Quantum computing and quantum machine learning:
Quantum Machine Learning

PD Dr. habil. Jeanette Miriam Lorenz
Fraunhofer Institute for Cognitive Systems IKS
&
LMU Munich
12.08.2023

Fraunhofer IKS

Fraunhofer Institute for Cognitive Systems IKS

LMU LUDWIG-MAXIMILIANS-UNIVERSITÄT MÜNCHEN

[https://www.umb.ch/blog/news/detail/quantencomputing-fuer-den-praktischen-einsatz-rueckt-naeher]

Introduction to QML

1. What is quantum machine learning?
2. Basics
3. Quantum neural networks & variants
4. Quantum reinforcement learning

Quantum machine learning (QML)

QML may have different meanings

C: classical
Q: quantum

Classical ML:

Data type \ Computer type		
	CC	CQ
	QC	QQ

Computer type

CQ – current practises: either enhancing classical ML by some quantum subroutine, or trying to resemble ML architecture by a quantum variant

History

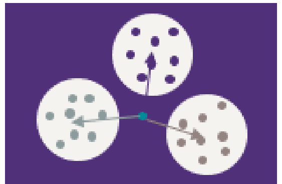
- 1980 → : Sporadic proposals to combine QC an ML
- 1995: first ideas about quantum models of neural networks → can quantum theories explain how the brain works?
- 2000s: quantum and statistical learning theory – first discussions
- 2009: Qboost algorithm on D-Wave quantum annealer
- 2013: Lloyd, Mohseni and Rebentrost mention the term 'Quantum Machine Learning'
- 2014: Peter Wittek's paper 'Quantum Machine Learning – What quantum computing means to data mining'
- 2014 → : rapidly more publications on merger of ML and QC appear, on all sorts of topics

QML algorithm

Quantum algorithms building on classical machine learning algorithms

Quantum Clustering finding

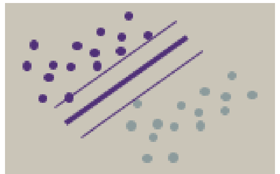
When the data is represented in a very large dimension space, it is very difficult to perform the clusetring with a classical computer. The use of quantum computers is a very good solution.



Paper : https://arxiv.org/pdf/1904.08992.pdf

Quantum Support Vector Machine

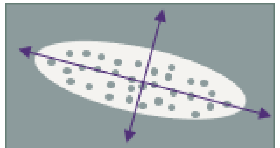
Finding the hyperplane that separates many data points that are represented in a high dimensional space is so difficult on a classical computer. on a quantum computer, it can be solved extremely efficiently.



Paper : https://dspace.mit.edu/bitstream/handle/1721.1/90391/PhysRevLett.113.130503.pdf?sequence=1&isAllowed=

Quantum PCA

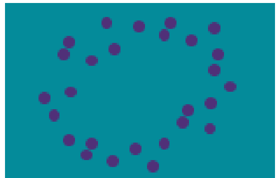
The goal of this algorithm is to find the proper axes along which to group this data. This is something that takes $O(N^2)$ on a classical computer. But in quantum version you can do it exponentially faster.



Paper : https://arxiv.org/abs/1903.03999

Feature topology

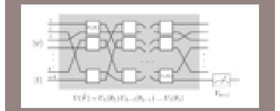
This is a method for finding the topological features of data. This problem can be mapped to a problem of finding the eigenvectors and eigenvalues of some huge, high-dimensional matrix.



Paper : https://arxiv.org/pdf/1801.06316.pdf

Quantum Deep Learning

Exciting breakthroughs may soon bring real quantum neural networks, specifically deep learning neural networks, to reality. Many research papers have shown remarkable results in quantum deep learning



Paper : http://csis.pace.edu/~ctappert/srd2018/2018PDF/a3.pd

+ Possibly hybrid versions taking the best of quantum and classical computing areas.

Source: Riahi Louriz, Highlighting Quantum Computing for Machine Learning

Theoretical speed-ups in QML

Quantum computing in particular interesting for problems showing an exponential speedup on quantum computers.

Theoretical speed-ups of quantum computing built on classical algorithms

Typically assumes perfect quantum computer, and typically ignores the problem of reading in the data

→ Very optimistic improvement

Method	Speedup	Amplitude amplification	HHL	Adiabatic	qRAM
Bayesian inference[106,107]	$O(\sqrt{N})$	Yes	Yes	No	No
Online perceptron[108]	$O(\sqrt{N})$	Yes	No	No	Optional
Least-squares fitting[9]	$O(\log N)$*	Yes	Yes	No	Yes
Classical Boltzmann machine[20]	$O(\sqrt{N})$	Yes/No	Optional/ No	No/Yes	Optional
Quantum Boltzmann machine[22,61]	$O(\log N)$*	Optional/No	No	No/Yes	No
Quantum PCA[11]	$O(\log N)$*	No	Yes	No	Optional
Quantum support vector machine[13]	$O(\log N)$*	No	Yes	No	Yes
Quantum reinforcement learning[30]	$O(\sqrt{N})$	Yes	No	No	No

*There exist important caveats that can limit the applicability of the method[51].

Source:Biamonte, Wittek, Pancotti, Rebentrost, Wiebe, Lloyd. Nature 549, 195–202(2017)

Quantum machine learning

QML one of the areas assumed to show a practical quantum advantage early

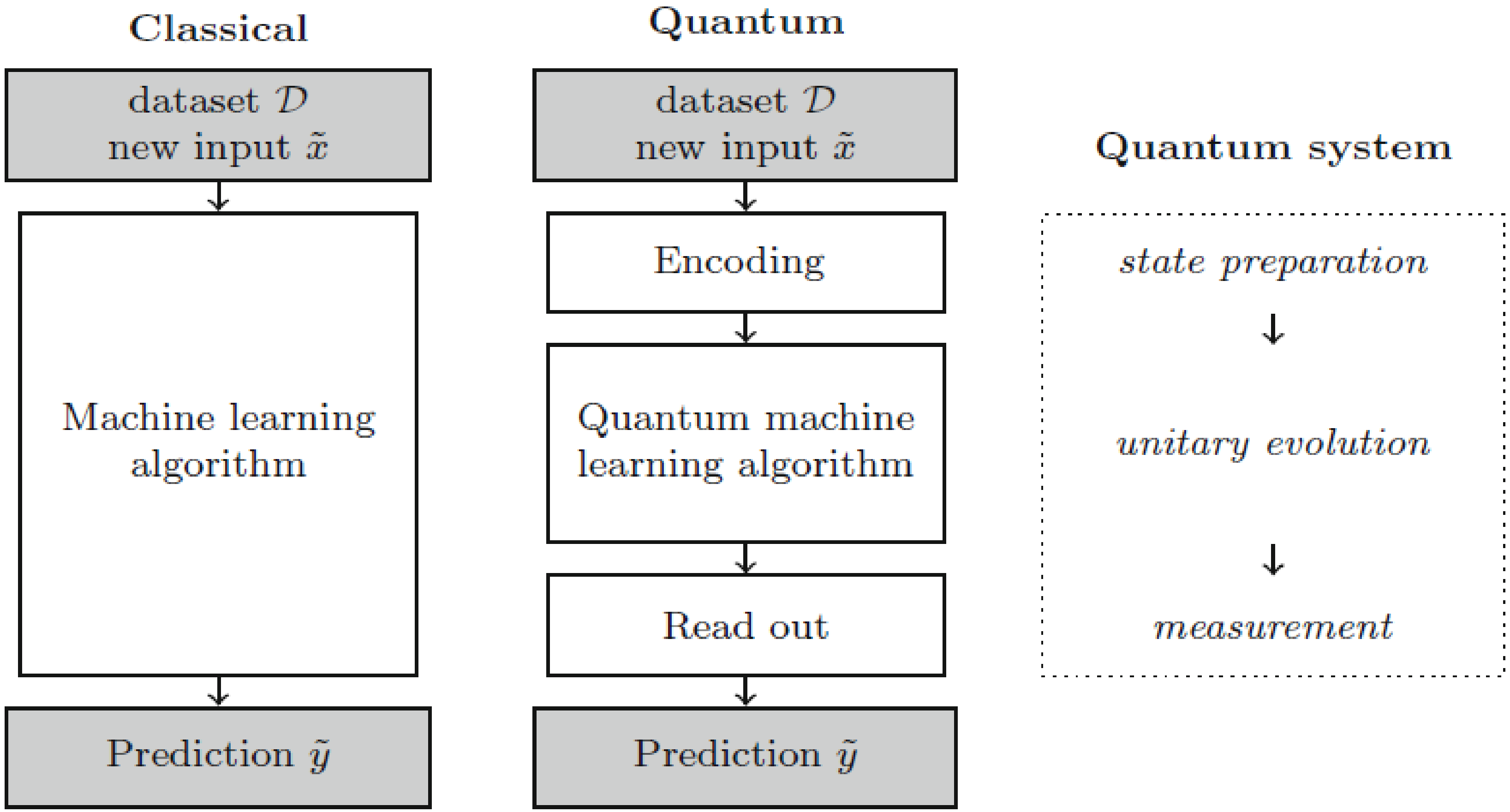
Selection of precise application field has to be done wisely, however

Comparison of QML and ML by Schuld et al.:

Property	Problems studied in quantum computing	Problems solved by machine learning
classical performance	**low** – problems are carefully selected to be provably difficult for classical computers	**high** – machine learning is applied on an industrial scale and many algorithms run in linear time in practice
size of inputs	**small** – near-term algorithms are limited by small qubit numbers, while fault-tolerant algorithms usually take short bit strings	**very large** – may be millions of tensors with millions of entries each
problem structure	**very structured** – often exhibiting a periodic structure that can be exploited by interference	**"messy"** – problems are derived from the human or "real-world" domain and naturally complex to state and analyse
theoretical accessibility	**high** – there is a large bias towards problems about which we can theoretically reason	**shifting** – theory is currently been re-built around the empirical success of deep learning
evaluating performance	**computational complexity** – the dominant measure to assess the performance of an algorithm is asymptotic runtime scaling	**practical benchmarks** – machine learning research puts a strong emphasis on empirical comparisons between methods

Source: M. Schuld, N. Killoran, Is quantum advantage the right goal for quantum machine learning?, arXiv:2203.01340 [quant-ph]

Comparison classical and quantum machine learning



Schuld, Maria, and Francesco Petruccione. Supervised Learning with Quantum Computers, Springer International Publishing AG, 2018.

The importance of the data encoding

Encoding data via a feature map

Classical data needs to be mapped from the input space to the space of the quantum system.

If an inner product is defined on the state space of the quantum system, this map is called feature map.

Different possibilities to realize feature map + very active field of research:

- Binary encoding
- Amplitude encoding
- Rotation/Angle encoding
- Higher-order encoding
- Data reuploading
- …

The encoding might result in a linear or non-linear transformation and influence the problem complexity.

Example: binary encoding

Scalar $x = (-1)^{b_s}\left(b_{\tau_{l-1}} 2^{\tau_{l-1}} + \ldots + b_0\, 2^0\right)$

-> Binary: $b = b_s b_{\tau_{l-1}} \ldots b_0$

-> Qubit representation: $| b_s b_{\tau_{l-1}} \ldots b_0 >$

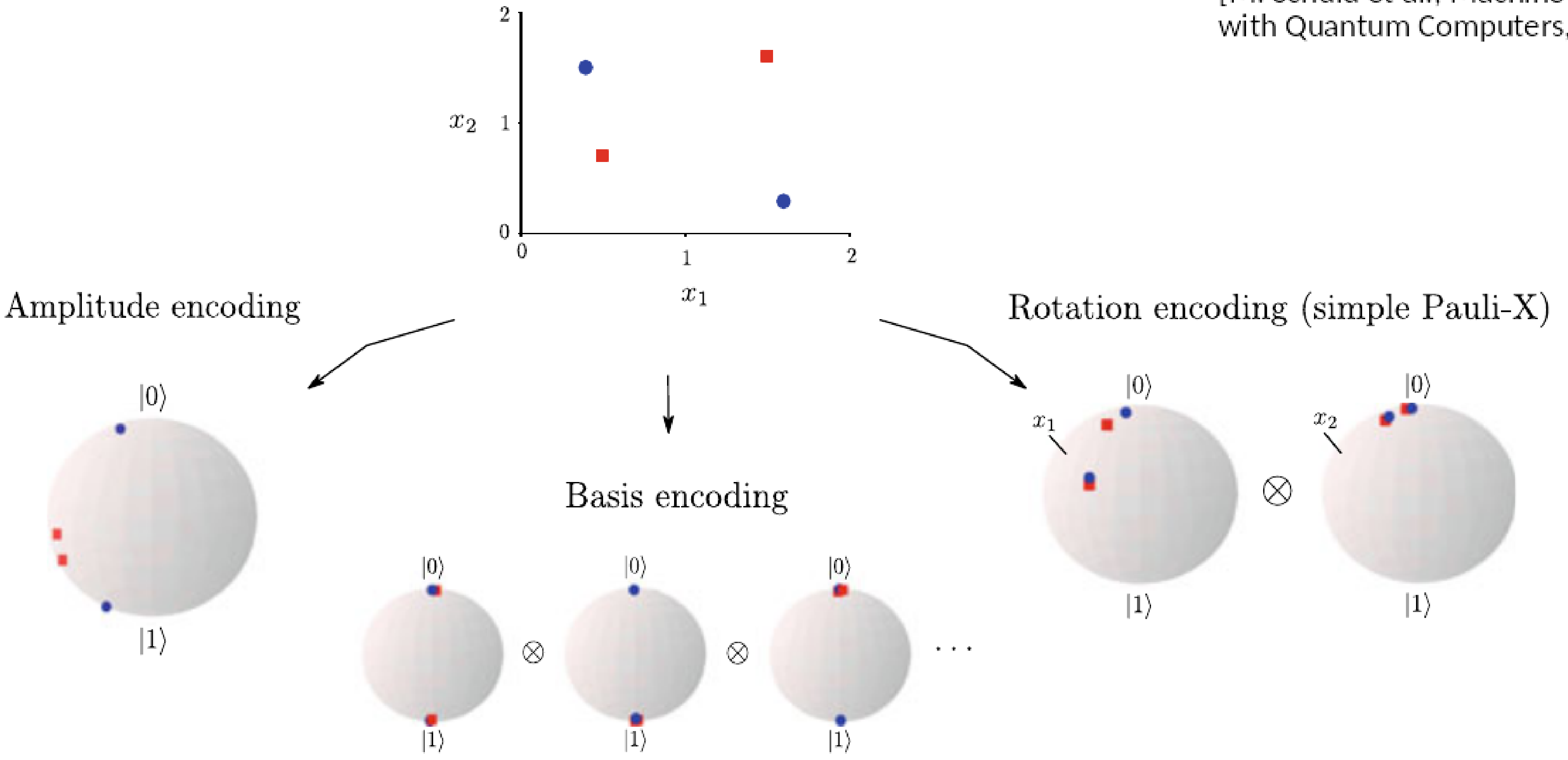
Encoding circuit via X-gates

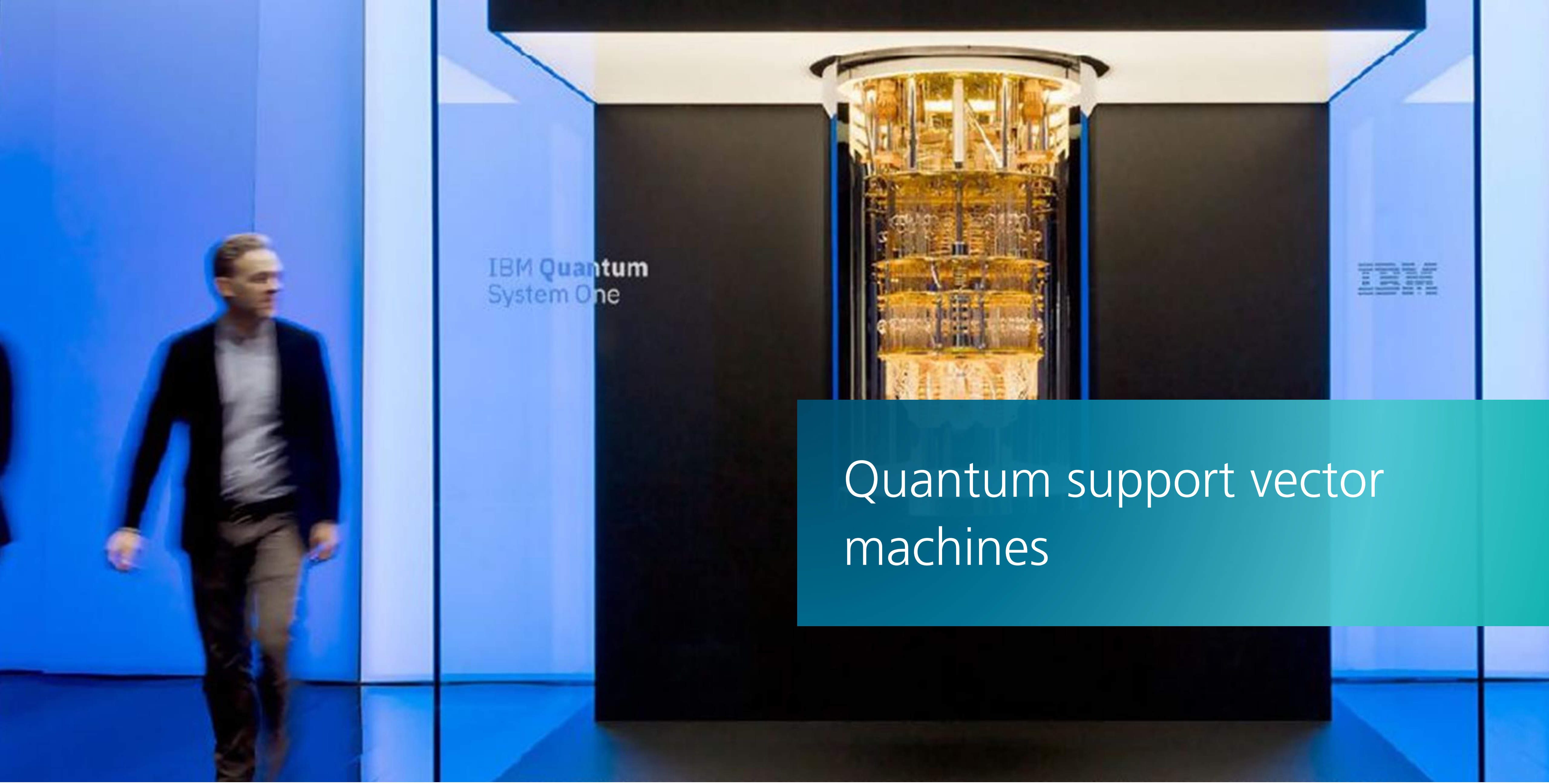
Non-linear transformation, but requires large number of qubits!

The effect of feature maps

Linear or non-linear transformation?

[M. Schuld et al., Machine Learning with Quantum Computers, Springer 2021]





Quantum support vector machines

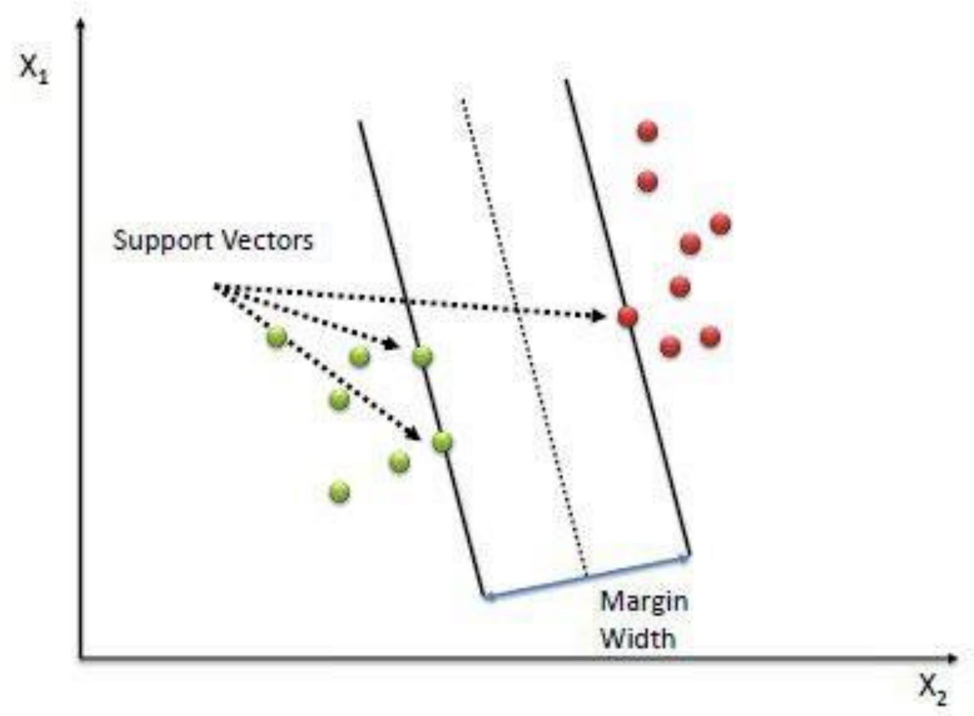
Classical support vector machines

Used for discrimination problems

Aim is to find a **hyperplane** that discriminates between two classes of **feature vectors** – done via maximising the distance between the closest data points, called **support vectors**, and the hyperplane

I.e. if x_i and y_i training data, $i = 1, \dots, n$, and y_i =+1 or = -1

Minimize $||\boldsymbol{\theta}||$ **subject to** $\boldsymbol{y_i(\theta^T x_i - b) \geq 1}$ **for** $\boldsymbol{i = 1, \dots, n}$



H Lamba, https://towardsdatascience.com/support-vector-machines-svm-c9ef22815589

Classical support vector machines – the kernel trick

In case of a non-linear separation of data points x_i → use a feature map $\phi(x_i)$ to embed into a higher dimensional space.

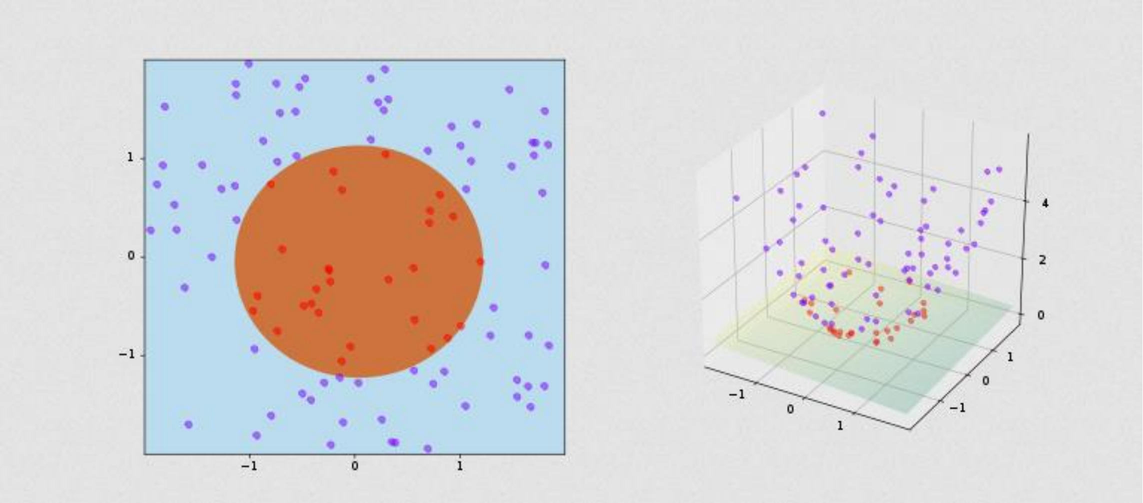
Reformulate problem to incorporate feature map.

→ Maximise

$$\sum_i \alpha_i - \frac{1}{2}\sum_{i,j} y_i y_j \alpha_i \alpha_j (\phi(x_i) \cdot \phi(x_j))$$

Subject to $0 \le \alpha_j \le C$ and $\sum_i \alpha_i y_i = 0$

The function $K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j)$ is called **kernel**.



S. Yi, wikimedia

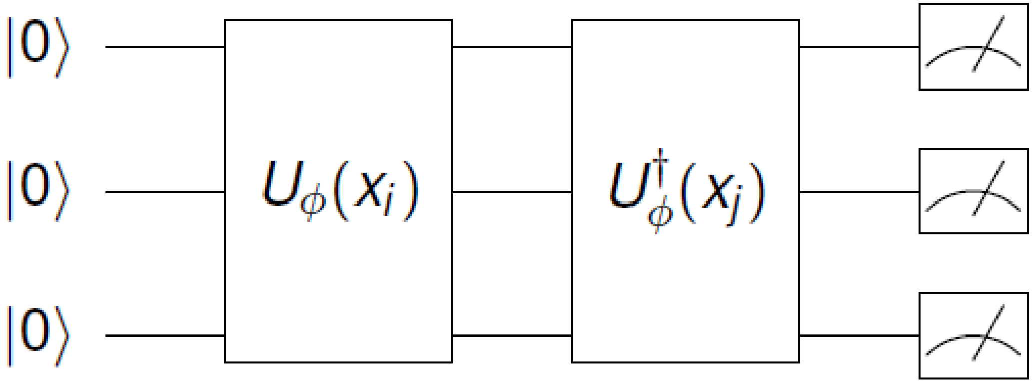
Quantum support vector machines: Kernel estimators

Quantum computers were proposed as kernel estimators in 2019 (Havlicek et al.)

Idea: Each data point x_i embeded in Hilbert space by means of a variational circuit $U_\phi(x_i)$ such that $U_\phi(x_i)|0> = |\phi(x_i)>$

Can estimate the kernel $|< \phi(x_i)|\phi(x_i) >|^2$ via running the circuit

and compute the relative frequency of |0>



Elías F. Combarro, A Practical Introduction to Quantum Computing: From Qubits to Quantum Machine Learning and Beyond

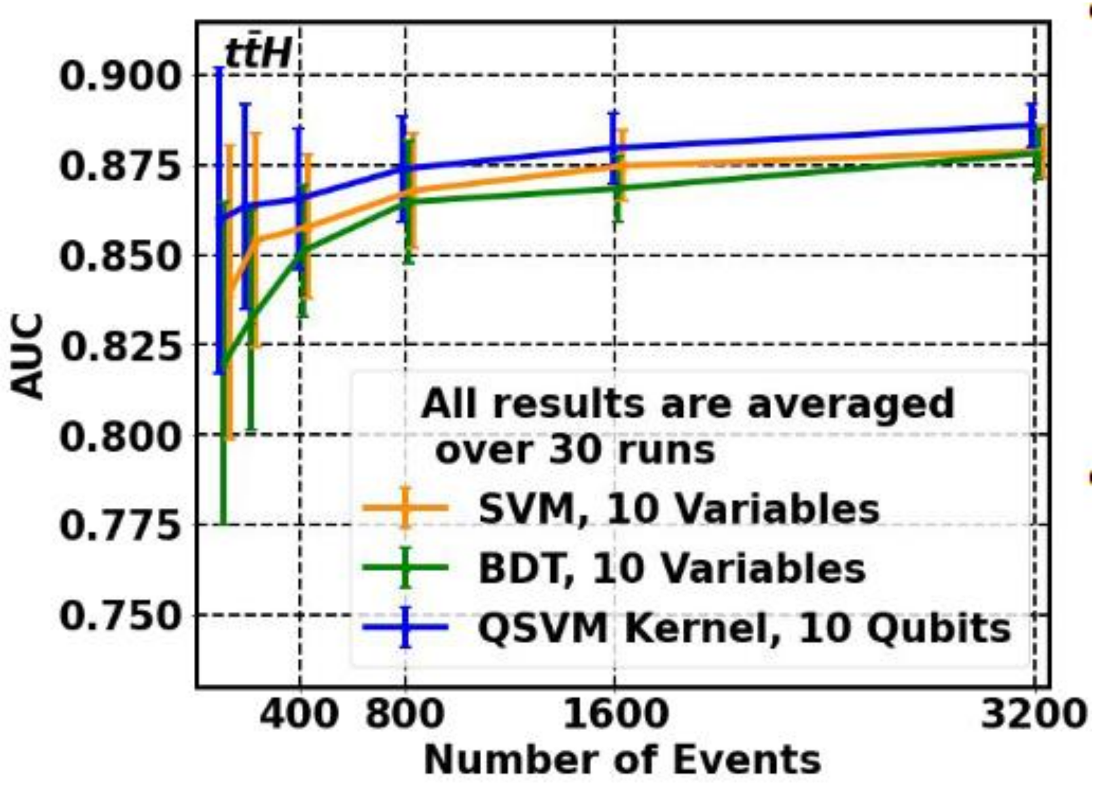
Example application

Identifying Higgs bosons in proton-proton collision data taken at the Large Hadron Collider at CERN.

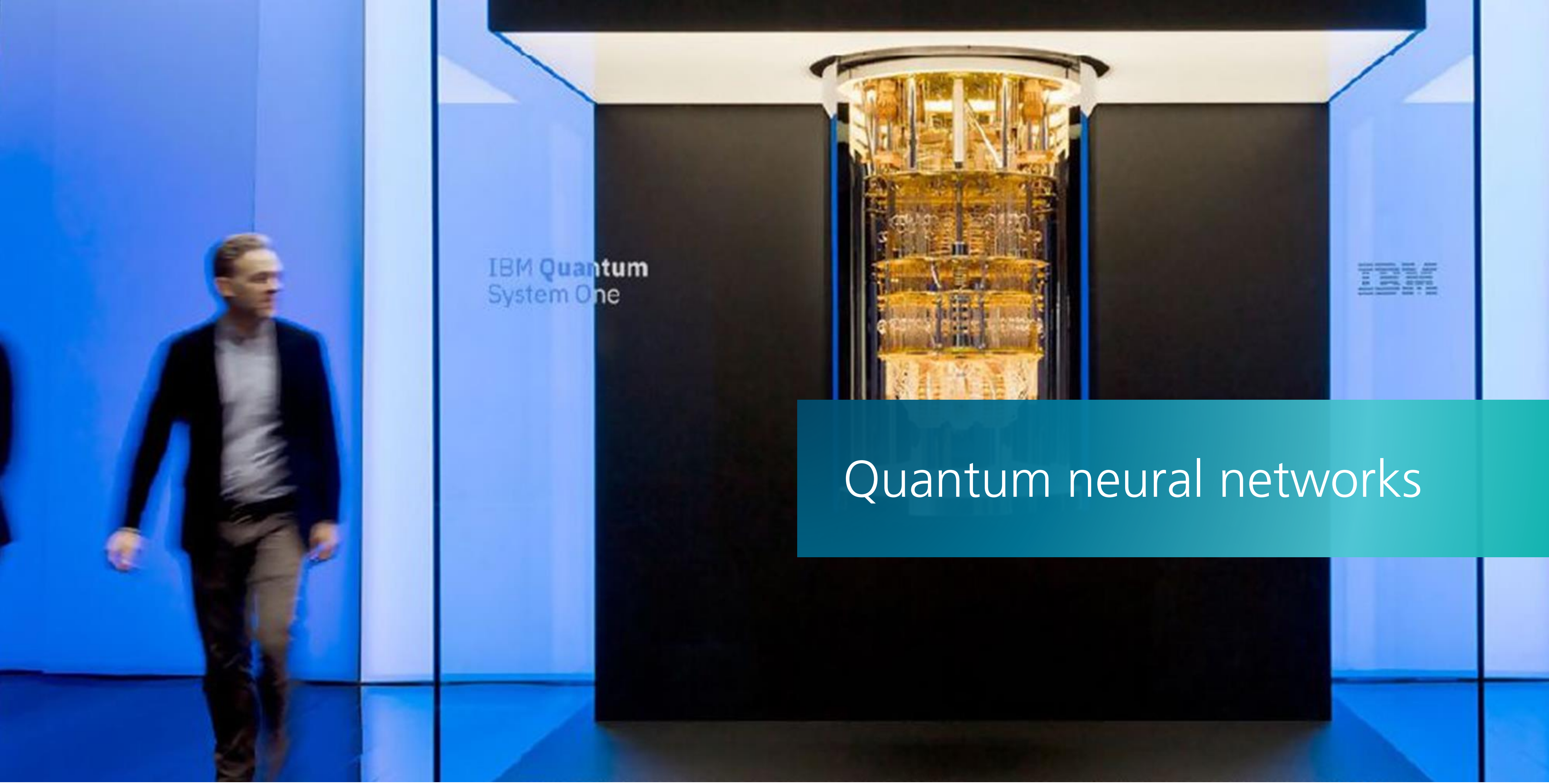
Proof-of-principle work by Sau Lan Wu and collaborators.

Comparison to classical support vector machines and boosted decision trees.

Hints that QSVM might result in better results for little training data.



Sau Lan Wu, QuantHEP Seminar



Quantum neural networks

Quantum Neural Networks

And their similarities and differences to classical neural networks

No precise definition of quantum neural networks (QNN) / different understandings – sometimes simply used for variational quantum models

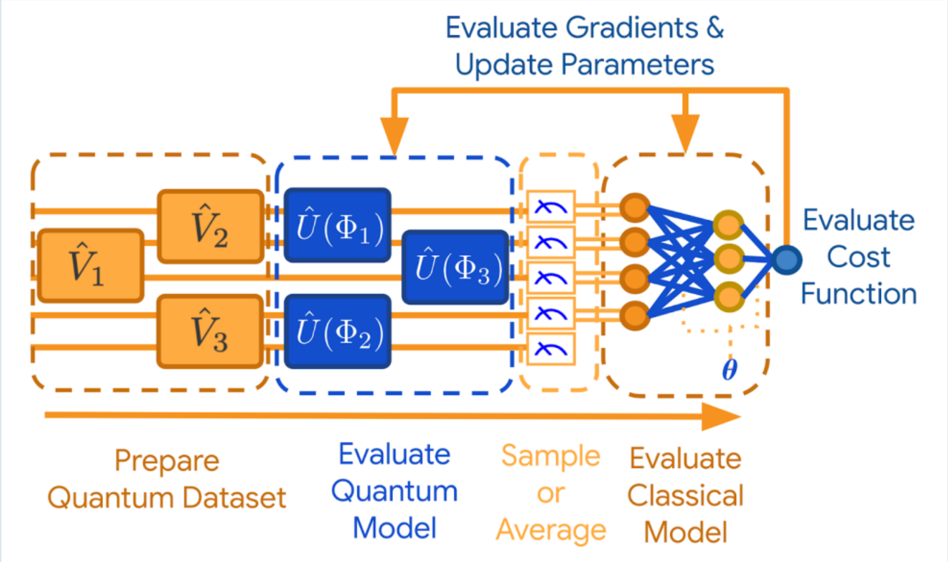
Important difference between QNNs and NNs:

Quantum circuits are linear / unitary transformations

↕

NNs are sequences of trainable linear and element-wise non-linear transformations

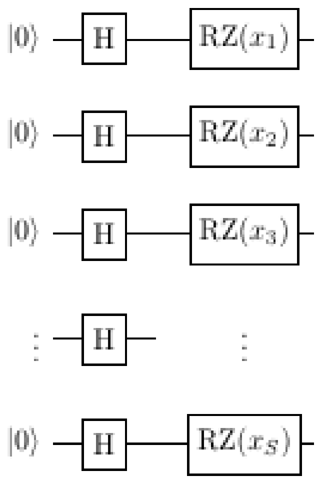
Variational quantum models:

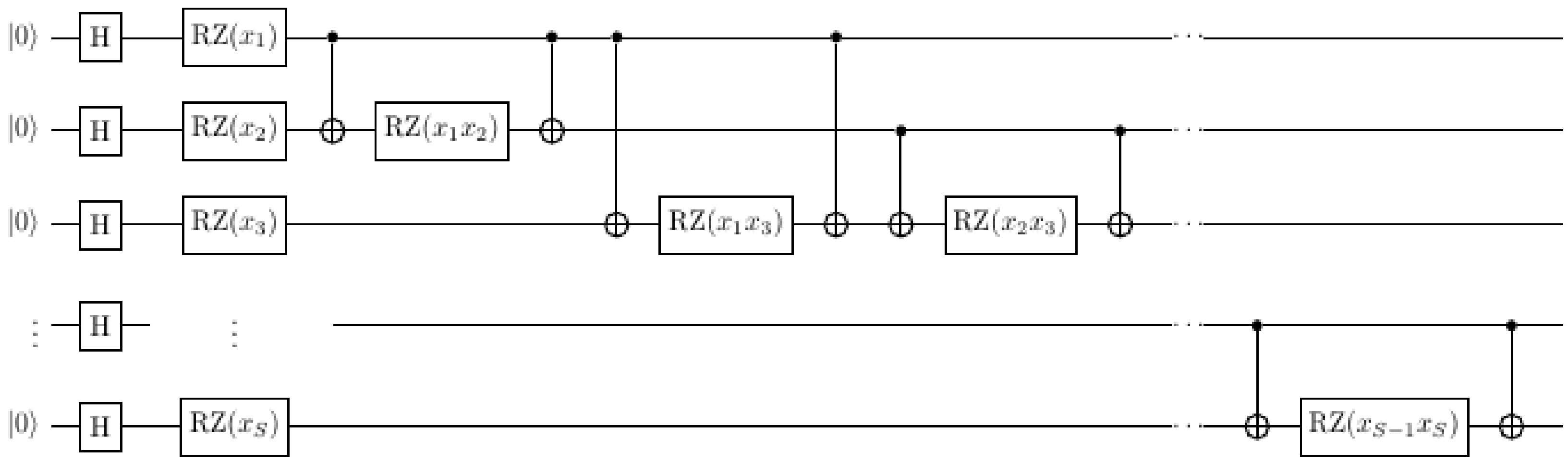


[Source: https://blog.tensorflow.org/2020/03/announcing-tensorflow-quantum-open.html]

Rotation encoding versus higher-order encoding

[A. Abbas et al, The power of quantum neural networks, Nat Comput Sci 1, 403-409 (2021)]





Achieves a non-trivial encoding, that results into a higher capacity of the model.

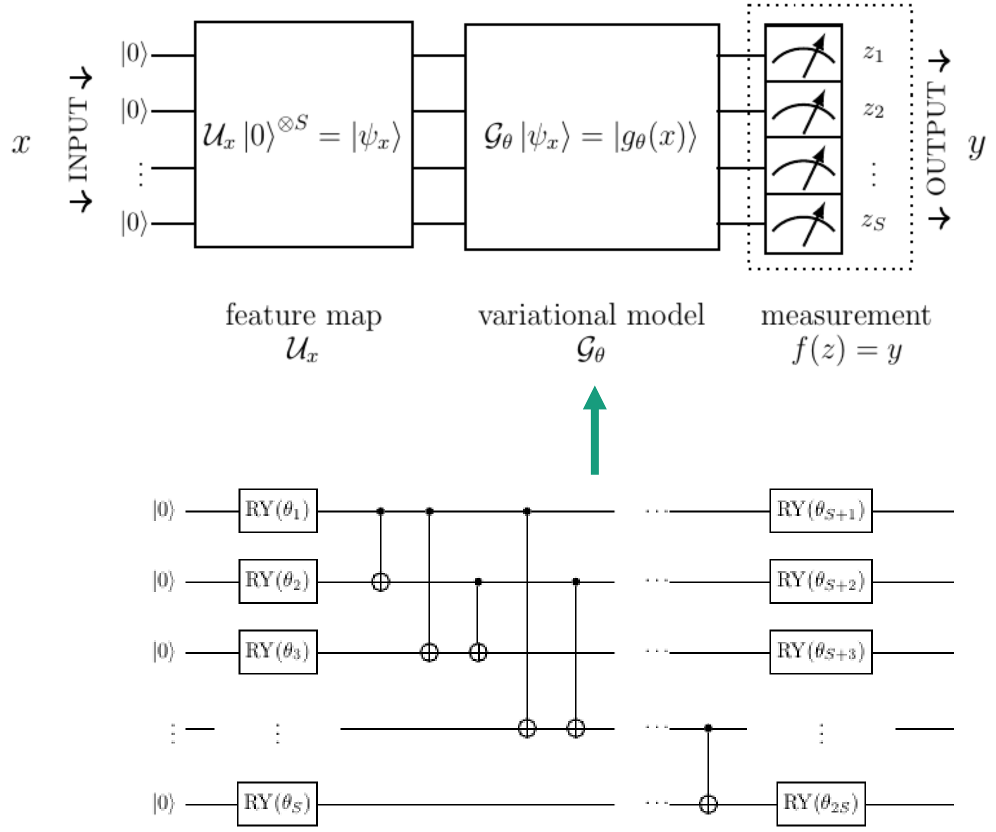
How to quantify the power of quantum neural networks

[A. Abbas et al, The power of quantum neural networks, Nat Comput Sci 1, 403-409 (2021)]

Which functions can the QNN present?

Capacity of the QNN?

Trainability?



→ **Partly assessed by (quantum) metrics:**

- Information about the trainability through the spectrum of the Fisher information matrix
- Capacity through the effective dimension
- Possibility to express functions via the expressibility
- Entanglement capability

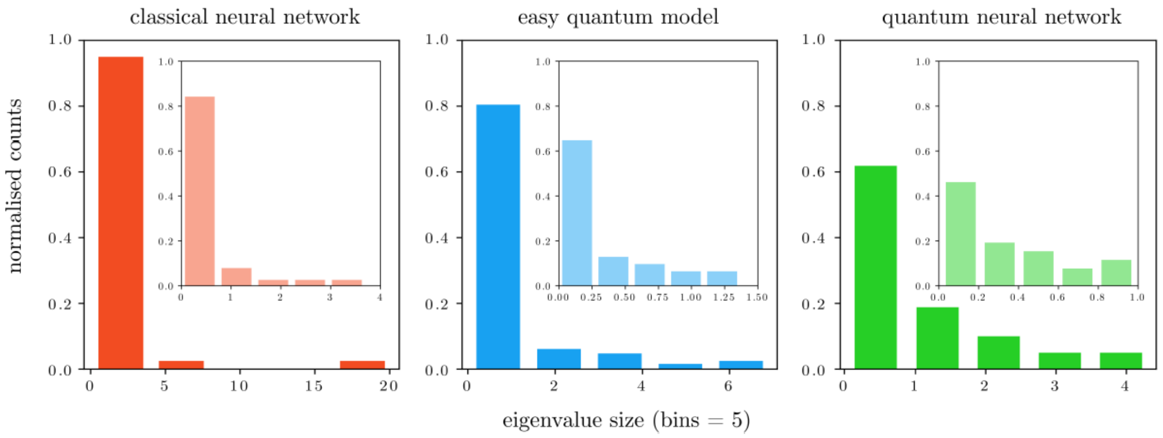
The Fisher Information Matrix

[A. Abbas et al, The power of quantum neural networks, Nat Comput Sci 1, 403-409 (2021)]

Measures the amount of information that a random variable X carries in dependency on an unknown parameter θ

If a neural network is defined as **statistical model** $p(x, y; \theta) = p\ (y|x; \theta)p(x)$, the Fisher information matrix is:

$$F(\theta) = \mathbb{E}_{(x,y)\sim p}[\frac{\partial}{\partial\theta} \log p(x, y; \theta) \frac{\partial}{\partial\theta} \log p(x, y; \theta)^T]$$



→ Measure for the sensitivity of the output of the neural network to movements in the parameter space.

The distribution of the eigenvalues hint to the trainability of (Q)NNs (better trainability if eigenvalues more evenly distributed)

How to quantify the power of quantum neural networks

The effective dimension

[A. Abbas et al, The power of quantum neural networks, Nat Comput Sci 1, 403-409 (2021)]

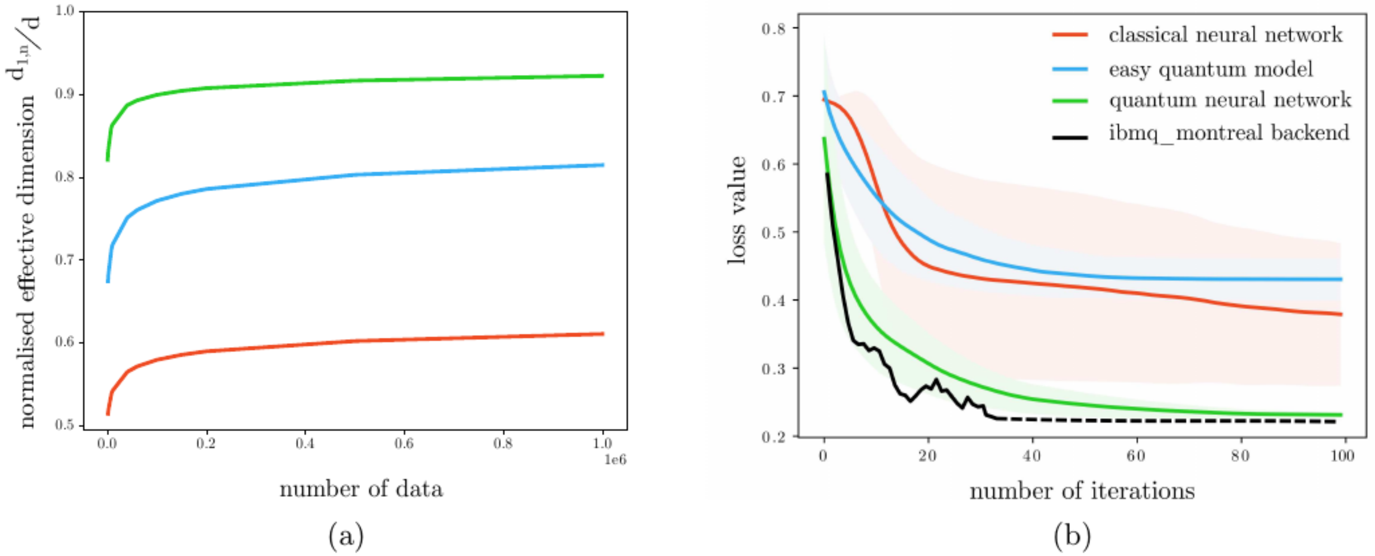
Purpose: to **estimate the size** that a model occupies in the model space, where the Fisher information matrix serves as the metrics.

Advantage: applicable for both quantum and classical models

Defined on a statistical model $M_{\boldsymbol{\theta}} \coloneqq \{p(\cdot,\cdot;\theta)\colon \theta \in \boldsymbol{\Theta}\}$ as:

$$d_{\gamma,n}(M_{\boldsymbol{\theta}}) \coloneqq 2\frac{\frac{1}{V_{\boldsymbol{\theta}}}\int_{\boldsymbol{\theta}} \sqrt{\det\left(id_d + \frac{\gamma n}{2\pi \log n}\hat{F}(\theta)\right)}\, d\theta}{\log\left(\frac{\gamma n}{2\pi \log n}\right)}$$

$$\hat{F}_{ij} \coloneqq d\frac{V_\theta}{\int_\theta tr(\mathrm{F}(\theta))\, d\theta} F_{ij}(\theta)$$



- n > 1, no. of data samples
- $\Theta \in \mathrm{R}^d$ is the parameter space
- V_θ is the volume of the parameter space
- $\gamma \in [0,1]$ hyperparameter for boundedness
- $\hat{F}_{ij}$ is the normalized Fisher matrix

How can QNNs be trained?

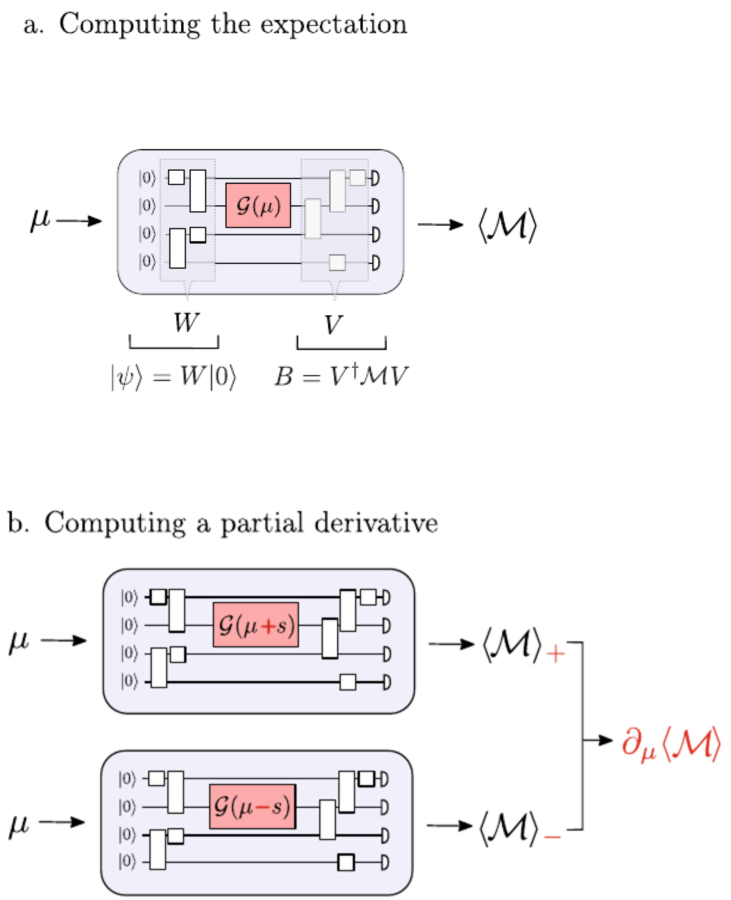
Training a QNN requires an update of the variational parameters

Achieved by minimizing a cost function by an automatic differentiation

-> requires the calculation of the partial derivative of the differentiable function $f(\theta)$ in $\partial_\mu C(\mu) = \frac{\partial C}{\partial f_\mu} \cdot \partial_\mu f_\mu$

Calculation typically by parameter-shift rule:

- Calculation by the **identity** $\partial_\mu f_\mu = \sum_i \alpha_i f_{\mu+s_i}$ with α_i and s_i real scalar values
- **Not an approximate solution, but exact!**
- Applies in particular for rotation gates



[M. Schuld et al., Machine Learning with Quantum Computers, Springer 2021]

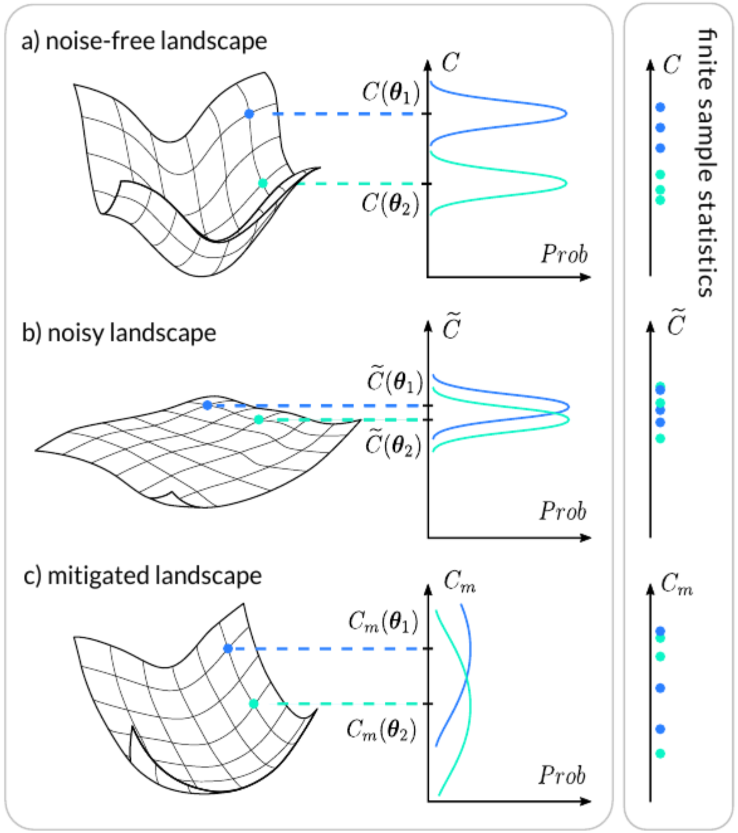
Issues preventing trainability

The trainability is affected by the shape of the cost landscape presented to the optimizer.

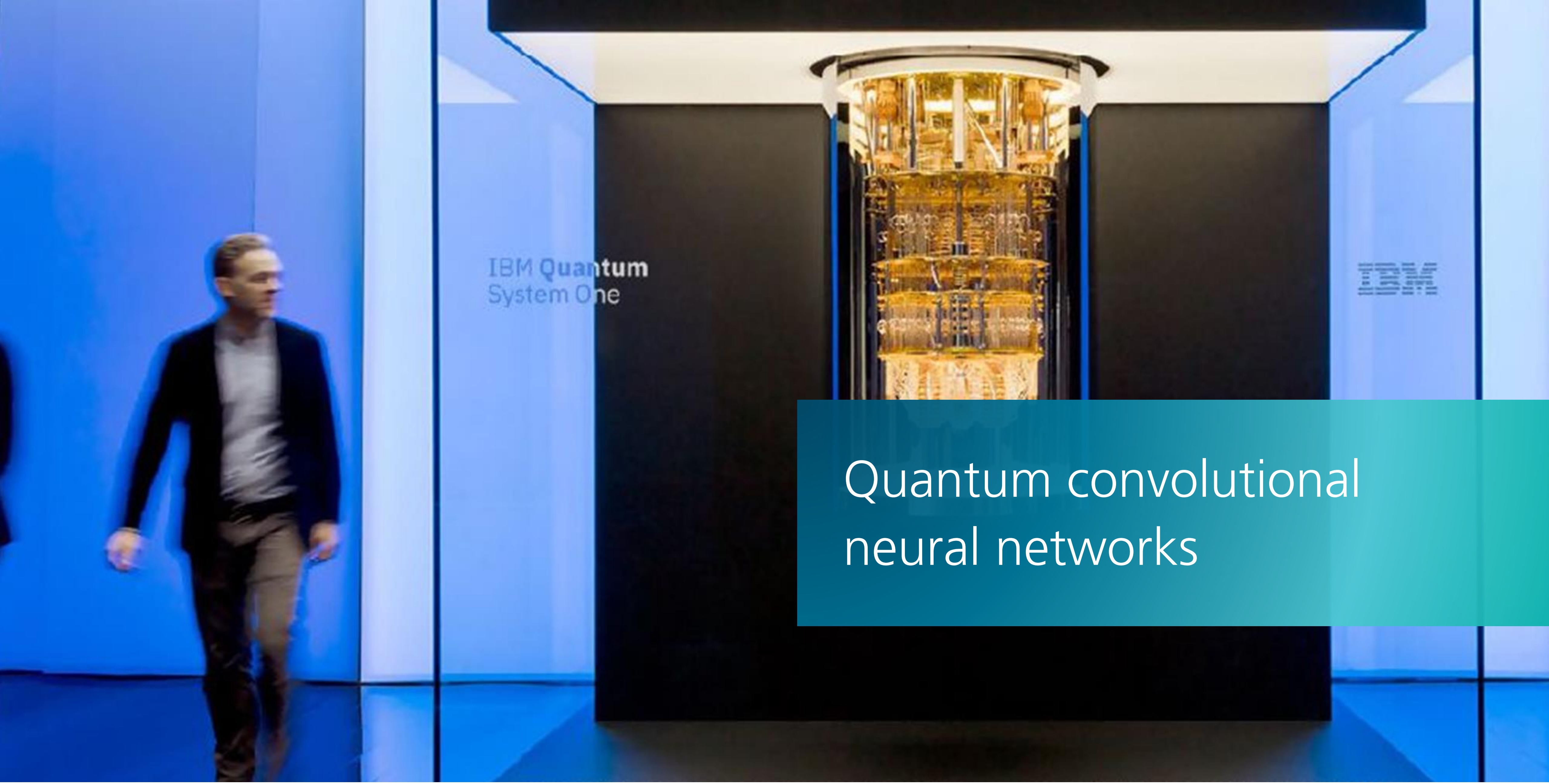
This shape may exhibit problems:

- **Barren plateaus** (vanishing gradients)
  - Slow and expensive optimization
  - Danger of random walk

- **Noise** present in NISQ devices may also affect shape
  - Possibly exponential many shots required
  - Quantum error mitigation might be helpful

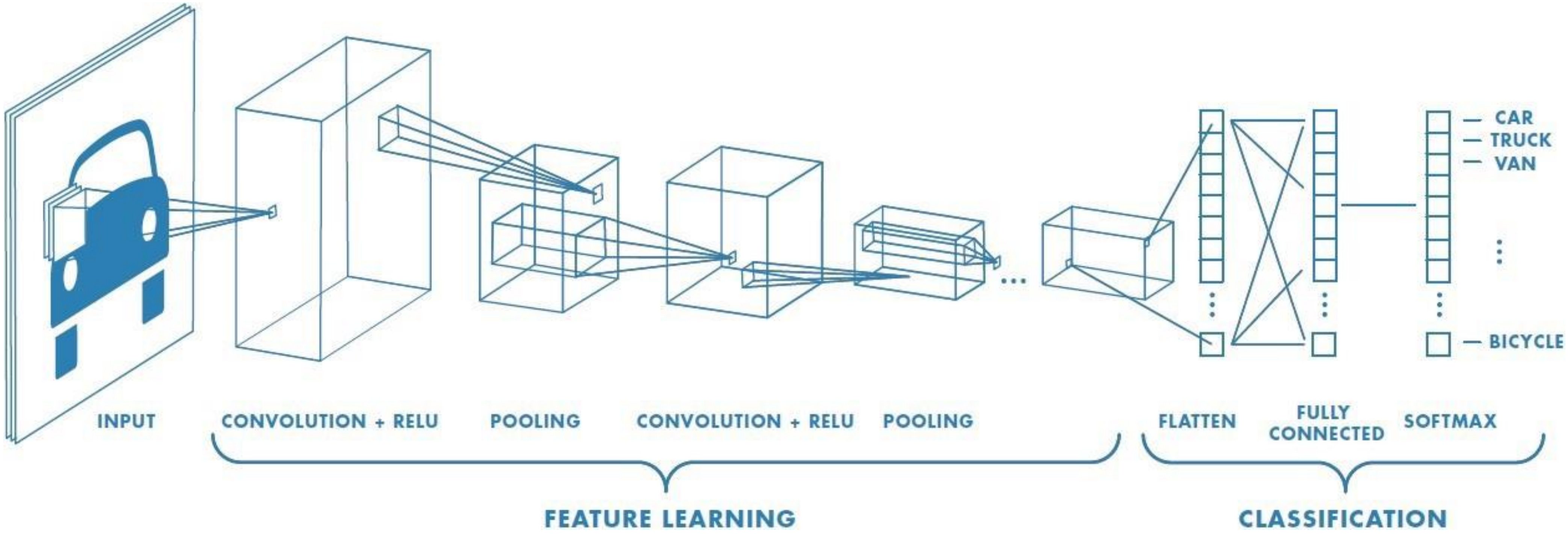


[S. Wang et al., Can Error Mitigation Improve Trainability of Noisy Variational Quantum Algorithms?, arXiv:2109.01051 [quant-ph]]

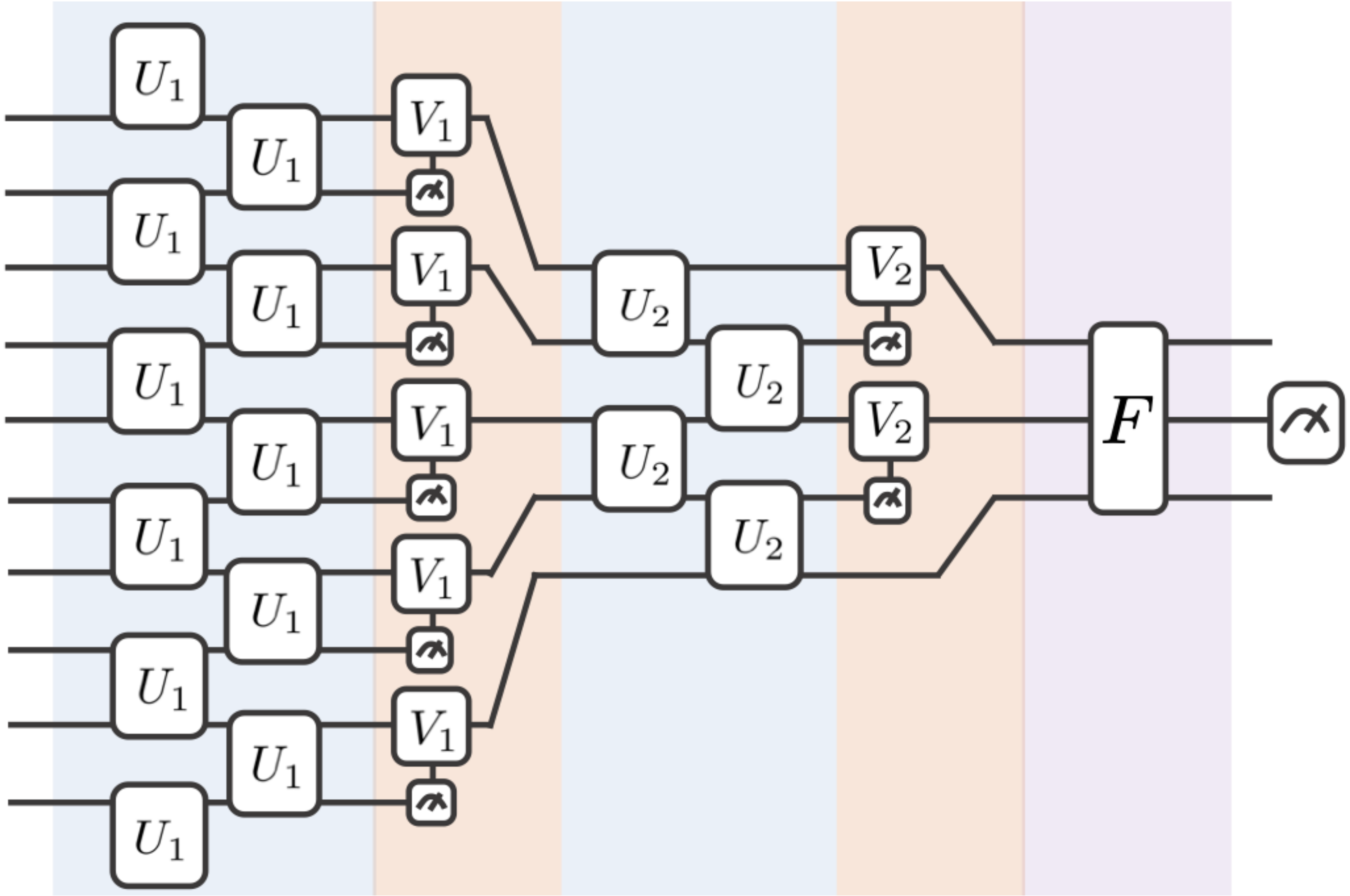


Quantum convolutional neural networks

Convolutional Neural Network



Quantum Convolutional Neural Network



Source: Iris Cong, Soonwon Choi, Mikhail D. Lukin.
"Quantum Convolutional Neural Networks"
arxiv:1810.03787, 2018.

Generalization with less data

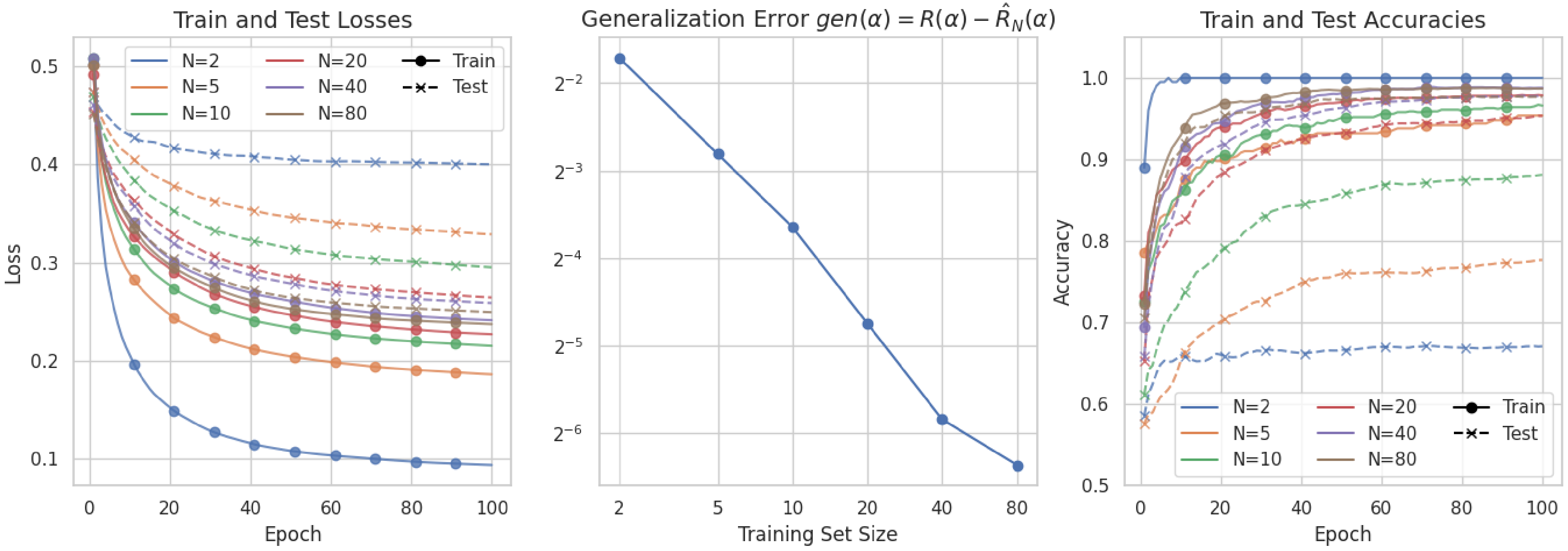
The **generalization error** is given by the difference between the expected true loss of a (Q)ML model and the average loss over the training dataset: $gen(\alpha) = R(\alpha) - \hat{R}_S(\alpha)$

Specifically for QCNN particularly favorable: $gen(\alpha) \sim O\left(\sqrt{\frac{T \log MT}{N}}\right)$

for T parametrized local quantum channels, M gates and N the training set size

Sources: Matthias C. Caro, Hsin-Yuan Huang, M. Cerezo, Kunal Sharma, Andrew Sornborger, Lukasz Cincio, Patrick J. Coles. “Generalization in quantum machine learning from few training data” arxiv:2111.05292, 2021.

Generalization with less data

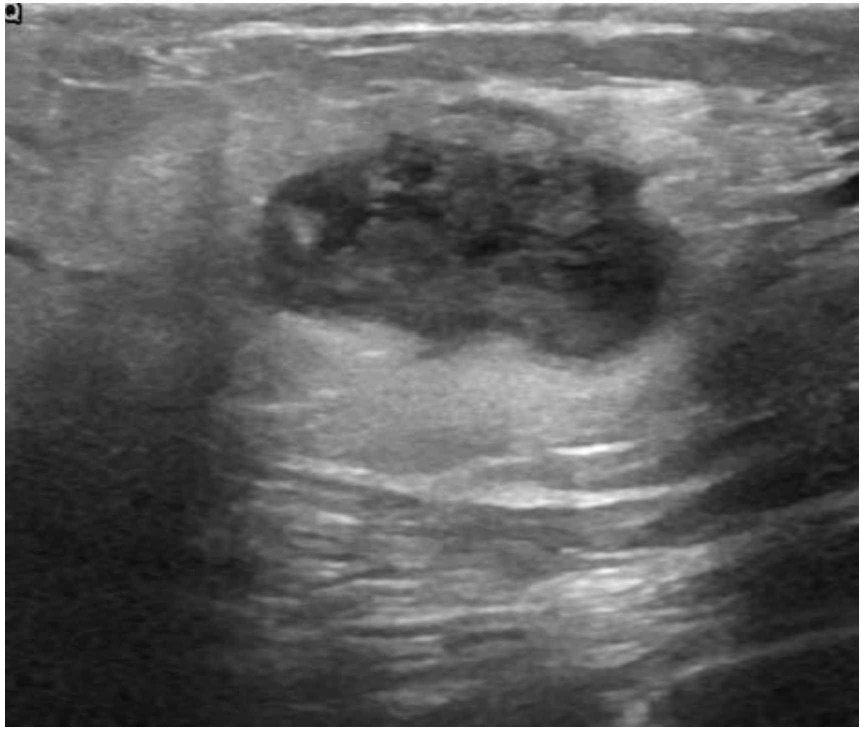


Source: Korbinian Kottmann, Luis Mantilla Calderon, Maurice Weber, Generalization in QML from few training data, Pennylane demonstration, https://pennylane.ai/qml/demos/tutorial_learning_few_data.html

Application example: Reliable QC-assisted AI for medical classification tasks

Identify lesions or nodules as benign or malign

Malign breast lesion



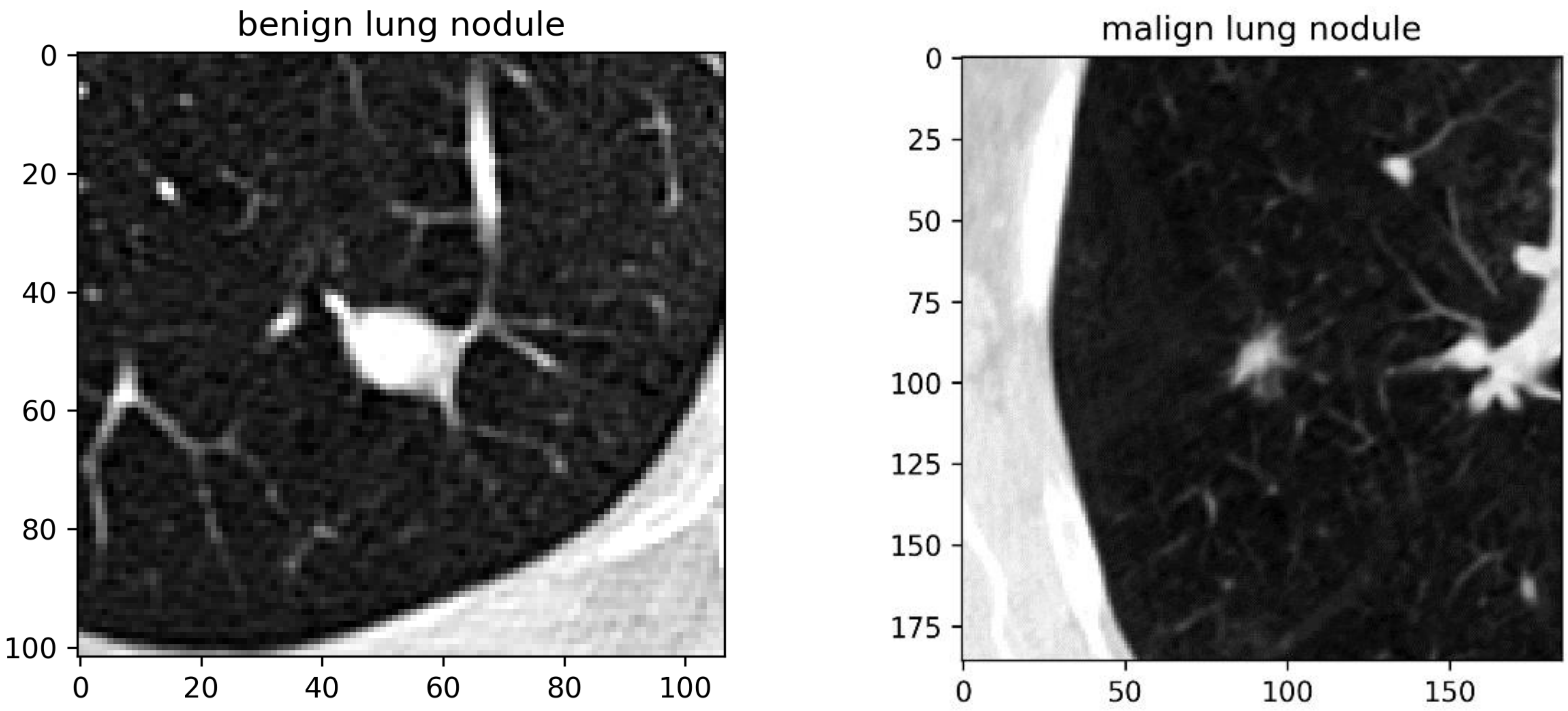
Benign breast lesion

Source: W. Al-Dhabyani, et al, "Dataset of breast ultrasound images". Data Brief, vol 28, pp 104863, 2020

  Public information

Application example: Reliable QC-assisted AI for medical classification tasks

Identify lesions or nodules as benign or malign



Reliable QC-assisted AI for medical classification tasks

Context: Artificial intelligence increases in importance in the medical diagnosis process (e.g. in imaging).

Challenges: Image data is expensive, complex and only available in small numbers (10^2 -10^3),

The decision process needs to be comprehensible and reliable.
→ Classical methods need large training datasets.

Target: Improvement of the medical classification tasks via hybrid, quantum computing-assisted machine learning methods.

Expected improvement: QC-assisted methods might result into a faster training of the algorithms - in particular in situations with little training data

Quantum-computing assisted machine learning

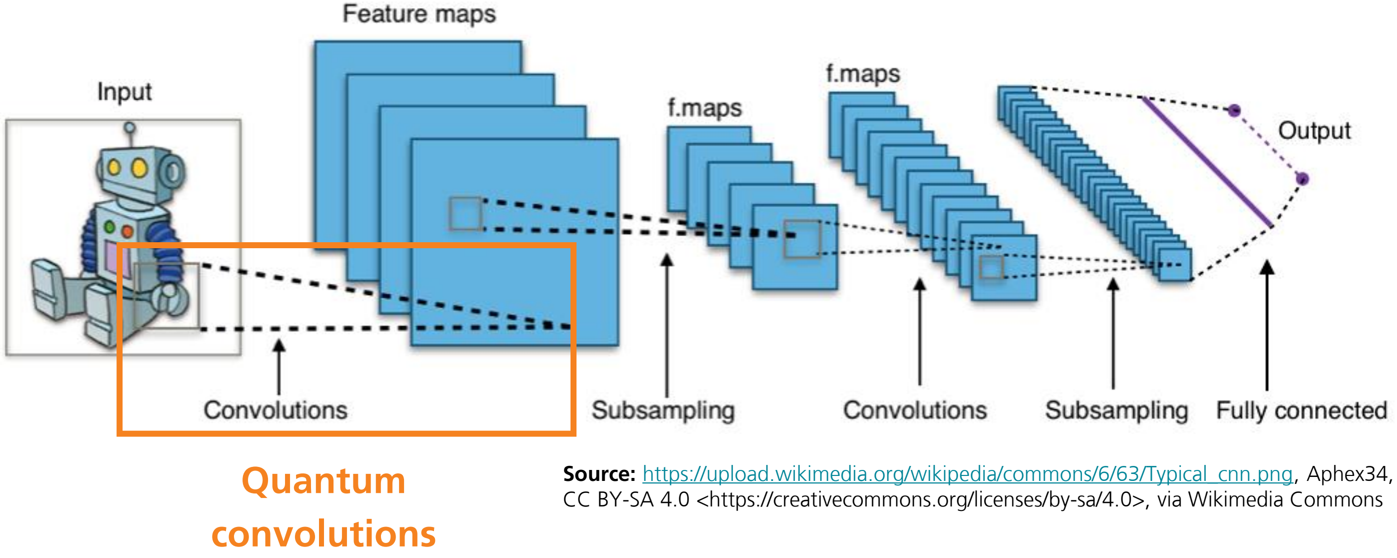
Hybrid quantum-classical convolutional neural networks

Idea: Replace some of the convolutional layers by quantum convolutional layers

QCCNNs promise to be better suited for situations with **little training data** → potentially more precise and faster training convergence

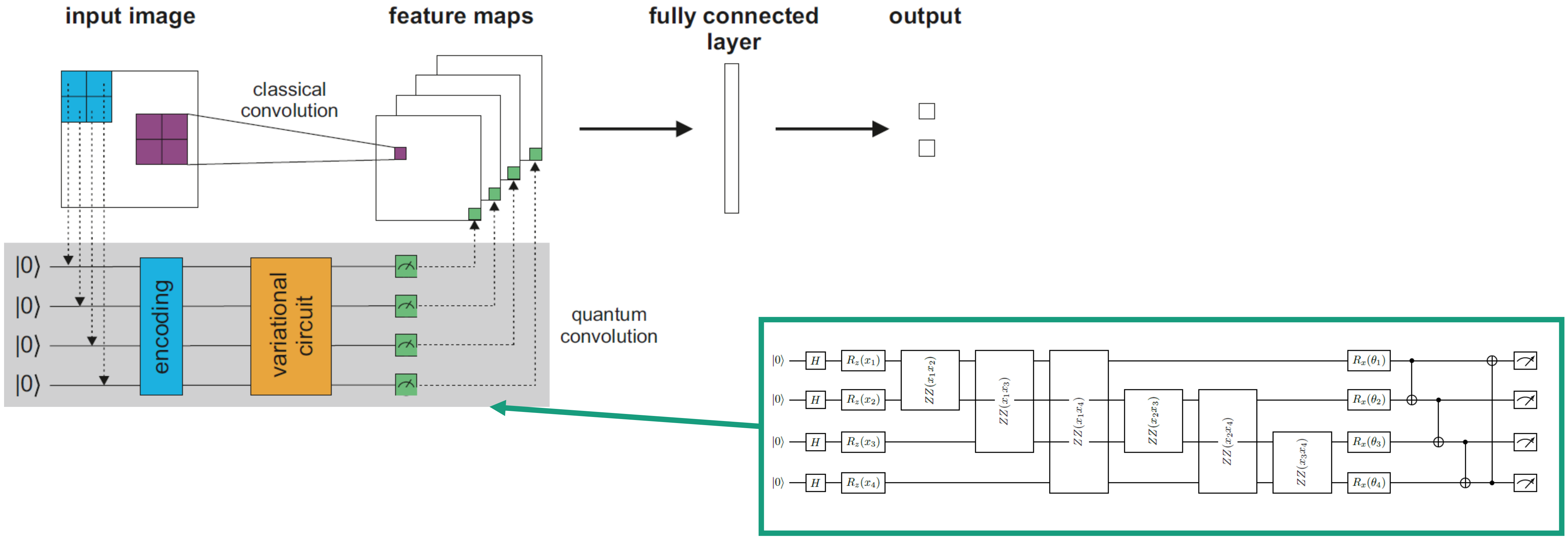
Hybrid ansatz → possible to execute on current or soon-available NISQ quantum computers

Hybrid QCCNN:



Source: https://upload.wikimedia.org/wikipedia/commons/6/63/Typical_cnn.png, Aphex34, CC BY-SA 4.0 <https://creativecommons.org/licenses/by-sa/4.0>, via Wikimedia Commons

Hybrid quantum-classical convolutional neural networks



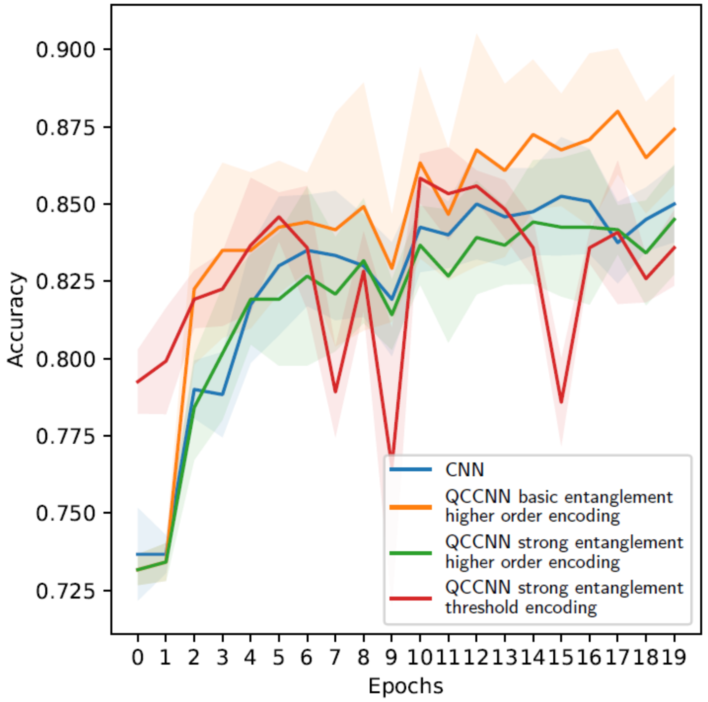
Classification example

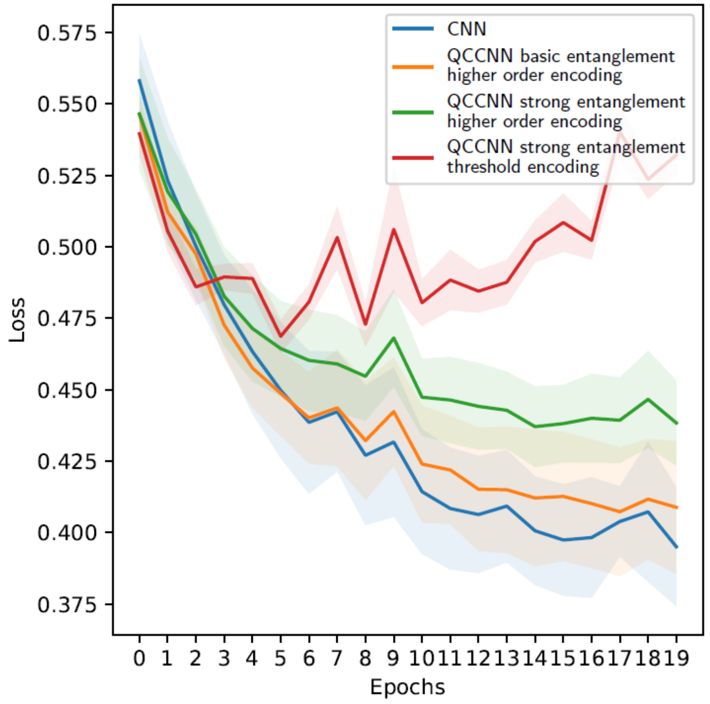
Identifying types of lesions in ultrasound images of the breast

Achieved promising performance of a hybrid quantum-classical convolutional neural networks, encouraging further studies.

More complex data encoding schemes more promising than simple encoding schemes.

All configurations used a series of one- and two-qubit gates.





Paper: A. Matic, M. Monnet, J. M. Lorenz, B. Schachtner, T. Messerer, Quantum-classical convolutional neural networks in radiological image classification, arXiv:2204.12390 [quant-ph], https://ieeexplore.ieee.org/document/9951255

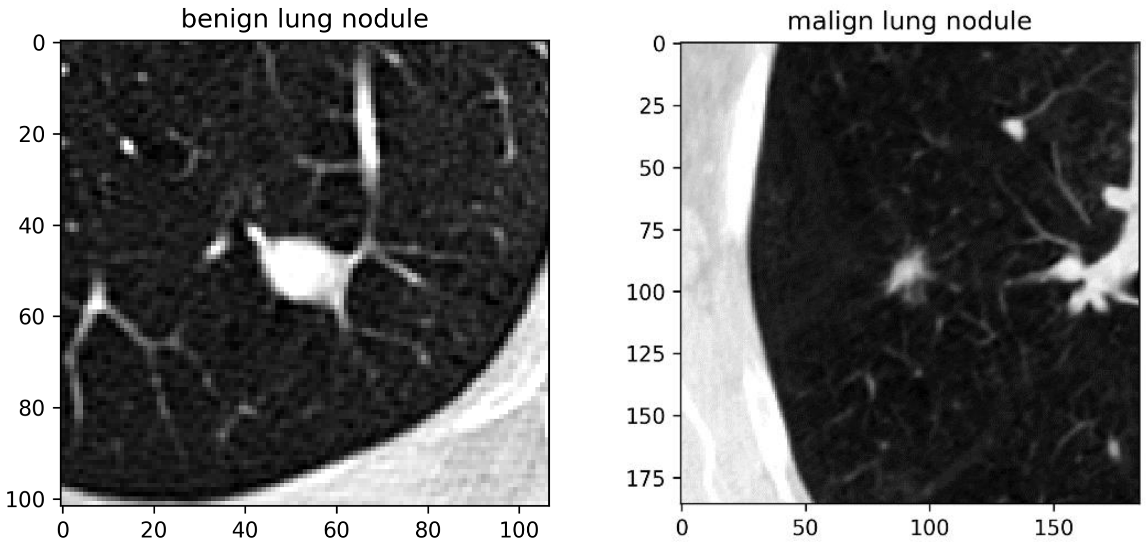
Hybrid QCCNN on 3D medical data

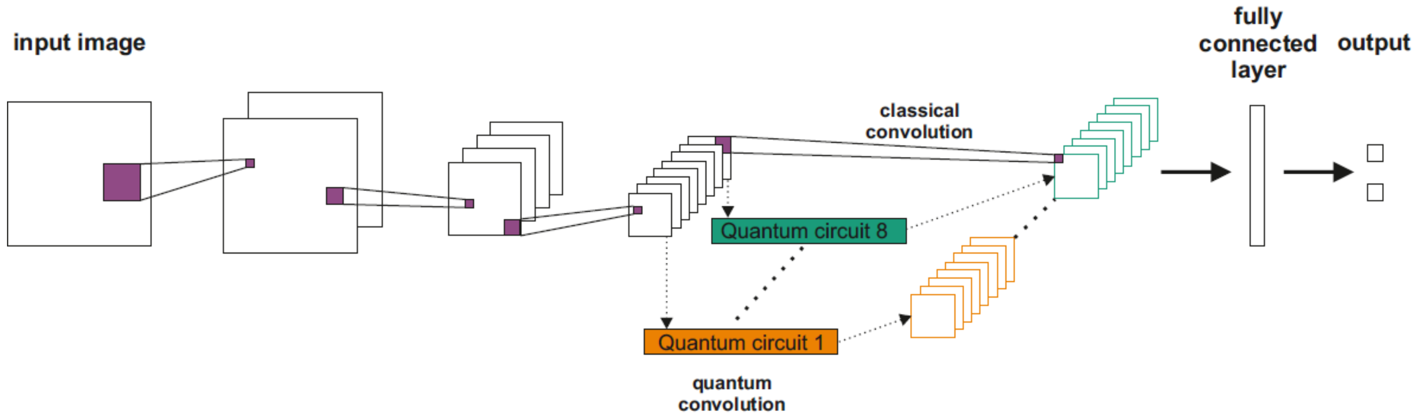
3D CT scans of the lung – classification of potentially malign nodules

Image size 128 x 128 x 64

Data compression required before quantum convolutional layer can be used

- Achieved by a sequence of classical convolutional layers with ReLU activation functions before a quantum convolutional layer
- 8 quantum kernels in parallel required within the quantum convolutional layer



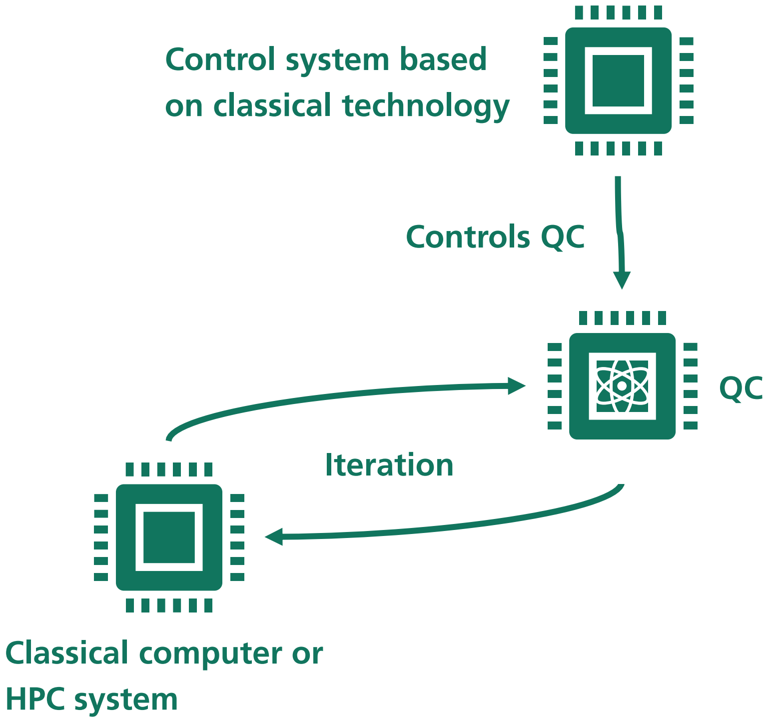


Considerations for QC hardware and software

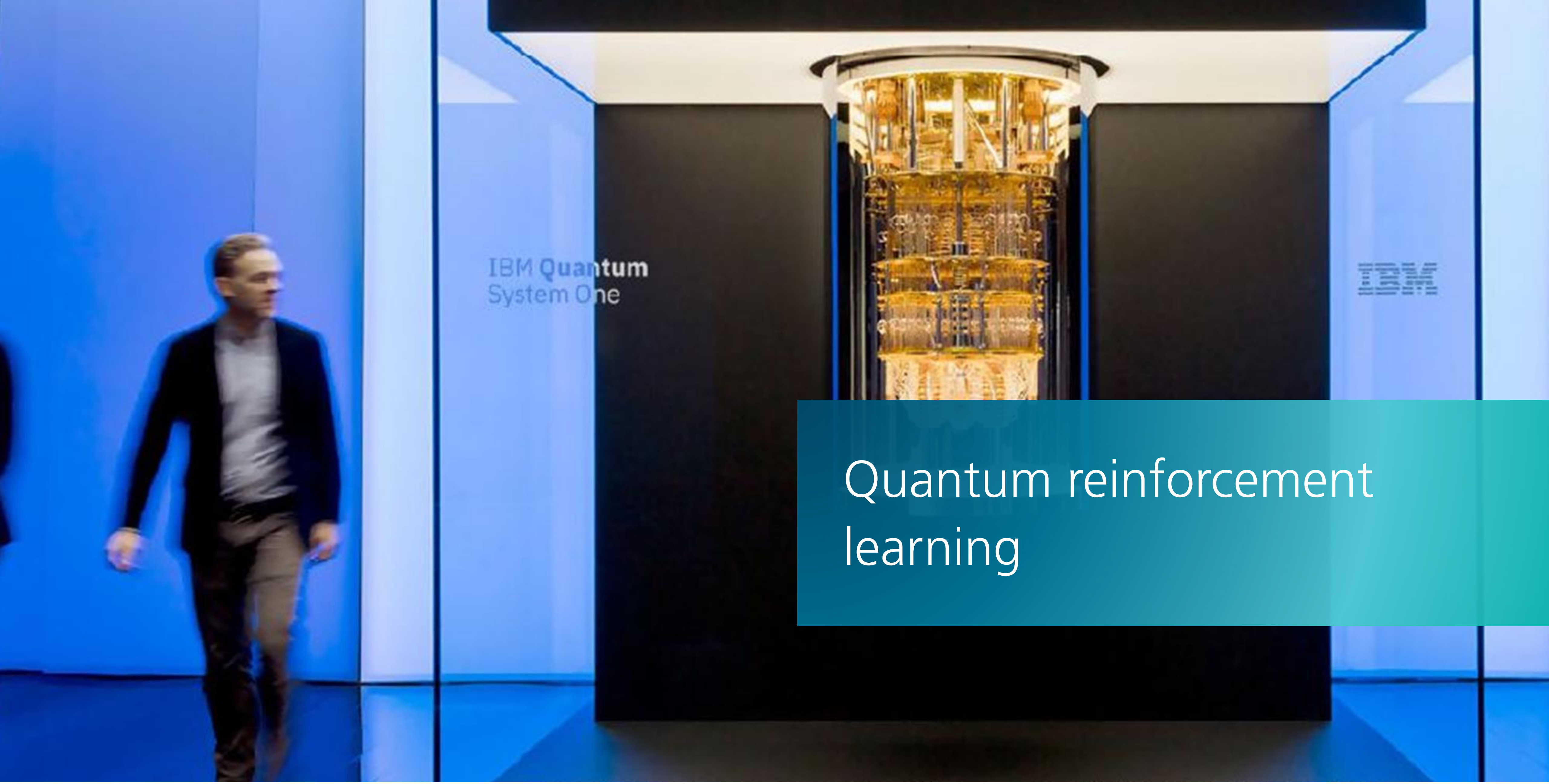
Current QC hardware limited in size, connectivity and affected by noise:

- Only small QML architectures possible (small number of qubits, small depth) – not full QCNN
- Smaller architectures can (theoretically) be processed by current hardware
- In practise for the specified 2D architecture required:
  ~ 1000 images (28x28) -> 1000 * 14 * 14 circuits = 196000 circuits
  * Shots
  * Number of training iterations
  = ~ 3.9 billions circuit evaluations
  + backpropagation procedure

- Without runtime environments: execution time is a couple of months
- With runtime, further downscaling of images, and a couple of tricks ~ hours – days



Improvements in runtime environments and excution times needed!



Quantum reinforcement learning

Reinforcement learning

Learning in reaction to an environment

Reinforcement learning (RL) is the third direction besides supervised and unsupervised learning

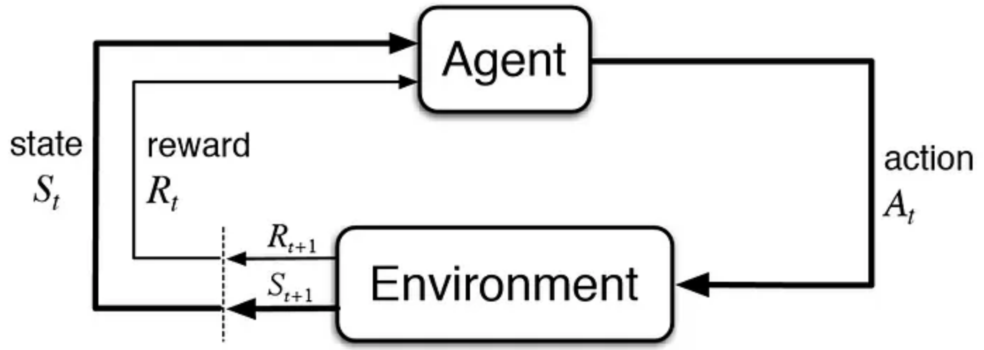
Main ideas:

- An agent learns in an interactive environment from the feedback it receives on actions/experiences
- Connected to **Markov Decision Processes**

- Conventions:
  - **States** S
  - Possible **actions** A
  - **Rewards** R

- At each learning operation/time step, the agent interacts + observes the reaction of the environment
- Probability to reach a new state s‘ and reward r‘:

$$p(s',r|s,a) = P_r\{S_t = s', R_t = r \mid S_{t-1} = s, A_{t-1} = a\}$$



[R. Sutor, Reinforcement Learning - An Introduction, http://incompleteideas.net/book/the-book-2nd.html]

Demonstration

Quantum reinforcment learning

Augment reinforcement learning by a QAlg

Quantum-inspired algorithms

Leverage amplitude estimation or Grover's algorithm

PQC-Based Function Approximations

Use QNNs to replace classical NNs approximating functions in a RL algorithm

NISQ compatible, but advantage unclear

Insert quantum subroutines

Similar to the idea of inserting PQC, but if using fault-tolerant QAlg possibly speed guarantees

Full quantum formulation

The whole algorithm runs on a quantum computer

  Public information

Better modeling decisions by the use of QC

[Source: D. Niraula, Quantum deep reinforcement learning for clinical decision support in oncology: application to adaptive radiotherapy, Scientific Reports volume 11, Article number: 23545 (2021)]

Differences in patient's genetics + physiology may alter responses to radiotherapy treatments.

→ Ideally adapt therapy to responses and enable a personalized medicine.

Decision under uncertainties.

→ Quantum algorithms expected to being able to well mimic this.

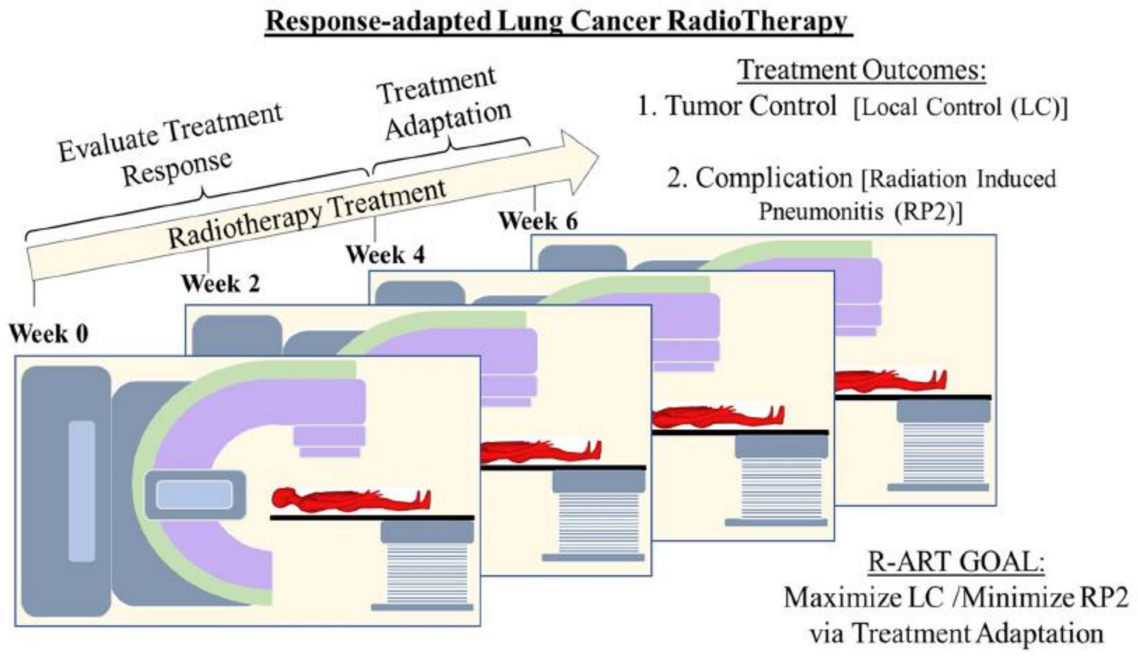
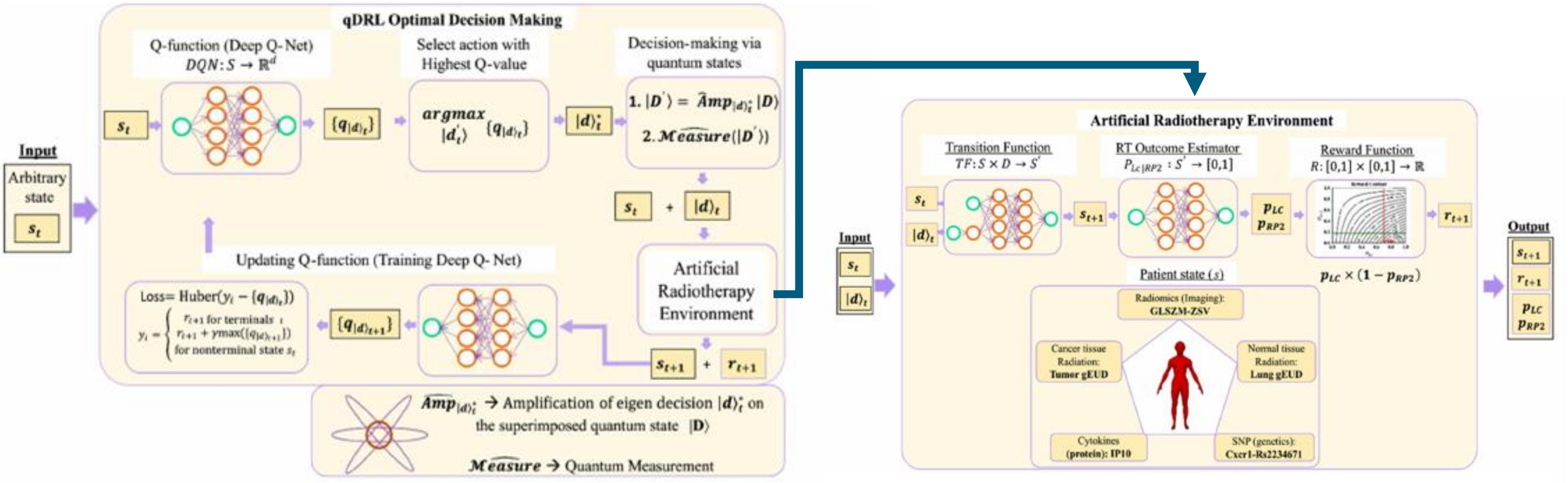


Figure 1. Schematics of response-adapted lung cancer radiotherapy. Response-adapted radiotherapy evaluates treatment response in the first two-thirds (week 1 to week 4) of the treatment period and then makes necessary adaptation in the last third (week 5 to week 6), with the goal of optimizing the treatment plan. For the case of lung cancer, optimization translates into maximizing tumor (local) control (LC) and minimizing radiation-induced pneumonitis of grade 2 or higher (RP2).

The algorithm

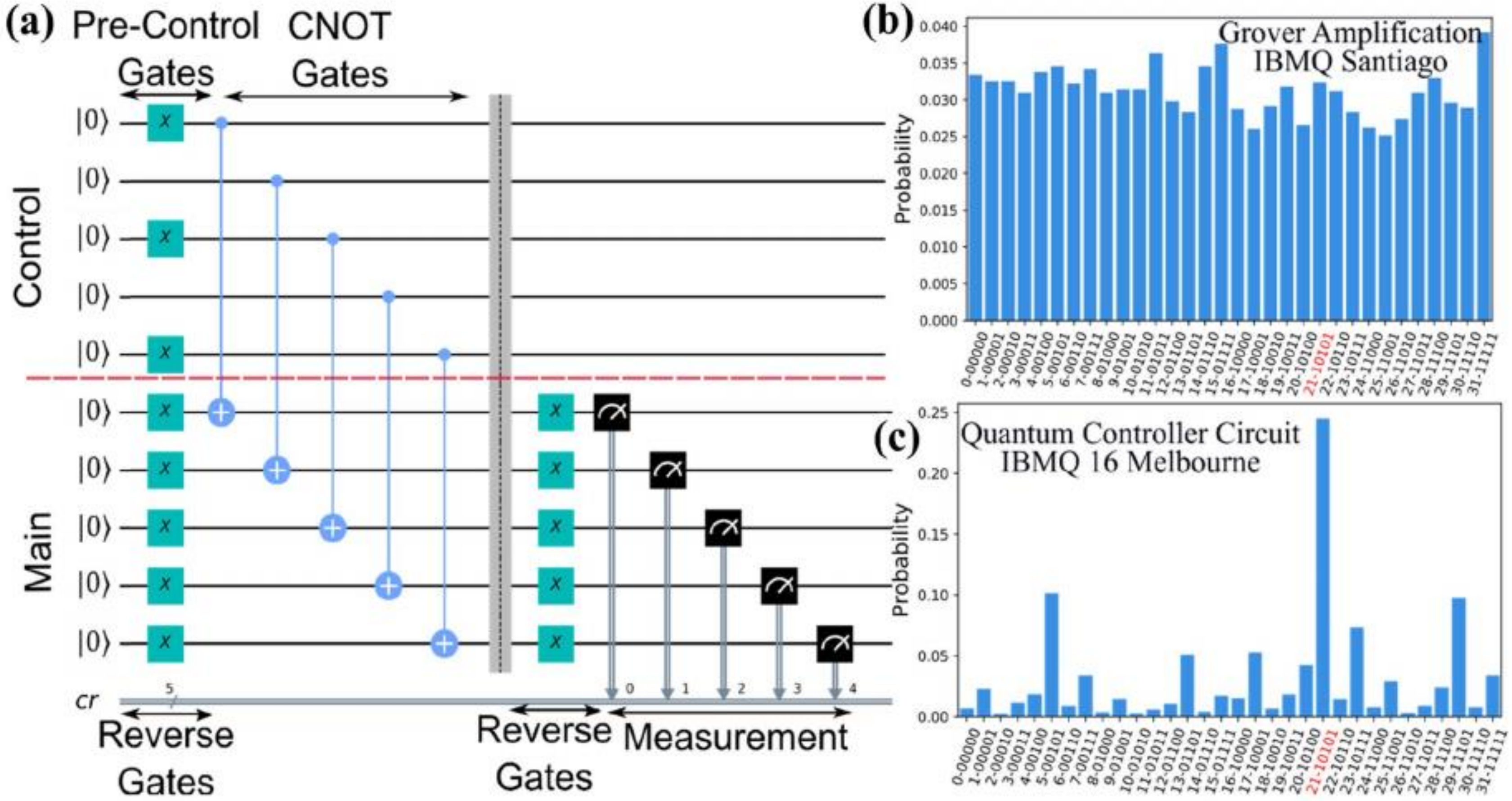
Construction of a quantum deep reinforcement learning framework, where a quantum decision process is paired with a model-based deep q-learning algorithm.



[Source: D. Niraula, Quantum deep reinforcement learning for clinical decision support in oncology: application to adaptive radiotherapy, Scientific Reports volume 11, Article number: 23545 (2021)]

The quantum controller circuit + results

[Source: D. Niraula, Quantum deep reinforcement learning for clinical decision support in oncology: application to adaptive radiotherapy, Scientific Reports volume 11, Article number: 23545 (2021)]



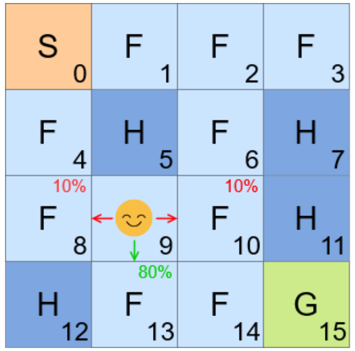
Quantum reinforcement learning

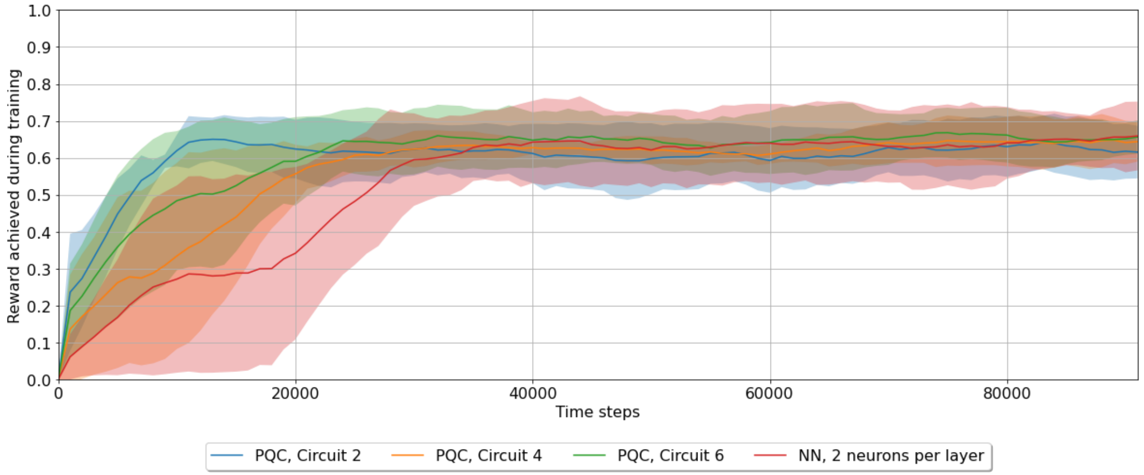
To accelerate reactions to environments

Possible application fields:
- Optimization of industrial production chains
- Guided robots

Advantage of using QRL:
- Less trainable parameters required than in classical reinforcement learning for achieving comparable or better performance.
- Less training steps required/faster time to solution.

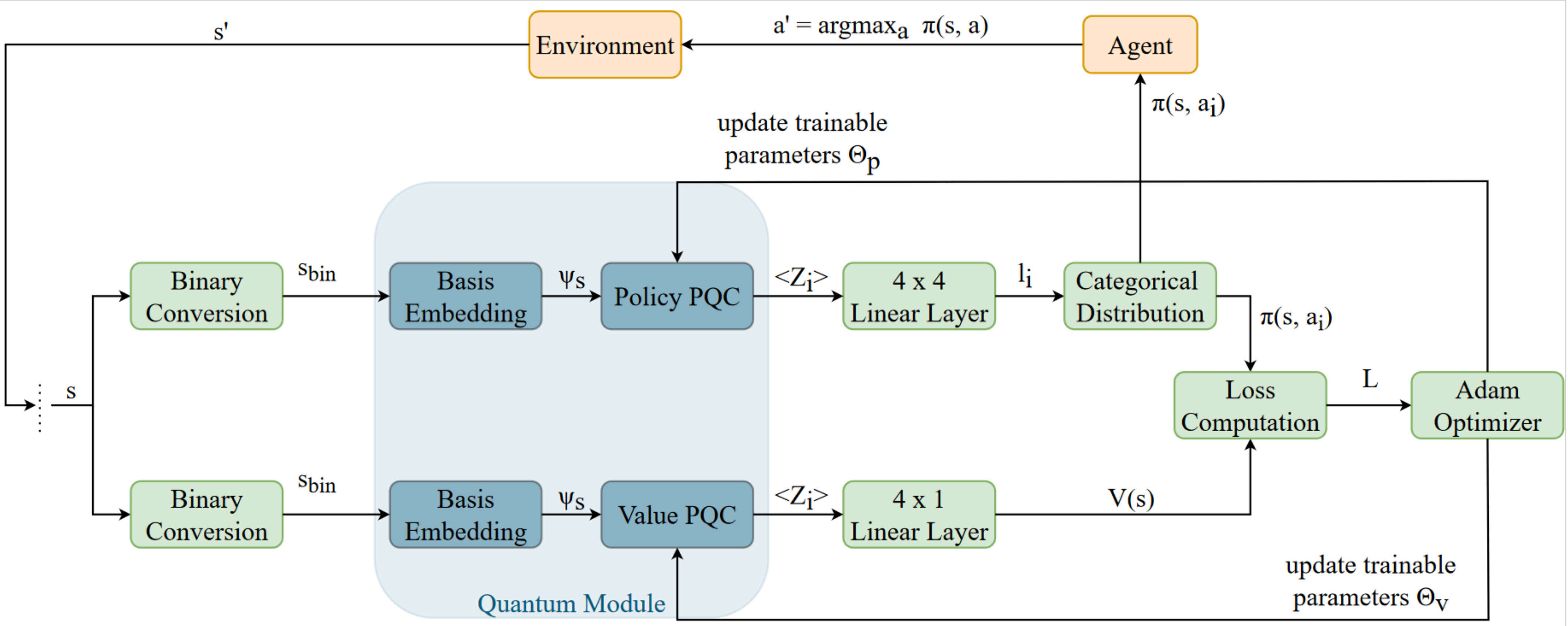
Proof-of-concept example:
- Stochastic frozen lake environment (20% probability to move to non-desired directions).
- Hybrid quantum-classical algorithm with quantum kernels in the agent.
- Quantum variants succeed to find solution faster (less time steps).



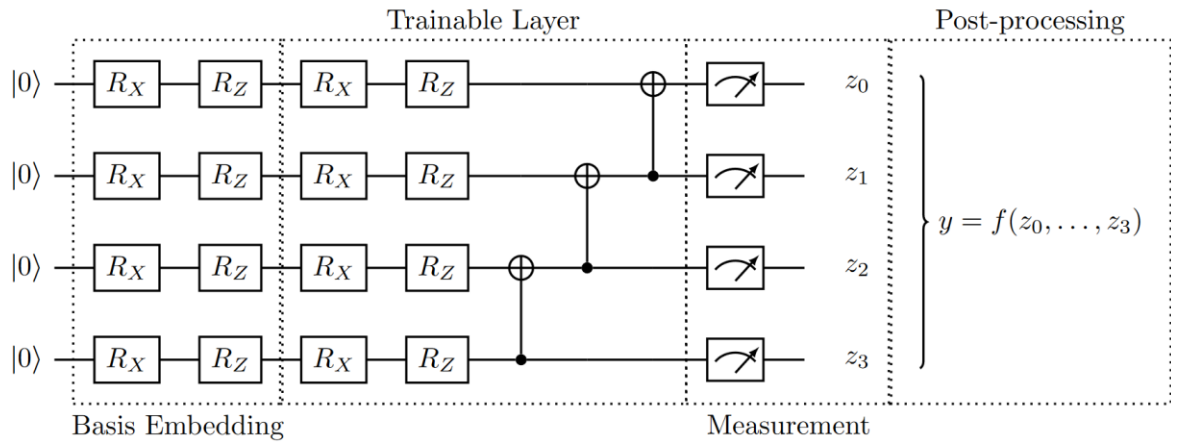


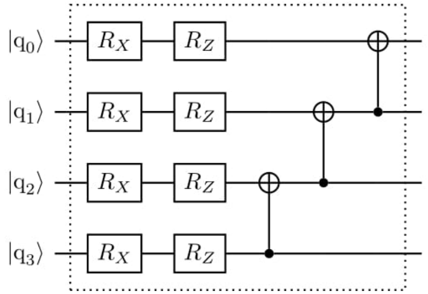
[T.-A. Dragan, M. Monnet, C.B. Mendl, J.M. Lorenz, Quantum Reinforcement Learning for Solving a Stochastic Frozen Lake Environment and the Impact of Quantum Architecture Choices, arXiv:2212.07932].

Architecture

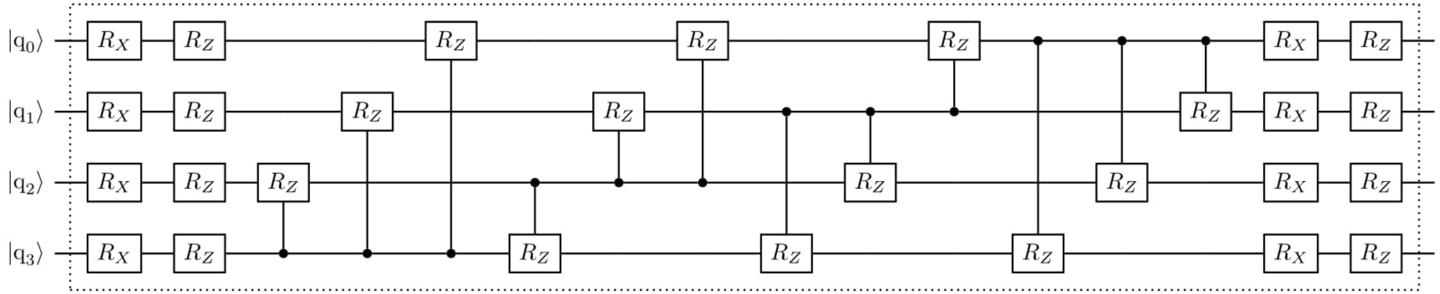


Different architecture choices for the quantum part

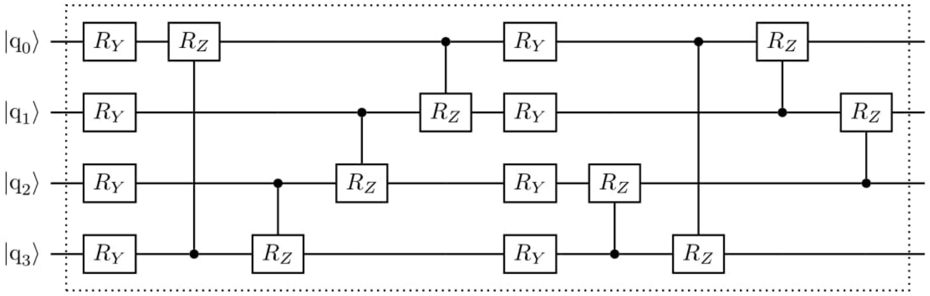




Circuit 2 – basic circuit with basic entanglement



Circuit 5 – programmable universal quantum circuit



Circuit 13 – generic model circuit architecture for classification tasks

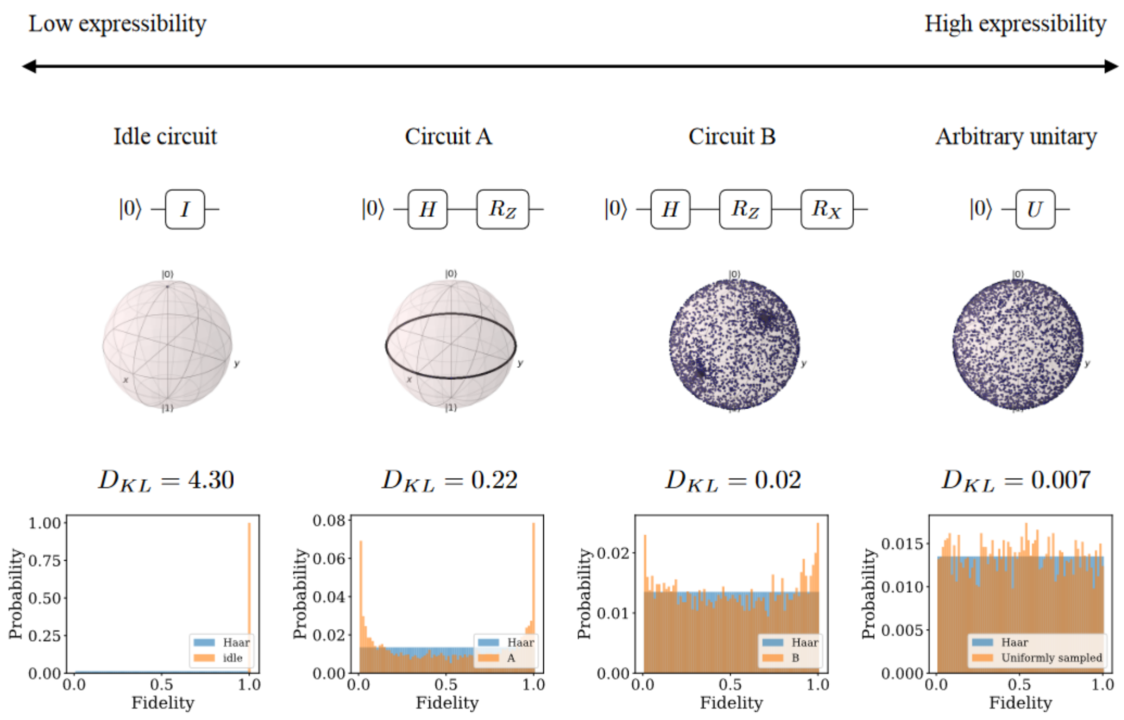
  Public information

Metrics

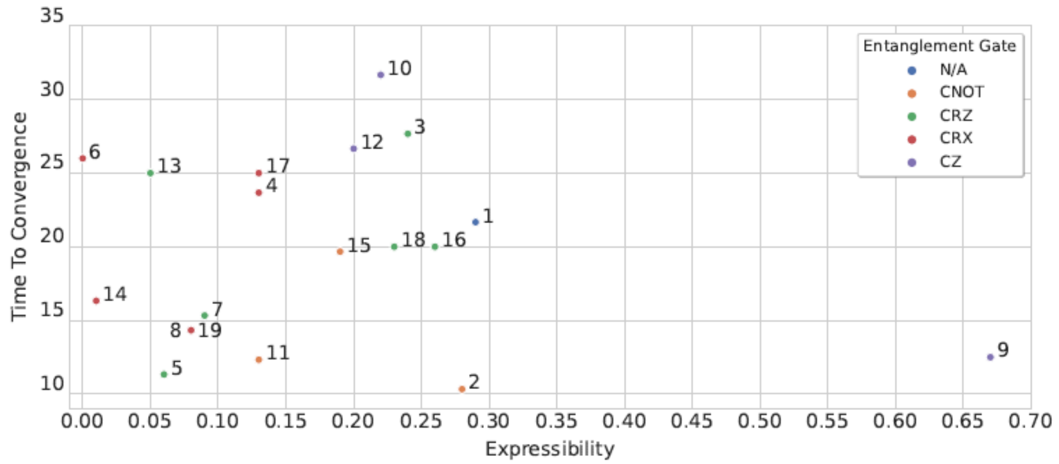
Expressibility

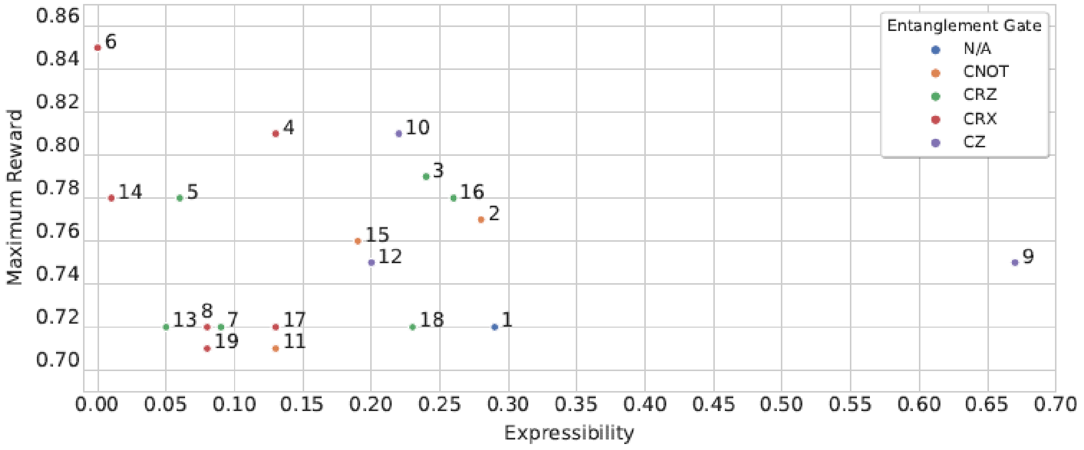
[T.-A. Dragan, M. Monnet, C.B. Mendl, J.M. Lorenz, Quantum Reinforcement Learning for Solving a Stochastic Frozen Lake Environment and the Impact of Quantum Architecture Choices, arXiv:2212.07932].

Expressibility: The ability of the circuit to generate states that represent the Hilbert space well



[S. Sim et al., Expressibility and entangling capability of parameterized quantum circuits for hybrid quantum-classical algorithms, Adv. Quantum Technol. 2 (2019) 1900070]





Conclusion

QC and QML is an emerging technology that is currently developing fast.

Theoretical and academic quantum adavantage has been proven, but the practical quantum advantage (also in HEP) remains to be demonstrated.

On the way to the demonstration many questions to anwer:

- How to construct quantum circuits to obtain benefits?
- How to build the software stack and perform the integration of QC into HPC systems to not lose an advantage again?
- Where will this new technology be useful in practise?
- How to deal with big data?

Contact

PD Dr. habil. Jeanette Miriam Lorenz
Head of Department Quantum-enhanced AI
Tel. +49 89 547088-334
Jeanette.miriam.lorenz@iks.fraunhofer.de

Fraunhofer IKS
Hansastr. 32
80686 München
www.iks.fraunhofer.de

Fraunhofer
IKS

Fraunhofer Institute for Cognitive Systems IKS