The prospect and applications of quantum machine learning algorithms in High Energy Physics

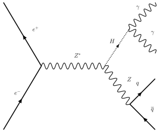
Abdualazem Fadol Mohammed
On behalf of the Quantum Computing Team

February 15, 2023

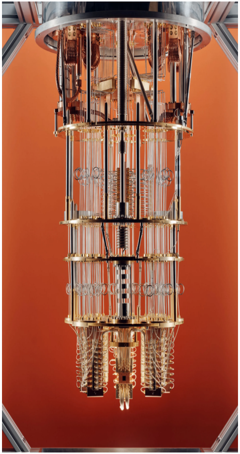
Overview

Introduction

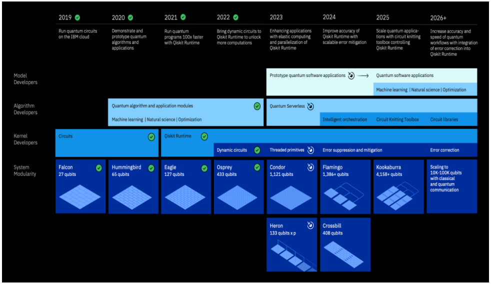
- Machine learning has blossomed in the last decades and becomes essential in many fields.
- It played a significant role in solving High Energy physics problems, such as reconstruction, particle identification.
- and handling high dimensional and complex problems using deep learning.
- Quantum computing is a new idea for our workstations to process data faster than currently achievable.
- Machine learning & quantum computing may:
  - locating more computationally complex feature spaces
  - better data classification
  - smarter algorithms that can give us accurate prediction.
- Companies such as Google, IBM and Origin are committed to accelerating the development of quantum technology.
- Objectives:
  - Apply quantum machine learning to high energy physics
  - Support-vector machine algorithm in quantum computers
  - Building the quantum algorithm using IBM quantum simulator
  - Comparing the performance in different real quantum computers



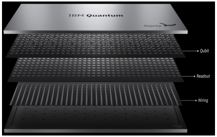
IBM quantum computer



Credited to Thomas Prior for TIME



- ☐ IBM has ambitious pursuits:
  - 433-qubit IBM Quantum Osprey
  - three times larger than the Eagle processor
  - going up to 10k-100k qubits
- ☐ Taking quantum computing out of the lab:
  - NY provides over 20 quantum computing
  - Scales the processors with high availability



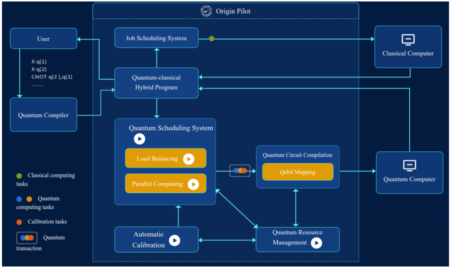
IBM Quantum Osprey

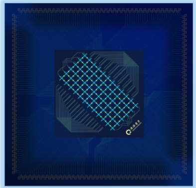
NY Quantum Computing Data Center

- ☐ IBM provides up to 7 qubits for free with an opportunity to apply for a researcher account with more qubits.

Origin quantum computer (Wuyuan)

- ☐ Origin Quantum 64-qubit superconductor QPU
  - single-qubit gate fidelity $> 99.9\%$
  - double-qubit gate fidelities $> 98\%$
  - readout fidelity $> 96\%$
- ☐ A quantum computing control system dedicated to superconducting quantum chips



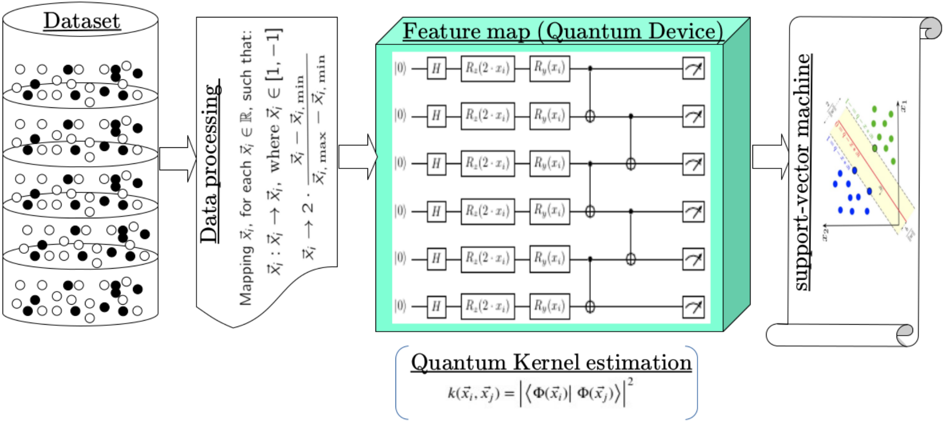


KF-C64-200

TJ-SQMC-300

☐ This is the first Quantum Computer Operating System in China. One could use up to 6 qubits for free.

Data encoding and processing

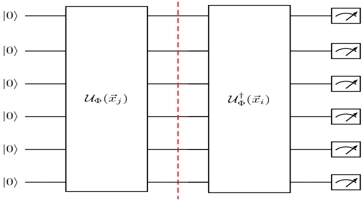


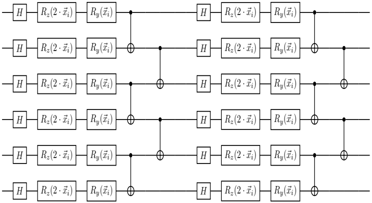
- ☐ Encoding the $e^+e^- \to ZH \to q\bar{q}\gamma\gamma$ and $e^+e^- \to (Z\gamma^*)\gamma\gamma$ datasets to high dimensional quantum dataset.
- ☐ Seven variables are passed through preliminary mapping and then passed to a quantum circuit for evaluation.
- ☐ The Quantum support-vector machines kernel (QSVM-Kernel) is evaluated for each data point and the rest.

Feature map and quantum kernel estimation

- Quantum feature map dictate the QSVM-Kernel:
  - two identical layers
  - single-qubit rotation gates
  - two-qubit CNOT entangling gates

Rotation	Depth	Events	Best AUC	Variation
$R_z(2 \cdot \vec{x}_i) + R_y(\vec{x}_i)$	2	5000	0.935	0.009
$R_z(\vec{x}_i) + R_y(\vec{x}_i)$			0.933	0.015
$R_y(\vec{x}_i) + R_x(\vec{x}_i)$			0.932	0.015
$R_z(\vec{x}_i) + R_z(\vec{x}_i)$			0.932	0.014
$R_y(\vec{x}_i)$			0.928	0.008
$R_z(\vec{x}_i)$			0.928	0.008



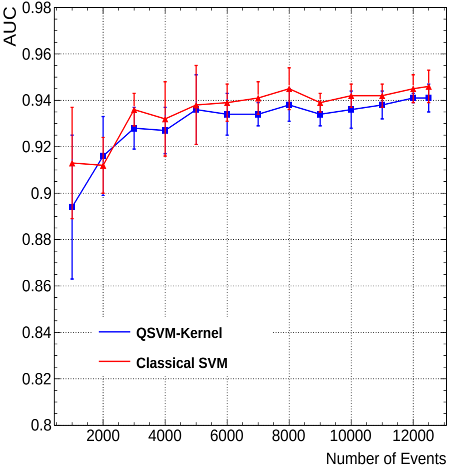


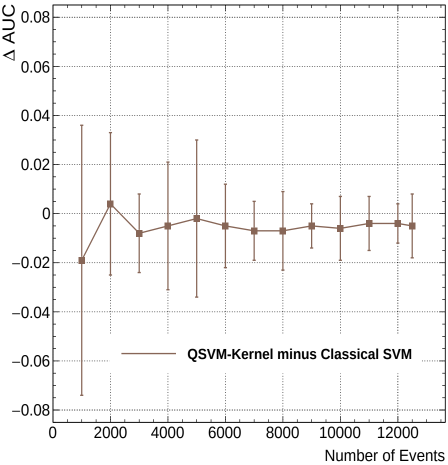
- QSVM-Kernel estimation:

$$k(\vec{x}_i, \vec{x}_j) = \left| \left\langle 0^{\otimes N} \right| \mathcal{U}^\dagger_{\Phi(\vec{x}_i)} \mathcal{U}_{\Phi(\vec{x}_j)} \left| 0^{\otimes N} \right\rangle \right|^2$$

  - Using 6 variables mapped to 6-qubit
  - the expectation of each data point

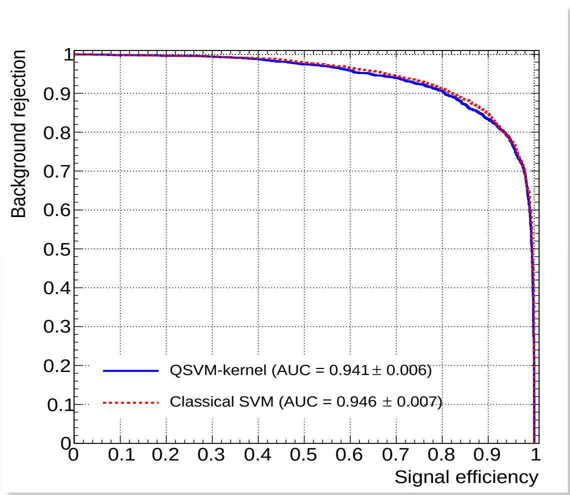
AUCs as function of the event





- ☐ The QSVM-Kernel and classical SVM classifiers with different dataset size from 1000 to 12500 events.
- ☐ The quoted errors are the standard deviations for AUCs calculated from several shuffles of the dataset.

Performance of the quantum simulator



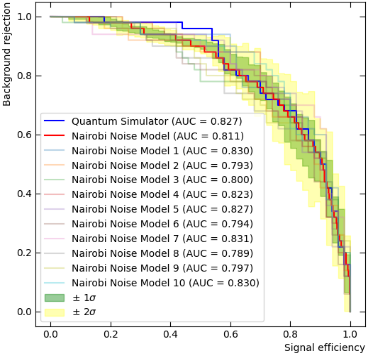
☐ The performance of the QSVM-Kernel and the SVM classification using StatevectorSimulator from IBM.

Nairobi Noise Model

Noise in quantum computers

Quantum computers are susceptible to all sort of noise sources: electromagnetic signal coming from a WiFi or a disturbance in the earth magnetic field etc. All these are considered as noise and can lead to error in the calculation.

- ☐ The device noise model used automatically generate a simplified noise model for a real device.
- ☐ The noise model takes into account the following:
  - the gate error probability of each basis gate
  - the gate length of each basis gate
  - T_1 and T_2 relaxation time constant
  - the readout error probability
- ☐ The standard deviation of results generated using different seeds is taken as statistical fluctuations.
- ☐ The estimated noise in IBM Nairobi computer is 0.017.

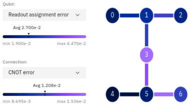


The performance with actual quantum computers

The receiver operating characteristic curve

- ☐ The ROC curves of the QSVM-Kernel classifiers from the IBM Nairobi quantum computer,
- ☐ the Origin quantum computer (Wuyuan) and the state-vector quantum simulators from IBM.

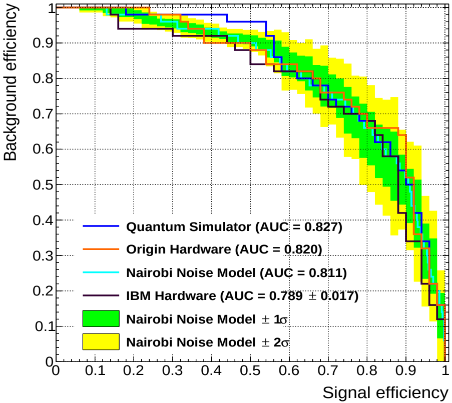
- ☐ Signal and Standard Model backgrounds:
  - $e^+e^- \to ZH \to q\bar{q}\gamma\gamma$
  - $e^+e^- \to (Z\gamma^*)\gamma\gamma$
- ☐ with 100 events for both signal and backgrounds
- ☐ 6-qubit superconducting quantum chip systems



IBM Nairobi quantum

Origin Wuyuan quantum

- ☐ Estimated uncertainties of:
  - Noise: ± 0.017
  - Statistical fluctuation: ± 0.022



Quantum Machine Learning Tutorial

Quantum Machine Learning documentation

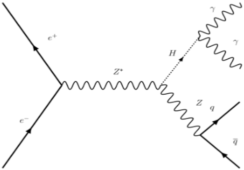
Search the docs ...

Theme by the Executable Book Project

Contents

Tutorial for the application of quantum machine learning in HEP

This tutorial is a quick example of running support-vector machines in classical computers and a simulated quantum computer using a state vector simulator from IBM. Data samples from the Central Electron Positron Collider (CEPC) are used to demonstrate the performance of the support-vector machine.



The $e^+e^- \to ZH$ signal and its related backgrounds are utilised for the study where the $H \to \gamma\gamma$ and $Z \to q\bar{q}$. They are plenty of events for the signal and backgrounds, but for this tutorial, you can use up o 2k events.

By the end of the tutorial you will learn the following

- Preparing the dataset from root fills as *NumPy* arrays
- Understand how to constuct a quantum feature map
- Run a support-vector machines algorithm in classical computers (SVM)
- Run a quantum support-vector machines in simulated computers (QSVM)

This tutorial is based on the results shown in this paper here on **inspirehep**. Before you run the tutorial, you need to download the following packages by uncommenting the lines in the cell below.

- ☐ Inspire further development in this exciting field, and identify opportunities for future research.
- ☐ We are developing this example in GitHub to help you get started (Still work in progress!)

Conclusion

- We studied the $e^+e^- \to ZH \to q\bar{q}\gamma\gamma$ signal optimisation using quantum/classical ML algorithm.
- Support-vector machines were compared:
  - Quantum support-vector machines (QSVM-Kernel) with IBM quantum simulator
  - Classical support-vector machines (SVM)
- Each QSVM and SVM algorithm is optimised to its best before comparing them.
- Real quantum computing system with 100 events for signal and background:
  - Wuyuan vs IBM
  - IBM vs IBM simulator
- We obtained a similar classification performance to the classical SVM algorithm with different dataset size.
- We also studied the effect of the noise based on a simple noise Model on IBM Nairobi.
- And providing a quick tutorial as an example for quantum machine learning using jupyter-lab.
- This talk is based on 2209.12788 [hep-ex]— submitted to Physics Letters B journal.