Dimensionality reduction with variational encoders based on subsystem purification

Raja Selvarajan[1], Manas Sajjan[2], Travis S. Humble[3], and Sabre Kais[1,2,4,5]

[1]Purdue University, Department of Physics and Astronomy, West Lafayette, IN 47907, USA
[2]Purdue University, Department of Chemistry, West Lafayette, IN 47907, USA
[3]Oak Ridge National Laboratory (ORNL), Oak Ridge, TN 37830
[4]Purdue Quantum Science and Engineering Institute, West Lafayette, IN 47907, USA
[5]Corresponding author: Sabre Kais, kais@purdue.edu

Efficient methods for encoding and compression are likely to pave way towards the problem of efficient trainability on higher dimensional Hilbert spaces overcoming issues of barren plateaus. Here we propose an alternative approach to variational autoencoders to reduce the dimensionality of states represented in higher dimensional Hilbert spaces. To this end we build a variational based autoencoder circuit that takes as input a dataset and optimizes the parameters of Parameterized Quantum Circuit (PQC) ansatz to produce an output state that can be represented as tensor product of 2 subsystems by minimizing $Tr(\rho^2)$. The output of this circuit is passed through a series of controlled swap gates and measurements to output a state with half the number of qubits while retaining the features of the starting state, in the same spirit as any dimension reduction technique used in classical algorithms. The output obtained is used for supervised learning to guarantee the working of the encoding procedure thus developed. We make use of Bars and Stripes dataset (BAS) for an 8x8 grid to create efficient encoding states and report a classification accuracy of 95% on the same. Thus the demonstrated example shows a proof for the working of the method in reducing states represented in large Hilbert spaces while maintaining the features required for any further machine learning algorithm that follow.

Sabre Kais: *kais@purdue.edu

1 Introduction

Variational quantum algorithms in the NISQ [15] era provides a promising route towards developing useful algorithms that allow for optimizing states in higher dimensional spaces by tuning polynomial number of parameters. The most prominent techniques within variational methods include Variational Quantum Eigensolver (VQE) [14], Quantum Approximate Optimization Algorithm (QAOA) [9] and other classical machine learning inspired ones. We ask the readers to refer [18] for an exhaustive study on quantum machine learning with applications in chemistry [17, 26, 11], physics [19],supervised image classification [6] and optimization [20]. Within the context of optimization and machine learning in general, some of the major problems that needs to be addressed includes encoding classical data, finding an expressible enough ansatz (Expressibility) [21], efficiently computing gradients(Trainability) [7], generalizability [1]. These problems are interlinked and thus not treated independently in general.

As we move away from the NISQ era towards deep Parameterized Quantum Circuits (PQC), one of the major problems with regards to trainability that needs addressing is the problem of vanishing gradients referred to as barren plateaus [13]. This might be an affect of working with large number of qubits [13], expressive circuit ansatz [10], noise induced[23] or the use of global cost functions in the learning [3]. Having efficient procedures to reduce the dimensionality of input quantum state representation will pave a path in developing efficient encoding schemes that could later be used as inputs to other machine learning algorithms where the cost functions on higher di-

mensional spaces with expressive ansatz are less likely to be trainable. To this end we develop machine learning techniques that allow for compact representations of given input quantum state.

Within the classical machine learning community autoencoders have been effectively used to develop low dimensional representation of samples generated from a given probability distributions [12]. Inspired from these techniques work on Quantum Autoencoders [16, 22] have allowed for people to develop compact representations against a fixed finite state. It is not clear that such tensor product states with a fixed finite state is always possible and retains the maximal possible information. Here we show that if one were to relax the condition towards maintaining a fixed finite state, a better compact representation can be generated that can be post processed towards classification. We develop techniques to create subsystem purifications for a given set of inputs, and follow it by creating superpositions of these purifications indexed by the subsystem number. This representation is further used for doing classification achieved by applying variational methods over parameterized quantum circuits restricted to this compact representation and show the learning of the method. We apply an ansatz to create subsystem purification on Bars and Stripes (BAS) dataset and show that one can reduce the number of qubits required to represent the data by half and achieve a 95% classification accuracy on the Bars and Stripes (BAS) dataset. The demonstrated example shows a proof for the working of the method in reducing states represented in large Hilbert spaces while maintaining the features required for any further machine learning algorithm that follow. The scheme thus proposed can be extended to problems with states in large Hilbert spaces where dimensionality reduction plays a key role with regards to the trainability of the parameterized quantum circuit.

2 Method

Given an ensemble of input states, $E = \{|\psi_i\rangle\}$, the objective is to construct a low dimensional representation of states sampled from this distribution E. Let $|\psi_i\rangle$ be a state over $n_A + n_B$ qubits. We design a protocol that allows for us to create an equivalent compact representation of $|\psi\rangle$ with

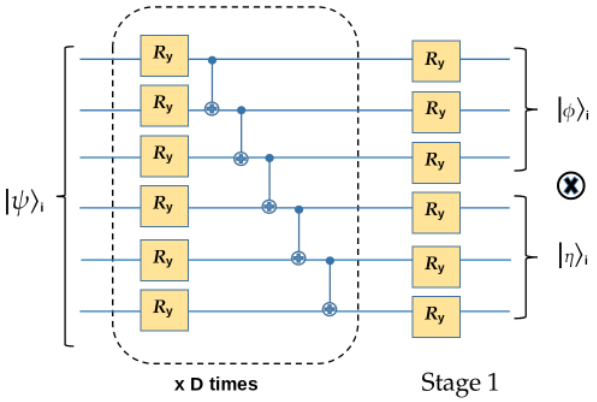

Figure 1: Ansatz used for Encoding circuit $U(\vec{\theta})$ in stage 1. The circuit shows D repeating layers of a unit consisting of R_y gates parameterized by one independent angle each and a ladder of $CNOT$ operators. The circuit is optimized over the dataset to generate equivalent states with subsystem tensor product structure. Thus we obtain $U(\vec{\theta})|\psi\rangle = |\phi\rangle \otimes |\eta\rangle$. ϕ is first subsystem and shall be indexed later with an ancilla state $|0\rangle$ and $|\eta\rangle$ is the second subsystem which shall be indexed by $|1\rangle$

$max(n_A, n_B) + 1$ qubits. To simplify the discussion lets assume that $n_A = n_B$ and thus we create a representation using half the qubits. We do this in 2 stages.

Stage 1:

In the first stage we apply a unitary $U(\vec{\theta})$ that decomposes $|\psi_i\rangle_{A,B}$ into $|\alpha(\theta)_i\rangle_A \otimes |\beta(\theta)_i\rangle_B$. To produce such a tensor product structure we could minimize the entropy on either of subsystems A or B till we get zero. Thus we could optimize over the cost function,

$$C_B^1(\vec{\theta}) = \left\langle S\Big(tr_A\Big[U(\vec{\theta})\,|\psi\rangle_{AB}\,\langle\psi|_{AB}\,U^\dagger(\vec{\theta})\Big]\Big)\right\rangle_{\{|\psi\rangle\}} \tag{1}$$

where tr_A represents the tracing operation over the qubits of subsystem A, $< . >_{\{|\psi\rangle\}}$ represents the averaging over the $\{|\psi\rangle\}$ and $S(\rho) = tr(\rho log(\rho))$ is the entropy of a given density matrix ρ.The cost function $C_B(\vec{\theta})$ attains a maximum value equal to $log(n_B)$ when ρ_B is maximally mixed, and equal to 0 when ρ_B is a pure state. Fig 1 shows a schematic representation of the ansatz used for $U(\theta)$.

Variational quantum algorithms have been studied in the past towards creating thermal systems by minimizing the output state against the free energy [24, 4]. The main problem tackled in

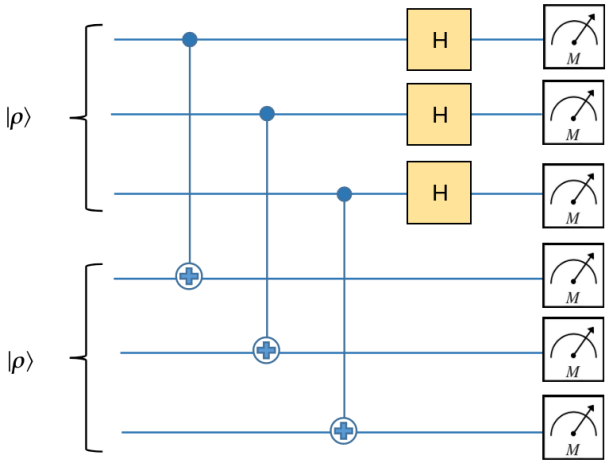

Figure 2: Quantum circuit above implements destructive swap test. Given 2 different density matrices as inputs the circuit computes fidelity of states $F(\gamma,\sigma) = (Tr(\sqrt{\sqrt{\gamma}\sigma\sqrt{\gamma}})^2$, where γ and σ are 2 density matrices. Here we use $\gamma = \sigma = |\rho\rangle\langle\rho|$. Post-processing of measurements with input as 2 copies of $|\rho\rangle$ is used to compute $Tr(\rho^2)$ [5]

these papers involves developing techniques that allows one to compute the gradients of Entropy required to be optimized over the training. The issue arises from not having exact representations that can compute logarithm of given density matrix efficiently. Further more to avoid numerical instabilities in the entropy function arising from the density matrix of pure states being singular, here we alternatively maximize over the cost function,

$$C_{AB}(\vec{\theta})) = \left\langle Tr_A({\rho_A}^2) + Tr_B({\rho_B}^2) \right\rangle_{\{|\psi\rangle\}} \quad (2)$$

where $\rho_A = Tr_B(U(\vec{\theta})\,|\psi\rangle_{AB}\,\langle\psi|_{AB}\,U^\dagger(\vec{\theta}))$ and $\rho_B = Tr_A(U(\vec{\theta})\,|\psi\rangle_{AB}\,\langle\psi|_{AB}\,U^\dagger(\vec{\theta}))$. C_{AB} attains a maximum value 2 when ρ_A or ρ_B are pure states resulting in $Tr(\rho^2_{A/B}) = Tr(\rho_{A/B}) = 1$ and attains a least value $2/n$.

The parameters $\vec{\theta}$ are variationally optimized to obtain $\vec{\theta}^* = \text{argmax}_{\vec{\theta}}\ C_{AB}(\vec{\theta})$. If $C_{AB}(\vec{\theta})$ reaches an optimal value of zero, we can express $|\psi\rangle_{AB} = |\phi\rangle_A \bigotimes |\eta\rangle_B$, thus expressing a state with 2^{n+m} degrees of freedom effectively using $2^n + 2^m$ degrees of freedom. Having expressed the input state as a tensor product of subsystems we now move to stage 2 of the algorithm.

Stage 2:

Note that the above representation still makes use of $2n$ qubits to capture the features of $|\psi\rangle$. We now show how this representation can be compressed to using $n+1$ qubits. We show how using an additional ancillary qubit, amplitude amplification and projective measurements one can create the state $|0\rangle|\phi\rangle + |1\rangle|\eta\rangle$ starting from $\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)|\phi\rangle|\eta\rangle$. To do this we apply a CSWAP (controlled swap/Fredkin) gate acting on the qubits of system A and B. Thus we get $|0\rangle|\phi\rangle|\eta\rangle + |1\rangle|\eta\rangle|\phi\rangle$. If $|\eta\rangle$ and $|\phi\rangle$ are not orthogonal states, then there exists atleast one basis element $|g\rangle$ in the computational basis with a nonzero coefficient in both these states. Without a loss of generality lets assume that the measurement collapses onto $|g\rangle$ giving raise to $\frac{1}{\sqrt{1+c^2}}(|0\rangle|\phi\rangle + ce^{i\alpha}|1\rangle|\eta\rangle) \otimes |g\rangle$, where c and α are real numbers. The factor $ce^{i\alpha}$ is generated from the relative difference in the coefficients of the state corresponding to $|g\rangle$. To ensure that the state collapse to a specific garbage state $|g\rangle$, we could choose $|g\rangle$ to have the maximum probability among all the basis projections. The factor $ce^{i\alpha}$ can now be absorbed as a global normalization if the ancilla register was prepared in the state $\frac{1}{\sqrt{(1+c^2)}}(ce^{i\alpha}|0\rangle + |1\rangle)$. This ensures that the output of this stage is $|0\rangle|\phi\rangle + |1\rangle|\eta\rangle$. To extend this description to the case when $|\phi\rangle$ and $|\eta\rangle$ are orthonormal, one just needs to apply a transformation controlled on the ancilla register to break this condition. Fig 3 shows a schematic representation of the main steps involved in creating a superposition with the ancilla register being used as an index to the subsystem outputs of Stage 1.

Output:

Thus we have successfully managed to convert the input state $|\psi\rangle$ to $|0\rangle|\phi\rangle + |1\rangle|\eta\rangle$, as required. Note that this procedure is reversible and hence the representation is unique, thus preserving all information content encoded into input state $|\psi\rangle$. To show its reversible, one just needs to take 2 copies of the output state $|0\rangle|\phi\rangle + |1\rangle|\eta\rangle$, measure the corresponding ancilla to project out $|\phi\rangle|\eta\rangle$, and then apply the inverse of $U(\vec{\theta})$ giving back $|\psi\rangle$. Thus the encoding scheme allows for us to create a representation of input state $|\psi\rangle$ with $2n$ qubits into only $n+1$ qubits. This procedure can be repeated iteratively as long as the output state vectors permit a size reduction quan-

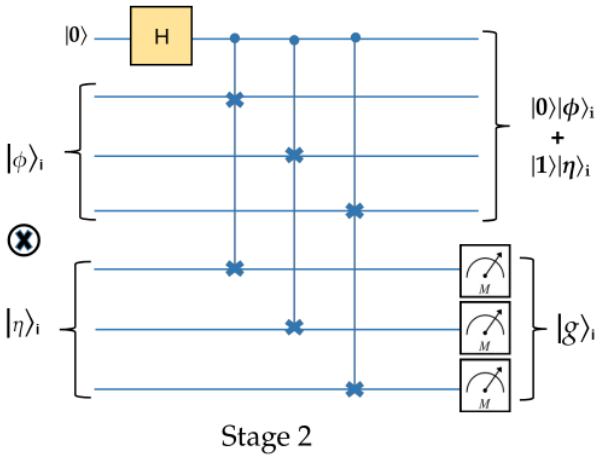

Figure 3: A schematic representation of the steps involved in Stage 2 to prepare the superposition state using an extra ancilla from the product state output of Stage 1. Controlled swap gates are used to generate $|0\rangle|\phi\rangle + |1\rangle|\eta\rangle$. Following this the second subsystem is measured in the computational basis imparting relative phase and amplitude (not shown in the above representation)

tified by the entropy. If repeated $log(n)$ times a $O(log(n))$ size qubit representation of the n qubit state is achievable. Such compact representations are very much reminiscent of the output representation of states on QRAM, where the features are put in a superposition with the index register working as ancillas.

3 Results

To demonstrate the working of the method described above, we pick a toy dataset with images of Bars and Stripes (BAS) and build a compact representation of it. The BAS dataset we consider is a square grid with either some columns being only vertically filled (Bars) or some rows being horizontally filled (Stripes) [2]. One can easily generate such a supervised dataset and realize that the distribution from which these images are sampled has a low entropy characterization. We randomly sample 1000 data points from a grid size of 16x16 BAS dataset consisting of 131068 datapoints represented using amplitude encoding on 8 qubits.

Applying the protocol described above we reduce the representation of the state into a tensor product of 2 subsystem of equal sizes. Fig 4,6 shows the learning of optimal parameters $\vec{\theta}$ as

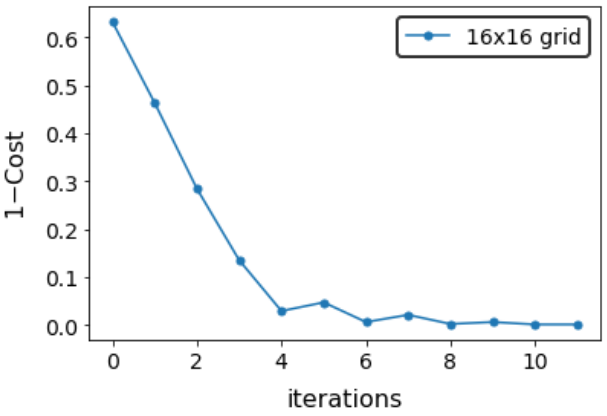

Figure 4: Stage 1: Training cost vs iterations for 16x16 grid.The unitary circuit thus trained creates equivalent tensor product representations using two equal half subsystems of 4 qubits. Note that $1-$Cost eventually saturates at 0 allowing us to create pure state product subsystem

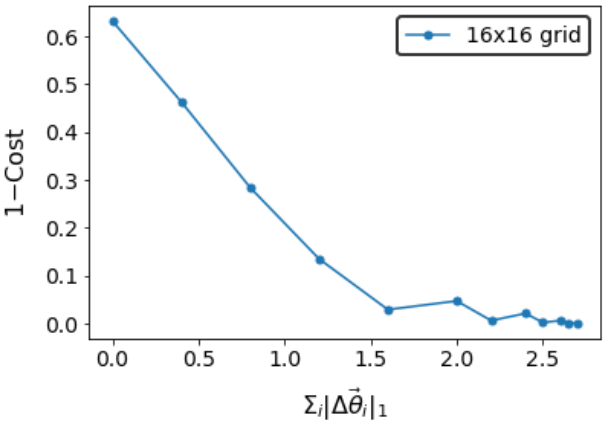

Figure 5: Stage 1: Training cost vs $\sum_i |\Delta\vec{\theta}_i|_1$ for 16x16 grid. The variation in angle as computed by the gradient of eqn 2 is minimized as one gets near to the saturation point ($|\Delta\vec{\theta}|_1$ measures the 1 norm increase in the angle contribution from the computed gradients with increasing epochs)

the cost function falls. We use standard gradient descent [25] approach in doing the training . Note the cost function drops to zero implying that the representation thus created is exact with a lossless transformation created by $U(\vec{\theta})$. For the 16x16 grid case, the ansatz $U(\vec{\theta})$ is made of D=5 layers, while for the 8x8 grid is made of D=3 layers. At this point we apply a layer of swap gates to reduce the 8 qubit representation of 16x16 grid samples into 5 qubits and the 6 qubit representation of 8x8 grid samples into 4 qubits.

We now use this as input for doing supervised classification. We use approx 80% of the samples from the output of the encoded samples for

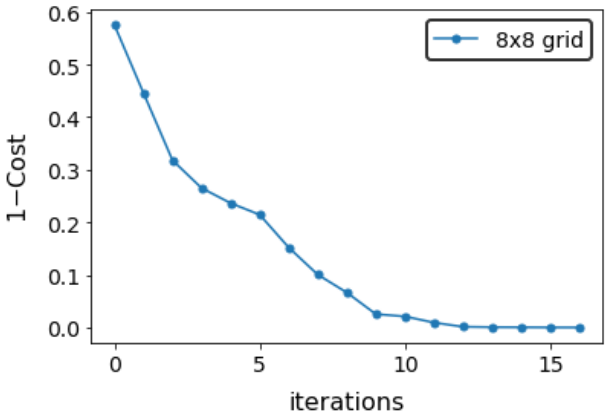

Figure 6: Stage 1: Training cost vs iterations for 8x8 grid. .The unitary circuit thus trained creates equivalent tensor product representations using two equal half subsystems of 3 qubits. Note that $1-$Cost eventually saturates at 0 allowing us to create pure state product subsystem

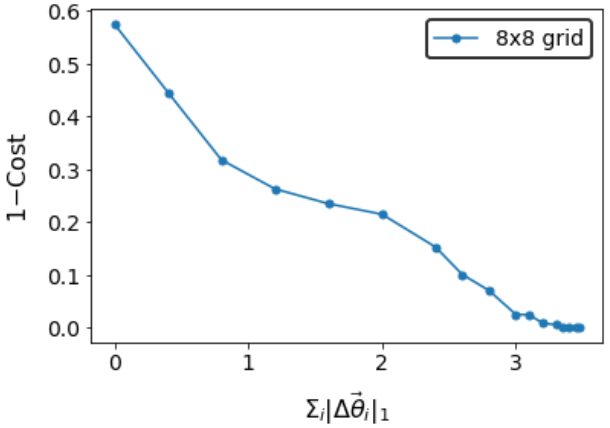

Figure 7: Stage 1: Training cost vs $\sum_i |\Delta\vec{\theta}_i|_1$ for 8x8 grid. The variation in angle as computed by the gradient of eqn 2 is minimized as one gets near to the saturation point ($|\Delta\vec{\theta}|_1$ measures the 1 norm increase in the angle contribution from the computed gradients with increasing epochs)

training and keep the remaining 20% of the samples for testing. An ansatz $V(\vec{\theta})$ with the same number of qubits as that of the input samples is trained, with the sign of expectation value of pauli-Z operator being used as a label for differentiating between bars and stripes. Input image is classified as a bars image if the expectation value is positive, and stripes image if negative. We use the sum of 2 norm errors over the dataset labels (1 for bars and -1 for stripes) as the cost function to be minimized over, i.e,

$$\text{cost} = \sum_i \left(l_i - \langle\tilde{\psi}_i| V^\dagger(\vec{\alpha})[Z \otimes I^{\otimes n-1}]V(\vec{\alpha}) |\tilde{\psi}_i\rangle\right)^2 \tag{3}$$

where the summation index i labels the dataset, l_i refers to the labels corresponding to the sample input and $|\psi_i\rangle$ is used to denote the compact representation of the state that the above encoding scheme provides. For the 8x8 grid, a total 508 bars and stripe images are produced with half of them belonging to each category. We use 400 of these samples for training and 108 samples for testing. Fig 8 shows the cost of optimizing the parameters of $V(\vec{\theta})$ as a function of the number of iterations. We get a 95% accuracy on the testing data, showing that the method use to generate the compact representation did not destroy the features of the input state.

4 Runtime analysis of Encoding scheme

Here we shall analytically compute the required runtime for the above described protocol. Lets assume that the input ensemble of N quantum states over n qubits supports a compact representation, allowing us to use the above protocol to encode with half the number of qubits. Let our ansatz to be optimized be made of d layers. Thus stage 1 involves optimization over $2ndN$ parameters for N samples. Using destructive swap test to compute fidelity with an error ϵ we would require $O(1/\epsilon^2)$ samples. Thus the runtime complexity scales evaluating $O(ndN/\epsilon^2)$ quantum circuits per iteration for Stage 1. Stage 2 involves projection onto the state with largest overlap. The overlap achievable onto any given computational basis $|g\rangle$ can be maximized using grovers with the worst case runtime of $O(2^{n/2})$ steps with query complexity of $O(1/\epsilon)$. Thus the overall runtime is bounded by $O(N(Tnd/\epsilon^2 + 2^{n/2}/\epsilon))$, where T is the number of iterations required in stage 1 optimization. In contrast the runtime of a classical autoencoder to prepare a compact state is $O(NTd2^n)$. We show in the appendix A, how for certain cost functions, using a specialized ansatz and carefully prepared index registers one can get around the exponential cost incurred in preparing the compact superposition state for machine learning tasks.

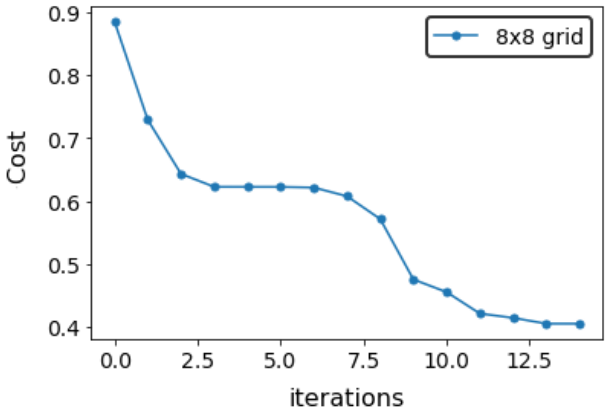

Figure 8: Classification cost vs iterations for 8x8 grid.Figure shows the saturation of classification cost as per eqn 3 after 13 iterations.

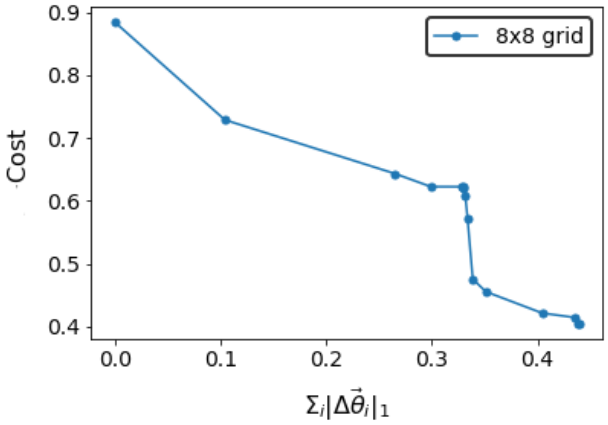

Figure 9: Classification cost vs $\sum_i |\Delta\vec{\theta}_i|_1$ for 8x8 grid. Figure shows that the variation in angle as computed by the gradient of eqn 3 is minimized as one gets near to the saturation point. ($|\Delta\vec{\theta}|_1$ measures the 1 norm increase in the angle contribution from the computed gradients with increasing epochs

5 Discussion and Conclusion

We discuss a scheme that allows for a compact representation of states in higher dimensional Hilbert spaces using half the number of qubits. The output thus created serves as good starting states for any further machine learning algorithm that might follow. The protocol is based on designing a quantum circuit that allows creating tensor product subsystems and demonstrate results on bars and stripes datasets for 8x8 grid and 16x16 grid. We further use this output to create compact representations with half the number of qubits as compared to the starting state. To show that this representation is a lossless encoding we use it to do supervised learning using variational circuits on the entire dataset of 8x8 grid and reproduce a 95% accuracy on the training dataset (consisting of 108 samples). Unlike quantum autoencoders where the compact representations rely on being able to optimize against a fixed garbage state, here the relaxed restriction on the tensor product helps provide compact representations in cases where a fixed garbage state would not be feasible. Further investigations on what the entanglement of the subsystems reveal about the probability distribution from which the data is sampled can lead to other useful applications of this protocol. One might also be interested in carrying out machine learning by using weighted quantum circuits that run on the subsystems independently and compare its performance against the compact representations created thereby. One can also imagine using low entropic entangled states that stage 1 protocol outputs as input states for entanglement forging [8] and look for useful applications with the same. We would like to conclude by saying that, efficient methods for encoding and compression are likely to pave way towards the problem of efficient trainability on higher dimensional Hilbert spaces, and this work serves as a step towards that direction.

6 Acknowledgements

This material is based upon work supported by the U.S. Department of Energy, Office of Science, National Quantum Information Science Research Centers, Quantum Science Center. This manuscript has been authored by UT-Battelle, LLC, under contract DE-AC05-00OR22725 with the US Department of Energy (DOE). Sabre Kais would like to acknowledge the support from National Science Foundation under award number 1955907.

References

[1] Leonardo Banchi, Jason Pereira, and Stefano Pirandola. “Generalization in Quantum Machine Learning: A Quantum Information Standpoint”. In: *PRX Quantum* 2.4 (Nov. 2021). DOI: 10.1103/prxquantum.2.040321. URL: https://doi.org/10.1103%2Fprxquantum.2.040321.

[2] Marcello Benedetti et al. “A generative modeling approach for benchmarking and training shallow quantum circuits”. In: *npj Quantum Information* 5.1 (2019), pp. 1–9.

[3] Marco Vinicio Sebastian Cerezo de la Roca et al. “Cost function dependent barren plateaus in shallow parametrized quantum circuits”. In: *Nature Communications* 12.LA-UR-19-32681 (2021).

[4] Anirban N. Chowdhury, Guang Hao Low, and Nathan Wiebe. *A Variational Quantum Algorithm for Preparing Quantum Gibbs States*. 2020. DOI: 10.48550/ARXIV.2002.00055. URL: https://arxiv.org/abs/2002.00055.

[5] Lukasz Cincio et al. “Learning the quantum algorithm for state overlap”. In: *New Journal of Physics* 20.11 (2018), p. 113022.

[6] Vivek Dixit et al. “Training a quantum annealing based restricted boltzmann machine on cybersecurity data”. In: *IEEE Transactions on Emerging Topics in Computational Intelligence* 6.3 (2021), pp. 417–428.

[7] Yuxuan Du et al. “Learnability of quantum neural networks”. In: *PRX Quantum* 2.4 (2021), p. 040337.

[8] Andrew Eddins et al. “Doubling the size of quantum simulators by entanglement forging”. In: *PRX Quantum* 3.1 (2022), p. 010309.

[9] Edward Farhi, Jeffrey Goldstone, and Sam Gutmann. “A quantum approximate optimization algorithm”. In: *arXiv preprint arXiv:1411.4028* (2014).

[10] Zoë Holmes et al. “Connecting ansatz expressibility to gradient magnitudes and barren plateaus”. In: *PRX Quantum* 3.1 (2022), p. 010313.

[11] Sumit Suresh Kale, Yong P Chen, and Sabre Kais. “Constructive Quantum Interference in Photochemical Reactions”. In: *Journal of Chemical Theory and Computation* 17.12 (2021), pp. 7822–7826.

[12] Diederik P. Kingma and Max Welling. “An Introduction to Variational Autoencoders”. In: *Foundations and Trends® in Machine Learning* 12.4 (2019), pp. 307–392. DOI: 10.1561/2200000056. URL: https://doi.org/10.1561%2F2200000056.

[13] Jarrod R McClean et al. “Barren plateaus in quantum neural network training landscapes”. In: *Nature communications* 9.1 (2018), pp. 1–6.

[14] Alberto Peruzzo et al. “A variational eigenvalue solver on a photonic quantum processor”. In: *Nature communications* 5.1 (2014), pp. 1–7.

[15] John Preskill. “Quantum computing in the NISQ era and beyond”. In: *Quantum* 2 (2018), p. 79.

[16] Jonathan Romero, Jonathan P Olson, and Alan Aspuru-Guzik. “Quantum autoencoders for efficient compression of quantum data”. In: *Quantum Science and Technology* 2.4 (2017), p. 045001.

[17] Manas Sajjan, Shree Hari Sureshbabu, and Sabre Kais. “Quantum machine-learning for eigenstate filtration in two-dimensional materials”. In: *Journal of the American Chemical Society* 143.44 (2021), pp. 18426–18445.

[18] Manas Sajjan et al. “Quantum machine learning for chemistry and physics”. In: *Chem. Soc. Rev.* 51 (15 2022), pp. 6475–6573. DOI: 10.1039/D2CS00203E. URL: http://dx.doi.org/10.1039/D2CS00203E.

[19] Raja Selvarajan, Manas Sajjan, and Sabre Kais. “Variational quantum circuits to prepare low energy symmetry states”. In: *Symmetry* 14.3 (2022), p. 457.

[20] Raja Selvarajan et al. “Prime factorization using quantum variational imaginary time evolution”. In: *Scientific reports* 11.1 (2021), pp. 1–8.

[21] Sukin Sim, Peter D Johnson, and Alán Aspuru-Guzik. “Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms”. In: *Advanced Quantum Technologies* 2.12 (2019), p. 1900070.

[22] Kwok Ho Wan et al. “Quantum generalisation of feedforward neural networks”. In: *npj Quantum information* 3.1 (2017), pp. 1–8.

[23] S Wang et al. “Noise-Induced Barren Plateaus in Variational Quantum Algorithms. arXiv 2020”. In: *arXiv preprint arXiv:2007.14384* ().

[24] Youle Wang, Guangxi Li, and Xin Wang. “Variational Quantum Gibbs State Preparation with a Truncated Taylor Series”. In: *Physical Review Applied* 16.5 (Nov. 2021). DOI: 10.1103/physrevapplied.16.054035. URL: https://doi.org/10.1103%2Fphysrevapplied.16.054035.

[25] David Wierichs et al. “General parameter-shift rules for quantum gradients”. In: *Quantum* 6 (Mar. 2022), p. 677. DOI: 10.22331/q-2022-03-30-677. URL: https://doi.org/10.22331%2Fq-2022-03-30-677.

[26] Rongxin Xia and Sabre Kais. “Quantum machine learning for electronic structure calculations”. In: *Nature communications* 9.1 (2018), pp. 1–6.

A Appendix

We shall show here that as far as the variational circuit is considered, the presence of phase in the index qubit can be eliminated by the choice of ansatz and the relative probabilities of $|\phi\rangle$ and $|\eta\rangle$ can be ignored with sufficient samples as they average out to be equal.

Let $V(\vec{\alpha})$ be the ansatz used for classification post creating the compact state representation. The classification cost function with respect to this ansatz is given by,

$$\text{Classification Cost} = \sum_i \ (l_i - \langle\tilde{\psi_i}| V^\dagger(\vec{\alpha})[I^{\otimes n-1} \otimes Z]V(\vec{\alpha}) |\tilde{\psi_i}\rangle)^2 \tag{4}$$

where, $|\tilde{\psi_i}\rangle = c_0 |0\rangle |\phi\rangle + c_1 e^{i\gamma} |1\rangle |\eta\rangle$, where $c > 0$ references the relative amplitude (s.t $c_0^2 + c_1^2 = 1$) and γ the relative phase acquired in measuring subsystem 2 after the sequence of controlled swaps performed. We start with creating an ansatz that decouples the effect of having a phase on the state. To this end, we modify the unitary ansatz V to have the following form, $V(\vec{\alpha_1}, \vec{\alpha_2}) = |0\rangle\langle 0| \otimes V_0(\vec{\alpha_0}) + |1\rangle\langle 1| \otimes V_1(\vec{\alpha_1})$, i.e, $V_1(\vec{\theta_1})$ acts only on $|\phi\rangle$ and $V_2(\vec{\theta_2})$ acts only on $|\eta\rangle$. Simplifying 4 we get,

$$\text{Classification Cost} = \sum_i \ (l_i - (c_0^2 \langle\phi| V_0^\dagger(\vec{\alpha_0})[I^{\otimes n-2} \otimes Z]V_0(\vec{\alpha_0}) |\phi\rangle + c_1^2 \langle\eta| V_1^\dagger(\vec{\alpha_1})[I^{\otimes n-2} \otimes Z]V_1(\vec{\alpha_1}) |\eta\rangle))^2 \tag{5}$$

Notice that the effect of having any phase is lost in the process as the ansatz and measurement is impervious to the presence of phase in the state. Now we will show how averaging over multiple realizations, its possible to get rid of the relative probabilities. The gradients computed from the above cost function computed for input sample i is given by,

$$\nabla\text{Cost}_i = \langle c_{i,0}^2\rangle \nabla \langle\phi| V_0^\dagger(\vec{\alpha_0})[I^{\otimes n-2} \otimes Z]V_0(\vec{\alpha_0}) |\phi\rangle + \langle c_{i,1}^2\rangle \nabla \langle\eta| V_1^\dagger(\vec{\alpha_1})[I^{\otimes n-2} \otimes Z]V_1(\vec{\alpha_1}) |\eta\rangle \tag{6}$$

where, $\langle . \rangle$ refers to averaging over multiple shots of the same input sample after the measurement of the second subsystem has been made. An estimate made of $\langle c_{i,0}^2\rangle$ and $\langle c_{i,1}^2\rangle$ can be estimated by measuring the index register following the measurement of subsystem 2 in the computational basis. We can thus prepare the index register in the state $\frac{\langle c_{i,1}^2\rangle}{(\langle c_{i,0}^2\rangle + \langle c_{i,1}^2\rangle)} |0\rangle + \frac{\langle c_{i,0}^2\rangle}{(\langle c_{i,0}^2\rangle + \langle c_{i,1}^2\rangle)} |1\rangle$ for measuring the gradient contribution from input sample i. This allows us to optimize for the gradients independent of the computational basis onto which system 2 gets projected, giving us,

$$\nabla\text{Cost}_i = \sum_i \nabla \langle\phi| V_0^\dagger(\vec{\alpha_0})[I^{\otimes n-2} \otimes Z]V_0(\vec{\alpha_0}) |\phi\rangle + \nabla \langle\eta| V_1^\dagger(\vec{\alpha_1})[I^{\otimes n-2} \otimes Z]V_1(\vec{\alpha_1}) |\eta\rangle \tag{7}$$

The gradients computed to this cost function is deterministic removing any probabilistic effects from the presence of a relative phase or amplitude within the cost function. This helps us circumvent the exponential scaling in the runtime that might arise from the need to project onto a specific computational basis state.