Innovating machine learning with near-term quantum computing

Maria Schuld

University of KwaZulu-Natal and Xanadu

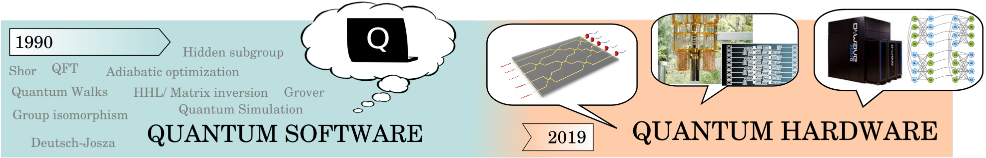
IPAM Workshop @ UCLA, November 2019

Agenda

- Motivation
- Mathematics of QML
- Variational quantum models
- Links to neural nets
- Links to kernel methods
- Open problems

MOTIVATION

Machine learning is important for the “quantum revolution”.



Machine learning is important for the "quantum revolution".

Players — Partners

Should AI researchers care about quantum computing?

How can quantum computers improve machine learning?

speed 

robustness 

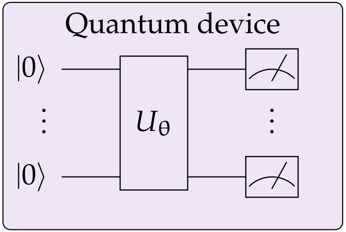
#data 

output 

The “first wave of QML” wanted to speed up ML.

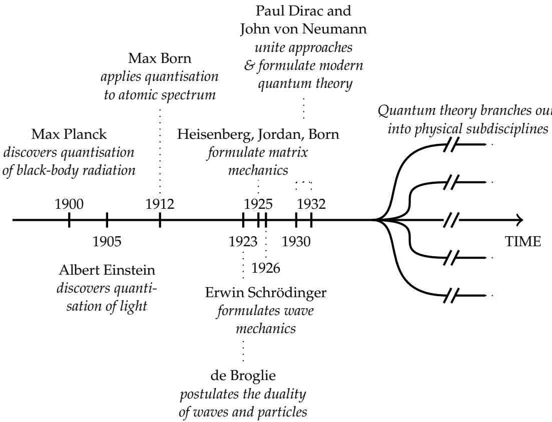
Problems	QIP tools	Applied to
Simulating linear algebra calculus with qubits		
matrix inversion, inner products, eigenvalue decomposition, singular value decomposition	quantum phase estimation, postselective amplitude update, Hamiltonian simulation, density matrix exponentiation	support vector machines, Gaussian processes, linear regression, discriminant analysis, recommendation systems, principal component analysis
Optimisation with Grover search		
finding closest neighbours, Markov chains	amplitude amplification, quantum walks	k-nearest neigbour, page ranking, clustering, associative memory, perceptrons, active learning agents, natural language processing
Sampling from quantum states		
sampling from model distribution	quantum annealing, quantum rejection sampling	Boltzmann machines, Bayesian nets, Bayesian inference
Optimisation with ground states of Hamiltonians		
combinatorial optimisation	adiabatic quantum computing, quantum annealing, quantum simulation	associative memory, boosting, debugging, variational Bayes inference, Bayesian networks, perceptron, EM algorithm, clustering

The "second wave of qml" trains quantum computations.



THE MATHEMATICS OF QML

Quantum theory is a math. framework invented in the 1930s.



Quantum theory calculates the expectations of measurements.

- A quantum state $|\psi\rangle$ lives in a **Hilbert space** $\mathcal{H}$ with scalar product $\langle\psi|\,\psi\rangle$.
- An **observable** is represented by a Hermitian operator O on $\mathcal{H}$. The eigenvectors of O form an orthonormal basis of $\mathcal{H}$ with real eigenvalues. Every $|\psi\rangle \in \mathbb{C}^N$ can hence be expressed in O's eigenbasis $\{|\psi_i\rangle\}_{i=1\ldots N}$, $|\psi\rangle = \sum_{i=1}^{N} a_i|\psi_i\rangle$, where the $a_i \in \mathbb{C}$ are the **amplitudes**.
- The effect of applying O to an element $|\psi\rangle \in \mathbb{C}^N$ is fully defined by the eigenvalue equations $O|\psi_i\rangle = \lambda_i|\psi_i\rangle$ with eigenvalues λ_i. **Expectation values** of the observable property are calculated by $\mathbb{E}(O) = \langle\psi|O|\psi\rangle$.
- The dynamic evolution of a quantum state is represented by a **unitary operator** $U = U(t_2, t_1)$ mapping $|\psi(t_1)\rangle$ to $U(t_2, t_1)|\psi(t_1)\rangle = |\psi(t_2)\rangle$ with $U^\dagger U = 1$. U is the solution of the corresponding **Schrödinger equation** $i\hbar\partial_t|\psi\rangle = H|\psi\rangle$ with **Hamiltonian** H.

From probabilities to amplitudes.

Consider a set of N measurement outcomes $\mathcal{X} = \{x_1, ..., x_N\}$ occurring with probability $p_1, ..., p_N$. The expectation of the measurement is given by:

$$\langle X \rangle = \sum_{i=1}^{N} p_i x_i = \vec{p}^T \vec{x}$$

From probabilities to amplitudes.

Consider

$$\vec{q} = \begin{pmatrix} \sqrt{p_1} \\ \vdots \\ \sqrt{p_N} \end{pmatrix} = \sqrt{p_1} \begin{pmatrix} 1 \\ \vdots \\ 0 \end{pmatrix} + \ldots + \sqrt{p_N} \begin{pmatrix} 0 \\ \vdots \\ 1 \end{pmatrix}, \qquad X = \begin{pmatrix} x_1 & \ldots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \ldots & x_N \end{pmatrix}.$$

The expectation value can now be written as

$$\langle X \rangle = \vec{q}^{\,T} X \vec{q} = \sum_{i=1}^{N} p_i x_i.$$

From probabilities to amplitudes.

Replace q with a complex *amplitude vector* $\psi = (\alpha_1, ..., \alpha_N)^T \in \mathbb{C}^N$.

Replace X by a complex, self-adjoint matrix $O \in \mathbb{C}^{N \times N}$.

The eigenvalues o_i of O correspond to the outcomes of measurements.

Unitary transformations describe evolutions.

Time evolutions of a quantum system are described by unitary transformations of the amplitude vector,

$$\begin{pmatrix} u_{11} & \dots & u_{1N} \\ \vdots & \ddots & \vdots \\ u_{N1} & \dots & u_{NN} \end{pmatrix} \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_N \end{pmatrix} = \begin{pmatrix} \alpha_1' \\ \vdots \\ \alpha_N' \end{pmatrix}, \qquad \sum_{i=1}^{N} |\alpha_i|^2 = \sum_{i=1}^{N} |\alpha_i'|^2 = 1.$$

Quantum computing is a special case.

Quantum system $\rightarrow n$ qubits

Measurement outcomes $\rightarrow X = \{00...0, ..., 11...1\}$.

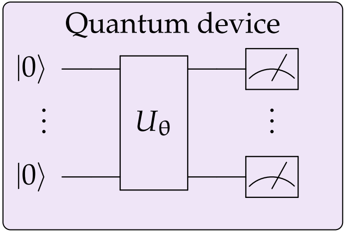
States $\rightarrow \psi = (\alpha_1, ..., \alpha_{2^n})^T$

Evolution $\rightarrow U = G_L, ..., G_1$

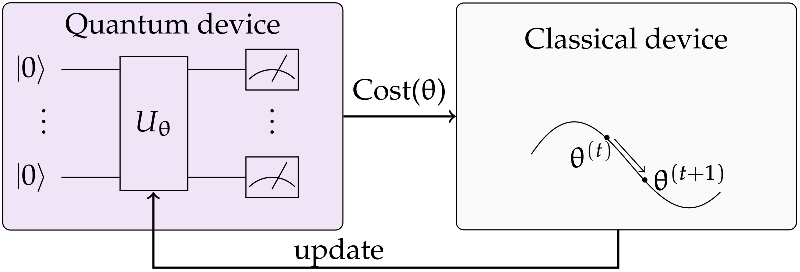
Expectation of measuring first qubit $\rightarrow \psi^\dagger (\sigma_z \otimes \mathbb{1}^{(n-1)})\psi$

VAR. QUANTUM MODELS

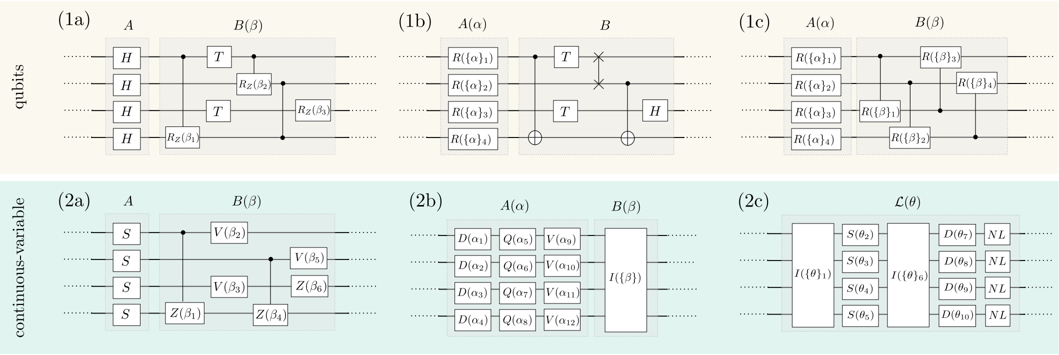
Variational models are trainable circuits.



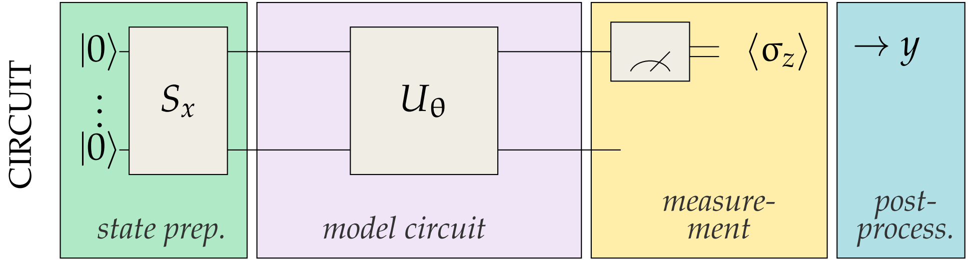
Variational models are trainable circuits.



Variational models are trainable circuits.

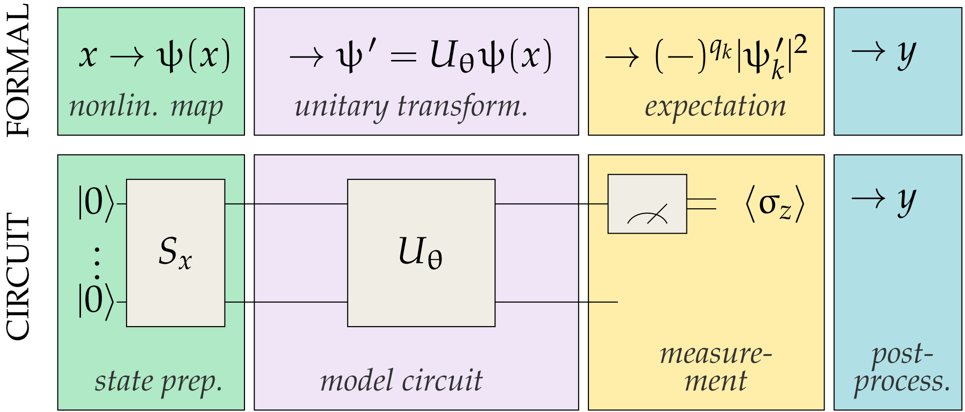


Variational models consist of three elements.



Farhi & Neven 1802.06002, Schuld et al. 1804.00633, Benedetti et al. 1906.07682

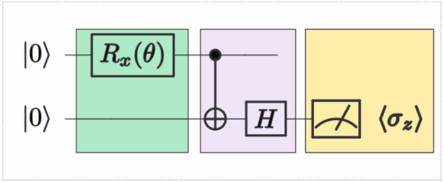
Variational models consist of three elements.

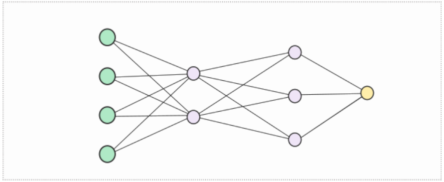


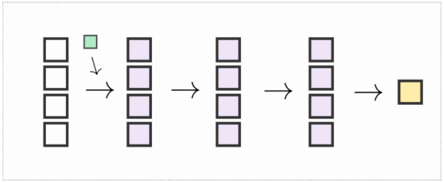
Farhi & Neven 1802.06002, Schuld et al. 1804.00633, Benedetti et al. 1906.07682

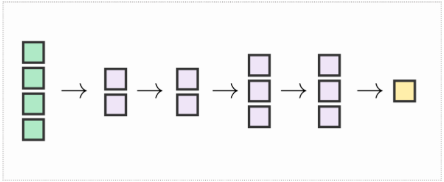
LINKS TO NEURAL NETS

Variational models are linear, symmetric neural networks.









Variational models are linear, symmetric neural networks.

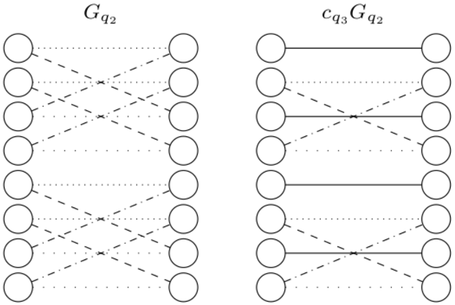
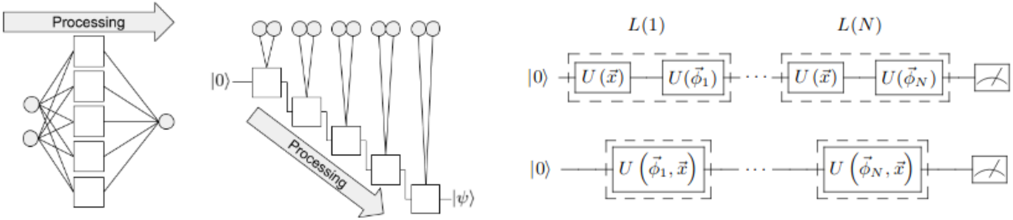


Fig. 8.1 The single qubit gate G_{q_2} (left) and the controlled single qubit gate $c_{q_3}G_{q_2}$ (right) from the examples applied to a system of 3 qubits drawn in the graphical representation of neural networks. The gates take a quantum state with amplitude

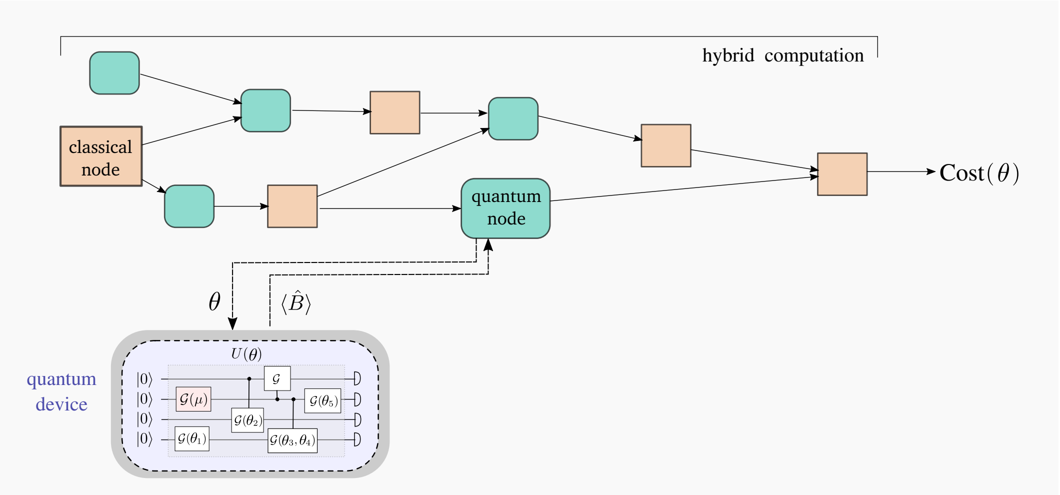
Schuld & Petruccione, Springer 2018

Variational models are linear, symmetric neural networks.



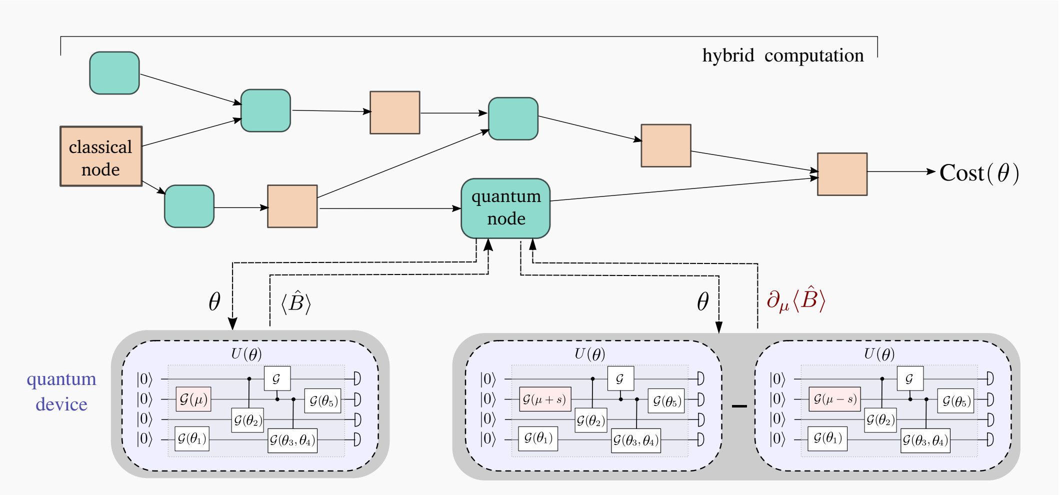
Pérez-Salinas et al. 1907.02085

We can do gradient descent on variational models.



Guerreschi & Smelyanskiy 1701.01450, Mitarai et al. 1803.00745, Schuld et al. 1811.11184

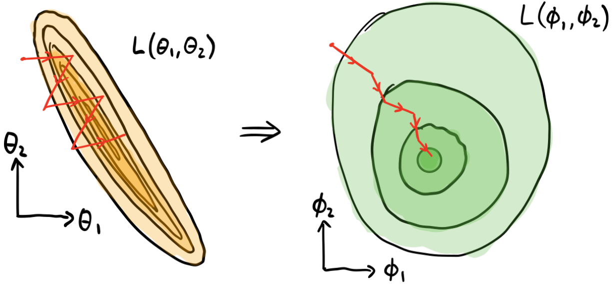
We can do gradient descent on variational models.



Guerreschi & Smelyanskiy 1701.01450, Mitarai et al. 1803.00745, Schuld et al. 1811.11184

We can do gradient descent on variational models.

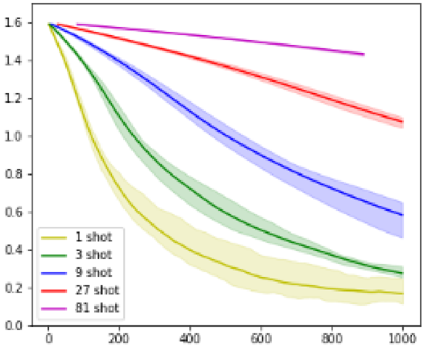
Quantum Natural Gradient.

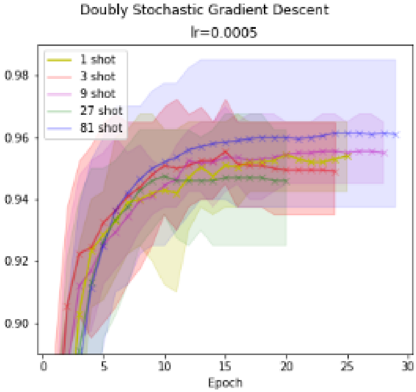


Stokes, Izaac, Killoran, Carleo 1909.02108

We can do gradient descent on variational models.

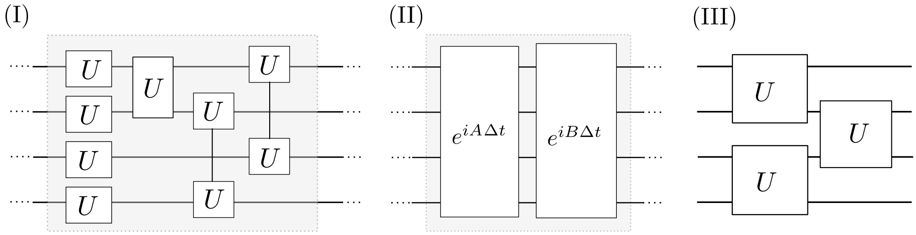
Stochastic gradient descent.



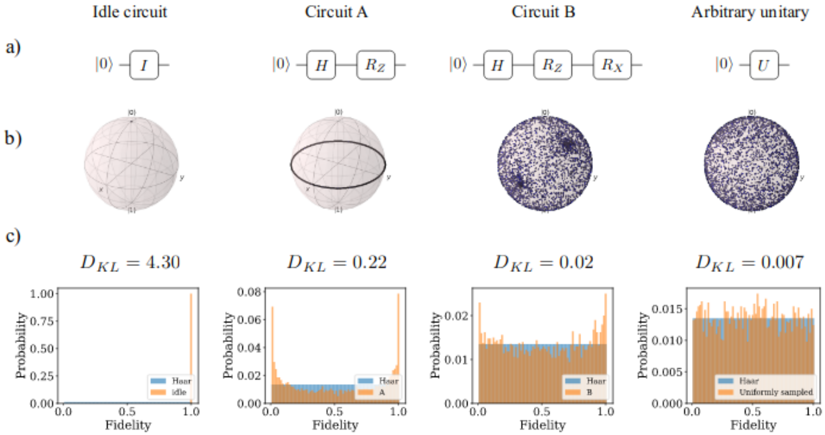


Sweke et al 1910.01155

We are investigating the expressivity of variational models.

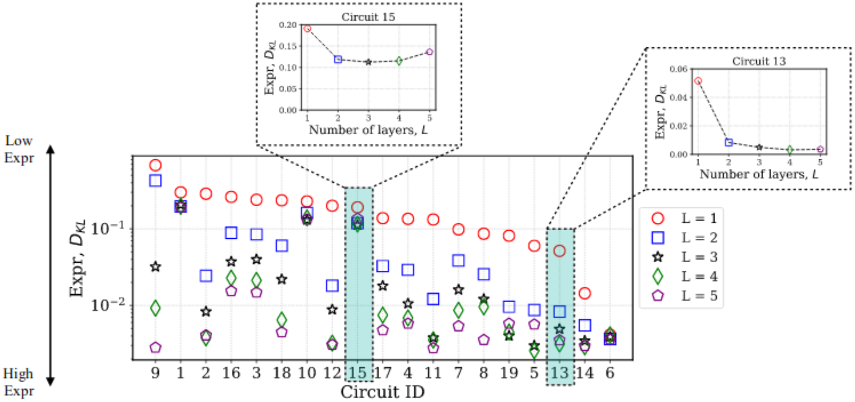


We are investigating the expressivity of variational models.



Sim, Johnson, Aspuru-Guzik 1905.10876

We are investigating the expressivity of variational models.



Sim, Johnson, Aspuru-Guzik 1905.10876

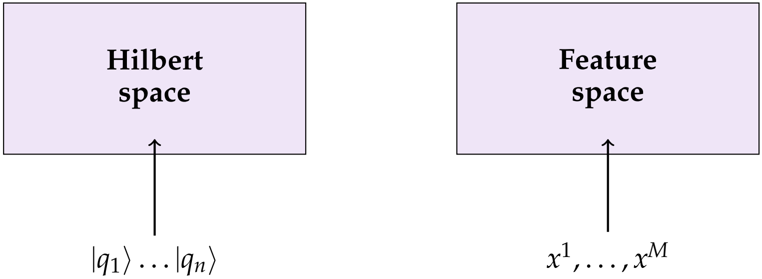
We are investigating the expressivity of variational models.

Theorem 3. *The expressive power of MPQCs and TPQCs with $O(poly(N))$ single qubits gates and CNOT gates, and classical neural networks with $O(poly(N))$ trainable parameters, where N refers to the number of qubits or the visible units, can be ordered as: MPQCs $>$ DBM $>$ long range RBM $>$ TPQCs $>$ short range RBM.*

Du, Hsieh, Liu, Tao, 1810.11922

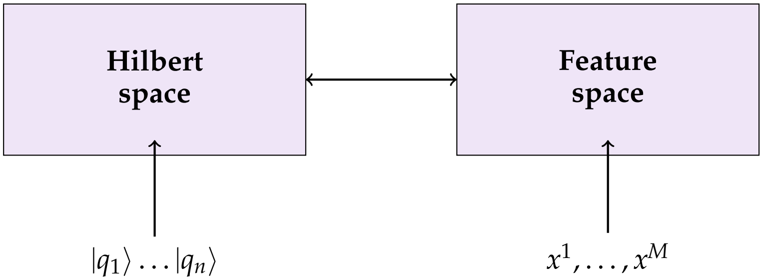
LINKS TO KERNELS

QC and kernel methods share the same basic idea.



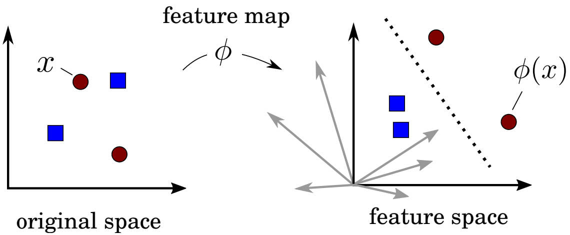
Schuld & Killoran 1803.07128, Havlicek et al. 1804.11326

QC and kernel methods share the same basic idea.



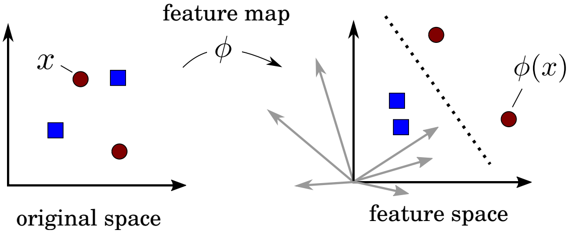
Schuld & Killoran 1803.07128, Havlicek et al. 1804.11326

Kernel methods use inner products in Hilbert space.



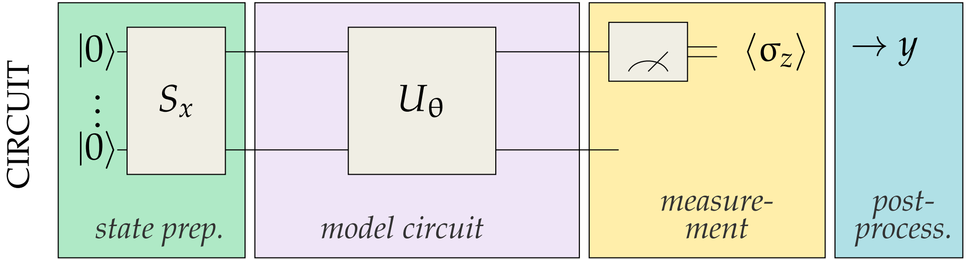
$$\kappa(x, x') = \langle \phi(x), \phi(x') \rangle$$

Kernel methods use inner products in Hilbert space.

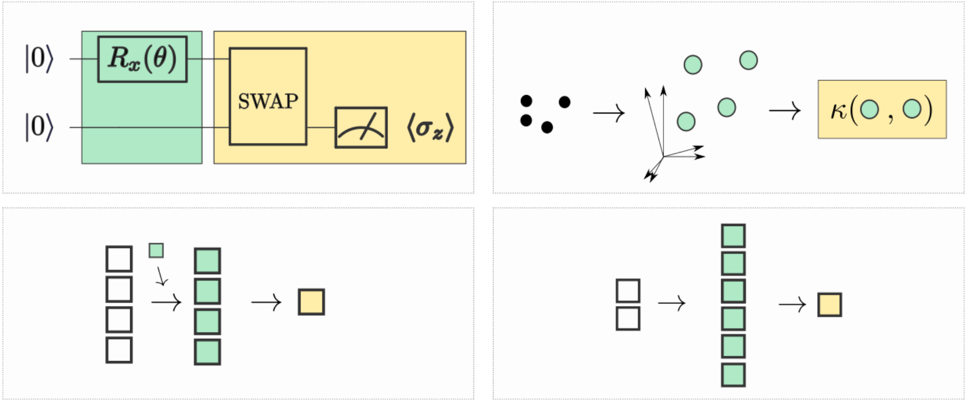


$$\kappa(x, x') = |\langle \phi(x) | \phi(x') \rangle|^2$$

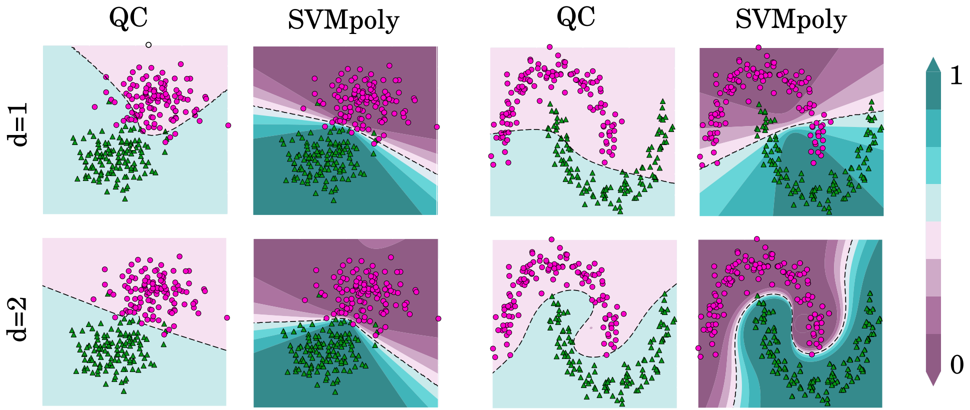
Variational QML models are similar to kernel methods.



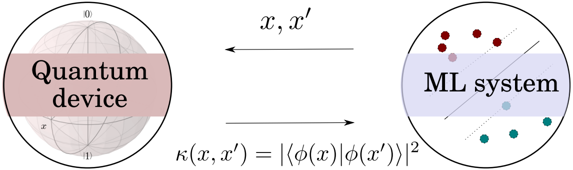
Variational QML models are similar to kernel methods.



Variational QML models are similar to kernel methods.

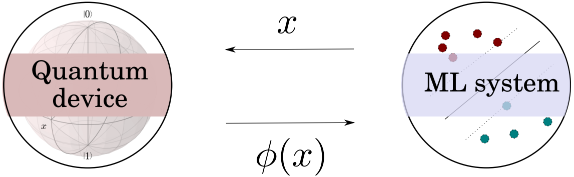


Variational QML models are similar to kernel methods.



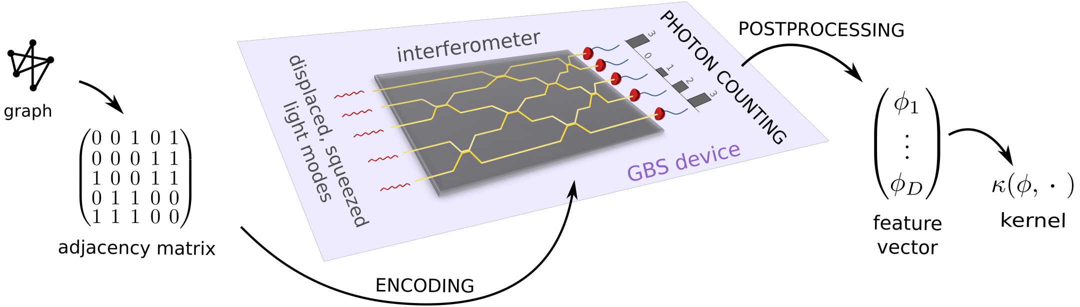
Schuld & Killoran 1803.07128, Havlicek et al. 1804.11326

Variational QML models are similar to kernel methods.



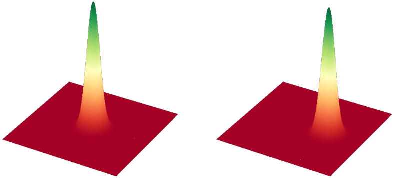
EXAMPLE OF A QUANTUM EMBEDDING

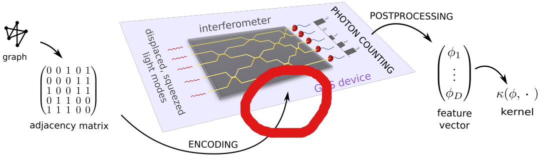
A Gaussian Boson Sampler can help to compute kernels.



Schuld, Bradler, Israel, Su, Gupt 1905.12646

The covariance matrix encodes the graph.





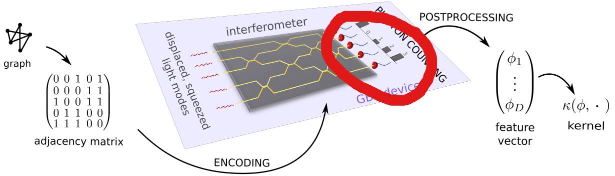
A Gaussian state of M optical modes is fully described by a *covariance matrix* $\sigma \in \mathbb{R}^{2M \times 2M}$ as well as a *displacement vector* $d \in \mathbb{R}^{2M}$.

We can associate such a state with an adjacency matrix A via

$$\sigma = (\mathbb{1} - X\tilde{A})^{-1} - \frac{\mathbb{1}}{2}, \text{ with } X = \begin{pmatrix} 0 & \mathbb{1} \\ \mathbb{1} & 0 \end{pmatrix}, \tilde{A} = c \begin{pmatrix} A & 0 \\ 0 & A \end{pmatrix}.$$

The features are probabilities of detecting certain photon events.

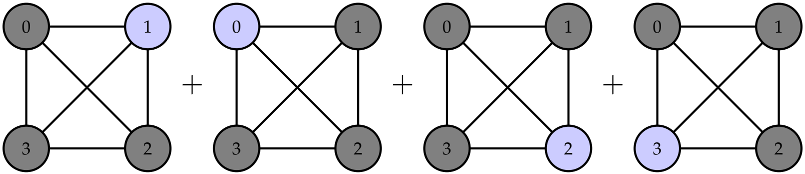
$p(O_{[000000]})$
$p(O_{[100000]})$
$p(O_{[200000]})$
$p(O_{[110000]})$
$p(O_{[300000]})$
$p(O_{[210000]})$
$p(O_{[111000]})$
$p(O_{[400000]})$
$p(O_{[220000]})$
$p(O_{[310000]})$
$p(O_{[211000]})$
$p(O_{[111100]})$
...



An *orbit* is a set of photon click events which are permutations of each other.

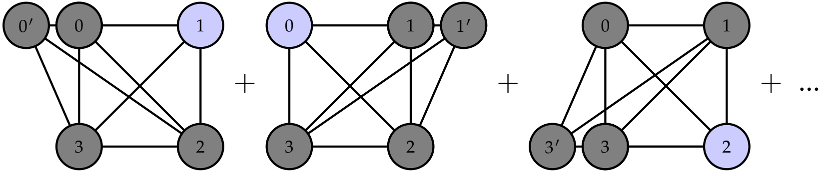
i.e., $O_{[111100]} = \{[111100], [111010], [010111], [100111], [001111], ...\}$

Orbits count the number of r-matchings in subgraphs.



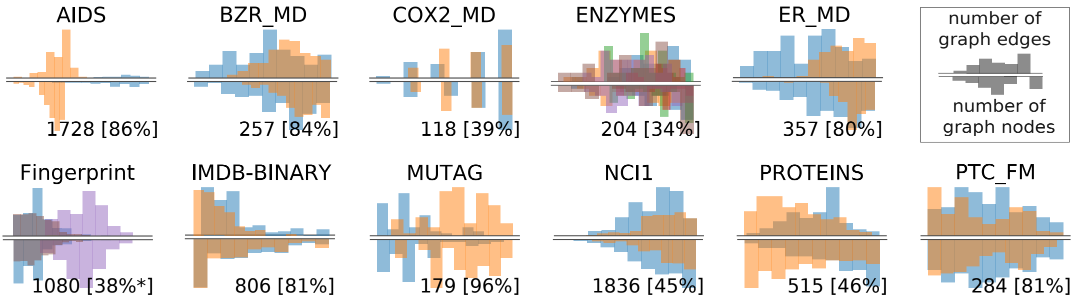
$$p(O_{[1,1,1,0]}) = p([1,1,1,0]) + p([1,1,0,1]) + p([1,0,1,1]) + p([0,1,1,1])$$

Orbits count the number of r-matchings in subgraphs.



$$p(O_{[2,1,1,0]}) = p([2,1,1,0]) + p([1,2,0,1]) + p([1,0,1,2]) + \dots$$

We can test the kernel on standard graph data sets.



The results are promising.

Dataset	GBS ($d = 0.0$)	GBS ($d = 0.25$)	GS	RW	SM
AIDS	99.60 ± 0.05	**99.62** ± 0.03	98.44 ± 0.09	56.95 ± 7.99	79.20 ± 0.68
BZR_MD	**62.73** ± 0.71	62.13 ± 1.44	60.60 ± 1.77	49.88 ± 3.74	61.90 ± 1.21
COX2_MD	44.98 ± 1.80	50.11 ± 0.97	55.04 ± 3.33	57.72 ± 3.26	**66.94** ± 1.22
ENZYMES	22.29 ± 1.60	28.01 ± 1.83	35.87 ± 2.19	21.13 ± 1.91	**36.70** ± 2.83
ER_MD	70.36 ± 0.78	**70.41** ± 0.47	65.65 ± 1.06	68.75 ± 0.53	68.21 ± 0.99
FINGERPRINT	65.42 ± 0.49	**65.85** ± 0.36	64.10 ± 1.52	47.69 ± 0.21	47.14 ± 0.62
IMDB-BIN	64.09 ± 0.34	68.71 ± 0.59	68.37 ± 0.62	66.38 ± 0.21	out of time*
MUTAG	**86.41** ± 0.33	85.58 ± 0.59	81.08 ± 0.93	83.02 ± 1.08	83.14 ± 0.24
NCI1	**63.61** ± 0.00	62.79 ± 0.00	49.96 ± 3.27	52.36 ± 2.63	51.36 ± 1.88
PROTEINS	**66.88** ± 0.22	66.14 ± 0.48	65.91 ± 1.29	56.27 ± 1.23	63.03 ± 0.84
PTC_FM	53.84 ± 0.96	52.45 ± 1.78	**59.48** ± 1.95	51.97 ± 2.68	54.92 ± 2.94

OPEN PROBLEMS

This is only the beginning of the journey..

- How can we benchmark quantum models?

This is only the beginning of the journey..

- How can we benchmark quantum models?
- Is there a connection between quantum theory and deep learning?

This is only the beginning of the journey..

- How can we benchmark quantum models?
- Is there a connection between quantum theory and deep learning?
- Do quantum models inherently regularise?

This is only the beginning of the journey..

- How can we benchmark quantum models?
- Is there a connection between quantum theory and deep learning?
- Do quantum models inherently regularise?
- How does noise impact applications?

This is only the beginning of the journey..

- How can we benchmark quantum models?
- Is there a connection between quantum theory and deep learning?
- Do quantum models inherently regularise?
- How does noise impact applications?
- What domains is QML good for?

This is only the beginning of the journey..

- How can we benchmark quantum models?
- Is there a connection between quantum theory and deep learning?
- Do quantum models inherently regularise?
- How does noise impact applications?
- What domains is QML good for?
- Do QML ideas scale?

This is only the beginning of the journey..

- How can we benchmark quantum models?
- Is there a connection between quantum theory and deep learning?
- Do quantum models inherently regularise?
- How does noise impact applications?
- What domains is QML good for?
- Do QML ideas scale?
- What optimization strategies work for variational circuits?

This is only the beginning of the journey..

- How can we benchmark quantum models?
- Is there a connection between quantum theory and deep learning?
- Do quantum models inherently regularise?
- How does noise impact applications?
- What domains is QML good for?
- Do QML ideas scale?
- What optimization strategies work for variational circuits?
- Which circuit architectures are good for ML?

Thank you!

www.pennylane.ai
www.xanadu.ai
@XanaduAI