December 14, 2021, OU-HET-1125, IPPP/21/54

Anomaly detection in high-energy physics using a quantum autoencoder

Vishal S. Ngairangbam,[a,b] **Michael Spannowsky,**[c,d] **and Michihisa Takeuchi**[e]

[a] *Theoretical Physics Division, Physical Research Laboratory,*
Shree Pannalal Patel Marg, Ahmedabad - 380009, Gujarat, India

[b] *Discipline of Physics, Indian Institute of Technology, Palaj,*
Gandhinagar - 382424, Gujarat, India

[c] *Institute for Particle Physics Phenomenology, Durham University,*
Durham DH1 3LE, United Kingdom

[d] *Department of Physics, Durham University,*
Durham DH1 3LE, United Kingdom

[e] *Department of Physics, Osaka University, Osaka 560-0043, Japan*

E-mail: vishalng@prl.res.in, michael.spannowsky@durham.ac.uk,
m.takeuchi@het.phys.sci.osaka-u.ac.jp

Abstract: The lack of evidence for new interactions and particles at the Large Hadron Collider has motivated the high-energy physics community to explore model-agnostic data-analysis approaches to search for new physics. Autoencoders are unsupervised machine learning models based on artificial neural networks, capable of learning background distributions. We study quantum autoencoders based on variational quantum circuits for the problem of anomaly detection at LHC. For a QCD $t\bar{t}$ background and a resonant heavy Higgs signals, we find that a simple quantum autoencoder outperforms dense classical autoencoders for the same input space and trains very efficiently. Moreover, this performance is reproducible on present quantum devices. This shows that quantum autoencoders can effectively analyse high-energy physics data in future LHC runs.

Keywords: Large Hadron Collider, Anomaly Detection, Quantum Machine Learning, Quantum Autoencoder

Contents

1 Introduction

In the absence of a confirmed new physics signal and in the presence of a plethora of new physics scenarios that could hide in the copiously produced LHC collision events, unbiased event reconstruction and classification methods[1–4] have become a major research focus of the high-energy physics community. Unsupervised machine learning frameworks [5–8], often used as anomaly-detection methods [9–13], are trained on Standard Model processes and should indicate if a collision event is irreconcilable with the kinematic features of events predicted by the Standard Model.

The most popular neural network-based approach is a so-called autoencoder [14]. Autoencoders consist of an encoder step that compresses the feature space into a latent space with reduced dimensionality. Subsequently, the latent space is decoded into an output of the same dimensionality as the input feature space. The entire network is then trained such that the loss function, which evaluates how well the output resembles the input, is minimised. The latent space acts as an information bottleneck, and its dimension is a hyperparameter of the network. The assumption is that the minimal dimension of the latent space for which the input features can still be reconstructed corresponds to the dimensionality of the information space required to describe the training sample, here Standard Model induced background processes. If the signal is kinematically sufficiently different from the background samples, the loss function or reconstruction error will be larger for signal than for background events. Such autoencoder can be augmented with convolutional neural networks [15, 16], graph neural networks [17, 18] or recurrent neural networks [19, 20] on its outset, making it a very flexible anomaly detection method for a vast number of use-cases.

With the advent of widely available noisy intermediate-scale quantum computers (NISQ) [21] the interest in quantum algorithms applied to high-energy physics problems has spurred. Today's quantum computers have a respectable quantum volume and can perform highly non-trivial computations. This technical development has resulted in a community-wide effort [22, 23] exploring the applications of quantum computers for studying quantum physics in general and in particular, the application to challenges in the theoretical description of particle physics. Some recent studies in the direction of LHC physics include evaluating Feynman loop integrals [24], simulating parton showers [25] and structure [26], quantum algorithm for evaluating helicity amplitudes [27], and simulating quantum field theories [28–33]. An interesting application of quantum computers is the nascent field of quantum machine learning–leveraging the power of quantum devices for machine learning tasks, with the capability of classical[1] machine learning algorithms for various applications at the LHC already recognised, it is only natural to explore whether quantum machine learning (QML) can improve the classical algorithms [34–41].

This work explores the feasibility and potential advantages of using quantum autoencoders (QAE) for anomaly detection. Most quantum algorithms consist of a quantum state, encoded through qubits, which evolves through the application of a unitary operator. The necessary compression and expansion of data in the encoding and decoding steps are manifestly non-unitary, which has to be addressed by the QAE using entanglement operations and reference states which disallow information to flow from the encoder to the decoder. To this end, a QAE should, in principle, be able to perform tasks ordinarily accomplished by a classical autoencoder (CAE) based on deep neural networks (DNN). The ability of DNNs are known to scale with data [42], and large datasets are necessary to bring out their better performance over other machine-learning algorithms. Interestingly, we find that a quantum autoencoder, augmented using quantum gradient descent [43] for its training, is much less dependent on the number of training samples and reaches optimal reconstruction performance with minuscule training datasets. Moreover, compared to CAEs, which use the same input variables as the QAE, QAEs have better anomaly detection capabilities, at least for the benchmark process we use in our study, i.e. $pp \to t\bar{t}$ for background and $pp \to H \to t\bar{t}$ for signal. This better performance is particularly interesting as the CAE has $\mathcal{O}(1000)$ parameters compared to just $\mathcal{O}(10)$ for the QAE. The study indicates the possibility to study quantum latent spaces of high-energy collisions, in analogy to classical autoencoders [17, 44–46].

The rest of the paper is organised as follows: In section 2, we present an introduction to classical autoencoders based on deep neural networks. We then describe the basic ideas of quantum machine learning and a quantum autoencoder in section 3. The details of the

[1] By classical, we mean any machine learning algorithm that leverages only discrete bit computations, while by quantum, we imply a computation that uses the properties of quantum mechanics and qubits, even if they are simulated on classical hardware.

data simulation, network architecture, and training we describe in section 4. We present the performance of a quantum autoencoder compared to a classical autoencoder in section 5. We conclude in section 6.

2 Classical Autoencoders

Autoencoders are neural networks utilised in various applications of unsupervised learning. They learn to map input vectors $\mathbf{x}$ to a *compressed* latent vector $\mathbf{z}$ via an encoder. This latent space feeds into a decoder that reconstructs the inputs. Denoting the encoder and decoder networks as $\mathbf{E}(\Theta_E, \mathbf{x})$ and $\mathbf{D}(\Theta_D, \mathbf{z})$ with Θ_E and Θ_D denoting the learnable parameters of the respective network, we have

$$\mathbf{z} = \mathbf{E}(\Theta_E, \mathbf{x}) \quad , \quad \hat{\mathbf{x}} = \mathbf{D}(\Theta_D, \mathbf{z}) \quad , \tag{2.1}$$

where $\hat{\mathbf{x}}$ denote the reconstructed output vector. The whole network is trained via gradient descent to reduce a faithful distance L, between the reconstructed output $\hat{\mathbf{x}}$ and the input vector $\mathbf{x}$. For instance L can be the root-mean-squared-error (RMSE),

$$L(\mathbf{x}, \hat{\mathbf{x}}) = \sqrt{\frac{\sum_{i=1}^{i=n} (\hat{x}^i - x^i)^2}{n}} \quad , \tag{2.2}$$

where $\hat{x}^i$ and x^i are the i^{th} component of the reconstructed and input vectors respectively, and n is their dimension. A faithful encoding should have an optimal latent dimension $k < n$, with k being the intrinsic dimension of the data set. This *dimensionality reduction* is crucial in many applications of autoencoders, which otherwise learns trivial mappings to reconstruct the output vectors $\hat{x}$. Unsupervised learning deals with learning probability distributions, and properly trained autoencoders are excellent for many applications. A dense CAE for a four feature input and two-dimensional latent space is shown in figure 1. The encoder and the decoders are also enclosed in red and blue boxes, respectively.

One popular usage of autoencoders in collider physics is anomaly detection. In various scenarios at the LHC, the background processes' contributions are orders of magnitude larger than most viable signals. However, a plethora of possible signal scenarios exist that could be realised in nature, making it unlikely that the signal-specific reconstruction techniques of supervised learning methods comprehensively cover all possible scenarios. This motivates unsupervised anomaly detection techniques, wherein a statistical model learns the probability distribution of the background to classify any data not belonging to it as anomalous (signal) data. Using an autoencoder as an anomaly detector, we train it to reconstruct the background data faithfully. Most signals generally have a higher intrinsic dimension than background data due to their increased complexity. Hence, they incur higher reconstruction losses. Thus, the loss function can be used as a discriminant to look for anomalous events.

3 Quantum autoencoders

Quantum machine learning broadly deals with extending classical machine learning problems to the quantum domain with variational quantum circuits [47]. We can divide these

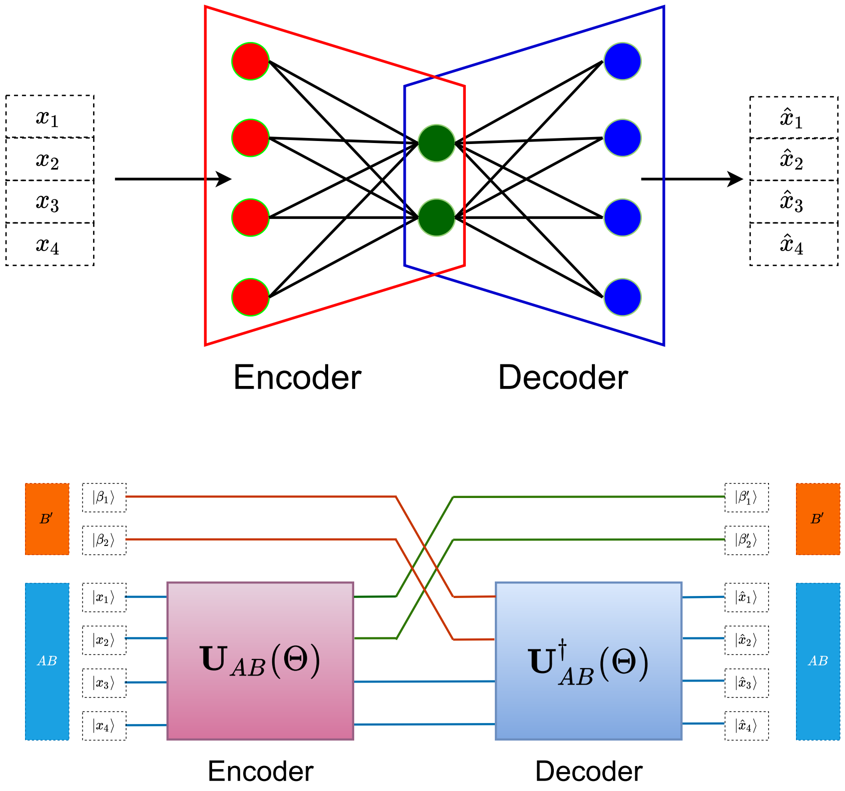

Figure 1: Schematic representation of a simple dense classical autoencoder (top) and a quantum autoencoder (bottom) for a four dimensional input space and a two dimensional latent space. To induce an information bottleneck in quantum unitary evolutions, we throw away states $|\beta'_i\rangle$ (trash states) at the encoder output (green lines), which are replaced by reference states $|\beta_i\rangle$ (shown in orange lines), containing no information of the input $|x_j\rangle$.

circuits into three blocks: a state preparation that encodes classical inputs into quantum states, a unitary evolution circuit that evolves the input states, and a measurement and post-processing part that measures the evolved state and processing the obtained observables further. For this discussion, we will always work in the computational basis with the basis vectors $\{|0\rangle, |1\rangle\}$ denoting the eigen states of the Pauli Z operator $\hat{\sigma}_z$ for each qubit.

There are many examples of state preparation in literature [48], which has their own merits in various applications. We prepare the states using *angle encoding*, which encodes real-valued observables ϕ_j as rotation angles along the x-axis of the Bloch sphere

$$|\Phi\rangle = \bigotimes_{i=1}^{n} R_x(\phi_j)\,|0\rangle = \bigotimes_{j=1}^{n} \left(\cos\frac{\phi_j}{2}\,|0\rangle - i\sin\frac{\phi_j}{2}|1\rangle\right)\;, \tag{3.1}$$

where $R_x = e^{-i\frac{\phi_j}{2}\hat{\sigma}_x}$ denote the rotation matrix. The number of qubits required n, is same as the dimensions of the input vector. A parametrised unitary circuit $\mathbf{U}(\Theta)$, with Θ

denoting the set of parameters, evolves the prepared state $|\Phi\rangle$ to a final state $|\Psi\rangle$,

$$|\Psi\rangle = \mathbf{U}(\Theta)\ |\Phi\rangle\ . \tag{3.2}$$

The final measurement step involves the measurement of an observable on the final state $|\Psi\rangle$. Since measurements in quantum mechanics are inherently probabilistic, we measure multiple times (called shots) to get an accurate result. In order to do that, we need quantum hardware that can prepare a large number of pure identical input states $|\Phi\rangle$ for each data point.

After defining a cost function, the parameters Θ can be trained and updated using an optimisation method. To better capture the geometry of the underlying Hilbert space and to achieve a faster training of the quantum network[2], we will use quantum gradient descent, modifying the Fubini-Study metric [49, 50].

While we have not discussed the specific form of the parametrised unitary operation $\mathbf{U}(\Theta)$, it is important to note that one of the major advantages of quantum computation is due to its ability to produce *entangled states*, a phenomenon absent in devices based on classical bits. The prepared input state is separable into the component qubits, and a product of unitaries acting on single-qubit states will not entangle the subsystems. The CNOT gate is a standard two-qubit gate, which will be used in our circuit to entangle the subsystems.

3.1 Variational Quantum autoencoders

Quantum autoencoders based on variational circuit models have been proposed for quantum data compression [51]. In our work, we want to learn the parameters of such a network to compress the background data efficiently. Along the same principles as anomaly detection on classical autoencoders, we expect that the compression and subsequent reconstruction will work poorly on data with different characteristics to the background.

A quantum autoencoder, in analogy to the classical autoencoders has an encoder circuit which evolves the input states $|\Phi\rangle$ to a latent state $|\chi\rangle$, via a unitary transformation $\mathbf{U}(\Theta)$, and then reconstructs the input state, via its hermitian conjugate $|\Phi\rangle = \mathbf{U}^{\dagger}(\Theta)|\chi\rangle$. However, note that since unitary transformations are probability conserving and act on spaces having identical dimensions, there is no data compression in such a set-up. In order to have data compression, some qubits at the initial encoding $|\chi\rangle$ are discarded and replaced by freshly prepared reference states. Such a setup for a four feature input and two dimensional latent space if shown in figure 1. The unitary operators output identical number of qubits, however at the encoder step, two of its outputs (shown by green lines) are replaced by freshly prepared reference states (shown in orange lines), devoid of any information of the input states. We describe the basics of quantum autoencoding in the following, mainly based on the discussion of quantum autoencoders for data compression from ref. [51].

[2] See [37] for a brief presentation of the Fubini-Study metric and a comparison of natural and quantum gradient descent for the training of classical and quantum networks. It was shown that quantum gradient descent improves the training of a variational quantum circuit significantly.

Let us denote the Hilbert space containing the input states by $\mathcal{H}$. For describing a quantum autoencoder, it is convenient to expand $\mathcal{H}$ as the product of three subspaces,

$$\mathcal{H} = \mathcal{H}_A \otimes \mathcal{H}_B \otimes \mathcal{H}_{B'} \ , \tag{3.3}$$

with subspace $\mathcal{H}_A$ denoting the space of qubits fed into the decoder from the encoder, and $\mathcal{H}_B$ denoting the space corresponding to the ones that are re-initialised, and $\mathcal{H}_{B'}$ denoting the Hilbert space containing the reference state. In the following, we will denote states belonging to any subspace with suffixes while the full set will have no suffix. For example, $|a\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$, $|\kappa\rangle \in \mathcal{H}$, $|b\rangle_{B'} \in \mathcal{H}_{B'}$ etc. We will use the same convention for operators acting on the various subspaces.

Since we entangle the separable input qubits in the subspaces $\mathcal{H}_A \otimes \mathcal{H}_B$ via $\mathbf{U}_{AB}(\Theta)$, the latent state $|\chi\rangle_{AB} \in \mathcal{H}_A \otimes \mathcal{H}_B$, in general, is not seperable. The input of the larger composite system including the reference state is $|\Phi\rangle_{AB} \otimes |\beta\rangle_{B'}$, with $|\beta\rangle_{B'}$ denoting a freshly prepared *reference state* not acted on by the unitary $\mathbf{U}_{AB}$. The process of encoding can be therefore written as,

$$|\chi\rangle_{AB} \otimes |\beta\rangle_{B'} = (\mathbf{U}_{AB}(\Theta) \otimes \mathbf{I}_{B'})\ |\Phi\rangle_{AB} \otimes |\beta\rangle_{B'} \quad , \tag{3.4}$$

where $\mathbf{I}_{B'}$ denotes the identity operator on $\mathcal{H}_{B'}$. Explicitly, the dimensions of the subspaces $\mathcal{H}_A$, $\mathcal{H}_B$, and $\mathcal{H}_{B'}$ are $2^{N_{lat}}$, $2^{N_{trash}}$, and $2^{N_{trash}}$, respectively, where N_{lat} is the number of qubits passed to the decoder directly from the encoder, while N_{trash} are the ones that are discarded. Swapping the B and B', gives the input to the decoder as

$$|\chi'\rangle = \mathbf{I}_A \otimes \mathcal{V}_{BB'}\ \ |\chi\rangle_{AB} \otimes |\beta\rangle_{B'} \ , \tag{3.5}$$

where $\mathcal{V}_{BB'}$ indicates a unitary that performs the swap operation[3], and $\mathbf{I}_A$ is the identity operator on $\mathcal{H}_A$. The output of the decoder can now be written as

$$|\Psi\rangle = \mathbf{U}^\dagger_{AB}(\Theta) \otimes \mathbf{I}_{B'}\ |\chi'\rangle \ , \tag{3.6}$$

with $\mathbf{I}_{B'}$ being the identity operator on $\mathcal{H}_{B'}$. The decoding, therefore, takes the swapped latent state $|\chi'\rangle$, and the unitary $\mathbf{U}^\dagger_{AB}$ evolves it with no information from the encoder in the subspace $\mathcal{H}_B$. The reconstruction efficiency of the autoencoder can be quantified in terms of the *fidelity* between the input and output states in the subspace $\mathcal{H}_A \otimes \mathcal{H}_B$, which quantifies their similarity. For two quantum states $|\psi\rangle$ and $|\phi\rangle$, it is defined as

$$F(|\phi\rangle, |\psi\rangle) = F(|\psi\rangle, |\phi\rangle) = |\langle\phi|\psi\rangle|^2 \ .$$

[3] For instance swapping the state of two qubits in the basis $\{|00\rangle, |01\rangle, |10\rangle, |11\rangle\}$, can be implemented via the unitary matrix

$$\mathcal{V}_{BB'} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad .$$

For normalized states, we have $0 \leq F \leq 1$, with $F = 1$ only when $|\phi\rangle$ and $|\psi\rangle$ are exactly identical. We can write the fidelity of the complete system as

$$F(|\Phi\rangle_{AB} \otimes |\beta\rangle_{B'}, |\Psi\rangle) = F(|\Phi\rangle_{AB} \otimes |\beta\rangle_{B'}, \mathbf{U}^{\dagger}_{AB}\, \mathcal{V}_{BB'}\, \mathbf{U}_{AB}\, |\Phi\rangle_{AB} \otimes |\beta\rangle_{B'}) \ ,$$

where we have implicitly assumed that the unitary operators are extended to the whole space via a direct product with the identity operator on the subspace it does not act on, for notational compactness. Noting that $\mathbf{U}_{AB}|\Phi\rangle_{AB} = |\chi\rangle_{AB}$, we can write this as,

$$F(|\Phi\rangle_{AB} \otimes |\beta\rangle_{B'}, |\Psi\rangle) = F(|\chi\rangle_{AB} \otimes |\beta\rangle_{B'}, \mathcal{V}_{BB'}\, |\chi\rangle_{AB} \otimes |\beta\rangle_{B'}) \ .$$

Writing the swaped state as $\mathcal{V}_{BB'}\ |\chi\rangle_{AB} \otimes |\beta\rangle_{B'} = |\chi\rangle_{AB'} \otimes |\beta\rangle_{B}$, we have

$$F(|\Phi\rangle_{AB} \otimes |\beta\rangle_{B'}, |\Psi\rangle) = F(|\chi\rangle_{AB} \otimes |\beta\rangle_{B'}, |\chi\rangle_{AB'} \otimes |\beta\rangle_{B}) \ . \tag{3.7}$$

Since we are interested in the wave functions belonging to the subspace $\mathcal{H}_A \otimes \mathcal{H}_B$, we trace over B' to get the required fidelity. However, a perfect fidelity between the input and outputs of the AB system can be achieved when the complete information of the input state passes to the decoder, i.e.

$$\mathbf{U}_{AB}|\Phi\rangle_{AB} = |\Phi^c\rangle_A \otimes |\beta\rangle_B \quad . \tag{3.8}$$

The state $|\Phi^c\rangle_A$ denotes a compressed form of $|\Phi\rangle_{AB}$, i.e it should contain the information of the AB system in the input, while $|\beta\rangle_B$ is equivalent to the reference state, with no information of the input. If the B and B' systems are identical during the swap operation, the entire circuit reduces to the identity map. The output of the B' system, hereby referred to as the *trash state*, is itself the determining factor of the output state fidelity. The output of the B' system can be obtained after tracing over the A system as: $\hat{\rho}_{B'} = \mathrm{Tr}_A\ \{|\chi\rangle\langle\chi|_{AB'}\}$ and the required fidelity of the B' system is $F(|\beta\rangle_{B'}, \hat{\rho}'_B)$.

A perfect reconstruction of the input is possible only when the trash state fidelity $F(|\beta\rangle_{B'}, \hat{\rho}'_B) = 1$. Thus a quantum autoencoder can be trained by maximising the trash state fidelity instead of the output fidelity, which has the advantage of reducing the resource requirements during training. Although, the output fidelity obtained by tracing over the B' system is numerically not equal to the trash state fidelity, we can use the later in anomaly detection as well, since it is a faithful measurement of the output fidelity. Thus, unlike vanilla classical autoencoders, we can reduce the execution and training of QAEs into the encoder circuit for anomaly detection.

The above discussions have focused on the underlying principles behind a quantum autoencoding process on single input states. As stated before, we need to prepare identical input-states for each data point and repeat the unitary evolution and measurement to get a useful estimate of the fidelity, evident also from the use of density operators to express the output state. Referring to the ensemble of the input states as $\{p_i, |\Phi_i\rangle_{AB}\}$, we obtain for the cost function

$$C(\Theta) = -\sum_i p_i\, F(|\beta\rangle_{B'}, \hat{\rho}'_B) \quad , \tag{3.9}$$

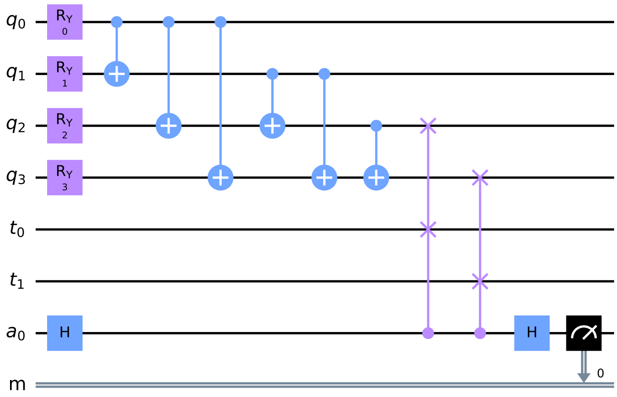

Figure 2: Quantum autoencoder circuit used for a four qubit input and two latent qubits. The inputs are embedded in q_i, which are then rotated by tunable angles in the y-direction of the Bloch sphere by R_y gates (shown in purple boxes). Each pair of these qubits are entangled via CNOT gates(shown with blue lines). For the trash training, we need a two dimensional reference state denoted by t_i qubits, and an ancilliary qubit $\mathtt{a}_0$. The fidelity between two qubits at the encoder output and the reference states is then measured via a SWAP test.

where the negative sign converts the optimisation process into minimising the cost function. It is important to note that the ensemble should not be taken as being analogous to the batch training in classical neural networks, as it is required for the accurate prediction of the network output even when testing the autoencoder network.

4 Analysis Setup

4.1 Data simulation

The background and the signal samples used in our analysis consist of the QCD $t\bar{t}$ continuum production, $pp \to t\bar{t}$, and the scalar resonance production $pp \to H \to t\bar{t}$, respectively. The background and the signal events are generated with a centre-of-mass energy of 14 TeV, as expected during future LHC runs. We require both to decay leptonically for the top quarks and focus on the decay into muons exclusively. We consider five different masses of the scalar resonance, $m_H = 1.0, 1.5, 2.0,$ and 2.5 TeV. All events are generated with `MadGraph5_aMC@NLO` [52], and showered and hadronization is performed by `Pythia8` [53]. `Delphes3` [54] is utilized for the detector simulation, where the jets are clustered using `FastJet` [55]. We generate about 30k events for the background samples, while for each signal sample, we generate about 15k. The background events are divided into 10k training, 5k validation and 15k testing samples.

For the object reconstruction, a standard jet definition using the anti-k_t algorithm [56] with the jet radius $R = 0.5$ is used. For the signal bottom jets, the output from Delphes 3 is used and require $p_T^b > 30$ GeV. For isolated leptons, we require $p_T^l > 30$ GeV and its isolation criteria with $R = 0.5$. We extracted four variables $\{p_T^{b_1}, p_T^{l_1}, p_T^{l_2}, \not{E}_T\}$ for our analysis, keeping in mind the limitations of current devices. To conserve the aperiodic topology of these variables in the angle embedding (given in eq. 3.4) we fix the range of each variable to $[0, 1000]$ by adding two points[4] and map it to a range $[0, \pi]$. The two added points are then removed from the dataset. This maps each feature's minimum and maximum to two distinct angles separated by a finite distance due to the selection criteria.

4.2 Network architecture and training

The QAE was implemented and trained using `Pennylane` [57]. As stated before, we train and test the QAE model with only the encoder circuit. After the input features are embedded as the rotation angle of the x-axis in the Bloch sphere, the unitary evolution $\mathbf{U}(\Theta)$ consists of two stages. In the first step, each qubit is rotated by an angle θ_i in the y-axis of the Bloch sphere. The values of these angles are to be optimized via gradient descent. After this, we apply the CNOT gate to all the possible pairs of qubits, with the ordering determined by the explicit number of the qubit. This circuit is shown in figure 2 for a four qubit input QAE with two-qubit latent space.

The number of qubits discarded at the encoder, the size of the trash-state, fixes the latent dimension[5] via $N_{lat} = N_{in} - N_{trash}$, with N_{lat} the latent dimension, N_{in} the size of the input state, and N_{trash} the number of discarded qubits. The reference state $|\beta\rangle_{B'}$, has the same number of qubits N_{trash}, and it is initialized to be

$$|\beta\rangle_{B'} = |0\rangle^{\otimes N_{trash}} \quad .$$

We measure the fidelity between the trash-state $\hat{\rho}_{B'}$ and the reference state $|\beta\rangle_{B'}$ via a SWAP test [58]. For any two states $|\phi\rangle$ and $|\psi\rangle$ with the same dimensions, the fidelity $F(|\phi\rangle, |\psi\rangle)$ can be measured as the output of an ancilliary qubit $|a\rangle_{anc}$ after the following operation,

$$\mathbf{H}_{anc} \otimes \mathbf{I} \text{ (c-SWAP) } \mathbf{H}_{anc} \otimes \mathbf{I} \;\; |0\rangle_{anc} \otimes |\phi\rangle \otimes |\psi\rangle \quad , \tag{4.1}$$

where $\mathbf{H}_{anc}$ is the Hadamard gate acting on the ancilliary qubit, and c-SWAP is the controlled swap operation between the states $|\phi\rangle$ and $|\psi\rangle$ controlled by the ancilliary qubit. Thus the total number of qubits required for a fixed N_{in} and N_{trash} is $N_{in} + N_{trash} + 1$. Due to the limitation of current quantum devices we limit the input feature to four, and scan over the possible latent dimensions.

The quantum network is trained by minimising the cost function (c.f eq. 3.9) with quantum gradient descent for the one, two and three-dimensional latent spaces. We train these instances for different training sizes of 10, 100, 1000 and 10000 events to study the

[4] Events with the variables lying above 1000 GeV are very rare and excluded in our case. In a realistic analysis, the upper bound can be determined from the data.

[5] In our discussion, we will use the number of latent qubits as the latent dimension, although the Hilbert space would have $2^{N_{lat}}$ dimensions.

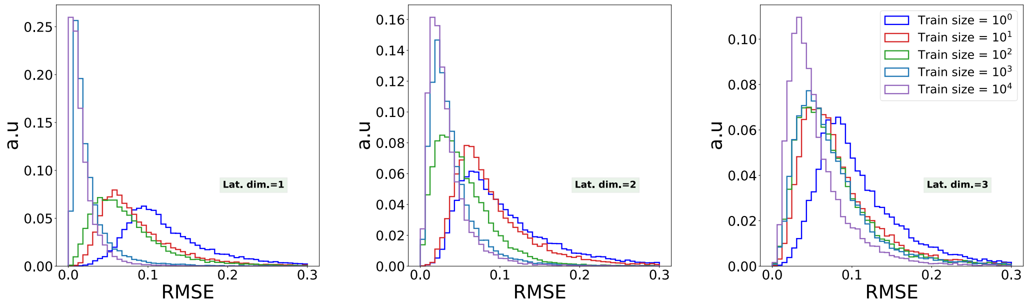

Figure 3: Loss distribution of the test background samples (15k) for different sizes of training dataset. We can see that the distribution shifts significantly towards the left (direction with lower loss) as one increases the training data size, which reflects that there is noticeable incraase in learning with larger data samples.

dependence of the QAE's performance on the size of the training data. We update the weights for each data sample, with 5000 shots in all training scenarios. For training sizes greater than or equal to 100, we train the networks for 50 epochs. In comparison, for sample sizes 1 and 10, we train the QAE for 500 and 200 epochs, respectively. To benchmark the performance of a QAE on a quantum computer, we train a QAE with the two inputs $p_T^{l_1}$ and $p_T^{b_1}$ with quantum-gradient descent on `Pennylane`, and compare the test performance with the simulation and the IBM-Q. For running on the IBM-Q, we build and implement the test circuits in `Qiskit` [59]. We also train classical autoencoders using `Keras-v2.4.0` [60] with `Tensorflow-v2.4.1` [61] for the same input features, for comparison. The encoder is a dense network mapping the input space to a latent dimension of $N_{lat} \in \{1, 2, 3\}$, and has three hidden layers with 20, 15, and 10 nodes. The hidden layers have `ReLU` activations while the latent output has `Linear` activation. The decoder has a symmetric configuration to the encoder. The networks are trained with `Adam` [62] optimiser with a learning-rate of 10^{-3} to minimise the root-mean-squared error between the input vector $\mathbf{x}$ and the reconstructed vector $\hat{\mathbf{x}}$. For the CAEs, we found that training with single data per update (technically batch size=1) has a volatile validation loss per epoch, with slow convergence. Therefore, we choose a batch size of 64 to train the CAEs.[6]

5 Results

Results of the various training scenarios are presented in this section. All results except the benchmarking on the IBM-Q are for the four variable input autoencoders. We first present a comparison of the QAE and CAE's background reconstruction efficiency and their dependence on the size of training data. We then quantify the performance of both autoencoders for the various signal scenarios. This is followed by a semi-realistic anomaly detection

[6] The network performs one update per epoch for training with a number of samples less than 64. These training sizes are too small for a CAE to have any good learning capability. Hence, we do not try to modify the batch sizes or interpret the test distributions.

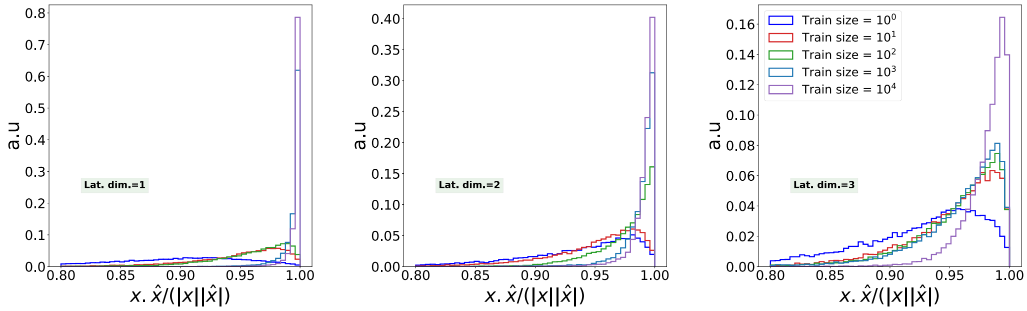

Figure 4: Cosine similarity (analogous to quantum fidelity) distribution of the test background samples (15k) for different sizes of training dataset of the CAE.

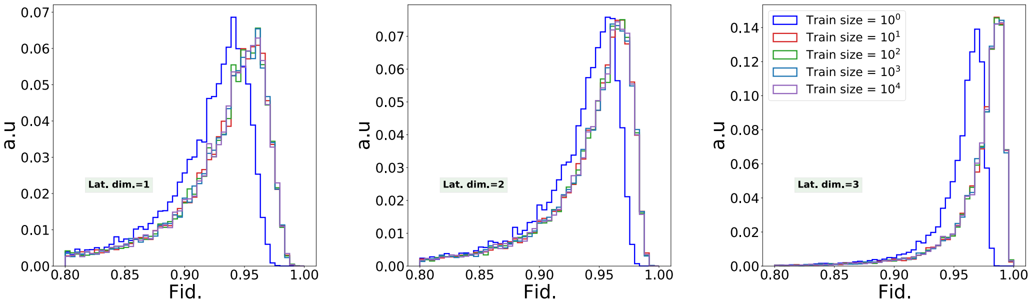

Figure 5: Fidelity distribution of the test background samples (15k) for different sizes of training dataset. The peak shifts towards the right in analogy to the CAE, however the shift is not as pronounced. With a single training sample, the network is not able to converge completely while for anything greater than 10, the increase in training size has practically no effect.

scenario, where we scale the events by different cross sections. Finally, we benchmark the performance of a two feature QAE on a real quantum device.

5.1 Dependence of test reconstruction efficiency on the number of training samples

The distribution of the loss function of the independent background test samples for different training sizes of the CAE is shown in figure 3. Although training with a single data point is inherently inaccurate, we perform such an exercise as a sanity check of the CAE's comparison to a QAE. The test distribution shifts towards the left as one increases the training size, thereby signifying increased reconstruction efficiency. For training sizes of up to 10^2, the limited statistics will produce a very high statistical uncertainty. Since it is not the main emphasis of our present work, we do not comment any further. Looking at the distribution across different latent dimensions for 10^3 and 10^4 training samples, one can see the impact of the information bottleneck. For a singular latent dimension, the passed information is already available from 10^3 samples, and hence the loss distribution is very close to the one trained on 10^4. This relative separation increases as we go to higher

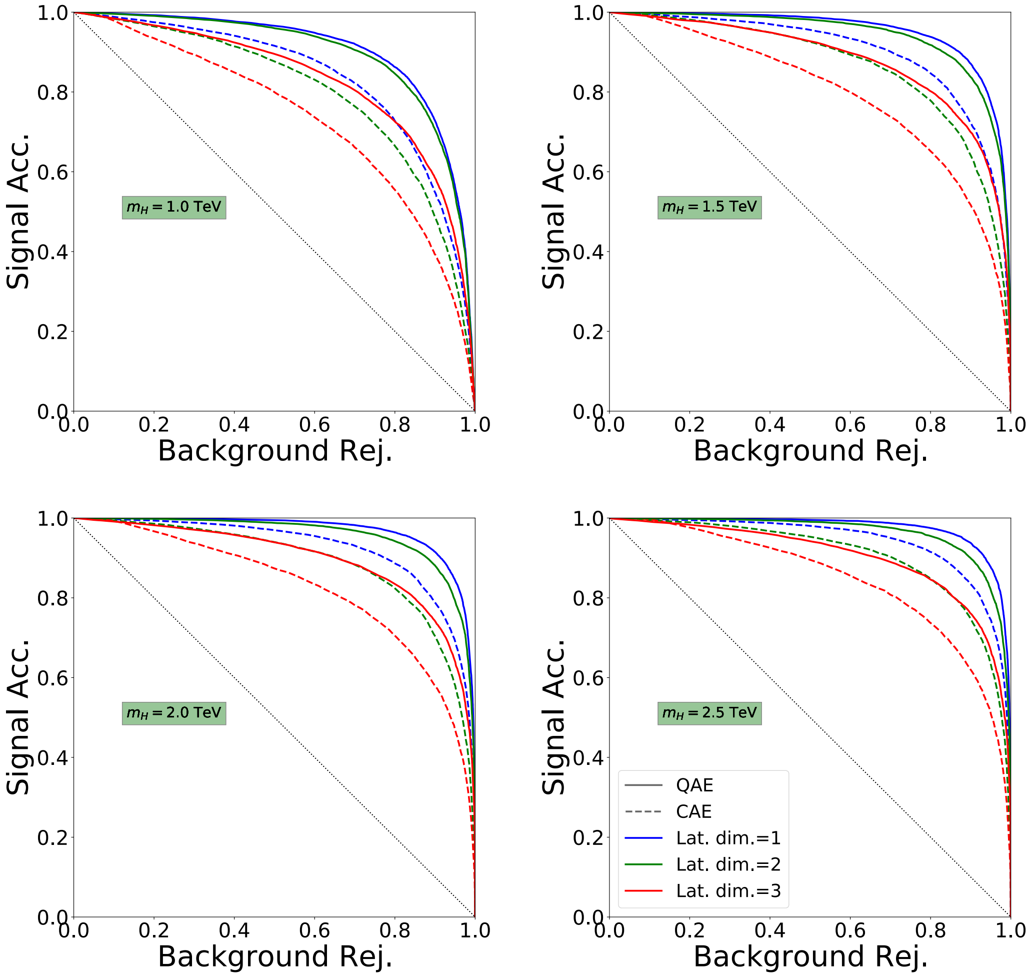

Figure 6: ROC curve between signal acceptance vs background rejection for Quantum Autoencoder (QAE) and Classical Autoencoder (CAE) for various values of m_H and different latent dimensions for a training datasize of 10k samples. The trend across latent dimensions is same for both QAE and CAE with QAEs performing better in all cases.

latent dimensions, denoting the higher information passed to the decoder to reconstruct the input, which is exploited with higher training samples. For an analogous comparison with the quantum fidelity, we define the cosine similarity between the input vector $\mathbf{x}$ and the reconstructed vector $\hat{\mathbf{x}}$ as,

$$\cos\alpha = \frac{\mathbf{x}.\hat{\mathbf{x}}}{|\mathbf{x}|\;|\hat{\mathbf{x}}|}\;, \tag{5.1}$$

where the dot product is done with a Euclidean signature. The distribution of the cosine similarity shown in figure 4, shows similar features to the loss function's distribution, with efficient reconstruction possible only when the train size is at least 10^3.

We have seen that CAEs cannot be trained with limited statistics to reconstruct the statistically independent test dataset. From the distribution of the test sample's fidelity in figure 5, we see that QAEs are much more effective in learning from small data samples. Although training with a single data point has not reached the optimal reconstruction efficiency, it is obtained with ten sample events. Unlike CAEs, see figs. 3 and 4, the test fidelity distribution for all latent spaces are identical for training sizes greater than or equal to ten. The independence of the sample size is particularly important in LHC searches where the background cross section is small. This particularly interesting feature may be due to the interplay of an enhancement of statistics via the uncertainty of quantum measurements and the relatively simple circuits employed in our QAE circuit. For a single input point and assuming that we have hardware capable of building exact copies, a finite number of measurement processes always introduces a non-zero uncertainty in the network output. This uncertainty can act as additional information in the quantum gradient minimisation, which is performed after the measurement process, increasing the convergence for smaller data samples.

5.2 Classification Performance

We compare the QAE and the CAE's performance for the four-dimensional input feature space. The metrics used in this presentation bear similarity to those used in a supervised framework. It also assumes that a randomly chosen event is equally likely to be either the background or the signal. This assumption is not sound in the context of LHC searches or in an anomaly detection technique since the background's cross-section is orders of

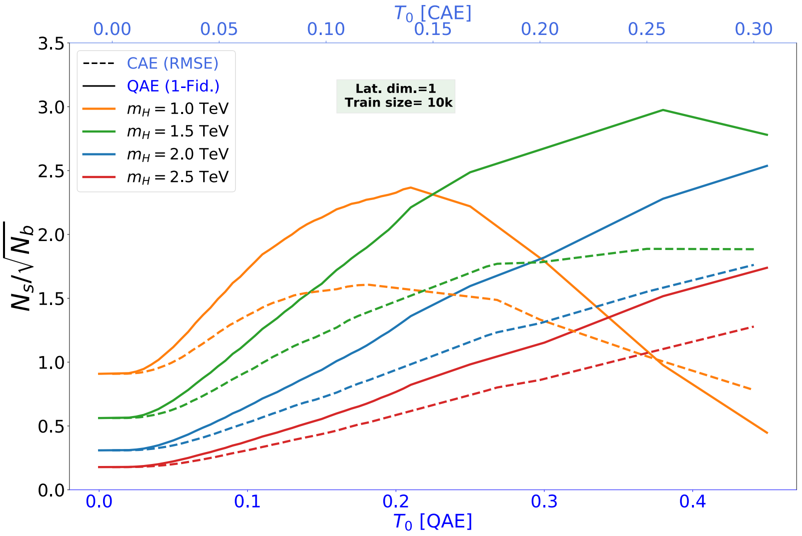

Figure 7: Significance as a function of the threshold T_0 on the fidelity and root-mean-squared-error (RMSE) of the QAE and the CAE, respectively, for each of the signal scenarios and singular latent dimension trained on 10k samples. To keep the signal rich region on the right side for both, we have used (1 − Fidelity) for the QAE. We fix the cross-section of all signals to 10 fb, and evaluate the yields at an integrated luminosity of 3000 fb^{-1}.

magnitude larger than that of the signal. Nevertheless, they are handy when comparing different classifiers.

For each value of m_H, we plot the Receiver-Operator-Characteristics (ROC) curve between the signal acceptance and the background background rejection in figure 6, for the networks trained with 10k samples. The black dotted lines denote the performance of a random classifier with no knowledge of either the signal or the background, and the lines further away from it indicate better performance than those in its vicinity. The performance reduces with increasing latent dimensions for CAEs and QAEs, with the highest background rejection coming for a singular latent dimension. Comparing the QAEs and the CAEs (dotted vs solid lines for each colour), we find that QAEs perform better than CAEs consistently in all latent dimensions and the different values of m_H. This better performance may be a universal property of QAEs. However, as our analysis is a proof-of-concept, an in-depth exploration of the properties of QAEs in general and anomaly detection at colliders, in particular, is needed to affirm this observation.

5.3 Anomaly detection

We now explore the performance of the autoencoders in a semi-realistic search scenario. When we scale the normalization of the signal and the background by their respective probability of occurrence, i.e. their respective cross-sections, we are essentially in an anomaly detection scenario since the background is orders of magnitude larger than the signal. The performance of the autoencoders can then be quantified in terms of statistical significance as a function of the threshold applied on the loss. For the background, we scale the cross-section obtained from Madgraph by a global k-factor of 1.8 [63], while for all the signal masses, we fix a reference value of 10 fb. The yield is then calculated as

$$N_p = \epsilon_p \, \sigma_p \, L \, E_p(T_0) \quad ,$$

where ϵ_p is the baseline selection efficiency, σ_p the cross-section, and $E_p(T_0)$ the efficiency at a threshold T_0 of the loss distribution, for a process p, while L is the integrated luminosity which we take to be 3000 fb^{-1}.

Since it is natural to use the best classifier in a search, we evaluate the significance of the autoencoders with one latent dimension, trained on 10k samples. We apply the threshold for the QAE and the CAE on the quantum trash state fidelity and the RMSE loss, respectively. We use $(1-\text{Fidelity})$ for the QAE to make the signal-rich regions same in both scenarios. RMSE loss is chosen over the cosine similarity since the former was found to have a higher performance. The significance $N_S/\sqrt{N_B}$ for each of the signal masses as a function of the threshold T_0 is shown in figure 7. We fix the threshold range so that there are enough background test statistics in the least background like bin. Looking at the peak of the significance, we note that QAEs outperform CAEs, which is only natural from the preceding discussions. However, an interesting development is the relative performance for the different masses. Even though the ROCs indicated higher discrimination with increasing mass, the significance increases for $m_H = 1.0$ TeV to 1.5 TeV and decreases for higher masses. Since we have fixed a fiducial cross-section for each signal mass, it plays no role in this irregularity. The trend arises via an interplay between the higher discrimination by the autoencoder output and the decrease in baseline efficiency with increasing mass m_H. The decreasing selection efficiency is due to the isolation criteria of the jets and the leptons, which would be naturally boosted when we go to higher resonant masses m_H, thereby becoming more collimated.

5.4 Benchmarking on a quantum device

We now compare the performance of the quantum simulator and the actual quantum hardware. Since there is a limitation for the available number of qubits, we limit the feature space in two dimension, which consists of $\{p_T^{b_1}, p_T^{l_1}\}$. For our QAE setup, in addition to the two qubits for embedding the input features, one qubit for the reference state and another ancilliary qubit for the SWAP test are needed. We use the simpler version of the quantum circuit shown in figure 2, which is implemented and trained using `PennyLane`. To compare the performance, we use the same circuit with the same optimized parameters both for `PennyLane` and for the IBM-Q belem backend. Accessing the IBM hardware was done through `Qiskit`.

In figure 8, we show the fidelity distributions for the background and the signal samples for our QAE circuit with the optimized circuit parameters computed by the simulator in `Pennylane` and in the actual quantum device of IBM-Q belem backend. Since IBM-Q does not let us run the CSWAP operation with a reasonably shallow circuit, the fidelity distributions are smeared toward 0.5, and especially it is worse around 1. One of the advantages of using the SWAP test is to reduce the number of qubits for the evaluations of the fidelity during the optimization process. For example, to check the performance of the current circuit, directly measuring the fidelity between the reference state and the output for the second qubit would be enough. It can be achieved by the simple Pauli

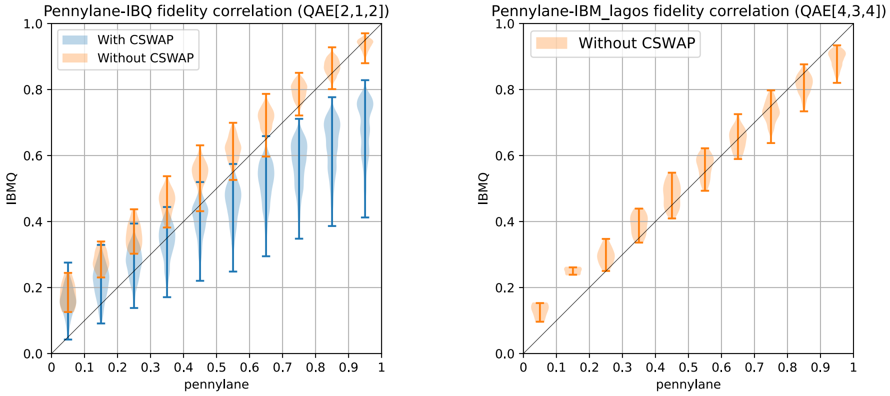

Figure 8: The correlation between the fidelity values obtained by `Pennylane` and by the IBM-Q backends. On the left we show the comparison of a 2-1-2 QAE, where we directly measure the trash state (orange) and with a SWAP test employing a CSWAP gate. We find that the shallower implementation without the CSWAP gate has lesser decoherence effects, and hence better agreement with the simulation. The correlation with the direct measurement for the 4-3-4 case is shown on the right.

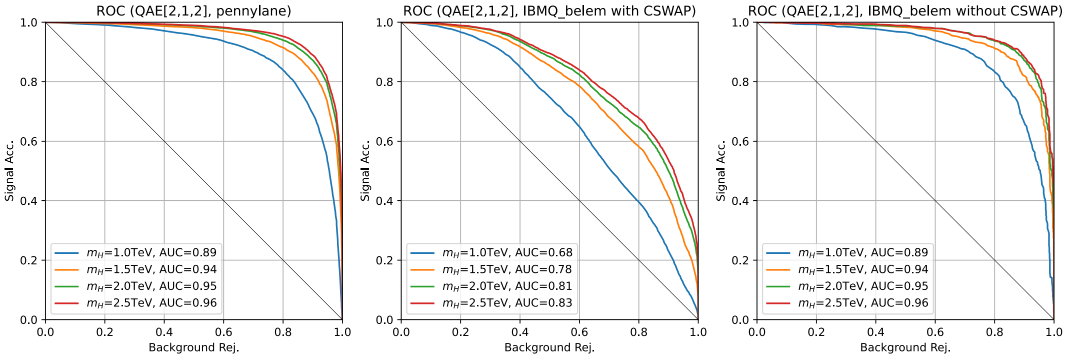

Figure 9: ROC curves based on the fidelity distributions. Those evaluated by the `Pennylane` simulator (left panel), by the quantum device IBM-Q belem backend with the SWAP test (central panel), and with the second qubit measurements (right panel) are shown.

z measurements. The correlation of the fidelities obtained by `Pennylane` and by IBM-Q belem, based on the SWAP test and on the Pauli z measurement are shown in the right panel as the violin plots, in blue and in orange, respectively. The correlation is better for the Pauli z measurements for the same circuit part with the identical input parameters. It suggests that the decoherence effects from a deeper circuit obscure the performance.

In figure 9, we show the ROC curves based on the fidelity distributions for the background and the signal samples evaluated by `Pennylane` simulator in the left panel. The central panel shows the ROC curves based on the fidelities evaluated by the SWAP test, while the right panel shows those by the second qubit Pauli z measurements, for the same IBM-Q device of belem backend. As one can see, the performances based on the Pauli z measurements on the IBM-Q device follow those obtained by the `Pennylane` simulator. The AUCs for them are also essentially the same. Thus, the deficit in the performance with the SWAP test is due to the too deep circuit realization for the CSWAP operation in the IBM-Q device. Therefore, the realisation of a CSWAP operation with a shallow circuit is necessary.

To check the efficacy of quantum hardware for the four input QAE, we evaluate the trash state fidelity of a QAE with four-dimensional feature space. Due to hardware limitations discussed above, we estimate it without the SWAP test for a single trash qubit giving us a three-dimensional latent space. The correlation between the `Pennylane` evaluated fidelity and the output from IBM-Q lagos, shown in figure 8, displays a good agreement between the simulation and the hardware.

6 Conclusion

The lack of evidence for new interactions and particles at the Large Hadron Collider has motivated the high-energy physics community to explore model-agnostic data-driven approaches to search for new physics. Machine-learning anomaly detection methods, such as autoencoders, have shown to be a powerful and flexible tool to search for outliers in data. Autoencoders learn the kinematic features of the background data by training the network to minimise the reconstruction error between input features and neural network output. As the kinematic characteristics of the signal are different to the background, the reconstruction error for the signal is expected to be larger, allowing signal events to be identified as anomalous.

Although quantum architecture capable of processing huge volumes of data is not yet feasible, noisy-intermediate scale devices could have very real applications at the Large Hadron Collider in the near future. With the origin of the collisions being quantum-mechanical, a quantum autoencoder could, in principle, learn quantum correlations in the data that a bit based autoencoder fails to see. We have shown that quantum-autoencoders based on variational quantum circuits have potential applications as anomaly detectors at the Large Hadron Collider. Our analysis shows that for the scenario we consider, i.e. the same set of input variables, quantum autoencoders outperform dense classical autoencoders based on artificial neural networks, asserting that quantum autoencoders can indeed go beyond their classical counterparts. They are very judicious with data and converge with very small training samples. This independence opens up the possibility of training quantum autoencoders on small control samples, thereby opening up data-driven approaches to inherently rare processes.

Acknowledgements

We acknowledge the use of IBM-Q in this work and thank Simon Williams for technical support. M.S. is supported by the STFC under grant ST/P001246/1. VSN is supported by the Physical Research Laboratory (PRL), Department of Space, Government of India. We thank Anupam Ghosh, Partha Konar, and Sudipta Show for valuable discussions on quantum machine learning. MT is supported, in part, by the Grant-in-Aid for Scientific Research C, No. 18K03611, the Grant-in-Aid on Innovative Areas, the Ministry of Education, Culture, Sports, Science and Technology, No. 16H06492, and by the JSPS KAKENHI Grant No. 20H00160.

References

[1] ATLAS collaboration, *Dijet resonance search with weak supervision using $\sqrt{s} = 13$ TeV pp collisions in the ATLAS detector*, *Phys. Rev. Lett.* **125** (2020) 131801 [2005.02983].

[2] CMS collaboration, *MUSiC: a model-unspecific search for new physics in proton–proton collisions at $\sqrt{s} = 13$ TeV*, *Eur. Phys. J. C* **81** (2021) 629 [2010.02984].

[3] G. Kasieczka et al., *The LHC Olympics 2020: A Community Challenge for Anomaly Detection in High Energy Physics*, 2101.08320.

[4] T. Aarrestad et al., *The Dark Machines Anomaly Score Challenge: Benchmark Data and Model Independent Event Classification for the Large Hadron Collider*, 2105.14027.

[5] K. Fraser, S. Homiller, R. K. Mishra, B. Ostdiek and M. D. Schwartz, *Challenges for Unsupervised Anomaly Detection in Particle Physics*, 2110.06948.

[6] B. Ostdiek, *Deep Set Auto Encoders for Anomaly Detection in Particle Physics*, 2109.01695.

[7] A. Andreassen, I. Feige, C. Frye and M. D. Schwartz, *JUNIPR: a Framework for Unsupervised Machine Learning in Particle Physics*, *Eur. Phys. J. C* **79** (2019) 102 [1804.09720].

[8] T. Dorigo, M. Fumanelli, C. Maccani, M. Mojsovska, G. C. Strong and B. Scarpa, *RanBox: Anomaly Detection in the Copula Space*, 2106.05747.

[9] R. T. D'Agnolo and A. Wulzer, *Learning New Physics from a Machine*, *Phys. Rev. D* **99** (2019) 015014 [1806.02350].

[10] J. A. Aguilar-Saavedra, J. H. Collins and R. K. Mishra, *A generic anti-QCD jet tagger*, *JHEP* **11** (2017) 163 [1709.01087].

[11] A. Blance, M. Spannowsky and P. Waite, *Adversarially-trained autoencoders for robust unsupervised new physics searches*, *JHEP* **10** (2019) 047 [1905.10384].

[12] V. Mikuni, B. Nachman and D. Shih, *Online-compatible Unsupervised Non-resonant Anomaly Detection*, 2111.06417.

[13] E. Govorkova et al., *Autoencoders on FPGAs for real-time, unsupervised new physics detection at 40 MHz at the Large Hadron Collider*, 2108.03986.

[14] M. A. Kramer, *Nonlinear principal component analysis using autoassociative neural networks*, *AIChE Journal* **37** (1991) 233 [https://aiche.onlinelibrary.wiley.com/doi/pdf/10.1002/aic.690370209].

[15] T. Heimel, G. Kasieczka, T. Plehn and J. M. Thompson, *QCD or What?*, *SciPost Phys.* **6** (2019) 030 [1808.08979].

[16] M. Farina, Y. Nakai and D. Shih, *Searching for New Physics with Deep Autoencoders*, *Phys. Rev. D* **101** (2020) 075021 [1808.08992].

[17] O. Atkinson, A. Bhardwaj, C. Englert, V. S. Ngairangbam and M. Spannowsky, *Anomaly detection with Convolutional Graph Neural Networks*, *JHEP* **08** (2021) 080 [2105.07988].

[18] S. Tsan, R. Kansal, A. Aportela, D. Diaz, J. Duarte, S. Krishna et al., *Particle Graph Autoencoders and Differentiable, Learned Energy Mover's Distance*, in *35th Conference on Neural Information Processing Systems*, 11, 2021, 2111.12849.

[19] D. D'Avino, D. Cozzolino, G. Poggi and L. Verdoliva, *Autoencoder with recurrent neural networks for video forgery detection*, 1708.08754.

[20] E. Marchi, F. Vesperini, S. Squartini and B. Schuller, *Deep recurrent neural Network-Based autoencoders for acoustic novelty detection*, *Comput Intell Neurosci* **2017** (1155/2017/4694860) .

[21] J. Preskill, *Quantum Computing in the NISQ era and beyond*, *Quantum* **2** (2018) 79 [1801.00862].

[22] R. P. Feynman, *Simulating physics with computers*, *International Journal of Theoretical Physics* **21** (1982) 467.

[23] I. M. Georgescu, S. Ashhab and F. Nori, *Quantum Simulation*, *Rev. Mod. Phys.* **86** (2014) 153 [1308.6253].

[24] S. Ramírez-Uribe, A. E. Rentería-Olivo, G. Rodrigo, G. F. R. Sborlini and L. Vale Silva, *Quantum algorithm for Feynman loop integrals*, 2105.08703.

[25] S. Williams, S. Malik, M. Spannowsky and K. Bepari, *A quantum walk approach to simulating parton showers*, 2109.13975.

[26] T. Li, X. Guo, W. K. Lai, X. Liu, E. Wang, H. Xing et al., *Partonic Structure by Quantum Computing*, 2106.03865.

[27] K. Bepari, S. Malik, M. Spannowsky and S. Williams, *Towards a quantum computing algorithm for helicity amplitudes and parton showers*, *Phys. Rev. D* **103** (2021) 076020 [2010.00046].

[28] S. P. Jordan, K. S. M. Lee and J. Preskill, *Quantum Algorithms for Fermionic Quantum Field Theories*, 1404.7115.

[29] J. Preskill, *Simulating quantum field theory with a quantum computer*, *PoS* **LATTICE2018** (2018) 024 [1811.10085].

[30] C. W. Bauer, W. A. de Jong, B. Nachman and D. Provasoli, *Quantum Algorithm for High Energy Physics Simulations*, *Phys. Rev. Lett.* **126** (2021) 062001 [1904.03196].

[31] S. Abel, N. Chancellor and M. Spannowsky, *Quantum computing for quantum tunneling*, *Phys. Rev. D* **103** (2021) 016008 [2003.07374].

[32] S. Abel and M. Spannowsky, *Observing the fate of the false vacuum with a quantum laboratory*, *P. R. X. Quantum.* **2** (2021) 010349 [2006.06003].

[33] Z. Davoudi, N. M. Linke and G. Pagano, *Toward simulating quantum field theories with controlled phonon-ion dynamics: A hybrid analog-digital approach*, *Phys. Rev. Res.* **3** (2021) 043072 [2104.09346].

[34] A. Mott, J. Job, J.-R. Vlimant, D. Lidar and M. Spiropulu, *Solving a higgs optimization problem with quantum annealing for machine learning*, *Nature* **550** (2017) 375.

[35] A. Blance and M. Spannowsky, *Unsupervised Event Classification with Graphs on Classical and Photonic Quantum Computers*, *JHEP* **21** (2020) 170 [2103.03897].

[36] S. L. Wu et al., *Application of quantum machine learning using the quantum variational classifier method to high energy physics analysis at the LHC on IBM quantum computer simulator and hardware with 10 qubits*, *J. Phys. G* **48** (2021) 125003 [2012.11560].

[37] A. Blance and M. Spannowsky, *Quantum machine learning for particle physics using a variational quantum classifier*, *JHEP* **2021** (2021) 212 [2010.07335].

[38] S. Abel, A. Blance and M. Spannowsky, *Quantum Optimisation of Complex Systems with a Quantum Annealer*, 2105.13945.

[39] S. L. Wu et al., *Application of quantum machine learning using the quantum kernel algorithm on high energy physics analysis at the LHC*, *Phys. Rev. Res.* **3** (2021) 033221 [2104.05059].

[40] S. Y.-C. Chen, T.-C. Wei, C. Zhang, H. Yu and S. Yoo, *Hybrid Quantum-Classical Graph Convolutional Network*, 2101.06189.

[41] K. Terashi, M. Kaneda, T. Kishimoto, M. Saito, R. Sawada and J. Tanaka, *Event Classification with Quantum Machine Learning in High-Energy Physics*, *Comput. Softw. Big Sci.* **5** (2021) 2 [2002.09935].

[42] I. H. Sarker, *Machine learning: Algorithms, real-world applications and research directions*, *SN Computer Science* **2** (2021) 160.

[43] P. Rebentrost, M. Schuld, L. Wossnig, F. Petruccione and S. Lloyd, *Quantum gradient descent and newton's method for constrained polynomial optimization*, 1612.01789.

[44] B. M. Dillon, T. Plehn, C. Sauer and P. Sorrenson, *Better Latent Spaces for Better Autoencoders*, *SciPost Phys.* **11** (2021) 061 [2104.08291].

[45] B. Bortolato, B. M. Dillon, J. F. Kamenik and A. Smolkovič, *Bump Hunting in Latent Space*, 2103.06595.

[46] B. M. Dillon, D. A. Faroughy and J. F. Kamenik, *Uncovering latent jet substructure*, *Phys. Rev. D* **100** (2019) 056002 [1904.04200].

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

[48] R. LaRose and B. Coyle, *Robust data encodings for quantum classifiers*, *Physical Review A* **102** (2020) .

[49] G. Fubini, *Sulle metriche definite da una forme hermitiana*, *Atti del Reale Istituto Veneto di Scienze, Lettere ed Arti* **63** (1904) 502–513.

[50] E. Study, *Kürzeste wege im komplexen gebiet*, *Mathematische Annalen* **60** (1905) 321.

[51] J. Romero, J. P. Olson and A. Aspuru-Guzik, *Quantum autoencoders for efficient compression of quantum data*, *Quantum Science and Technology* **2** (2017) 045001 [1612.02806].

[52] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer et al., *The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations*, *JHEP* **07** (2014) 079 [1405.0301].

[53] T. Sjöstrand, S. Ask, J. R. Christiansen, R. Corke, N. Desai, P. Ilten et al., *An introduction to PYTHIA 8.2*, *Comput. Phys. Commun.* **191** (2015) 159 [1410.3012].

[54] DELPHES 3 collaboration, *DELPHES 3, A modular framework for fast simulation of a generic collider experiment*, *JHEP* **02** (2014) 057 [1307.6346].

[55] M. Cacciari, G. P. Salam and G. Soyez, *FastJet User Manual*, *Eur. Phys. J. C* **72** (2012) 1896 [1111.6097].

[56] M. Cacciari, G. P. Salam and G. Soyez, *The anti-k_t jet clustering algorithm*, *JHEP* **04** (2008) 063 [0802.1189].

[57] V. Bergholm, J. Izaac, M. Schuld, C. Gogolin, M. S. Alam, S. Ahmed et al., *Pennylane: Automatic differentiation of hybrid quantum-classical computations*, 1811.04968.

[58] H. Buhrman, R. Cleve, J. Watrous and R. de Wolf, *Quantum fingerprinting*, *Phys. Rev. Lett.* **87** (2001) 167902.

[59] M. S. ANIS, H. Abraham, AduOffei, R. Agarwal, G. Agliardi, M. Aharoni et al., *Qiskit: An open-source framework for quantum computing*, 2021. 10.5281/zenodo.2573505.

[60] F. Chollet et al., “Keras.” https://keras.io, 2015.

[61] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro et al., *TensorFlow: Large-scale machine learning on heterogeneous systems*, 1603.04467.

[62] D. P. Kingma and J. Ba, *Adam: A method for stochastic optimization*, 1412.6980.

[63] M. Czakon, P. Fiedler and A. Mitov, *Total Top-Quark Pair-Production Cross Section at Hadron Colliders Through $O(\alpha_S^4)$*, *Phys. Rev. Lett.* **110** (2013) 252004 [1303.6254].