Data re-uploading for a Universal quantum classifier

Alba Cervera-Lierta
University of Toronto

QTML 2020

the matter lab

Outlook

From classical to quantum NN

Single-qubit classifier

Universality

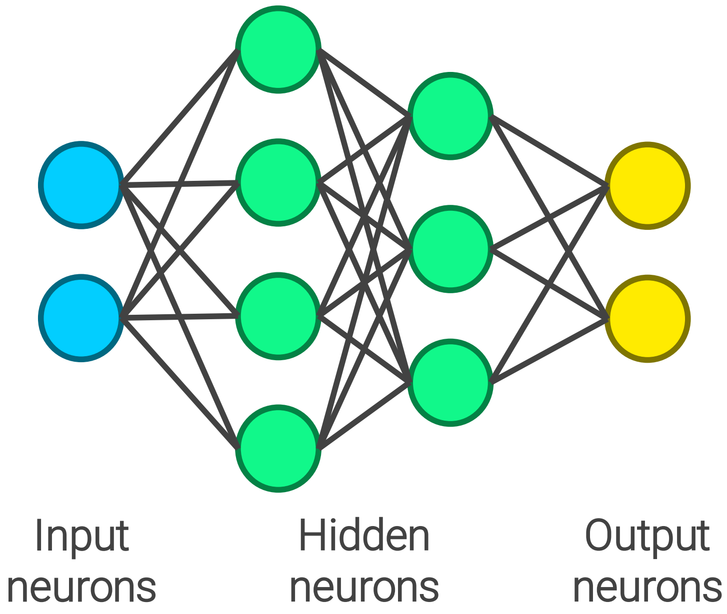
Multi-qubit classifier

Benchmarks

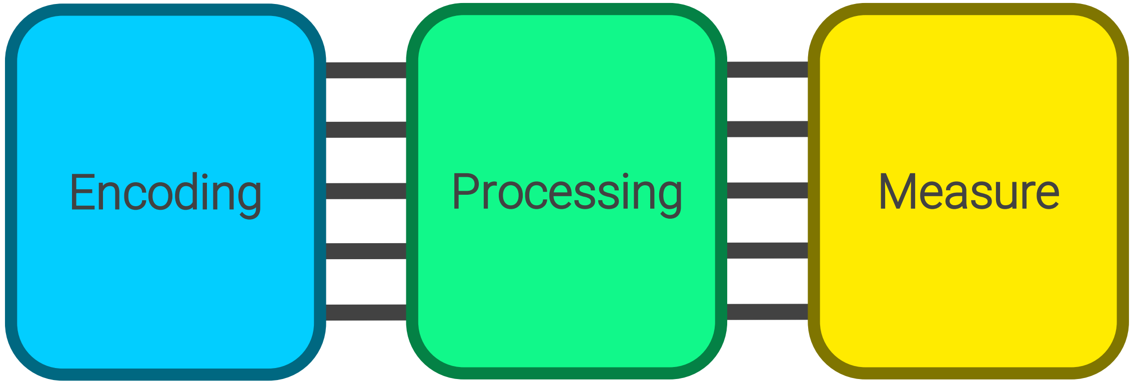
Conclusions and remarks

From classical to quantum NN

Classical



Quantum
(circuit centric)



K Mitarai, M Negoro, M Kitagawa, K Fujii Phys. Revs A 98 (3), 032309 (2018)

E. Farhi and H.Neven, arXiv:1802.06002 (2018)

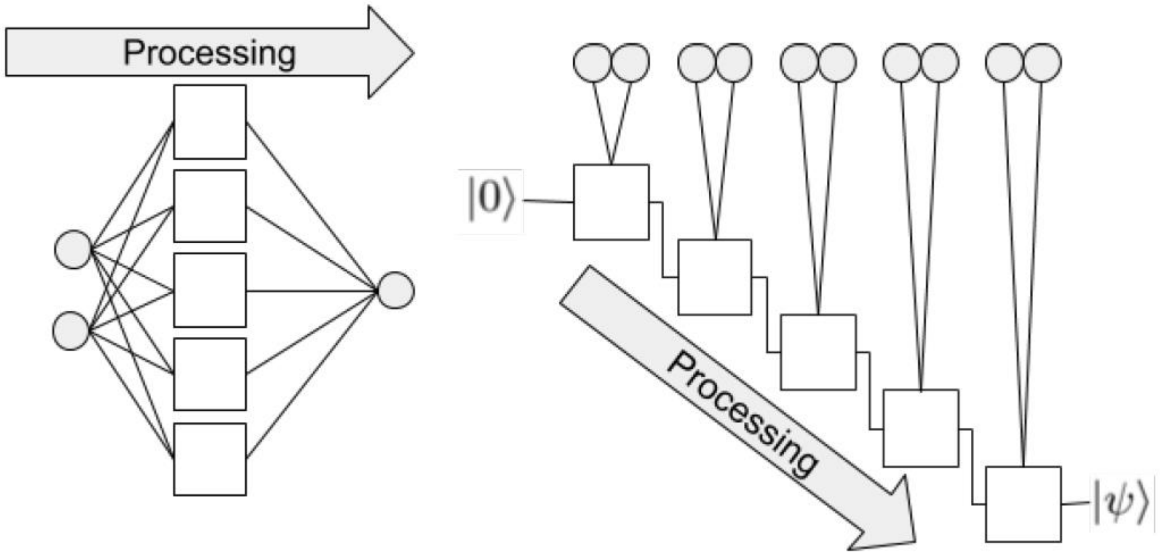
M. Schuld and N. Killoran, Phys. Rev. Lett. 122, 040504 (2019)

M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, Phys. Rev. A 101, 032308 (2020)

The minimal QNN

What is the most simple (but universal) NN? → Single hidden layer NN

What is the most simple (but universal) QNN? → Single-qubit QNN



(a) Neural network (b) Quantum classifier

A. Pérez-Salinas, ACL,, E. Gil-Fuster and J. I. Latorre, Quantum 4, 226 (2020), arXiv:1907.02085

Single-qubit quantum classifier

What are the power and minimal needs of a quantum circuit to carry out a general supervised classification task?

- Qubits
- Operations
- Parameters

1 qubit is enough *if* data is re-uploaded along the circuit *and if* assisted with a classical optimization subroutine.

Encoding the data

A product of free single-qubit unitaries can be written with another single-qubit unitary

Insufficient to carry out any non-trivial task

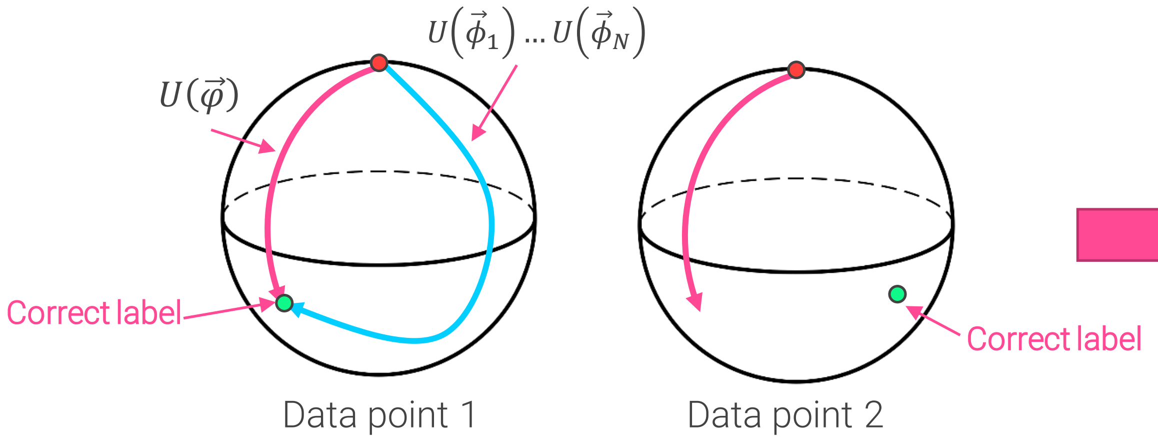
$$U(\vec{\phi}_1)\dots U(\vec{\phi}_N) \equiv U(\vec{\varphi})$$

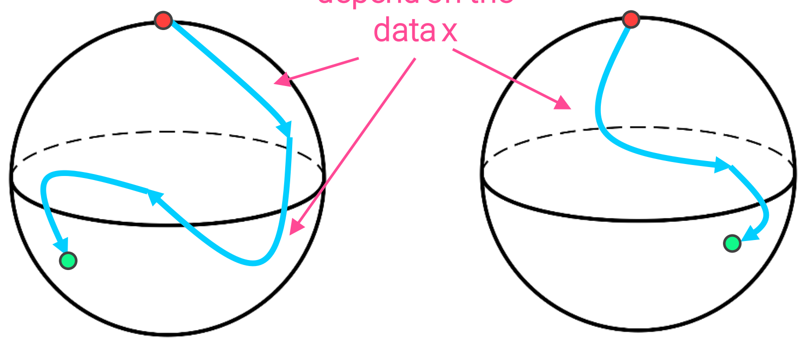
If we add some fixed parameter dependency (the data), the operation becomes flexible and data-depedent.

Data re-uploading

$$\mathcal{U}(\vec{\phi},\vec{x}) \equiv U(\vec{\phi}_N)U(\vec{x})\dots U(\vec{\phi}_1)U(\vec{x})$$

The paths depend on the data x

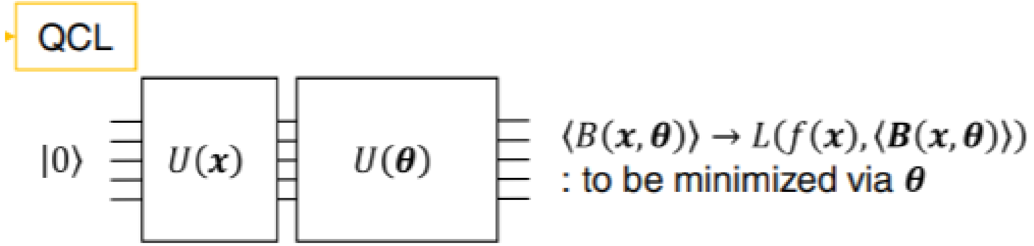


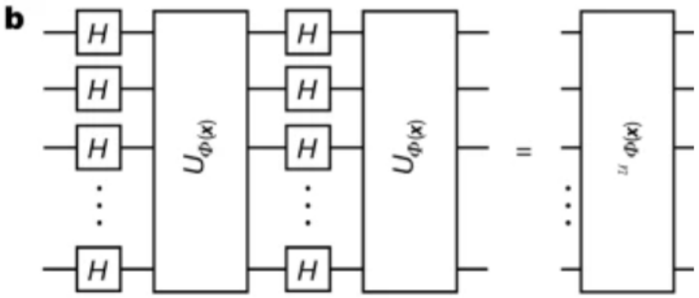


“data re-uploading” in other works

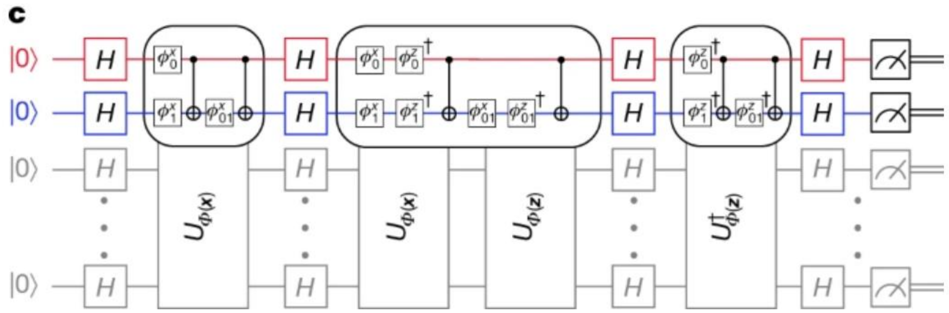
The idea of encoding data in more than one circuit operation is not new.
However, the motivation and methodology varies from one proposal to another.

Circuit-centric classification:
construct a classically hard feature map





Kernel methods:
Construct the Kernel to measure it.



K Mitarai, M Negoro, M Kitagawa, K Fujii Phys. Revs A 98 (3), 032309 (2018).
M. Schuld, N. KilloranPhys. Rev. Lett. 122, 040504 (2019).
Vojtěch Havlíček et. al. Nature 567, 209 (2019).

Data re-uploading layers

The total unitary is divided into layers.
Each layer encodes the data.

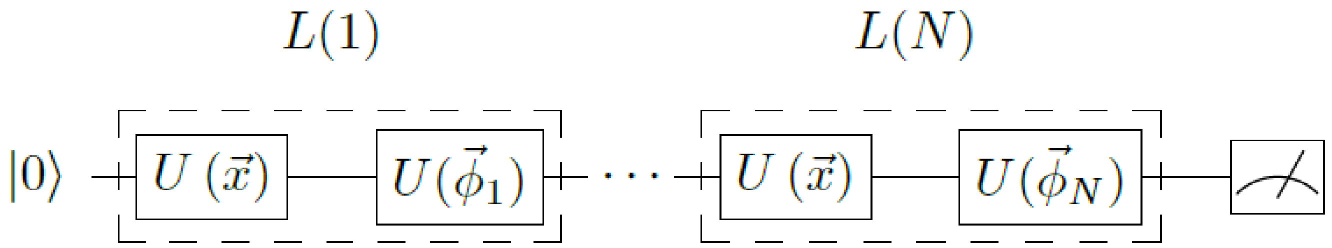
$$\mathcal{U}(\vec{\phi},\vec{x}) = L(N)\dots L(1)$$

$$L(i) \equiv U(\vec{\phi}_i)U(\vec{x})$$

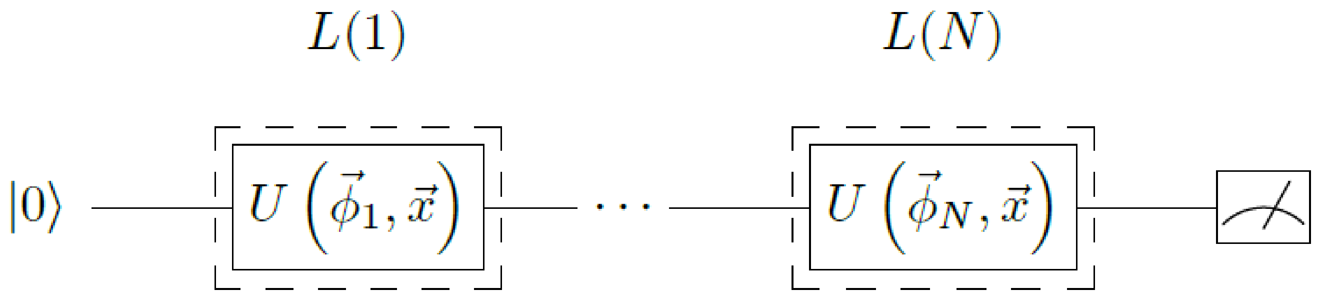
$$\Big\downarrow \text{Single operation}$$

$$L(i) = U\left(\vec{\theta}_i + \vec{w}_i \circ \vec{x}\right)$$

Why this particular encoding?

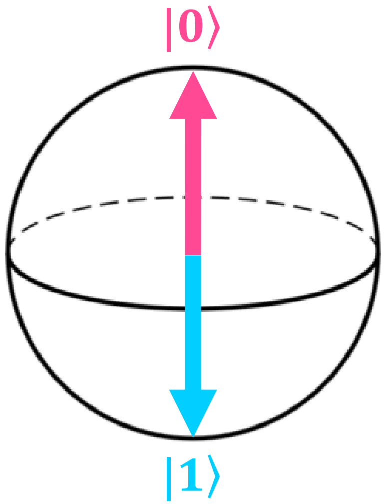


(a) Original scheme



(b) Compressed scheme

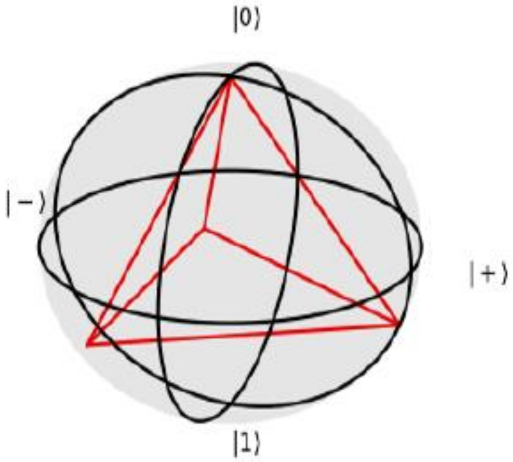
Target states



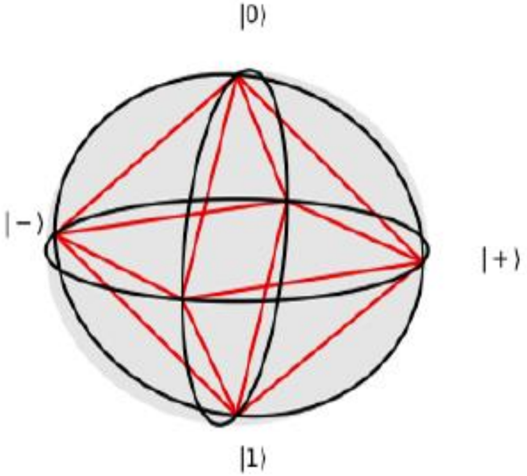
Convenient: choose the most ortogonal states to define each target state.

Single-qubit → Divide the Bloch sphere into Nclass sections

C. W. Helstrom, *Quantum detection and estimation theory*, Academic Press New York (1976).



4 clases



8 classes

Extension for multi-qubits:
S. Lloyd, M. Schuld, A. Ijaz, J. Izaac, N. Killoran, arXiv:2001.03622

Measurement and cost function

Target state: one for each label/class.

Compute the fidelity (overlap) between the quantum circuit state and the target state

Training points

Circuit state wavefunction

$$\chi_f^2(\vec{\theta}, \vec{w}) = \sum_{\mu=1}^{M} \left(1 - |\langle \tilde{\psi}_s | \psi(\vec{\theta}, \vec{w}, \vec{x}_\mu) \rangle|^2\right)$$

Target state wavefunction

Weighted fidelity (for multiclassification):
compute the overlap w.r.t. target state – distance w.r.t. other class target state

classes

$$\chi_{wf}^2(\vec{\alpha}, \vec{\theta}, \vec{w}) = \frac{1}{2} \sum_{\mu=1}^{M} \left(\sum_{c=1}^{\mathcal{C}} \left(\alpha_c F_c(\vec{\theta}, \vec{w}, \vec{x}_\mu) - Y_c(\vec{x}_\mu) \right)^2 \right)$$

Universality

$$L(i) = U\left(\vec{\theta}_i + \vec{w}_i \circ \vec{x}\right)$$

Why this particular encoding?

The choose of this encoding allows us to connect the classifier with the Universality proof.

Universal Approximation Theorem

activation function

\# neurons

Any continous function $f(x)$ can be approximated with ϵ accuracy by the function

$$h(\vec{x}) = \sum_{i=1}^{N} \alpha_i \varphi(\vec{w}_i \cdot \vec{x} + b_i)$$

$$\alpha_i, b_i \in \mathbb{R} \quad \vec{w}_i \in \mathbb{R}^m$$

output weigths

weigths

biases

where φ is a nonconstant, bounded and continuous function.

A single-layer neural network can approximate any continous function
(providing enough neurons in the hidden layer)

Universal Quantum Circuit approximation

Single-qubit quantum gate = SU(2) operator:

$$U(\vec{\phi}) = e^{i\vec{\omega}(\vec{\phi})\cdot\vec{\sigma}}$$

generators

Linear encoding

$$\vec{\phi}(\vec{x}) = (\phi_1(\vec{x}), \phi_2(\vec{x}), \phi_3(\vec{x})) = \vec{\theta} + \vec{w}\circ\vec{x}.$$

Multiple products of SU(2) operators are also a SU(2) operator

$$\mathcal{U}(\vec{x}) = U_N(\vec{x})U_{N-1}(\vec{x})\cdots U_1(\vec{x}) = \prod_{i=1}^{N} e^{i\vec{\omega}(\vec{\phi}_i(\vec{x}))\cdot\vec{\sigma}}$$

Circuit layers

Continous, bounded, nonconstant

$$\omega_1(\vec{\phi}) = d\,\mathcal{N}\sin((\phi_2-\phi_3)/2)\sin(\phi_1/2)$$

$$\left(\sqrt{1-\cos^2 d}\right)^{-1}$$

$$\cos d = \cos((\phi_2+\phi_3)/2)\cos(\phi_1/2)$$

Applying the BCH formula:

$$\mathcal{O}_{corr} = \vec{\varrho}(\vec{x})\cdot\vec{\sigma}$$

$$\mathcal{U}(\vec{x}) = \exp\left[i\sum_{i=1}^{N}\vec{\omega}(\vec{\phi}_i(\vec{x}))\cdot\vec{\sigma} + \mathcal{O}_{corr}\right] = e^{i\vec{f}(\vec{x})\cdot\vec{\sigma}+i\vec{\varrho}(\vec{x})\cdot\vec{\sigma}}$$

$$\left(\omega_1(\vec{\theta}_i + \vec{w}_i\circ\vec{x}), \omega_2(\vec{\theta}_i + \vec{w}_i\circ\vec{x}), \omega_3(\vec{\theta}_i + \vec{w}_i\circ\vec{x})\right)$$

$$= (f_1(\vec{x}), f_2(\vec{x}), f_3(\vec{x}))$$

Continous functions

A. Pérez-Salinas, ACL,, E. Gil-Fuster and J. I. Latorre, Quantum 4, 226 (2020), arXiv:1907.02085

Multi-qubit quantum classifier

A single-qubit quantum classifier can be simulated classically.

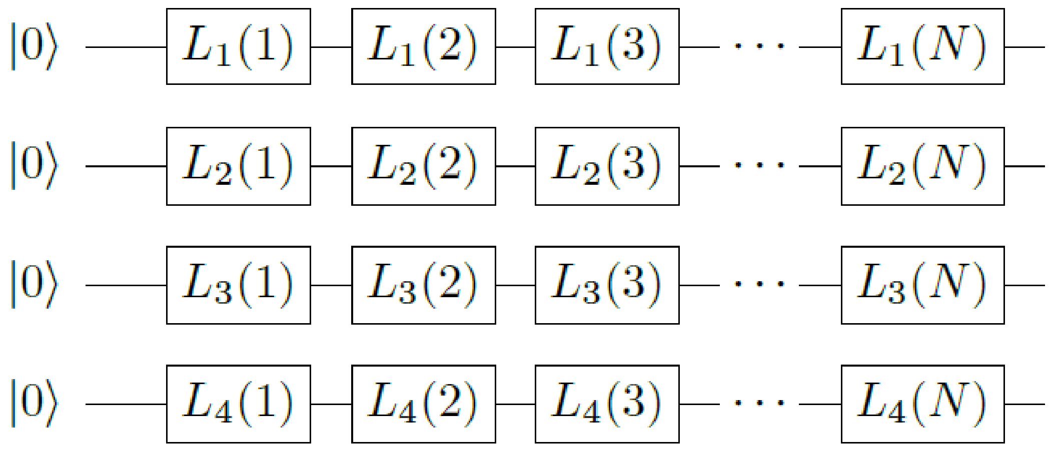
We need to introduce entanglement (therefore, more qubits) to eventually prove any quantum advantage.

Multi-qubit quantum classifier

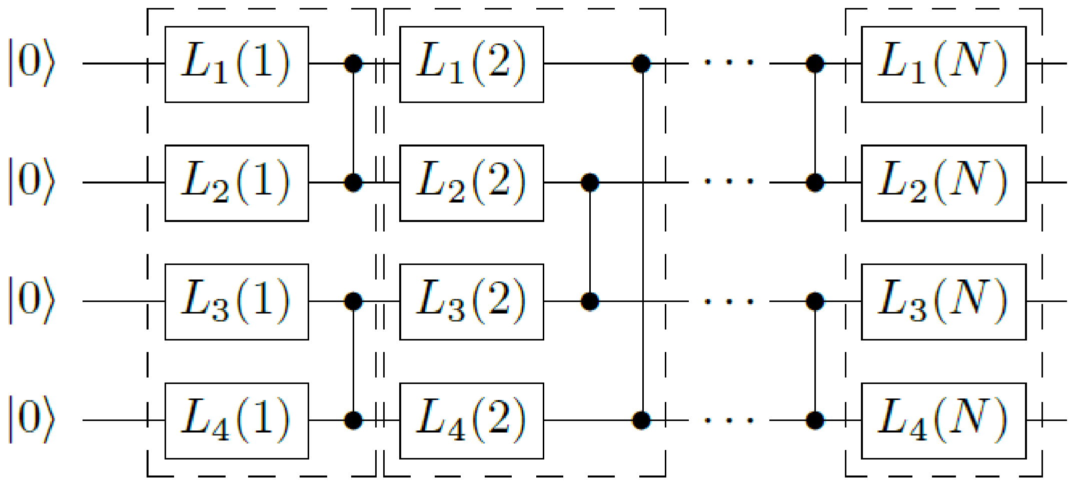
No mathematical proofs: heuristic experiment.

Is entanglement playing any role or just the fact that we are considering more qubits?

Which entanglement ansatz should we use? We tried alternating entanglement ansatz.



(a) Ansatz with no entanglement



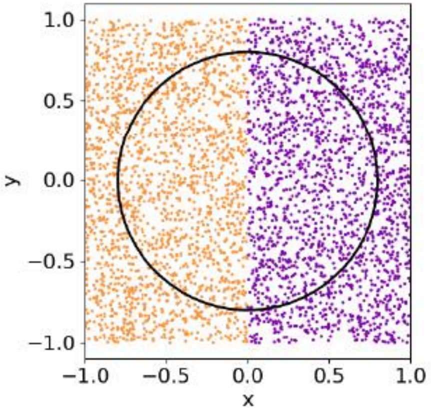
(b) Ansatz with entanglement

A. Pérez-Salinas, ACL,, E. Gil-Fuster and J. I. Latorre, Quantum 4, 226 (2020), arXiv:1907.02085

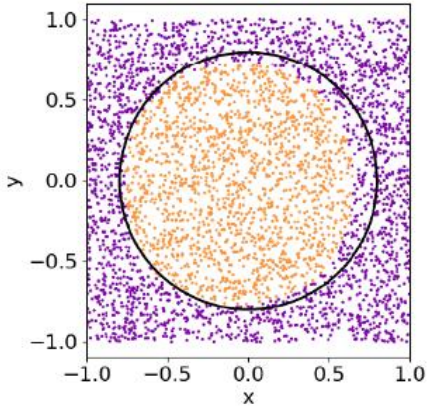
Benchmarks

Problem	# Classes	Dimension
Circle	2	2
3 circles	4	2
Hypersphere	2	4
Annulus	3	2
Non-convex	2	2
Binary annulus	2	2
Sphere	2	3
Squares	4	2
Wavy lines	4	2

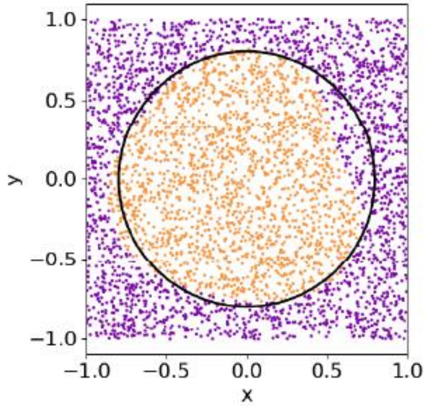
2D circle



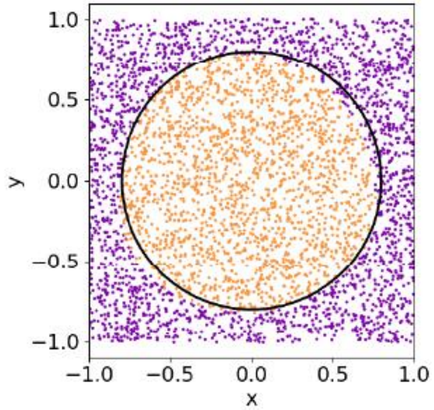
(a) 1 layer



(b) 2 layers



(c) 4 layers

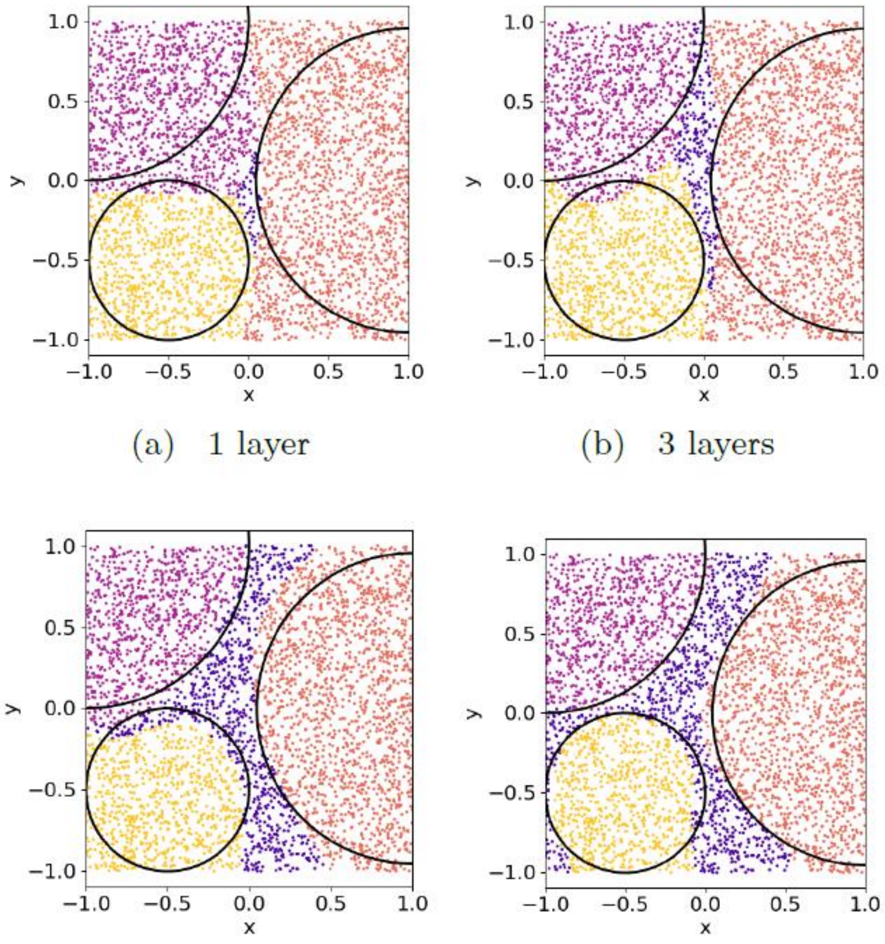


(d) 8 layers

	χ^2_f			χ^2_{wf}				
Qubits	1	2		1	2		4	
Layers		No Ent.	Ent.		No Ent.	Ent.	No Ent.	Ent.
1	0.50	0.75	–	0.50	0.76	–	0.76	–
2	0.85	0.80	0.73	0.94	0.96	0.96	0.96	0.96
3	0.85	0.81	0.93	0.94	0.97	0.95	0.97	0.96
4	0.90	0.87	0.87	0.94	0.97	0.96	0.97	0.96
5	0.89	0.90	0.93	0.96	0.96	0.96	0.96	0.96
6	0.92	0.92	0.90	0.95	0.96	0.96	0.96	0.96
8	0.93	0.93	0.96	0.97	0.95	0.97	0.95	0.96
10	0.95	0.94	0.96	0.96	0.96	0.96	0.96	0.97

Training/test points = 200/4000
Random accuracy = 50%

2D: 3 circles

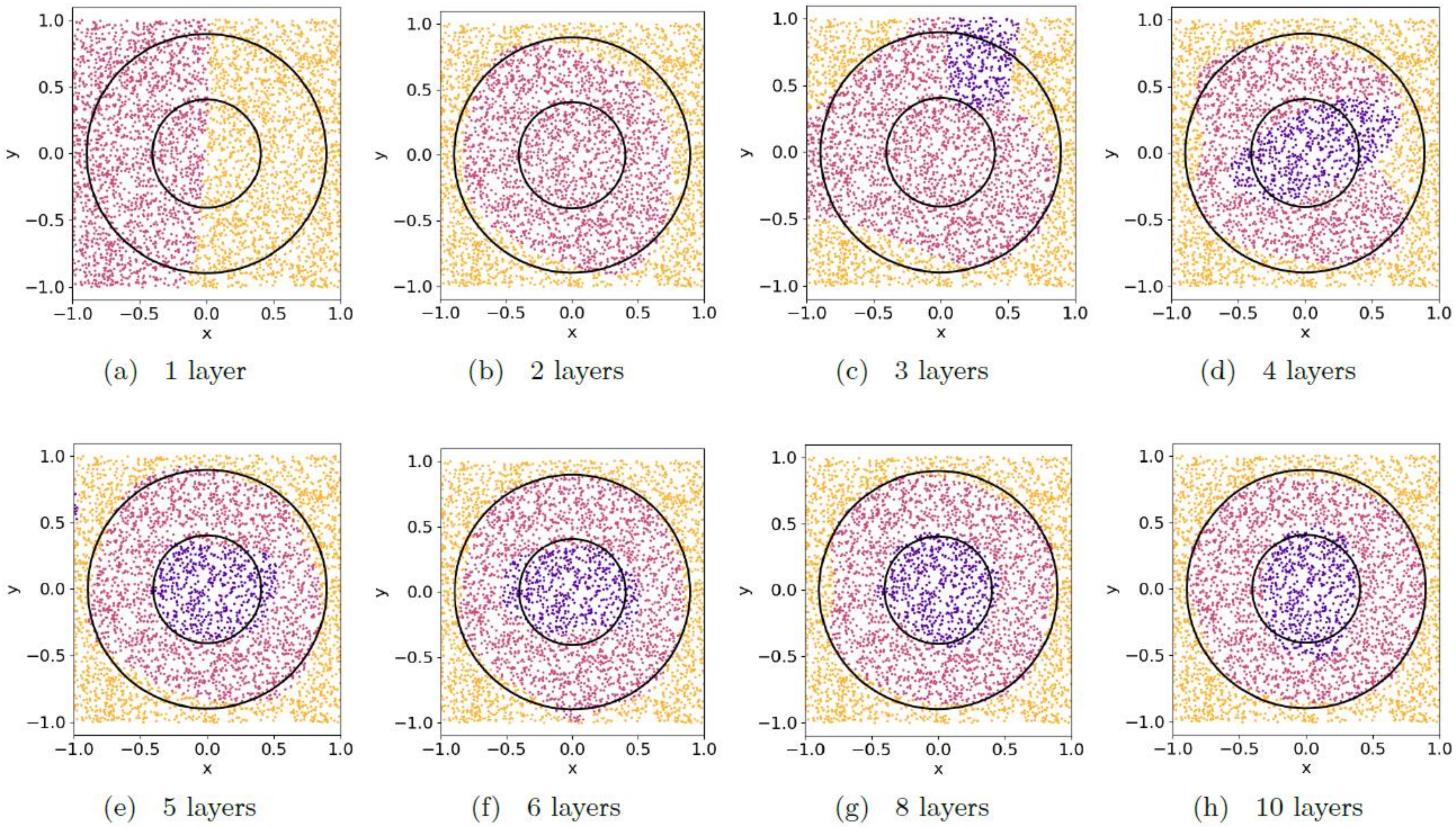


(a) 1 layer (b) 3 layers (c) 4 layers (d) 10 layers

Qubits	χ^2_f 1	χ^2_f 2		χ^2_{wf} 1	χ^2_{wf} 2		χ^2_{wf} 4	
Layers		No Ent.	Ent.		No Ent.	Ent.	No Ent.	Ent.
1	0.73	0.56	–	0.75	0.81	–	0.88	–
2	0.79	0.77	0.78	0.76	0.90	0.83	0.90	0.89
3	0.79	0.76	0.75	0.78	0.88	0.89	0.90	0.89
4	0.84	0.80	0.80	0.86	0.84	0.91	0.90	0.90
5	0.87	0.84	0.81	0.88	0.87	0.89	0.88	0.92
6	0.90	0.88	0.86	0.85	0.88	0.89	0.89	0.90
8	0.89	0.85	0.89	0.89	0.91	0.90	0.88	0.91
10	0.91	0.86	0.90	0.92	0.90	0.91	0.87	0.91

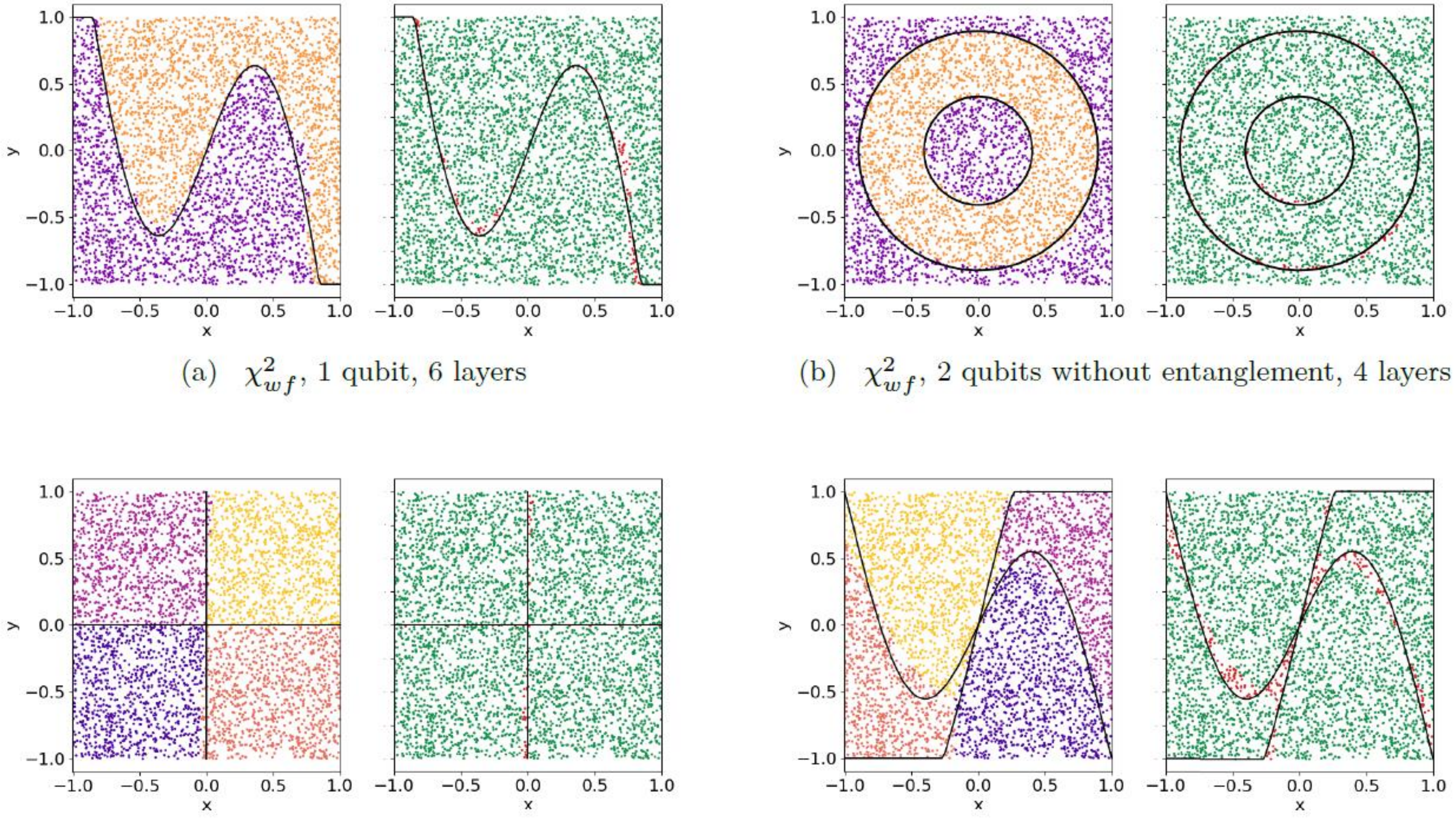
Training/test points = 200/4000
Random accuracy = 25%

2D: Annulus



Training/test points = 200/4000
Random accuracy = 33%

Other 2D problems



(a) χ^2_{wf}, 1 qubit, 6 layers

(b) χ^2_{wf}, 2 qubits without entanglement, 4 layers

(c) χ^2_{f}, 2 qubits without entanglement, 6 layers

(d) χ^2_{wf}, 2 qubits with entanglement, 6 layers

Summary and classical comparison

NN: one hidden layer with 100 neurons, ReLu, L-BFGS-B.

SVM: default sklearn.svm.SVC.

Problem	Classical classifiers		Quantum classifier	
	NN	SVC	χ^2_f	χ^2_{wf}
Circle	0.96	0.97	0.96	0.97
3 circles	0.88	0.66	0.91	0.91
Hypersphere	0.98	0.95	0.91	0.98
Annulus	0.96	0.77	0.93	0.97
Non-Convex	0.99	0.77	0.96	0.98
Binary annulus	0.94	0.79	0.95	0.97
Sphere	0.97	0.95	0.93	0.96
Squares	0.98	0.96	0.99	0.95
Wavy Lines	0.95	0.82	0.93	0.94

The aim of this classical benchmarking is not to make an extended review of what classical machine learning is capable to perform.

The aim is to compare our simple quantum classifier to simple models such as shallow neural networks and simple support vector machines.

The result of the single-qubit classifier is comparable with classical models

Conclusions and remarks

- A single-qubit is capable of performing a multiclassification task when:
    1. Assisted with a classical optimization subroutine (VQA).
    2. Data is re-uploaded along the circuit.

- Its performance is comparable with other classical methods such as NN and SVM.

- Its extensión to multiple qubits and the entanglement role should be studied in more detail.

- Is it affected by the barren plateau problem?
  There exist a correlation between the different layers: the data points encoded.

- Are other encoding strategies better than the linear encoding?

- Use this model beyond classification: meta-VQE algorithm uses data re-uploading strategy.

If you don't know how to construct your encoding, let the quantum circuit do that for you!

ACL, J. S. Kottmann, A. Aspuru-Guzik, arXiv:2009.13545

Aknowledgements

Adrián Pérez-Salinas

Elies Gil-Fuster

José Ignacio Latorre

UNIVERSITAT DE BARCELONA

the matter lab

BSC Barcelona Supercomputing Center
Centro Nacional de Supercomputación