Quantum Machine Learning

Dario Gerace

Dipartimento di Fisica, Università di Pavia (IT)

workshop on «Quantum Computers» – Parma, 28/10/2019
Classical Machine Learning

- Set of mathematical tools and computational algorithms allowing to recognize patterns in data, and generate new patterns from trained output

  i.e., giving suitable meaning to previously unknown data

- Large amount of classical data are elaborated, inevitably reaching the limits of computational complexity and storage capacity on currently available data centers

- Machine learning is part of Artificial Intelligence
Classes of learning algorithms

E.g., support vector machines

E.g., deep neural networks

supervised learning

unsupervised learning

reinforcement learning
For example: data classification

Image recognition through supervised learning of a DNN
Quantum Machine Learning

- Machine Learning algorithms are able to recognize the patterns they produce

- Quantum Mechanics is known to produce atypical data patterns

- Maybe small QC can recognize and produce patterns that are classically untractable

*Quantum machine learning* is the quest for quantum algorithms allowing to solve machine learning tasks on a QC

...more efficiently? Hard to say

Support vector machines (SVM) and kernel methods

- Kernel as inner product of two vectors in a (mapped) feature space

\[ k(x, x') = \langle \phi(x), \phi(x') \rangle_F \]

- Linear kernel SVM: finding the best separating hyperplane through definition of ‘support vectors’

- Nonlinear kernel SVM: Maps input data into higher dimensional feature space to find best separating hyperplane
Quantum SVM

Limitations due to large computational space might be solved by QC


The IBM result

Havlicek et al.,
*Supervised learning with quantum-enhanced feature spaces*,
Artificial neural networks (ANN)

- Alternative paradigm to ML
- 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

\[ \sum_{j} i_{j} w_{j} \] \[ \forall i_{j}, w_{j} \in \mathbb{R} \]

Rosenblatt, Psychol. Rev. 65, 386 (1958)
McCulloch-Pitts artificial neurons

- Simplified model with binary valued input/output entries
  
  \[ i_j, w_j \in \{0,1\} \text{ or } i_j, w_j \in \{-1, +1\} \]

- Introduced to model the activity of real biological neurons

Bulletin Math. Biophys. 5, 115-133 (1943)
The classical perceptron is the simplest linear classifier. It requires extension to a multilayer structure to be able to perform nonlinear tasks.
McCulloch-Pitts neurons on a digital quantum computer

The key function

\[ \vec{i} \cdot \vec{w} = \sum_j i_j w_j \]

Encoding input and weights

\[ \vec{i} = \begin{pmatrix} i_0 \\ i_1 \\ \vdots \\ i_{2^N - 1} \end{pmatrix} \]

\[ \vec{w} = \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_{2^N - 1} \end{pmatrix} \]

McCulloch-Pitts neuron model

\[ i_j, w_j = -1, +1 \]

\[ |\psi_i\rangle = \frac{1}{\sqrt{2^N}} \sum_{j=0}^{2^N-1} i_j |j\rangle \]

\[ |\psi_w\rangle = \frac{1}{\sqrt{2^N}} \sum_{j=0}^{2^N-1} w_j |j\rangle \]

\[ |j\rangle \in \{ |000\cdots00\rangle, |000\cdots01\rangle, \ldots |111\cdots11\rangle \} \]

Tacchino et al., npj Quant. Info 5, 26 (2019)
McCulloch-Pitts neurons on a digital quantum computer

The quantum algorithm: a circuit model

Constraints on the unitaries:

\[ |\psi_i\rangle = U_i |0\rangle^{\otimes N} \]
\[ |1\rangle^{\otimes N} = U_w |\psi_w\rangle \]

\[ |0\rangle^{\otimes N} |0\rangle_a \rightarrow \sum_{j=0}^{2^N-2} c_j |j\rangle |0\rangle_a + c_{2^N-1} |2^N - 1\rangle |1\rangle_a \]

with \( c_{2^N-1} = \langle \psi_i |\psi_w \rangle \)

Tacchino et al., npj Quant. Info 5, 26 (2019)
Elementary pattern recognition

\[ N = 2 \]

\[
\begin{pmatrix}
i_0 \\
i_1 \\
i_2 \\
i_3
\end{pmatrix}
\]

\[
\begin{pmatrix}
w_0 \\
w_1 \\
w_2 \\
w_3
\end{pmatrix}
\]

\[
\begin{array}{cc}
i_0 & i_1 \\
i_2 & i_3
\end{array}
\]

\[ +1 = \text{white} \]

\[ -1 = \text{black} \]

\[ \langle \psi_i | \psi_w \rangle^2 = 1 \]

It’s me!

\[ \langle \psi_i | \psi_w \rangle^2 = 1 \]

It’s still me! (in negative colors)

\[ \langle \psi_i | \psi_w \rangle^2 = 0 \]

It’s not me!
Running the algorithm on NISQ-hardware

Exact result (N = 2)

Experiment (N = 2 + 1 ancilla)

\[
\sum_{j=1}^{w} w_{ij}^2 = 0.84
\]

\[
\sum_{j=1}^{w} w_{ij}^2 = 0.07
\]

IBM-Q Experience 5-qubit ‘Tenerife’ processor

Tacchino et al., npj Quant. Info 5, 26 (2019)
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)
Quantum ANN
Example (to be run on NISQ hardware)

\[
\sum_{j} i_j w_j \]

\[
\sum_{j} i_j v_j \]

\[
o_1 x_1 + o_2 x_2 \]

neuron 1

neuron 2

neuron 3

output
Open questions and challenges

- How does it scale?
- How efficient is it?
- Quantum training?
- Test on real hardware based on different technologies
- Test with larger input data (use cases?)
People

F. Tacchino  D. Bajoni  C. Macchiavello

P. Barkoutsos  I. Tavernelli