65}$$

We note that (65) may be realized by preparing one of the pure states (64) with equal probability. The cost function we use for this classifier is the indicator cost function given by

$$C := \frac{1}{M} \sum_{i=1}^{M} \mathbb{1}(\hat{y}_i(\tilde{\rho}_{\boldsymbol{x}_i}) \neq y_i). \tag{66}$$

Here, the indicator $\mathbb{1}$ evaluates to the truth value of its argument—i.e., $\mathbb{1}(\hat{y}_i \neq y_i) = 0$ if $y_i = \hat{y}_i$, otherwise 1. We note again that $C = C(\boldsymbol{\alpha})$ is parameterized by some angles $\boldsymbol{\alpha}$ which we omit for brevity. We now state and prove the following theorem which provides an upper bound on the size of the robust set.

Theorem 7. Consider a quantum classifier using the mixed state encoding (65) and indicator cost function (66). Assuming that a noise channel $\mathcal{E}$ acts only on the encoded feature vectors $\rho_{\boldsymbol{x}_i}$ (and not the encoded labels $|y_i\rangle\langle y_i|$), then

$$|\mathcal{R}(\mathcal{E}, E, \hat{y})| \geqslant M - \sqrt{1 - F(\mathcal{E}(\tilde{\sigma}), \tilde{\sigma})}, \tag{67}$$

where F is the fidelity

$$F(\tau, \omega) := \text{Tr}[\sqrt{\sqrt{\tau}\omega\sqrt{\tau}}]^2 \ \forall \ \tau, \omega \in \mathcal{D}_n.$$

This theorem is useful since, if one has an expression for the output fidelity of a quantum circuit after a noise channel is applied, one can directly determine a bound on how the classification results will be affected. To prove it, we relate the difference between the noisy cost and the noiseless cost to the size of the robust set.

Proof. Let the indicator cost (66) of the noisy classifier be $C_{\mathcal{E}}$ and the noiseless classifier be C. Define

$$\Delta_{\mathcal{E}} C := |C_{\mathcal{E}} - C| \tag{68}$$

to be the magnitude of the difference between noisy and noiseless costs. Due to our choice of cost function (66), we can relate $\Delta_{\mathcal{E}} C$ to the δ-robustness of the model in Def. 15 as follows.

First, it is easy to see that the quantity $1 - \Delta_{\mathcal{E}} C$ is the the fraction δ of robust points (32) so

$$|\mathcal{R}(\mathcal{E}, E, \hat{y})| = M(1 - \Delta_{\mathcal{E}} C). \tag{69}$$

Now the QNN acts only on the "data subsystem" of the mixed state encoding (65) so the evolved state before measurement is

$$\tilde{\sigma} = \frac{1}{M} \sum_{i=1}^{M} \tilde{\rho}_{\boldsymbol{x}_i} \otimes |y_i\rangle\langle y_i|. \tag{70}$$

By assumption, the channel $\mathcal{E}$ leaves the "label subsystem" $|y_i\rangle\langle y_i|$ invariant so

$$\mathcal{E}(\tilde{\sigma}) = \frac{1}{M} \sum_{i=i}^{M} \mathcal{E}(\tilde{\rho}_{\boldsymbol{x}_i}) \otimes |y_i\rangle\langle y_i|.$$

Now, a related cost C' can be evaluated by measuring the expectation of

$$D := I^{\otimes n-1} \otimes Z_c \otimes Z_l,$$

where c and l denote classification and label qubits, respectively [67]. That is, the noiseless cost is given by

$$C' = \text{Tr}(D\tilde{\sigma})$$

and the noisy cost is given by

$$C'_{\mathcal{E}} = \text{Tr}(D\mathcal{E}(\tilde{\sigma})).$$

Now, it is easy to see that $0 \leqslant \Delta_{\mathcal{E}}C' \leqslant 2$. When all points are robust, $\Delta_{\mathcal{E}}C' = 0$, and when a single point goes from robust to not robust, $\Delta_{\mathcal{E}}C'$ increases by at most $2/M$. As such, we can write down a corresponding inequality for the cost, C', as in (69) so that

$$|\mathcal{R}(\mathcal{E}, E, \hat{y})| \geqslant M\left(1 - \frac{\Delta_{\mathcal{E}}C'}{2}\right). \tag{71}$$

The difference in cost (68) due to noise is thus

$$\begin{aligned}
\Delta_{\mathcal{E}}C' &:= |C'_{\mathcal{E}} - C'|, \\
&= |\text{Tr}[D(\mathcal{E}(\tilde{\sigma}) - \tilde{\sigma})]|, \\
&\leqslant ||D||_{\infty} ||\mathcal{E}(\tilde{\sigma}) - \tilde{\sigma}||_1, \\
&\leqslant 2\sqrt{1 - F(\mathcal{E}(\tilde{\sigma}), \tilde{\sigma})}. \tag{72}
\end{aligned}$$

The third line in this derivation follows from Hölders inequality and the last line from the Fuchs-van de Graaf inequality [75,76]. We also used the fact that $||D||_{\infty} := \max_j |\lambda_j(D)| = 1$.

Substituting the inequality (71) in (72) completes the proof. ∎

We can also show the following tighter bound based on the average trace distance between the individual encoded states (i.e., not using the mixed state encoding):

$$\Delta_{\mathcal{E}}C' \leqslant \frac{2}{M} \sum_{i=1}^{M} \sqrt{1 - F(\mathcal{E}(\tilde{\rho}_{x_i}), \tilde{\rho}_{x_i})}. \tag{73}$$

A proof is included in Appendix B.

In Sec. IV D, we use these inequalities to bound the size of the robust set for several different encodings on an example implementation.

IV. NUMERICAL RESULTS

In this section, we present numerical evidence to reinforce the theoretical results proved in Sec. III and build on the discussions. In Sec. IV A, we show classes of learnable decision boundaries for example encodings, building on the previous discussion in Sec. III A. We then plot the robust sets for partially robust encodings in Sec. IV B to visualize the differences that arise from different encodings. We also generalize some encodings defined in Sec. II B to include hyperparameters and study the effects. This leads us to attempt to train over these hyperparameters, and we present an encoding learning algorithm in Sec. IV C to perform this task. Finally, in Sec. IV D we compute upper bounds on the size of robust sets

based on Sec. III F. We note that we include code to reproduce all results in this section in Ref. [77]. For all numerical results in the following sections related to single qubit classifier, we use three simple data sets; the first is the moons data set from scikit-learn [78] and two we denote vertical and diagonal. Representative examples can be found in Appendix C.

A. Decision boundaries and implementations

In Sec. II B, we defined an encoding (7) and gave several examples. In Sec. III A, we showed that a classifier with the amplitude encoding (17) can learn decision boundaries that are straight lines, while the same classifier with the dense angle encoding (14) can learn sinusoidal decision boundaries. We show this in Fig. 4, and we build on this discussion in the remainder of this section.

Figure 4(c) shows a striped decision boundary which was learned by a superdense angle encoding, defined below. The superdense encoding introduces a linear combination of features into the qubit (angle) encoding (12).

Definition 18 (Ssuperdense angle encoding) Let $x = [x_1, ..., x_N]^T \in \mathbb{R}^N$ be a feature vector and $\theta, \phi \in \mathbb{R}^N$ be parameters. Then, the superdense angle encoding maps $x \mapsto E(x)$ given by

$$|x\rangle = \bigotimes_{i=1}^{\lceil N/2 \rceil} \cos(\theta_i x_{2i-1} + \phi_i x_{2i})|0\rangle + \cos(\theta_i x_{2i-1} + \phi_i x_{2i})|1\rangle. \tag{74}$$

For a single qubit, the superdense angle encoding (SDAE) is

$$|x\rangle := \cos(\theta x_1 + \phi x_2)|0\rangle + \sin(\theta x_1 + \phi x_2)|1\rangle.$$

We observe that $\phi = 0$ recovers the qubit (angle) encoding (12) considered by [34,66,67] and (74) encodes two features per qubit.

We note that Def. 18 includes hyperparameters θ and ϕ. The reason for this will become clear in Sec. IV C when we consider optimizing over encoding hyperparameters to increase robustness. As previously mentioned, a similar idea was investigated by Lloyd *et al.* in Ref. [73] for the purpose of (in our notation) learnability.

As a final example to explore the importance of encodings, we consider an example implementation on a standard data set using different encodings. The data set we consider is the Iris flower data set [79] in which each flower is described by four features so $x \in \mathbb{R}^4$. The original data set includes three classes (species of flower) but we only consider two for binary classification. A quantum classifier using the qubit angle encoding (12) and a tree tensor network (TTN) Ansatz was considered in Ref. [35]. Using this encoding and QNN, the authors were able to successfully classify all points in the data set.

Since the angle encoding maps one feature into one qubit, a total of four qubits was used for the example in Ref. [35]. Here, we consider encodings which map two features into one qubit and thus require only two qubits. Descriptions of the encodings, QNN Ansätze, and overall classification accuracy are shown in Table I.

TABLE I. Classification accuracy achieved on the Iris data set using different encodings and QNNs in the quantum classifier. The top row is from Ref. [35] and the remaining rows are from this paper. The heading N_p indicates number of parameters in the QNN and n is the number of qubits in the classifier. The accuracy is the overall performance using a train-test ratio of 80% on classes 0 and 2. (See Ref. [77] for full implementation details.)

Encoding	QNN	N_P	n	Accuracy
Angle	TTN	7	4	100%
Dense angle	$U(4)$	12	2	100%
Amplitude	$U(4)$	12	2	100%
Superdense angle	$U(4)$	12	2	77.6%

As can be seen, we are able to achieve 100% accuracy using the amplitude and dense angle encoding. For the SDAE, the accuracy drops. Because the SDAE performs worse than other encodings, this implementation again highlights the importance of encoding on learnability. Additionally, the fact that we can use two qubits instead of four highlights the importance of encodings from a resource perspective. Specifically, NISQ applications with fewer qubits are less error prone due to fewer two-qubit gates, less crosstalk between qubits, and reduced readout errors. The reduction in the number of qubits here due to data encoding parallels, e.g., the reduction in the number of qubits in quantum chemistry applications due to qubit tapering [80]. For QML, such a reduction is not always beneficial as the encoding may require a significantly higher depth. For this implementation, however, the dense angle encoding has the same depth as the angle encoding, so the reduction in number of qubits is meaningful.

B. Robust sets for partially robust encodings

In Sec. III C, we proved conditions under which an encoding is robust to a given error channel. Typically, in practice, encodings may not completely satisfy such robustness criteria but will exhibit partial robustness—i.e., some number of training points will be robust but not all. In this section, we characterize such robust sets for different partially robust encodings. We emphasize two points: (i) the number of robust points is different for different encodings and (ii) the location of robust points is different for different encodings.

To illustrate the first point, we consider amplitude damping noise—which has robustness condition (46)—for two different encodings: the dense angle encoding and the amplitude encoding. For each, we use a data set which consists of 500 points in the unit square separated by a vertical decision boundary at $x_1 = 0.5$.

The results for the dense angle encoding are shown in Fig. 5 . Without noise, the classifier is able to reach an accuracy of $\sim$99% on the training set. When the amplitude damping channel with strength $p = 0.2$ is added, the test accuracy reduces to $\sim$78%. This encoding is thus partially robust, and the set of robust points is shown explicitly in Fig. 5(c).

The results for the amplitude encoding are shown in Fig. 6 . Here, the classifier is only able to reach $\sim$82% test accuracy without noise. When the same amplitude damping channel with strength $p = 0.4$ is added, the test accuracy drops to $\sim$43%. We also consider the effect of amplitude damping noise with strength $p = 0.2$ in Fig. 6, for which the classifier achieves test accuracy $\sim$61%. The robust set for both channels is also shown in Fig. 6.

C. Encoding learning algorithm

For this purpose, we introduce an encoding learning algorithm to try and search for good encodings. The goal is crudely illustrated in Fig. 7 . As mentioned above, Ref. [73] trains over hyperparameters using the re-uploading structure of Ref. [36] to increase learnability. Here, the encoding learning algorithm adapts to noise to increase robustness. We note the distinction that in our implementations we train the unitary in a noiseless environment and do not alter its parameters during the encoding learning algorithm. By training the encoding, we change E and hence also the model family. This is a desirable property for our purpose to search for robust encodings, since as we have illustrated E directly impacts the robustness of the model family. This could be incorporated alongside training the unitary itself to deal with coherent noise, for example as seen in Refs. [23,72].

The encoding learning algorithm is similar to the Rotoselect algorithm [81] which is used to learn circuit structure. For
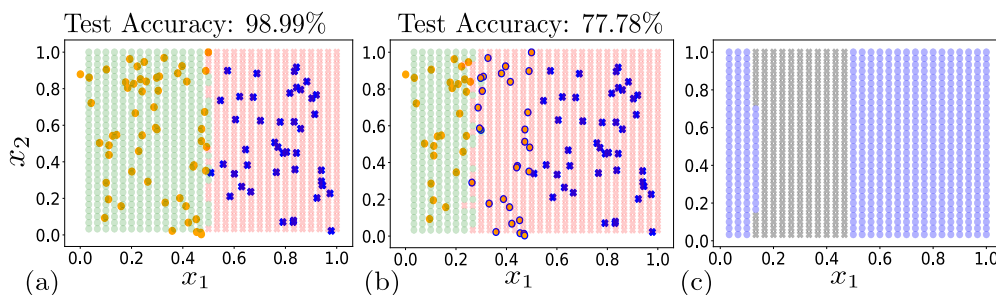

FIG. 5. Partial robustness for the dense angle encoding. The data set consists of 500 points in the unit square separated by a vertical decision boundary and we use a train-test split of 80%. Red crosses and green circles indicate decision boundary location. Orange circles and blue crosses indicate points labeled as each class. Misclassified points have opposite color border. Panel (a) shows the classifier test accuracy after optimizing the unitary without noise. Panel (b) shows the reduced accuracy after amplitude damping noise of strength $p = 0.4$ is added. The robust set is at the far left and far right of the unit square, explicitly shown in panel (c). Here, blue circles indicates the robust set and black crosses indicates its complement.
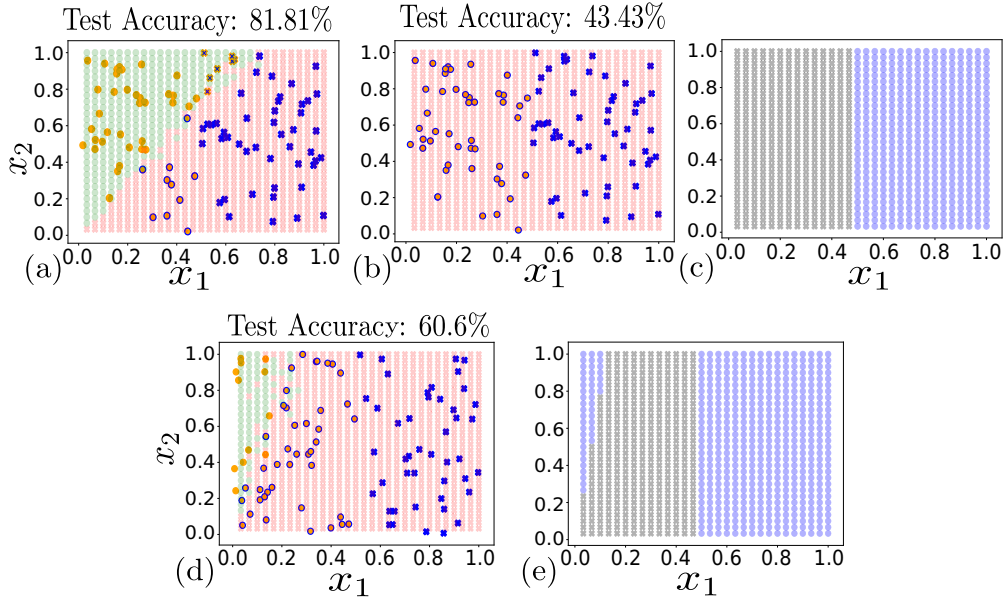
FIG. 6. Partial robustness for the amplitude encoding. The data set consists of 500 points in the unit square separated by a vertical decision boundary, and we use a train-test split of 80%. Panel (a) shows classifier test accuracy after optimizing the unitary without noise. Panel (b) shows the reduced accuracy after adding amplitude damping noise with strength $p = 0.4$. The robust set is shown explicitly in panel (c) where a blue circle indicates a robust point and a black cross indicates a misclassified point. Panel (d) is the same as (b) but with decreased strength $p = 0.2$ of the amplitude damping channel. Test accuracy reduces from 81% to 60% in this case. Panel (e) shows the robust set for (d).

each function pair (f_j, g_j) from a discrete set of parameterized functions $\{f_i(\boldsymbol{\theta}_i), g_i(\boldsymbol{\theta}_i)\}_{i=1}^K$ we train the unitary $U(\boldsymbol{\alpha})$ to minimize the cost while keeping the encoding (hyper)parameters $\boldsymbol{\theta}_j$ fixed. Next, we add a noise channel $\mathcal{E}$ which causes some points to be misclassified. Now, we optimize the encoding parameters $\boldsymbol{\theta}_j$ in the noisy environment. For this optimization, the same cost function is used, and the goal is to further decrease the cost (and hence increase the set of robust points) by varying the encoding hyperparameters. Pseudocode for the algorithm is shown in Appendix C.
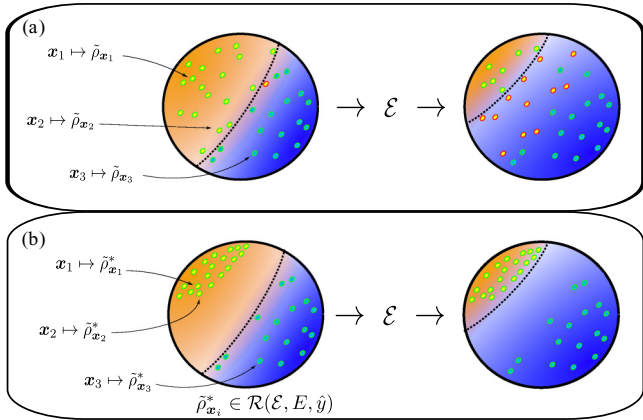

FIG. 7. Cartoon illustration of the encoding learning algorithm with a single qubit classifier. In (a), a preset encoding with no knowledge of the noise misclassifies a large number of points. In (b), the encoding learning algorithm detects misclassifications and tries to adjust points to achieve more robustness, attempting to encode into the robust set for the channel.

We test the algorithm on linearly separable and nonlinearly separable data sets in Fig. 8 . In particular, we use three different encodings on three data sets. The encodings used are the dense angle encoding, superdense angle encoding, and a specific instance of the generalized amplitude encoding from Def. 5 given by

$$|\boldsymbol{x}\rangle := \frac{\sqrt{1 + \theta x_2^2}}{||\boldsymbol{x}||_2} x_1 |0\rangle + \frac{\sqrt{1 - \theta x_1^2}}{||\boldsymbol{x}||_2} x_2 |1\rangle,$$

where θ is a free parameter which is trained over.

Using these encodings and the data sets shown in Appendix C, we study performance for the noiseless case, noisy case, and the effect of the encoding learning algorithm. We observe that the algorithm is not only capable of recovering the noiseless classification accuracy achieved, but is actually able to outperform it in some cases, as can be seen in Fig. 8.

Finally, we consider the discussion in Sec. III E about the trade-off between learnability (expressive power) and robustness. We make this quantitative in Fig. 9 by plotting accuracy (percent learned correctly) and robustness against hyperparameters θ and ϕ in a generalized dense angle encoding

$$|\boldsymbol{x}\rangle = \cos(\theta x_1)|0\rangle + e^{i\phi x_2} \sin(\theta x_1)|1\rangle.$$

More specifically, in Fig. 9, we illustrate how the noise affects the hyperparameters, θ^* and ϕ^*, which give maximal classification *accuracy* in both the noiseless and noisy environments, and also those which give maximal *robustness* (in the sense of Def. 15). Figure 9(a) shows the percentage misclassified in the noiseless environment, where red indicates the lowest accuracy on the test set, and blue indicates the highest
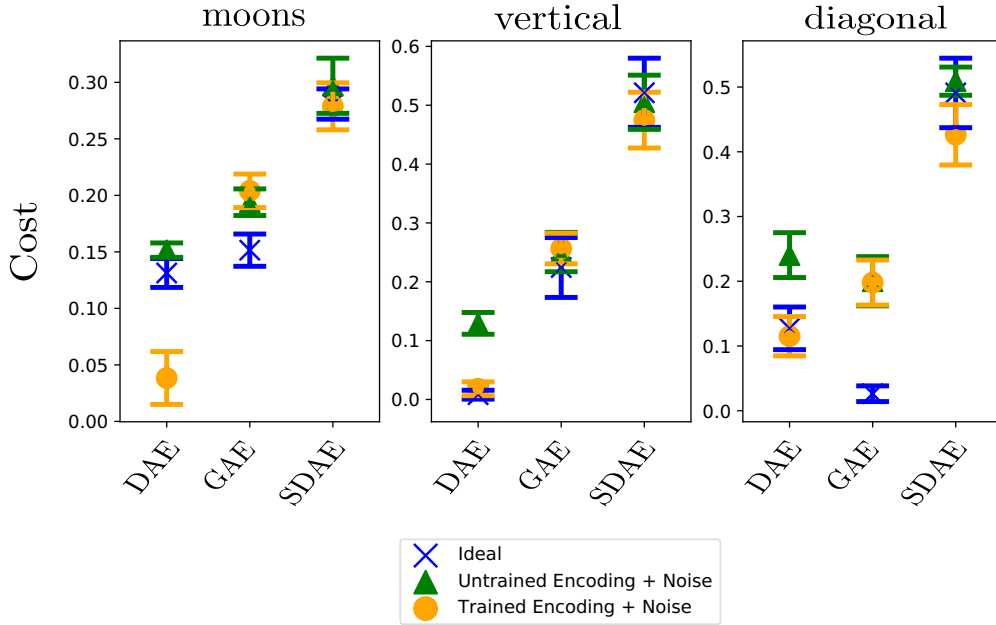
FIG. 8. Minimum cost achieved (vertical axis) from applying the encoding learning algorithm to three example data sets (each plot) using three different encodings (horizontal axis). The blue crosses [×] show the minimum cost achieved when training over only unitary parameters without any noise present (ideal). The green triangles [▲] show the same case with the addition of amplitude damping noise of strength $p = 0.3$. The orange circles [●] show minimum cost after applying the encoding learning algorithm. The dense angle encoding (DAE) is seen to perform well on all data sets and is capable of adapting well to noise, even outperforming the ideal case without noise and fixed encoding. The superdense angle encoding (SDAE) does not perform well on any shown data set since the generated decision boundary is highly nonlinear and cannot correctly classify more than half the data set. The generalized amplitude encoding (GAE) performs well on the diagonal boundary, since it generates a suitable decision boundary.

accuracy. We then repeat this in Figs. 9(b) and 9(c) to find the parameters which maximize accuracy in the presence of noise and the maximal robustness. As expected, for the amplitude damping channel, the best parameters (with noise) are closer to the fixed point of the channel (i.e., $\theta^* \to 0$ implies encoding in the $|0\rangle$ state), thereby demonstrating the trade-off between learnability and robustness.

D. Fidelity bounds on partial robustness

As a final numerical implementation, we compute the upper bounds on partial robustness proved in Sec. III F for several different encodings and error channels. The implementation we consider is the previously discussed Iris data sest classification problem using two qubits. The results are shown in Fig. 10. In this figure, each plot corresponds to a different error channel with strength varied across the horizontal axis. Each curve in the top row corresponds to the fidelity
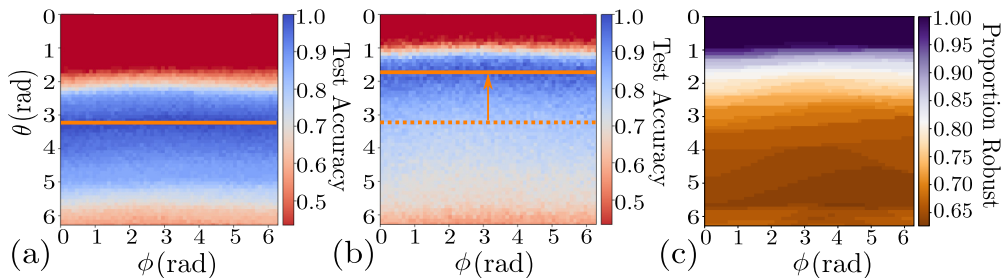

FIG. 9. Learnability versus robustness on the "vertical" data set using the parameterized dense angle encoding. The horizontal and vertical axes show the encoding hyperparameters ϕ and θ, respectively. Panels (a) and (b) show the classifier accuracy while panel (c) shows the proportion of robust points. Panel (a) shows accuracy without noise as a function of encoding parameters. Panel (b) shows accuracy with the addition of an amplitude damping channel of strength $p = 0.3$. Panel (c) shows δ robustness for different parameter values. As expected, the robust set is largest when all points are encoded into the zero state, i.e., $\theta = 0$. This leads to all points labeled 1 being misclassified, with a resulting accuracy of approximately 50%. The orange [▬] filled line indicates optimal θ parameters in each panel. From (a) to (b), the θ parameters corresponding to highest accuracy are shifted towards the robust points (i.e., toward $\theta = 0$) in (c). See Appendix C for the optimal parameters found in each case.
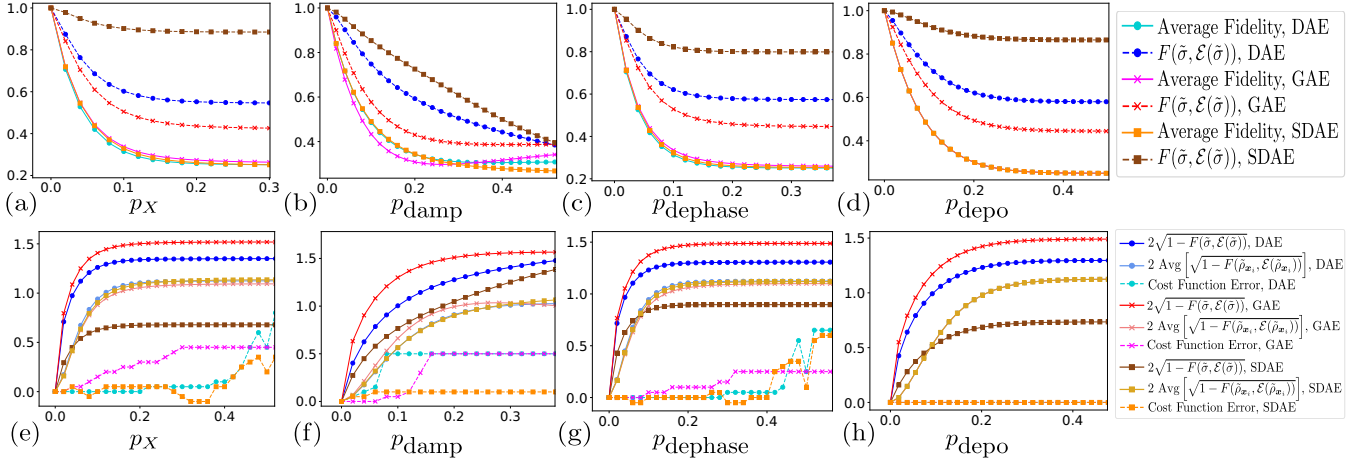
FIG. 10. Top row: Fidelity between noisy and noiseless states for different encodings on the Iris data set. Comparing average fidelity over every encoded state in data set, versus fidelity of noisy and noiseless encoded mixed states. From left to right, the noise models are bit-flip noise, amplitude damping noise, dephasing noise, and global depolarizing noise, with strengths varied across the horizontal axis. Bottom row: Upper bounds on partial robustness in terms of fidelity using the bounds (72) and (73). For each plot (in both rows), three different data encodings are considered—curves corresponds to three product qubit encodings: dense angle encoding, amplitude encoding, and superdense angle.

of noisy and noiseless states using different encodings. Each curve in the bottom row shows the upper bounds on partial robustness proved in Sec. III F.

As can be seen in the bottom row of Fig. 10, upper bounds on partial robustness are different for different encodings, particularly at small noise values. (Recall that a trivial upper bound on the size of partial robustness is one so that curves at large channel strengths above one are mostly uninformative.) For such low values of noise, they give us some information about the maximum cost function deviation we can expect. Based on the average fidelity over the datas ets, in Figs. 10(a)–10(d), all three encodings behave qualitatively the same. However, the cost function error for the three encodings is significantly different, especially for bit flip and dephasing errors, Figs. 10(f) and 10(g). As expected, a depolarizing channel causes no misclassification, as seen in Fig. 10(h), despite the decrease in fidelity of the states. The apparent erratic behavior of the cost function error is largely due to the low number of samples in the Iris data set. [Recall that the superdense angle encoding was not able to achieve perfect classification accuracy on the Iris data set, so under amplitude damping noise, e.g., the cost function error can only decrease by about 25% ($\sim$77% $\rightarrow$ 50%).] We can also observe that the dense angle encoding is less susceptible to bit flip and phase errors than the amplitude encoding in Figs. 10(f) and 10(g).

V. DISCUSSION AND CONCLUSIONS

In this paper, we have formally defined a model for a binary quantum classifier common in recent literature and studied encoding functions in detail. In particular, we showed that different encodings lead to different learnable decision boundaries and thus have an important effect on the overall

success of the classifier. We introduced and formally defined the concept of robust points as well as robust sets and (completely) robust encodings. In addition to studying the robustness criteria for several common noise models, we have characterized robust sets, for example, channels and discussed their relationship to the fixed points of the channels. We used this connection to provide an existence proof of robust encodings and discussed an empirically observed tradeoff between learnability (expressive power) and robustness. Finally, we considered an embedded cost function classifier using an indicator cost function and provided upper bounds on the robustness of an encoding in terms of fidelities between ideal and noisy evolution.

In addition to the above theoretical results, we performed several numerical implementations to confirm and extend our findings. Specifically, we numerically evaluated decision boundaries for different encodings and performed example implementations on standard datasets in machine learning. Additionally, we used numerics to show that different encodings lead to different robust sets, and quantified the size and location of such sets for different encodings. Finally, we presented an encoding learning algorithm which optimizes over hyperparameters in the encoding to attempt to find robust sets. We provided proof of principle implementations on three data sets to show the performance of this algorithm. Finally, we showed upper bounds on partial robustness by computing the fidelities between ideal and noisy final states for an example implementation. We provide code to reproduce all numerical results at Ref. [77].

Within QML, this paper adds to the growing body of work studying encoding functions in detail. Our concept of robust points and robust encodings and some of our analytical results help explain phenomena observed in recent literature—namely, that misclassifications due to depolarizing noise found in Ref. [35] are solely due to finite shot statistics.

As this paper discusses a relatively understudied area in QML, there are several avenues for future research. While we have provided multiple analytic results, the framework we introduced for proving these results is perhaps more impactful. In future work, these ideas can be applied to prove more robustness results for different channels and different decision boundaries than the one we considered in this paper. Specific tasks we leave to future work include generalizing results to classes of quantum channels (e.g., unital channels), quantifying the trade-off between learnability and robustness, and characterizing the conditions under which a *completely* robust encoding exists. It is also interesting from a purely theoretical perspective to extend the notion of a robust point to a generalized fixed point of a channel, i.e., points which satisfy $f(\mathcal{E}(\rho)) = f(\rho)$ for some function f. (When $f = \hat{y}$, the generalized fixed point is a robust point, but other arbitrary functions f could be considered.) Finally, one could consider data encodings using ideas from quantum error correction, for example, by encoding into decoherence-free subspaces [82,83].

From an applications perspective, a clear task for future work is to test the ideas introduced here on a NISQ computer. While the channels we considered are standard theoretical tools for analyzing noise, more complicated effects such as crosstalk occur in real devices. Implementations on NISQ computers would assess our results *in situ* and potentially give additional insight into how robust encodings can be designed. Also, for future numerical work one could consider additional data sets (e.g., the MNIST data set [84]) and test the performance of different encodings both in terms of learnability and robustness. Last, to further incorporate with recent literature, one could consider robustness in the context of data reuploading [36] or in the adversarial setting of Refs. [85,86].

More broadly for NISQ applications, our work introduces a problem-specific strategy for dealing with errors. In contrast to error mitigation or even error correction techniques which allow errors to occur then attempt to mitigate (correct) them, our strategy of robust data encoding attempts to set up the problem such that any errors which do occur have no effect on the final result. We exploit the natural machine-learning concept of data representation to achieve this effect, but in principle a similar idea could be used in other settings. This work defines fundamental concepts and proves several results for important questions that must be addressed on the path to practical applications with near-term quantum computers.

APPENDIX A: PRELIMINARIES AND USEFUL FORMULAE

The single qubit Pauli operators are

$$I := \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, X := \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, Y := \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, Z := \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}.$$

Let ρ be a single qubit state with matrix elements ρ_{ij}, i.e.,

$$\rho := \begin{bmatrix} \rho_{00} & \rho_{01} \\ \rho_{10} & \rho_{11} \end{bmatrix}.$$

Then

$$X \rho X = \begin{bmatrix} \rho_{11} & \rho_{10} \\ \rho_{01} & \rho_{00} \end{bmatrix},$$

$$Y \rho Y = \begin{bmatrix} \rho_{11} & -\rho_{10} \\ -\rho_{01} & \rho_{00} \end{bmatrix},$$

$$Z \rho Z = \begin{bmatrix} \rho_{00} & -\rho_{01} \\ -\rho_{10} & \rho_{11} \end{bmatrix}.$$

Defining the projectors $\Pi_0 = |0\rangle\langle 0|$ and $\Pi_1 = |1\rangle\langle 1|$, one can show

$$\mathrm{Tr}[\Pi_0 X \rho X] = \mathrm{Tr}[\Pi_1 \rho],$$
$$\mathrm{Tr}[\Pi_0 Y \rho Y] = \mathrm{Tr}[\Pi_1 \rho],$$
$$\mathrm{Tr}[\Pi_0 Z \rho Z] = \mathrm{Tr}[\Pi_0 \rho],$$

For any Hermitian matrix $A = [A_{ij}]$ and any unitary matrix $U = [U_{ij}]$, we have

$$\mathrm{Tr}[\Pi_0 U A U^\dagger]$$
$$= |U_{00}|^2 A_{00} + 2\mathrm{Re}[U_{00}^* U_{01} A_{10}] + |U_{01}|^2 A_{11}. \quad (A1)$$

Similarly, one can show that

$$\mathrm{Tr}[\Pi_1 U A U^\dagger]$$
$$= |U_{10}|^2 A_{00} + 2\mathrm{Re}[U_{11}^* U_{10} A_{01}] + |U_{11}|^2 A_{11}. \quad (A2)$$

If we further assume the single qubit parameterized unitary, $U(\boldsymbol{\alpha})$, has the following decomposition: $R_z(2\alpha_1)R_y(2\alpha_2)R_z(2\alpha_3)$ (up to a global phase) [68], we get

$$U(\boldsymbol{\alpha}) = \begin{bmatrix} e^{i(-\alpha_1-\alpha_3)}\cos\alpha_2 & -e^{i(-\alpha_1+\alpha_3)}\sin\alpha_2 \\ e^{i(\alpha_1-\alpha_3)}\sin\alpha_2 & e^{i(\alpha_1+\alpha_3)}\cos\alpha_2 \end{bmatrix}. \quad (A3)$$

Therefore, we get the various terms to be

$$|U_{00}|^2 = \cos^2(\alpha_2),$$
$$|U_{01}|^2 = |U_{10}|^2 = \sin^2(\alpha_2),$$
$$|U_{11}|^2 = \cos^2(\alpha_2),$$
$$U_{00}^* U_{01} = -e^{i2\alpha_3}\cos(\alpha_2)\sin(\alpha_2) = -\tfrac{1}{2}e^{i2\alpha_2}\sin(2\alpha_2),$$
$$U_{11}^* U_{10} = e^{-i2\alpha_3}\cos(\alpha_2)\sin(\alpha_2) = \tfrac{1}{2}e^{-2i\alpha_3}\sin(2\alpha_2).$$

So, the conditions ((A1), (A2)) become

$$\mathrm{Tr}[\Pi_0 U A U^\dagger]$$
$$= |U_{00}|^2 A_{00} + 2\mathrm{Re}[U_{00}^* U_{01} A_{10}] + |U_{01}|^2 A_{11}$$
$$= \cos^2(\alpha_2) A_{00} + \sin^2(\alpha_2) A_{11} - \mathrm{Re}[e^{2i\alpha_3}\sin(\alpha_2) A_{10}],$$

$\mathrm{Tr}[\Pi_1 U A U^\dagger]$

$$= |U_{10}|^2 A_{00} + 2\mathrm{Re}[U_{11}^* U_{10} A_{01}] + |U_{11}|^2 A_{11}$$

$$= \sin^2(\alpha_2) A_{00} + \cos^2(\alpha_2) A_{11} + \mathrm{Re}[e^{-2i\alpha_3} \sin(2\alpha_2) A_{01}].$$

APPENDIX B: PROOFS AND ADDITIONAL RESULTS

Here we give the explicit proofs for the remaining theorems (which we also repeat here for completeness) in the main text, and some others introduced here.

First, we revisit Theorem 1 in the main text and elaborate on the corollaries of it.

Theorem 8 (Theorem 1 in main text.). Let $\mathcal{E}_{\mathbf{p}}^{\mathrm{P}}$ be a Pauli channel (19) and consider a quantum classifier on data from the set $\mathcal{X}$. Then, for any encoding $E : \mathcal{X} \to \mathcal{D}_2$, we have complete robustness,

$$\mathcal{R}(\mathcal{E}_{\mathbf{p}}^{\mathrm{P}}, E, \hat{y}) = \mathcal{X},$$

if $p_X + p_Y \leqslant 1/2$. (Recall that $\mathbf{p} = [p_I, p_X, p_Y, p_Z]$.)

As discussed, by changing the decision rule to one in the Pauli-X basis, $\hat{z}$, we get the following.

Corollary 1. Consider a quantum classifier on data from the set $\mathcal{X}$ with modified decision rule

$$\hat{z}[\tilde{\rho}_{\boldsymbol{x}}] = \begin{cases} 0 & \text{if } \mathrm{Tr}[\Pi_+ \tilde{\rho}_{\boldsymbol{x}}] \geqslant 1/2 \\ 1 & \text{otherwise.} \end{cases} \quad \text{(B1)}$$

Here, $\Pi_+ := |+\rangle\langle+|$ is the projector onto the $+1$ eigenstate $|+\rangle$ of Pauli X. Then, for any $E : \mathcal{X} \to \mathcal{D}_2$,

$$\mathcal{R}(\mathcal{E}_{\mathbf{p}}^{\mathrm{P}}, E, \hat{z}) = \mathcal{X},$$

for a Pauli channel $\mathcal{E}_{\mathbf{p}}^{\mathrm{P}}$ such that $p_Y + p_Z \leqslant 1/2$.

The proof mimics that of Theorem 1. We note that the analogous statement for measurements in the Y basis also holds.

Theorem 1 also implies the following result for dephasing noise, which is a Pauli channel with $p_X = p_Y = 0$.

Theorem 9. Let $\mathcal{E}_p^{\mathrm{deph}}$ be a dephasing channel (22), and consider a quantum classifier on data from the set $\mathcal{X}$. Then, for any encoding $E : \mathcal{X} \to \mathcal{D}_2$,:

$$\mathcal{R}(\mathcal{E}_p^{\mathrm{deph}}, E, \hat{y}) = \mathcal{X}.$$

This result follows immediately from the discussion of the dephasing channel in the above Sec. III B.

Similar to Corollary 1, we can consider a modified decision rule to achieve robustness for a bit-flip channel.

Corollary 2. Consider a quantum classifier on data from the set $\mathcal{X}$ with modified decision rule $\hat{z}$ defined in Eq. (B1). Then, for any encoding $E : \mathcal{X} \to \mathcal{D}_2$:

$$\mathcal{R}(\mathcal{E}_p^{\mathrm{BF}}, E, \hat{z}) = \mathcal{X}.$$

We note that a decision rule which measures in the Y basis yields robustness to combined bit/phase-flip errors. [That is, the error channel $\mathcal{E}(\rho) = (1-p)\rho + pY\rho Y$.]

Finally, we revisit robustness for depolarizing noise (23). A simple calculation shows that

$$\mathrm{Tr}[\Pi_0 \mathcal{E}_p^{\mathrm{depo}}(\tilde{\rho}_{\boldsymbol{x}})] = p/2 + (1-p)\mathrm{Tr}[\Pi_0 \tilde{\rho}_{\boldsymbol{x}}]. \quad \text{(B2)}$$

If $\hat{y} = 0$ so $\mathrm{Tr}[\Pi_0 \tilde{\rho}_{\boldsymbol{x}}] \geqslant 1/2$, then we have that $\mathrm{Tr}[\Pi_0 \mathcal{E}_p^{\mathrm{depo}}(\tilde{\rho}_{\boldsymbol{x}})] \geqslant 1/2$. Similarly for the case $\hat{y} = 1$. As

mentioned in the main text, this formulation is equivalent to choosing $p_Z = p_X = p_Y = p/4$, $p_I = (1 - 3p/4)$ in the Pauli channel, $\mathcal{E}_{\mathbf{p}}^{\mathrm{P}}$. Thus, we have shown the following.

Theorem 10. Let $\mathcal{E}_p^{\mathrm{depo}}$ be a depolarizing channel (23), and consider a quantum classifier on data from the set $\mathcal{X}$. Then, for any encoding $E : \mathcal{X} \to \mathcal{D}_2$,

$$\mathcal{R}(\mathcal{E}_p^{\mathrm{depo}}, E, \hat{y}) = \mathcal{X}.$$

1. Robustness to measurement noise

Just as the case of quantum compilation [23], we can deal with measurement noise in the classifier.

Definition 19. *Measurement noise* is defined as a modification of the standard positive-operator valued measure (POVM) basis elements, $\{\Pi_0 = |0\rangle\langle0|, \Pi_1 = |1\rangle\langle1|\}$ by the channel $\mathcal{E}_{\mathbf{p}}^{\mathrm{meas}}$ with assignment matrix $\mathbf{p}$ for a single noiseless qubit:

$$\Pi_0 = |0\rangle\langle0| \xrightarrow{\mathcal{E}_p^{\mathrm{meas}}} \tilde{\Pi}_0 = p_{00}|0\rangle\langle0| + p_{01}|1\rangle\langle1|,$$

$$\Pi_1 = |1\rangle\langle1| \xrightarrow{\mathcal{E}_p^{\mathrm{meas}}} \tilde{\Pi}_1 = p_{10}|0\rangle\langle0| + p_{11}|1\rangle\langle1|, \quad \text{(B3)}$$

$$\mathbf{p} := \begin{pmatrix} p_{00} & p_{01} \\ p_{10} & p_{11} \end{pmatrix},$$

where $p_{00} + p_{10} = 1$, $p_{10} + p_{11} = 1$, and hence p_{kl} is the probability of getting the k outcome given the l input. Furthermore, we assume that $p_{kk} > p_{kl}$ for $k \neq l$.

The definition for the general case of n qubit measurements can be found in Ref. [23] but we shall not need it here, since we only require measuring a single qubit to determine the decision function. More general classifiers which measure multiple qubits (e.g., and then take a majority vote for the classification) could also be considered but these
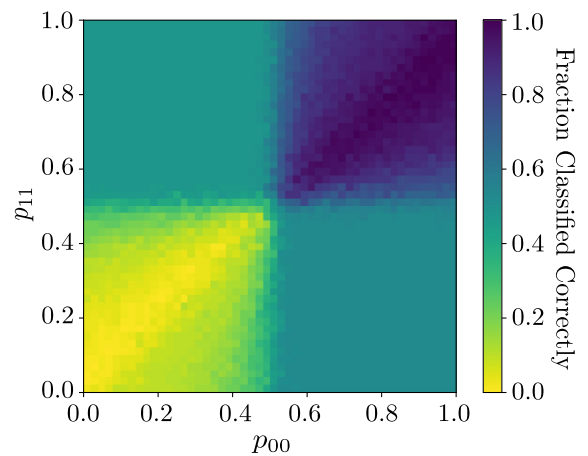
FIG. 11. Misclassification percentage as a result of (b) measurement noise as a function of probabilities $\{p_{00}, p_{11}\}$. If either p_{00} or p_{11} is less than $1/2$, then half the correctly classified points will be misclassified, with the probability increasing as expected, with the number of misclassified points increasing as the of-diagonal terms, $p_{01}, p_{10} \to 1$, as expected from Theorem 11. By classified correctly, in this context, we mean the fraction of points which are classified *the same* with and without noise.
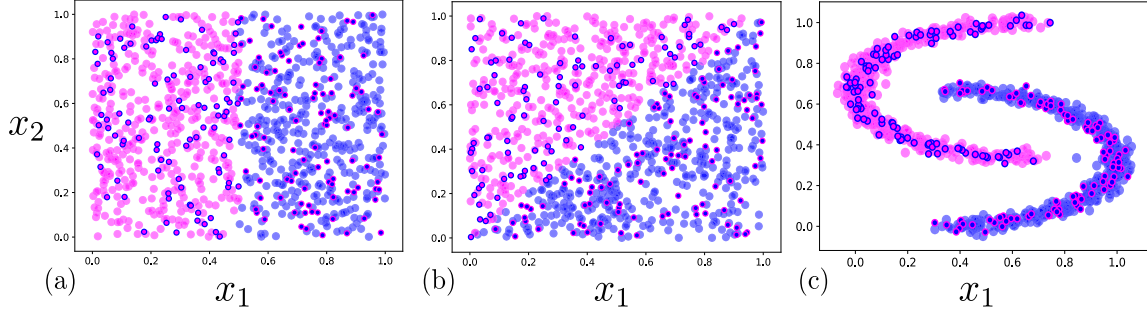
FIG. 12. Three single qubit (two dimensional) data sets which we use. (a) vertical, (b) diagonal, and (c) the moons data set from SCIKIT-LEARN [78] rotated by 90° with a noise level of 0.05. 20% of each set is test data, indicated by the points circled with the opposite color. We chose the latter two due to the fact that the moons and vertical data sets can be well classified by the dense angle encoding, while the diagonal data set can be well classified by the amplitude encoding, which can be seen by studying the decision boundaries generated in Fig. 4.

are outside the scope of this paper. Now we can show the following result in a similar fashion to the above proofs.

Theorem 11. Let $\mathcal{E}_{\boldsymbol{p}}^{\mathrm{meas}}$ define measurement noise acting on the classification qubit and consider a quantum classifier on data from the set $\mathcal{X}$. Then, for any encoding $E : \mathcal{X} \to \mathcal{D}_2$, we have complete robustness

$$\mathcal{R}(\mathcal{E}_{\boldsymbol{p}}^{\mathrm{meas}}, E, \hat{y}) = \mathcal{X}$$

if the measurement assignment probabilities satisfy $p_{00} > p_{01}$, $p_{11} > p_{10}$.

Proof. We can write the measurement noise channel acting on the POVM elements as

$$\mathcal{E}_{\boldsymbol{p}}^{\mathrm{meas}}(\Pi_0^{(c)} \otimes I^{\otimes n-1}) = (p_{00}|0\rangle\langle 0| + p_{01}|1\rangle\langle 1|) \otimes I^{\otimes n-1}.$$

Again, if we had correct classification before the noise, $\mathrm{Tr}(|0\rangle\langle 0|\tilde{\rho}_{\boldsymbol{x}}) \geqslant 1/2$, then

$$\begin{aligned}
&\mathrm{Tr}[\mathcal{E}_{\boldsymbol{p}}^{\mathrm{meas}}(\Pi_0^{(c)} \otimes I^{\otimes n-1})\tilde{\rho}_{\boldsymbol{x}}] \\
&= \mathrm{Tr}[(\{p_{00}|0\rangle\langle 0| + p_{01}|1\rangle\langle 1|\} \otimes I^{\otimes n-1})\tilde{\rho}_{\boldsymbol{x}}] \\
&= p_{00}\mathrm{Tr}[(|0\rangle\langle 0| \otimes I^{\otimes n-1})\tilde{\rho}_{\boldsymbol{x}}] + p_{01}\mathrm{Tr}[(|1\rangle\langle 1| \otimes I^{\otimes n-1})\tilde{\rho}_{\boldsymbol{x}}] \\
&= p_{00}\mathrm{Tr}[(|0\rangle\langle 0| \otimes I^{\otimes n-1})\tilde{\rho}_{\boldsymbol{x}}] \\
&\quad + p_{01}(1 - \mathrm{Tr}[(|0\rangle\langle 0| \otimes I^{\otimes n-1})\tilde{\rho}_{\boldsymbol{x}}]) \\
&= (p_{00} - p_{01})\mathrm{Tr}[(|0\rangle\langle 0| \otimes I^{\otimes n-1})\tilde{\rho}_{\boldsymbol{x}}] + p_{01} \\
&\geqslant (p_{00} - p_{01})1/2 + p_{01} = 1/2(p_{00} + p_{01}) = 1/2,
\end{aligned}$$

where in the last line, we used the fact that $p_{00} + p_{01} = 1$ and our assumption that $p_{00} > p_{01}$. The same result holds if the vector was classified as 1, and hence the classifier is robust to measurement noise. ■

Just as above, we can replace the ideal state, $\tilde{\rho}_{\boldsymbol{x}_i}$ with a noisy state, $\mathcal{E}(\tilde{\rho}_{\boldsymbol{x}_i})$, where the operator accounts for other forms of noise, not including measurement noise. We can see this allows us to take a model which is robust without measurement noise, and upgrade it to one which is. However, we may be able to find looser restrictions by considering different types of noise together rather than in this modular fashion.

To illustrate the results of Theorem 11, we focus on the dense angle encoding, which can achieve nearly 100% accuracy on the "vertical" data set. We then compute the

percentage which would be misclassified as a function the assignment probabilities in the noisy projectors in (B3). The results are seen in Fig. 11.

Algorithm 1: Quantum Encoding Learning Algorithm (QELA)

Input : Noise parameters, $\boldsymbol{p}$, parameterized quantum circuit, $U(\boldsymbol{\alpha})$, M Labelled data examples $(\boldsymbol{x}_i, y_i)_{i=1}^M$, encoding set $\{f_l, g_l\}_{l=1}^K$, cost function C.

Result: Optimized encoding for noise and dataset.

1 Initialize encoding, $(f^*, g^*) \leftarrow \{f_l, g_l\}_{l=1}^K$ and parameters, $\{\boldsymbol{\theta}_j\} \leftarrow (0, 2\pi]_j \; \forall j$ heuristically or at random. Initialize $C^* = M$;

2 **for** $j=1\ldots K$ **do**

3 $\quad$ Select subset of data: $(\boldsymbol{x}_i, y_i)_{i=1}^D \leftarrow (\boldsymbol{x}_i, y_i)_{i=1}^M$;

4 $\quad$ Encode each sample using encoding choice,

$\quad$ (f_j, g_j): prepare $\{\tilde{\rho}_{\boldsymbol{x}_i}^{\boldsymbol{\alpha}, \boldsymbol{\theta}_j}\}_{i=1}^D$;

5 $\quad$ $\boldsymbol{\alpha}^* \leftarrow \arg\min_{\boldsymbol{\alpha}} C_D(\boldsymbol{\alpha}, \boldsymbol{\theta}_j)$;

6 $\quad$ Add noise with parameters $\boldsymbol{p}$:

$\quad$ $\{\tilde{\rho}_{\boldsymbol{x}}^{\boldsymbol{\alpha}^*, \boldsymbol{\theta}_j}\}_{i=1}^D \leftarrow \{\mathcal{E}_{\boldsymbol{p}}(\tilde{\rho}_{\boldsymbol{x}}^{\boldsymbol{\alpha}^*, \boldsymbol{\theta}_j})\}_{i=1}^D$;

7 $\quad$ $\{\boldsymbol{\theta}_j^*\} \leftarrow \arg\min_{\boldsymbol{\theta}_j} C_D(\boldsymbol{\alpha}^*, \boldsymbol{\theta}_j)$;

8 $\quad$ $C_j^* \leftarrow C_D(\boldsymbol{\alpha}^*, \boldsymbol{\theta}_j^*)$;

9 $\quad$ **if** $C_j^* \leqslant C^*$ **then**

10 $\quad\quad$ $C^* \leftarrow C_j^*$;

11 $\quad\quad$ $(f^*, g^*) \leftarrow f_j(\boldsymbol{\theta}_j^*), g_j(\boldsymbol{\theta}_j^*)$;

12 $\quad$ **end**

13 **end**

$\quad$ **output:** $C^*, \boldsymbol{\alpha}^*, f^*, g^*$

---

2. Robustness for factorizable noise

Theorem 12. If $\mathcal{E}$ is any noise channel which factorizes into a single qubit channel, and a multiqubit channel as follows:

$$\mathcal{E}(\rho) = \mathcal{E}_{\bar{c}}(\tilde{\rho}_{\boldsymbol{x}}^{\bar{c}}) \otimes \mathcal{E}_c(\tilde{\rho}_{\boldsymbol{x}}^c),$$

where without loss of generality $\mathcal{E}_c$ acts only on the classification qubit $[\tilde{\rho}_{\boldsymbol{x}}^c = \mathrm{Tr}_{\bar{c}}(\tilde{\rho}_{\boldsymbol{x}})]$ after encoding and unitary

evolution, and $\mathcal{E}_{\bar{c}}$ acts on all other qubits arbitrarily, $[\tilde{\rho}_{\boldsymbol{x}}^{\bar{c}} = \text{Tr}_c(\tilde{\rho}_{\boldsymbol{x}})]$. Further assume the state meets the robust classification requirements for the single qubit error channel $\mathcal{E}_c$. Then the classifier will be robust to $\mathcal{E}$.

Proof. The correct classification again depends on the classification qubit measurement probabilities: $\text{Tr}(|0\rangle\langle 0|^{(c)} \otimes I^{\otimes n-1}\tilde{\rho}_{\boldsymbol{x}})$, $\text{Tr}(|1\rangle\langle 1|^{(c)} \otimes I^{\otimes n-1}\tilde{\rho}_{\boldsymbol{x}})$. If $\tilde{\rho}_{\boldsymbol{x}}$ is robust to the single qubit error channel $\mathcal{E}_c$, this means

$$\text{Tr}\left(\Pi_0^c \tilde{\rho}_{\boldsymbol{x}}\right) \geqslant 1/2 \Rightarrow \text{Tr}\left(\Pi_0^c \mathcal{E}_c(\tilde{\rho}_{\boldsymbol{x}})\right) \geqslant 1/2,$$

$$\text{Tr}\left(\Pi_1^c \tilde{\rho}_{\boldsymbol{x}}\right) < 1/2 \Rightarrow \text{Tr}\left(\Pi_1^c \mathcal{E}_c(\tilde{\rho}_{\boldsymbol{x}})\right) < 1/2.$$

Then WLOG, assume the point $\boldsymbol{x}$ classified as $y(\tilde{\rho}_{\boldsymbol{x}}) = 0$ before the noise, then

$$\begin{aligned}
\text{Tr}\left(\Pi_0^c \mathcal{E}(\tilde{\rho}_{\boldsymbol{x}})\right) &= \text{Tr}\left(\Pi_0^c \left[\mathcal{E}_{\bar{c}}(\tilde{\rho}_{\boldsymbol{x}}^{\bar{c}}) \otimes \mathcal{E}_c(\tilde{\rho}_{\boldsymbol{x}}^c)\right]\right) \\
&= \text{Tr}_{\bar{c}}\left(\mathcal{E}_{\bar{c}}(\tilde{\rho}_{\boldsymbol{x}}^{\bar{c}})\right)\text{Tr}_c\left(|0\rangle\langle 0|\mathcal{E}_c(\tilde{\rho}_{\boldsymbol{x}}^c)\right) \\
&= \text{Tr}\left(|0\rangle\langle 0|\mathcal{E}_c(\tilde{\rho}_{\boldsymbol{x}}^c)\right) \geqslant 1/2.
\end{aligned}$$ ∎

The above theorem is a simple consequence of causality in the circuit, only errors which have to happen before the measurement can corrupt the outcome. As such, outside of single qubit errors, we only need to consider errors before the measurement which specifically involve the classification qubit.

3. Fidelity bound

Here we derive the fidelity bound (73) in a similar fashion to (72):

$$\begin{aligned}
\Delta_{\mathcal{E}}C &:= |C_{\mathcal{E}} - C|, \\
&= |\text{Tr}[D(\mathcal{E}(\tilde{\sigma}) - \tilde{\sigma})]| \\
&\leqslant \frac{1}{M}\sum_{i=1}^{M} |\text{Tr}(D[\mathcal{E}(\tilde{\rho}_{\boldsymbol{x}_i}) \otimes |y_i\rangle\langle y_i| - \tilde{\rho}_{\boldsymbol{x}_i} \otimes |y_i\rangle\langle y_i|])| \\
&\leqslant \frac{1}{M}\sum_{i=1}^{M} ||D||_{\infty}||[\mathcal{E}(\tilde{\rho}_{\boldsymbol{x}_i}) - \tilde{\rho}_{\boldsymbol{x}_i}] \otimes |y_i\rangle\langle y_i|||_1 \\
&\leqslant \frac{2}{M}\sum_{i=1}^{M} \sqrt{1 - F(\mathcal{E}(\tilde{\rho}_{\boldsymbol{x}_i}), \tilde{\rho}_{\boldsymbol{x}_i})}. \qquad \text{(B4)}
\end{aligned}$$

TABLE II. Optimal parameters $[\theta, \phi]$ for dense angle encoding (with parameters in $U(\boldsymbol{\alpha})$ trained in noiseless environment) in (a) noiseless environment, (b) noisy environment (i.e., amplitude damping channel is added), and (c) for maximal robustness. Optimal parameters in noisy environment are closer to fixed point of amplitude damping channel ($|0\rangle$, i.e., $\theta \equiv 0$) and give a higher value of δ robustness.

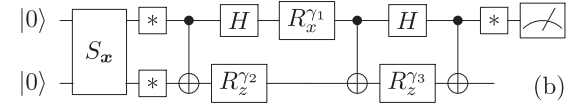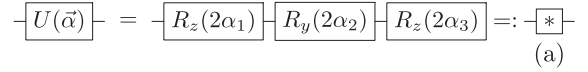| Parameters	Accuracy w/o noise	Accuracy w/noise	δ robustness
$[\theta, \phi] = [2.9, 2.9]$	100%	84%	70%
$[\theta, \phi] = [1.6, 3.9]$	49%	100%	81%
$[\theta, \phi] = [0, 0]$	43%	43%	100%

FIG. 13. Circuit diagrams for the QNN Ansätze we use to obtain numerical results. (a) Ansatz for the single qubit classifier. Here, R_z^{α} and R_y^{β} denote rotations around the z axis and y axis with angles α, β, respectively, of the Bloch sphere. We note that any element of $U(2)$ can be represented by this Ansatz [68]. (b) Ansatz for the two qubit classifier, with 12 parameters, $\{\alpha_k\}_{k=1}^{12}$. The first six parameters, $\alpha_1, \ldots, \alpha_6$, are contained in the first two single qubit unitaries, and $\alpha_{10}, \ldots, \alpha_{12}$ are the parameters of the final single qubit gate. The parameters of the intermediate rotations are defined by $\{\gamma_1, \gamma_2, \gamma_3\} := \{2\alpha_7 + \pi, 2\alpha_8, 2\alpha_9\}$. This decomposition can realize any two qubit unitary [87] up to global phase with the addition of a single qubit rotation on the bottom qubit at the end of the circuit. We omit this rotation since we only measure the first qubit for classification. As such, we reduce the number of trainable parameters ($\boldsymbol{\alpha}$) from 15 to 12.

Again, we use Hölder's, the Fuchs-van de Graaf and the triangle inequalities, with $||D||_{\infty} := \max_j |\lambda_j(D)| = 1$.

APPENDIX C: MORE DETAILS ON NUMERICAL RESULTS

In this Appendix, we present supplementary numerical results to those in the main text. Firstly, in Fig. 12, we illustrate the three single qubit data sets we employ here, namely. the vertical, diagonal, and moons. The former two are linearly separable, whereas the moons data set is nonlinear.

Second, to complement the results of Fig. 9 in the main text, in Table II we provide the best parameters found in the
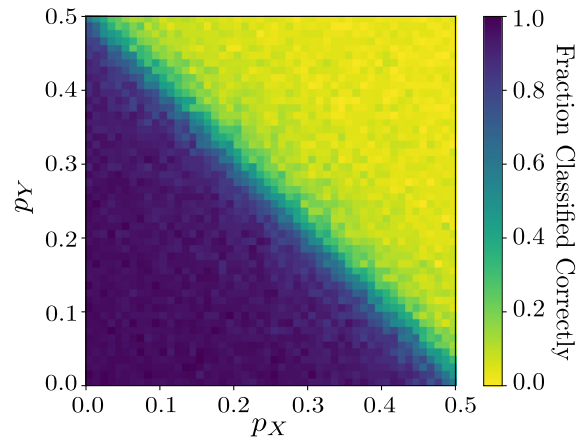

FIG. 14. Misclassification percentage as a result of Pauli noise with strengths $\{p_X, p_Y, p_Z = 0\}$ As expected, classification is robust for $p_X + p_Y < 1/2$ based on Theorem 1, and a sharp transition occurs when this constraint is violated to give maximal misclassification. By classified correctly, in this context, we mean the fraction of points which are classified *the same* with and without noise.

procedure. Each set of parameters (each row, measured in radians) performs optimally in one of three areas. The first is the noiseless environment, in which a $\theta \approx 2.9$ parameter performs optimally. The second is the amplitude-damped environment, in which $\theta \approx 1.6$ achieves the best accuracy and, finally, $\theta = 0$ is the most robust point to encode in, for the whole data set. For each of these parameter sets, we also test them in the other scenarios, for example, the best parameters found in the noisy environment ($[\theta, \phi] = [1.6, 3.9]$) have a higher δ-Robustness (81%) than those in the noiseless environment (70%), since these parameters force points to be encoded closer to the $|0\rangle$ state, i.e., the fixed point of the channel in question.

Figure 13 illustrates the specific decompositions for the single and two qubit classifiers we utilize for the numerical results in the text. For the matrix representation of the circuit shown in Fig. 13(a), see Appendix (A3).

To illustrate the results of Theorem 1, we focus on the dense angle encoding, which can achieve nearly 100% accuracy on the vertical data set. We then compute the percentage which would be misclassified as a function of and Pauli noise parameters. The results are seen in Fig. 14, similar to that observed in Fig. 11. We note here that for values of $p_X + p_Y > 1/2$, one has two strategies to achieve robustness. The first is to adjust the measurement basis as per Corollary 1 and requires changing the model itself. Alternatively, one can apply an extra step of post processing and relabel every output $\hat{y} = 0$ to $\hat{y} = 1$ and vice versa.

Algorithm 1 contains pseudocode for the encoding learning algorithm discussed in the main text.

[1] R. P. Feynman, Simulating physics with computers, Int. J. Theor. Phys. **21**, 467 (1982).

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

[3] R. LaRose, Overview and comparison of gate level quantum software platforms, Quantum **3**, 130 (2019).

[4] J. Preskill, Quantum Computing in the NISQ era and beyond, Quantum **2**, 79 (2018).

[5] C. Gidney and M. Ekerå, How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits, arXiv:1905.09749.

[6] 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 *et al.*, Quantum supremacy using a programmable superconducting processor, Nature **574**, 505 (2019).

[7] J. R. McClean, J. Romero, R. Babbush, and A. Aspuru-Guzik, The theory of variational hybrid quantum-classical algorithms, New J. Phys. **18**, 023023 (2016).

[8] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik, and J. L. O'Brien, A variational eigenvalue solver on a photonic quantum processor, Nat. Commun. **5**, 4213 (2014).

[9] E. Farhi, J. Goldstone, and S. Gutmann, A quantum approximate optimization algorithm, arXiv:1411.4028.

[10] R. LaRose, A. Tikku, E. O'Neel-Judy, L. Cincio, and P. J. Coles, Variational quantum state diagonalization, npj Quant. Info. **5**, 57 (2019).

[11] S. Khatri, R. LaRose, A. Poremba, L. Cincio, A. T. Sornborger, and P. J. Coles, Quantum-assisted quantum compiling, Quantum **3**, 140 (2019); Publisher: Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften.

[12] T. Jones, A. Brown, I. Bush, and S. C. Benjamin, QuEST and high performance simulation of quantum computers, Sci. Rep. **9**, 10736 (2019).

[13] N. Klco, E. F. Dumitrescu, A. J. McCaskey, T. D. Morris, R. C. Pooser, M. Sanz, E. Solano, P. Lougovski, and M. J. Savage, Quantum-classical computation of Schwinger model dynamics using quantum computers, Phys. Rev. A **98**, 032331 (2018).

[14] N. Klco and M. J. Savage, Digitization of scalar fields for quantum computing, Phys. Rev. A **99**, 052335 (2019).

[15] C. Bravo-Prieto, R. LaRose, M. Cerezo, Y. Subasi, L. Cincio, and P. J. Coles, Variational quantum linear solver: A hybrid algorithm for linear systems, arXiv:1909.05820.

[16] X. Xu, J. Sun, S. Endo, Y. Li, S. C. Benjamin, and X. Yuan, Variational algorithms for linear algebra, arXiv:1909.03898.

[17] H.-Y. Huang, K. Bharti, and P. Rebentrost, Near-term quantum algorithms for linear systems of equations, arXiv:1909.07344.

[18] A. Arrasmith, L. Cincio, A. T. Sornborger, W. H. Zurek, and P. J. Coles, Variational consistent histories as a hybrid algorithm for quantum foundations, Nat. Commun. **10**, 3438 (2019).

[19] J. Biamonte, Universal Variational Quantum Computation, arXiv:1903.04500.

[20] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven, Barren plateaus in quantum neural network training landscapes, Nat. Commun. **9**, 4812 (2018).

[21] E. Grant, L. Wossnig, M. Ostaszewski, and M. Benedetti, An initialization strategy for addressing barren plateaus in parametrized quantum circuits, Quantum **3**, 214 (2019).

[22] M. Cerezo, A. Sone, T. Volkoff, L. Cincio, and P. J. Coles, Cost-function-dependent barren plateaus in shallow quantum neural networks, arXiv:2001.00550.

[23] K. Sharma, S. Khatri, M. Cerezo, and P. Coles, Noise resilience of variational quantum compiling, New J. Phys. **22**, 043006 (2020).

[24] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning* (MIT Press, 2016).

[25] E. Farhi and H. Neven, Classification with quantum neural networks on near term processors, arXiv:1802.06002.

[26] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, Parameterized quantum circuits as machine learning models, Quantum Sci. Technol. **4**, 043001 (2019).

[27] G. Verdon, M. Broughton, and J. Biamonte, A quantum algorithm to train neural networks using low-depth circuits, arXiv:1712.05304.

[28] 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 Quant. Info. **5**, 45 (2019).

[29] J.-G. Liu and L. Wang, Differentiable learning of quantum circuit Born machines, Phys. Rev. A **98**, 062324 (2018).

[30] B. Coyle, D. Mills, V. Danos, and E. Kashefi, The Born supremacy: Quantum advantage and training of an Ising born machine, npj Quantum. Inf. **6**, 60 (2020).

[31] P.-L. Dallaire-Demers and N. Killoran, Quantum generative adversarial networks, Phys. Rev. A **98**, 012324 (2018).

[32] A. Mari, T. R. Bromley, J. Izaac, M. Schuld, and N. Killoran, Transfer learning in hybrid classical-quantum neural networks, arXiv:1912.08278.

[33] M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, Circuit-centric quantum classifiers, Phys. Rev. A **101**, 032308 (2020).

[34] M. Schuld and F. Petruccione, *Supervised Learning with Quantum Computers*, Quantum Science and Technology (Springer International Publishing, 2018).

[35] E. Grant, M. Benedetti, S. Cao, A. Hallam, J. Lockhart, V. Stojevic, A. G. Green, and S. Severini, Hierarchical quantum classifiers, npj Quant. Info. **4**, 65 (2018).

[36] A. Pérez-Salinas, A. Cervera-Lierta, E. Gil-Fuster, and J. I. Latorre, Data re-uploading for a universal quantum classifier, Quantum **4**, 226 (2020).

[37] C. Blank, D. K. Park, J.-K. K. Rhee, and F. Petruccione, Quantum classifier with tailored quantum kernel, npj Quant. Info. **6**, 41 (2020).

[38] A. Abbas, M. Schuld, and F. Petruccione, On quantum ensembles of quantum classifiers, Quantum Mach. Intell. **2**, 1 (2020).

[39] M. Schuld and N. Killoran, Quantum Machine Learning in Feature Hilbert Spaces, Phys. Rev. Lett. **122**, 040504 (2019).

[40] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, Supervised learning with quantum-enhanced feature spaces, Nature **567**, 209 (2019).

[41] J. M. Kübler, K. Muandet, and B. Schölkopf, Quantum mean embedding of probability distributions, Phys. Rev. Research **1**, 033159 (2019).

[42] Y. Suzuki, H. Yano, Q. Gao, S. Uno, T. Tanaka, M. Akiyama, and N. Yamamoto, Analysis and synthesis of feature map for kernel-based quantum classifier, Quantum Machine Intelligence **2**, 1 (2020).

[43] P. Wittek, *Quantum Machine Learning: What Quantum Computing Means to Data Mining* (Elsevier, 2014).

[44] E. C. Behrman, J. Niemel, J. E. Steck, and S. R. Skinner, A Quantum Dot Neural Network (1996), http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.56.1507.

[45] S. Arunachalam and R. de Wolf, Optimal quantum sample complexity of learning algorithms, in *Proceedings of the 32nd Computational Complexity Conference*, CCC '17, (*Riga, Latvia*) (Schloss Dagstuhl–Leibniz-Zentrum fuer Informatik, 2017), pp. 1–31.

[46] L. G. Wright and P. L. McMahon, The capacity of quantum neural networks, arXiv:1908.01364.

[47] Y. Du, M.-H. Hsieh, T. Liu, and D. Tao, The expressive power of parameterized quantum circuits, Phys. Rev. Research **2**, 033125 (2020).

[48] R. A. Servedio and S. J. Gortler, Equivalences and separations between quantum and classical learnability, SIAM J. Comput. **33**, 1067 (2004).

[49] S. Arunachalam, A. B. Grilo, and H. Yuen, Quantum statistical query learning, arXiv:2002.08240.

[50] S. Kuyucak and A.E. Stuchbery, Comment on "$\Delta I = 4$ Bifurcation in Ground Bands of Even-Even Nuclei and the Interacting Boson Model", Phys. Rev. Lett. **82**, 1999 (1999).

[51] K. Temme, S. Bravyi, and J. M. Gambetta, Error Mitigation for Short-Depth Quantum Circuits, Phys. Rev. Lett. **119**, 180509 (2017).

[52] T. Giurgica-Tiron, Y. Hindy, R. LaRose, A. Mari, and W. J. Zeng, Digital zero noise extrapolation for quantum error mitigation, arXiv:2005.10921.

[53] J. R. McClean, Z. Jiang, N. C. Rubin, R. Babbush, and H. Neven, Decoding quantum errors with subspace expansions, Nat. Commun. **11**, 636 (2020).

[54] C. Xue, Z.-Y. Chen, Y.-C. Wu, and G.-P. Guo, Effects of quantum noise on quantum approximate optimization algorithm, arXiv:1909.02196.

[55] M. Alam, A. Ash-Saki, and S. Ghosh, Analysis of quantum approximate optimization algorithm under realistic noise in superconducting qubits, arXiv:1907.09631.

[56] J. Marshall, F. Wudarski, S. Hadfield, and T. Hogg, Characterizing local noise in QAOA circuits, IOP SciNotes **1**, 025208 (2020).

[57] A. W. Harrow, A. Hassidim, and S. Lloyd, Quantum Algorithm for Solving Linear Systems of Equations, Phys. Rev. Lett. **103**, 150502 (2009)

[58] I. Kerenidis and A. Prakash, Quantum recommendation systems, arXiv:1603.08675.

[59] I. Kerenidis and A. Luongo, Quantum classification of the MNIST dataset via slow feature analysis, Phys. Rev. A **101**, 062327 (2020).

[60] D. Dervovic, M. Herbster, P. Mountney, S. Severini, N. Usher, and L. Wossnig, Quantum linear systems algorithms: a primer, arXiv:1802.08227.

[61] Z. Zhao, J. K. Fitzsimons, P. Rebentrost, V. Dunjko, and J. F. Fitzsimons, Smooth input preparation for quantum and quantum-inspired machine learning, arXiv:1804.00281.

[62] E. Knill, R. Laflamme, and G. J. Milburn, A scheme for efficient quantum computation with linear optics, Nature **409**, 46 (2001).

[63] T. L. Fine, S. L. Lauritzen, M. Jordan, J. Lawless, and V. Nair, *Feedforward Neural Network Methodology*, 1st ed. (Springer-Verlag, Berlin, 1999).

[64] I. Cong, S. Choi, and M. D. Lukin, Quantum convolutional neural networks, Nat. Phys. **15**, 1273 (2019).

[65] M. Henderson, S. Shakya, S. Pradhan, and T. Cook, Quanvolutional neural networks: Powering image recognition with quantum circuits, Quantum Machine Intelligence **2**, 1 (2020).

[66] E. Stoudenmire and D. J. Schwab, Supervised Learning with Tensor Networks, in *Advances in Neural Information Processing Systems 29*, edited by D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett (Curran Associates, Inc., 2016), pp. 4799–4807.

[67] S. Cao, L. Wossnig, B. Vlastakis, P. Leek, and E. Grant, Cost function embedding and dataset encoding for machine learning with parameterized quantum circuits, Phys. Rev. A **101**, 052309 (2020).

[68] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quant. Info.*, 10th anniversary ed. (Cambridge University Press, Cambridge, 2010).

[69] M. Raghu, B. Poole, J. Kleinberg, S. Ganguli, and J. Sohl-Dickstein, On the expressive power of deep neural networks, in *Proceedings of the 34th International Conference on Machine Learning*, edited by D. Precup and Y. W. Teh, Proceedings of

Machine Learning Research, Vol. 70 (International Convention Centre, Sydney, Australia) (PMLR, 2017), pp. 2847–2854.

[70] J. Preskill, Quant. Info. and Computation (1998), https://www.lorentz.leidenuniv.nl/quantumcomputers/literature/.

[71] J. Watrous, *The Theory of Quantum Information* (Cambridge University Press, 2018), p. 598.

[72] P. J. J. O'Malley, R. Babbush, I. D. Kivlichan, J. Romero, J. R. McClean, R. Barends, J. Kelly, P. Roushan, A. Tranter, N. Ding, B. Campbell, Y. Chen, Z. Chen, B. Chiaro, A. Dunsworth, A. G. Fowler, E. Jeffrey, A. Megrant, J. Y. Mutus, C. Neill, C. Quintana, D. Sank, A. Vainsencher, J. Wenner, T. C. White, P. V. Coveney, P. J. Love, H. Neven, A. Aspuru-Guzik, and J. M. Martinis, Scalable Quantum Simulation of Molecular Energies, Phys. Rev. X **6**, 031007 (2016).

[73] S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, and N. Killoran, Quantum embeddings for machine learning, arXiv:2001.03622.

[74] J. Schauder, Der Fixpunktsatz in Funktionalraümen, Studia Math. **2**, 171 (1930).

[75] L. Gentini, A. Cuccoli, S. Pirandola, P. Verrucchi, and L. Banchi, Noise-assisted variational hybrid quantum-classical optimization, arXiv:1912.06744.

[76] C. A. Fuchs and J. v. d. Graaf, Cryptographic distinguishability measures for quantum-mechanical states, IEEE Trans. Inf. Theory **45**, 1216 (1999).

[77] B. Coyle, NoiseRobustClassifier: Noise Robust Data Encodings for Quantum Classifiers (2020), https://zenodo.org/badge/latestdoi/221948307.

[78] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, Scikit-learn: Machine learning in Python, J. Mach. Learn. Res. **12**, 2825 (2011).

[79] R. A. Fisher, the use of multiple measurements in taxonomic problems, Ann. Eugenics **7**, 179 (1936).

[80] S. Bravyi, J. M. Gambetta, A. Mezzacapo, and K. Temme, Tapering off qubits to simulate fermionic Hamiltonians, arXiv:1701.08213.

[81] M. Ostaszewski, E. Grant, and M. Benedetti, Quantum circuit structure learning, arXiv:1905.09692.

[82] D. A. Lidar, I. L. Chuang, and K. B. Whaley, Decoherence Free Subspaces for Quantum Computation, Phys. Rev. Lett. **81**, 2594 (1998).

[83] D. A. Lidar, Review of decoherence free subspaces, noiseless subsystems, and dynamical decoupling, Adv. Chem. Phys. **154**, 295 (2014).

[84] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, Gradient-based learning applied to document recognition, Proc. IEEE **86**, 2278 (1998).

[85] S. Lu, L.-M. Duan, and D.-L. Deng, Quantum adversarial machine learning, Phys. Rev. Research **2**, 033212 (2020).

[86] S. Kechrimparis, C. M. Kropf, F. Wudarski, and J. Bae, Channel coding of a quantum measurement, arXiv:1908.10735.

[87] G. Vidal and C. M. Dawson, Universal quantum circuit for two-qubit transformations with three controlled-NOT gates, Phys. Rev. A **69**, 010301(R) (2004).

Correction: An error in wording in the penultimate sentence of the abstract has been fixed.