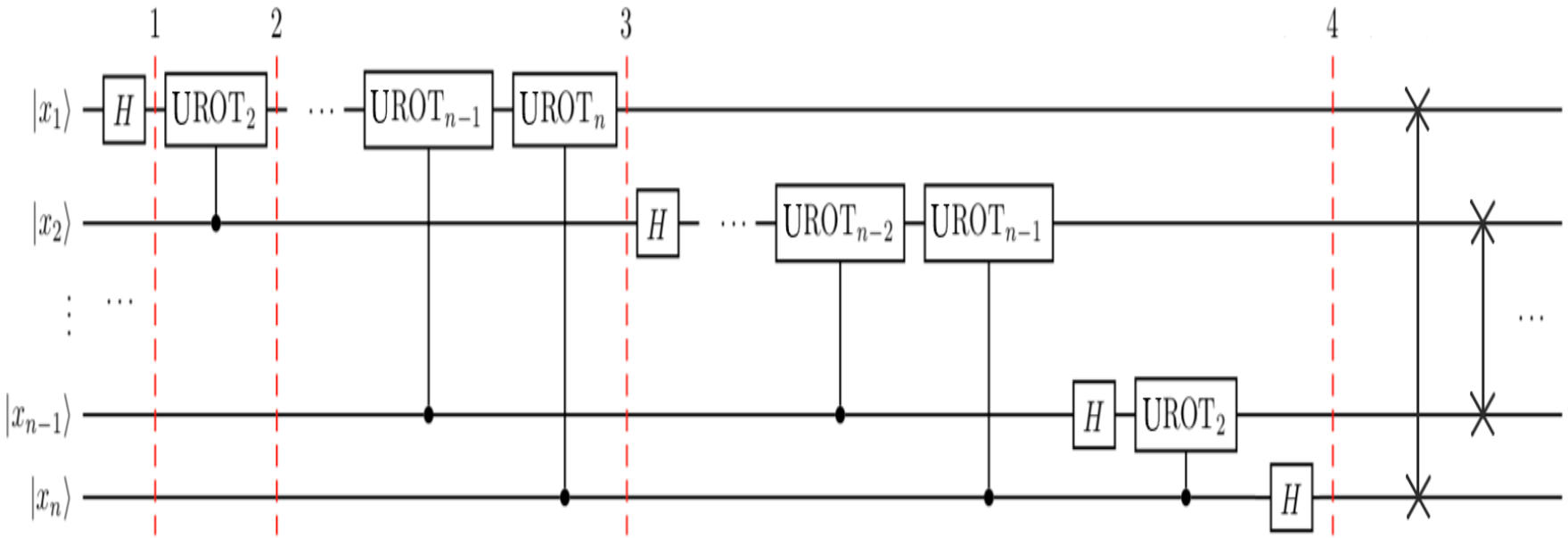
Weekly update

Abdualazem Fadol

March 24, 2023

中國科學院高能物理研究所
Institute of High Energy Physics
Chinese Academy of Sciences

Quantum Fourier Transform



- Quantum Fourier Transform (QFT) consist of:
  - A single-qubit Hadamard gate (H)
  - A two-qubit controlled rotation CP

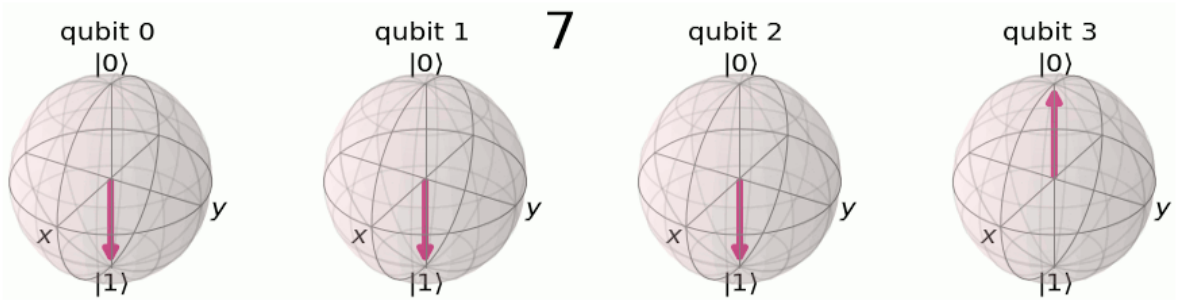
$$CP(\theta) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\theta} \end{bmatrix}$$

$$\theta = \pi/2^{k-1}$$

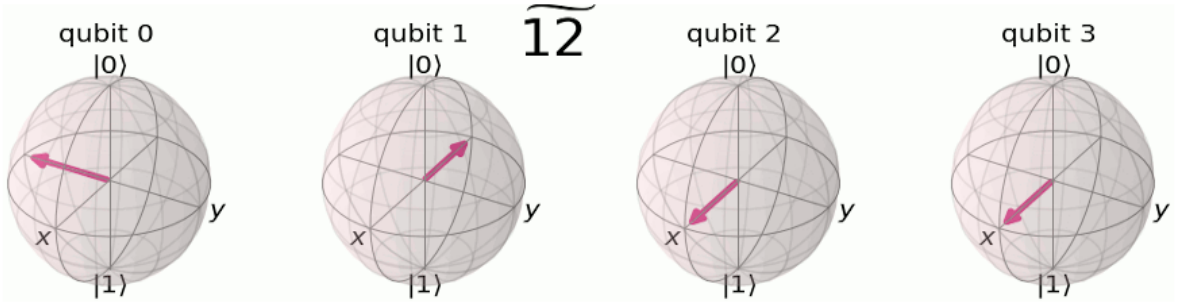
Counting in the Fourier basis on the Bloch Sphere

2.1 Counting in the Fourier basis:

In the computational basis, we store numbers in binary using the states $|0\rangle$ and $|1\rangle$:



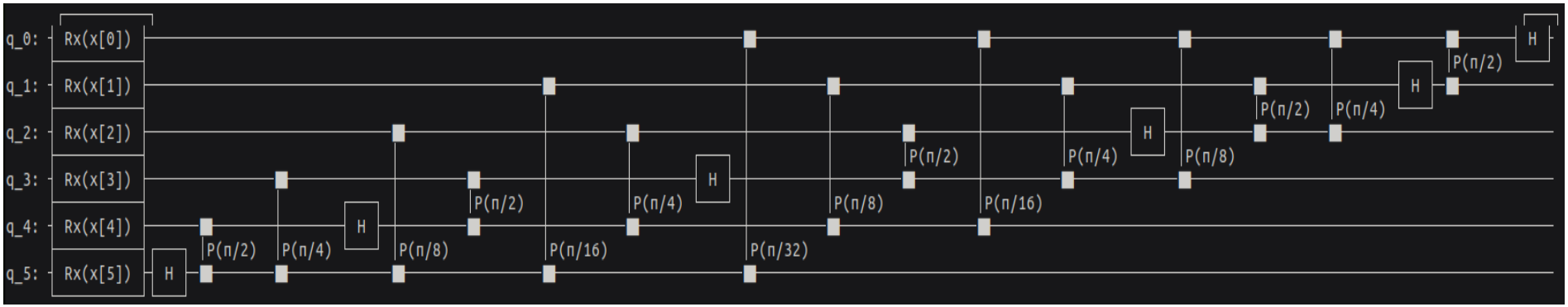
Note the frequency with which the different qubits change; the leftmost qubit flips with every increment in the number, the next with every 2 increments, the third with every 4 increments, and so on. In the Fourier basis, we store numbers using different rotations around the Z-axis:



The number we want to store dictates the angle at which each qubit is rotated around the Z-axis. In the state $|\tilde{0}\rangle$, all qubits are in the state $|+\rangle$. As seen in the example above, to encode the state $|\tilde{5}\rangle$ on 4 qubits, we rotated the leftmost qubit by $\frac{5}{2^n} = \frac{5}{16}$ full turns ($\frac{5}{16} \times 2\pi$ radians). The next qubit is turned double this ($\frac{10}{16} \times 2\pi$ radians, or $10/16$ full turns), this angle is then doubled for the qubit after, and so

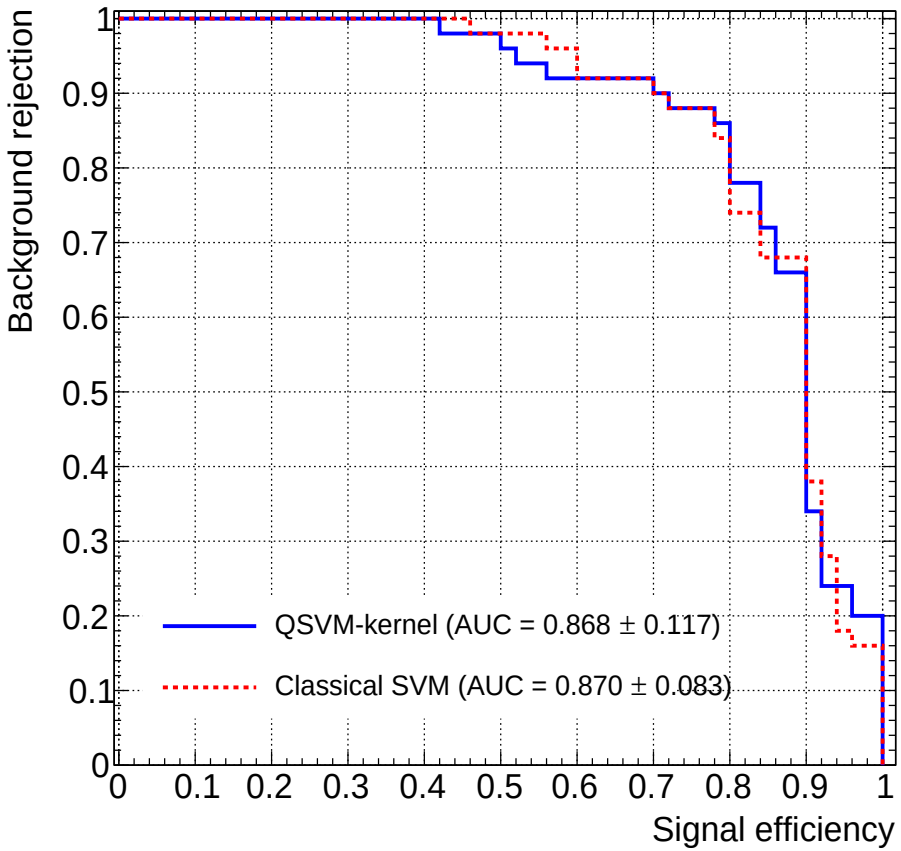
Qiskit explanation

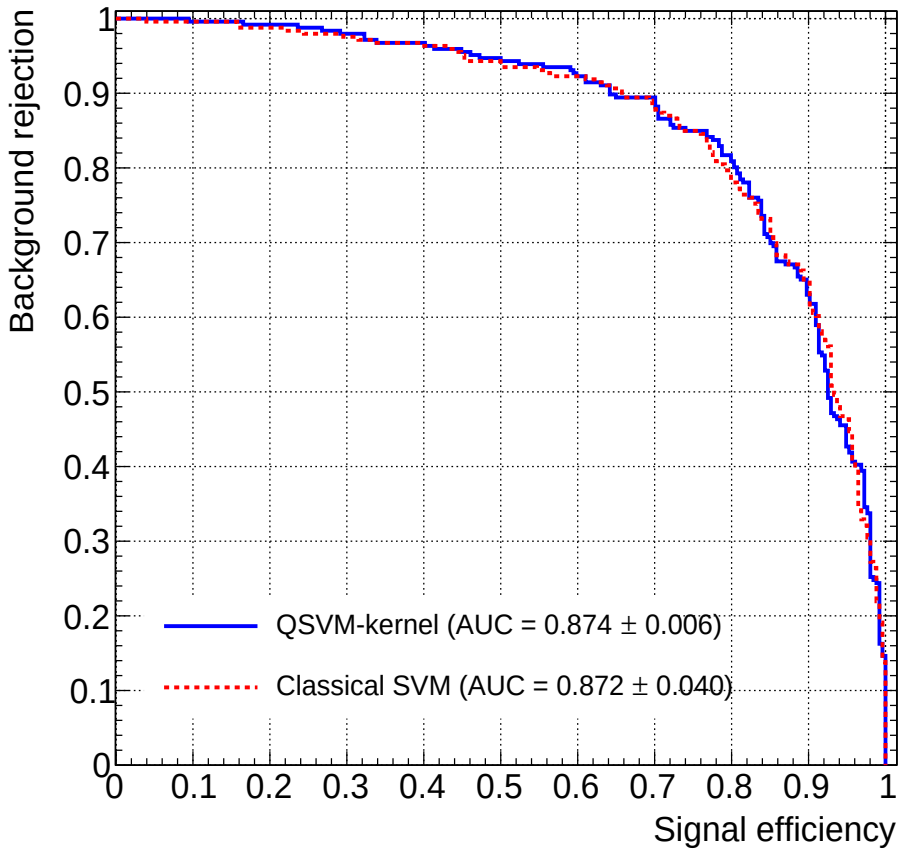
The implementation of QFT in ML



- We add a additional rotation in front of each qubit $R_x(\vec{x}_i)/R_y(\vec{x}_i)$.
- We should avoid the rotation around the z-axis since the Fourier basis rotate around it.
- Using a few qubits would work well with the QFT.
- However, the problem is how to scale this with the n-qubit case?

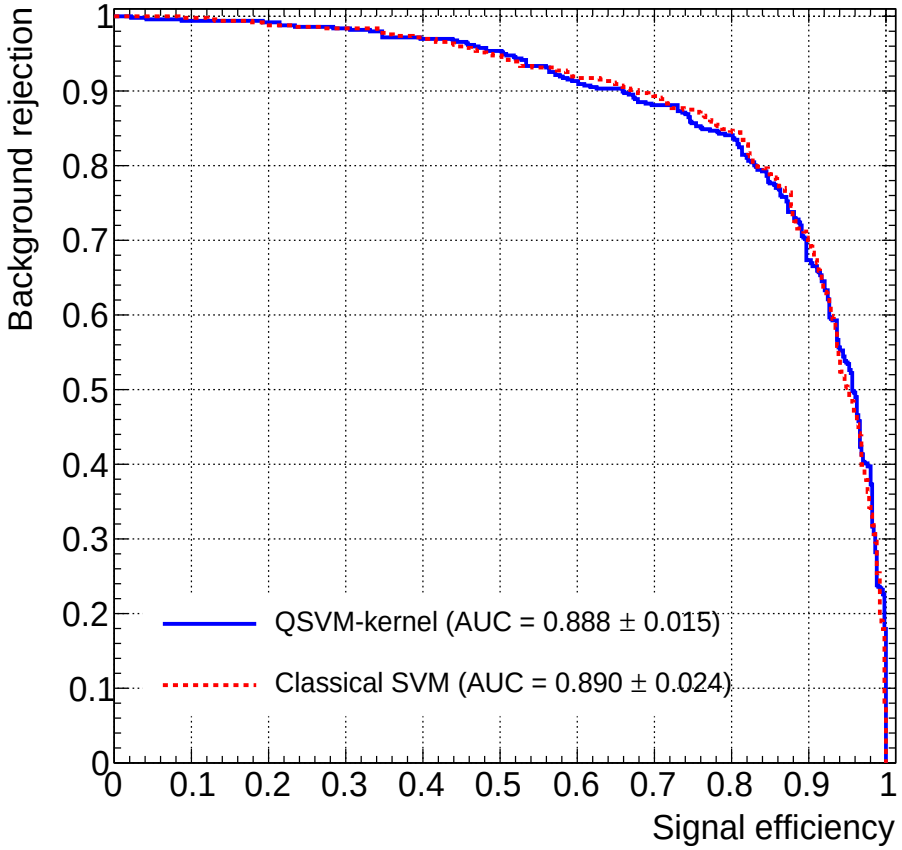
The implementation of QFT in ML

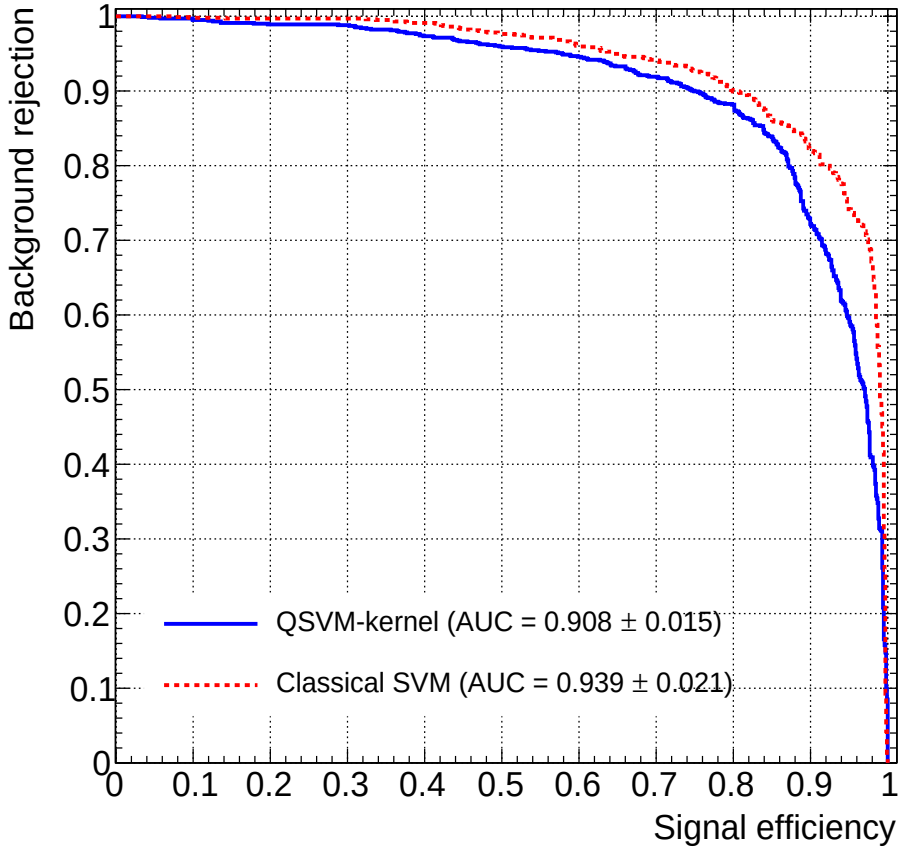




- Using 6 qubits along with the QFT and an additional $R_x(\vec{x}_i)$ gate.
- Testing a few events—200 (left) and 1000 (right) events.

The implementation of QFT in ML

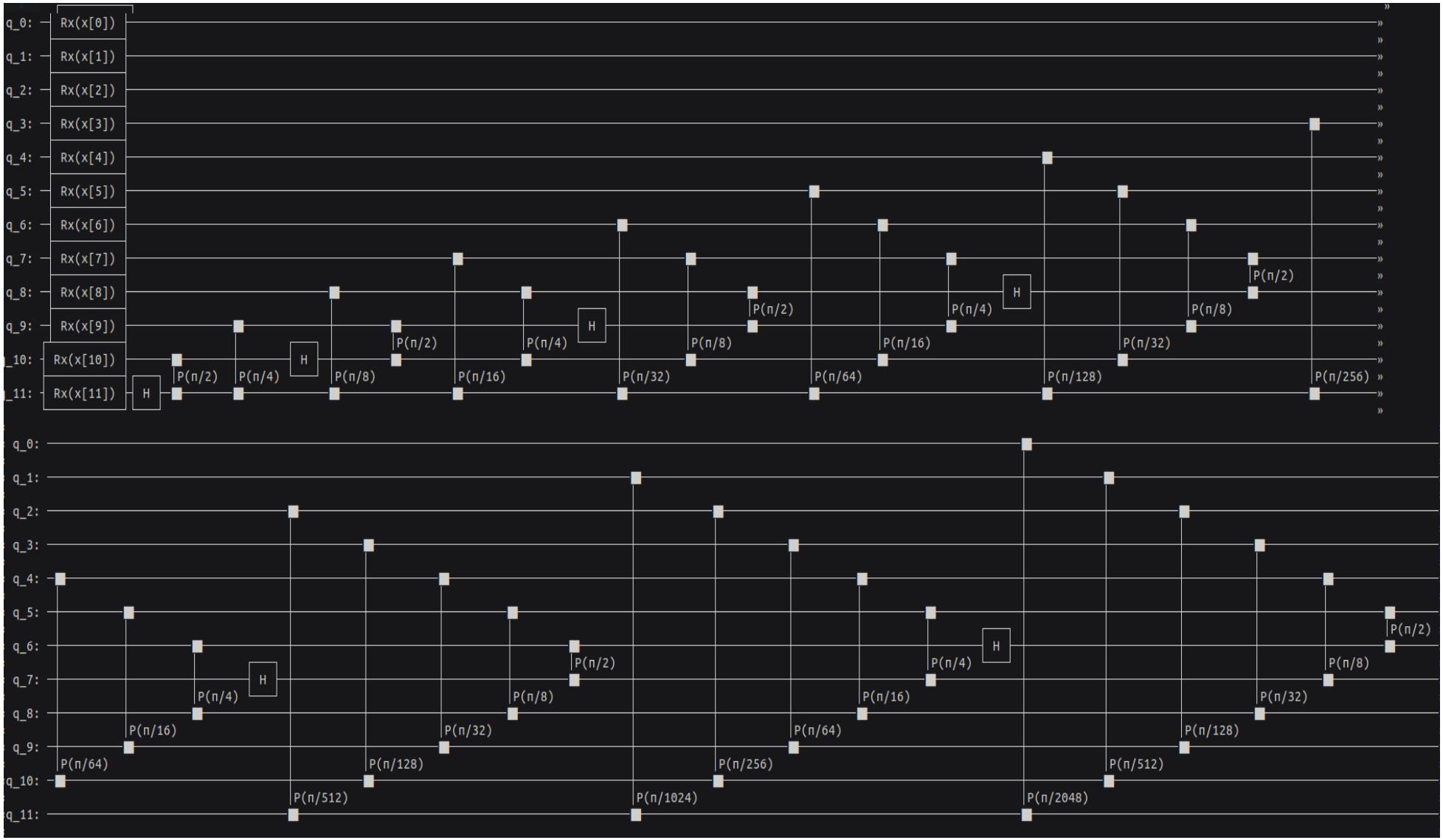




- 2000 (left) and 4000 (right) events with 6 qubits.
- The right plot looks a bit strange as the QSVM gets more worst by adding more events.

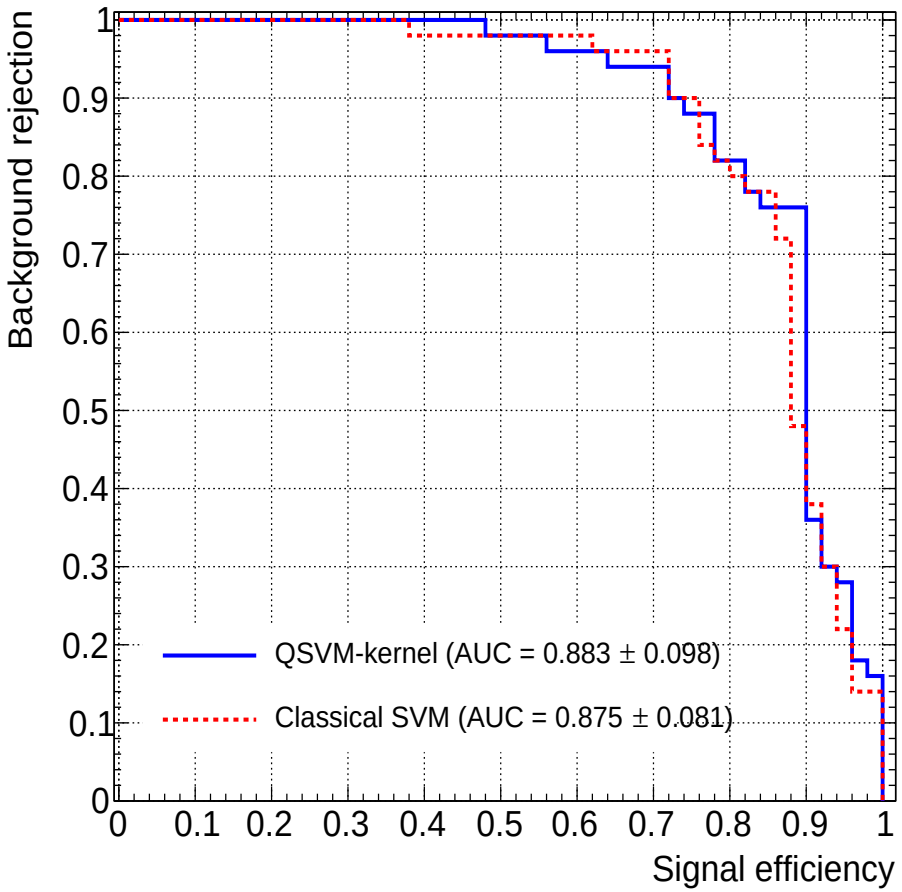
The implementation of QFT in ML

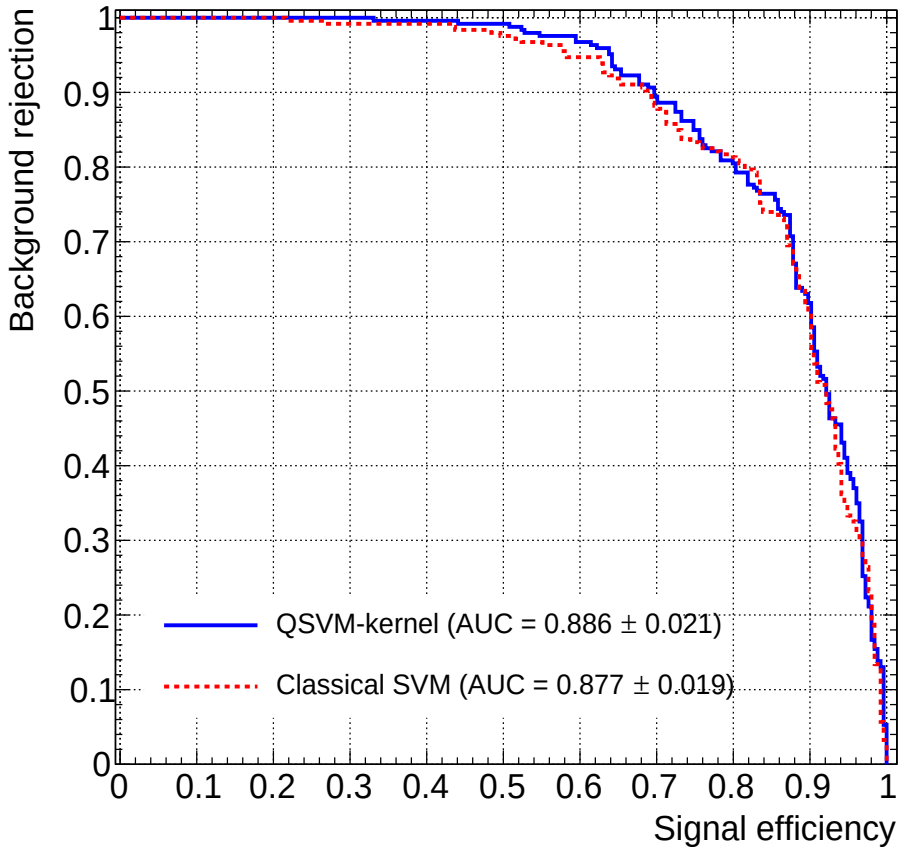
Checking the scalability of QFT with 12 qubits



The implementation of QFT in ML

Checking the scalability of QFT with 12 qubits

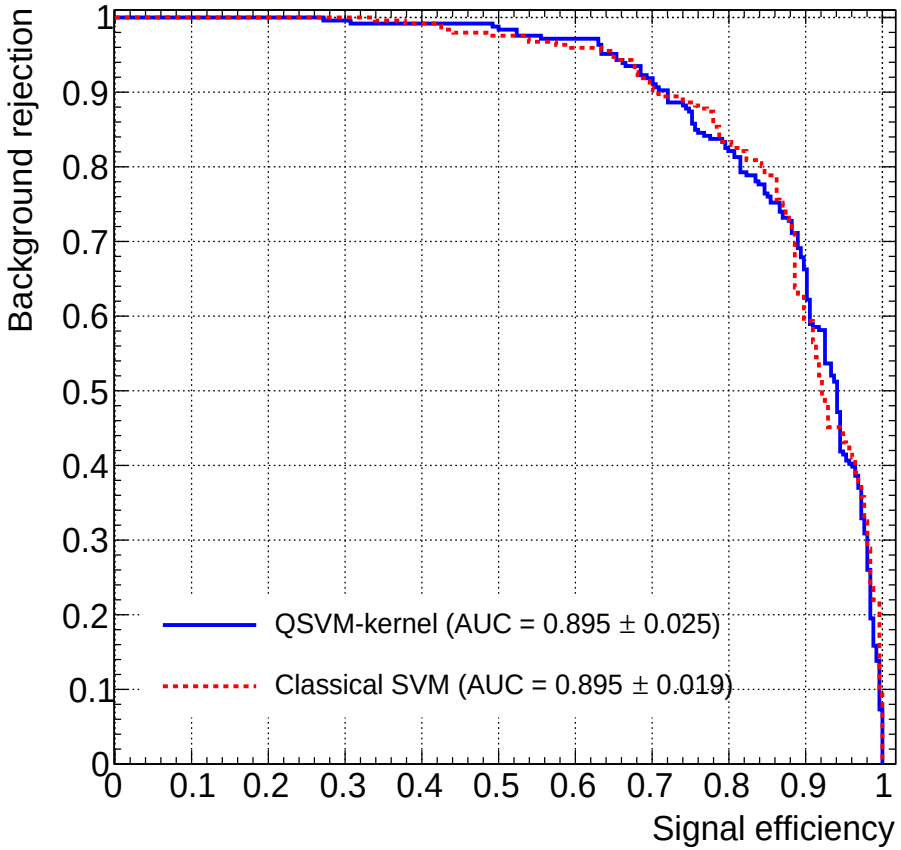


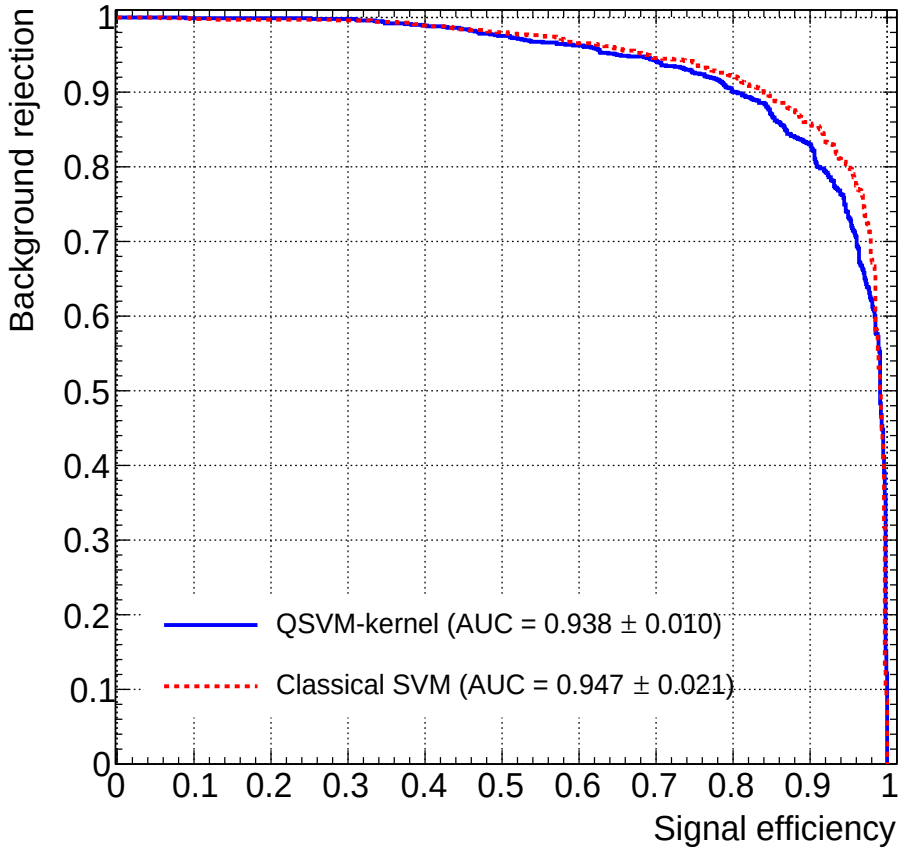


- 200 (left) and 1000 (right) events with 12 qubits.

The implementation of QFT in ML

Checking the scalability of QFT with 15 qubits

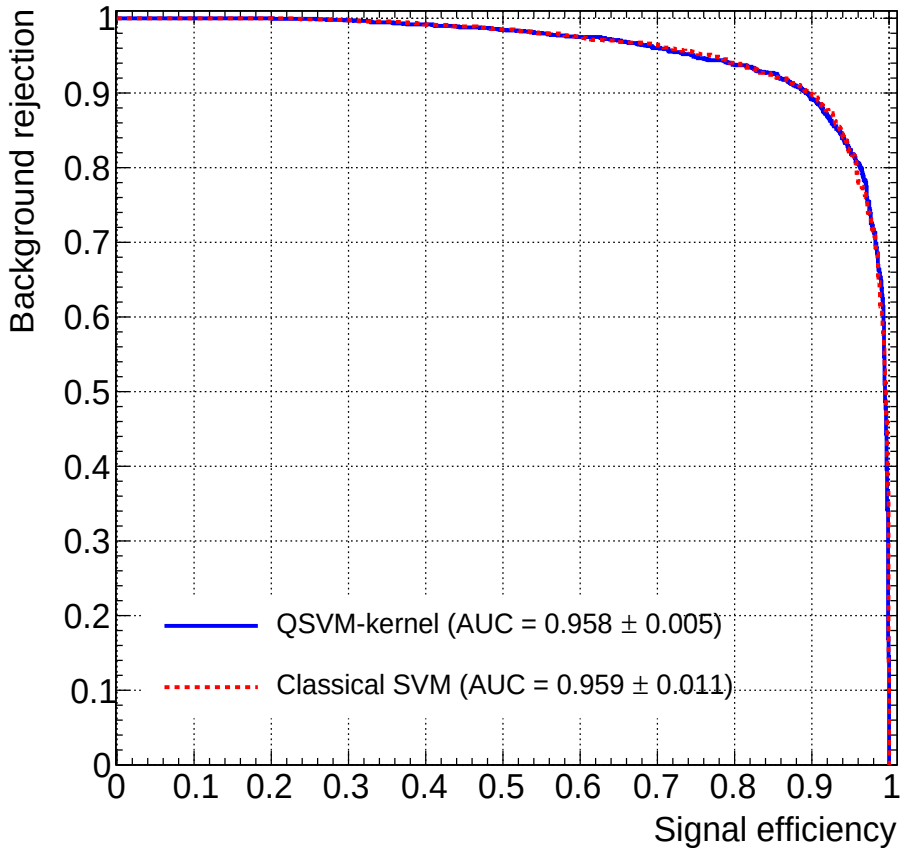




- 1000 events with 15 qubits (left) and 4000 (left) with 12 qubits.

The implementation of QFT in ML

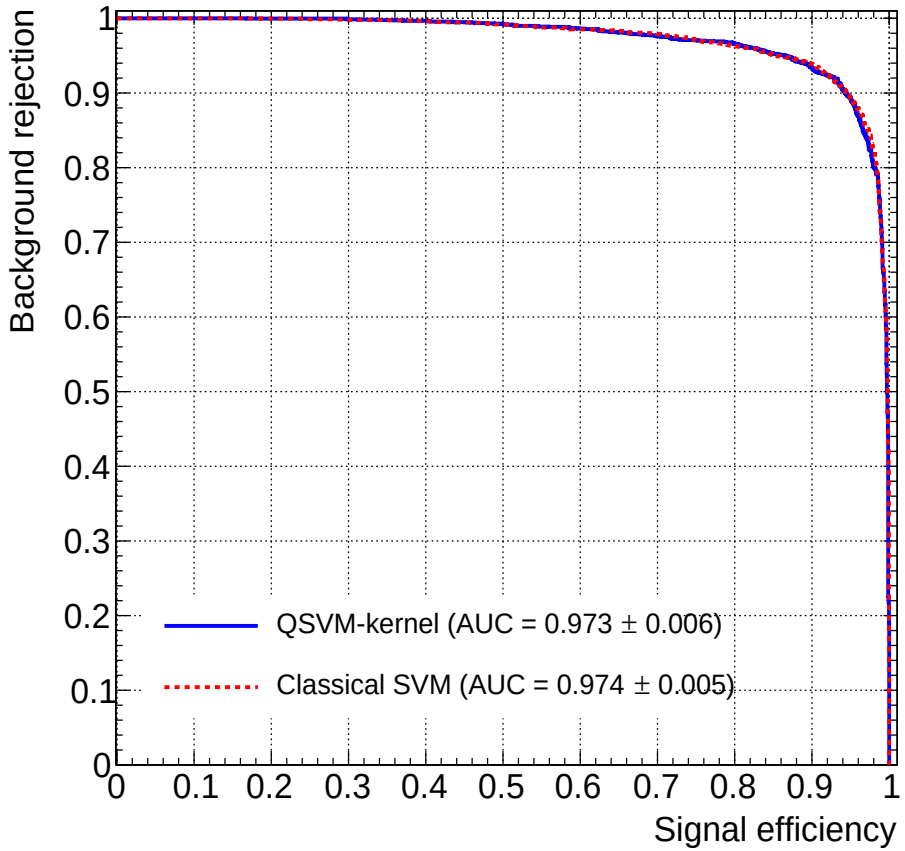
Checking the scalability of QFT with 15 qubits



- 6000 events with 15 qubits.

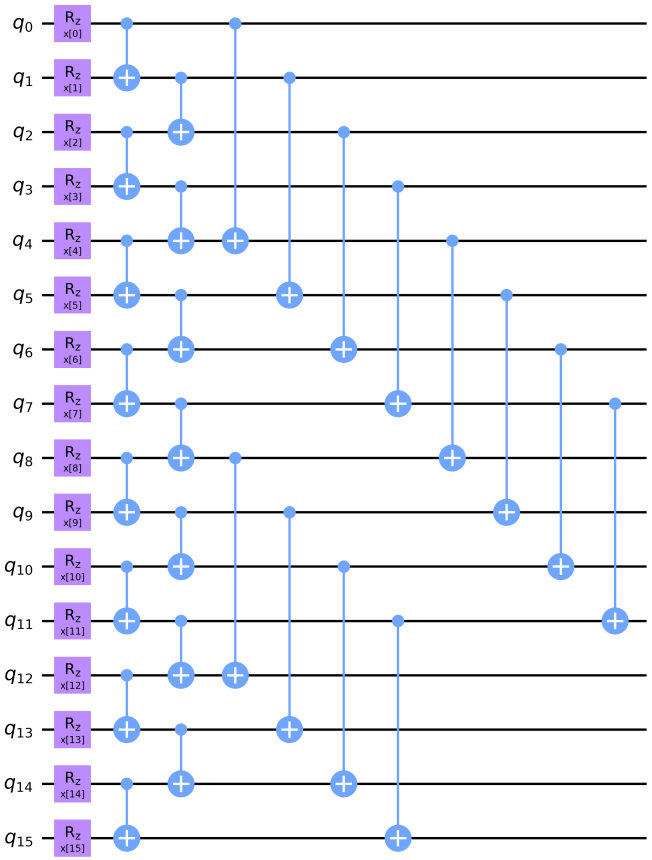
The implementation of QFT in ML

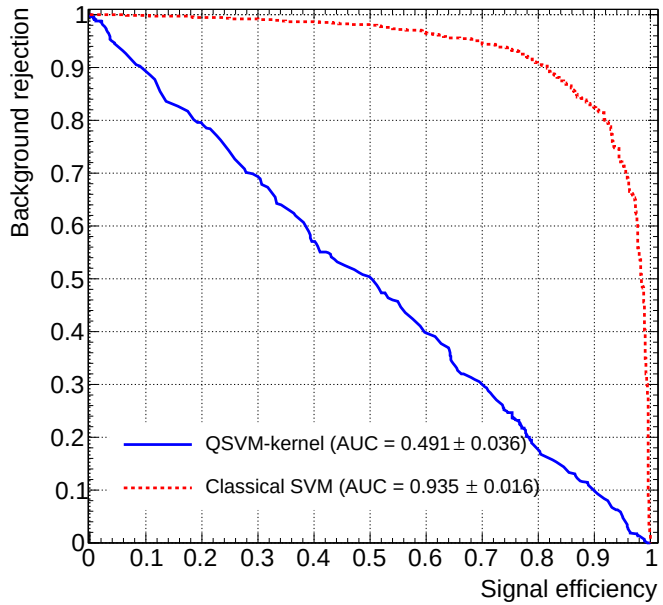
Checking the scalability of QFT with 16 qubits



- 12000 events with 16 qubits.

A new feature map





- 3000 events and 16 variables.

Summary

- Encoded QFT gives results comparable to the classical SVM with 6 qubits.
- And it's very well scalable to up to 15 qubits.
- There are a fewer parameters to optimise on such as $R_x(\vec{x}_i)$ and the two-qubit controlled rotation gate (θ)