22 Jan. 2020
CERN Openlab Technical Workshop

Event Classification with Quantum Machine Learning

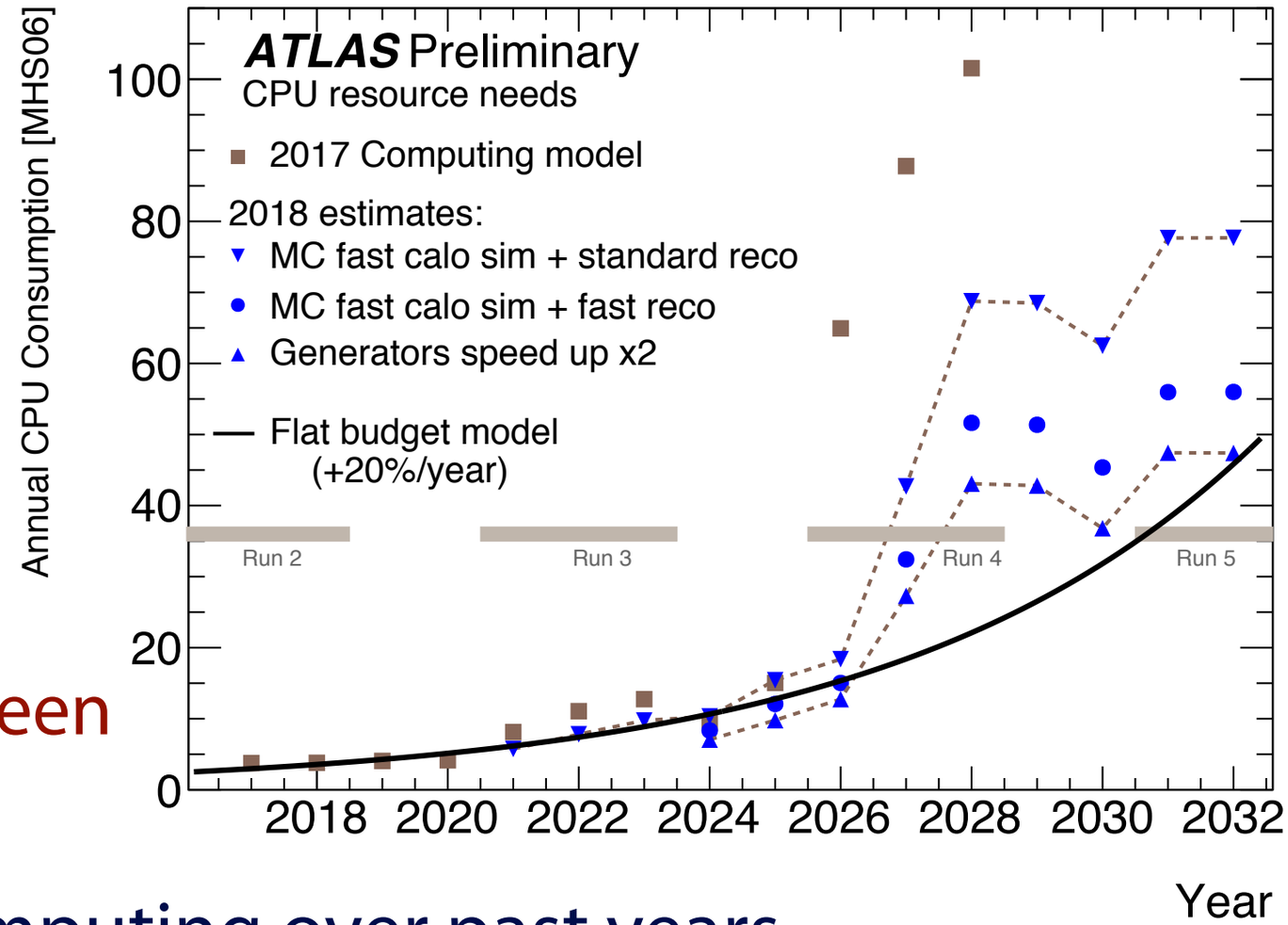
K. Terashi, M. Kaneda, T. Kishimoto,
M. Saito, R. Sawada, J. Tanaka
ICEPP, University of Tokyo

HEP and Quantum Computing

HL-LHC (2026~)

- High luminosity
- Upgraded detectors

➔ Huge increase of required computational resources foreseen



Rapid progress of quantum computing over past years

- Pre-error corrected quantum device with O(10-100)-qubits in hand

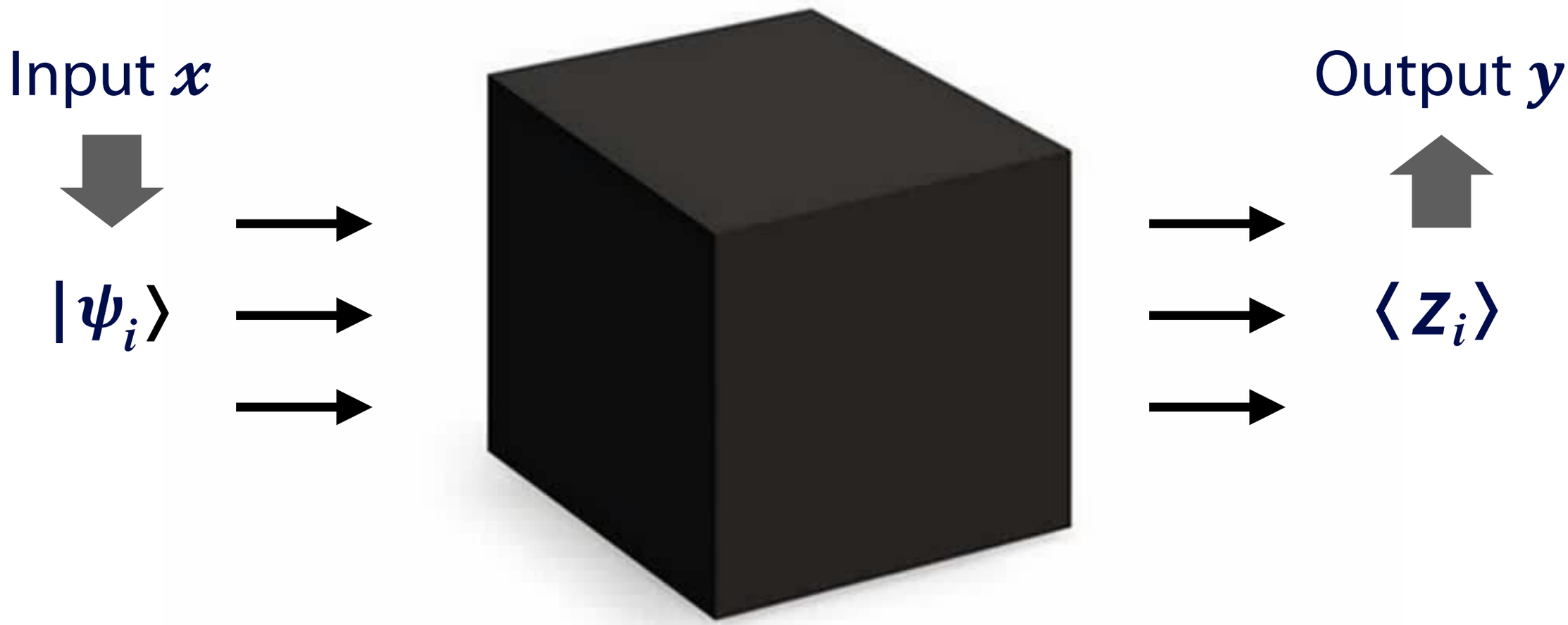
Attempt of QC application to HEP in progress at several fronts

- Machine learning, Tracking, Parton shower simulation, Jet algorithm, …

➡ Focus here on gate-based quantum machine learning, especially application to event classification in HEP data analysis

Quantum Machine Learning

Learn the function f that relates x to y



$$y = f(\,|\psi(x)\rangle\langle\psi(x)|\,)$$

Potential for enhancing ML by exploiting more representational power in quantum Hilbert space

Quantum Machine Learning

Focus on discrimination of physics events (e.g, signal vs background)

Study two implementations based on variational quantum algorithm:

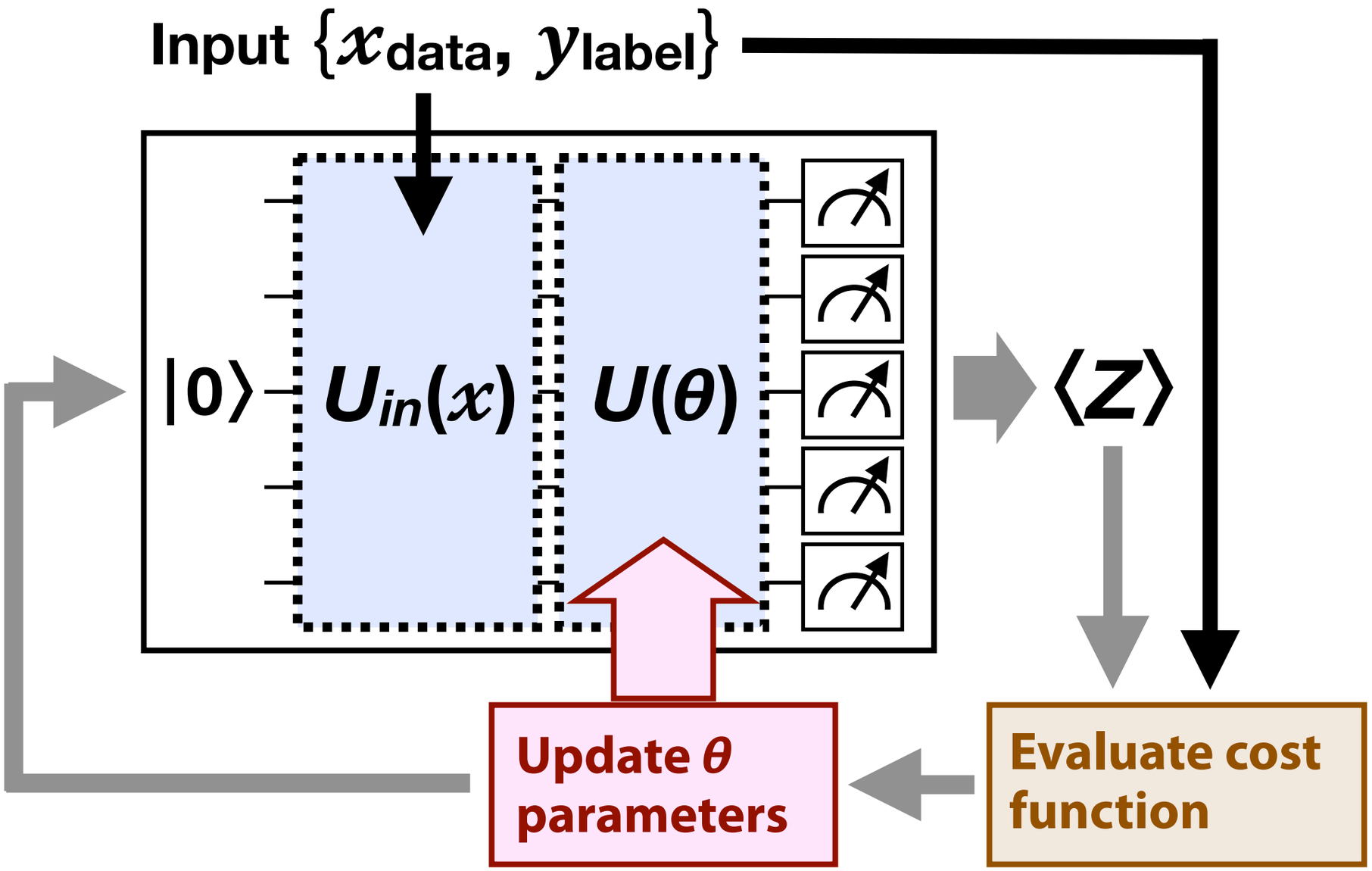
- **Quantum Circuit Learning (QCL)** K. Mitarai *et al.*, arXiv:1803.00745
- **Variational Quantum Classification (VQC)** V. Havlicek *et al.*, arXiv:1804.11326

Variational Quantum Algorithm

Focus on discrimination of physics events (e.g, signal vs background)

Study two implementations based on variational quantum algorithm:

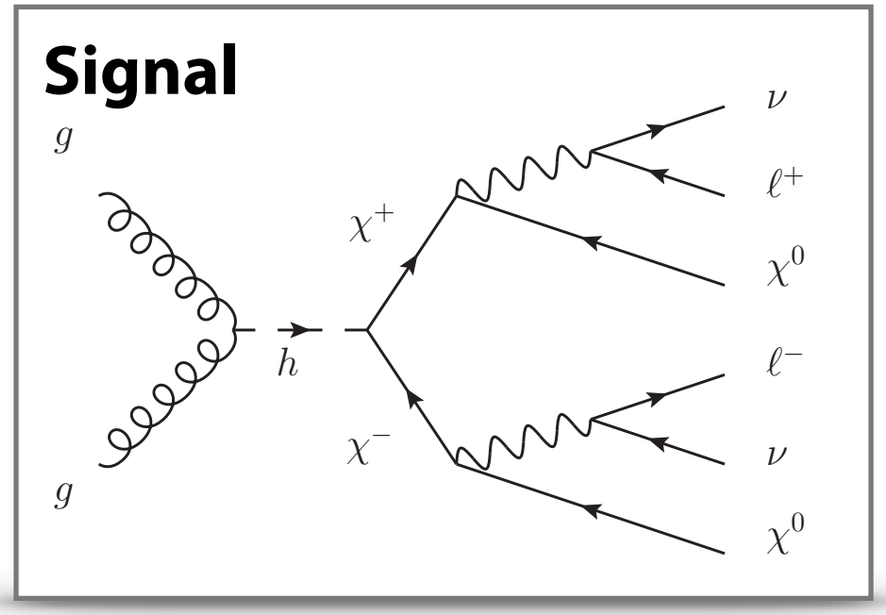
- **Quantum Circuit Learning (QCL)** K. Mitarai *et al.*, arXiv:1803.00745
- **Variational Quantum Classification (VQC)** V. Havlicek *et al.*, arXiv:1804.11326

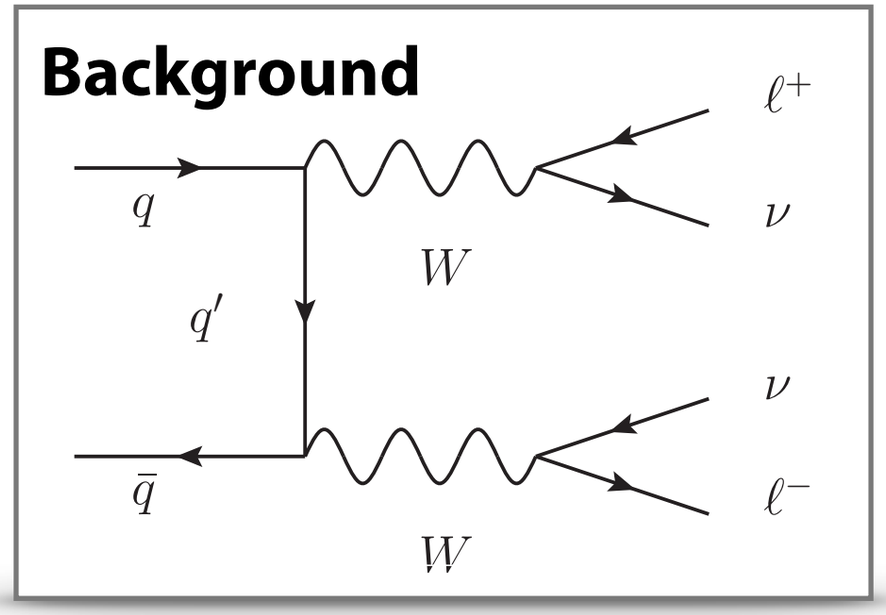


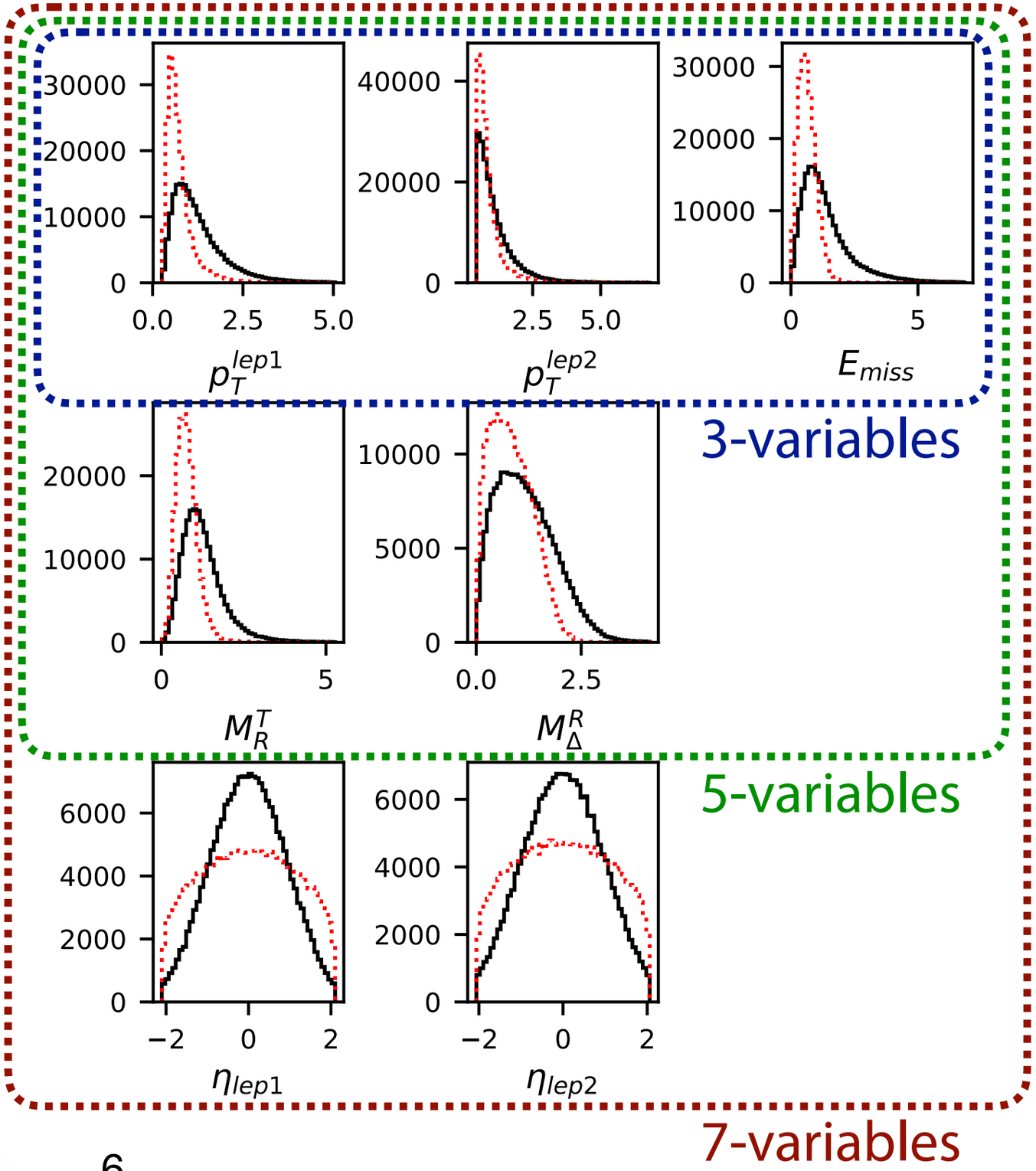
Benchmark Classification

Classification of SUSY signal from Standard Model background

SUSY dataset in UCI machine learning repository used



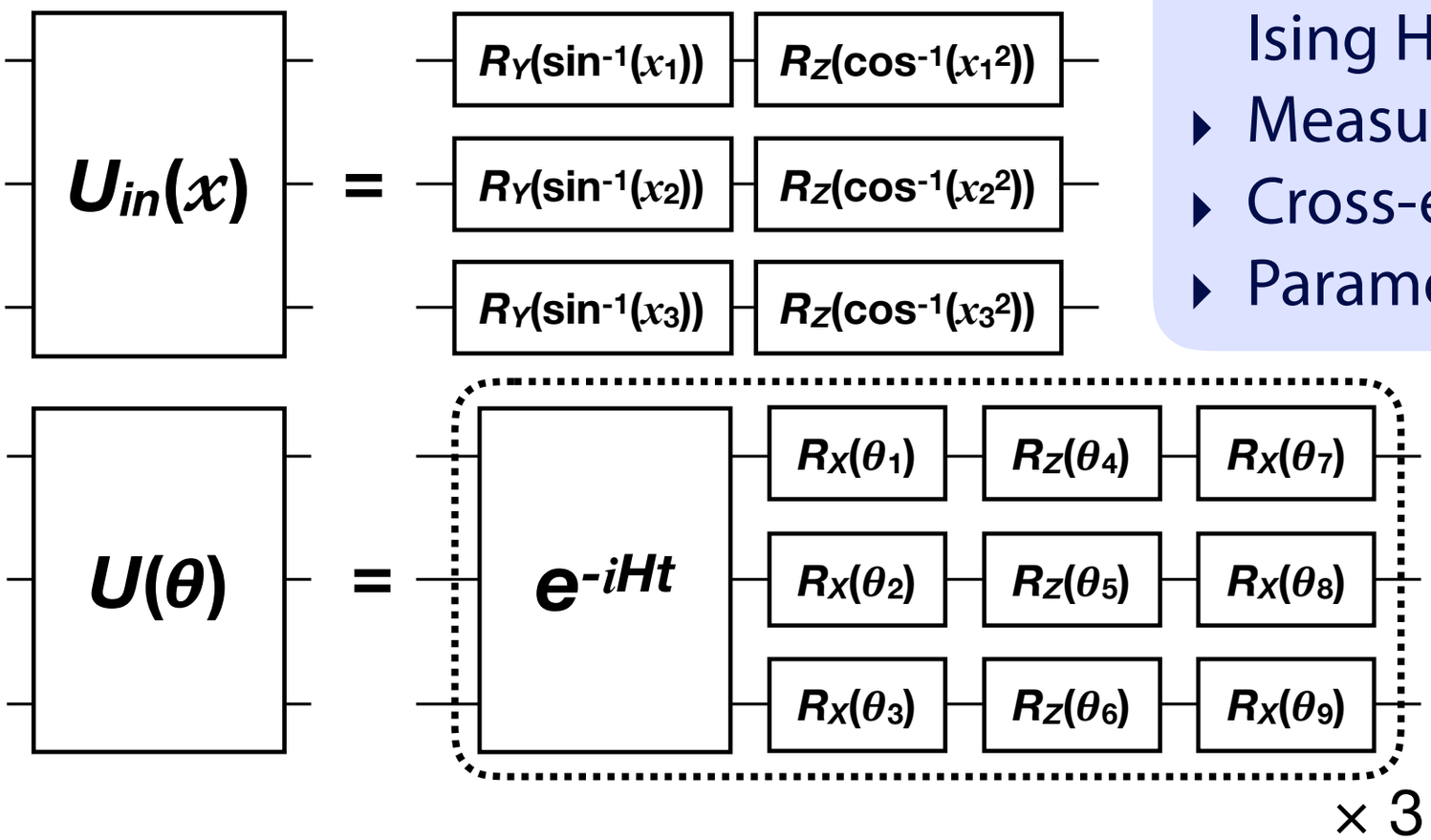




Quantum Circuit Learning

K. Mitarai *et al.*,
arXiv:1803.00745

Implemented using Qulacs simulator
(implemented in C/C++ with Python interface)



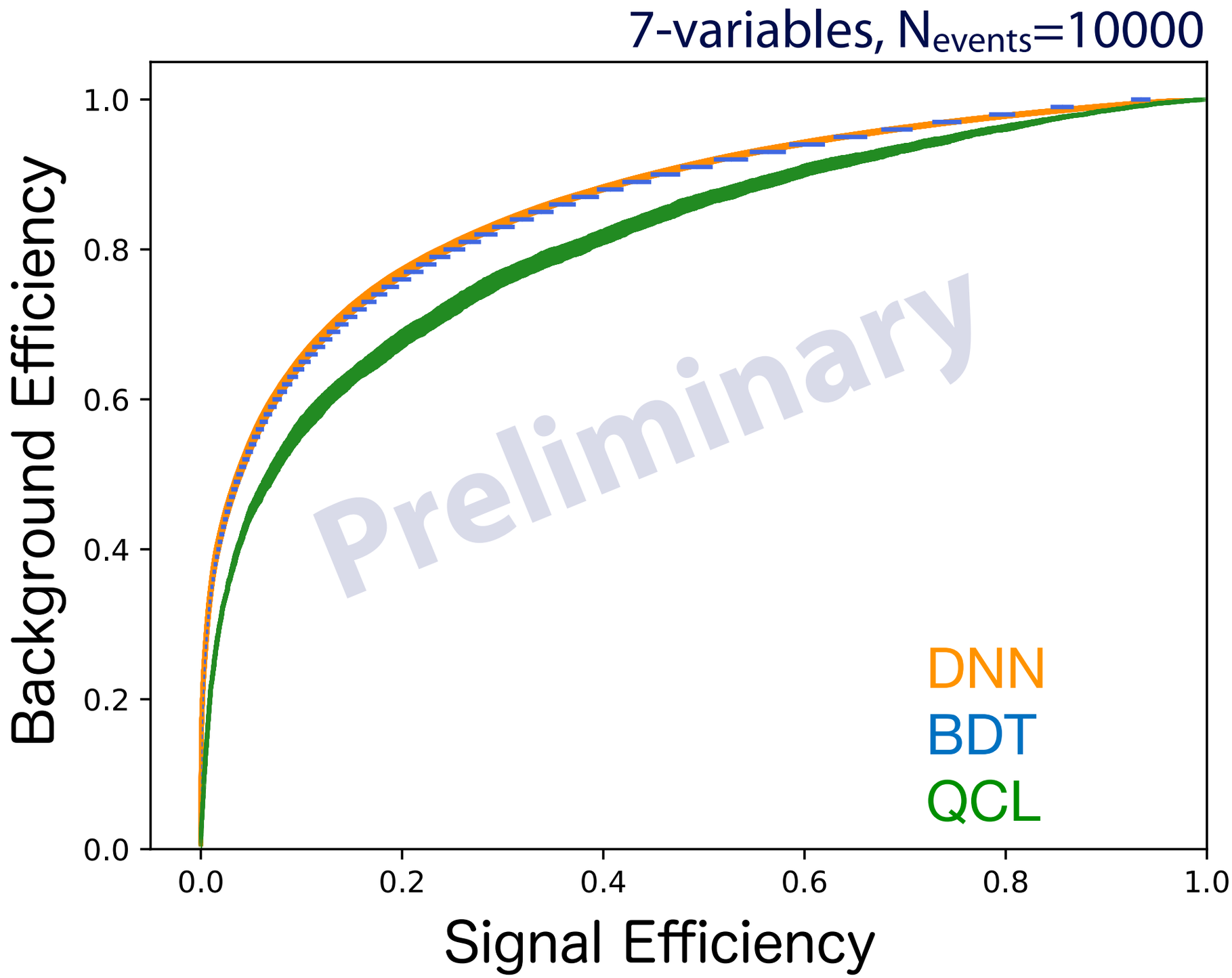
- Time-evolution gate e^{-iHt} in $U(\theta)$ with Ising Hamiltonian H
- Measure output states for Pauli-Z
- Cross-entropy loss as cost function
- Parameters optimized using COBYLA

Compared with ML methods: Boosted-Decision Tree and Deep NN

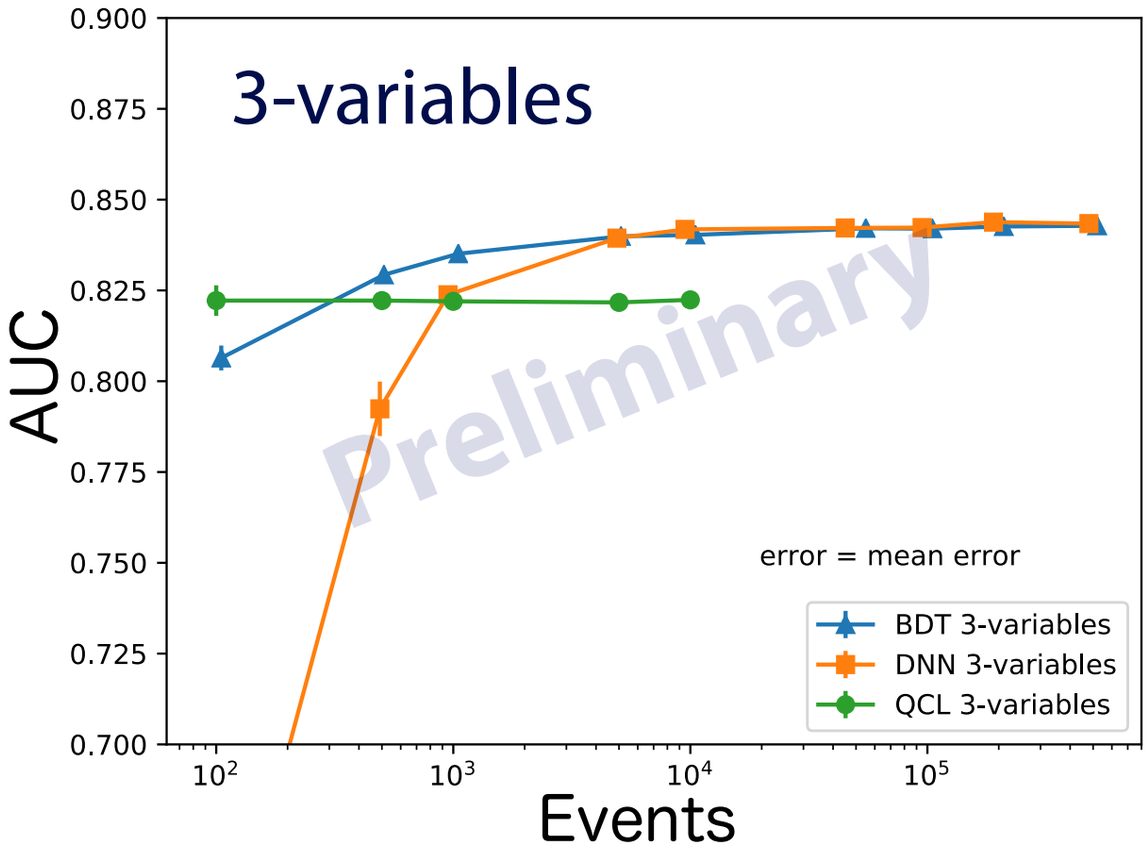
- BDT : Gradient boost, 1-3 max depth, 10-1000 #trees
- DNN : Dense, 1-5 hidden layers, 16-256 nodes, RELU, Adam, $\varepsilon_{learning}$=0.001

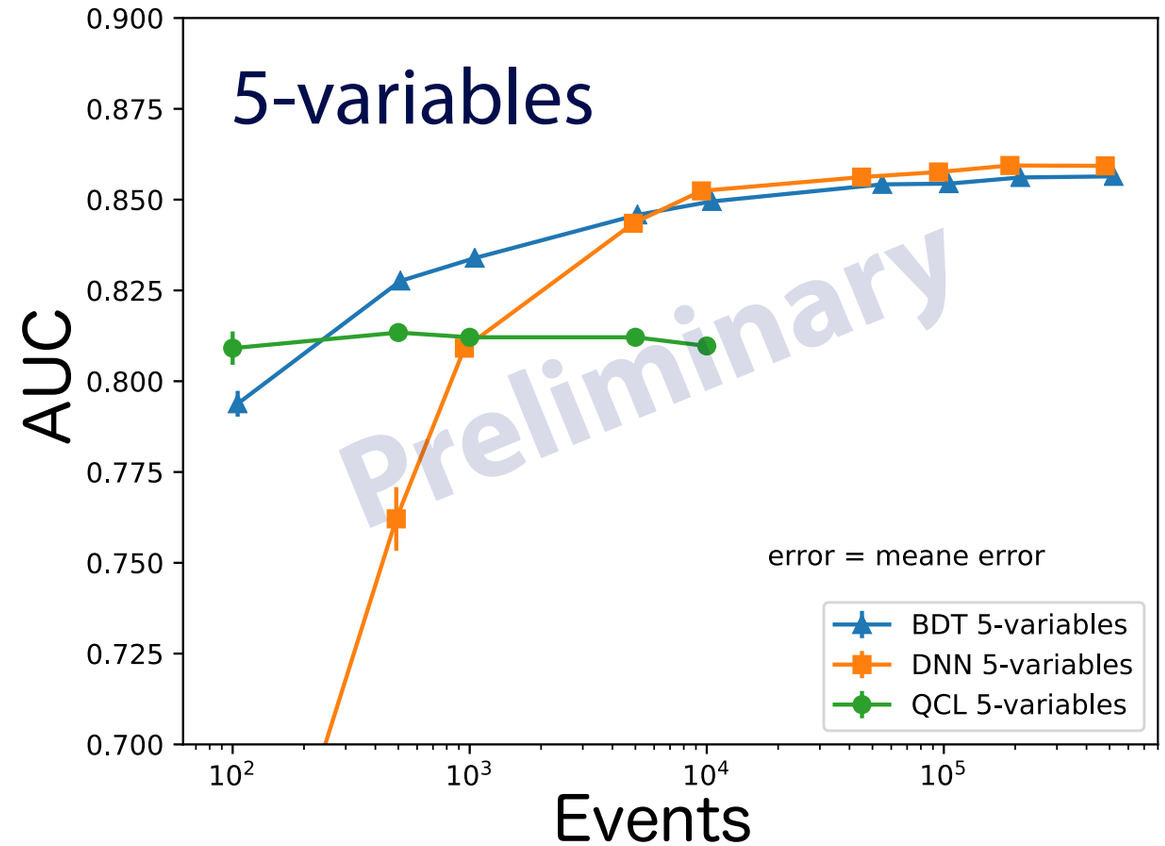
➡ Performance evaluated using AUC (area under ROC curve)

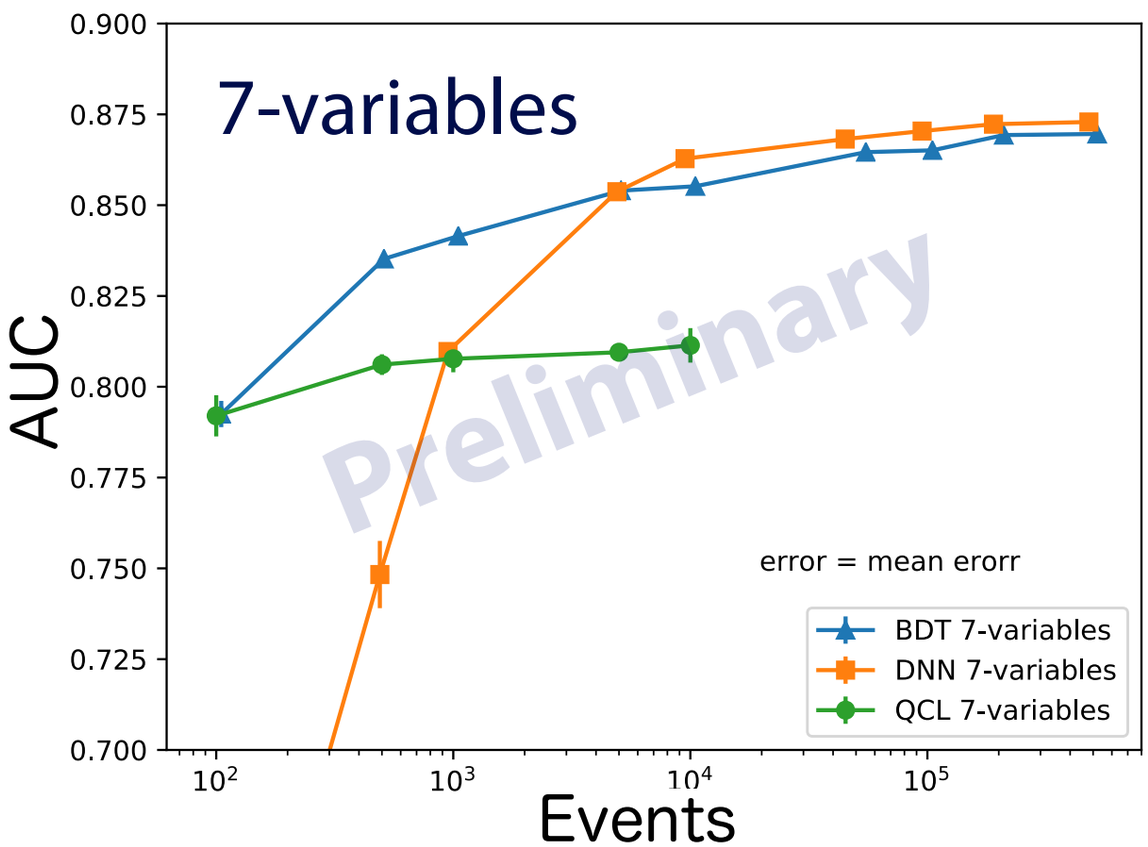
Preliminary QCL Results



Preliminary QCL Results







QCL performance relatively flat

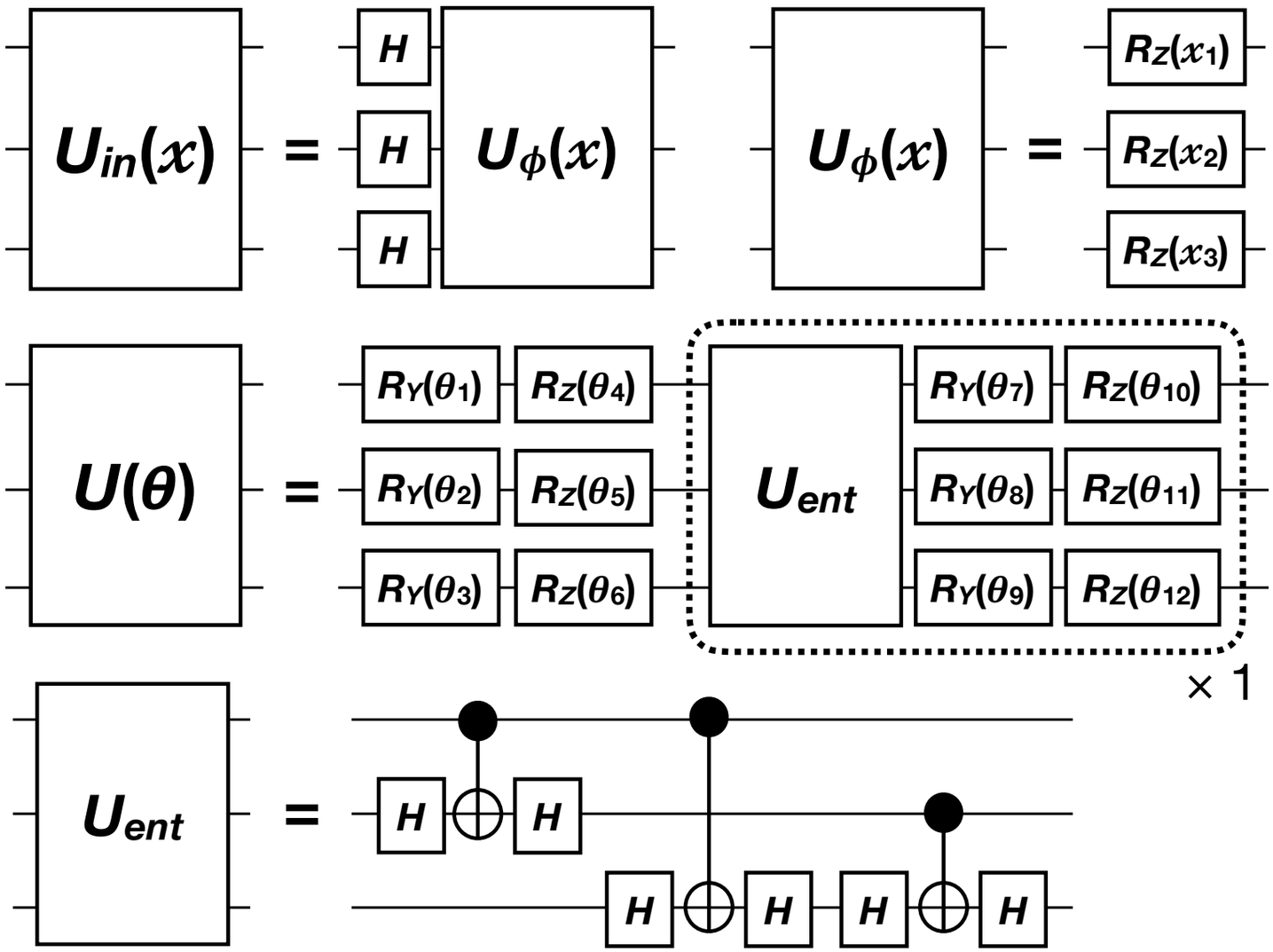
➡ Comparable to BDT or DNN at small samples with small # of input variables

Hard to run QCL with ≳10K events or ≳10 variables (exponential growth in time with #variables for ***U(θ)*** creation and COBYLA minimization)

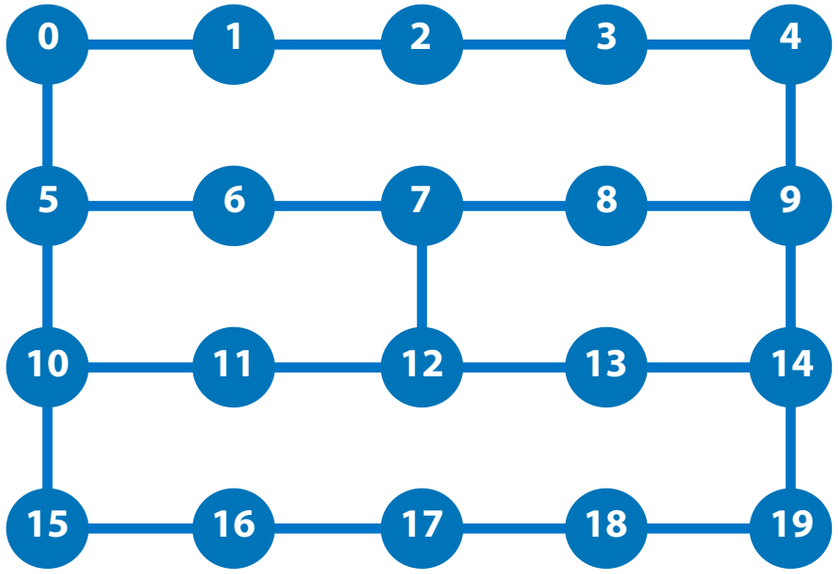
Variational Quantum Classification

Implemented using Qiskit Aqua framework

V. Havlicek *et al.*, arXiv:1804.11326



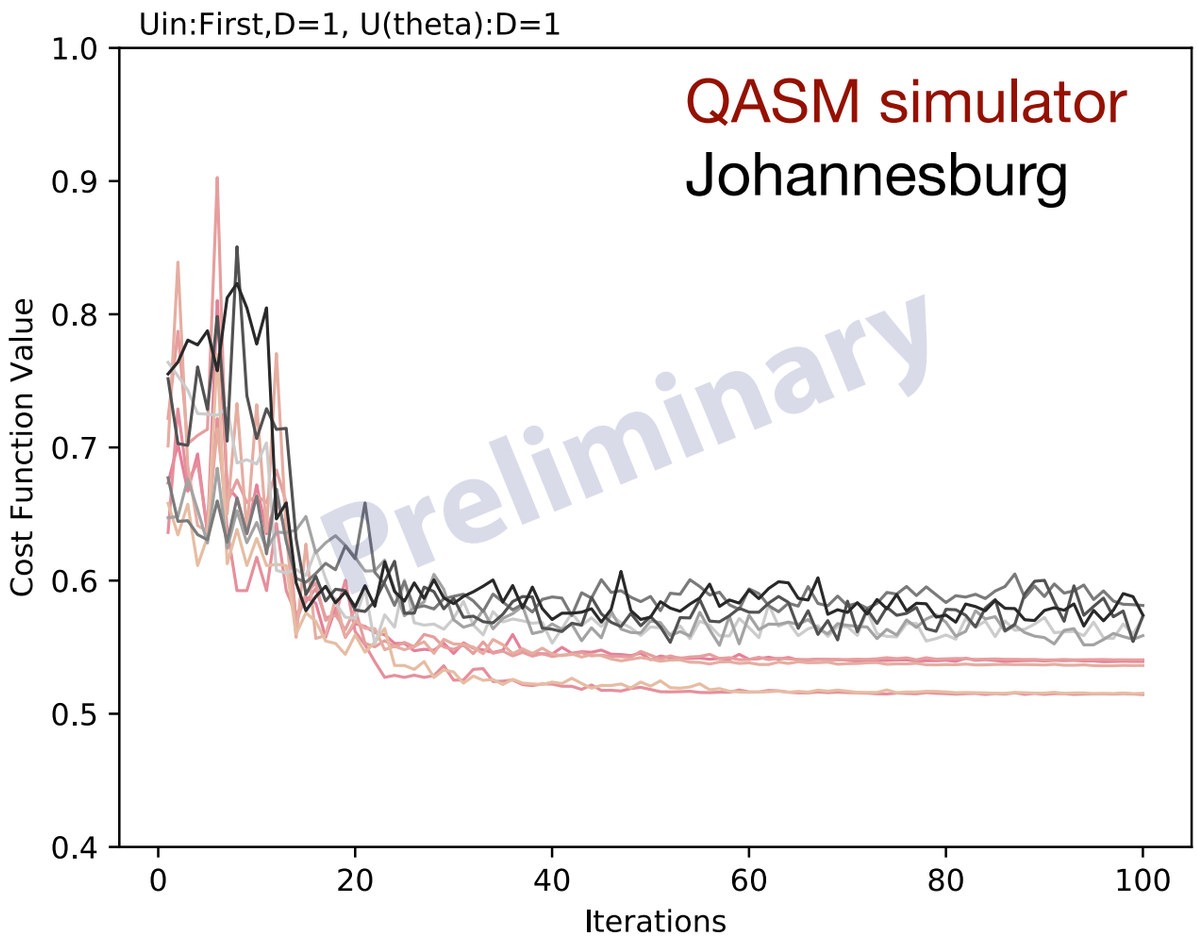
Johannesburg

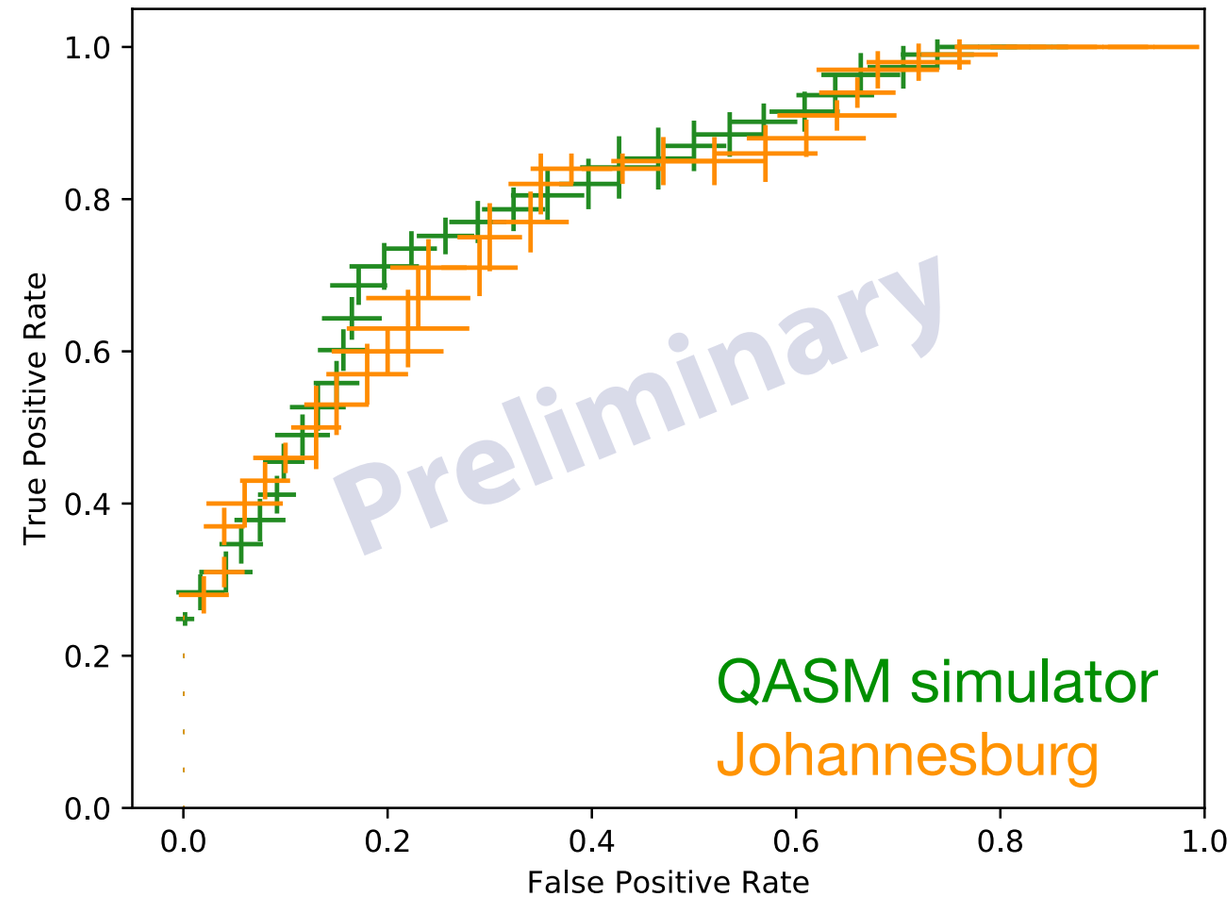


- *"First order expansion"* used in $U_{in}(x)$
- Entangling gate (U_{ent}) + rotation gates in $U(\theta)$
- Cross-entropy loss with COBYLA minimization

Tested using 20-qubit IBM Q Network device and QASM simulator

Preliminary VQC Results





Tested 3-variable classification with 40 training events

- Cost function reaches minimum after ~50 iterations
  - Slight offset for real device (likely caused by error due to noise)
  - Indication of improvement with measurement error mitigation (under study)
- ROC curves indicate VQC acquires discrimination power with real device
  - VQC with >3 variables and/or more extended circuits under study

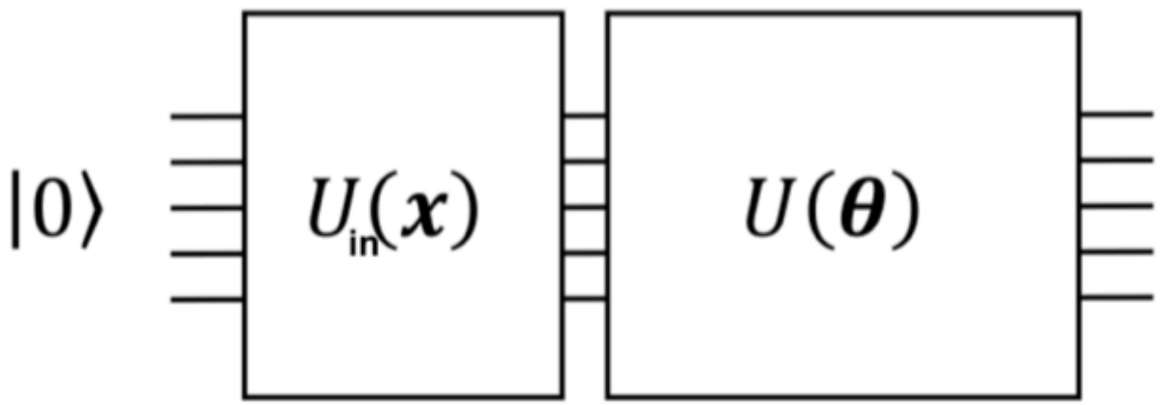
Summary

- Investigating the application of gate-based quantum machine learning algorithm to the event classification in HEP data analysis
- Variational quantum algorithm has comparable performance (in terms of AUC) to the conventional ML methods like BDT or DNN at small sample sizes and input variables
- On the way towards understanding/mitigation of errors due to hardware noise to improve performance with real device

Backup

Quantum Circuit Learning

PRA 98, 032309 (2018)



$\langle B(\boldsymbol{x},\boldsymbol{\theta})\rangle \rightarrow L(f(\boldsymbol{x}),\langle \boldsymbol{B}(\boldsymbol{x},\boldsymbol{\theta})\rangle)$

: to be minimized via $\boldsymbol{\theta}$

1. Prepare learning data : $(\boldsymbol{x}, \boldsymbol{y}^{\text{truth}})$ ($\boldsymbol{x}$=input data, $\boldsymbol{y}^{\text{truth}}$=teacher data)
2. Create input state $|\psi_{\text{in}}(\boldsymbol{x})\rangle = U_{in}(\boldsymbol{x})|0\rangle$ by encoding input data $\boldsymbol{x}$ with circuit $U_{in}(\boldsymbol{x})$

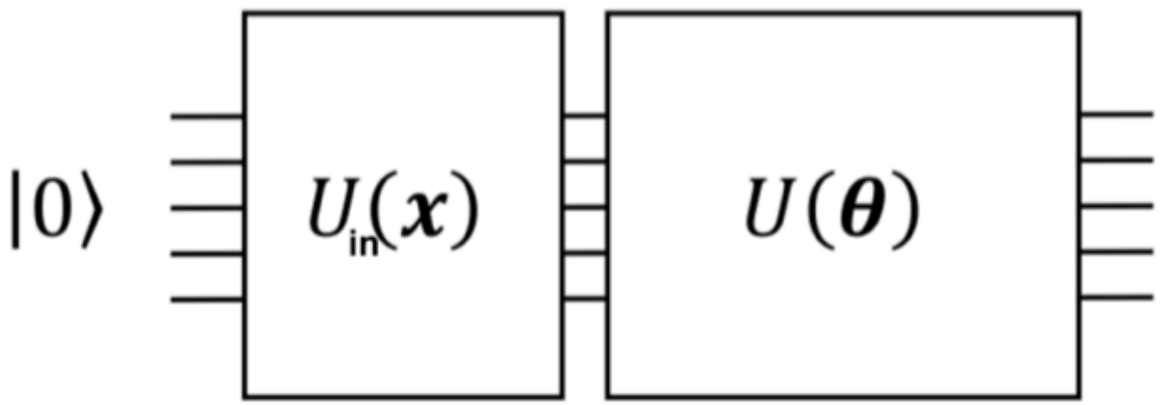
$$R_j^X(\theta) = e^{-i\theta X_j/2}, R_j^Z(\theta) = e^{-i\theta Z_j/2}$$

$$U_{\text{in}}(x) = \prod_j R_j^Z(\cos^{-1} x^2) R_j^Y(\sin^{-1} x)$$

$$|\psi_{\text{in}}(x_i)\rangle = U_{\text{in}}(x_i)|00\ldots0\rangle$$

Quantum Circuit Learning

PRA 98, 032309 (2018)



$\langle B(\boldsymbol{x},\boldsymbol{\theta})\rangle \rightarrow L(f(\boldsymbol{x}),\langle \boldsymbol{B}(\boldsymbol{x},\boldsymbol{\theta})\rangle)$

: to be minimized via $\boldsymbol{\theta}$

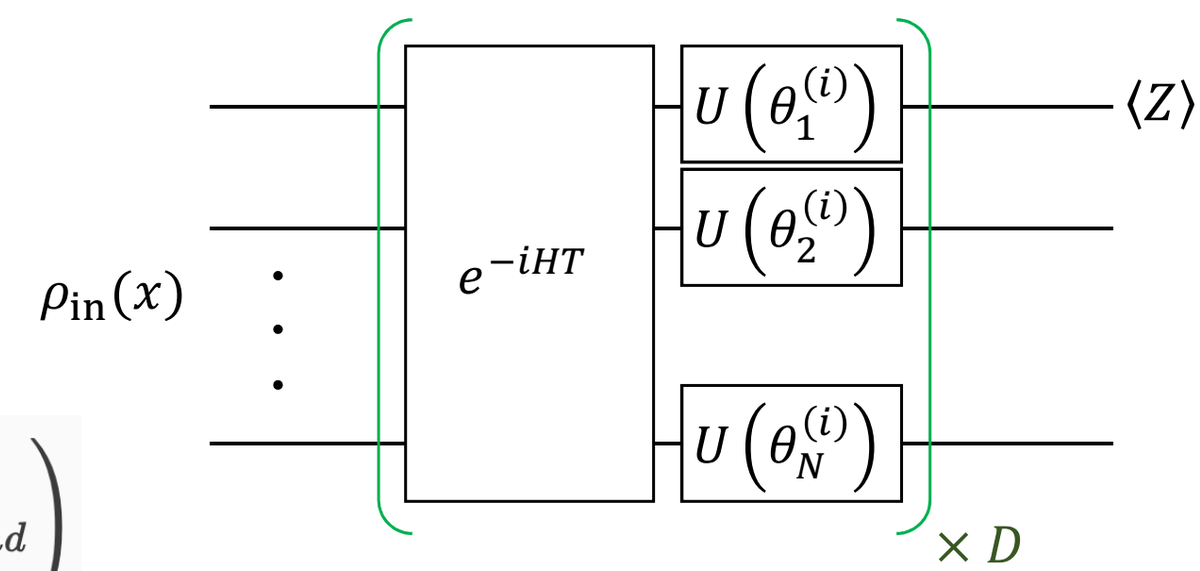
1. Prepare learning data : $(\boldsymbol{x}, \boldsymbol{y}^{\mathbf{truth}})$ ($\boldsymbol{x}$=input data, $\boldsymbol{y}^{\mathbf{truth}}$=teacher data)
2. Create input state $|\psi_{\text{in}}(\boldsymbol{x})\rangle = U_{in}(\boldsymbol{x})|0\rangle$ by encoding input data $\boldsymbol{x}$ with circuit $U_{in}(\boldsymbol{x})$
3. Apply variational circuit with parameter $\boldsymbol{\theta}$ to the input state and create output state $|\psi_{\text{out}}(\boldsymbol{x}, \boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\psi_{\text{in}}(\boldsymbol{x})\rangle$

$$H = \sum_{j=1}^{N} a_j X_j + \sum_{j=1}^{N}\sum_{k=1}^{j-1} J_{jk} Z_j Z_k$$

$$U_{rand} = e^{-iHt}$$

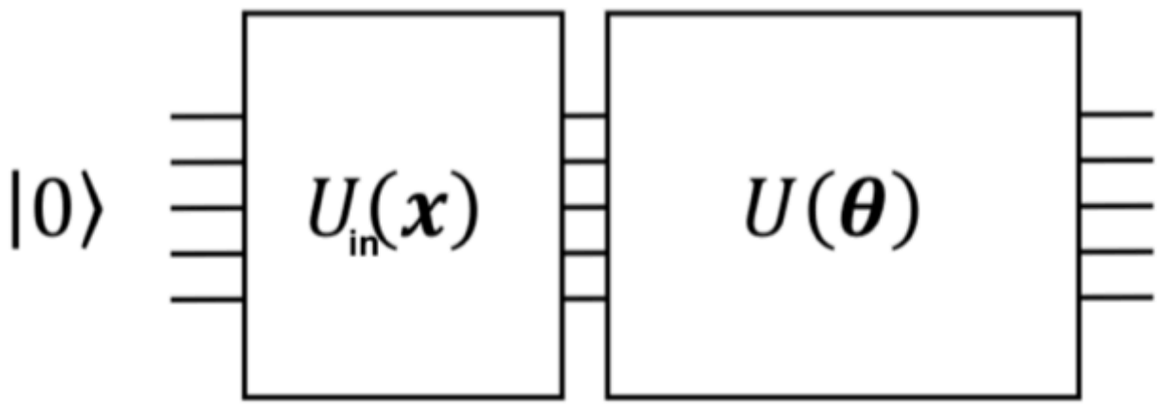
$$U_{rot}(\theta_j^{(i)}) = R_j^X(\theta_{j1}^{(i)}) R_j^Z(\theta_{j2}^{(i)}) R_j^X(\theta_{j3}^{(i)})$$

$$U\left(\{\theta_j^{(i)}\}_{i,j}\right) = \prod_{i=1}^{d}\left(\left(\prod_{j=1}^{n} U_{rot}(\theta_j^{(i)})\right)\cdot U_{rand}\right)$$



Quantum Circuit Learning

PRA 98, 032309 (2018)



$\langle B(\boldsymbol{x},\boldsymbol{\theta})\rangle \rightarrow L(f(\boldsymbol{x}),\langle \boldsymbol{B}(\boldsymbol{x},\boldsymbol{\theta})\rangle)$
: to be minimized via $\boldsymbol{\theta}$

1. Prepare learning data : $(\boldsymbol{x}, \boldsymbol{y}^{\mathbf{truth}})$ ($\boldsymbol{x}$=input data, $\boldsymbol{y}^{\mathbf{truth}}$=teacher data)
2. Create input state $|\psi_{\text{in}}(\boldsymbol{x})\rangle = U_{in}(\boldsymbol{x})|0\rangle$ by encoding input data $\boldsymbol{x}$ with circuit $U_{in}(\boldsymbol{x})$
3. Apply variational circuit with parameter $\boldsymbol{\theta}$ to the input state and create output state $|\psi_{\text{out}}(\boldsymbol{x}, \boldsymbol{\theta})\rangle = U(\boldsymbol{\theta})|\psi_{\text{in}}(\boldsymbol{x})\rangle$
4. Measure certain observables with output state
   (e.g, expectation values of Z-gate at *qubit i* : $\langle Z_i\rangle = \langle\psi_{\text{out}}|Z_i|\psi_{\text{out}}\rangle$)
5. Determine output $\boldsymbol{y}(\boldsymbol{x}, \boldsymbol{\theta})$ using some function $F = F(\langle Z_i\rangle)$, e.g, softmax
6. Define a cost function $L(\boldsymbol{\theta})$ from $\boldsymbol{y}(\boldsymbol{x}, \boldsymbol{\theta})$ and $\boldsymbol{y}^{\mathbf{truth}}$, e.g, cross-entropy loss
7. Find $\boldsymbol{\theta} = \boldsymbol{\theta}_{\mathbf{min}}$ that minimizes $L(\boldsymbol{\theta})$ ➔ **Trained model =** $\boldsymbol{y}(\boldsymbol{x}, \boldsymbol{\theta}_{\mathbf{min}})$

Input Features