PHSX801 Midterm Review
Resolving Collider Two-fold Ambiguity Using Data Reuploading

Vivan Bhatt*

Department of Physics and Astronomy, University of Kansas,
Lawrence, KS 66045, USA

vivanbhatt@ku.edu

Contents

1 Introduction

The Large Hadron Collider (LHC) at CERN in Geneva generates 1 petabyte (10^{15} bytes) of collision data per second during operation. By the year 2030, the High Luminosity upgrade (HL-LHC) is projected to bring about a seven-fold increase in this collision data, out of which 99.999% of the data is discarded by real-time trigger systems requiring exabyte computing. Machine learning has long been present in the domain of particle physics, as neural networks operating across the Worldwide LHC Computing Grid function to produce a throughput equivalent to 10 million

*Work done with Gaurab Sedhain

Bitcoin transactions per second. The challenges faced in an experiment such as the LHC are not restricted to the realms of particle physics or engineering - it is a computational moonshot. Thus, it becomes pertinent to explore the applicability of quantum algorithms in the context of high-energy physics.

1.1 Combinatorial Problem at Colliders

This paper attempts to use quantum algorithms to resolve combinatorial problems that are ubiquitous in hadron colliders, especially in the context of searches for new physics, with missing momentum arising from Beyond Standard Model (BSM) particles. In particular, we look at the two-fold ambiguity in the dilepton channel for top quark pair production ($t\bar{t} \rightarrow b\bar{b} + l^+l^- + \not{E}_t$). Before any further analysis of the data, it is pertinent to correctly identify the pairing of the bottom (b) quarks and leptons, which is currently carried out using kinematical constraints on the invariant mass m_{bl} of the b-quark and the lepton. Previous studies have shown the utility and scalability of classical machine learning methods, including Deep Neural Networks (DNNs) and Lorentz Boost Networks (LBNs)[1], as well as hybrid quantum-classical approaches[2]. Here, we explore the possibility of using data reuploading [3] for creating a single qubit binary classifier.

1.2 Data Reuploading

Any variational quantum algorithm involves a parametrized quantum circuit upon which a measurement is made and then optimized so as to best suit the problem at hand. Traditional wisdom utilizes multiple qubits to encode the given data, and is thus impractical for larger datasets due to current constraints on quantum hardware. Data reuploading, on the other hand, can work with just a single qubit, wherein the input data is encoded as a set of parametrized phase rotation gates acting on the solitary qubit. Measurements of an appropriate Hermitian operator and subsequent optimization help us fix the parameters that act as weights for the input data in the rotation operators.

2 Ansatz

For the first proof of concept, we only use two input features - both the invariant masses. This will later be scaled to include other features including low-level variables such as the particle four-momenta or high-level variables such as the pseudorapidity and other kinematical information.

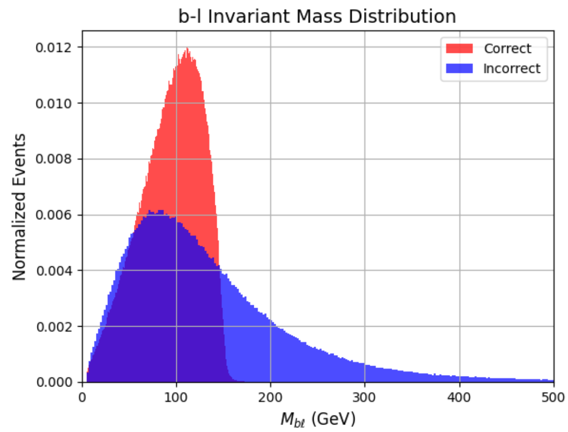
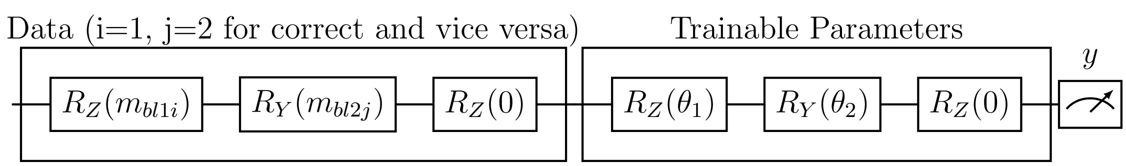


Fig. 1: The distribution of invariant masses of the $b-l$ system for both combinations

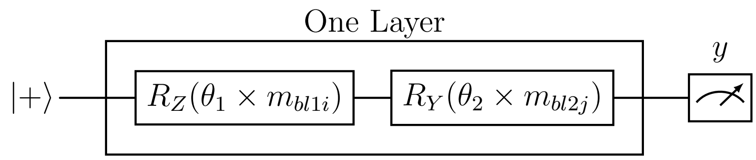
It would be most efficient to have our total number of features in multiples of three so as to fully utilize all parameters in our quantum circuit and avoid padding our rotation gates with zeroes since we use *qml.ROT*, which takes three Euler angles as inputs for the rotation

$$R(\phi,\theta,\omega) = RZ(\omega)RY(\theta)RZ(\phi) = \begin{bmatrix} e^{-i(\phi+\omega)/2}\cos(\theta/2) & -e^{i(\phi-\omega)/2}\sin(\theta/2) \\ e^{-i(\phi-\omega)/2}\sin(\theta/2) & e^{i(\phi+\omega)/2}\cos(\theta/2) \end{bmatrix}. \quad (1)$$

Thus, with our data being embedded into the parametrized rotation gates, we can add additional trainable parameters - that will later be optimized - to our required quantum circuit (rendered using Quantikz[4]).



This circuit is equivalent to one with rotations weighted by the parameters θ_i. Since we are only using two input features here, our circuit can be simplified. This constitutes one “layer” of our circuit. For better results, we increase the number of layers in the circuit, thus increasing its depth and improving our training. Additionally, we initialize with the qubit being in the $|+\rangle$ state to avoid any biases.



Initializing our qubit in the state $|0\rangle$ and performing phase rotations according to the given data, we measure the expectation value of the Hermitian observable "y", which is the output state density matrix. We know that the density matrix corresponding to a particular pure state $|\psi\rangle$ is given by

$$\rho = |\psi\rangle \langle\psi| \,. \tag{2}$$

Defining the labels for the correct and incorrect observables as the usual computational basis kets $|0\rangle$ and $|1\rangle$ respectively, we thus get our operator Y as

$$Y_{correct} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \text{ and } Y_{incorrect} = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}. \tag{3}$$

Thus, the expectation value of this operator Y allows us to define the cost function as the difference between the expectation value of the non-linear projection operator ρ and the ideal expectation value for either of the two combinations - one. The average loss thus gives us

$$\text{Cost} = \frac{\sum (1 - \langle\rho\rangle)^2}{N_{data}}. \tag{4}$$

This allows us to compute the fidelity (or closeness) of the measured output state to the known labels corresponding to measurements of either $|0\rangle$ and $|1\rangle$. Here, this expectation value of our observable Y includes the density matrix that supplies the projection operator, serving the function of the inner product between the output state and the known class as seen in the fidelity cost function defined in [3] as

$$\chi_f^2(\vec{\theta}, \vec{w}) = \sum_{\mu=1}^{M} \left(1 - \left|\langle\tilde{\psi}_s|\psi(\vec{\theta}, \vec{w}, \vec{x}_\mu)\rangle\right|^2\right), \tag{5}$$

where the data point μ has the correct label corresponding to the state $\left|\tilde{\psi}_s\right\rangle$. Repeated measurements enable us to compute the probabilities of our final output state being “closer” to either of the poles on the Bloch sphere. We essentially want to obtain a set of parameters that act as weights for any set of input features and take the correct combinations to the “northern hemisphere”, and incorrect ones to the “southern hemisphere” on the Bloch sphere.

Having obtained the cost for a randomized set of parameters, we then classically optimize our parameter set so as to minimize the cost function. For this purpose, we currently use Adam Optimizer[5] that is built into PennyLane, but can further explore other in-built options such as Gradient Descent, optimize our parameters using a neural network as done in Ref.[6], or even utilize phase kick-back as shown in Ref.[7] where a cost based on measurements of an ancilla qubit is optimized. Thus, we obtain the best possible parameters that aid us in maximizing the overlap between the final output state and the desired label ket, allowing us to identify the invariant masses that characterized the set of rotations as arising from a correct or incorrect combination of the b-quark and the lepton.

3 Results

We can see a crude geometrical visualization of our output after optimizing the 30 parameters in the quantum circuit.

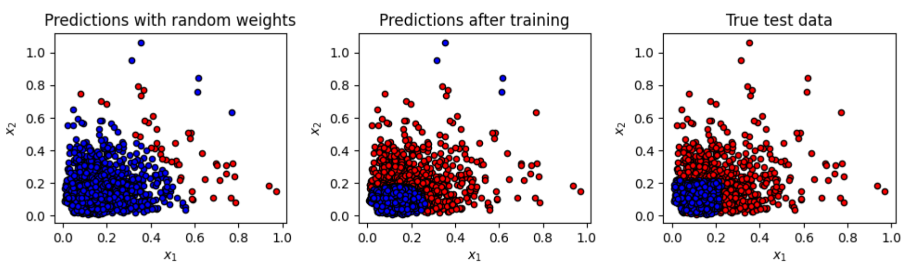
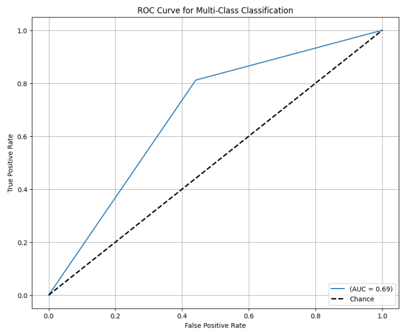
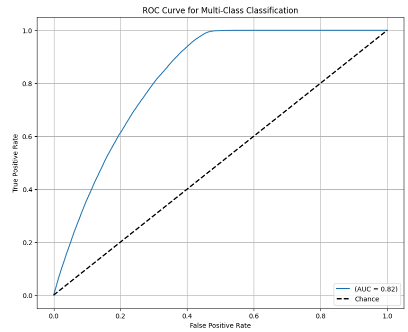


Fig. 2: A geometric visualization of our dataset with the axes representing the two invariant masses.

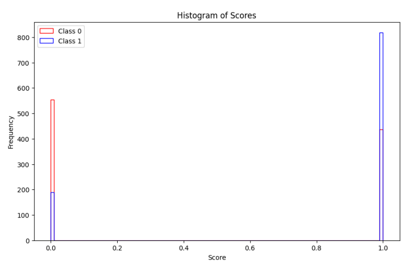
To better understand the results, we compare the performance of our single-qubit classifier with that of a DNN with 17, 665 trainable parameters. At this point, the DNN clearly performs better than our quantum algorithm (although they both have some interesting features that need further consideration), but we will later appropriately penalize the neural network to provide a fair comparison between the two classifiers, which currently have vastly different numbers of trainable parameters.

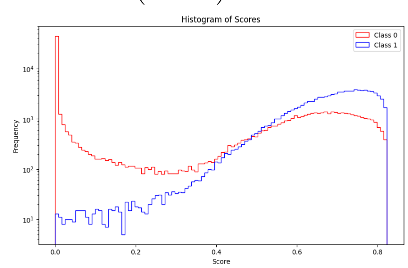




(a) (b)

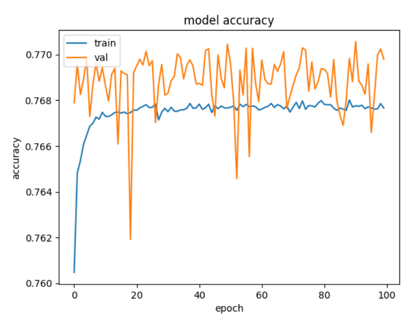
Receiver Operator Characteristic (ROC) Curves



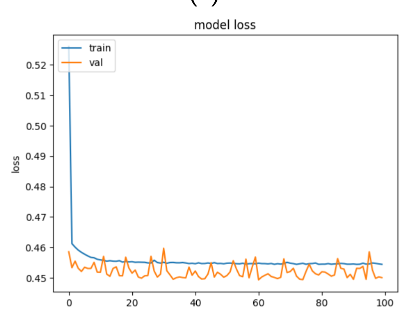


(c) (d)

Score Distributions



(e) (f)



(g)

Accuracy and Loss Curves

Fig. 3: Comparing the performance of data reuploading (left) with a DNN (right).

4 Future Work

So far, we have only used two features of the available input data - the invariant masses of both b-quark & lepton subsystems for both correct and incorrect pairings. The next goal of this study would be to modify our ansatz and use additional input features. Scaling this method while restricting our use to a single qubit allows us to compare the performance for different input features, including both low-level and high-level kinematic variables. This would allow us to explore the kind of information the network "learns" and thus compare it with the expressibility of classical neural networks [8]. We will also compare the performance of the system on increasing the depth (number of layers) and the width (number of qubits) of the circuit.

References

[1] Haider Alhazmi, Zhongtian Dong, Li Huang, Jeong Han Kim, Kyoungchul Kong, and David Shih. Resolving combinatorial ambiguities in dilepton $t\bar{t}$ event topologies with neural networks. *Phys. Rev. D*, 105(11):115011, 2022.

[2] Jacob L. Scott, Zhongtian Dong, Taejoon Kim, Kyoungchul Kong, and Myeonghun Park. Hybrid quantum-classical approach for combinatorial problems at hadron colliders. 10 2024.

[3] Adrián Pérez-Salinas, Alba Cervera-Lierta, Elies Gil-Fuster, and José I Latorre. Data re-uploading for a universal quantum classifier. *Quantum*, 4:226, 2020.

[4] Alastair Kay. Tutorial on the quantikz package, 2023.

[5] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[6] Eraraya Ricardo Muten, Togan Tlimakhov Yusuf, and Andrei Voicu Tomut. Modified layerwise learning for data re-uploading classifier in high-energy physics event classification. In *2021 IEEE International Conference on Quantum Computing and Engineering (QCE)*, pages 82–88, 2021.

[7] Yujin Kim and Daniel K Park. Expressivity of deterministic quantum computation with one qubit. *Physical Review A*, 111(2):022429, 2025.

[8] Zhongtian Dong, Kyoungchul Kong, Konstantin T Matchev, and Katia Matcheva. Is the machine smarter than the theorist: Deriving formulas for particle kinematics with symbolic regression. *Physical Review D*, 107(5):055018, 2023.