Quantum Generative Models in HEP

Andrea Delgado

Physics Division, Oak Ridge National Laboratory, USA
Outline

• Quantum Generative Models: An Overview

• Applications in data analysis in HEP

• Characterizing quantum generative models through model capacity and trainability.
Quantum Generative Modeling

What?

• Quantum Generative Models are a powerful tool in QML to reproduce the statistics of a target distribution or quantum state ensemble, that can accordingly be used to generate new samples.

Quantum Generative Modeling

**What?**

- Quantum Generative Models are a powerful tool in QML to reproduce the statistics of a target distribution or quantum state ensemble, that can accordingly be used to generate new samples.

**Why?**

- Motivated by the capacity of quantum processors to **learn, represent, and sample from high-dimensional probability distributions**.
- Relatively simple quantum systems can generate data whose statistics cannot be generated efficiently by any classical system.

Represent vectors in N-dimensional spaces using $\log(N)$ qubits

Perform manipulations of sparse and low-rank matrices in time $O(\text{poly}(\log(N)))$
Quantum Generative Models

What?

• **Quantum Generative Models** are a powerful tool in QML to reproduce the statistics of a target distribution or quantum state ensemble, that can accordingly be used to generate new samples.

Why?

• Motivated by the capacity of quantum processors to **learn, represent, and sample from high-dimensional probability distributions**.
• Relatively simple quantum systems can generate data whose statistics **cannot be generated efficiently by any classical system**.

How?

• Several architectures explored, algorithms developed for quantum annealers, discrete and continuous variable.
• These models are **inspired by classical neural network models** and have been translated either as a standalone VQC or as a component in a hybrid network.

*Many quantum computing libraries have been developed that leverage existing classical ML libraries –* TensorFlow Quantum, TorchQuantum, PennyLane.*
Quantum Generative Models in HEP

• Inspired by applications in data augmentation, simulation, data compression tasks.
• HEP datasets provide a natural alternative to synthetic datasets to explore entanglement, expressiveness and scalability in QML models.
  • If proven to be scalable can demonstrate to have an advantage in generating high-dimensional correlated events.
• Also, can potentially have applications in large-scale quantum sensor networks, anomaly detection in quantum-enhanced probes for BSM physics, data embedding.
• Trained by minimizing the energy of a model (q-RBMs), the error when sampling from a target posterior (QCMBs), or through adversarial methods (q-GANs).
Quantum Generative Models in HEP

• (Restricted) Boltzmann machines are physically motivated NNs capable of generating new samples similar to the training data.

• Weights and biases are optimized by finding the ground state of a system’s Hamiltonian – thus, are perfectly suited for quantum annealers.


Quantum Generative Models in HEP

An autoencoder is based on a two-component network:

- A network maps an input vector \( x \) to a compressed “latent space”.
- A second network maps back the latent vector into feature space.
- Network is trained to minimize the error of the reconstructed input state or vector.

In the quantum setting, q-AEs can be used for generative modeling, data compression and anomaly detection.

Quantum Generative Models in HEP

- Adversarial training with quantum generator and classical or quantum discriminator.

Applications:

\[
\min_{\phi_g} \mathcal{L}_G(\phi_g, \phi_d), \quad \max_{\phi_d} \mathcal{L}_D(\phi_g, \phi_d)
\]
Quantum Generative Models in HEP

- QCBMs are parameterized quantum circuits used with the objective of preparing a target distribution with high-fidelity.

- A QCBM rotates a fixed initial state to a final state, then samples from that final state.

- A commonly used ansatz for QCBMs is the "hardware efficient ansatz": constructed by alternating layers of rotation gates with layers of two-qubit entangling operations.

- The design space for QCBM models is large – there are many choices for initial state, ansatz, and measurement setting.

- Building scalable QCBM models for HEP must balance reproducing a target distribution with high fidelity with trainability and noise robustness.
Unsupervised Quantum Circuit Learning

QCBM training is a **hybrid quantum-classical workflow**

Parameterized circuit

Rotation

Entangling

\[ |0\rangle \quad \theta_1^\alpha \quad |0\rangle \quad \theta_1^\alpha \quad \cdots \quad \theta_1^\alpha \]

\[ \cdots \quad |0\rangle \quad \theta_1^\alpha \quad |0\rangle \quad \theta_1^\alpha \quad \cdots \quad \theta_1^\alpha \]

\[ \cdots \quad |0\rangle \quad \theta_1^\alpha \quad |0\rangle \quad \theta_1^\alpha \quad \cdots \quad \theta_1^\alpha \]

\[ \cdots \quad |0\rangle \quad \theta_1^\alpha \quad |0\rangle \quad \theta_1^\alpha \quad \cdots \quad \theta_1^\alpha \]

Sampled Distribution

Loss Function Evaluation

Parameter Update

Gradient

Classical Optimization

Measure
An Example: QCBMs for fitting 2 and 3-dimensional joint distributions

- Can we fit the marginal distributions?
- Are the correlations also preserved on the generated distribution?
- Ansatz choice: Trial and error
Hyperparameter Tuning

QCBM training simulated on CPU
- Using PennyLane library and Qulacs
- Adam optimizer

Larger circuits lead to faster convergence

Higher shot sizes improves performance

Ansatz 1

Ansatz 2
Marginal Fitting: 2D Joint Distributions (8 qubits)

Ansatz 1

Ansatz 2
Marginal Fitting: 3D Joint Distributions (12 qubits)

Ansatz 1
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_1})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_1})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})

Ansatz 2
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
|0> \rightarrow U_{\text{rot}}(\theta_l^{a_0})
Correlation Fitting

MC (Truth)

Ansatz 1

With $d=1$ layer all $n$-qubits can be entangled together

Ansatz 2

With $d=1$ layer $n$-qubits are prepared as separated sub-systems of $m$-qubits
The road to quantum advantage...

**Harder**
Train circuits that are harder to simulate in classical devices (classically intractable regime of QML)?

**Better**
Can we do better than trial and error when selecting an Ansatz? Produce a systematic method to characterize PQC in GM?

**Faster**
Can we train circuits faster? By optimizing circuit design and reduce the number of executions on hardware.

**Stronger**
Develop stronger, scalable error mitigation/correction techniques.
The road to quantum advantage…

**Harder**

Train circuits that are **harder** to simulate in classical devices (classically intractable regime of QML)?

**Better**

Can we do **better** than trial and error when selecting an Ansatz? Produce a systematic method to characterize PQC’s in GM?

**Faster**

Can we train circuits **faster**? By optimizing circuit design and reduce the number of executions on hardware.

**Stronger**

Develop **stronger**, scalable error mitigation/correction techniques.

Actively working on it…

trained QCBMs up to 15 qubits

Preliminary results on some ideas to tackle these
Building Symmetries into Quantum Circuit Learning

- How can we reduce the number of parameters in our circuit?
- Is there any symmetry we can exploit for QML applications?
- Inspired by ORB-type circuits.
  - Parameters are grouped into “orbits” with shared parameters.

(1) Take SU(2) Ansatz
(a) Free
(2) Group rotational gates on qubits that can be swapped without affecting symmetry
(b) ORB
(3) Operations on the same orbit share parameters, reducing the number of trainable parameters

Building Symmetries into Quantum Circuit Learning

More layers needed in Orb-type circuits to achieve similar performance than fully-parameterized circuits.

But effective number of trainable parameters is reduced.
Capacity and Trainability of Quantum Generative Models

As we add more layers, we get to lower JS values.

Performance also related to initial parameters.

As we add more layers, we get to lower JS values.
Capacity and Trainability of Quantum Generative Models

But... there is a limit to the model capacity.

... after a critical number of layers, performance can’t get any better.

Coinciding with the saturation in the QFI matrix rank!

But how does it relate to the circuit entangling structure?
Summary

- Quantum generative models are currently a promising candidate for quantum advantage in QML, with current performance comparable to classical methods. Still a lot of open questions:
  - Scalability?
  - Model capacity and how it is affected by entanglement in circuit.
  - Transitions in trainability.
  - Scalable error correction.

- Promising applications in HEP.
  - Finding complex correlations in data.
  - As a data augmentation tool.
  - As input models for other quantum algorithms.
  - To complement quantum-enhanced searches for BSM physics – i.e. quantum sensor networks.

An exciting time to work on QML!
Thank you!

Collaborators: Dr. Kathleen Hamilton (ORNL).

This work was partially supported by the Quantum Information Science Enabled Discovery (QuantiSED) for High Energy Physics program at ORNL under FWP ERKAP61. This work was partially supported by the Laboratory Directed Research and Development Program of ORNL, managed by UT-Battelle, LLC, for the U.S. Department of Energy. This work was partially supported as part of the ASCR Testbed Pathfinder Program at Oak Ridge National Laboratory under FWP ERKJ332. This work was partially supported as part of the ASCR Fundamental Algorithmic Research for Quantum Computing Program at Oak Ridge National Laboratory under FWP ERKJ354.

This research used quantum computing system resources of the Oak Ridge Leadership Computing Facility, which is a DOE Office of Science User Facility supported under Contract No. DE-AC05-00OR22725. Oak Ridge National Laboratory manages access to the IBM Q System as part of the IBM Q Network.