Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Data Re-Uploading on with a single qudit

Noah Wach, Manuel Rudolph, Fred Jendrzejewski, Sebastian Schmitt

2023

Preprint:

This is a post-peer-review, pre-copyedit version of an article published in Quantum Machine Intelligence. The final authenticated version is available online at: https://doi.org/10.1007/s42484-023-00125-0

Quantum Machine Intelligence (2023) 5:36
https://doi.org/10.1007/s42484-023-00125-0

RESEARCH ARTICLE

Data re-uploading with a single qudit

Noah L. Wach[1] · Manuel S. Rudolph[2,3] · Fred Jendrzejewski[1,4] · Sebastian Schmitt[5]

Received: 26 February 2023 / Accepted: 3 July 2023
© The Author(s) 2023

Abstract
Quantum two-level systems, i.e., qubits, form the basis for most quantum machine learning approaches that have been proposed throughout the years. However, higher dimensional quantum systems constitute a promising alternative and are increasingly explored in theory and practice. Here, we explore the capabilities of multi-level quantum systems, so-called qudits, for their use in a quantum machine learning context. We formulate classification and regression problems with the data re-uploading approach and demonstrate that a quantum circuit operating on a single qudit is able to successfully learn highly non-linear decision boundaries of classification problems such as the MNIST digit recognition problem. We demonstrate that the performance strongly depends on the relation between the qudit states representing the labels and the structure of labels in the training data set. Such a bias can lead to substantial performance improvement over qubit-based circuits in cases where the labels, the qudit states, and the operators employed to encode the data are well-aligned. Furthermore, we elucidate the influence of the choice of the elementary operators and show that a squeezing operator is necessary to achieve good performances. We also show that there exists a trade-off for qudit systems between the number of circuit-generating operators in each processing layer and the total number of layers needed to achieve a given accuracy. Finally, we compare classification results from numerically exact simulations and their equivalent implementation on actual IBM quantum hardware. The findings of our work support the notion that qudit-based algorithms exhibit attractive traits and constitute a promising route to increasing the computational capabilities of quantum machine learning approaches.

Keywords Quantum machine learning · Qudits · Parameterized quantum circuits · Data re-uploading

1 Introduction

In recent years, the field of quantum machine learning has attracted much attention. There, quantum circuits are employed as central processing units for data-driven applications (Biamonte et al., 2017; Schuld and Petruccione, 2018; Dunjko and Briegel, 2018). While it is currently not clear whether or not quantum processing can provide a benefit on practical machine learning problems (Schuld and Killoran, 2022; Schuld, 2021), there has been some evidence that quantum machine learning models can outperform classical models in certain tasks (Liu et al., 2021; Sweke et al., 2021; Gyurik and Dunjko, 2022; Gyurik et al., 2022). While most studies, theoretical (Bharti et al., 2022; Montanaro, 2016) as well as experimental (Graham et al., 2022; Pino et al., 2021; Kjaergaard et al., 2020), focus on quantum systems consisting of two-level quantum systems, i.e., quantum bits (qubits), quantum computing hardware and algorithms can also be based on d-level systems, which are typically called qudits (Wang et al., 2020; Ringbauer et al., 2022). Such qudit systems were shown to have advantages in specific contexts (Cozzolino et al., 2019; Sheridan and Scarani, 2010) and have already been applied to several tasks (Bravyi et al., 2022; Deller et al., 2023; Weggemans et al., 2022). However, a full evaluation on application-relevant tasks is still lacking.

Currently, it is not clear whether there is a fundamental advantage or disadvantage of utilizing qudit systems over

✉ Sebastian Schmitt
sebastian.schmitt@honda-ri.de

1 Kirchhoff-Institut für Physik, Universität Heidelberg, Im Neuenheimer Feld 227, Heidelberg 69120, Germany

2 Zapata Computing Canada Inc., 25 Adelaide St E, Toronto, Ontario M5C3A1, Canada

3 Institute of Physics, Ecole Polytechnique Fédérale de Lausanne (EPFL), Station 3, Lausanne 1015, Switzerland

4 Alqor UG (haftungsbeschränkt), Alexanderstr. 65, Frankfurt am Main 60489, Germany

5 Honda Research Institute Europe GmbH, Carl-Legien-Str. 30, Offenbach 63073, Germany

Published online: 14 August 2023

their qubit counterparts for quantum machine learning or other application domains such as quantum optimization. However, qudits provide a complementary route to increasing the Hilbert space size in pursuit of better computation performance. This is an alternative direction to qubit systems, where the route to larger Hilbert spaces is to increase the number of qubits. Among other challenges, increasing the number of qubits comes with increased difficulty of engineering high-fidelity interactions between separate qubits, which are technologically challenging and typically involve larger gate errors than single qubit operations, see for example, Kjaergaard et al. (2020), Fedorov et al. (2022) or Resch and Karpuzcu (2019). In that regard, expanding the local Hilbert space dimension by utilization of qudit systems seems promising, since certain interactions between spatially separated qubits could instead be implemented by local qudit operations, and thus might prove to be more efficient (Fischer et al., 2022; Ringbauer et al., 2022). We discuss several of such aspects of qudit system at various parts in this work.

We explore the usage of qudit systems in the context of the data re-uploading (Pérez-Salinas et al., 2020) quantum machine learning approach to build regression and multi-class classification models. In this scheme, a single qudit provides a natural way of encoding multiple classes by representing each class label as an orthogonal basis state. We focus on the prominent data re-uploading architecture, since it was shown in the original work that a single qubit is already sufficient to implement a universal classifier. Due to the comparably small resource requirement, data re-uploading circuits were also experimentally implemented on a trapped ion device (Dutta et al., 2022). Additionally, the encoding characteristics of these models is well understood, as it generates increasingly higher Fourier terms with more layers (Schuld et al., 2021). Thus, it is natural to first evaluate and confirm these same properties for a single qudit as well. Extending these circuits to multi-qubit/multi-qudit circuits would allow for more sophisticated setups including entanglement. However, in this work, we refrain from doing this and instead focus on the most simple, non-trivial case to investigate the fundamental aspects of qubits and qudits related to basic operators and Hilbert spaces.

This paper is structured as follows: In Sect. 2, we introduce the mathematical description of qudits. Section 3 illustrates the implementation of the data re-uploading algorithm (Pérez-Salinas et al., 2020) with qudits and shows the circuit structure as well as the chosen loss functions used during training. Section 4 presents the training procedure which is done numerically and on IBMQ hardware. To be able to run our model on qubit-based quantum hardware, we present a way to encode qudits with multiple qubits. In Sect. 5, we verify the expressivity of our model by testing it on a simple regression problem. We then go on and test our model on multi-class classification problems where we show an intrinsic bias between the qudit state representation and the data structure. Finally, we present numerical results of the model when being trained on the MNIST handwritten digits data set (LeCun and Cortes, 2005). We additionally investigate an equivalent qubit-based implementation on IBMQ hardware and the effect of entangling operations on the performance of the model.

2 Qudits

d level quantum systems, typically called qudits, are a generalization of qubits to $d > 2$ and can serve as a basis for quantum information processing. The Hilbert space is spanned by d orthonormal basis vectors, denoted by $|0\rangle, |1\rangle, ...|d-1\rangle$ and arbitrary qudit states can be represented by the supersposition $|\psi\rangle = \sum_{k=0}^{d-1} c_k|k\rangle$ with the normalization condition $\sum_{k=0}^{d-1} |c_k|^2 = 1$.

Inspired by cold atom systems (Kasper et al., 2022), we interpret a d-level qudit as a spin with total angular momentum $\ell = \frac{d-1}{2}$, such that the basis state $|k\rangle$ corresponds to the spin eigenstate with angular momentum $m = \frac{2k-d+1}{2}$. A natural set of operations on qudits states, which is also easily implementable in experiments, is given by the angular momentum operators $\{L_x, L_y, L_z\}$. These generate rotations around the corresponding axes and obey the canonical commutation relations of the special unitary group $SU(2)$, $[L_i, L_j] = i\epsilon_{ijk}L_k$. The action of the angular momentum operators on the qudit basis state is given by

$$L_x|k\rangle = \tfrac{1}{2}\big(\gamma_{d,k+1}|k+1\rangle + \gamma_{d,k-1}|k-1\rangle\big) \tag{1}$$

$$L_y|k\rangle = \tfrac{1}{2i}\big(\gamma_{d,k+1}|k+1\rangle - \gamma_{d,k-1}|k-1\rangle\big) \tag{2}$$

$$L_z|k\rangle = \tfrac{2k-d+1}{2}|k\rangle \tag{3}$$

where $k \in [0, d-1]$ and with $\gamma_{d,k} = \sqrt{(d-k-1)(k+1)}$.

There are various ways to define a universal gate set to realize arbitrary actions in the qudit Hilbert space, see, e.g., Wang et al. (2020) and Luo et al. (2014). For a single qudit, as it is considered in this work, we choose the two angular momentum operators L_x and L_z and the squeezing or one-axis twisting operator $L_{z^2} = L_z^2$. For $d > 2$, this additional gate is needed to be able to generate any state by (possibly many) repeated finite rotations as detailed in Kasper et al. (2022) and Giorda et al. (2003). The reason for this is that the iterated commutators of these three operators generate all $d^2 - 1$ Hermitian basis operators, which are necessary to generate all unitary operations of the $SU(d)$ group.

Equivalent to qubits, the gates for qudit circuits are then generated by exponentiation of the basic operators which implements the rotations

$$R_j(\theta) = e^{-i\theta L_j}, \tag{4}$$

with $j \in \{x, z, z^2\}$ and where $\theta \in \mathbb{R}$ are free parameters.

In Fig. 1, we illustrate the qudit states (panel A) and the action of the elementary operators by showing the Husimi-Q quasi-probability distribution (Husimi, 1940) on a generalized Bloch sphere in panel B. The pure rotation does not deform the probability distributions, but applying the squeezing operation leads to a deformation of the state.

By choosing the operator set $\{L_x, L_z, L_z^2\}$, we impose a ladder structure on the qudit target states. This primarily arises from the operator L_x, which couples each state $|k\rangle$ to its adjacent states $|k+1\rangle$ and $|k-1\rangle$, as it is illustrated in Eq. 1.

3 Data re-uploading with a single qudit

We utilize a single d-level qudit and implement the quantum machine learning model as a data re-uploading quantum circuit (Pérez-Salinas et al., 2020; Jerbi et al., 2023). The quantum circuit in its general form is build up from L layers and encodes a quantum state, which depends on the input data $\mathbf{x}$ as

$$|\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\theta}\rangle = \prod_{l=1}^{L} U(\mathbf{x}, \boldsymbol{\omega}^{(l)}, \boldsymbol{\theta}^{(l)})|0\rangle\,, \tag{5}$$

where $|0\rangle$ is the initial state. The characteristic of the data re-uploading architecture is that the unitary operation of each layer l encompasses data dependent unitaries, which are parametrized by the scaling parameters $\boldsymbol{\omega}^{(l)}$ and data independent operations with free parameters $\boldsymbol{\theta}^{(l)}$.

We tested several layer structures and found that they all produce similar results. In the following, we report results for two architectures. The first one is inspired by classical Euler rotations, where the unitaries of each layer have the structure:

$$U(\mathbf{x}, \boldsymbol{\omega}^{(l)}, \boldsymbol{\theta}^{(l)}) = W(\boldsymbol{\theta}^{(l)})S(\mathbf{x}, \omega^{(l)})\,. \tag{6}$$

The data encoding block S of each layer consists of alternating x and z rotations:

$$S(\mathbf{x}, \boldsymbol{\omega}^{(l)}) = R_\alpha(x_D\omega_D^{(l)}) \cdots R_z(x_2\omega_2^{(l)})R_x(x_1\omega_1^{(l)}) \tag{7}$$

where the number of rotations is determined by the dimensionality D of the input data and consequently $\alpha = x$ ($\alpha = z$) in case D is odd (even). The alternating x and z rotations are chosen to ensure that these are non-commuting. This is necessary to distinguish between the individual dimensions of the data vector. If only x (or z) rotations were chosen to encode the data, the classifier would only be able to learn the sum of the input data.

The data-independent block in each layer is composed of a sequence of three rotations followed by a squeezing gate R_{z^2} as the last operation, i.e.,

$$W(\boldsymbol{\theta}^{(l)}) = R_{z^2}(\theta_4^{(l)})R_x(\theta_3^{(l)})R_z(\theta_2^{(l)})R_x(\theta_1^{(l)}) \tag{8}$$

The first three operations give rise to Euler angles, starting from $|0\rangle$, and thus the freedom to create overlap with an arbitrary qudit state. The total number of adjustable parameters of a L-layer model is given by $(4+D)L$. The action of this circuit structure is illustrated in Fig. 1c.

The second architecture considered here is inspired by the simplified form presented by Pérez-Salinas et al. (2020) and has the structure:

$$U(\mathbf{x}, \boldsymbol{\omega}^{(l)}, \boldsymbol{\theta}^{(l)}) = e^{-i\sum_j^D(\theta_j^{(l)}+\omega_j^{(l)}x_j)L_{c(j)}-i\theta_{D+1}^{(l)}L_{z^2}}\,. \tag{9}$$

Here, the first term in the exponent is the sum over the three angular momentum operators, i.e., the generators of the $SU(2)$, and the function $c(j) = (j \mod 3)$ selects one of them. The second term in the exponent is a generalization of the original simplified architecture by including the squeezing operator in each layer. The total number of adjustable parameters for an L-layer model of this structure is given by $(2D+1)L$.

We like to point out that in this form, both architectures implement the aforementioned ladder structure, since the data encoding is done with the angular momentum operators, that only couple each qudit basis state to its adjacent states.

In our work, we approach supervised classification and regression tasks by using qudit-based quantum circuits. In a supervised learning setting, there exists a set of N training samples $(\mathbf{x}_i, y_i)$ with $i = 1, \ldots, N$, which consist of pairs of input samples $\mathbf{x}$ with corresponding output values y. The input samples are real-valued D-dimensional vectors $\mathbf{x} \in \mathbb{R}^D$, while the output values are either real numbers or a finite set of integers for regression and classification tasks, respectively.

For classification problems, the output values $y \in (0, 1, \ldots, d-1)$ indicate to which of the d classes the data sample $\mathbf{x}$ belongs. In the quantum formulation, each class label is represented by a basis state of the d-level qudit $|y\rangle$ with $y = 0, \ldots, d-1$. The model prediction, i.e., the probability that a data sample $\mathbf{x}$ belongs to class y, can then conveniently be calculated from the overlap of the label state and the qudit wave function obtained from the quantum circuit with input $\mathbf{x}$:

$$P(y|\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\theta}) = |\langle y|\psi(\mathbf{x}, \boldsymbol{\omega}, \boldsymbol{\theta})\rangle|^2,\,. \tag{10}$$

Training the quantum circuit is achieved by minimizing a loss function over the given training data set $\mathcal{D} =$

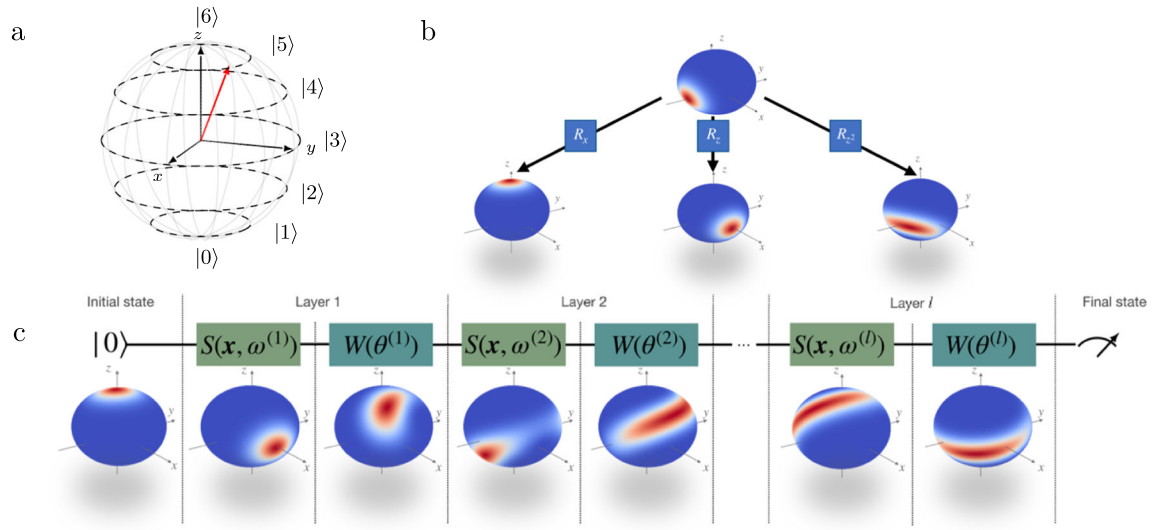

Fig. 1 **a** Schematic representation of $d = 7$ qudit states on a generalized Bloch sphere. **b** Schematic illustration of the action of the three operators R_x (left), R_z (middle), and R_{z^2} (right) on a qudits state. **c** Illustration of the data re-uploading circuit structure

$\{(\mathbf{x}_i, y_i)\}_{i=1,\ldots,N}$. The overlap of Eq. 10 is the basis for formulating the mean squared error (MSE) loss function:

$$\mathcal{L}_{\text{MSE}}(\boldsymbol{\omega}, \boldsymbol{\theta}) = \frac{1}{N} \sum_{i=1}^{N} \left(\langle \bar{y}_i \rangle - y_i\right)^2 \tag{11}$$

with the average predicted label of the quantum model

$$\langle \bar{y}_i \rangle = \sum_{y=0}^{d-1} y \, P(y|\mathbf{x}_i, \boldsymbol{\omega}, \boldsymbol{\theta}) \, . \tag{12}$$

Another popular choice is the overlap loss as used in Pérez-Salinas et al. (2020):

$$\mathcal{L}_{\text{overlap}}(\boldsymbol{\omega}, \boldsymbol{\theta}) = \sum_{i=1}^{N} \left(1 - P(y_i|\mathbf{x}_i, \boldsymbol{\omega}, \boldsymbol{\theta})\right). \tag{13}$$

The learning procedure amounts to adjusting the parameters of the quantum circuit $(\boldsymbol{\omega}, \boldsymbol{\theta})$ in order to minimize the loss function, which is done by running a classical optimization algorithm.

After the quantum circuit has been trained, its accuracy is evaluated by analyzing its predictions of the output variables on a test data set, i.e., a set of data samples not used during the training procedure. For a classification task, the predicted output label is given by the basis state with the highest probability in the quantum state with the corresponding input data sample, i.e.,

$$y_i^{\text{predicted}} = \text{argmax}_y P(y|\mathbf{x}_i, \boldsymbol{\omega}_{\text{opt}}, \boldsymbol{\theta}_{\text{opt}}) \, , \tag{14}$$

where $(\boldsymbol{\omega}_{\text{opt}}, \boldsymbol{\theta}_{\text{opt}})$ are the optimized values of the circuit parameters. The accuracy of the trained model can then be evaluated by calculating the fraction of correctly predicted labels in the test data set:

$$\text{Accuracy} = \frac{1}{N_{\text{test}}} \sum_{i \in \mathcal{D}_{\text{test}}} \delta_{y_i^{\text{predicted}}, y_i} \, , \tag{15}$$

where $\mathcal{D}_{\text{test}}$ denotes the test data set which contains N_{test} data samples and $\delta_{a,b}$ is the Kronecker delta.

The MSE loss of Eq. 11 is also suitable to learn regression tasks. In that case, the output variables are finite-range continuous variables, $y \in [0, d-1]$. The resulting qudit state of the trained quantum circuit is then a superposition of basis states, and the predicted output value for input $\mathbf{x}_i$ is calculated as the expectation value of Eq. 12:

$$\langle \bar{y}_i \rangle^{\text{predicted}} = \sum_{y=0}^{d-1} y \, P(y|\mathbf{x}_i, \boldsymbol{\omega}_{\text{opt}}, \boldsymbol{\theta}_{\text{opt}}) \, . \tag{16}$$

For simulations running on actual quantum hardware, the probability distribution of Eq. 10 is estimated by performing a finite number of measurement shots and recording the measurement results as a histogram. This approximative distribution is then used to calculate the predicted output values of Eqs. 14 and 16.

4 Experimental setup

We perform the training of the quantum circuit utilizing exact numerical simulations of the qudit states and quan-

tum gates. The parameter values of $\boldsymbol{\omega}$ and $\boldsymbol{\theta}$ are initialized randomly in the range $[-\pi, \pi]$. As classical optimization algorithms for minimizing the loss function, we employ the ADAM optimizer (Kingma and Ba, 2014) paired with the exact gradients calculated by the automatic differentiation package JAX (Bradbury et al., 2018). Additionally, we also used L-BFGS-B and Powell optimization approaches from the `scipy` library (Virtanen et al., 2020), where we do not utilize automatic differentiation. Apart from differences in the run time for the training, the results obtained with all approaches were always comparable. If not stated otherwise, all models are trained on randomly selected data sets with size $N = 750$. The performance is evaluated on a separate randomly selected test data set, which contains $\frac{N}{3} = 250$ data samples. In order to obtain reliable statistics of the results, we run 60 different simulations for each setting, each time randomly varying the data set and the model initialization. To visualize the decision boundaries of the trained classifiers, we utilize the visualization library *orqviz* (Rudolph et al., 2021).

In addition to the numerically exact simulation, in 5.6, we re-train and evaluate the qudit quantum circuit on the IBM `ibmq_lima` hardware using the qiskit (Treinish et al., 2023) framework. This allows us to estimate the impact of gate errors and noise of actual NISQ hardware on the learning performance. As the IBM hardware naturally operates on qubits, we employ a mapping of the d-level qudit Hilbert space to $d-1$ qubits, which is inspired by cold atom systems (Kasper et al., 2022; Santra et al., 2022). The qudit basis state $|k\rangle$ is represented by the qubit Dicke-state $|D_k^{d-1}\rangle$, i.e., $|k\rangle \rightarrow |D_k^{d-1}\rangle$ (Gasieniec et al., 2019). The kth Dicke state $|D_k^{d-1}\rangle$ of $d-1$ qubits is given by the equal superposition of all states which have k qubits in the state $|1\rangle$ and $d-1-k$ in state $|0\rangle$, i.e.,

$$|D_k^{d-1}\rangle = \binom{d-1}{k}^{-\frac{1}{2}} \sum_{x\in\{0,1\}^{d-1}, hw(x)=k} |x\rangle\,. \tag{17}$$

where $hw(x)$ indicates the Hamming weight of string x, i.e., the number of 1's in x. The angular momentum operators for the $d-1$ qubit states are defined as the sum over the single-qubit operators, $L_a^{\text{tot}} = \sum_{j=0}^{d-1} L_a^j$ with $a = \{x, y, z\}$, and they act as described by Eq. 1 on the Dicke states. In particular, the Dicke state $|D_k^{d-1}\rangle$ and the qudit state $|k\rangle$ have the same z-component of the angular momentum, i.e., $L_z^{\text{tot}}|D_k^{d-1}\rangle = \frac{2k-d+1}{2}|D_k^{d-1}\rangle$.

In this representation, the squeezing operation consists of all pairwise qubit interactions:

$$L_{z^2}^{\text{tot}} = \left(L_z^{\text{tot}}\right)^2 = \sum_{i,j} L_z^i L_z^j \tag{18}$$

which is implemented as two-qubit zz-rotation gates on the hardware (Treinish et al., 2023). From this form, it is clear that the squeezing operator has the capability to generate correlation and entanglement between the individual qubits (Santra et al., 2022). On the qubit device, we start from the optimized parameters found by the training with the exact simulation and re-train the model on the actual hardware. There, we perform 512 measurement shots for each qiskit circuit evaluation.

For comparison, we also report the accumulated results on classification tasks of a standard classical machine learning approach, namely, the `scikit-learn` (Pedregosa et al, 2018) implementation of the random forest (RF) classifier with 100 estimators, a k-nearest neighbor classifier with $k = 3$ ($k$nn), and a support vector classifier (SVC). The performance of these approaches were always comparable, and we report the cumulative results of running each algorithm 50 times on randomized data stets.

5 Results

5.1 Expressivity in one dimension

First, we examine the expressivity of the data re-uploading circuit with a single qudit on a continuous one-dimensional regression problem analogous to Schuld et al. (2021). We train a model with one qutrit, i.e., a $d = 3$ qudit, to learn the simple function

$$f(x) = \tfrac{1}{2}\big(\cos(2x) + \cos(3.5x)\big) \tag{19}$$

with $x \in [-\pi, \pi]$. As the model output, we use the expectation value of Eq. 12 but shift it to match the data range, i.e., $f_i^{predicted} = \langle \bar{y}_i \rangle - 1$. For training the quantum circuit, we employ the MSE loss function of Eq. 11 and use a training set of 100 linearly distributed samples.

We show the results of a $L = 1$ layer model in the left panel of Fig. 2. It is evident that this model is not able to learn the function properly. This confirms the previous insight for qubit circuits (Schuld et al., 2021) that data re-uploading models learn truncated Fourier series and that a circuit containing only one angle parametrized by the input can only encode one Fourier component. In the right panel, we show the results of a model with $L = 2$ layers. As expected, this model is now able to learn the function with two Fourier coefficients exactly. The derivation is analogous to the qubit case (Schuld et al., 2021), but is not shown in this work. Apart from the results obtained by numerically exact simulations, we also include results from the evaluation on the IBM `ibmq_lima` hardware (green dots in the plot). These predicted values are generally noisy which result from the

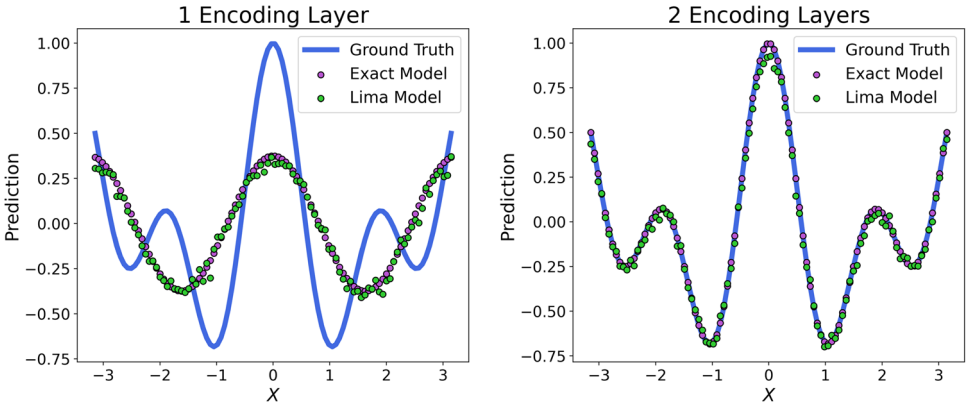

Fig. 2 Results of trained data re-uploading models with structure of Eqs. 6–8 using a $d = 3$ qutrit with one (left) and two (right) layers for the regression task of learning the function in Eq. 19 (blue line). The purple dots indicate the results of the numerically exact model simulation obtained and the green dots the results from re-training and evaluation on the IBM `ibmq_lima` hardware

inherent errors in the quantum circuit, as well as the shot noise from the finite number of samples used to estimate the probability distribution of Eq. 10.

5.2 Two-dimensional classification tasks

Inspired by the original benchmark of the data re-uploading algorithm (Pérez-Salinas et al., 2020), we investigate the performance of the qudit circuit with the structure of Eqs. 6–8 on various two-dimensional multi-class classification problems. The data samples are located on a two-dimensional square $\mathbf{x} = (x_1, x_2) \in [-1, 1]^2$, where each sample is associated with one out of d classes. In the first problem setting, the classes are arranged in parallel horizontal stripes. Figure 3 shows the results for seven classes, where we used a $d = 7$ qudit to represent the classes in the quantum circuit.

In the upper row of the figure, we show results from the case where we align the qudit states with the labels in such a way that the z-components of the spin are in order with the class label. In that case, adjacent classes are represented by qudit states with adjacent z-spin values. Under these circumstances, the model has a strong inductive bias towards the data set caused by the chosen ladder structure. This is illustrated in the leftmost figure where we draw the qudit states on the generalized Bloch sphere. The lowest qudit state $|0\rangle$ is associated with the class of the lowest (blue) stripe, the second lowest qudit $|1\rangle$ with the second lowest stripe, and so on. The corresponding results show that the data re-uploading circuit can predict this data set almost perfectly with three or more layers, $L \geq 3$ (blue graph in top right plot). Remarkably, the performance of the data re-uploading circuit is even better than the classical machine learning models shown on the far right of the right panel. However, the learned classes and decision boundaries, as shown in the middle panels of the figure, can still differ from the ground truth. This is due to the small size of the training data set, which necessarily leads to small random variations of the learned decision boundaries, depending on the precise location of the training data close to the decision boundaries.

The importance of the squeezing operation R_{z^2} in the W operator of Eq. 8 is highlighted by observing the massively degraded performance of the same circuits without this gate (green graph in the top right panel of Fig. 3). Without squeezing, the median accuracy saturates at around 0.7, while with squeezing, the median accuracy reaches 0.95 and higher. This performance difference is attributed to the fact that the squeezing operator is necessary to allow for the representation of arbitrary unitary operations in the qudit Hilbert space.

The second row of Fig. 3 shows the performance when the class labels are randomly assigned to the qudit states. This removes the inductive bias of the model and makes the problem much harder to learn for the qudit quantum circuit, since the ladder structure used for the data encoding does not align to the data set. Consequently, the accuracy drops significantly, and many more layers are necessary to recover the performance level of the scenario with aligned labels.

The plots of the accuracies also show the accumulated results of the three classical machine learning classifiers (RF, SVC, and $k$nn) for comparison. The performance of the quantum circuits including squeezing with randomized assignment of qudit states to labels is comparable to the classical approaches. For aligned labels, the quantum circuit even slightly outperforms the classical approaches.

We also considered several other two-dimensional multi-class classification problems and trained data re-uploading models on several horizontal stripe data sets with varying number of stripes (i.e., classes), as well as data sets where the stripes are not horizontal but rotated by an angle. We also investigated the models on data sets where the class regions are given by concentric rings with approximately the same width and where the center of the rings is somewhere in the plane. The results (not shown) on those models were always qualitatively similar to the ones presented here. Squeezing is always necessary to achieve good performance, and when the label order is aligned with the qudit states, the performance

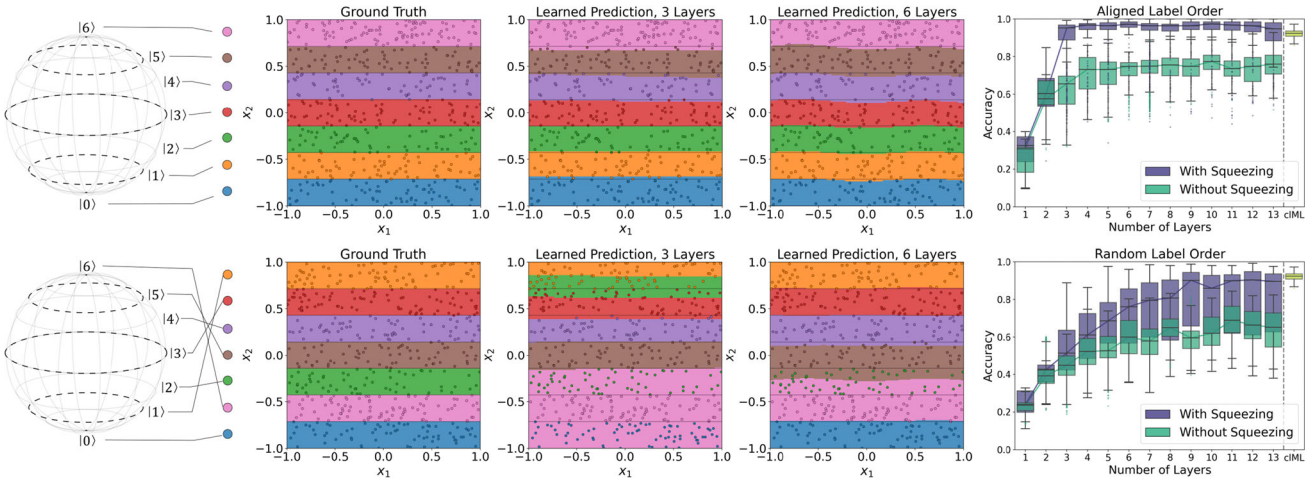

Fig. 3 Results of the seven-class horizontal stripe classification problem with circuit structure of Eqs. 6–8. In the upper row, the qudit states are aligned with the class labels, while in the lower row, the class labels are randomly assigned to the qudit states as illustrated in the pictures on the left. The first stripe pattern shows the ground truth and the training data samples. The middle and right stripe patterns show the classification regions obtained after training with three and six layers, respectively. The rightmost plots show the statistics of classification accuracy, obtained from 50 independent training runs, as a function of layers in the quantum circuit with (blue) and without (green) the squeezing operation in the W blocks. Colored boxes indicate 25% and 75% percentiles and horizontal whiskers the minimal and maximal values. In the right panels, cML indicates the results of the classical machine learning approaches

is consistently higher. The overall performance was slightly reduced for the tilted stripes and the concentric rings, and the variance in the results was also slightly larger as compared to the horizontal stripes cases presented here.

5.3 Classifying MNIST data

As the next application example, we train the model of Eqs. 6–8 on subsets of a `scikit-learn` (Pedregosa et al, 2018) version of the MNIST handwritten digits data set (LeCun and Cortes, 2005). This version includes down-sampled images with 8x8 pixels instead of the 28x28 pixels of the original MNIST data set. Since it is very computationally demanding to encode $8 \times 8 = 64$ dimensional data samples into the quantum circuit, we reduce the input dimension further using the principal component analysis (PCA) (Jolliffe and Cadima, 2016).

In Fig. 4a–c, we visualize the classification boundaries of a $L = 5$ layer circuit for two, three, and five classes, where the input dimension is reduced to $D = 2$ using a PCA. The plots show that the data re-uploading classifier is able to learn highly non-linear decision boundaries. It can also be seen that the classifier tends to produce disconnected classification regions due to the oscillatory nature of parametrized quantum circuits. The statistics of the accuracy for various classes is shown in Fig. 4d as a function of the quantum circuit layers. It can be observed that the accuracy, as expected, increases with more layers and that increasing the number of classes in the problem reduces the accuracy. For more classes, the reduction to two dimensions using PCA leads to more spacial

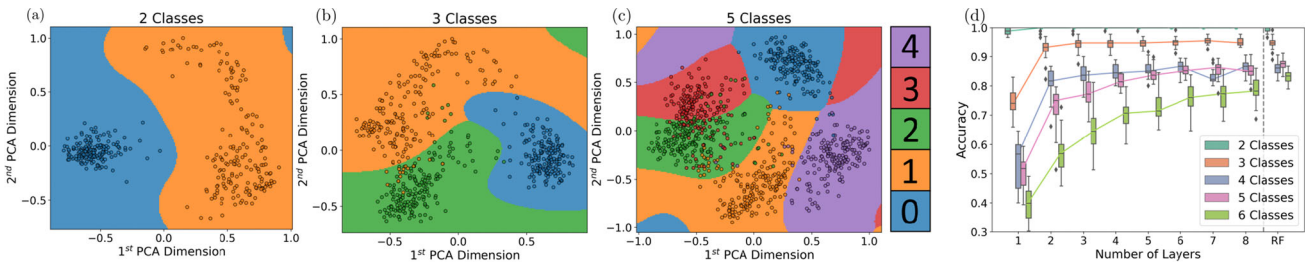

Fig. 4 Learned classification regions of a single qudit classifier circuit with $L = 5$ layers trained on the first two (**a**), three (**b**), and five (**c**) classes (i.e., digits) of the MNIST data set, which was compressed down to two dimensions using PCA. Each background color indicates one region associated with the same label. The training data samples are shown in the plots and colored according to their true label. **d** Accuracy of the qudit classifier as a function of the number of layers for various numbers of classes. RF denotes the random forest classifier for comparison

overlap between classes, which makes the problems inherently noisy and limits the overall achievable accuracy. The overlapping classes can be directly observed in panel (c). Additionally, panel (d) shows the result from 50 runs of a random forest (RF) classifier. The accuracy of the quantum-based classifier approaches the values of the RF model with increasing number of layers. However, the variance in the result is significantly larger for the data re-uploading circuit. One reason for this is found in the increasingly more complex classical optimization problem when increasing the number of parameters in the quantum circuit.

5.4 Qudit vs qubit

The quantum circuits we employed up to this point used a single d-level qudit to solve classification problems with d classes, where each basis state encodes one class label. While it may appear less natural, multi-class classification problems can also be learned with a data re-uploading circuit operating with a single qubit (Pérez-Salinas et al., 2020). In those approaches, the different classes are represented by single-qubit quantum states which are chosen to be maximally orthogonal. Unless a two-class classification problem is considered, where the label states are $|0\rangle$ and $|1\rangle$, these states cannot be fully orthogonal to each other. For a $d = 6$ class example, we choose the eigenstates of the three spin operators, i.e., $|y\rangle \in \{|0\rangle, |1\rangle, \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle), \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle), \frac{1}{\sqrt{2}}(|0\rangle + i|1\rangle), \frac{1}{\sqrt{2}}(|0\rangle - i|1\rangle)\}$, as the maximally orthogonal label states. Note that in this situation, the overlap of Eq. 10 as a function of the labels is no longer a proper probability distribution since $\sum_y P(y) > 1$.

The natural question which arises is whether there is any difference when using qubit or qudit data re-uploading circuits for multi-class classification problems. Therefore, we study and compare the performance of a single qubit as well as a single qudit on several problems. For the single qubit approach, we utilize the simplified data re-uploading structure of Eq. 9. Here, the squeezing operator can be removed from the circuit since it is proportional to the identity, $L_{z^2} = \frac{\mathbb{1}}{4}$, and therefore only applies a global phase.

Figure 5 shows the results where the qubit and qudit models were trained on two different classification problems with $d = 6$ classes and $D = 2$ input dimensions. The left panel of Fig. 5 shows the classification accuracy as a function of the circuit layers for the tilted stripes data set, where the horizontal stripes from Fig. 3 are rotated by an angle of 27°. It can be seen that the accuracy of the qubit and qudit circuits are comparable for the case where the class labels were randomly assigned to the qudit basis states. However, for labels aligned with the qudit states, the performance is consistently higher for the qudit circuits. This result is similar to the one obtained for the horizontal stripe data set reported in Fig. 3 and can be explained by the ladder structure employed in the qudit circuits. This gives rise to the inductive bias and leads to a performance improvement in case the structure of the data encoding operators is aligned with the data. This view is supported by a slight variation of the setup: we consider aligned labels and basis states but replace the L_x operator in the encoding scheme with a randomized version, $\tilde{L}_x = \sum_k (L_{\mathcal{P}(k+1),\mathcal{P}(k)} + L_{\mathcal{P}(k-1),\mathcal{P}(k)})$. Here, $\mathcal{P}$ denotes a random permutation of the label set $k = 0, \ldots, d-1$ and $L_{k,k'}$ generates a transition from state k' to state k. These operators do not implement a ladder structure, since the $\tilde{L}_x$ generate transitions between random states. We indeed observe in Fig. 5 that the median performance then decreases to the similar level as for the situation with a ladder structure, but randomized labels.

The right panel shows the results of the qubit and qudit data re-uploading circuits for the MNIST handwritten digits data set limited to six randomly selected digits (i.e., classes) and where each image is reduced to two input dimensions using a PCA. Here, qubit and qudit approaches are compara-

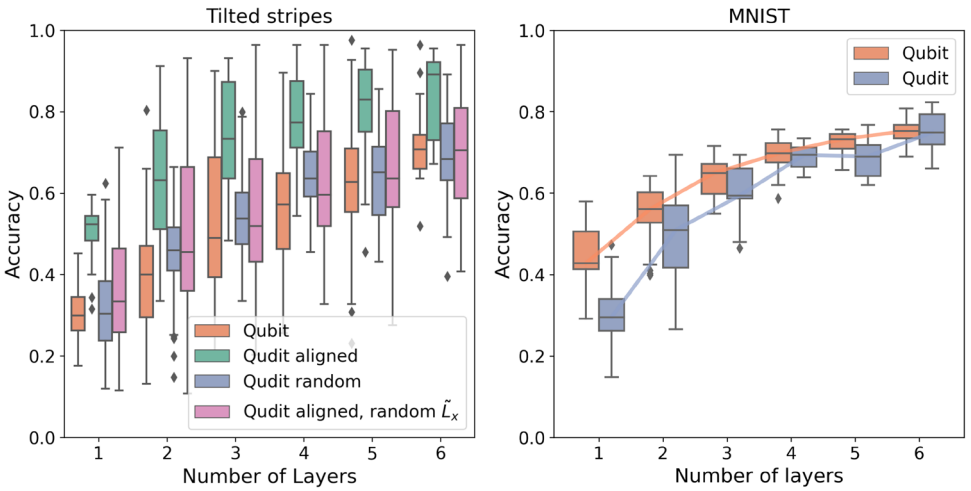

Fig. 5 Comparison of $d = 6$ multi-class classification performance of qubit and qudit data re-uploading circuits with the simplified structure of Eq. 9. Left: tilted $D = 2$ dimensional stripes data set with a rotation angle of 27°. Qudit accuracies are shown for aligned and randomized label and state assignment, as well as with aligned labels and qudit states, but with a randomized $\tilde{L}_x$ in the encoding. Right: six randomly selected digits of the MNIST data set which are reduced to $D = 2$ input dimensions using a PCA

ble in their performance, while the qubit architectures seem to slightly outperform the qudit ones, especially for smaller circuit depths. In this problem, the decision boundaries between the different classes are highly non-linear as discussed before. Therefore, there is no apparent favorable alignment between qudit basis states and class labels, and consequently, there is also no inductive bias which could lead to an improved qudit performance.

In our simulations, the loss function and the classification accuracy are calculated numerically exact with full access to the quantum state. When running on real quantum hardware, this cannot be done, and one needs to prepare and sample, i.e., measure, the quantum state multiple times in order to estimate the probabilities/overlaps of Eq. 10. For the qudit system, where each class label is associated with an orthogonal basis state in the measurement basis, the statistics of the measurement outcomes directly translate to estimating the probabilities. For qubit systems with non-orthogonal label states (i.e., for $d > 2$), different measurement protocols such as quantum state tomography or quantum state discrimination (Barnett and Croke, 2009) need to be employed. For fixed number for measurement shots, these lead to a slightly increased error probability for discriminating between non-orthogonal states. Additionally, the difference in overlap between a true label state and an undesired label state is less pronounced when different label states have a finite overlap. For qudits, this difference is always one since $P_{\text{qudit}}(y|y') = \delta_{y,y'}$ for label states y, y' (here, basis states). In the qubit representation, the difference is ≤ 1 since $P_{\text{qubit}}(y|y') = \delta_{y,y'} + \sum_{y'' \neq y'} c_{y''} \delta_{y,y''}$ with $c_y \geq 0$. For example, in the case of $d = 6$ classes considered here, $P_{\text{qubit}}(0|y') = |\langle 0|y'\rangle|^2 = \frac{1}{2}$ for $y' \in \{2, 3, 4, 5\}$. This can lead to a reduced training signal in the loss functions using the overlap of Eq. 10, since even contributions from incorrect label states have non-zero overlap and therefore reduce the loss. Obtaining the overlaps using a finite number of measurement samples necessarily adds noise to the estimates, which in turn makes the problem of discriminating between correctly and wrongly predicted labels more difficult. As a consequence, the shot noise is expected to have a more pronounced negative effect on the training as well as prediction performance for multi-class problems when overlapping label states are used.

5.5 Circuit structure and basic operators

For qubit circuits, the spin-$\frac{1}{2}$ Pauli matrices allow to represent arbitrary single-qubit unitary operations inside each layer. For d-level qudits, one instead needs to include all $d^2 - 1$ generators of the special unitary group in d dimensions $SU(d)$, to achieve the same arbitrary control. However, as indicated above and shown in Kasper et al. (2022) and Giorda et al. (2003), the three operators L_x, L_z, and L_{z^2} are sufficient to represent any unitary operation by repeated finite rotations with multiple layers. This leads to the question whether there is a benefit when more than this reduced set of three operators are used in the data re-uploading circuit. We test this hypothesis by adding the following two types of operators:

$$\begin{aligned} X_j &= |0\rangle\langle 0| - |j\rangle\langle j| \qquad (1 \leq j \leq d-1) \\ Y_j &= |0\rangle\langle j| + |j\rangle\langle 0| \,. \end{aligned} \tag{20}$$

The motivation behind this choice is that these operators directly couple the initial state $|0\rangle$ to all other states $|j\rangle$ and thus may allow for a more efficient learning. We then use the simplified structure of Eq. 9 and replace the squeezing operator by the sum over all operators in Eq. 20.

The results of the three types of circuit structures on the six-class reduced MNIST data set are shown in Fig. 6. The left panel shows the accuracies as a function of the number of circuit layers. First, one can observe that there is no significant difference between the simplified structure of Eq. 9 and circuit structure of Eqs. 6–7. On the other hand, the performance of the circuits with the extended set of operators is

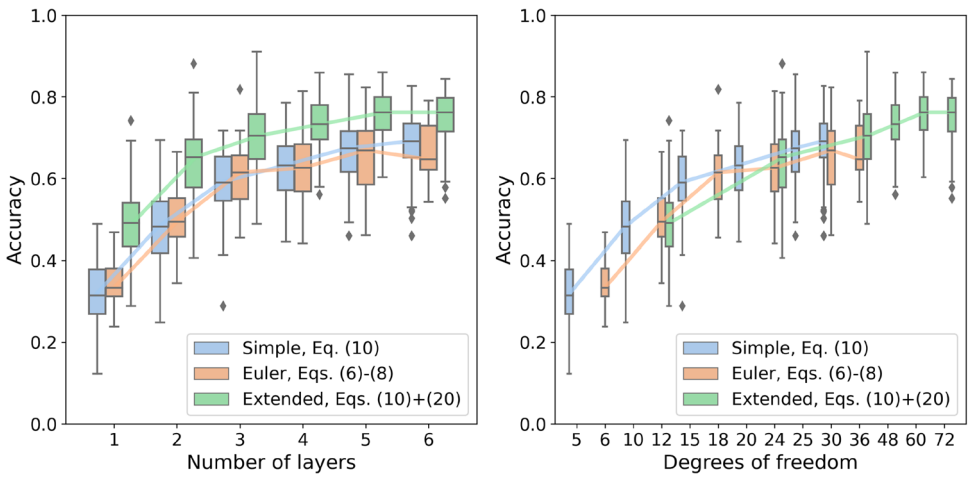

Fig. 6 Comparison of results of qudit data re-uploading circuits of different structures on the $d = 6$ class MNIST data set with input dimension $D = 2$. The left panel shows the results as function of the number of layers, while the right panel shows it as function of the degrees of freedom, i.e., the number of tunable parameters in the circuit (notice: the x-axis is not to scale in the right plot)

significantly better. At first glance, this seems to support the hypothesis that adding more operators to the set generators does in fact enhance the trainability of qudit quantum circuits. However, adding more operators also introduces more free parameters to the quantum circuit for a given number of layers, which allows for more flexibility in the trainable circuit. When comparing the accuracies as a function of number of free parameters, as shown in the right panel of Fig. 6, one can observe that the performance is statistically the same for all circuit structures and number of operators. Therefore, we conclude that it is not the number of operators which is determining the trainability, but rather the number of trainable parameters.

This reveals a tradeoff between the complexity of each layer, i.e., the number of operators and free parameters in each layer, and the total number of layers which are necessary to achieve a certain accuracy. Including more operators in each layer allows to achieve good performance with fewer number of layers. On quantum hardware, where gate errors play an important role, the freedom to choose the elemental operations provides additional flexibility. And because different hardware may natively support different gate sets, this allows to choose configurations which result in running variational quantum circuits with less errors. For qubits, this is not possible since the three Pauli matrices form a basis and no additional operators can be constructed.

5.6 Re-training on IBM hardware

Finally, we compare the classification performance of the model of Eqs. 6–8 with and without the squeezing operation between numerically exact results and the case where the re-training and evaluation is done on actual `ibmq_lima` hardware. For the qubit implementation of the qudit circuits, we use the Dicke-state encoding described in Sect. 4. The left panel of Fig. 7 shows the accuracy as a function of the quantum circuit layers for the first five digits, i.e., qudit dimension $d = 5$, and with input dimension $D = 5$. The numerically exact simulations show the same trend as previously described for two input dimensions and approach the values of the traditional RF classifier with an increasing circuit depth. However, when re-trained and evaluated on actual hardware, the performance is only comparable for the first two layers; after that, it saturates and then decreases strongly. This is due to the noise and infidelities in the actual hardware realization of the entangling gates used to implement the squeezing operations. Removing the squeezing operations from the circuit reveals this explicitly, since the accuracies of both approaches are then comparable as shown in the right panel of Fig. 7.

This is an explicit example of a situation where is more efficient to increase the local Hilbert space and work with qudits, instead of increasing the number of qubits.

6 Discussion

In this work, we demonstrated that multi-level qudit systems are well-suited to be applied to multi-class classification problems, as each qudit basis state can naturally encode one class of the data. We implemented data re-uploading quantum circuits, where we used the angular momentum and the squeezing operators to build a universal gate set. We illustrated the capabilities of the qudit-based approach on regression and classification benchmarks. Owing to the ability to learn highly non-linear classification boundaries, the models were able to successfully learn on various data sets and achieve performances comparable to standard classical machine learning models. Interestingly, the achievable performance was strongly dependent on the qudit states representing the class labels and their relation to the structure of the labels in the data set. This intrinsic bias due to the label alignment can boost the performance of qudit circuits substantially, when the data set is structured accordingly, which might be beneficial for certain types of application problems.

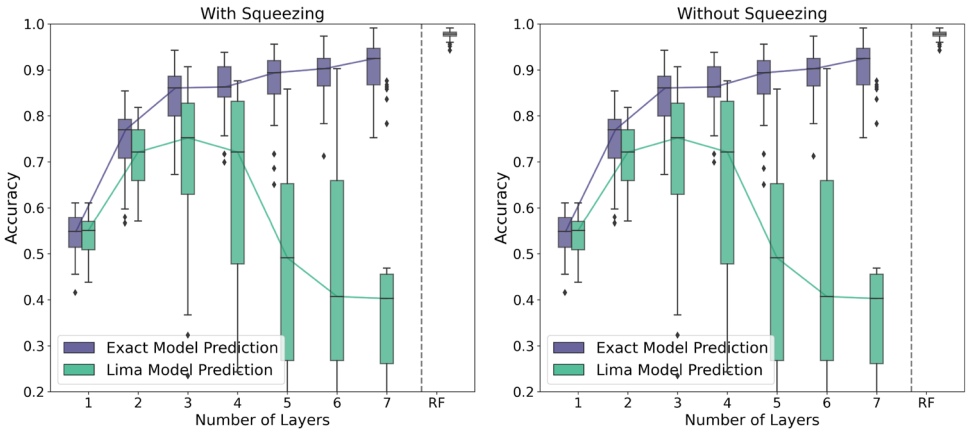

Fig. 7 Results for the classification of the first five digits of the MNIST data set with input dimension $D = 5$ with (left) and without (right) squeezing as a function of the layers in the quantum circuit operating with a $d = 5$ qudit. The numerical exact simulations are shown in blue, whereas the results from `ibmq_lima` simulations are in green. The performance of a random forest (RF) model is displayed to the right in each plot

We also studied the influence of the choice of the elementary operations and the layer structure in qudit quantum circuits. There, we found that the structure, i.e., the particular sequence and types of rotations, does not appear to have significant influence on the performance, as long as the number of degrees of freedom was accounted for. This reveals a trade-off between the number of elementary operators in the gate set and the number of layers to achieve the same performance. In some situations, it might be advantageous to utilize the minimum set of operators and to employ more layers, while in others, more operators and less layers might be the better choice. This is especially interesting on quantum hardware where different gates typically have different error rates, and the freedom to choose between several configurations may allow to minimize the influence of errors on the resulting performance.

However, our data encoding layers always employed only L_z and L_x operators and thus implemented a ladder structure. This is linked to the observed intrinsic bias, and selecting different operators for the data encoding might mitigate this bias. However, we did not investigate such extensions and leave them for future work.

It should be noted that we did not employ any of the more sophisticated techniques to improve the performance of single-qudit data re-uploading circuits, for example, employing different classical optimizers (Deller et al., 2023; Lavrijsen et al., 2020), fine-tuning hyperparameter settings (Moussa et al., 2022), and finding better parameter initializations (Sack and Serbyn, 2021; Grant et al., 2019; Egger et al., 2021). All techniques have been shown to be very beneficial in related contexts and can be used in the future to improve the current approach.

A necessary next step is to investigate the performance of multi-qudit circuits, since here we only investigated single-qudit circuits. Interesting research questions include how the intrinsic bias of a single qudit influences a multi-qudit circuit and its learning performance, and what the role of the set of elementary operators is in these cases. Crucially, the effect of multi-qudit entangling gates needs to be elucidated.

In summary, our results and discussion support the conclusion that qudit systems offer a promising alternative quantum computing architecture. There are several differences to qubit-based systems which could potentially be leveraged to eventually provide practical benefits for quantum algorithms and quantum machine learning tasks in particular.

Acknowledgements NLW acknowledges funding from the Honda Research Institute Europe GmbH for attending the conference on Quantum Techniques in Machine Learning (QTML) 2022. NLW would like to thank the Van der Waals-Zeeman Institute in Amsterdam for their extended hospitality. SS acknowledges funding by the European Union under Horizon Europe Programme—Grant Agreement 101080086—NeQST. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Climate, Infrastructure and Environment Executive Agency (CINEA). Neither the European Union nor the granting authority can be held responsible for them.

Author Contributions FJ and NLW conceived the original idea on the simulations; NLW, FJ, and SS conduced the numerical simulations; NLW, MSR, and SS analyzed the results data. All authors jointly wrote and reviewed the manuscript.

Availability of data and materials The datasets and numerical details necessary to replicate this work are available from the corresponding author upon reasonable request.

Declarations

Conflict of interest The authors declare no competing interests.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/.

References

Barnett SM, Croke S (2009) Quantum state discrimination. Adv Opt Photon 1(2):238. https://doi.org/10.1364/AOP.1.000238

Bharti K, Cervera-Lierta A, Kyaw TH, et al. (2022) Noisy intermediate-scale quantum algorithms. Rev Mod Phys 94(1):015,004. https://doi.org/10.1103/RevModPhys.94.015004

Biamonte J, Wittek P, Pancotti N et al (2017) Quantum machine learning. Nat 549(7671):195–202. https://doi.org/10.1038/nature23474

Bradbury J, Frostig R, Hawkins P, et al. (2018) JAX: composable transformations of Python+NumPy programs. http://github.com/google/jax

Bravyi S, Kliesch A, Koenig R, et al. (2022) Hybrid quantum-classical algorithms for approximate graph coloring. Quantum 6:678. https://doi.org/10.22331/q-2022-03-30-678

Cozzolino D, Da Lio B, Bacco D, et al. (2019) High dimensional quantum communication: benefits, progress, and future challenges. Adv Quant Technol 2(12):1900,038. https://doi.org/10.1002/qute.201900038

Deller Y, Schmitt S, Lewenstein M, et al. (2023) Quantum approximate optimization algorithm for qudit systems. Phys Rev A 107(6):062,410. https://doi.org/10.1103/PhysRevA.107.062410

Dunjko V, Briegel HJ (2018) Machine learning & artificial intelligence in the quantum domain: a review of recent progress. Rep Prog Phys 81(7):074,001. https://doi.org/10.1088/1361-6633/aab406

Dutta T, Pérez-Salinas A, Cheng JPS, et al. (2022) Single-qubit universal classifier implemented on an ion-trap quantum device. Phys Rev A 106(1):012,411. https://doi.org/10.1103/PhysRevA.106.012411

Egger DJ, Mareček J, Woerner S (2021) Warm-starting quantum optimization. Quantum 5:479. https://doi.org/10.22331/q-2021-06-17-479

Fedorov AK, Gisin N, Beloussov SM, et al. (2022) Quantum computing at the quantum advantage threshold: a down-to-business review. arXiv

Fischer LE, Chiesa A, Tacchino F, et al. (2022) Towards universal gate synthesis and error correction in transmon qudits. arXiv

Gasieniec LA, Bärtschi A, Eidenbenz S (2019) Deterministic preparation of Dicke states. In: Jansson J, Levcopoulos C (eds) Fundamentals of computation theory, vol 11651. Springer International Publishing, Cham, p 126–139, series Title: Lecture notes in computer science. https://doi.org/10.1007/978-3-030-25027-0_9

Giorda P, Zanardi P, Lloyd S (2003) Universal quantum control in irreducible state-space sectors: application to bosonic and spin-boson systems. Phys Rev A 68(6):062,320. https://doi.org/10.1103/PhysRevA.68.062320

Graham TM, Song Y, Scott J et al (2022) Multi-qubit entanglement and algorithms on a neutral-atom quantum computer. Nature 604(7906):457–462. https://doi.org/10.1038/s41586-022-04603-6

Grant E, Wossnig L, Ostaszewski M, et al. (2019) An initialization strategy for addressing barren plateaus in parametrized quantum circuits. Quantum 3:214. https://doi.org/10.22331/q-2019-12-09-214

Gyurik C, Dunjko V (2022) On establishing learning separations between classical and quantum machine learning with classical data. https://doi.org/10.48550/arXiv.2208.06339

Gyurik C, Cade C, Dunjko V (2022) Towards quantum advantage via topological data analysis. Quantum 6:855. https://doi.org/10.22331/q-2022-11-10-855

Husimi K (1940) Some formal properties of the density matrix. https://doi.org/10.11429/ppmsj1919.22.4_264

Jerbi S, Fiderer LJ, Poulsen Nautrup H et al (2023) Quantum machine learning beyond kernel methods. Nat. Commun. 14(1):517. https://doi.org/10.1038/s41467-023-36159-y

Jolliffe IT, Cadima J (2016) Principal component analysis: a review and recent developments. Philos Trans Royal Soc A: Math Phys Eng Sci 374(2065):20150,202. https://doi.org/10.1098/rsta.2015.0202

Kasper V, González-Cuadra D, Hegde A, et al. (2022) Universal quantum computation and quantum error correction with ultracold atomic mixtures. Quantum Sci Technol 7(1):015,008. https://doi.org/10.1088/2058-9565/ac2d39

Kingma DP, Ba J (2014) Adam: a method for stochastic optimization. https://doi.org/10.48550/ARXIV.1412.6980

Kjaergaard M, Schwartz ME, Braumüller J et al (2020) Superconducting qubits: current state of play. Ann Rev Condens Matter Phys 11(1):369–395. https://doi.org/10.1146/annurev-conmatphys-031119-050605

Lavrijsen W, Tudor A, Muller J, et al. (2020) Classical optimizers for noisy intermediate-scale quantum devices. In: 2020 IEEE International Conference on Quantum Computing and Engineering (QCE). IEEE, Denver, CO, USA, pp 267–277. https://doi.org/10.1109/QCE49297.2020.00041

LeCun Y, Cortes C (2005) MNIST handwritten digit database. http://yann.lecun.com/exdb/mnist/

Liu Y, Arunachalam S, Temme K (2021) A rigorous and robust quantum speed-up in supervised machine learning. Nat Phys 17(9):1013–1017. https://doi.org/10.1038/s41567-021-01287-z

Luo MX, Chen XB, Yang YX et al (2014) Geometry of quantum computation with qudits. Sci Rep 4(1):4044. https://doi.org/10.1038/srep04044

Montanaro A (2016) Quantum algorithms: an overview. npj Quantum Inf 2(1):15,023. https://doi.org/10.1038/npjqi.2015.23

Moussa C, van Rijn JN, Bäck T, et al. (2022) Hyperparameter importance of quantum neural networks across small datasets. In: Pascal P, Ienco D (eds) Discovery Science, vol 13601. Springer Nature Switzerland, Cham, p 32–46, series Title: Lecture notes in computer science. https://doi.org/10.1007/978-3-031-18840-4_3

Pedregosa F, Varoquaux G, Gramfort A, et al. (2018) Scikit-learn: machine learning in Python. https://doi.org/10.48550/arXiv.1201.0490

Pino JM, Dreiling JM, Figgatt C et al (2021) Demonstration of the trapped-ion quantum CCD computer architecture. Nat 592(7853):209–213. https://doi.org/10.1038/s41586-021-03318-4

Pérez-Salinas A, Cervera-Lierta A, Gil-Fuster E, et al. (2020) Data re-uploading for a universal quantum classifier. Quantum 4:226. https://doi.org/10.22331/q-2020-02-06-226

Resch S, Karpuzcu UR (2019) Quantum computing: an overview across the system stack. arXiv

Ringbauer M, Meth M, Postler L et al. (2022) A universal qudit quantum processor with trapped ions. Nat Phys 18(9):1053–1057. https://doi.org/10.1038/s41567-022-01658-0

Rudolph MS, Sim S, Raza A, et al. (2021) ORQVIZ: visualizing high-dimensional landscapes in variational quantum algorithms. https://doi.org/10.48550/ARXIV.2111.04695

Sack SH, Serbyn M (2021) Quantum annealing initialization of the quantum approximate optimization algorithm. Quantum 5:491. https://doi.org/10.22331/q-2021-07-01-491

Santra GC, Jendrzejewski F, Hauke P, et al. (2022) Squeezing and quantum approximate optimization. arXiv

Schuld M (2021) Supervised quantum machine learning models are kernel methods. https://doi.org/10.48550/arXiv.2101.11020

Schuld M, Killoran N (2022) Is quantum advantage the right goal for quantum machine learning? PRX Quantum 3(3):030,101. https://doi.org/10.1103/PRXQuantum.3.030101

Schuld M, Petruccione F (2018) Supervised learning with quantum computers. Quantum science and technology. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-96424-9

Schuld M, Sweke R, Meyer JJ (2021) Effect of data encoding on the expressive power of variational quantum-machine-learning models. Phys Rev A 103(3):032,430. https://doi.org/10.1103/PhysRevA.103.032430

Sheridan L, Scarani V (2010) Security proof for quantum key distribution using qudit systems. Phys Rev A 82(3):030,301. https://doi.org/10.1103/PhysRevA.82.030301

Sweke R, Seifert JP, Hangleiter D, et al. (2021) On the quantum versus classical learnability of discrete distributions. Quantum 5:417. https://doi.org/10.22331/q-2021-03-23-417

Treinish M, Gambetta J, Thomas S, et al. (2023) Qiskit/qiskit: Qiskit 0.41.0. https://doi.org/10.5281/ZENODO.2573505

Virtanen P, Gommers R, Oliphant TE, et al. (2020) SciPy 1.0: fundamental algorithms for scientific computing in Python. Nat Methods 17(3):261–272. https://doi.org/10.1038/s41592-019-0686-2

Wang Y, Hu Z, Sanders BC et al (2020) Qudits and high-dimensional quantum computing. Front Phys 8(589):504. https://doi.org/10.3389/fphy.2020.589504

Weggemans JR, Urech A, Rausch A, et al. (2022) Solving correlation clustering with QAOA and a Rydberg qudit system: a full-stack approach. Quantum 6:687. https://doi.org/10.22331/q-2022-04-13-687

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.