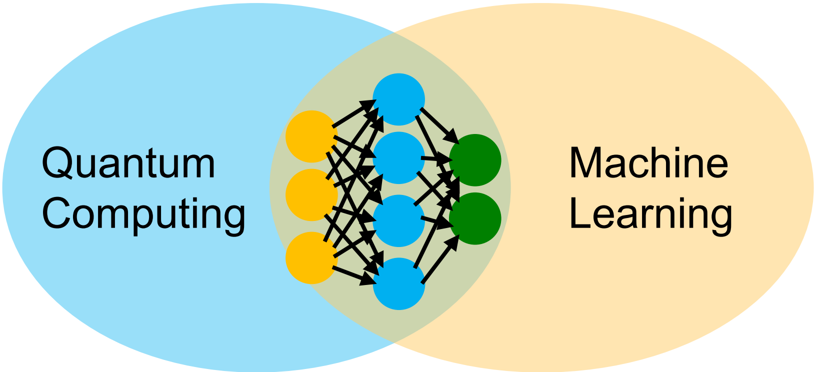
Quantum Machine Learning on NISQ hardware



UNIVERSITÀ DI PAVIA

Dario Gerace

Dipartimento di Fisica, Università di Pavia (IT)

INT online program on «Scientific Quantum Computing and Simulation on Near-Term Devices» – 26/10/2020

The NISQ era

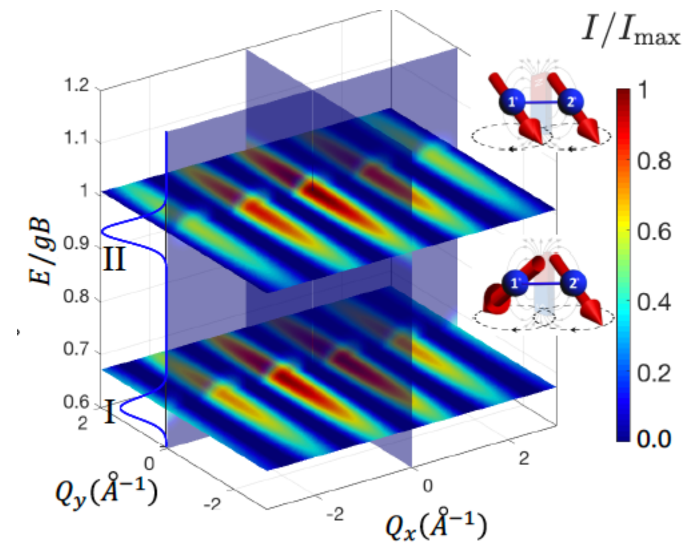
What can we do with near-term quantum computing hardware?

→ Any practical quantum advantage to be expected?

➢ Quantum Simulations

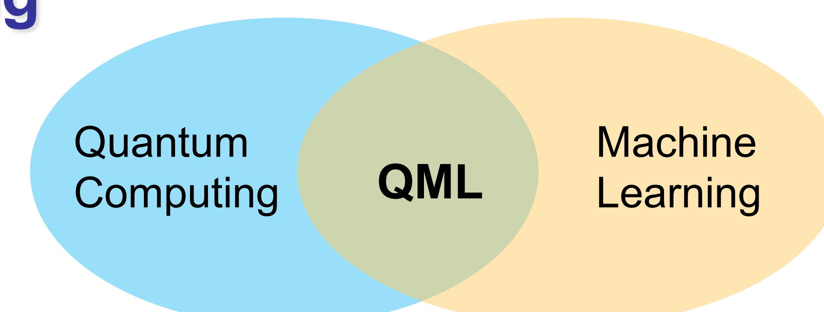
E.g.: inelastic neutron scattering cross section from magnetic molecules simulated on IBM-Q

Chiesa et al., *Nature Physics* **15**, 455 (2019)



➢ Quantum Machine Learning

new knowledge at the forefront of classical and quantum computing
→ potential for *practical* advantage

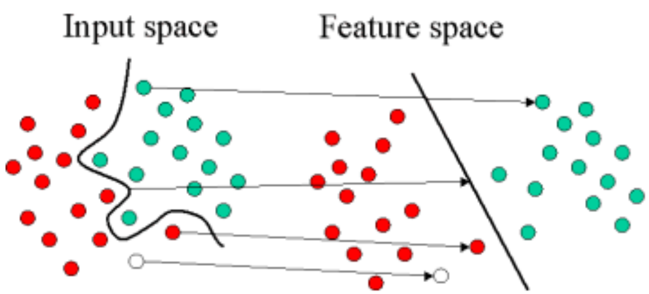


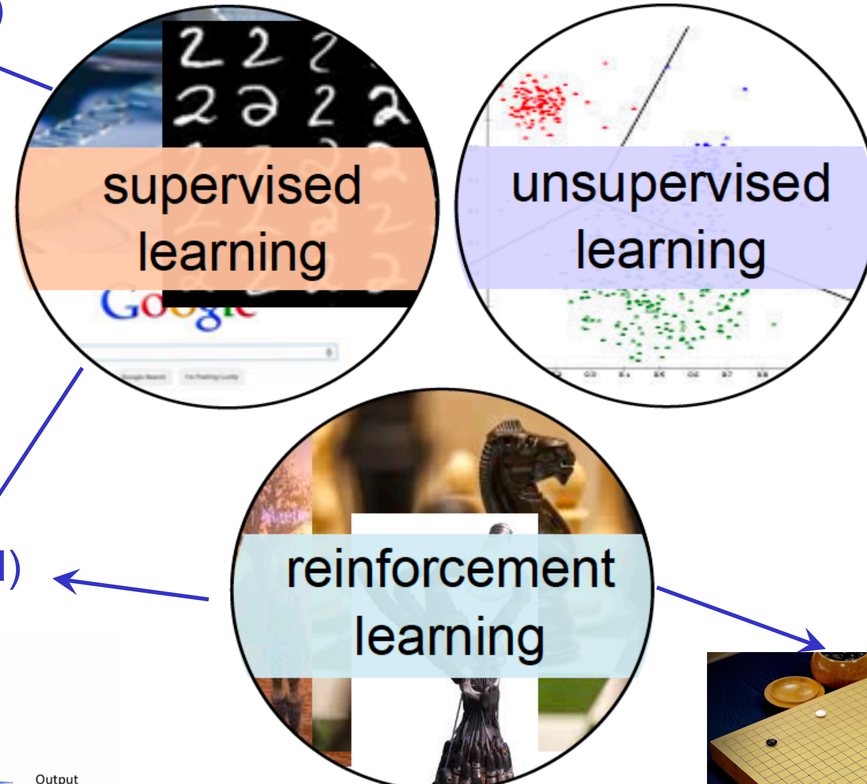
Machine Learning

ML is based on finding suitable mathematical models (functions) mapping **input data** into **output predictions**

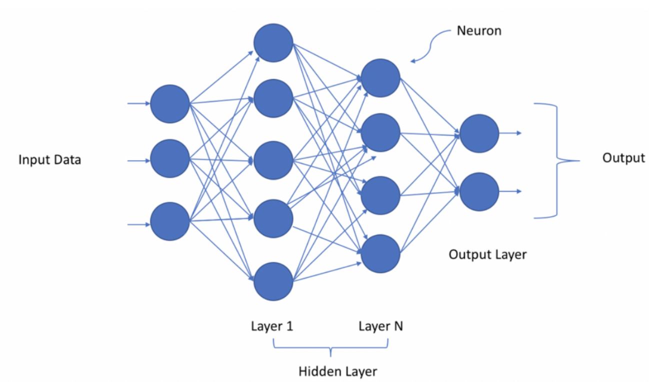
UNIVERSITÀ DI PAVIA

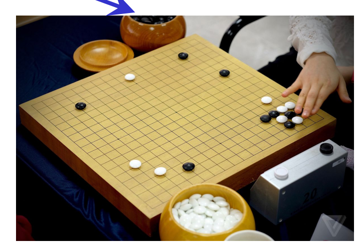
E.g., support vector machines (SVM)





E.g., deep neural networks (DNN)



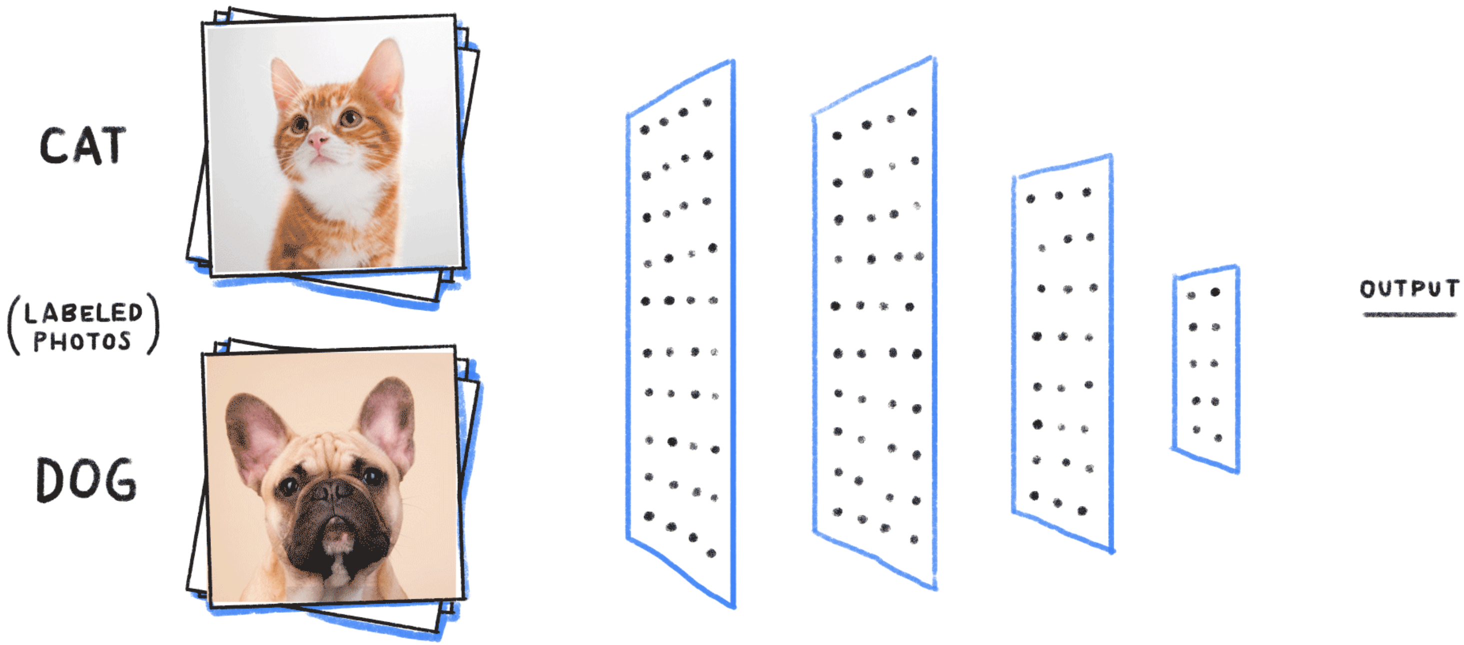


D. Silver et al., Nature **550**, 354 (2017)

An example of supervised learning

UNIVERSITÀ DI PAVIA

- Image classification through training of a DNN



<u>Credits</u>: becominghuman.ai/building-an-image-classifier-using-deep-learning-in-python-totally-from-a-beginners-perspective-be8dbaf22dd8

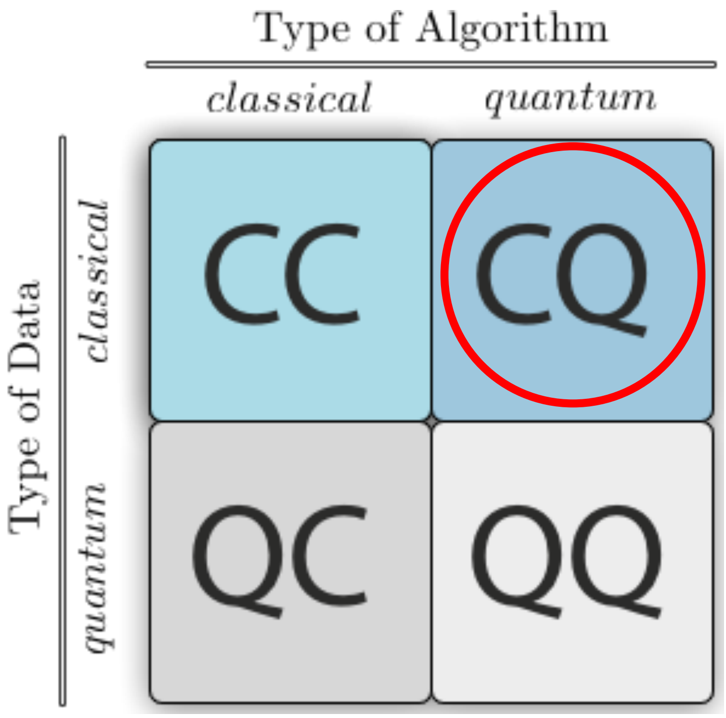
- In general:

$$\boldsymbol{x} \longrightarrow f(\boldsymbol{x},\theta) \longrightarrow \boldsymbol{y}$$

ML task is to learn how f maps $\boldsymbol{x}$ into $\boldsymbol{y}$, on varying θ, such that the algorithm can correctly predict $\boldsymbol{y}$ upon being fed with previously unknown $\boldsymbol{x}$

Quantum Machine Learning

Applying quantum computing resources to ML tasks



Schuld & Petruccione, *Supervised learning with Quantum Computers* (Springer, 2018)

Biamonte et al., *Quantum Machine Learning*, Nature **549**, 195 (2017)

Kernel methods

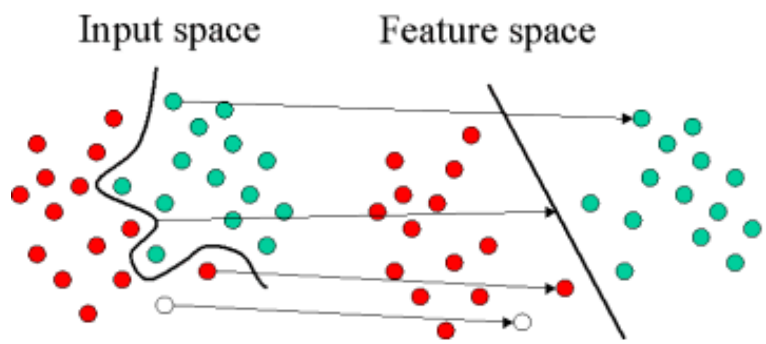
- Map data into a much larger ‘feature’ space

$$\boldsymbol{x} \longrightarrow \phi(\boldsymbol{x})$$

- Define the kernel (inner product)

$$k(x, x') \doteq \langle \phi(\boldsymbol{x}), \phi(\boldsymbol{x}') \rangle_{\mathcal{F}}$$

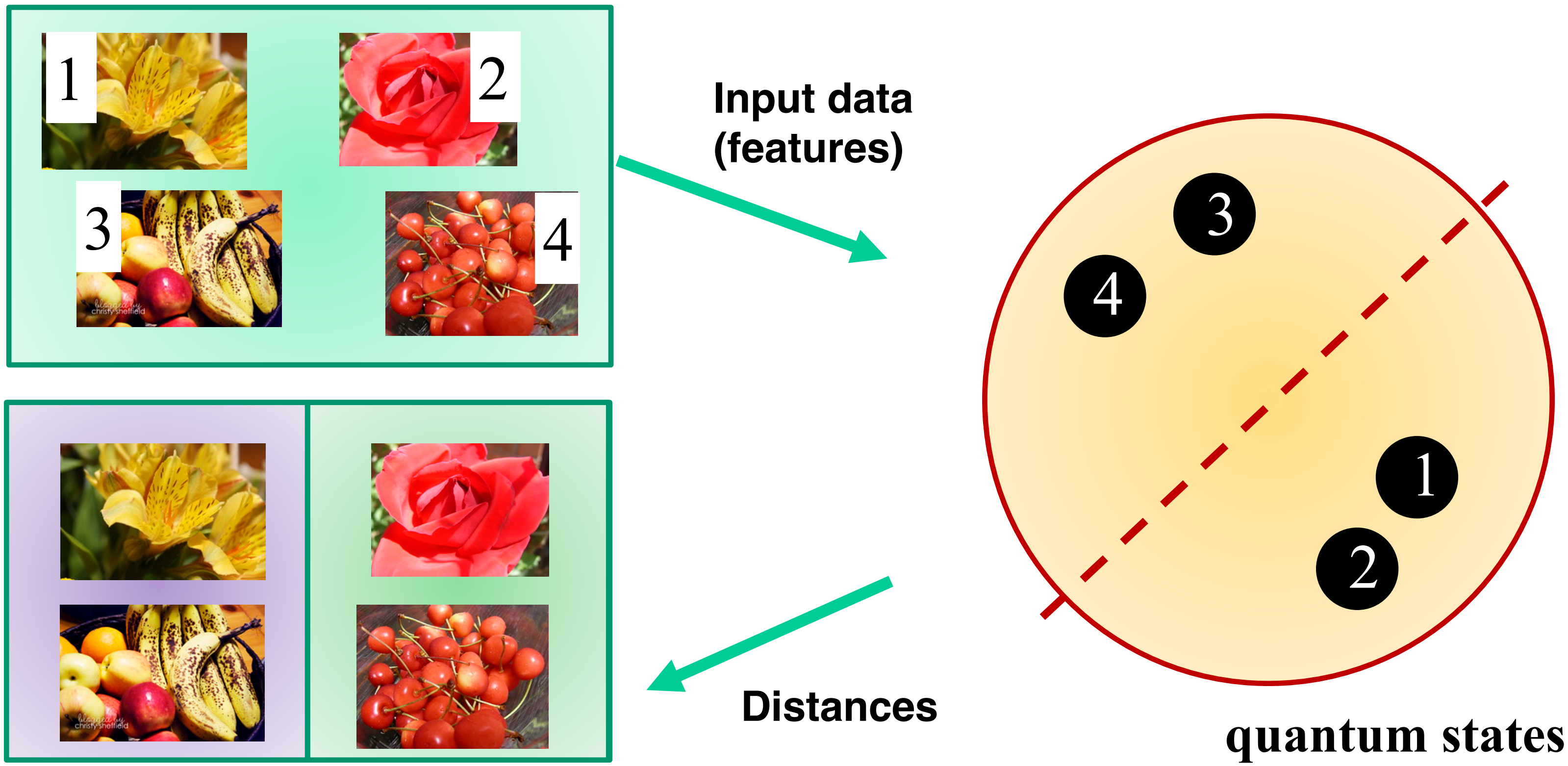
- Find the best separating hyperplane in the higher dimensional feature space (through the given kernel)



- Support vector machines (SVM) belong to this class of algorithms

Quantum SVM

Mapping classical data to a (exp large) feature space and finding distances is what a quantum computer can do best



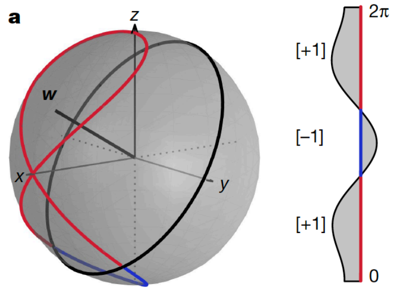
Schuld & Killoran, *Quantum machine learning in feature Hilbert spaces*, Phys. Rev. Lett. **122**, 040504 (2019)

Schuld, *Machine learning in quantum spaces*, Nature **567**, 179 (2019)

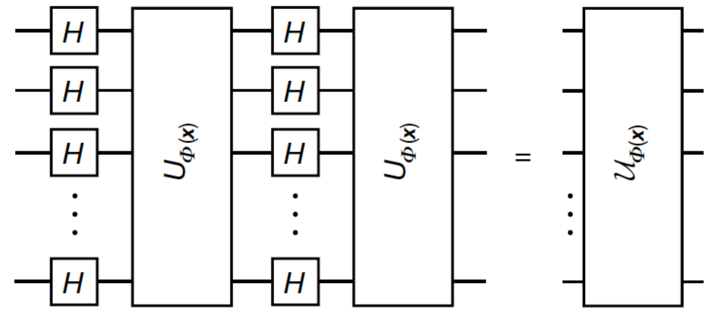
The IBM result on NISQ hardware

UNIVERSITÀ DI PAVIA

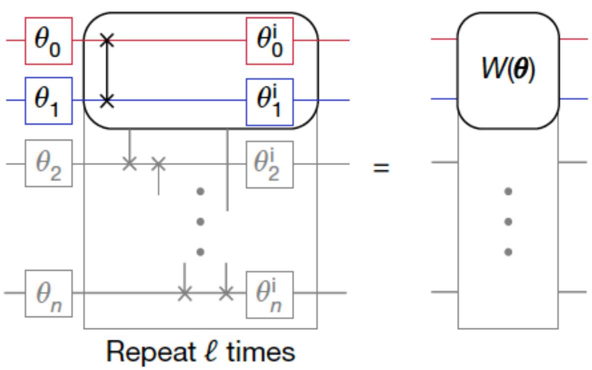
1- Binary data

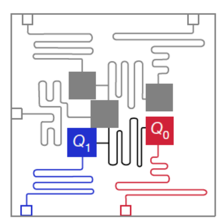


2- Quantum feature map $x \longrightarrow |\Phi(x)\rangle$



3 - Parametrized variational circuit



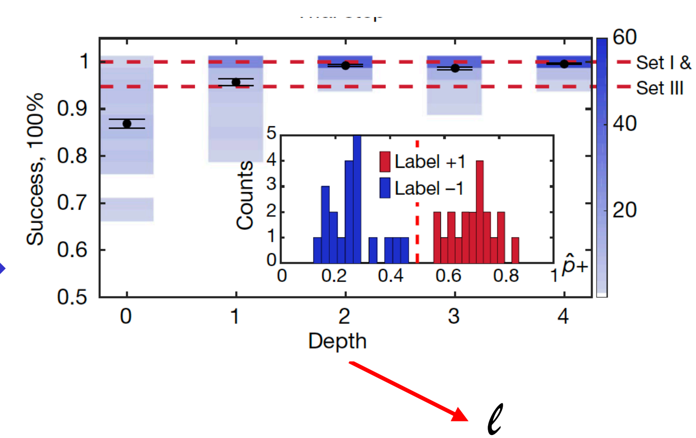


4 – measurement with binary output

$$p_y(\hat{x}) = \langle \Phi(x)|W^\dagger(\theta) M_y \hat{W}(\theta)|\Phi(x)\rangle$$

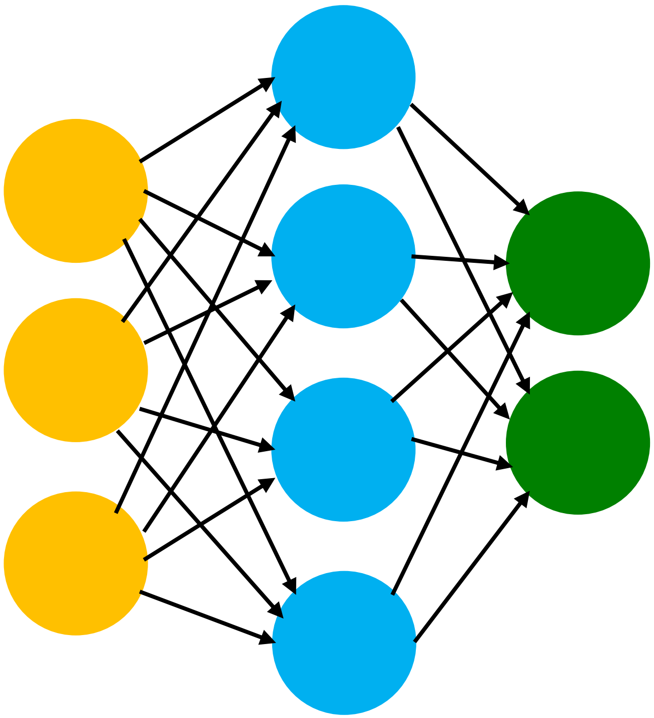
θ is trained through classical optimization algorithm

classification after training →



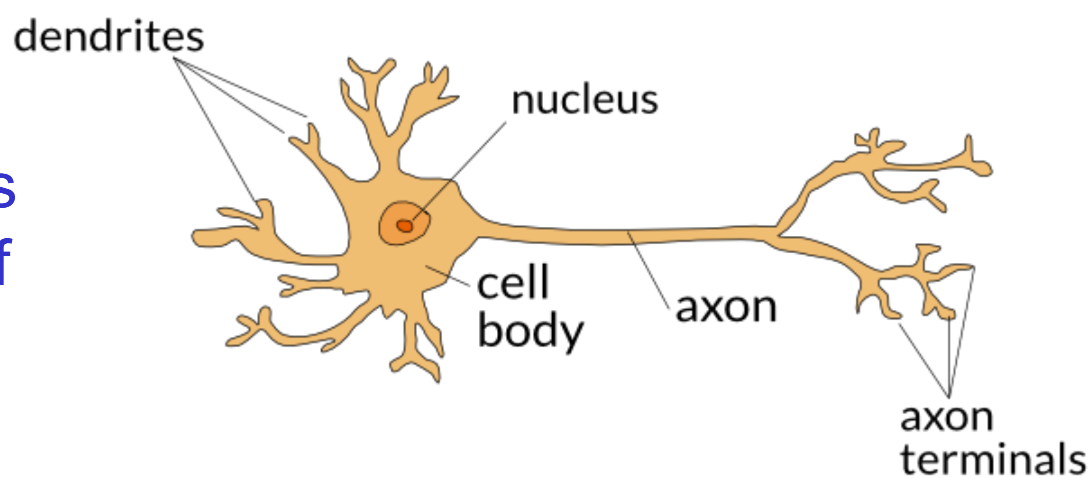
Havlicek et al.,
Nature **567**, 209 (2019)

Artificial neural networks (ANN)

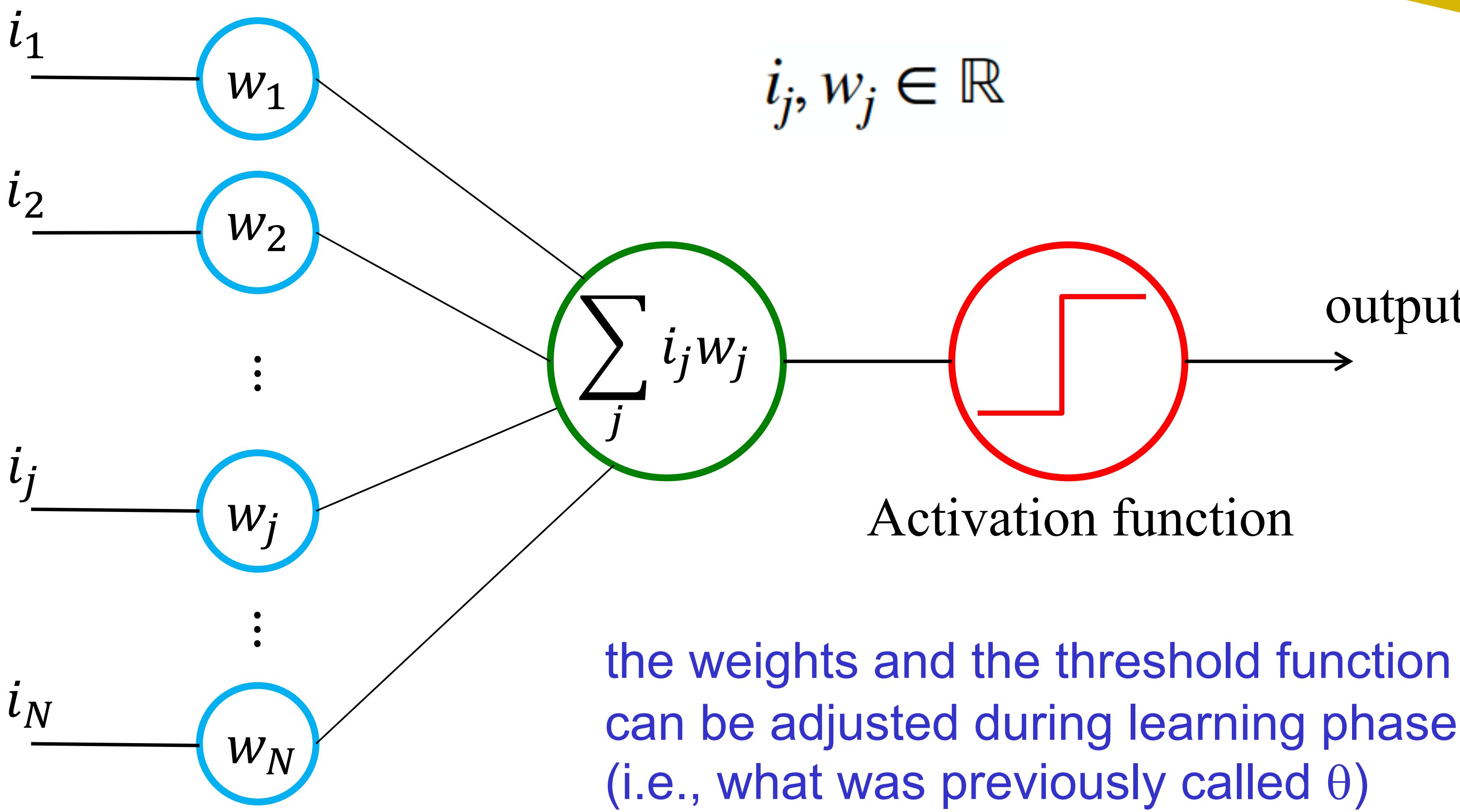


- Basis for several AI algorithms
- applications in pattern recognition, speech recognition, classification, …

Each node mimics the functionality of a single neuron



The classical perceptron as a model of artificial neuron

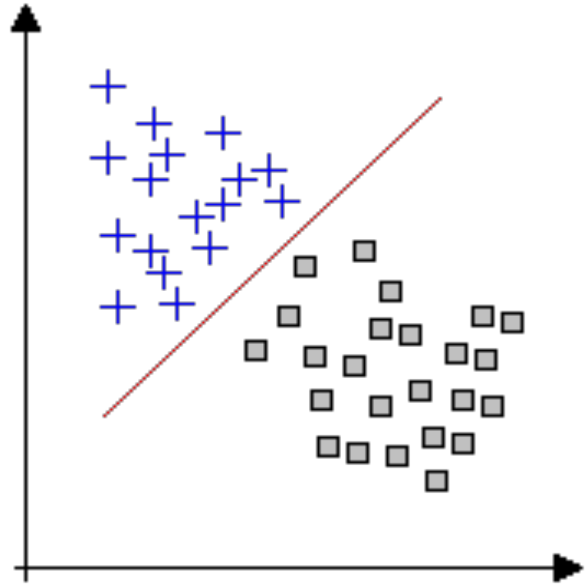


$i_j, w_j \in \mathbb{R}$

the weights and the threshold function can be adjusted during learning phase (i.e., what was previously called θ)

Rosenblatt, Psychol. Rev. **65**, 386 (1958)

Linear classifier



The classical perceptron is the simplest linear classifier

It requires extension to a multilayer structure to be able to perform nonlinear tasks

Quantum neural network models

➢ Quantum perceptrons

Schuld et al., Phys. Lett. A **7**, 660 (2015)

N. Wiebe, A. Kapoor and K. M. Svore, arXiv:1602.04799 (2016)

Y. Cao, G. G. Guerreschi and A. Aspuru-Guzik, arXiv:1711.11240 (2017)

Torrontegui et al., EPL (Europhysics Letters) 125 (2019)

…

➢ Quantum algorithms for artificial neural networks

Schuld et al., EPL **119**, 60002 (2017)

Wan et al., npj Quant Info **3**, 36 (2017)

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

Rebentrost et al., Phys. Rev. A **98**, 042308 (2018)

Grant et al., npj Quant Info **4**, 65(2018)

Killoran et al., Phys. Rev. Research **1**, 033063 (2019)

Cong et al., Nature Physics (2019)

Mari et al., Quantum **4**, 340 (2020)

…

UNIVERSITÀ DI PAVIA

McCulloch-Pitts neurons on a quantum computer

The key function $\vec{i} \cdot \vec{w} = \sum_j i_j w_j$

- Encoding input and weights

$$\vec{i} = \begin{pmatrix} i_0 \\ i_1 \\ \vdots \\ i_{2^N-1} \end{pmatrix} \qquad \vec{w} = \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_{2^N-1} \end{pmatrix}$$

McCulloch-Pitts neuron model

$$i_j, w_j = -1, +1$$

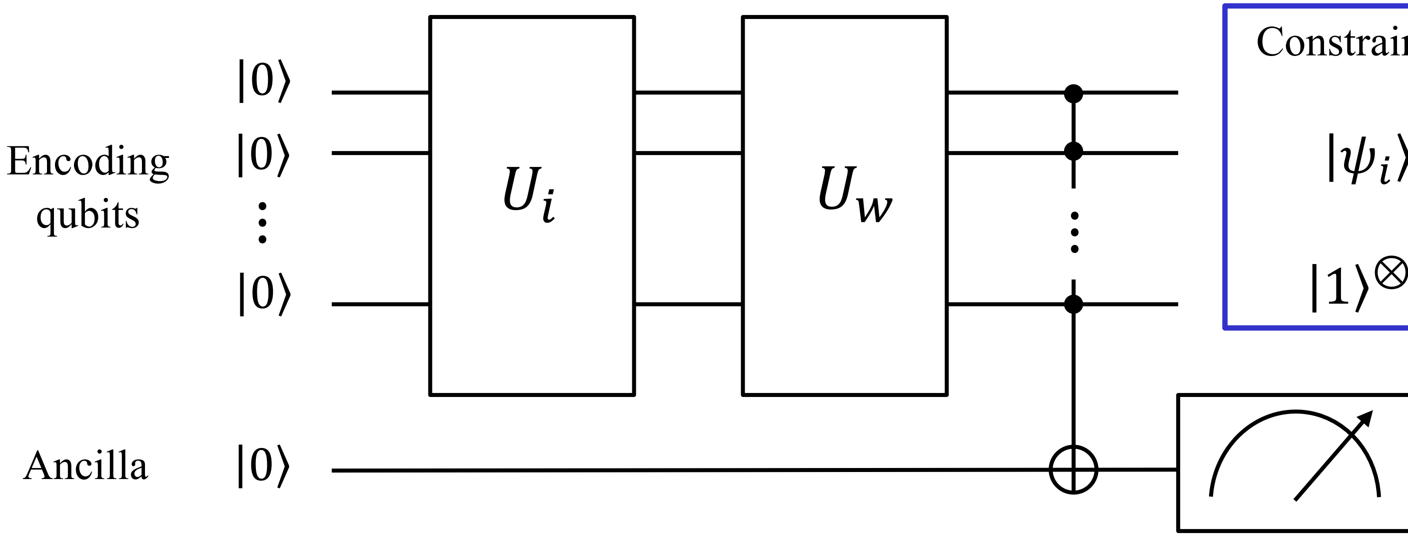
$$|\psi_i\rangle = \frac{1}{\sqrt{2^N}} \sum_{j=0}^{2^N-1} i_j |j\rangle \qquad |\psi_w\rangle = \frac{1}{\sqrt{2^N}} \sum_{j=0}^{2^N-1} w_j |j\rangle$$

$$|j\rangle \in \{ |000\cdots00\rangle, |000\cdots01\rangle, \cdots |111\cdots11\rangle \}$$

Tacchino et al., npj Quant. Info **5**, 26 (2019)

McCulloch-Pitts neurons on a quantum computer

The quantum algorithm: a circuit model



$$\Rightarrow \left| c_{2^N-1} \right|^2 = \left| \sum_j i_j w_j \right|^2$$

Constraints on the unitaries:

$$|\psi_i\rangle = U_i|0\rangle^{\otimes N}$$

$$|1\rangle^{\otimes N} = U_w|\psi_w\rangle$$

$$|0\rangle^{\otimes N}|0\rangle_a \longrightarrow \sum_{j=0}^{2^N-2} c_j|j\rangle|0\rangle_a + c_{2^N-1}|2^N-1\rangle|1\rangle_a \quad \text{with } c_{2^N-1} = \langle\psi_i|\psi_w\rangle$$

Tacchino et al., npj Quant. Info **5**, 26 (2019)

Elementary pattern recognition

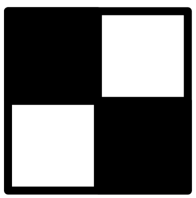
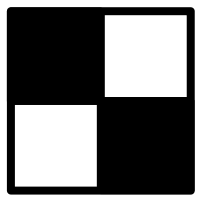
UNIVERSITÀ DI PAVIA

$\boxed{N = 2}$

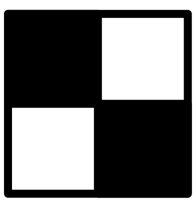
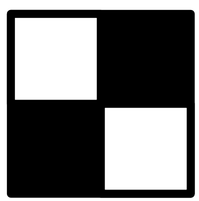
$$\vec{i} = \begin{pmatrix} i_0 \\ i_1 \\ i_2 \\ i_3 \end{pmatrix} \qquad \vec{w} = \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \end{pmatrix}$$

i_0	i_1
i_2	i_3

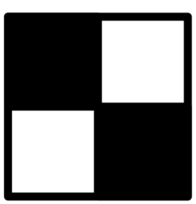
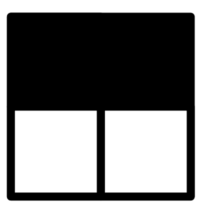
$+1 = \text{white}$
$-1 = \text{black}$

$\vec{i} =$  $\vec{w} =$  $|\langle \psi_i | \psi_w \rangle|^2 = 1$

It's me!

$\vec{i} =$  $\vec{w} =$  $|\langle \psi_i | \psi_w \rangle|^2 = 1$

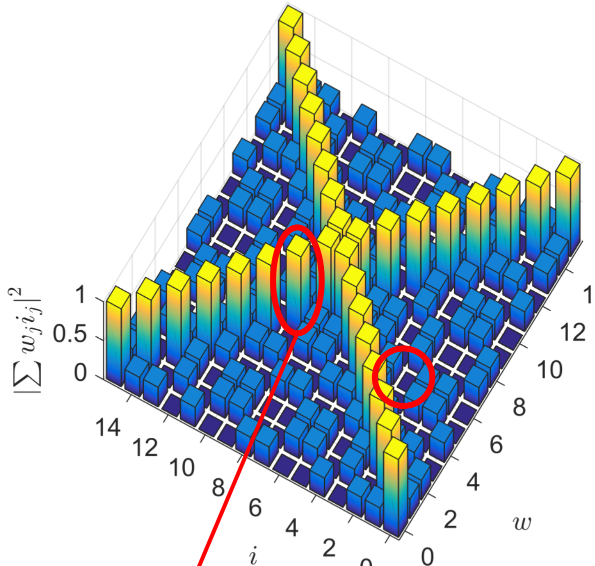
It's still me!
(in negative colors)

$\vec{i} =$  $\vec{w} =$  $|\langle \psi_i | \psi_w \rangle|^2 = 0$

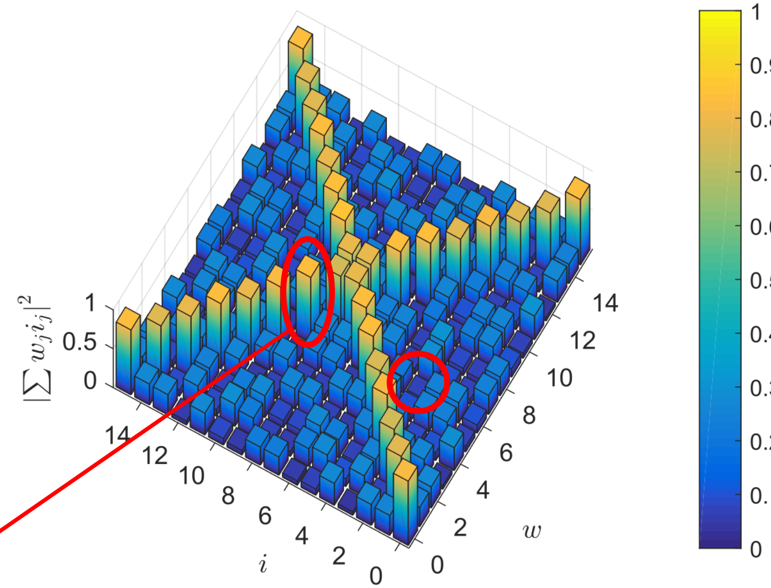
It's not me!

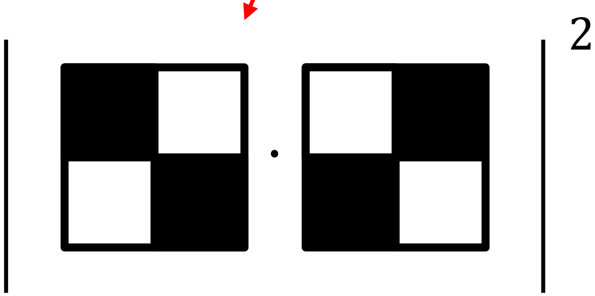
Running the algorithm on NISQ-hardware

Exact result (N = 2)

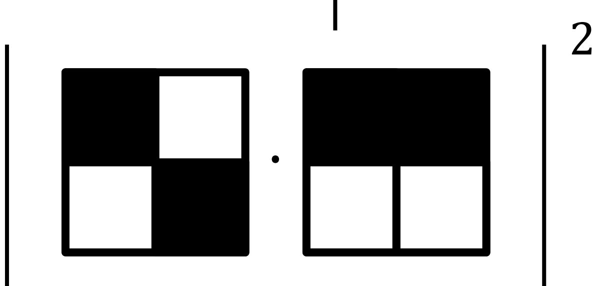


Experiment (N = 2 + 1 ancilla)



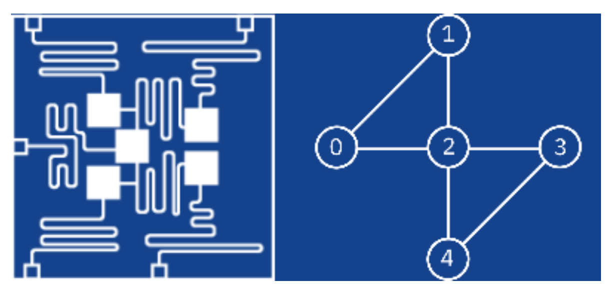


= 1 [0.84]



= 0 [0.07]

IBM-Q Experience 5-qubit ‘Tenerife’ processor



Tacchino et al., npj Quant. Info **5**, 26 (2019)

Can we train it?

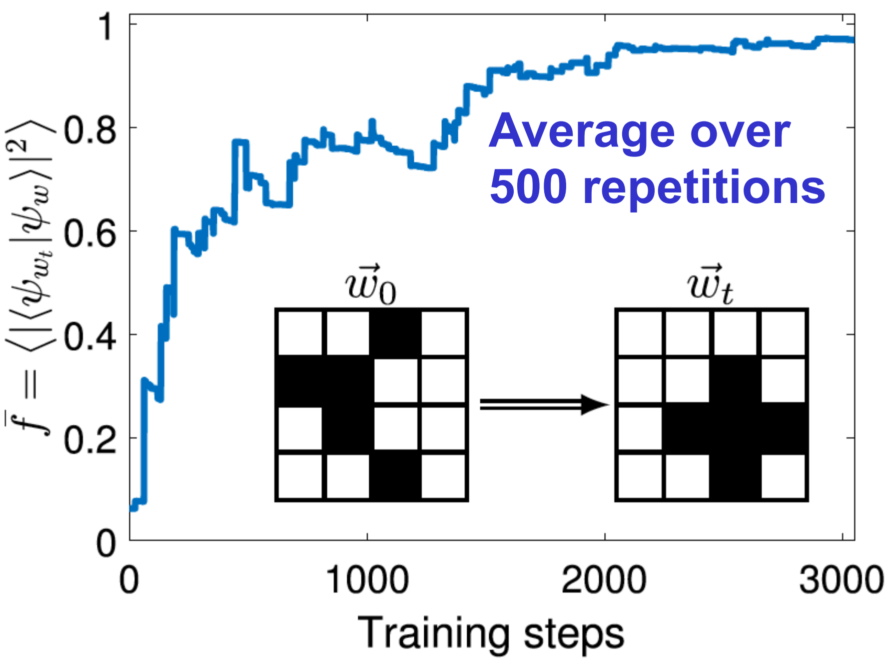
YES! → simple perceptron update rule

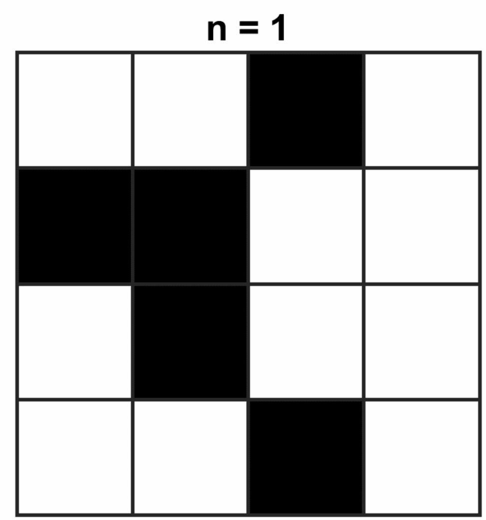
Chosen a target $\vec{w}_t$, build a training set by assigning positive (negative) labels to few inputs $\vec{\imath}$ for which $\vec{\imath} \cdot \vec{w}_t > \theta$ ($\vec{\imath} \cdot \vec{w}_t < \theta$), then randomly initialize $\vec{w}$ to be trained and:

- If $\vec{\imath} \cdot \vec{w}$ is positive but should be negative (κ common entries), randomly flip $\eta\kappa$ signs ($0<\eta<1$ 'learning' rate)
- If $\vec{\imath} \cdot \vec{w}$ is negative but should be positive (κ opposite entries), randomly flip $\eta\kappa$ different signs
- If $\vec{\imath} \cdot \vec{w}$ is correct, do nothing

Elementary training on IBM simulator

- Theoretical simulation of the algorithm for N=4 qubits + 1 ancilla (NOT on real quantum hardware, yet)
- Recognize a cross (or its negative) out of a training set of input vectors (e.g., 50 positive, 3000 negative)





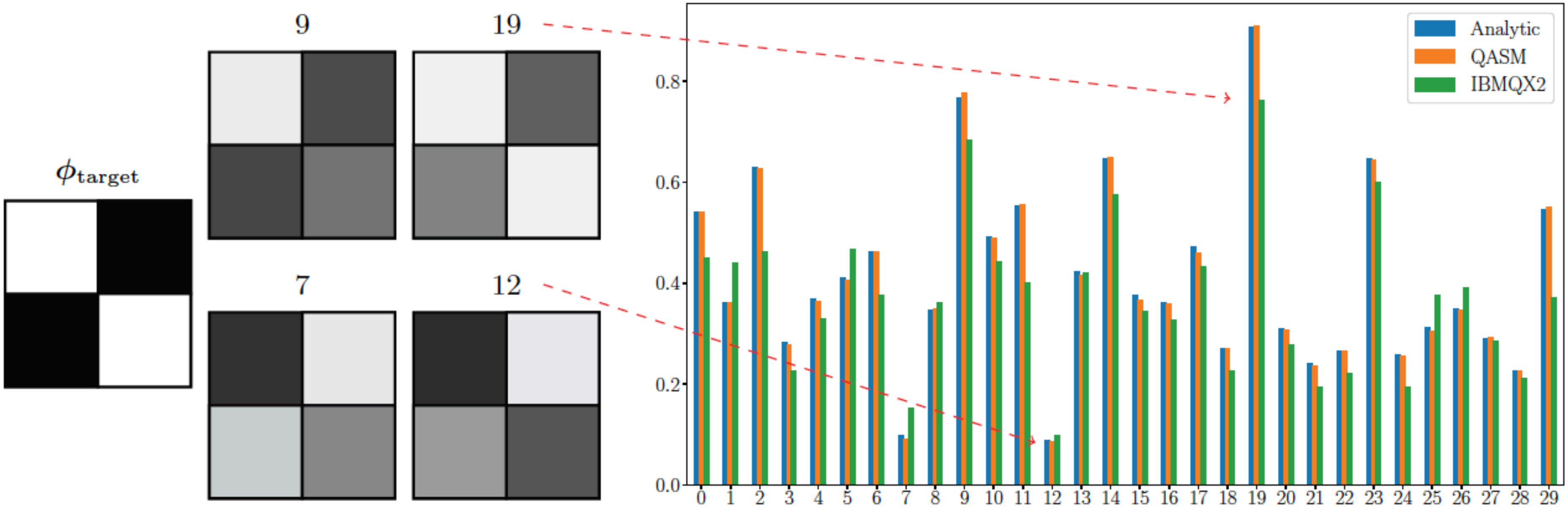
Tacchino et al., npj Quant. Info **5**, 26 (2019)

Recently extended: continuous valued input data

An array of real-valued input $\boldsymbol{\theta} = (\theta_0, \ldots, \theta_{N-1})$ with $\theta_i \in [0, \pi]$

Can be encoded as $\vec{i} = (e^{i\theta_0}, e^{i\theta_1}, \cdots, e^{i\theta_{N-1}})$ → $|\psi_i\rangle = \frac{1}{2^{n/2}} \sum_{k=0}^{2^n-1} i_k |k\rangle$

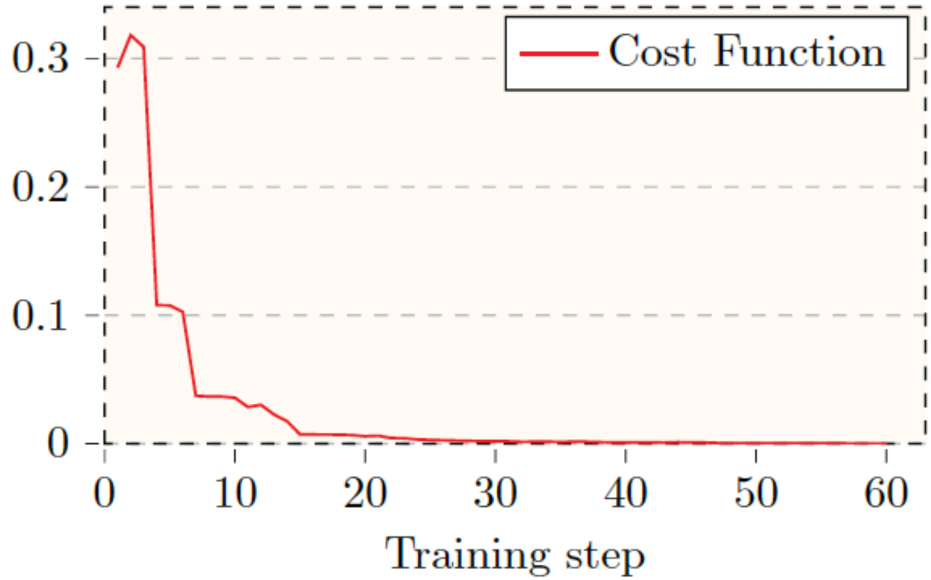
Allows to classify grey scale images without increasing the number of qubits



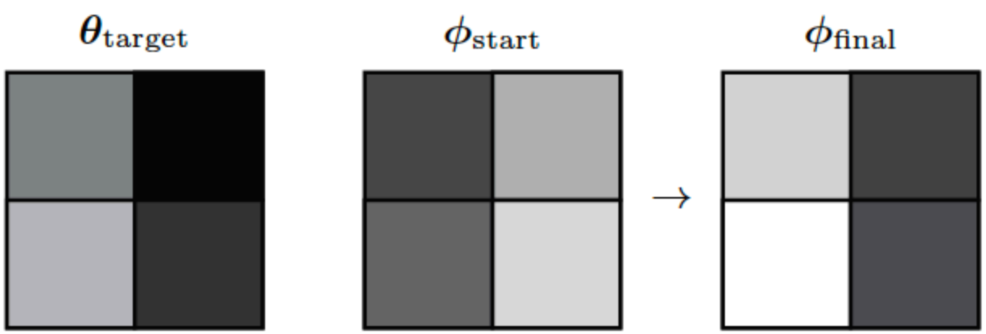
Mangini et al., Mach. Learn.: Sci. Technol. **1**, 045008 (2020)

Hybrid quantum/classical learning

Quantum algorithm trained through classical backpropagation



(a)

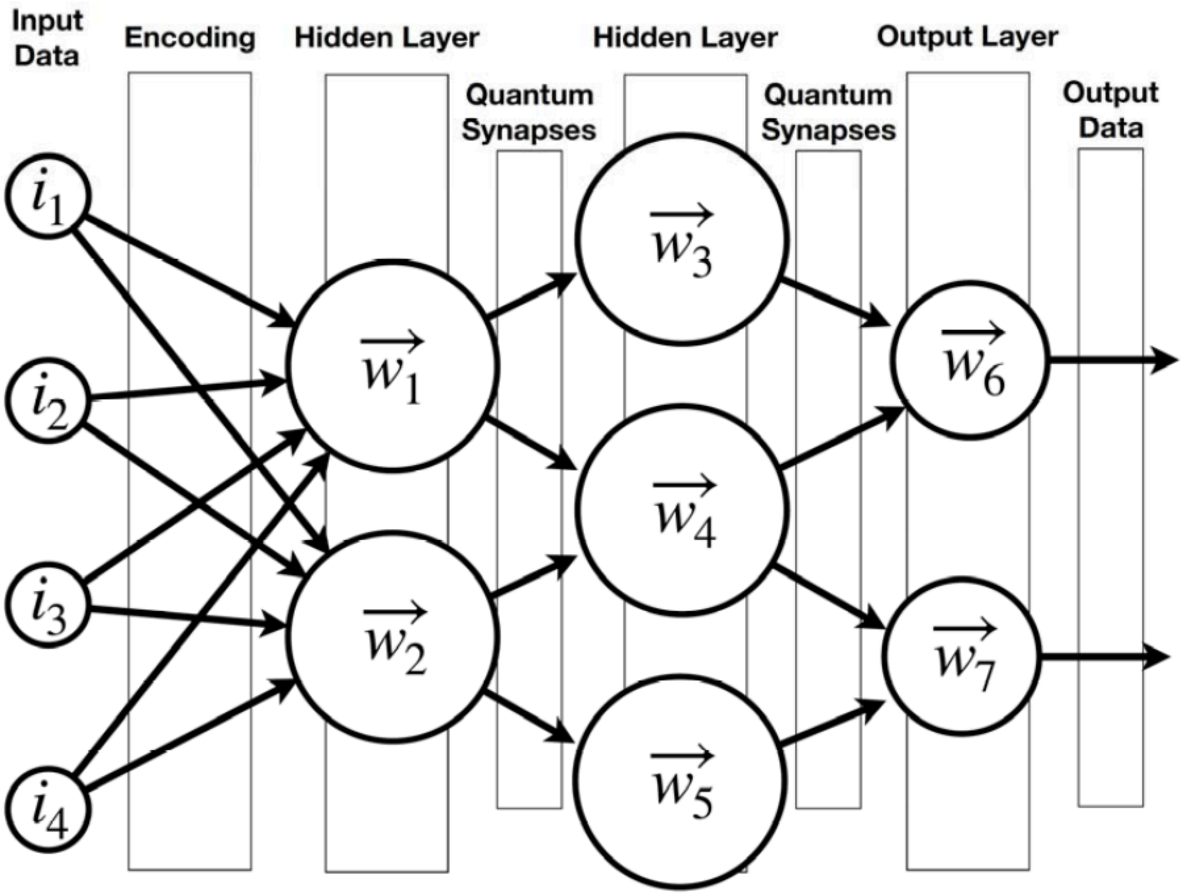


(b)

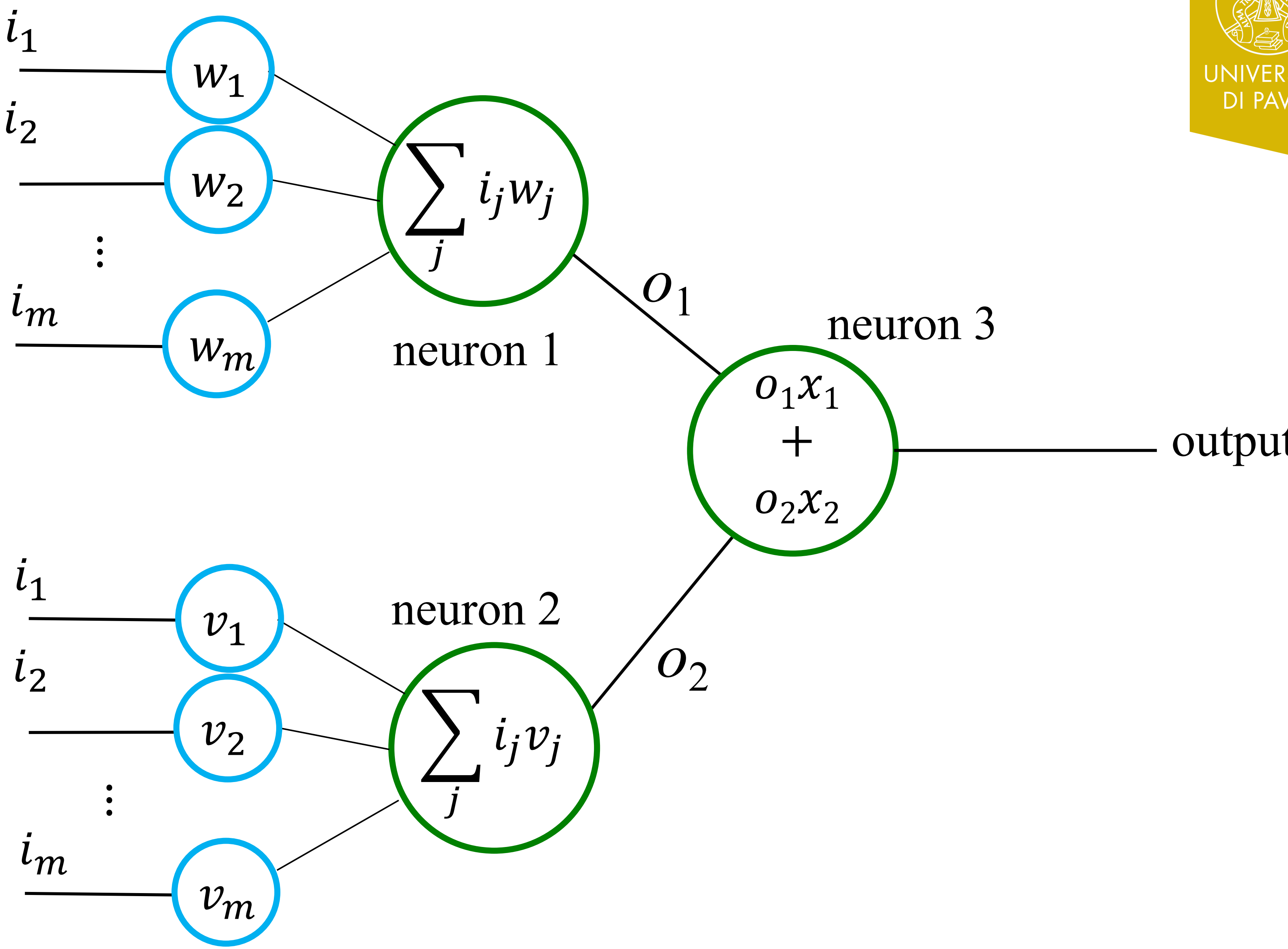
Mangini et al., Mach. Learn.: Sci. Technol. **1**, 045008 (2020)

Quantum ANN

A deep neural network is required to perform more complex ML tasks

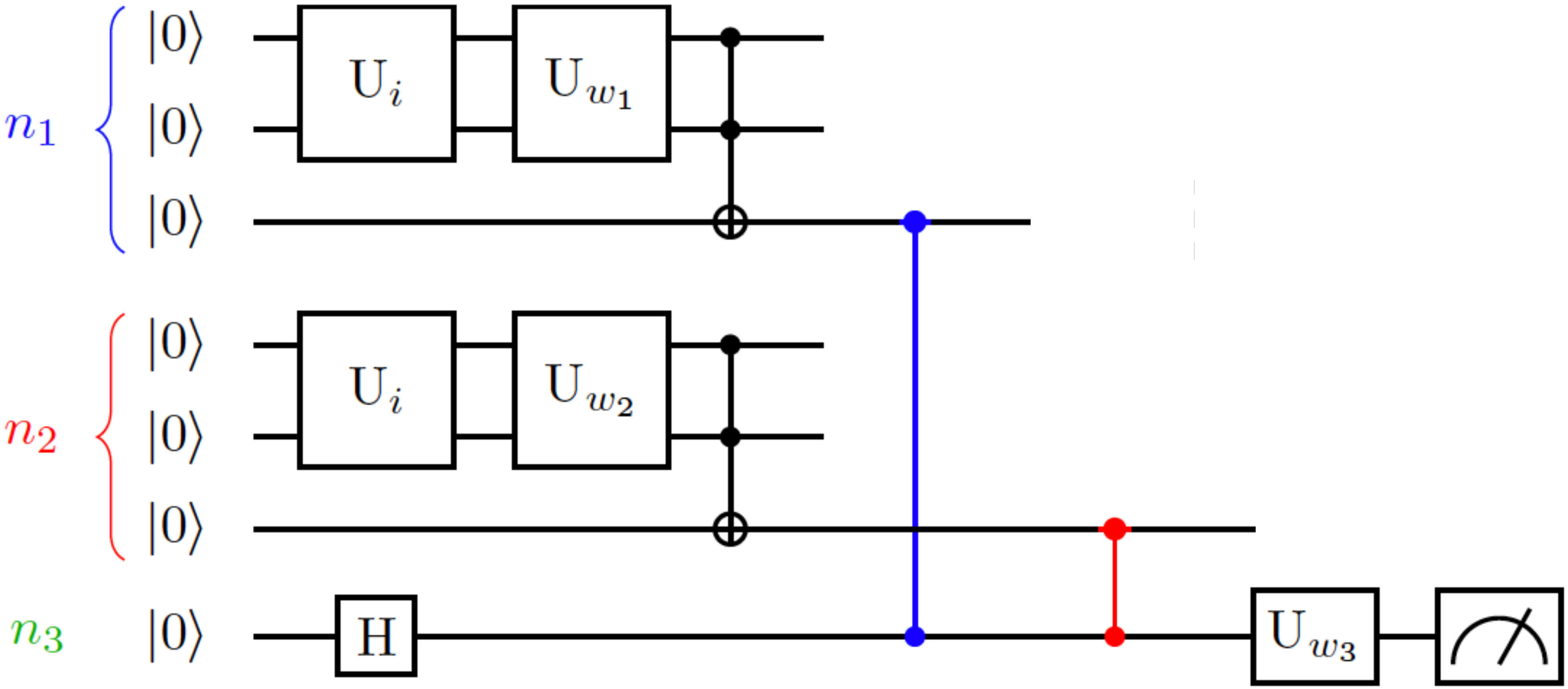


Example (still runs on NISQ hardware)



The corresponding quantum circuit

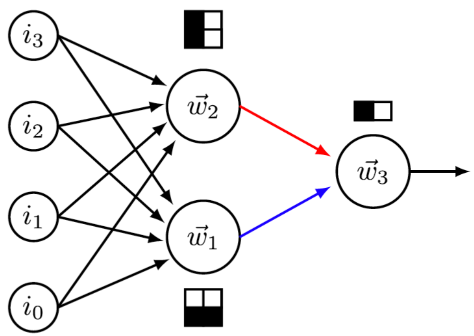
Quantum synapses → multi-controlled operations



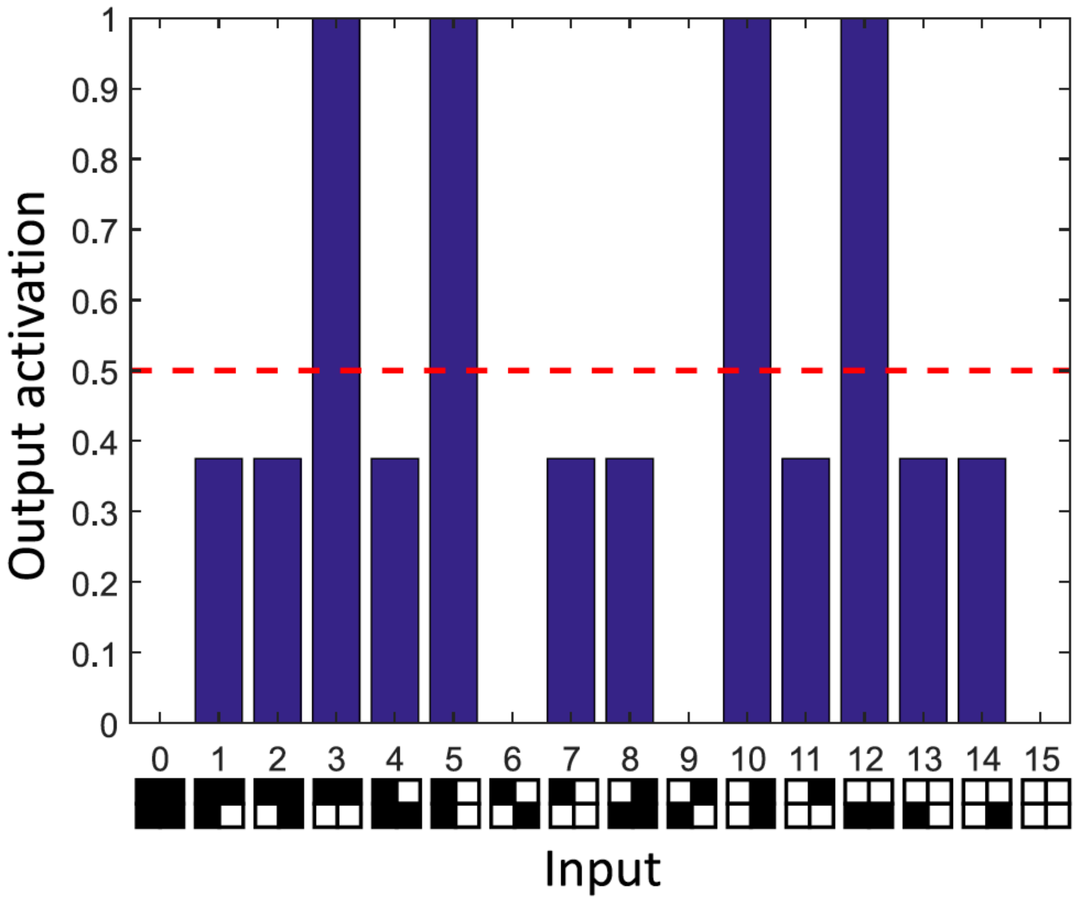
Tacchino et al., Quant. Sci. Technol. **5**, 044010 (2020)

UNIVERSITÀ DI PAVIA

A classification task that is impossible to a single perceptron

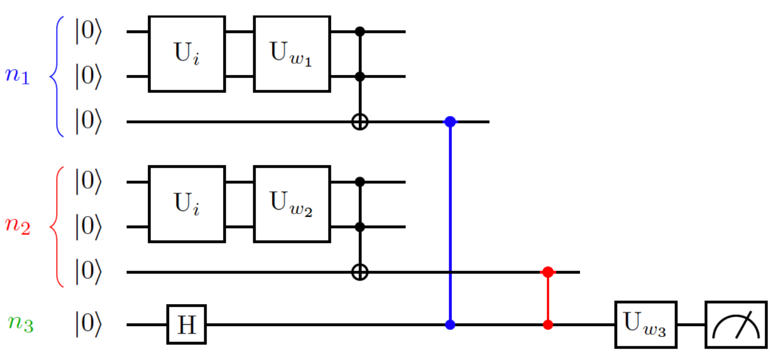


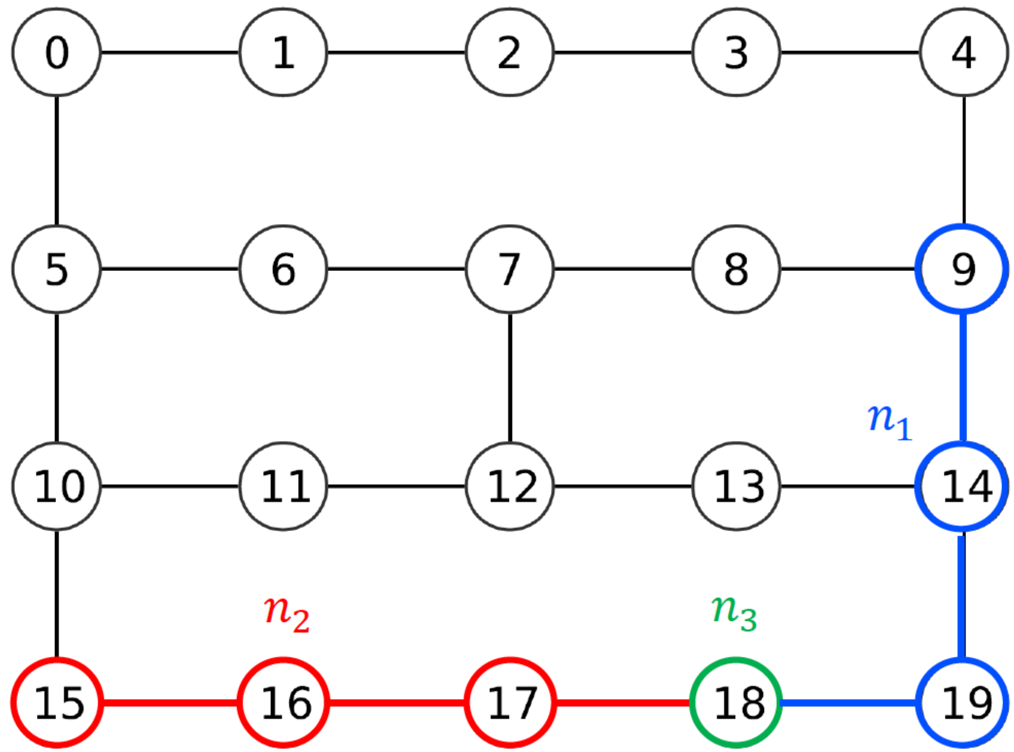
Ideal outcome of the ANN



Tacchino et al., Quant. Sci. Technol. **5**, 044010 (2020)

Implementation on IBM - Q



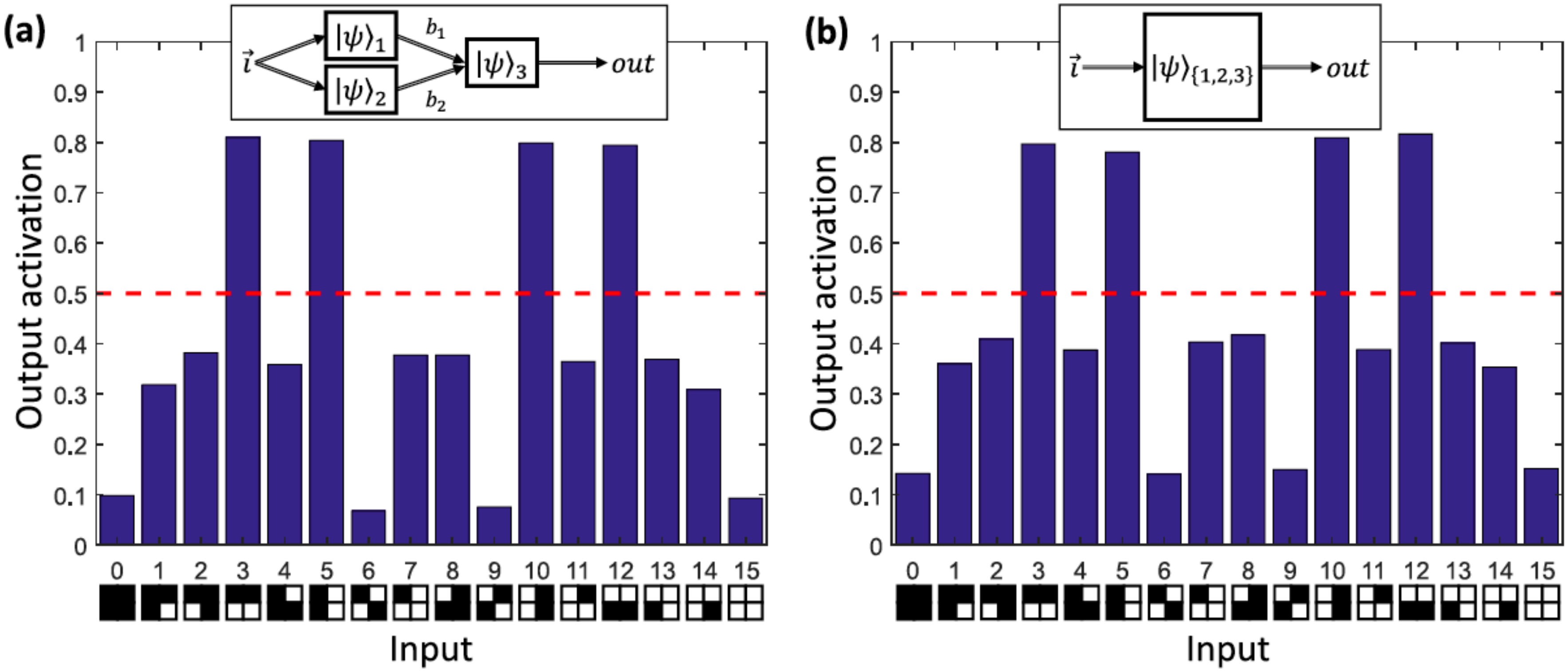


Layout of the IBM Q – Poughkeepsie NISQ processor

Results on IBM Q - Poughkeepsie

Hybrid configuration

Fully coherent configuration



with activation threshold at 0.5, the quantum hardware is able to fully classify these patterns with 100% success

Tacchino et al., Quant. Sci. Technol. **5**, 044010 (2020)

Open questions and challenges

- How does it scale?
- How efficient is it?
- Quantum training?
- Test on real hardware based on different technologies (e.g., trapped ions)
- Test with larger input data (possible use cases)
- Input quantum states (QQ)

People

UNIVERSITÀ DI PAVIA

UNIVERSITÀ DI PAVIA

F. Tacchino*

S. Mangini

D. Bajoni

C. Macchiavello

IBM

IBM Research | Zurich

P. Barkoutsos

I. Tavernelli

* Now at IBM Research, Zurich