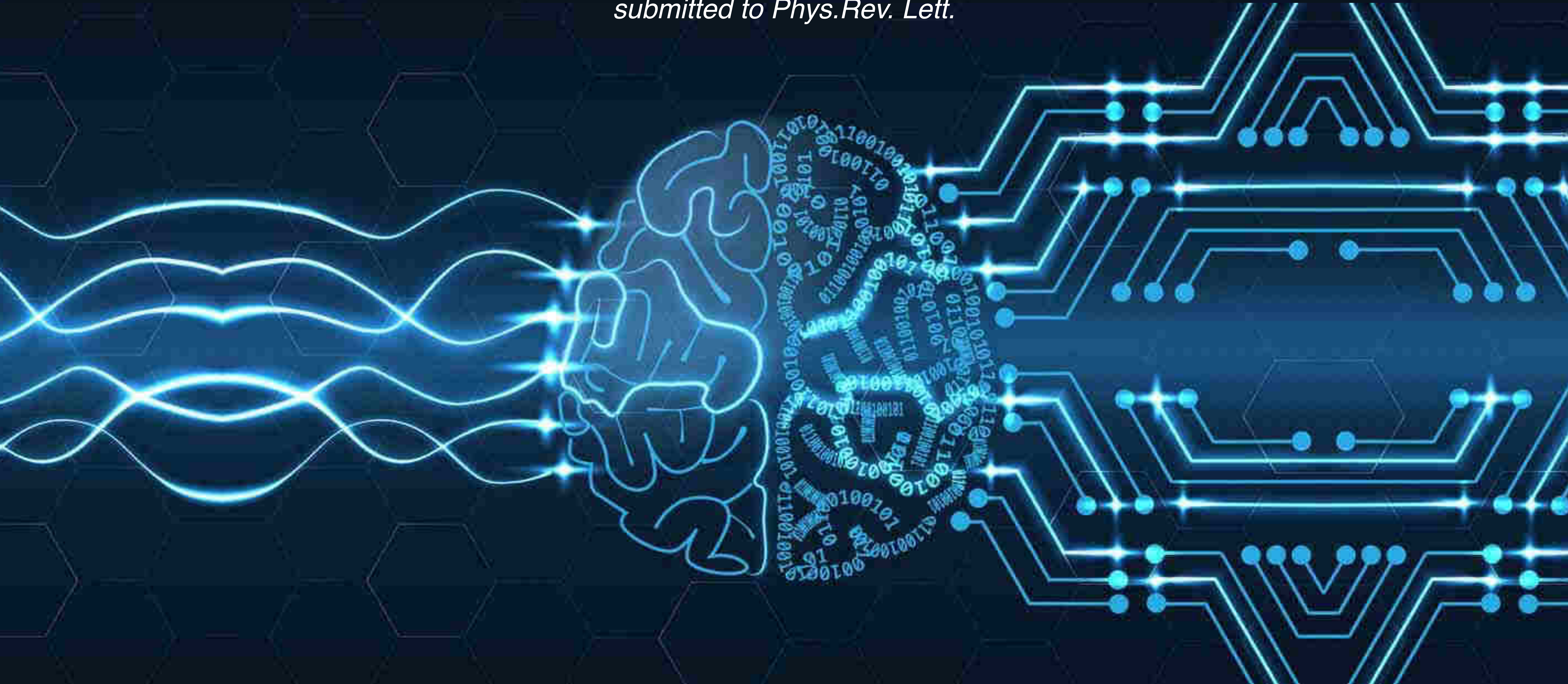
Quantum Machine Learning

arXiv: 1801.00934 (2018)
submitted to Phys.Rev. Lett.



Erik Torrontegui

IFF - CSIC

13/12/2018

Table of contents

Classical vs Quantum

1- bit

 or 

1 0

1 parameter

1- qubit

 and  = 

$|1\rangle$ $|0\rangle$ $|\Psi\rangle = a_1|1\rangle + a_0|0\rangle$

2 parameters

Classical vs Quantum

1- bit

or

1 0

1 parameter

1- qubit

and

$=$

$|1\rangle$ $|0\rangle$ $|\Psi\rangle = a_1|1\rangle + a_0|0\rangle$

2 parameters

Classical vs Quantum

1- bit

 or 

1 0

1 parameter

1- qubit

 and  = 

$|1\rangle$ $|0\rangle$ $|\Psi\rangle = a_1|1\rangle + a_0|0\rangle$

2 parameters

Classical vs Quantum

1- bit

 or 

1 0

1 parameter

1- qubit

 and  = 

$|1\rangle$ $|0\rangle$ $|\Psi\rangle = a_1|1\rangle + a_0|0\rangle$

2 parameters

Classical vs Quantum

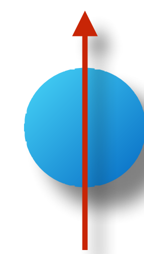
1- bit

or

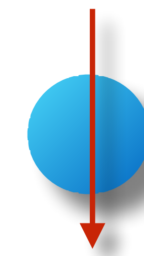
1 0

1 parameter

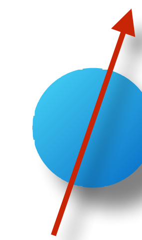
1- qubit



and



$=$



$|1\rangle$ $|0\rangle$ $|\Psi\rangle = a_1|1\rangle + a_0|0\rangle$

2 parameters

Classical vs Quantum

1- bit

 or 

1 0

1 parameter

1- qubit

 and  = 

$|1\rangle$ $|0\rangle$ $|\Psi\rangle = a_1|1\rangle + a_0|0\rangle$

2 parameters

2- bits

 or  or  or 

11 10 01 00

1+1=2 parameters

2- qubits

 and  and  and 

$$|\Psi\rangle = a_3|11\rangle + a_2|10\rangle + a_1|01\rangle + a_0|00\rangle$$

2x2=4 parameters

Classical vs Quantum

1- bit

or

1 0

1 parameter

1- qubit

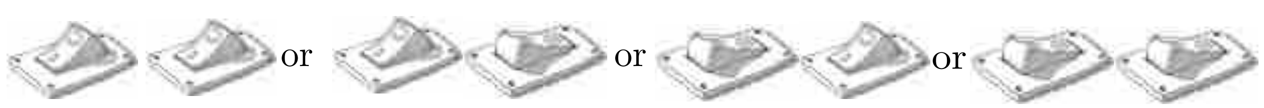
and

=

$|1\rangle$ $|0\rangle$ $|\Psi\rangle = a_1|1\rangle + a_0|0\rangle$

2 parameters

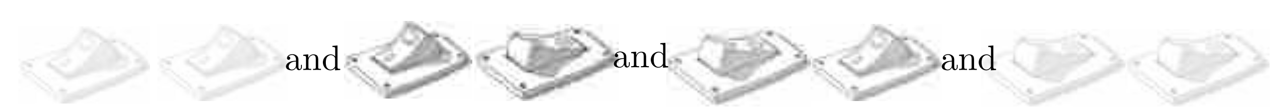
2- bits



11 10 01 00

1+1=2 parameters

2- qubits



$$|\Psi\rangle = a_3|11\rangle + a_2|10\rangle + a_1|01\rangle + a_0|00\rangle$$

$|11\rangle = |\phi_3\rangle$, $|10\rangle = |\phi_2\rangle$, $|01\rangle = |\phi_1\rangle$, $|00\rangle = |\phi_0\rangle$

N- bits

111…11 or 111…10 or 111…01 or … or 000…00

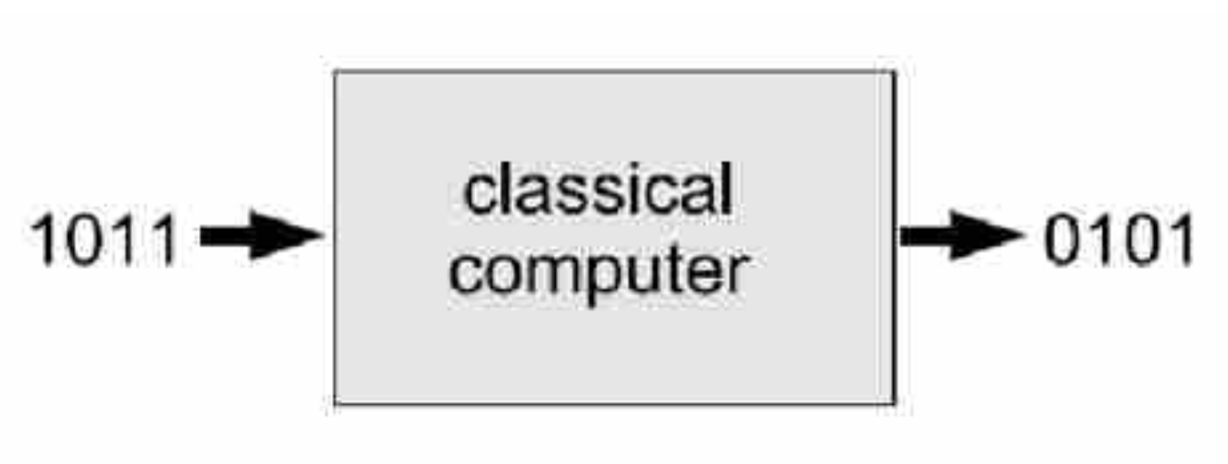
1+1+…+1=N parameters

N- qubits

$$|\Psi\rangle = \sum_{i=1}^{2^N} a_{i-1}|\phi_{i-1}\rangle$$

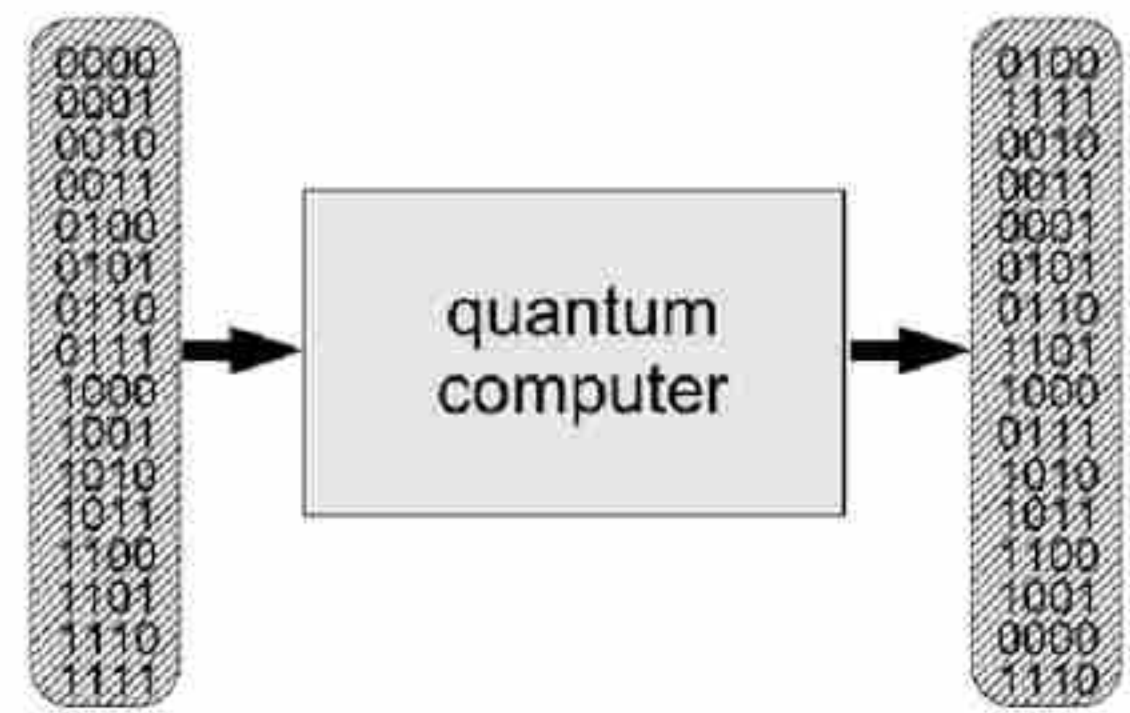
2x2x…x2=2^N parameters

Classical vs Quantum



$|\Psi\rangle$

$|\Psi'\rangle = \hat{U}|\Psi\rangle$



$$\dim(\hat{U}) = (2^N \times 2^N)$$

enormous parallelization

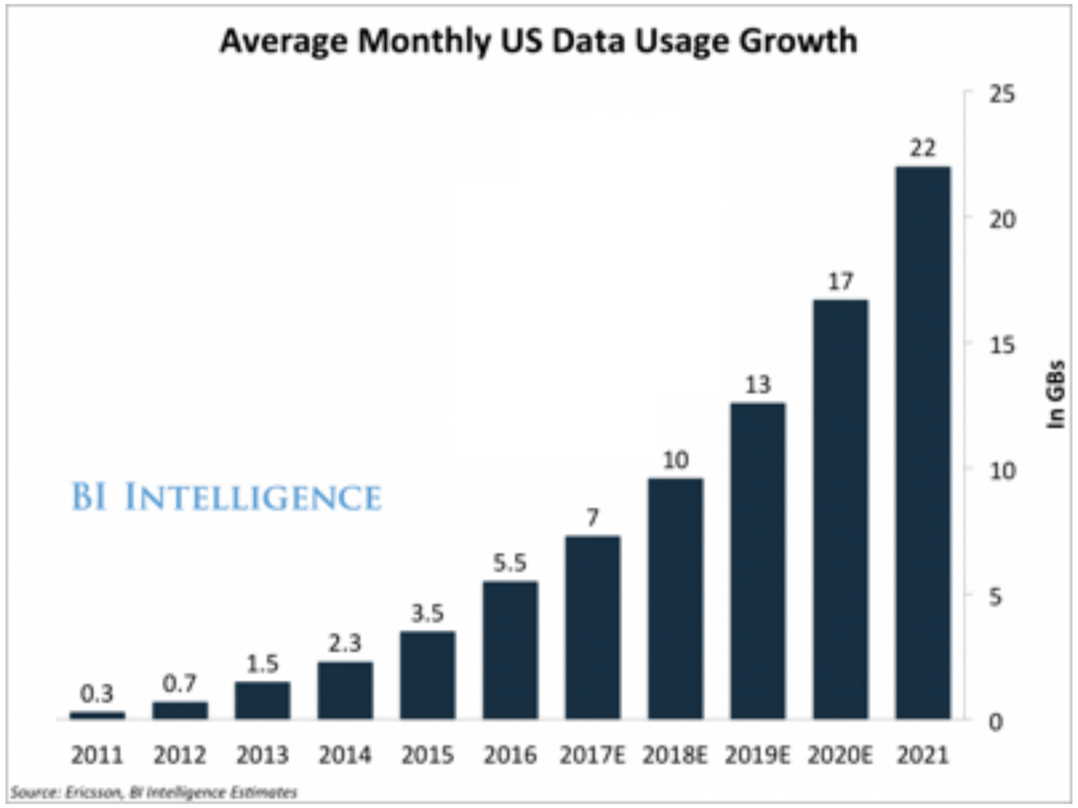
Preliminar

The quantum neuron

Outlook

Motivation

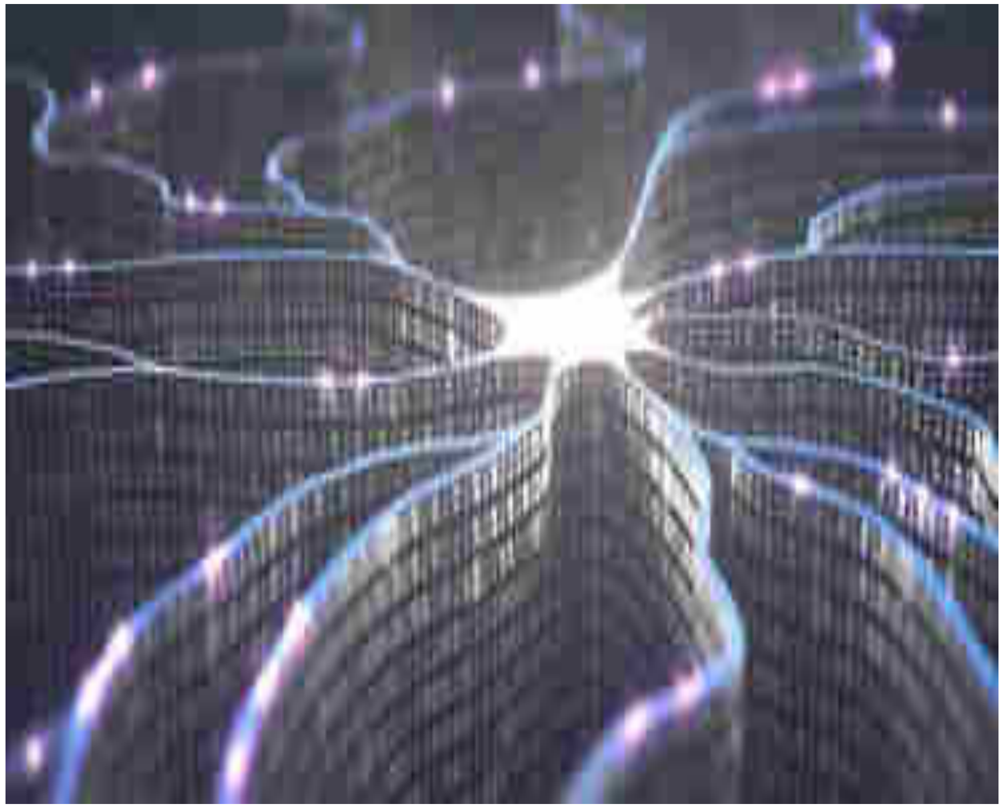
Amount



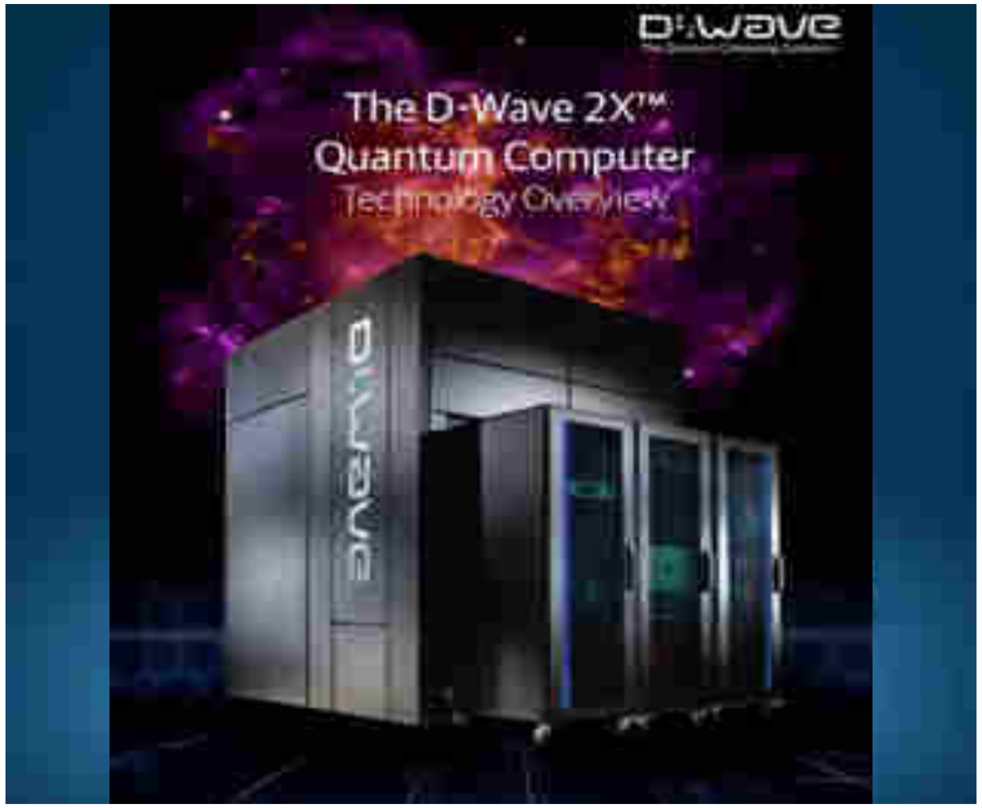
Complexity

Motivation

Machine Learning



Quantum Computing



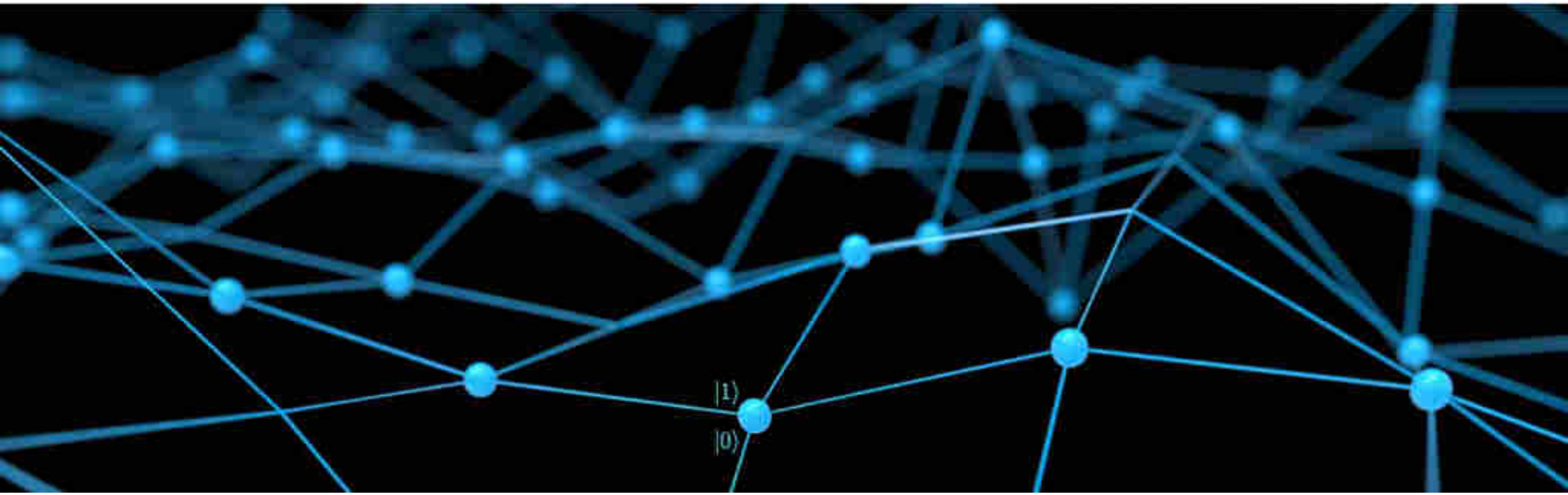
$$|\Psi\rangle = \sum_{i=1}^{2^N} a_i |\phi_i\rangle \qquad |\Psi'\rangle = \hat{U}|\Psi\rangle$$

$$\dim(\hat{U}) = (2^N \times 2^N)$$

enormous parallelization

Motivation

Quantum Machine Learning



Machine Learning

$$q(\mathbf{x}) \simeq h_\theta(\mathbf{x})$$

Machine Learning

Representation

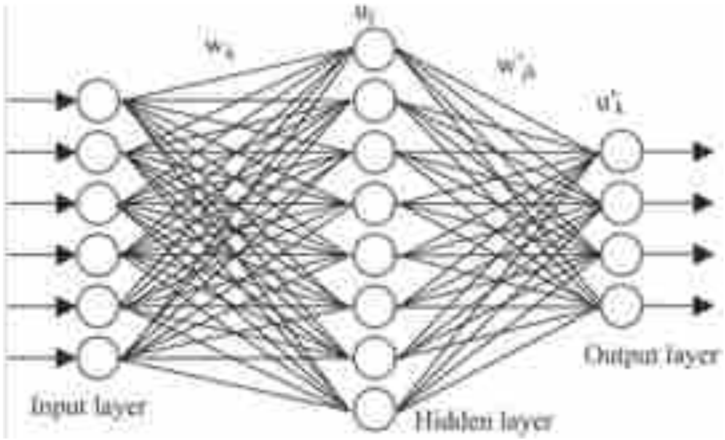
Training set & Cost function

Minimizer

Machine Learning

Representation

neural networks



$\theta \equiv [\omega_{ij}, \omega'_{jk}]$

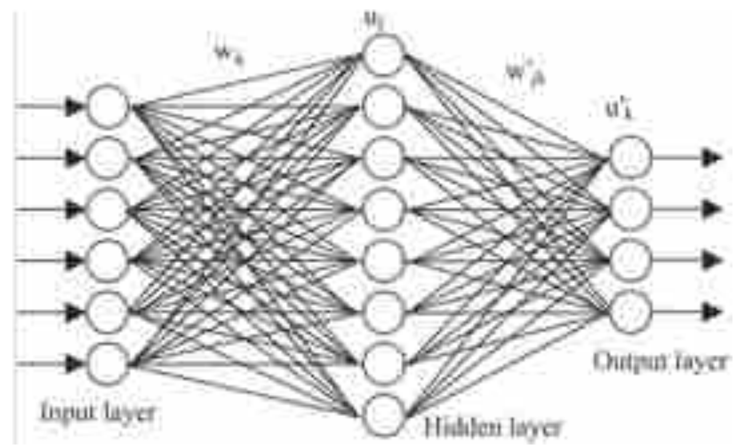
Training set & Cost function

Minimizer

Machine Learning

Representation

neural networks

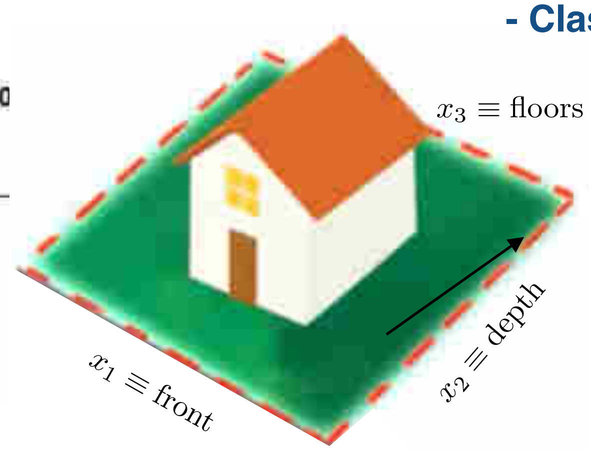


$$\theta \equiv [\omega_{ij}, \omega'_{jk}]$$

Training set & Cost function

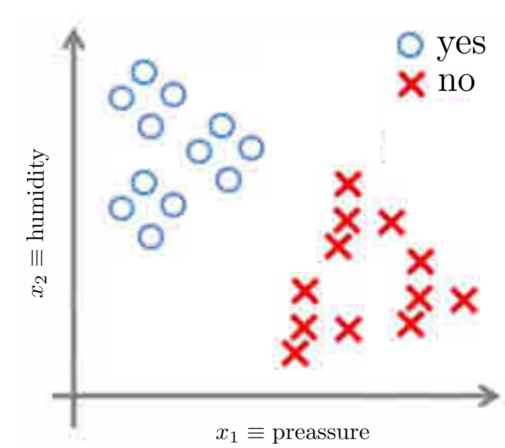
- Regression (y-continuous)

Size (feet²) x_1	Number of bedrooms x_2	Number of floors x_3	Age of home (years) x_4	Price ($100 y
2104	5	1	45	460
1416	3	2	40	232
1534	3	2	30	315
852	2	1	36	178
...	...	...	...	...



- Classification (y-discrete)

- will rain?

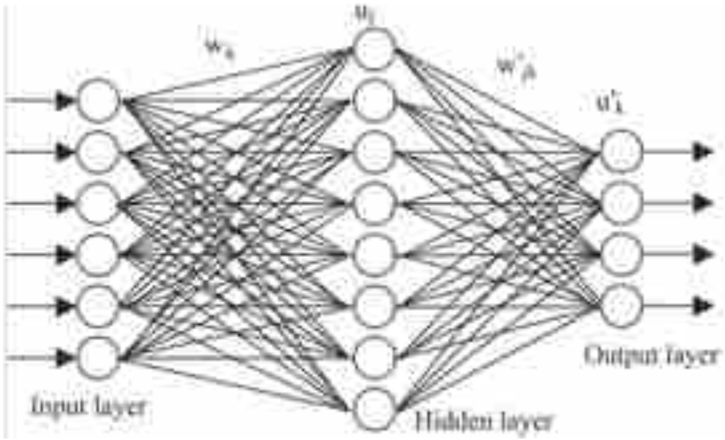


Minimizer

Machine Learning

Representation

neural networks



$\theta \equiv [\omega_{ij}, \omega'_{jk}]$

Training set & Cost function

- Regression (y-continuous)

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

- Classification (y-discrete)

$$J(\theta) = -\frac{1}{m}[\sum_{i=1}^{m} y^{(i)}\log h_\theta(x^{(i)}) + (1-y^{(i)})\log(1-h_\theta(x^{(i)}))]$$

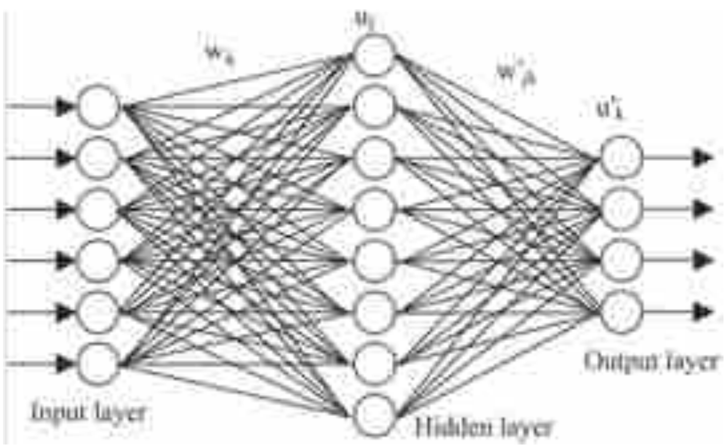
$h_\theta \equiv \text{hypotheis}$

Minimizer

Machine Learning

Representation

neural networks



$$\theta \equiv [\omega_{ij}, \omega'_{jk}]$$

Training set & Cost function

- Regression (y-continuous)

$$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$$

- Classification (y-discrete)

$$J(\theta) = -\frac{1}{m}[\sum_{i=1}^{m} y^{(i)}\log h_\theta(x^{(i)}) + (1 - y^{(i)})\log(1 - h_\theta(x^{(i)}))]$$

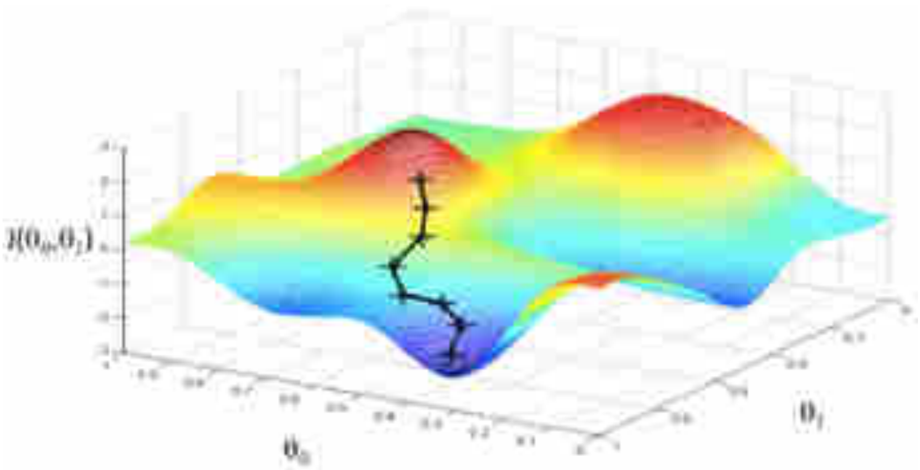
$$h_\theta \equiv \text{hypotheis}$$

Minimizer

- Gradient Descent,

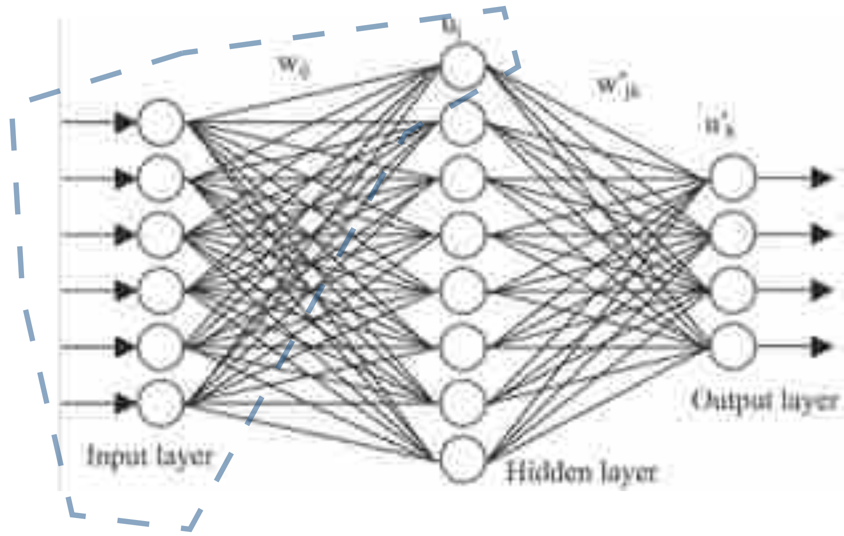
- Krotov,

- Quasi-Newton methods,

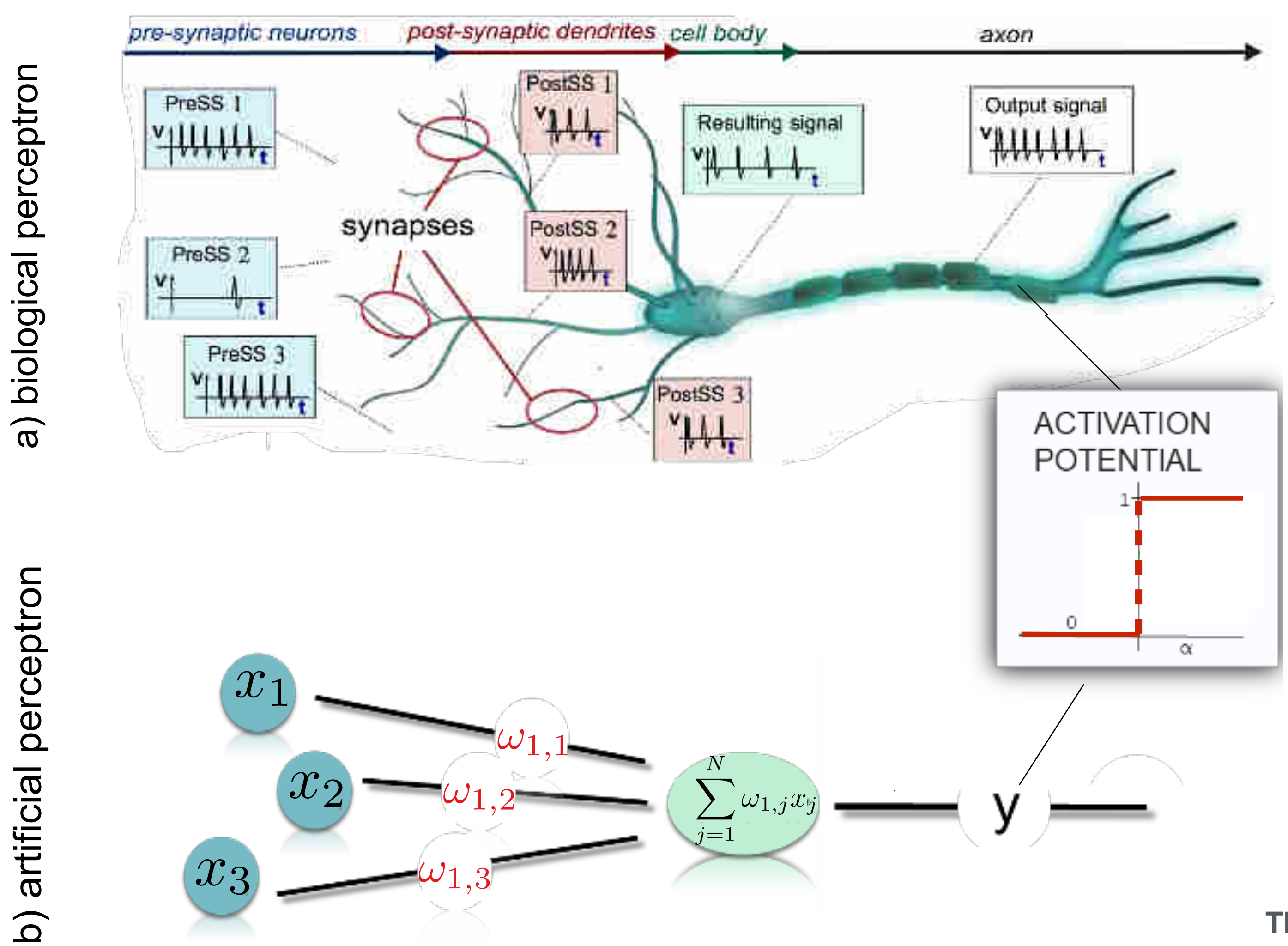


$$\frac{\delta J}{\delta \theta} = 0$$

Artificial Neural Networks



The perceptron

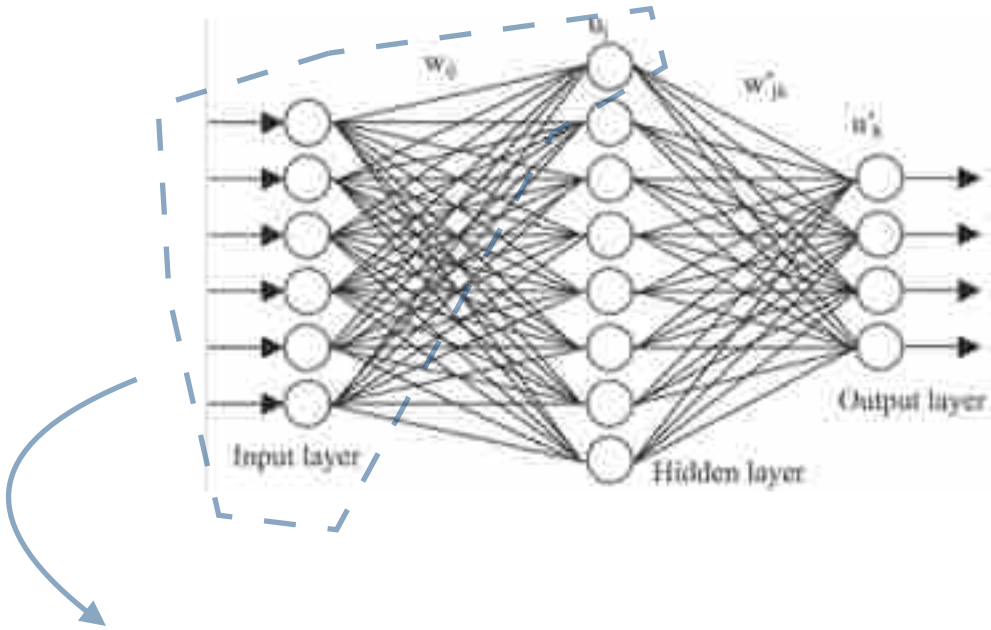


$$x_i = \begin{cases} 1, & \text{if} \quad \sum_{j=1}^{N} w_{ij} x_j \geq \theta_i \\ 0, & \text{else} \end{cases}$$

The perceptron

Physiological Review 65, 386 (1958)

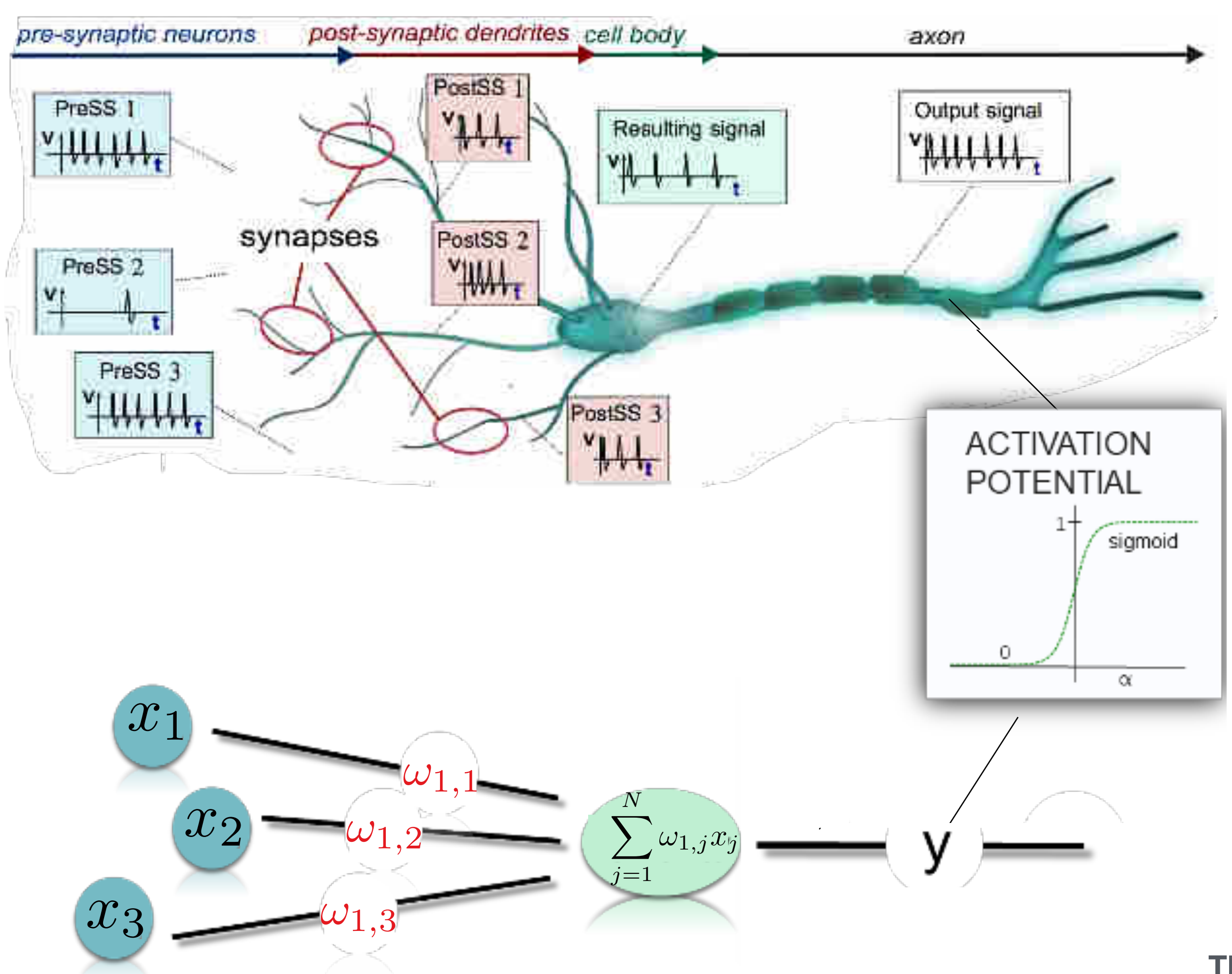
Artificial Neural Networks



The perceptron

a) biological perceptron

b) artificial perceptron



$$h(\mathbf{x}) = h(x_1, \ldots, x_N) = \sum_{j=1}^{M_\epsilon} \beta_j f\left(\sum_{k=1}^{N} \omega_{jk} - \theta_j\right)$$

Universal approximation theorem

Math. of Control, Signal, and Systems 2, 303 (1989)

The perceptron

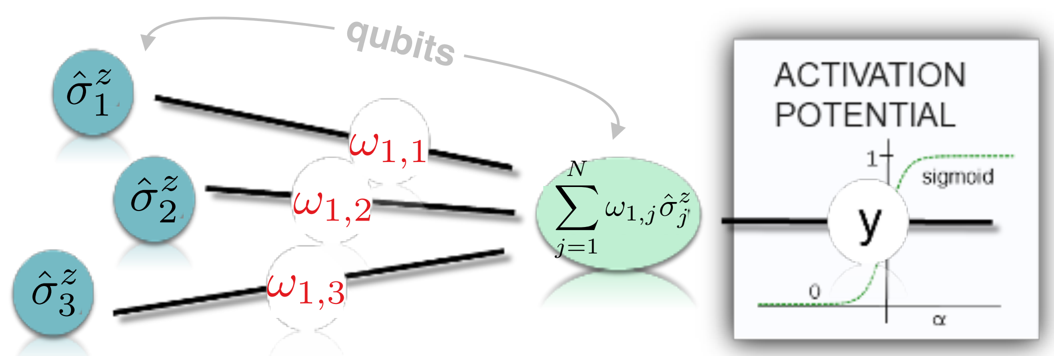
Physiological Review 65, 386 (1958)

Quantum

Quantum Neural Networks

$$\text{rest} \equiv 0 \rightarrow |0\rangle \qquad \text{active} \equiv 1 \rightarrow |1\rangle$$

Discrete inputs



Continuous outputs

Neuron activation corresponds to qubit excitation. The continuous output is the excitation probability

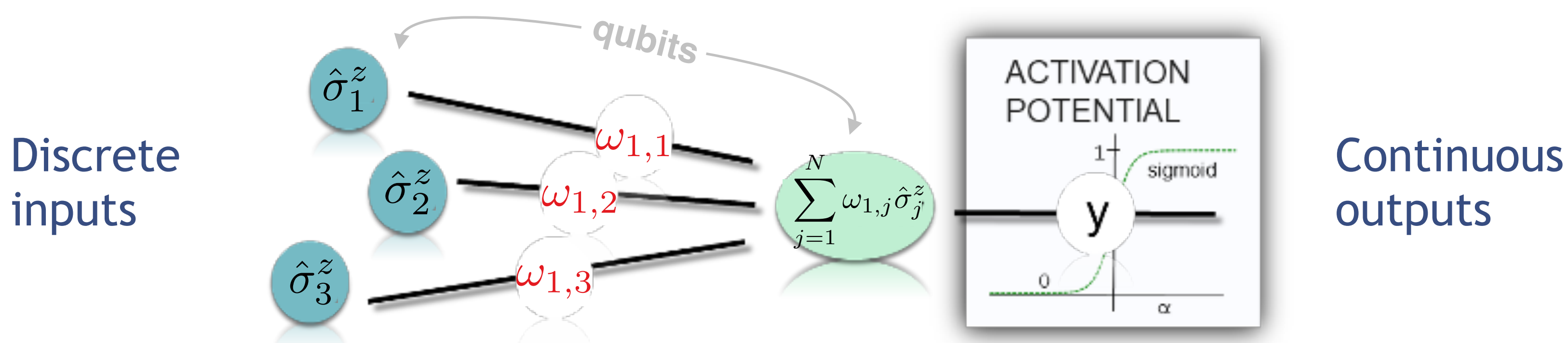
$$|0\rangle \rightarrow \sqrt{1 - f(input)}|0\rangle + \sqrt{f(input)}|1\rangle$$

$$input \equiv \hat{x}_i = \sum_{j=1}^{N} \omega_{ij}\hat{\sigma}_j^z - \theta_i$$

Quantum

Quantum Neural Networks

$$\text{rest} \equiv 0 \rightarrow |0\rangle \qquad \text{active} \equiv 1 \rightarrow |1\rangle$$



Neuron activation corresponds to qubit excitation. The continuous output is the excitation probability

$$|0\rangle \rightarrow \sqrt{1 - f(input)}|0\rangle + \sqrt{f(input)}|1\rangle \qquad input \equiv \hat{x}_i = \sum_{j=1}^{N} \omega_{ij}\hat{\sigma}_j^z - \theta_i$$

Universality

$$\langle Q(\hat{\sigma}_{in,1}^z \ldots \hat{\sigma}_{in,N}^z)\rangle = \sum_j 2\beta_j \langle f\left(\sum_k \omega_{jk}\hat{\sigma}_{in,k}^z - \theta_j\right)\rangle$$

Adiabatic implementation

$$\hat{H}(t) = \frac{\hbar}{2}\left[-\Omega(t)\hat{\sigma}_j^x - \hat{x}_j\hat{\sigma}_j^z\right]$$

Input, constant of motion

Output, conditionally excited

$t = 0 \Rightarrow |\Omega(0)| \gg |x_j|$

$t = t_f \Rightarrow |\Omega(t_f)| \ll |x_j|$

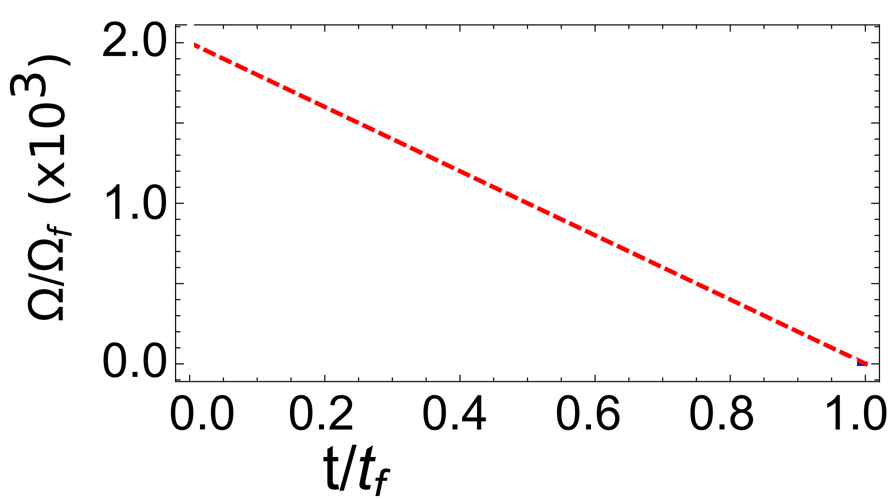
$$\Rightarrow \quad |0_j^x\rangle \xrightarrow{\hat{U}(\hat{x}_j, t_f)} \sqrt{1 - f(\hat{x}_j)}|0_j^z\rangle + \sqrt{f(\hat{x}_j)}|1_j^z\rangle \equiv |\phi_T(\hat{x}_j)\rangle$$

rest — active

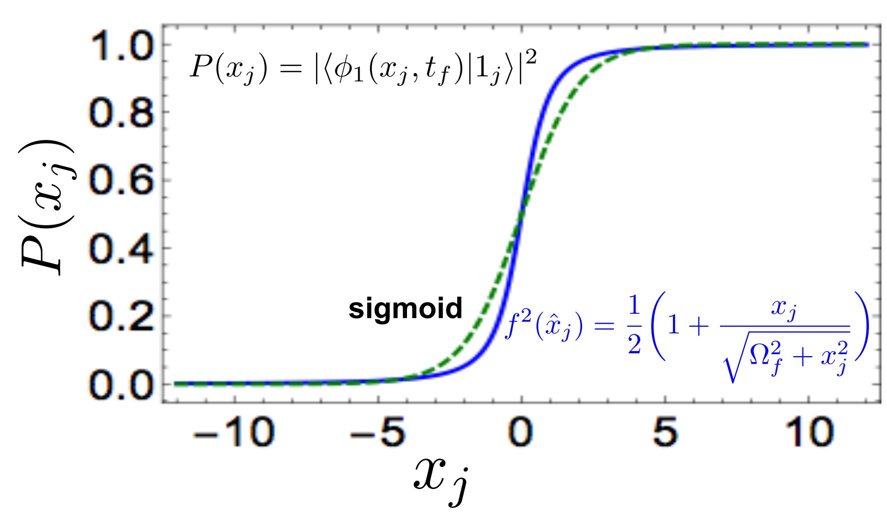
ADIABATIC EVOLUTION

$$\mu(t) = \hbar\left|\frac{\langle\phi_1(t)|\partial_t\phi_2(t)\rangle}{E_1(t) - E_2(t)}\right| \ll 1 \quad \forall t$$

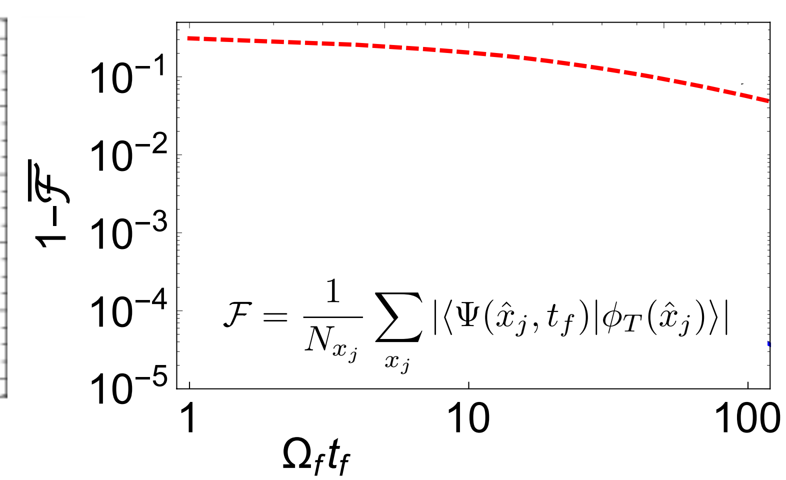
control



excitation probability

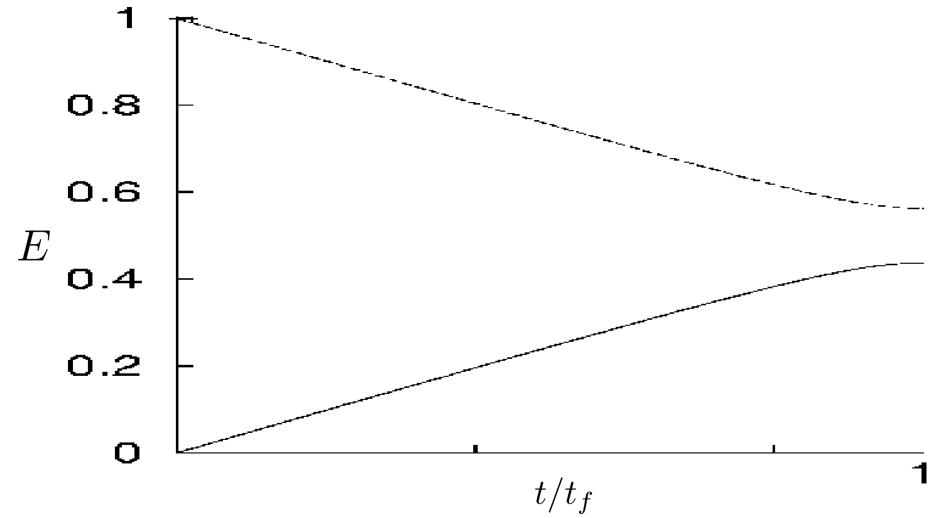


fidelity



FAst QUasi-ADiabatic (FAQUAD)

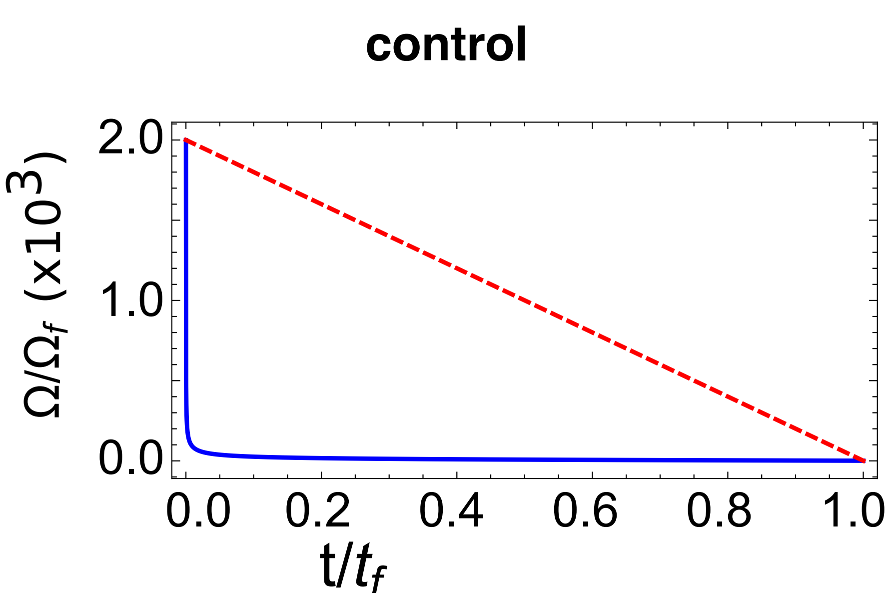
FAst QUasi-ADiabatic (FAQUAD)

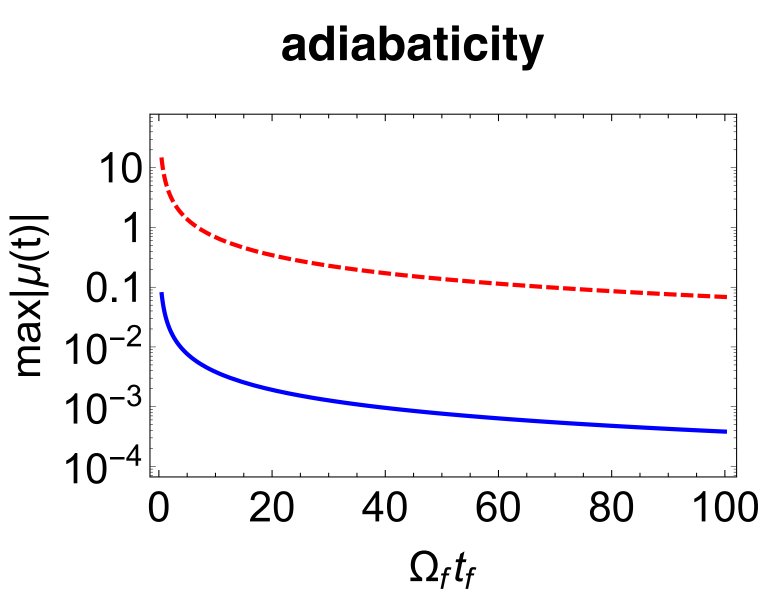


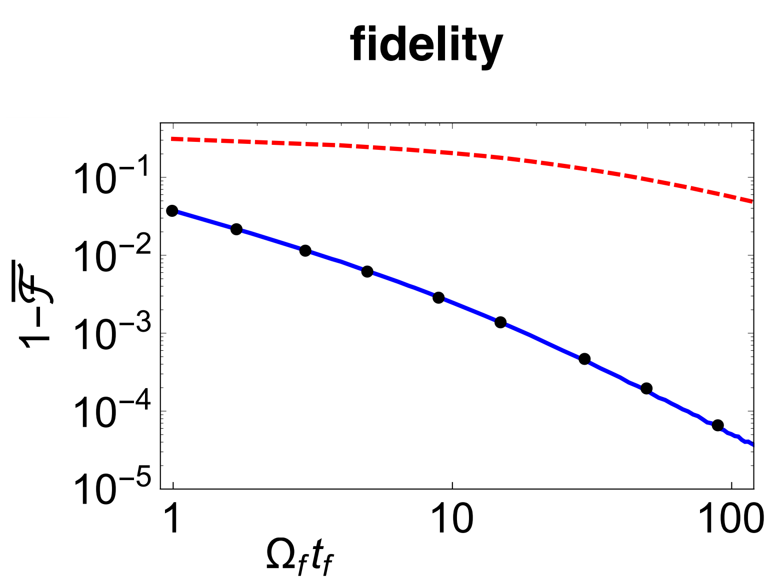
$$\mu = \hbar \left| \frac{\langle \phi_1(t) | \partial_t \phi_2(t) \rangle}{E_1(t) - E_2(t)} \right| = c \neq c(t)$$

delocalization of the transition probability among the adiabatic levels

Phys. Rev. A 92, 043406 (2015)







Training the Network

training set

$\{(X_i, Y_i)\}_{i=1}^{S}$

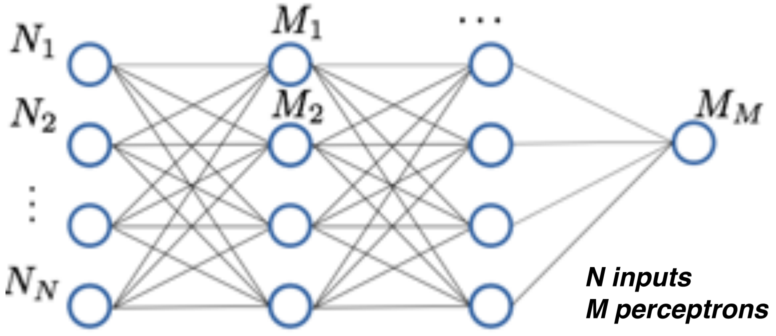
X_i	Y_i
$X_1 = 1$	$Y_1 = 1$ (prime = True)
$X_2 = 2$	$Y_2 = 1$ (prime = True)
$X_3 = 3$	$Y_3 = 1$ (prime = True)
$X_4 = 4$	$Y_4 = 0$ (prime = False)
$\cdots$	$\cdots$

Training the Network

data representation

$$X_i = (x_{i1}, x_{i2}, \ldots, x_{iN}) \in \mathbb{Z}_2^N$$

$$Y_i = Q(X_i) \in [0, 1]$$



Training the Network

QNN action

$$\hat{U}_j = \exp\Bigg[- i\hat{\sigma}^y_{N+j}\chi\Bigg(\sum_{k<N+j} \omega_{j,k}\hat{\sigma}^z_k + \theta_j \Bigg)\Bigg]$$

$$\chi(x) = \arcsin[f(x)^{1/2}]$$

$$\hat{U}_{tot} = \prod_{j=1}^{M} \hat{U}_j$$

Training the Network

feed the QNN

$$|\Psi(X_i)\rangle = |x_{i1}, x_{i2}, \ldots, x_{iN}, 0_{N+1}, \ldots, 0_{N+M}\rangle$$

$$\hat{U}_{tot}|\Psi(X_i)\rangle$$

$$\begin{aligned} p(X_i) &= \frac{1}{2}\left(\langle\Psi(X_i)|\hat{U}^\dagger\hat{\sigma}^z_{out}\hat{U}|\Psi(X_i)\rangle + 1\right) \\ &\simeq Y_i = Q(X_i). \end{aligned}$$

Training the Network

define cost function

$$\mathcal{C}(\omega,\theta) = \frac{1}{S}\sum_{i=1}^{S} H(Y_i, p(X_i)) = \frac{1}{S}\sum_{i=1}^{S} \left[Y_i \log p(X_i) + (1 - Y_i)\log(1 - p(X_i))\right]$$

Training the Network

optimize the QNN

$$\text{training the QNN} = \begin{cases} \frac{\delta \mathcal{C}}{\delta \omega} = 0 \\ \\ \frac{\delta \mathcal{C}}{\delta \theta} = 0 \end{cases}$$

Training the Network

make new predictions

$$X_{S+1} = 9 \rightarrow Y_{S+1} = 0 \ (\text{prime} = \text{False})$$

Preliminar

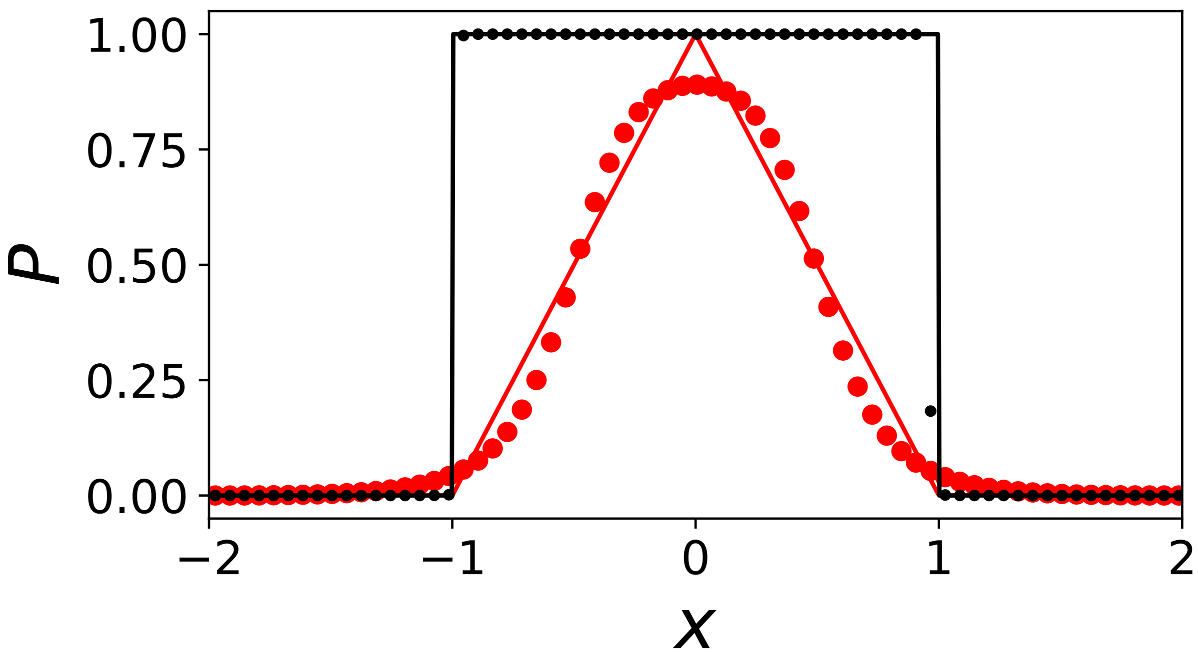
The quantum neuron

Outlook

Outlook & Applications

Multiqubit-gates & quantum sensors

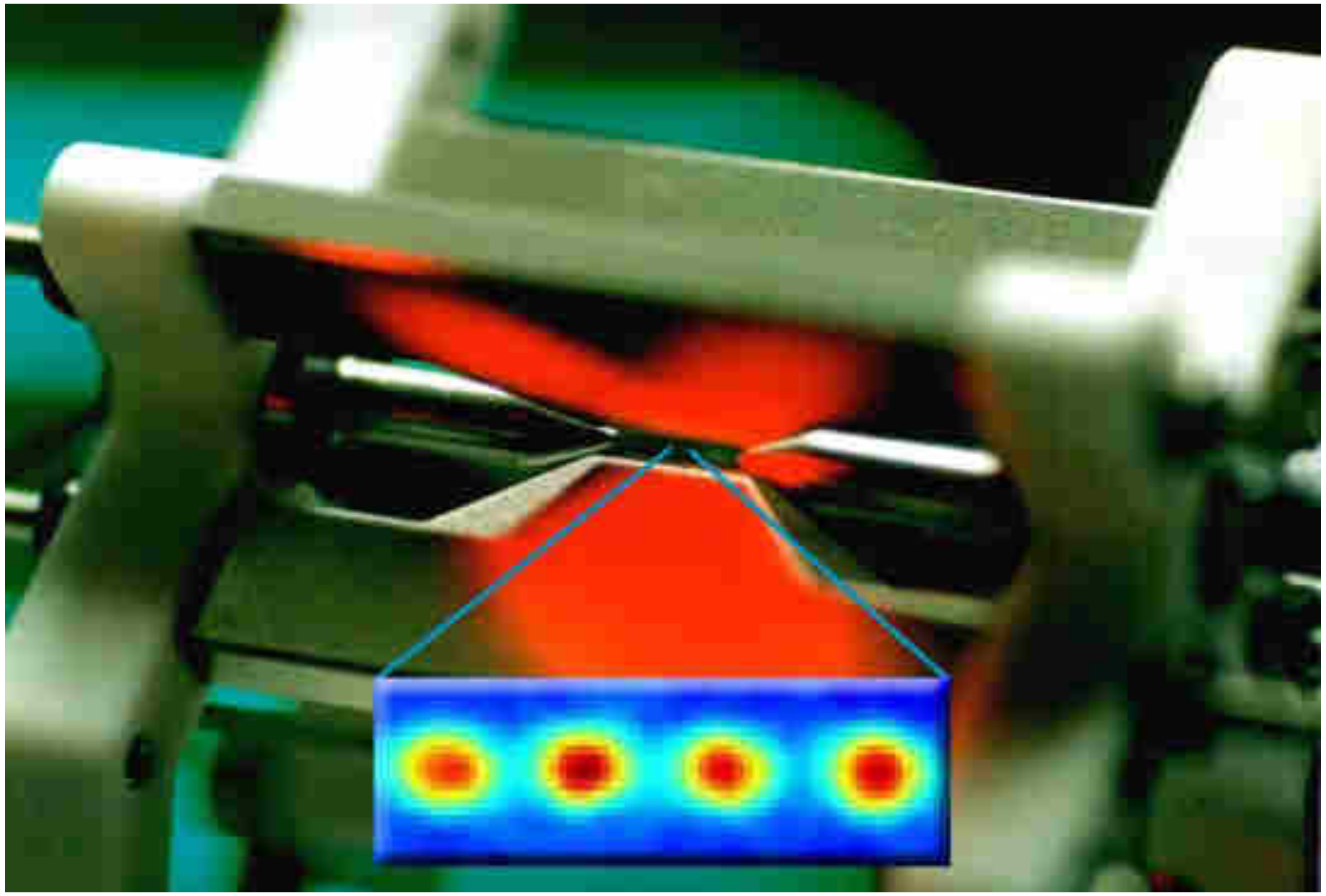
$$\begin{aligned} \hat{W}_{mqb} &= \exp[i\hat{Q}(\hat{\sigma}_1^z, \ldots, \hat{\sigma}_{j-1}^z)\hat{\sigma}_j^y] \\ &\simeq \prod_n \hat{U}_j(\sum_{k<j} \omega_{jk}^{(n)} \hat{\sigma}_k^z - \theta_j^{(n)}; f) \end{aligned}$$



$$\text{XOR for } M_1 < \sum_{i=1}^{N} s_i < M_2$$

Outlook & Applications

Experiment



Prof. C. Wunderlich
U. Siegen

trapped ions setup

$$\hat{H}(t) = \sum_{i=1}^{N} \Omega(t)\hat{\sigma}_i^x + \sum_{j<i=1}^{N} J_{i,j}\hat{\sigma}_i^z\hat{\sigma}_j^z$$

Take home

The quantum perceptron is an universal approximator. It has at least the same approximation power as classical neural networks.

We provide a straightforward physical implementation using an Ising Hamiltonian: trapped ions, cold atoms, superconducting circuits, …

The quantum perceptron constitutes the building block of new quantum technologies: multiqubit-gates, quantum sensing, …

Take home

The quantum perceptron is an universal approximator. It has at least the same approximation power as classical neural networks.

We provide a straightforward physical implementation using an Ising Hamiltonian: trapped ions, cold atoms, superconducting circuits, …

The quantum perceptron constitutes the building block of new quantum technologies: multiqubit-gates, quantum sensing, …

Take home

The quantum perceptron is an universal approximator. It has at least the same approximation power as classical neural networks.

We provide a straightforward physical implementation using an Ising Hamiltonian: trapped ions, cold atoms, superconducting circuits, …

The quantum perceptron constitutes the building block of new quantum technologies: multiqubit-gates, quantum sensing, …

Welcome to QUINFOG


"That's all Folks!"