Generalization in quantum machine learning from few training data

Generalization error of a QML model with T trainable gates scales at worst as

$$\sqrt{\frac{T}{N}}$$

$N \rightarrow$ number of training data points.

and for $K \ll T$, the generalization error improves to $\sqrt{\frac{K}{N}}$.

So, the compilation of unitaries into polynomial number of native gates which generally uses exponential size training data, can be sped up signif.

A good generalization is guranteed from few training data

There are theoretical results related to an upper bound on the generalization error as a function of the training data and model complexity.

Based on experiments, under certain circumstances, QML models may offer some advantage over classical models for classical data analysis.
and for quantum data these QML models may provide an exponential advantage in sample complexity for analyzing quantum data.

But, what is quantum data?
A collection of data that describes quantum systems and their evolution. Eg: Hamiltonian of the system, quantum state of syst., unitary transformation, projection operator

The training data size of Quantum Machine Learning generalization has yet to be fully studied.

Exponential number of data points are needed when training a function acting on an exponentially large Hilbert space.

Make use of well-known unitaries that can be represented by a polynomial-depth quantum circuit and are exponentially smaller than arbitrary unitaries.

Basically consider a QML model with T parameterized gates and relate the training data size N needed for generalization to T.
More general would be to consider generalization error a dynamic quantity that varies during the optimization.

Task in focus was a Quantum Convolutional Neural Network.

$$T \in \mathcal{O}(\log n)$$

This work guarantees that QCNNs have good generalization error for quantum phase recognition with only polylogarithmic training resources.

$$N \in \mathcal{O}(\log^2 N)$$

Side Note: CPTP → Complete Positive Trace Preserving Map
This is a mathematical operation that describes how a quantum system evolves under various transformation. This preserves the physical properties of a quantum state.

Results

Quantum Machine Learning Model (QMLM) → $\mathcal{E}_{\alpha}^{QMLM}(\cdot)$

where $\alpha = (\theta, k)$

- θ: continuous parameters (inside gate)
- k: Discrete parameters (allow gate structure to vary)

During optimization, one would vary/optimize the continuous parameters θ and potentially also the structure 'k' of the QMLM.

Input data $x \longmapsto \rho(x)$

We allow QMLM to act on a subsystem of state $\rho(x)$. Hence the output

becomes:

$$\left(\mathcal{E}_{\alpha}^{\text{OMLM}} \otimes \mathbb{I}\right)(\rho(x_i))$$

For a given data point (x_i, y_i) we can write the loss function as

$$\ell(\alpha : x_i, y_i) = \text{Tr}\left[O_{x_i,y_i}^{\text{loss}} \left(\mathcal{E}_{\alpha}^{\text{OMLM}} \otimes \mathbb{I}\right)(\rho(x_i)) \right] \quad \text{---} \quad (1)$$

for some Hermitian observable $O_{x_i y_i}^{\text{loss}}$

The prediction Error bounds will depend on the largest value that the loss function can attain

$$C_{\text{loss}} = \sup_{x,y} \| O_{x,y}^{\text{loss}} \| < \infty$$

i.e the spectral norm can be bounded uniformly over all possible loss observables.

↙ operator norm - a way to measure the size or magnitude of a matrix. For a square matrix, the spectral norm is the largest singular value of that matrix. It quantifies how much the matrix can stretch a vector without changing its direction.

$$\ell(\alpha : x_i, y_i) = \text{Tr}\left[O_{x_i,y_i}^{\text{loss}} \left(\mathcal{E}_{\alpha}^{\text{OMLM}} \otimes \mathbb{I}\right)(\rho(x_i)) \right]$$

in this equation we take measurement to act on a single copy of the output of the OMLM $\mathcal{E}_{\alpha}^{\text{OMLM}}(\cdot)$ upon input of the data encoding state $\rho(x_i)$.

For a training data set $S = \{(x_i, y_i)\}_{i=1}^{N}$, the average loss for parameters α on the training data is

$$\hat{R}_S(\alpha) = \frac{1}{N} \sum_{i=1}^{N} \ell(\alpha : x_i, y_i) \qquad \text{training error}$$

When we receive a new input x, the prediction error of a parameter setting α^* is

taken to be the expected loss.

$$R(\alpha) = \mathop{\mathbb{E}}_{(x,y)\sim P}\left[\ell(\alpha; x, y)\right]$$

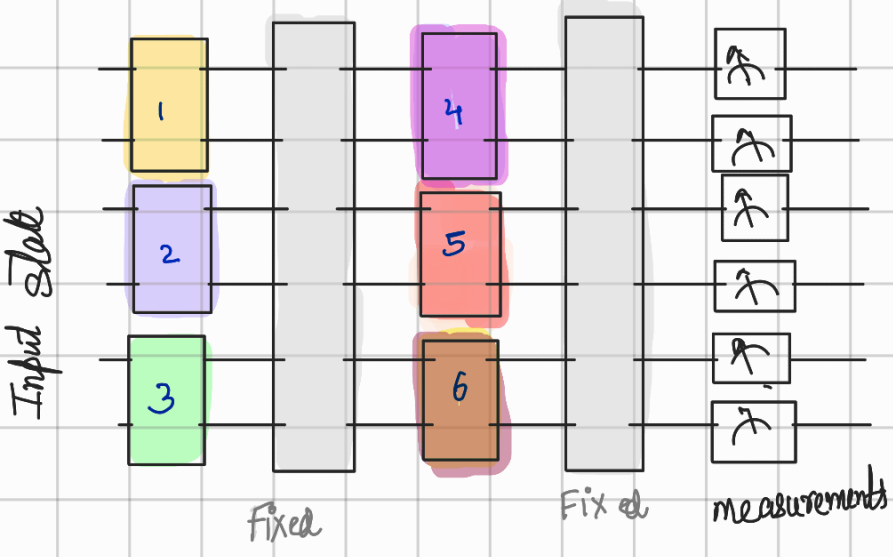
where the expectation is with respect to the distribution P from which the training samples are generated.

We want this $R(\alpha)$ to be small. ($\hat{R}_S(\alpha)$ as a proxy for $R(\alpha)$)

$$\text{gen}(\alpha) = R(\alpha) - \hat{R}_S(\alpha)$$

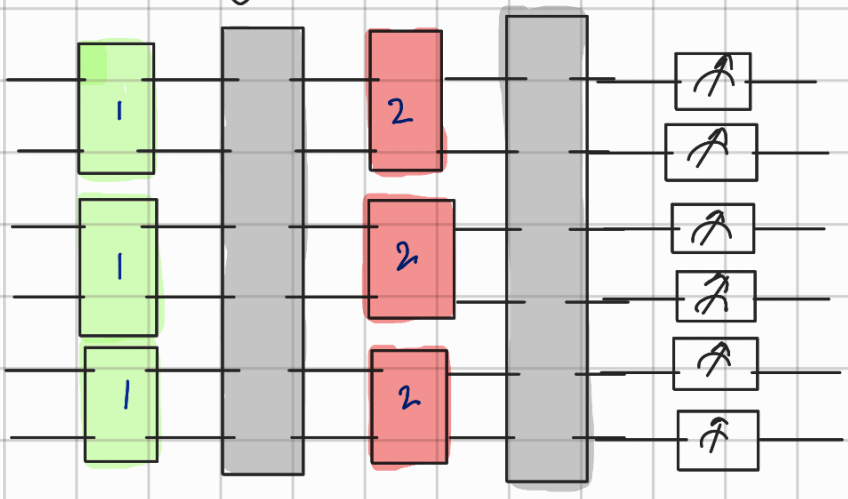
Analytical Result

a) Basic QMLM



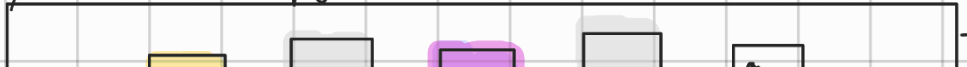
A QMLM with $T=6$ independently parameterized gates. The grey boxes are the global evolution that are not trainable.

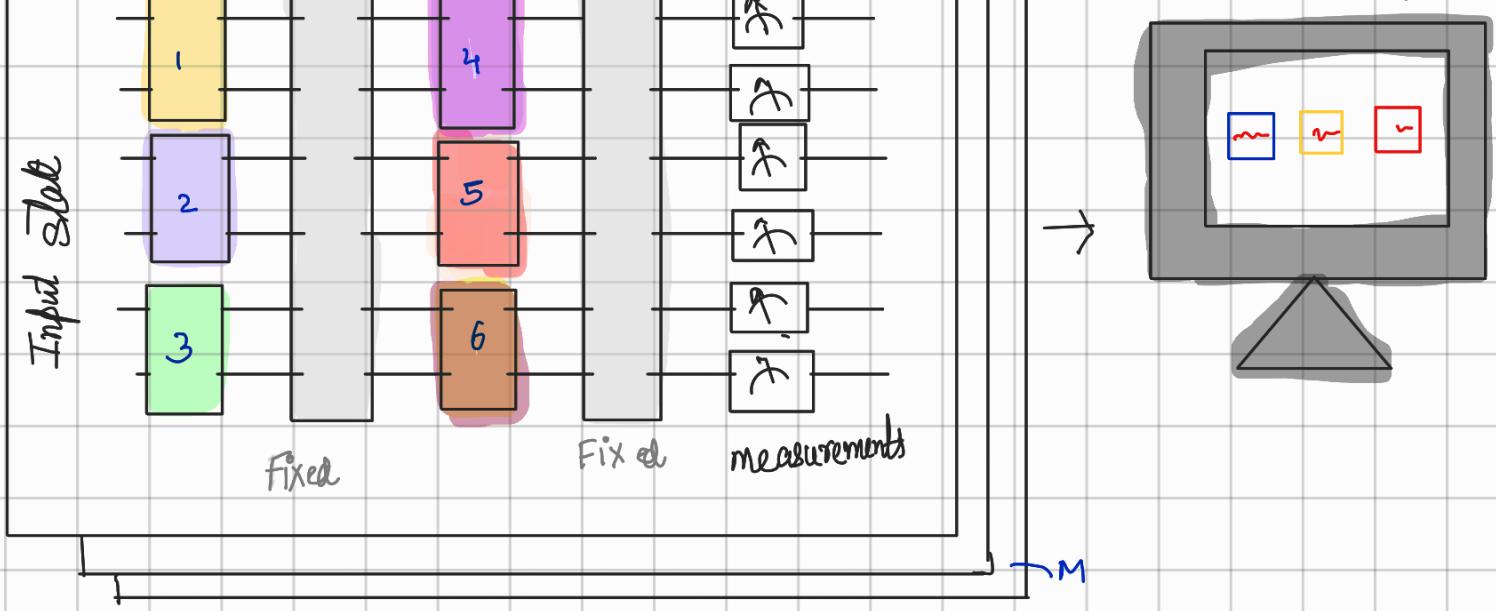
b) Gate sharing QMLM



A gate sharing QMLM with $T=2$ independently parameterized gates, each gate is repeatedly used for $M=3$ times

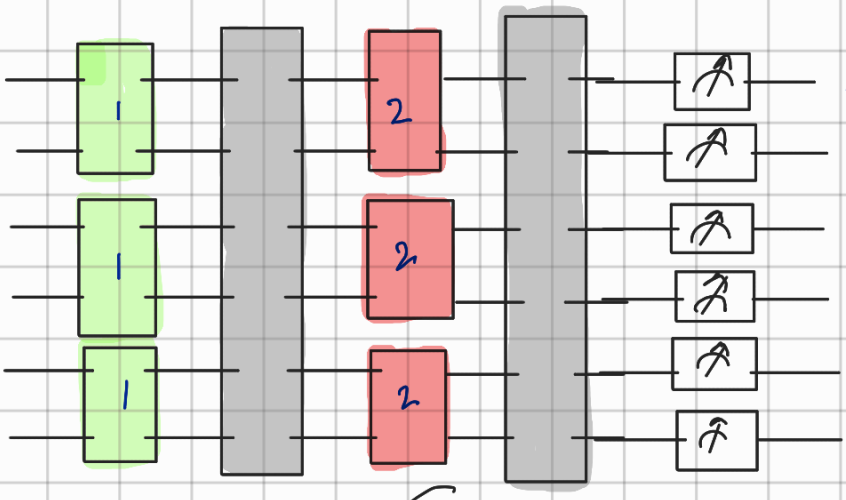
c) Multi copy QMLM





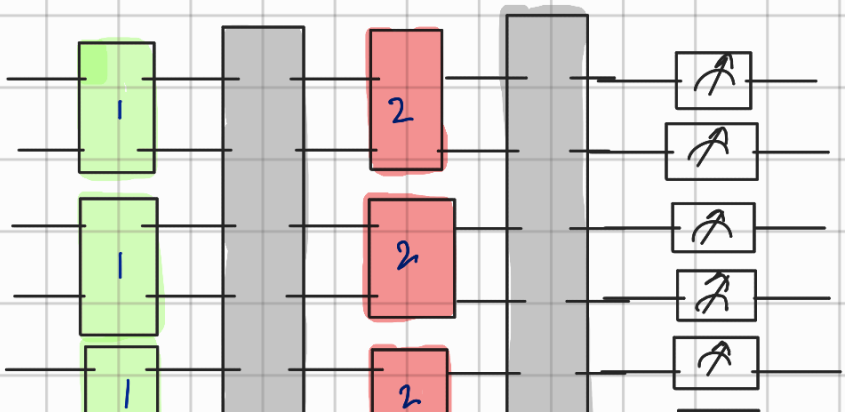
We take measurement data from M rounds of a basic QMLM with T=6 parameterized gates and post-process the measurement outcome to produce an output. Running M copies of a basic QMLM with T gates is equivalent to running a gate-sharing QMLM with T=6 parameterized gates, in which each gate is repeated M times.

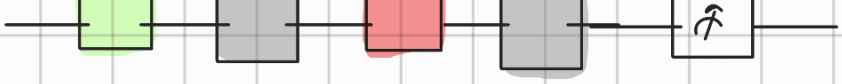
d) Gate sharing QMLM with optimization



The parameterized gates to the left undergoes a small change, while the one on the right undergoes a very large change.

optimization process





Theorem 1: For a QMLM with T parameterized local quantum channels, with a high probability over training data of size N, we have

$$\text{gen}(\alpha^*) \in \mathcal{O}\left(\sqrt{\frac{T \log T}{N}}\right)$$

For any $\varepsilon > 0$, we can, with high success probability, guarantee that $\text{gen}(\alpha^*) \leq \varepsilon$ already with training data of size $N \sim T \log T/\varepsilon^2$, which scales effectively linearly with T, the number of parameterized gates.