Application of Quantum Machine Learning using the Quantum Kernel Algorithm on High Energy Physics Analysis at the LHC

Shaojun Sun

Jay Chan, Alkaid Cheng, Wen Guan, Shaojun Sun, Alex Wang, Sau Lan Wu, Rui Zhang, Chen Zhou
Physics Department, University of Wisconsin-Madison

Miron Livny
Computer Sciences Department, University of Wisconsin-Madison

Federico Carminati, Alberto Di Meglio
CERN Openlab, IT Department, CERN

Panagiotis Barkoutsos, Ivano Tavernelli, Stefan Woerner, Jennifer Glick
IBM Research Zurich and IBM T.J. Watson Research Center

Andy Li, Joseph Lykken, Panagiotis Spentzouris
Quantum Institute, Fermilab

Samuel Yen-Chi Chen, Shinjae Yoo
Computational Science Initiative, BNL

Tzu-Chieh Wei
C.N. Yang Institute for Theoretical Physics, State University of New York at Stony Brook

DPF Meeting, July 12-14 2021

Our program with Quantum Machine Learning

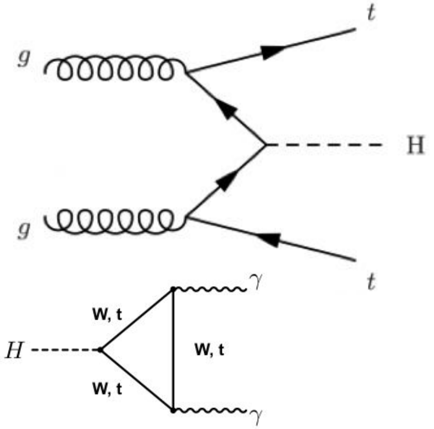
Our Goal:

to perform LHC High Energy Physics analysis with Quantum Machine Learning, to explore and to demonstrate that the potential of quantum computers can be a new computational paradigm for big data analysis in HEP, as a proof of principle.

Our present program is to employ Quantum Support Vector Machine (QSVM) Kernel method to LHC High Energy Physics analysis, for example ttH ($H \rightarrow \gamma\gamma$) analysis.

ttH (H → $\gamma\gamma$) analysis at the LHC

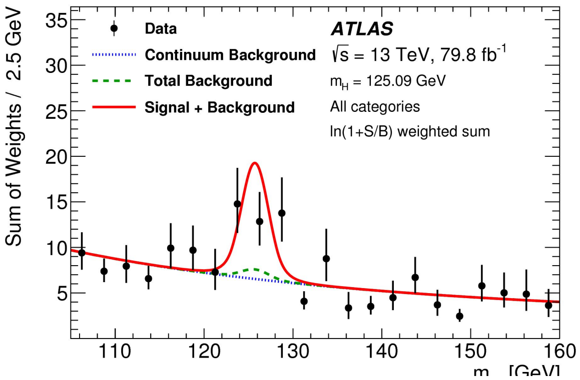
The observation of ttH production (Higgs boson production in association with a top quark pair) by ATLAS and CMS at the LHC directly confirmed the interaction between the Higgs boson and the top quark, which is the heaviest known fundamental particle



(Top)

(Higgs)

(Anti-top)



[Phys. Lett. B 784 (2018) 173] $M_{\gamma\gamma}$ [GeV]

- Using Boosted Decision Tree (BDT, a classical machine learning technique) with XGBoost package, the ATLAS Collaboration observes the ttH (H→γγ) process
- Our study performs the event classification of the ttH (H→γγ) analysis (hadronic channel) with delphes simulation samples [JHEP 02 057 (2014)] and QSVM kernel method.

QSVM Kernel method

QSVM Kernel method ***(introduced by IBM, published in Nature 567 (2019) 209):***

- *map classical data $\vec{x}$ to a quantum state $|\Phi(\vec{x})\rangle$ using a Quantum Feature Map circuit;*
- *calculate the similarity between any two data events ("kernel entry") as $K(\vec{x}_1,\vec{x}_2)=|\langle\Phi(\vec{x}_1)|\Phi(\vec{x}_2)\rangle|^2$ using a quantum computer;*
- *then using the kernel entries to find an optimal hyperplane that separates signal from background.*

map classical data

$\vec{x}_1 \rightarrow |\Phi(\vec{x}_1)\rangle$

$\vec{x}_2 \rightarrow |\Phi(\vec{x}_2)\rangle$

$\vec{x}_3 \rightarrow |\Phi(\vec{x}_3)\rangle$

...

calculate kernel entries

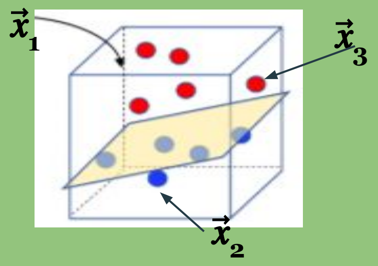
$K(\vec{x}_1,\vec{x}_2)=|\langle\Phi(\vec{x}_1)|\Phi(\vec{x}_2)\rangle|^2$

$K(\vec{x}_1,\vec{x}_3)=|\langle\Phi(\vec{x}_1)|\Phi(\vec{x}_3)\rangle|^2$

$K(\vec{x}_2,\vec{x}_3)=|\langle\Phi(\vec{x}_2)|\Phi(\vec{x}_3)\rangle|^2$

...

find separating hyperplane



Employing QSVM Kernel method for ttH ($H \rightarrow \gamma\gamma$) analysis

We are performing the ttH analysis using QSVM Kernel method with up to 20 qubits:

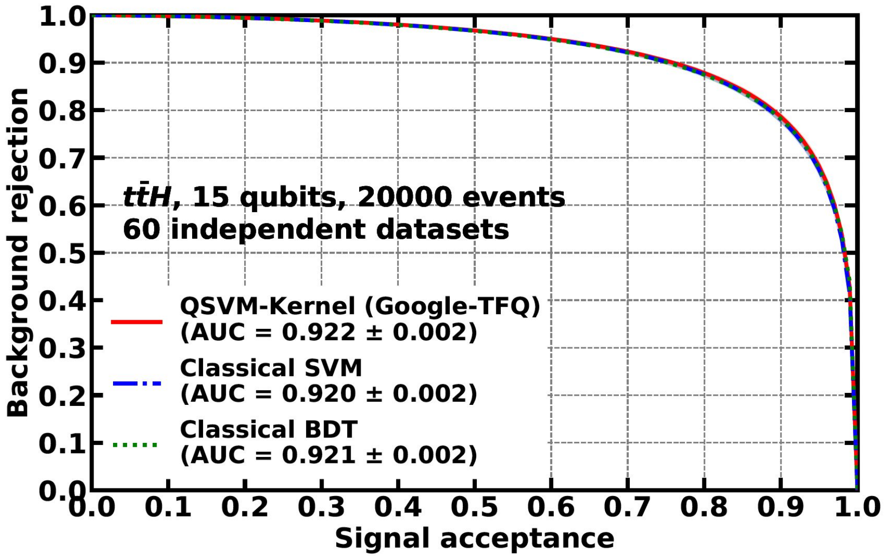
- *Principal Component Analysis (PCA) method is used to reduce the number of features to match the number of qubits.*
- *A customized FeatureMap is used. The quantum FeatureMap circuit encodes classical data to a quantum state.*
- *Grid-Search with cross-validation is used to optimize the performance of QSVM Kernel and the classical machine learning methods.*
- *Results submitted to arXiv:2104.05059*

Employing QSVM Kernel method with quantum simulators for ttH ($H \rightarrow \gamma\gamma$) analysis

Our group has implemented the QSVM Kernel algorithm using the qsim Simulator from the Google TensorFlow Quantum (TFQ) framework, the Statevector Simulator from the IBM Qiskit framework and the Local Simulator from the Amazon Braket framework.

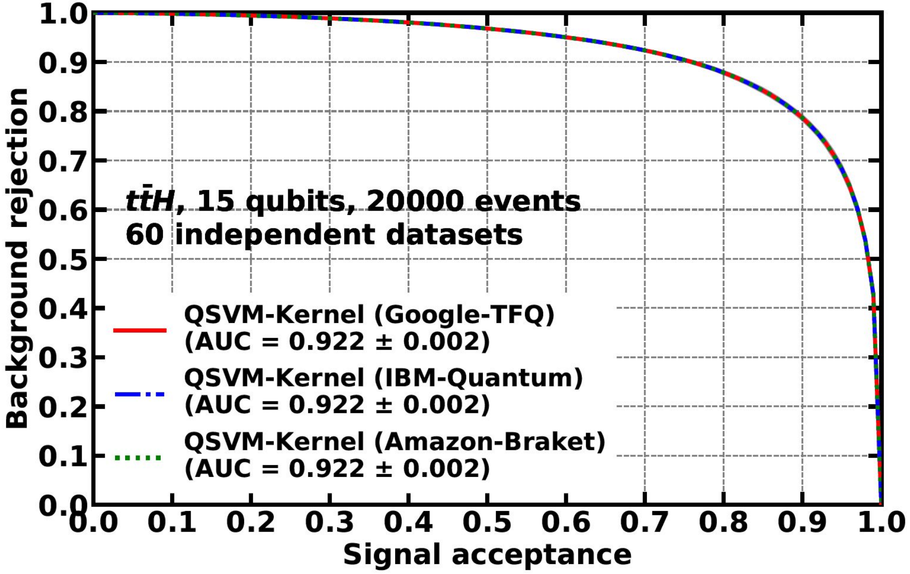
- ***These simulators represent the ideal quantum hardware that performs infinite measurement shots and experiences no hardware noise.***
- ***We have overcome the challenges of heavy computing resources in the use of up to 20 qubits and up to 50000 events on the quantum computer simulators.***

Employing QSVM Kernel method with quantum simulators for ttH ($H \rightarrow \gamma\gamma$) analysis



Using ttH analysis dataset (20000 events, 15 variables), QSVM Kernel on simulator (red) achieves similar performances with classical SVM (blue) and classical BDT (green). (Results are averaged over sixty datasets)

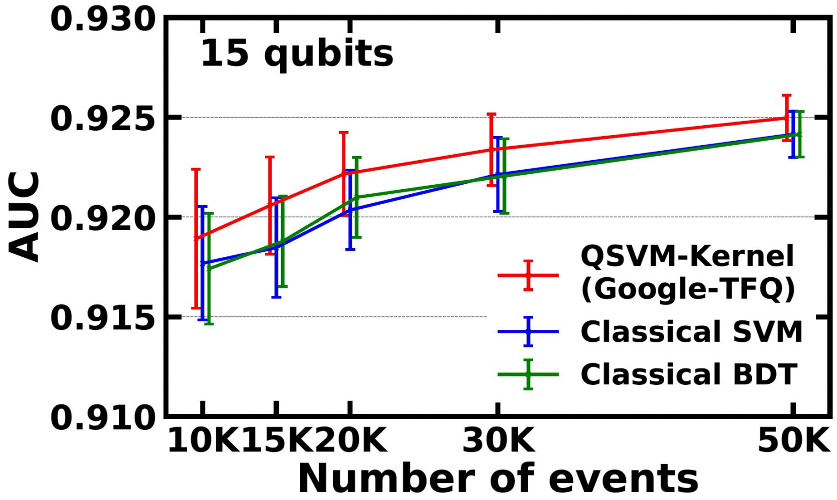
Employing QSVM Kernel method with quantum simulators for ttH ($H \rightarrow \gamma\gamma$) analysis



Using ttH analysis dataset (20000 events, 15 variables), Google qsim simulator (red), IBM statevector simulator (blue), and Amazon local simulator (green) provide identical performances for QSVM Kernel method.

Employing QSVM Kernel method with quantum simulators for ttH ($H \to \gamma\gamma$) analysis

AUC vs number of events

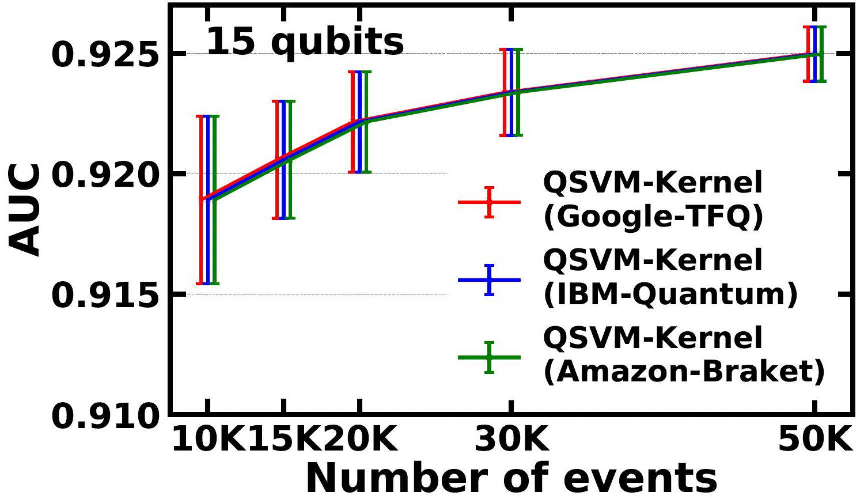


- Averaged over 60 independent datasets
- QSVM Kernel method and noiseless simulators enable us to work with a larger number of events.

Using ttH analysis dataset (10000-50000 events, 15 variables), QSVM Kernel on simulator (red) achieves similar performances with classical SVM (blue) and classical BDT (green).

Employing QSVM Kernel method with quantum simulators for ttH ($H \rightarrow \gamma\gamma$) analysis

AUC vs number of events

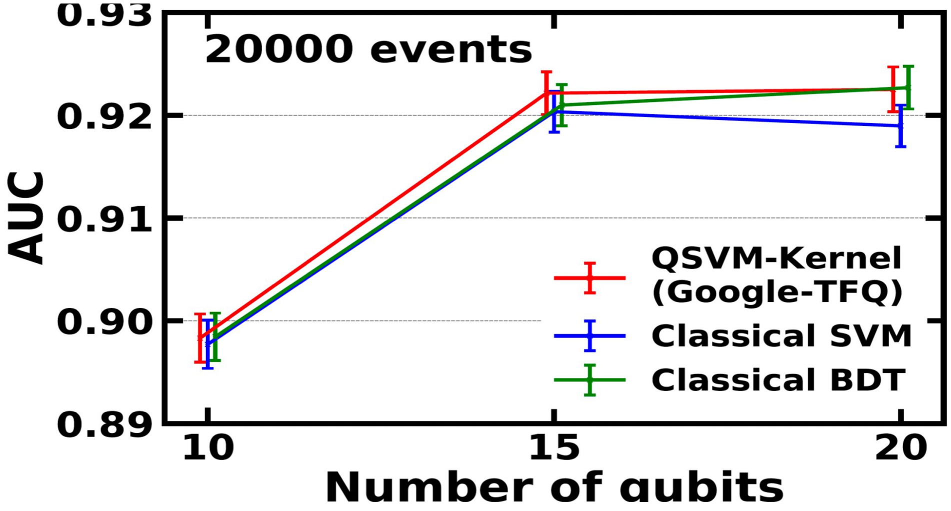


- Averaged over 60 independent datasets
- QSVM Kernel method and noiseless simulators enable us to work with a larger number of events.

Using ttH analysis dataset (10000-50000 events, 15 variables), Google qsim simulator (red), IBM statevector simulator (blue), and Amazon local simulator (green) provide identical performances for QSVM Kernel method.

Employing QSVM Kernel method with quantum simulators for ttH ($H \rightarrow \gamma\gamma$) analysis

AUC vs number of qubits

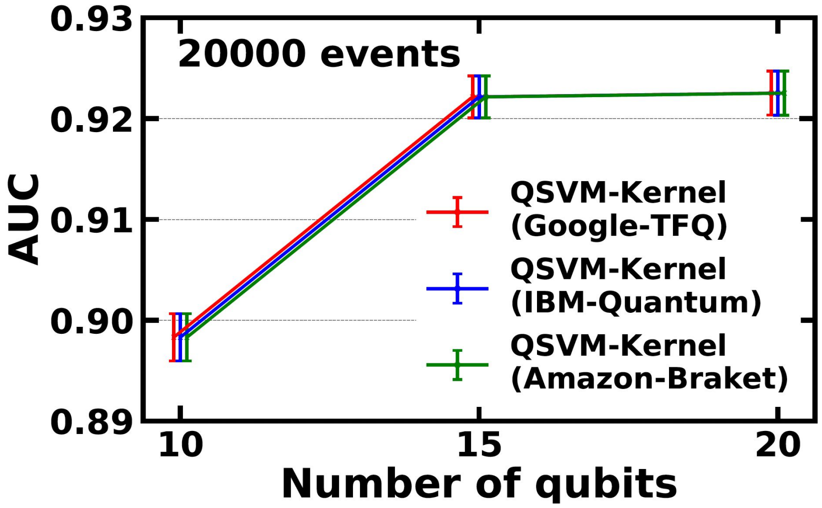


- Averaged over 60 independent datasets
- QSVM Kernel method and noiseless simulators also enable us to work with a larger number of qubits.

Using ttH analysis dataset (20000 events, 10-20 variables), QSVM Kernel on simulator (red) achieves similar performances with classical SVM (blue) and classical BDT (green).

Employing QSVM Kernel method with quantum simulators for ttH ($H \rightarrow \gamma\gamma$) analysis

AUC vs number of qubits



- Averaged over 60 independent datasets
- QSVM Kernel method and noiseless simulators also enable us to work with a larger number of qubits.

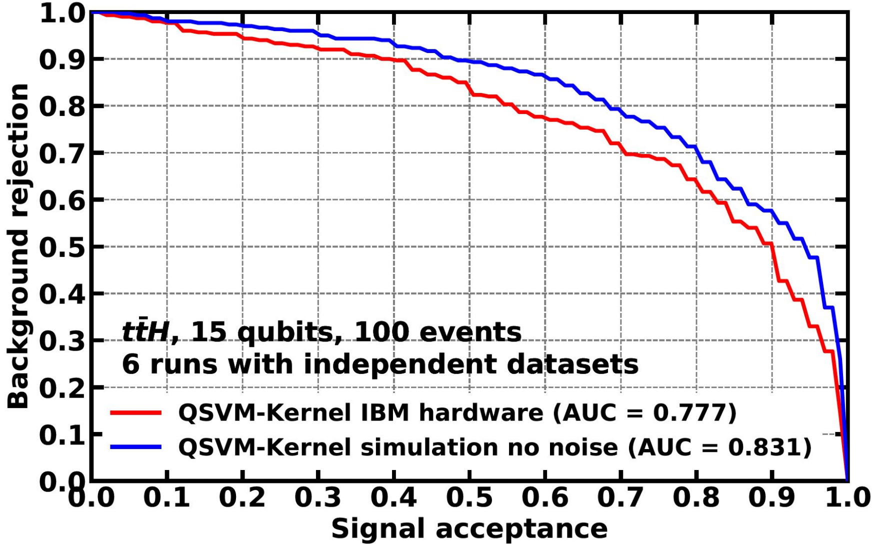
Using ttH analysis dataset (20000 events, 10-20 variables), Google qsim simulator (red), IBM statevector simulator (blue), and Amazon local simulator (green) provide identical performances for QSVM Kernel method.

Employing QSVM Kernel with IBM hardware (ibmq_paris, a 27-qubit machine) for ttH ($H \rightarrow \gamma\gamma$) analysis

We have also been running the QSVM Kernel algorithm on quantum computer hardware provided by IBM (based on superconducting circuits).

- ***to assess the QSVM Kernel performances on today's noisy quantum hardware device;***
- ***due to current limitation of access time on imbq_paris, we only process six datasets of 100 training events and 100 test events;***
- ***for the six datasets, the average running time is approximately 680 minutes.***

Employing QSVM Kernel with IBM hardware (ibmq_paris, a 27-qubit machine) for ttH ($H \rightarrow \gamma\gamma$) analysis



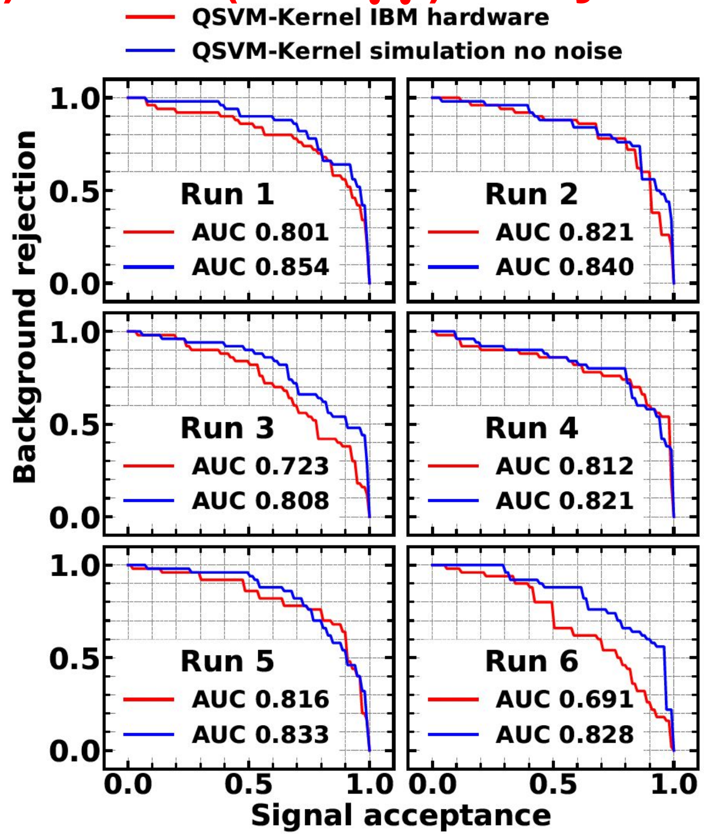
hardware AUC = 0.777

simulator AUC = 0.831

Using ttH analysis dataset (100 events, 15 variables), the QSVM Kernel results on the Quantum Hardware are promising and approaching the QSVM Kernel results on Quantum Simulator (the difference is likely due to effect of hardware noise).

Employing QSVM Kernel with IBM hardware (ibmq_paris, a 27-qubit machine) for ttH ($H \rightarrow \gamma\gamma$) analysis

- **ROC curves for each of the six runs:**
- **The effect of quantum hardware noise seems to fluctuate among the runs.**



Summary:

- *We employed QSVM Kernel method to ttH ($H \rightarrow \gamma\gamma$) analysis with Delphes simulation events.*
- *With up to 50000 events and up to 20 qubits, QSVM Kernel method on Google, IBM and Amazon Quantum Simulators performs similarly as classical SVM and classical BDT in the ttH ($H \rightarrow \gamma\gamma$) channel, and the three simulators provide identical performance for QSVM kernel method.*
- *With 100 events and 15 Qubits, for QSVM Kernel method, IBM Quantum Hardware and IBM Quantum Simulator show comparable performance.*
- *Our results (on both simulators and hardware) demonstrate quantum machine learning on the gate-model quantum computers has the ability to distinguish signal from background in realistic physics datasets.*