

# Quantum State Discrimination Using Noisy Quantum Neural Networks

Andrew Patterson<sup>\*1</sup>, Hongxiang Chen<sup>2,3</sup>, Leonard Wossnig<sup>2,3</sup>, Simone Severini<sup>3</sup>, Dan Browne<sup>1</sup>, and Ivan Rungger<sup>†1,4</sup>

<sup>1</sup>*Department of Physics and Astronomy, University College London, Gower Street, London, WC1E 6BT, United Kingdom*

<sup>2</sup>*Rahko Ltd. Finsbury Park, N4 3JP, United Kingdom*

<sup>3</sup>*Dept. of Computer Science, University College London, Gower Street, London, WC1E 6BT, United Kingdom*

<sup>4</sup>*National Physical Laboratory, Teddington, TW11 0LW, United Kingdom*

Near-term quantum computers are noisy, and therefore must run algorithms with a low circuit depth and qubit count. Here we investigate how noise affects a Quantum Neural Network (QNN) for state discrimination, which is applicable on near-term quantum devices as it fulfils the above criteria. We find that for the required gradient calculation on a noisy device a quantum circuit with a large number of parameters is disadvantageous. By introducing a new smaller circuit ansatz we overcome this limitation, and find that the QNN performs well at noise levels of current quantum hardware. We also show that networks trained at higher noise levels can still converge to useful parameters. Our findings demonstrate that noisy quantum computers can be used for state discrimination and other applications, such as classifiers of the output of quantum generative adversarial networks.

## 1 Introduction

Quantum state discrimination is important in many emerging quantum technologies: quantum cryptography [1], entanglement concentration [2], quantum cloning [3], and quantum metrology and sensing [4, 5]. Quantum circuits trained for classification could also be used in quantum machine learning problems as a classifier of quantum data. They could classify the output of other quantum circuits, e.g. the output of a quantum generative adversarial network (GAN) [6]. Current quantum computing devices are subject to non-negligible amounts of noise [7, 8, 9], and therefore algorithm design for devices in the near future must take this into account. Here we present an extension to noisy devices of the approach for quantum state discrimination outlined in Ref. [10], a quantum analogue of a neural network used for state discrimination. In Ref. [10] simulations of shallow quantum circuits were trained to find the optimal Positive Operator Valued Measure (POVM), or measurement, to distinguish between two families of non-orthogonal quantum states. Given an input state chosen randomly from one of the families, the output of the network should indi-

cate which family the input was chosen from. To do this the network is trained on a set of labelled data, performing supervised learning [11]. The ideal POVM was learned via a classical optimiser using a gradient descent algorithm on the quantum parameters, which correspond to the rotation gates in the quantum circuit. There are similarities between a classical unitary neural network [12] and this algorithm. It contains a layer of symmetric fully connected neurons, followed by arbitrary numbers of non-linear layers, or dropout layers. The non-linearity in the quantum network is introduced by measurement of some of the qubits.

In this work we extend the simulations done previously from pure vector states to simulations of states represented as density matrices, so that we can model noise in the quantum device. We also simulate calculation of the parameter gradients on the quantum device, which would also be subject to noise in a real machine. We find that with these extensions including the effect of noise the previous algorithm proposed for noiseless systems no longer performs optimally. To recover performance we reduce the number of trainable parameters through consideration of the circuit structure.

This paper is structured as follows: we begin by outlining the theory of state discrimination and the QNN.

Corresponding authors:

<sup>\*</sup>andrew.patterson.10@ucl.ac.uk,

<sup>†</sup>ivan.rungger@npl.co.uk

We then discuss the simulation methods, gradient calculation and measurement. In Section 3 we present the results, including the effect of reducing the number of parameters and the effects of noise on training the circuits.

## 2 Methods

### 2.1 Quantum State Discrimination

We wish to discriminate a two-qubit input state,  $|\psi_{in}\rangle$ , which in general can be represented as a normalised vector with 4 complex components. The state is chosen randomly from two families of states, labelled  $a$  and  $b$ :

$$|\psi_{in,a}\rangle = (\sqrt{1-a^2}, 0, a, 0), \quad (1)$$

$$|\psi_{in,b}\rangle = (0, \pm \frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}, 0), \quad (2)$$

Note however that our results are expected to be applicable also to other families of states, since the circuit learning algorithm can be applied to general states.

To simulate the effect of noise we use density matrices to represent quantum states:

$$\rho_{a(b)} = |\psi_{a(b)}\rangle\langle\psi_{a(b)}|. \quad (3)$$

The value of  $a$  is drawn from a probability distribution  $a \in (0, 1]$  which is characterised by its mean,  $\mu_a$  and standard deviation,  $\sigma_a$ . Note that the  $b$  states can take two values, depending on the sign in the second vector element; these are selected randomly, with equal chance. Primarily these states are chosen to reproduce the work in [10] and in [13], where the state discrimination was performed in a laboratory. Secondly, these states are non-orthogonal and can therefore not be discriminated from each other perfectly, making the problem harder for the algorithm.

### 2.2 The Quantum Neural Network

In a classical neural network some nodes are discarded during training, which is called dropout and stops the network from over-fitting [14]. We can also think of dropout as the introduction of non-linearity into the network. A network with dropout cannot be represented by any smaller, linear network, whereas a many-layered linear network can always be reduced to a single linear layer. Quantum evolution is unitary and linear, so if we wish to introduce non-linearity into a quantum neural network we need to include a measurement. Figure 1 shows the structure of the QNN, where the choice of the second step of the circuit,  $V_{1,2}$  is conditioned on the outcome of a measurement on

the first qubit. The measurement results are then used as the output of the neural network [10].

There are two non-orthogonal states to discriminate, so if we wish to have a network that can be trained to not commit any errors, we must allow for it to produce an inconclusive result [13]. This allows the network to give a ‘don’t know’ result as opposed to an erroneous one. Therefore we have a minimum of three outputs, necessitating two measurement qubits.

The output of the network is determined by the measurement outcome. As we begin in a random configuration and are training the system, we can arbitrarily select which label a measurement outcome corresponds to:

$$\{|00\rangle : a, |01\rangle : b, |10\rangle : a, |11\rangle : \text{inconclusive}\}. \quad (4)$$

The choice of unbalanced labels may have an effect upon the outcome (e.g. more errors in  $b$  states), which could be mitigated with a choice of two inconclusive outcomes. In general one might adapt the assignment of measurement outcomes to labels according to the considered specific task.

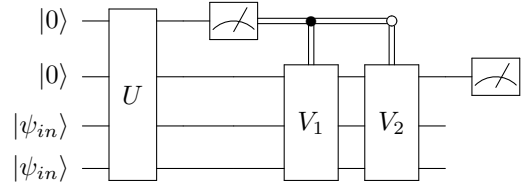
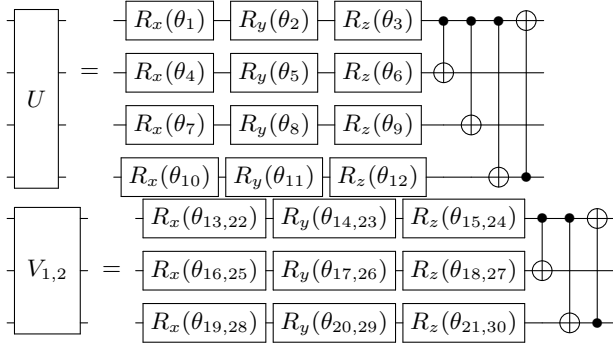


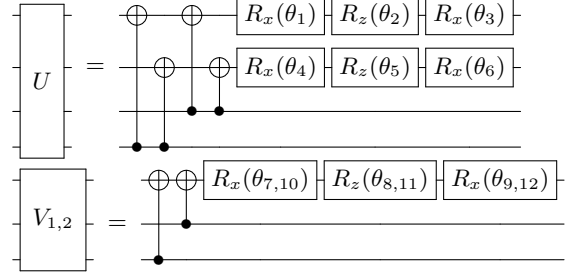
Figure 1: The general form of the quantum circuits used in this work. The input state is on the bottom two qubits, and measuring the first qubit introduces a non-linear dropout layer. The sub-circuits  $U$ ,  $V_{1,2}$  are shown in Figure 2.

The structure of the  $U$  and  $V_{1,2}$  circuit blocks is given in Figure 2, where 2a shows the same circuits used in [10] and 2b shows the reduced circuits introduced here, which we will discuss in more detail below. These circuits are small and have low-depth, so that they can be ran on a quantum computer which supports measurement as the circuit is running and classical feedback. This requires fast measurement and fast classical processing which is not possible in many current systems, but has been achieved in an ion-trap device [15], meaning this algorithm could run on a current device.

The state discrimination task is then as follows: given a set of randomly selected and labelled input states, the classical optimiser must optimise the rotation angles  $\theta_{1..n}$  of the quantum circuit to maximise the likelihood of a correct determination of the state. A correct determination is found when the measurement output of the quantum circuit is equal to the corresponding input state label as defined in Equation 4.



(a) The  $U$  and  $V$  circuit blocks originally used in [10].



(b) The form of the  $U$  and  $V$  blocks with a reduced number of parameters.

Figure 2: The circuits showing the trainable parameters, which are used in this work. Comparison of results obtained for the circuits 2b and 2a is made in Section 3.1.

## 2.3 Optimisation

Since the input states are initially labelled, the task for the classical optimiser is a supervised learning task [11]. The optimiser used in this experiment is Adam [16], which has been found to work well in a number of quantum variational algorithms [10, 17, 18, 19, 20]. It has also been shown classically that Adam deals well with noisy gradients [21], which will be the output of our noisy quantum computer. This is possible since Adam uses the concept of momentum, where the gradients of past steps contribute to the current step. Other optimisers such as RotoSolve [22] have been proposed, and a comparison of performance can be made in future work.

Noisy gradients are a feature of the work here: as gradient calculation must be performed on the noisy quantum device, we expect that the output gradients will be noisy. We also expect that there will be non-optimal local minima in our loss landscape, as this is also a feature of the loss function in the noiseless case [10]. Finally we also expect that the loss landscape may feature ‘barren plateaus’, as these have been shown to be a feature of quantum optimisation problems [23]. This further motivates the choice of a gradient-based optimiser such as Adam.

We define the function to minimise, the cost function, as

$$C = \alpha_{\text{err}} P_{\text{err}} + \alpha_{\text{inc}} P_{\text{inc}}, \quad (5)$$

where the positive real numbers  $\alpha_{\text{err}}, \alpha_{\text{inc}}$  are the cost parameters used to bias the network towards minimising errors or inconclusive results ( $P_{\text{err}}, P_{\text{inc}}$  are defined below). If for example we require the network to produce fewer errors, we can do this at the cost of recording more inconclusive results by increasing the value of  $\alpha_{\text{err}}$  relative to the value of  $\alpha_{\text{inc}}$ . We discuss the effect of changing the cost parameters in Section 3.1.

The measurement probabilities of a state,  $\rho$ , for a

generalised measurement,  $M = |\phi\rangle\langle\phi|$ , are given by:

$$\langle\rho\rangle = \text{Tr}(|\phi\rangle\langle\phi|\rho), \quad (6)$$

and the quantum state after measurement is given by

$$\rho_{\text{measured}} = \frac{|\phi\rangle\langle\phi|\rho|\phi\rangle\langle\phi|}{\text{Tr}(|\phi\rangle\langle\phi|\rho)}. \quad (7)$$

Using this we can find the probability of an erroneous or inconclusive measurement

$$P_{\text{err}} = \sum_{\rho_i \in b} (\langle\rho_i\rangle_{00} + \langle\rho_i\rangle_{10}) + \sum_{\rho_i \in a} \langle\rho_i\rangle_{01}, \quad (8)$$

$$P_{\text{inc}} = \sum_{\rho_i \in a, b} \langle\rho_i\rangle_{11}, \quad (9)$$

where  $\rho_i$  is the input state, and  $\langle\rho_i\rangle_{jk}$  refers to the probability of obtaining a measurement of  $|jk\rangle$  from the circuit.

Discrimination of these states, without the use of a variational algorithm, has been shown in the laboratory to reach the theoretical best success probability,  $P_{\text{suc}}$  of 0.833 for  $\mu_a = 0.25, \sigma_a = 0.01$  [13]. This is a minimum loss,  $L = 1 - P_{\text{suc}} = P_{\text{err}} + P_{\text{inc}}$ , of 0.166. For the equal probability case,  $P(|00\rangle) = P(|01\rangle) = P(|10\rangle) = P(|11\rangle) = 0.25$ , the success rate is 0.385, this translates into a loss of 0.635. This gives us lower and upper expected bounds to compare our results for the loss to.

## 2.4 Gradient Calculation

Unlike gradient-free optimisers (such as Nelder-Mead [24]) the Adam optimiser requires the calculation of parameter derivatives ( $\frac{\partial\langle C \rangle}{\partial\theta_{0..n}}$ ). In the previous work this was done using the forward differences formula [10], which requires direct access to the components of the wavefunction. In a real quantum computer this is difficult to achieve, and hence here we use a more

practical approach. Calculation of the gradients of quantum parameters has received attention recently [25, 26, 27] due to the introduction of variational methods such as the variational quantum eigensolver (VQE) [28]. The gradient of the loss function with respect to a parameter  $\theta_i$  is calculated by the method outlined in [25], which requires two extra repetitions of the circuit for each  $\theta_i$ :

$$\frac{\partial \langle C \rangle}{\partial \theta_i} = \langle C \rangle^+ - \langle C \rangle^-, \quad (10)$$

where  $\langle C \rangle^\pm$  is calculated by changing  $\theta_i$  by  $\pm \frac{\pi}{4}$ , and leaving all other parameters in the circuit constant.

## 2.5 Reduced circuit

For the probability distribution of  $a$  determined by  $\mu_a = 0.25$  and  $\sigma_a = 0.01$  the maximum theoretical success rate ( $P_{\text{suc}}$ ) is 0.8333 [13], which was obtained with the long circuit in Ref. [10]. However, after optimisation of circuit parameters for our larger circuit in Figure 2a we reach only 0.72, which is significantly smaller than the theoretical limit. We attribute this discrepancy to the different implementations of the optimisation procedure, and to the different calculation of the gradients. To overcome this sub-optimal result we designed the shorter circuits in Fig. 2b. The choice of the reduced circuit is motivated by the consideration that for this task the rotations on the state qubits have a smaller effect on the measurement outcomes than rotations on the measurement qubits. This choice of structure is so that the input states are entangled with both output qubits, and then the measurement qubits are rotated. The choice of rotations about the x-axis, followed by the z-axis, and then again the x-axis allows for the initial state to be transformed to any other state on the surface of the Bloch sphere [29]. With this short circuit (Figure 2b) we obtain a success rate of 0.826, close to optimal performance. This is the circuit used for the results presented, except where we explicitly note that the longer circuit is used.

We note that as the shorter circuits do not explore the full Hilbert space of all the qubits, they may not be necessarily optimal for all discrimination tasks. Investigations into the capability of different variational quantum circuits have been made in [30]. Here we present evidence that when used on a noisy device, the smaller variational circuit converges to better results than the larger circuit. In general a trade-off needs to be made between this better resilience to noise and the ability of the circuit to distinguish very complex states.

## 2.6 Noise

Noise in quantum computers can be modelled by a superoperator,  $\mathcal{E}(\rho)$ , which is a completely positive, trace-preserving map on the state  $\rho$  [29]. We can give the operator-sum representation of  $\mathcal{E}$  by introducing the Kraus operators,  $E_k$ :

$$\mathcal{E}(\rho) = \sum_k E_k \rho E_k^\dagger, \quad (11)$$

and to preserve the trace of  $\rho$ , they must obey the relation

$$\sum_k E_k^\dagger E_k = \mathbb{1}. \quad (12)$$

For the single-qubit noise channel our operators are the single qubit Pauli operators, modified by the noise probability,  $p$ , to give the depolarising channel:

$$\begin{aligned} E_0 &= \sqrt{1 - \frac{3p}{4}} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \quad E_1 = \sqrt{\frac{p}{4}} \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \\ E_2 &= \sqrt{\frac{p}{4}} \begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}, \quad E_3 = \sqrt{\frac{p}{4}} \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \end{aligned} \quad (13)$$

For the two qubit noise channel, which is applied after a two-qubit gate, the Kraus operators are tensor products of the combinations of these operators, i.e.  $E_0 \otimes E_0, \dots, E_1 \otimes E_2, \dots, E_3 \otimes E_3$ . The probability of the single qubit noise channel is  $p_{1q} = \frac{4}{5}p_{2q}$ . This is the one-qubit marginal probability of error for the two-qubit gates [31], i.e. the probability of a single qubit error without condition of an error on the other qubit. This is a commonly used assumption in the quantum error correction literature [32], which assumes that the error process in single and two qubit gates is the same. In real devices the process can be quite different, but we nevertheless choose this method as it is an upper limit on the error probability of the single qubit gate. When quoting the noise level in this paper, we will always refer to  $p_{2q}$ . We set the highest noise level in our simulations to  $p_{2q} = 0.1$ , as this is an upper limit on two-qubit gate fidelities reported on current quantum hardware [7, 8, 9].

Note that here we have not considered asymmetric noise or different quality qubits. However, we believe that correcting for a systematic bias such as this is possible for a variational algorithm, as seen in [33]. Furthermore, in actual devices the single qubit noise probability reported is much lower than 4/5 of the two qubit gate noise level. For example, the single qubit gate error rate reported in [9] is  $1.4 \times 10^{-3}$ , whereas the two qubit gate fidelity is  $9.3 \times 10^{-3}$ , and the ratio between these is approximately 3/20, at least a factor of 5 lower. In our simulations the single qubit noise is set to the higher limit of 4/5, so that we are more demanding of the algorithm.

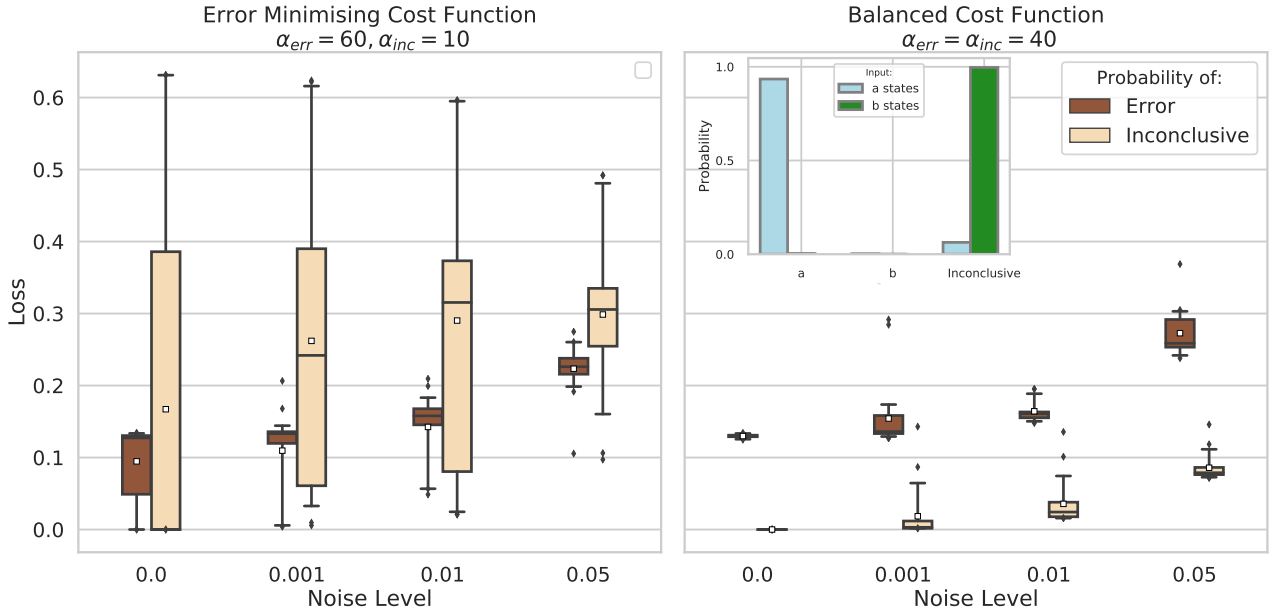


Figure 3: The distribution of  $P_{inc}$  and  $P_{err}$  from 25 repeats of a network biased towards reducing errors, and one with a balanced cost function, for  $\mu_a = 0.5$ , and  $\sigma_a = 0.15$ . An example undesirable output for a single minimising error run is in the inset, where no  $b$  states are measured correctly, but the network still converges (the x-axis shows the output label and the colour is the input state). The interquartile range is contained within the box, and the 5th and 95th percentiles are marked by the whiskers. Outliers of this range are marked by a diamond. The mean is marked with a white square, and the median is the line across the box.

Simulations of the quantum device were performed on a simulator built using the Tensorflow machine learning package [34], and verified with the Cirq [35] quantum simulation package. In our simulations we set the initial angles, which are our parameters to be optimised, at random values. The labelled quantum state is an input to the circuit in Figure 1, that circuit is ran and the measurement probabilities calculated and with them the cost. The gradient of the cost with respect to each parameter is then calculated by the method described in section 2.4, and the parameters are updated according to the Adam optimiser to minimise the cost. This routine is repeated until the cost no longer significantly decreases.

### 3 Results

#### 3.1 Effect of cost function choice and circuit depth

In Figure 3 we compare the obtained optimised  $P_{err}$  and  $P_{inc}$  for an error minimising cost function ( $\alpha_{err} = 60, \alpha_{inc} = 10$ ) and a balanced cost function ( $\alpha_{err} = 40, \alpha_{inc} = 40$ ). The error minimising cost function often results in a practically unusable network, because while it gives a low probability of error, the probability of inconclusive results is too high, as seen for an extreme case in the inset of Figure 3. Note that in this particular case all  $b$  states are detected as incon-

clusive, and one could in principle switch the *inconclusive* and  $b$  labels to obtain a good discrimination. However, for the more general case this will not be possible.

In comparison to the error minimising setting, the results for the balanced cost function are stable and generally give both small  $P_{err}$  and  $P_{inc}$ , with some  $P_{err}$  comparable to the error minimising setting. For the remaining analysis we therefore use the balanced cost function ( $\alpha_{err} = \alpha_{incon} = 40$ ). We note that as the noise level is increased,  $P_{inc}$  and  $P_{err}$  progressively tend to larger values. The effect of noise will be analysed in detail in the next section.

We next investigate the influence of the number of parameters in the quantum circuit on the loss. In Figure 4 we compare the distributions of loss between the circuit with more trainable parameters in Figure 2a to the circuit with fewer parameters in Figure 2b. It can be seen that the reduced circuits consistently perform better than the long circuits. It is more difficult to train circuits with a large number of parameters both without and with noise, as seen in Figure 5. We see that the higher noise cases always converge to a higher loss, and that the reduced circuits perform better in both cases.

Even with very low noise, the output is worse for larger circuits. This suggests that with more parameters the algorithm struggles to optimise, when the gradient calculations are performed on the quantum



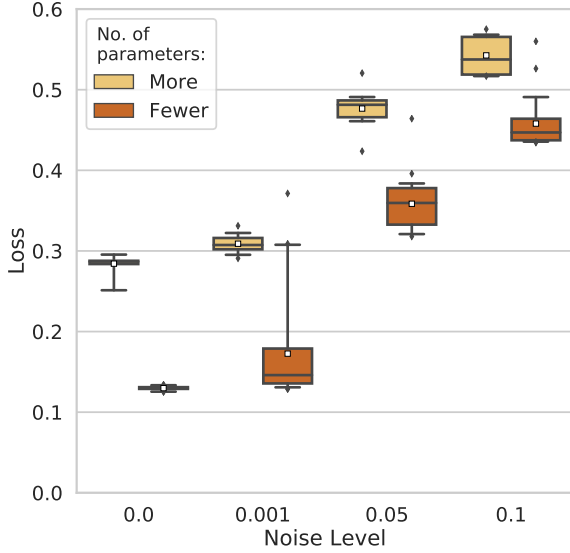


Figure 4: The distributions of loss ( $P_{\text{err}} + P_{\text{inc}}$ ) at different noise levels for the two circuits shown in Figure 2. Both have other parameters fixed,  $\mu_a = 0.5$ ,  $\sigma_a = 0.15$ ,  $\alpha_{\text{err}} = \alpha_{\text{inc}} = 40$ . We observe that reducing the number of parameters is advantageous at all noise levels.

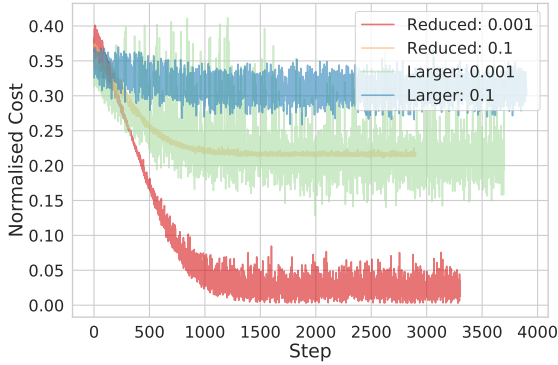


Figure 5: Evolution of the normalised cost functions for larger and reduced circuits for  $\mu_a = 0.5$  and  $\sigma_a = 0.15$ , with noise levels of 0.001 and 0.1. Shown here is the number of steps taken to converge. Note that the time taken to complete a single step of the longer circuit is much greater than for the reduced circuit.

device. Good performance of the short circuit also in presence of noise can be due to the noisy gradient regularising the training, thereby optimising performance [11]. Moreover, the Adam optimiser has been designed to work well with noisy gradients [16].

### 3.2 Effect of noise

In Figure 6 the noise-less case is compared to resulting optimised loss for increasing noise levels (note that in this section we always use the reduced circuit). It can be seen that using this algorithm with zero noise produces the lowest loss, as one expects intuitively.

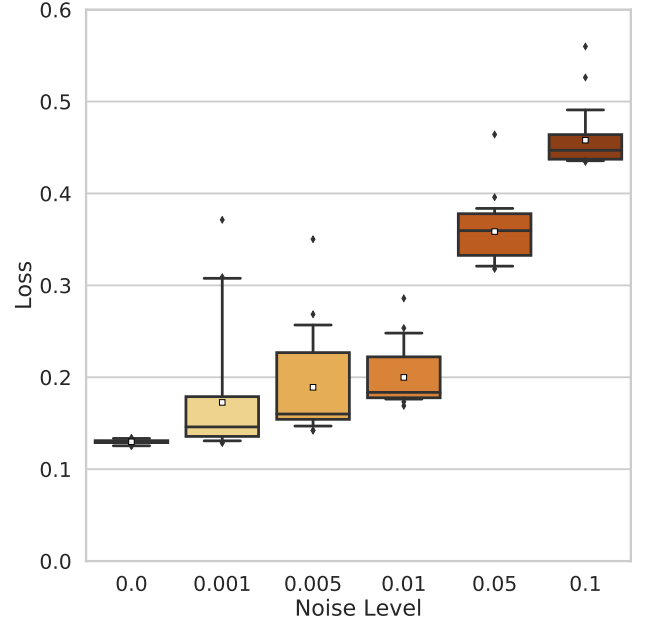


Figure 6: The distribution of loss for 25 repeats of training the network. The cost function is balanced,  $\alpha_{\text{err}} = \alpha_{\text{inc}} = 40$ ,  $\mu_a = 0.5$ , and  $\sigma_a = 0.15$ . At levels of noise present in current devices, 0.01, the loss value is favourable, an average of 0.2.

With increasing noise the average loss increases continuously. In presence of noise there are a few high-loss outliers, which we attribute to the optimiser becoming stuck in local minima of the cost function. As the noise is increased, performance deteriorates, but is no worse than the random output limit, 0.635. Importantly, at noise levels comparable to current devices,  $p_{2q} = 0.01$ , the algorithm is still performing well, at an average loss of 0.2.

The overlap of the two states to be discriminated between is equal to  $a/\sqrt{2}$ , and increases with  $\mu_a$ . States which overlap more should be harder to discriminate, which we indeed see in Figure 7. Increasing noise in the system reduces the difference between higher and lower values of  $\mu_a$ , since the loss approaches the result for a fully random output. At high noise levels and high  $\mu_a$ , some runs are performing even worse than the random output limit (0.635), but on average the loss remains well below that value. In general we conclude that the tolerable levels of noise depend on the overlap between the states, where small overlap allows the states to be discriminated even for higher noise in the quantum computer.

In general a high level of noise always leads to a higher loss. However, we find that when noise is applied only during the training of the parameters, the optimised parameters are rather resilient to this training noise. To show this in Figure 8 we present the results when training the device at one noise level, and validating at another. We see that even with high levels of training

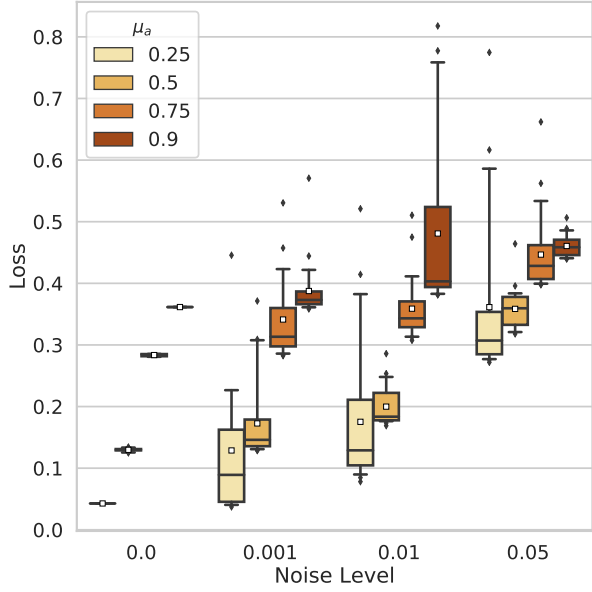


Figure 7: The distribution of loss ( $P_{\text{err}} + P_{\text{inc}}$ ) and the effect of different values of  $\mu_a$ . The cost function is balanced,  $\alpha_{\text{err}} = \alpha_{\text{inc}} = 40$ , and  $\sigma_a = 0.15$ . We see that for lower values of  $\mu_a$ , corresponding to smaller overlap between the states to be discriminated, the discrimination task is performed better.

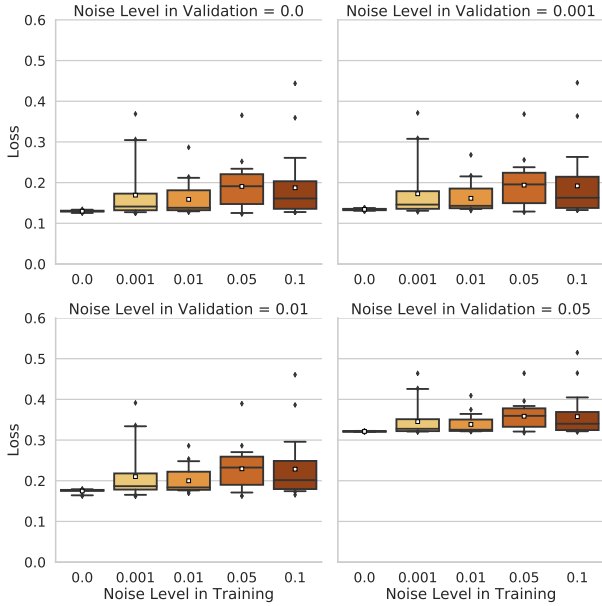


Figure 8: Distribution of loss ( $P_{\text{err}} + P_{\text{inc}}$ ) against training noise for a selection of noise levels in the validation circuit.

noise the optimiser converges onto good parameters, as we find comparably low loss levels when validating those parameters trained at a high noise level with low noise in the validation step. Also here we find that when validating at noise levels seen in current devices,  $p_{2q} = 0.01$ , the average loss does not increase above 0.25, which would be acceptable to use for state

discrimination.

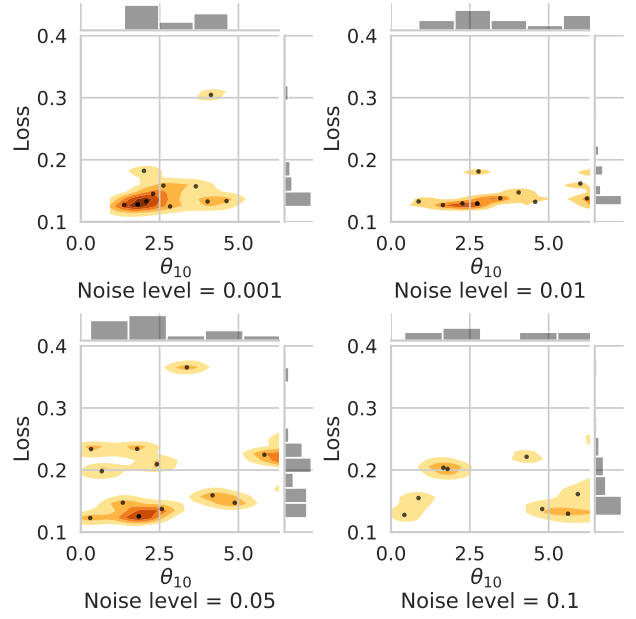


Figure 9: The distribution of loss and  $\theta_{10}$  obtained at different noise levels. Here we can see the effect of noise on the values of  $\theta_{10}$  that the optimiser converges to. We only show a single representative parameter  $\theta_{10}$ , since we have a approximately similar behaviour for all other parameters.

Finally, we investigate the effect of the noise during training on the actual values of the optimised parameters in the circuit. In Figure 9 we present the distribution of  $\theta_{10}$  for different values of noise. The values of  $\theta$  are all taken modulo  $2\pi$ , and at zero validation noise to remove the effect of validation errors on the loss. We see that the range of angles converged upon increases as the noise in the circuit increases. Some values become stuck at high loss, and there can be different values for the minimal loss parameters. The increase in noise seems to change not just the final loss, but the parameters found that minimise loss. We cannot rule out the correlation between different parameters as the noise level changes. Combined with what we see in Figure 8, that good parameters are still found at higher noise levels, we may conclude that noise in the circuit can push the optimiser out of local minima, so that it can find some other local minima at lower loss.

From the results presented here we see that this algorithm performs well in the presence of noise in the training and validation steps (Figure 6), and that parameters found on a noisy device work well when validated on a device with low noise (Figure 8). When calculating parameter gradients on a noisy quantum device, reducing the number of parameters has a positive effect, as shown in Figure 4 and Figure 5.

## 4 Conclusion

We have shown that a QNN can be trained for the task of state discrimination on a noisy device, with noise levels found in current NISQ devices. We have also shown that gradient descent algorithms are viable on noisy quantum devices, given a good choice of classical algorithm. As discussed in [30], choice of training circuits in variational quantum algorithms has a large effect upon success. Here we reduced the number of parameters by removing rotation gates from the input states, and indeed show that the low circuit depth and qubit count of our algorithm is beneficial in the presence of noise. We were able to reduce the circuit size by using our knowledge of the variational problem. The smaller circuit size has the extra advantage of reducing the training time. While we specifically considered the task of quantum state discrimination, the algorithm presented here can be equally applied to such problems as verification of general quantum machine learning outputs, and applications in sensing, imaging and metrology.

## Acknowledgements

AP is supported by the InQUBATE Training and Skills Hub [grant EPSRC EP/P510270/1]. IR acknowledges financial support from the UK Department of Business, Energy and Industrial Strategy (BEIS). HC acknowledges the support through a Teaching Fellowship from UCL. LW acknowledges kindly the support through the Google PhD Fellowship in Quantum Computing. DB acknowledges support from the EPSRC Prosperity Partnership in Quantum Software for Modelling and Simulation [grant No. EP/S005021/1]. AP acknowledges the use of the UCL Myriad High Throughput Computing Facility (Myriad@UCL), and associated support services, in the completion of this work.

## References

- [1] C. H. Bennett. “Quantum cryptography using any two nonorthogonal states”. In: *Phys. Rev. Lett.* 68.21 (1992), pp. 3121–3124. ISSN: 0031-9007. DOI: [10.1103/PhysRevLett.68.3121](https://doi.org/10.1103/PhysRevLett.68.3121).
- [2] A. Chefles. “Quantum state discrimination”. In: *Contemp. Phys.* 41.6 (2000), pp. 401–424. ISSN: 0010-7514. DOI: [10.1080/00107510010002599](https://doi.org/10.1080/00107510010002599).
- [3] L.-M. Duan and G.-C. Guo. “Probabilistic Cloning and Identification of Linearly Independent Quantum States”. In: *Phys. Rev. Lett.* 80.22 (1998), pp. 4999–5002. ISSN: 0031-9007. DOI: [10.1103/PhysRevLett.80.4999](https://doi.org/10.1103/PhysRevLett.80.4999).
- [4] V. Giovannetti, S. Lloyd, and L. Maccone. “Advances in quantum metrology”. In: *Nat. Photonics* 5.4 (2011), pp. 222–229. ISSN: 1749-4885. DOI: [10.1038/nphoton.2011.35](https://doi.org/10.1038/nphoton.2011.35).
- [5] S. Lloyd. “Enhanced Sensitivity of Photodetection via Quantum Illumination”. In: *Science* 321.5895 (2008), pp. 1463–1465. ISSN: 0036-8075. DOI: [10.1126/science.1160627](https://doi.org/10.1126/science.1160627).
- [6] M. Benedetti et al. “Adversarial quantum circuit learning for pure state approximation”. In: *New J. Phys.* 21.4 (2019), p. 043023. ISSN: 1367-2630. DOI: [10.1088/1367-2630/ab14b5](https://doi.org/10.1088/1367-2630/ab14b5).
- [7] IBM. *Quantum systems - IBM Q*. 2019.
- [8] S. S. Hong et al. “Demonstration of a Parametrically-Activated Entangling Gate Protected from Flux Noise”. In: *Preprint* (2019). arXiv: [1901.08035](https://arxiv.org/abs/1901.08035).
- [9] Y. Chen et al. “Qubit Architecture with High Coherence and Fast Tunable Coupling”. In: *Phys. Rev. Lett.* 113.22 (2014), p. 220502. ISSN: 0031-9007. DOI: [10.1103/PhysRevLett.113.220502](https://doi.org/10.1103/PhysRevLett.113.220502).
- [10] H. Chen et al. “Universal discriminative quantum neural networks”. In: *Preprint* (2018). arXiv: [1805.08654](https://arxiv.org/abs/1805.08654).
- [11] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT Press, 2016.
- [12] M. Arjovsky, A. Shah, and Y. Bengio. “Unitary Evolution Recurrent Neural Networks”. In: *Preprint* (2015). arXiv: [1511.06464](https://arxiv.org/abs/1511.06464).
- [13] M. Mohseni, A. M. Steinberg, and J. A. Bergou. “Optical Realization of Optimal Unambiguous Discrimination for Pure and Mixed Quantum States”. In: *Phys. Rev. Lett.* 93.20 (2004), p. 200403. ISSN: 0031-9007. DOI: [10.1103/PhysRevLett.93.200403](https://doi.org/10.1103/PhysRevLett.93.200403).
- [14] N. Srivastava et al. “Dropout: A simple way to prevent neural networks from overfitting”. In: *Journal of Machine Learning Research* 15 (2014), pp. 1929–1958. ISSN: 15337928.
- [15] V. Negnevitsky et al. “Repeated multi-qubit readout and feedback with a mixed-species trapped-ion register”. eng. In: *Nature* 563.7732 (2018), p. 527. DOI: [10.1038/s41586-018-0668-z](https://doi.org/10.1038/s41586-018-0668-z).
- [16] D. P. Kingma and J. Ba. “Adam: A Method for Stochastic Optimization”. In: *Preprint* (2014). arXiv: [1412.6980](https://arxiv.org/abs/1412.6980).
- [17] P. Gokhale et al. “Partial Compilation of Variational Algorithms for Noisy Intermediate-Scale Quantum Machines”. In: *Proceedings of the 52nd Annual IEEE/ACM International Symposium on Microarchitecture - MICRO '52*. New York, New York, USA: ACM Press, 2019, pp. 266–278. ISBN: 9781450369381. DOI: [10.1145/3352460.3358313](https://doi.org/10.1145/3352460.3358313). arXiv: [1909.07522](https://arxiv.org/abs/1909.07522).



- [18] R. Sweke et al. “Stochastic gradient descent for hybrid quantum-classical optimization”. In: *Preprint* (2019). arXiv: [1910.01155](#).
- [19] K. E. Hamilton, E. F. Dumitrescu, and R. C. Pooser. “Generative model benchmarks for superconducting qubits”. In: *Phys. Rev. A* 99.6 (2019), p. 062323. ISSN: 2469-9926. DOI: [10.1103/PhysRevA.99.062323](#). arXiv: [1811.09905](#).
- [20] J.-G. Liu et al. “Variational quantum eigensolver with fewer qubits”. In: *Phys. Rev. Research* 1.2 (2019), p. 023025. ISSN: 2643-1564. DOI: [10.1103/PhysRevResearch.1.023025](#). arXiv: [1902.02663](#).
- [21] A. Neelakantan et al. “Adding Gradient Noise Improves Learning for Very Deep Networks”. In: *Preprint* (2015). arXiv: [1511.06807](#).
- [22] M. Ostaszewski, E. Grant, and M. Benedetti. “Quantum circuit structure learning”. In: *Preprint* (2019). arXiv: [1905.09692](#).
- [23] J. R. McClean et al. “Barren plateaus in quantum neural network training landscapes”. In: *Nat. Commun.* 9.1 (2018), p. 4812. ISSN: 2041-1723. DOI: [10.1038/s41467-018-07090-4](#).
- [24] J. A. Nelder and R. Mead. “A Simplex Method for Function Minimization”. In: *The Computer Journal* 7.4 (1965), pp. 308–313. ISSN: 0010-4620. DOI: [10.1093/comjnl/7.4.308](#).
- [25] K. Mitarai et al. “Quantum circuit learning”. In: *Phys. Rev. A* 98.3 (2018), p. 032309. ISSN: 2469-9926. DOI: [10.1103/PhysRevA.98.032309](#). arXiv: [1803.00745](#).
- [26] M. Schuld et al. “Evaluating analytic gradients on quantum hardware”. In: *Phys. Rev. A* 99.3 (2019), p. 032331. ISSN: 2469-9926. DOI: [10.1103/PhysRevA.99.032331](#). arXiv: [1811.11184v1](#).
- [27] R. M. Parrish et al. “Hybrid Quantum/Classical Derivative Theory: Analytical Gradients and Excited-State Dynamics for the Multistate Contracted Variational Quantum Eigensolver”. In: *Preprint* (2019). arXiv: [1906.08728](#).
- [28] A. Peruzzo et al. “A variational eigenvalue solver on a photonic quantum processor”. In: *Nat. Commun.* 5.1 (2014), p. 4213. ISSN: 2041-1723. DOI: [10.1038/ncomms5213](#).
- [29] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information: 10th Anniversary Edition*. 10th. New York, NY, USA: Cambridge University Press, 2011. ISBN: 1107002176, 9781107002173.
- [30] S. Sim, P. D. Johnson, and A. Aspuru-Guzik. “Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms”. In: *Preprint* (2019). arXiv: [1905.10876](#).
- [31] E. Knill. “Quantum computing with realistically noisy devices”. In: *Nature* 434.7029 (2005), pp. 39–44. ISSN: 0028-0836. DOI: [10.1038/nature03350](#). arXiv: [0410199 \[quant-ph\]](#).
- [32] D. Gottesman. “An introduction to quantum error correction and fault-tolerant quantum computation”. In: 2010, pp. 13–58. DOI: [10.1090/psapm/068/2762145](#). arXiv: [0904.2557](#).
- [33] J. R. McClean et al. “The theory of variational hybrid quantum-classical algorithms”. In: *New J. Phys.* 18.2 (2016), p. 023023. ISSN: 1367-2630. DOI: [10.1088/1367-2630/18/2/023023](#). arXiv: [1509.04279v1](#).
- [34] M. Abadi et al. “TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems”. In: *Preprint* (2016). arXiv: [1603.04467](#).
- [35] C. Gidney, D. Bacon, and The Cirq Developers. *quantumlib/Cirq: A python framework for creating, editing, and invoking Noisy Intermediate Scale Quantum (NISQ) circuits*. 2018.