

# Quantum Machine Learning

## Opportunities and Challenges

Rathour Param Jitendrakumar  
190070049

*Department of Electrical Engineering  
Indian Institute of Technology Bombay  
paramrathour@ee.iitb.ac.in*

**Abstract**—Nowadays, a general purpose quantum computer seems closer to reality than ever before, yet, it is so far. Can the near-term quantum technologies create an impact on ML and guide us towards quantum supremacy? This report contains an introduction to machine learning and quantum mechanics. We also discuss a quantum algorithm to solve linear system of equations and then move onto discussing Quantum-Assisted Machine Learning, its the opportunities and challenges in the near-term future.

**Index Terms**—quantum computing, machine learning, quantum machine learning, quantum-assisted machine learning, near-term quantum computers, quantum-assisted Helmholtz machines

### I. INTRODUCTION

Quantum Computing truly has the potential but, when will it live up to the (rising) expectations? There has been a steady progress in this field, from increasing qubits to the increase in algorithms and design techniques. Popular quantum computing algorithms, e.g, Shor's algorithm, Grover's algorithm, Deutsch-Jozsa, BB84 Protocol, Quantum Fourier Transform and others have all been instrumental achievements in their fields. But in Machine Learning there is no such defining algorithm yet. Is their a feasible Quantum Approach for Machine Learning (ML) in near-term future? The estimated number of qubits in near-term future is 100-1000 which unfortunately is not sufficient for large ML tasks. We will focus on quantum approaches on a very promising field of unsupervised ML after getting comfortable with all the basics.

### II. MACHINE LEARNING

Machine Learning (ML) is the practice of estimating models that make predictions on 'new' data using available data. This is especially helpful when the actual model is very complex and no known algorithms can discover such model or they take large amount of time or other resources.

#### A. Machine Learning Paradigms

1) *Supervised ML*: In supervised ML, we 'supervise' the model by training it on a labelled data set. The data set contains generally large number of 'testcases' where both input value and the correct output value is specified. The applications of this approach includes

a) *Classification Problems*: In such problems, the model classifies the input into some property. This classification is either qualitative (Categorical) or quantitative (Numerical) .

Examples of qualitative property – gender, species, bug severity.

Examples of quantitative property – age, height, count of objects.

b) *Regression Problems*: In these problems, the model estimates a relation between the output variables using the given input variables (generally independent)

2) *Unsupervised ML*: In unsupervised ML, there is no labelled data, the model takes only input variables. The model will then find a relation between input variables. The biggest benefit of unsupervised versus supervised ML is its ability to gain knowledge without needing expensive labelled data. Its applications includes

a) *Clustering*: Clustering is used to group given data by finding a pattern between them

b) *Association*: Association is used to discover relations between input variables of some data set.

3) *Semi-Supervised ML*: Here, some part of data is labelled whereas the other part is not. So, using unsupervised techniques a model can label unlabelled data and feed it into a supervised model.

4) *Reinforcement Learning*: Reinforcement Learning (RL) algorithms are interested in learning the behaviour of intelligent agents in an environment which maximises their total reward. The focus is on the explore-exploit trade-off, whether to explore other options or exploit the current option. RL has huge potential applications some of which include marketing and advertising, game solving, self driving cars.

#### B. Issues with ML

- Lack of good quality data sets.
- Possibility of errors e.g., underfitting or overfitting possible.
- Result are sensitive to small perturbations.
- Safety and Privacy concerns exists.
- Lack of explainability of model.
- Slow process. Training large data sets eats up time.

Quantum Computing (QC) can improve a lot of these issues.

### III. QUANTUM MECHANICS FOR QUANTUM COMPUTATION

A quantum bit abbreviated as *qubit*, is the fundamental data unit of a quantum computer. The notion of qubit is analogous to bit from the classical computers. A bit has two states 0 or 1 whereas a qubit  $|\psi\rangle$  coexists in the states  $|0\rangle$  and  $|1\rangle$  i.e. it  $|\psi\rangle$  is a linear combination  $|0\rangle$  and  $|1\rangle$ ,

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle. \quad (1)$$

$|0\rangle$  and  $|1\rangle$  are unit vectors and form an orthonormal basis. They are standard-basis vectors also denoted by

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad \text{and} \quad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}.$$

If  $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$  and  $|\varphi\rangle = \gamma|0\rangle + \delta|1\rangle$  and then

*Definition 3.1 (Inner Product):*

$$\langle\psi|\varphi\rangle = \begin{bmatrix} \alpha^* & \beta^* \end{bmatrix} \begin{bmatrix} \gamma \\ \delta \end{bmatrix} = \alpha^*\gamma + \beta^*\delta$$

where  $*$  is the complex conjugate operation.

*Definition 3.2 (Outer Product):*

$$|\psi\rangle\langle\varphi| = \begin{bmatrix} \alpha \\ \beta \end{bmatrix} \begin{bmatrix} \gamma^* & \delta^* \end{bmatrix} = \begin{bmatrix} \alpha\gamma^* & \alpha\delta^* \\ \beta\gamma^* & \beta\delta^* \end{bmatrix}$$

*Definition 3.3 (Norm of a vector):*

$$\|\langle\psi|\| = \sqrt{\langle\psi|\psi\rangle}$$

#### A. Postulates of Quantum Mechanics

*Definition 3.4 (Hilbert Space):* A vector space with distance function and inner product among its elements.

*Definition 3.5 (State Space):* The Hilbert space corresponding to an isolated physical system.

*Definition 3.6 (State Vector):* The state of the system.

a) **State Space:** The state vector can be completely represented by a unit vector in the state space.

*Definition 3.7 (Hermitian adjoint):*  $H$  and  $H^*$  are Hermitian conjugates of each other when

$$\langle\Phi|H|\Psi\rangle = \langle H^*\Phi|\Psi\rangle.$$

*Definition 3.8 (Hermitian operator):* A Hermitian operator  $H$  is its own Hermitian conjugate, i.e  $H = H^*$ .

*Definition 3.9 (Observable):* A Hermitian operator  $\mathcal{O}$  which describes the projective measurement. It acts upon the state space of the system to be measured and is represented as

$$\mathcal{O} = \sum_{\lambda} \lambda P_{\lambda}$$

and  $P_{\lambda}$  is the *projector* with eigenvalue  $\lambda \in$  eigenspace of  $\mathcal{O}$ .

*Definition 3.10 (Unitary Transformation):* A transformation that preserves inner product. So, a unit vector remains a unit vector after the transformation. It is denoted by  $U$  and it satisfies

$$U^*U = UU^* = I.$$

b) **Measurement:** A measurement on  $\mathcal{O}$  will result in the eigenvalues  $\lambda$  with the probability of collapsing to  $\lambda$  is

$$p_{\lambda} = \|P_{\lambda}|\psi\rangle\|^2 = \langle\psi|P_{\lambda}|\psi\rangle.$$

resulting in the irreversibly collapse of the state of system to

$$\frac{1}{\sqrt{p_{\lambda}}P_{\lambda}}|\psi\rangle.$$

c) **Evolution I:** The system evolution is governed by the Schrödinger's equation

$$H|\psi(t)\rangle = i\hbar \frac{d}{dt}|\psi(t)\rangle.$$

d) **Evolution II:** If  $|\psi_0\rangle, |\psi_1\rangle$  describes the state of system at times  $t_0, t_1$  respectively then the evolution of system depends only on  $t_0$  and  $t_1$ . It is given by

$$|\psi_1\rangle = U|\psi_0\rangle.$$

So far, we have studied one-qubit systems. For multi-qubit systems we introduce the following notation.

*Definition 3.11 (Tensor Product (aka Kronecker Product)):* For a given  $m, n$  dimensional vector spaces  $V, W$ . The tensor product of  $V$  with  $W$  (denoted by  $V \otimes W$ ) is an  $mn$  dimensional vector spaces with elements are linear combinations of tensor products  $|v\rangle \otimes |w\rangle$ .

We can also define the tensor product of linear operators  $A, B$  over vector spaces  $V, W$  as

$$(A \otimes B)(|v\rangle \otimes |w\rangle) = A|v\rangle \otimes B|w\rangle.$$

e) **Composition:** For  $n$  isolated systems with states  $|\psi_0\rangle, |\psi_1\rangle, \dots, |\psi_{n-1}\rangle$ , the state of the composite system is given by  $|\psi_0\rangle \otimes |\psi_1\rangle \otimes \dots \otimes |\psi_{n-1}\rangle$

Now, let's look at the power of quantum algorithms with an example.

### IV. QUANTUM ALGORITHM FOR LINEAR SYSTEMS OF EQUATIONS

#### A. Problem

Given a matrix  $A$  ( $N \times N$  and condition number  $\kappa$ ) and a vector  $\vec{b}$ , let  $\vec{x}$  be the solution of  $A\vec{x} = \vec{b}$ . Consider, a situation where the actual  $\vec{x}$  is not needed and the problem is to calculate  $\vec{x}^*M\vec{x}$  for some matrix  $M$ .

The fastest known classical algorithm is  $N\sqrt{\kappa}$ . But [1] shows a quantum approach when the runtime is polynomial in  $\log(N)$  and  $\kappa$ .

#### B. Outline of the approach

First, represent  $\vec{b}$  as a quantum state

$$|b\rangle = \sum_{i=1}^N b_i |i\rangle.$$

Next if  $A$  is Hermitian, construct a unitary operator  $e^{iAt}$  by transforming  $A$ . Here,  $A$  is assumed as efficiently row computable and  $s$  sparse. If  $A$  is not Hermitian then construct

$$\tilde{A} = \begin{bmatrix} 0 & A \\ A^* & 0 \end{bmatrix} \rightarrow \tilde{A}\vec{y} = \tilde{A} \begin{bmatrix} \vec{b} \\ 0 \end{bmatrix} \rightarrow y = \begin{bmatrix} 0 & \vec{x} \end{bmatrix}$$

where  $\tilde{A}$  is Hermitian.

Then, we apply  $e^{iAt}$  to  $|b\rangle$  to find the linear combinations at each time  $t$ . This is followed by decomposing  $|b\rangle$  in the eigenspace using phase estimation [2] to get the eigenvalues.

Let  $|u_j\rangle$  be the eigenvectors of  $A$  with eigenvalues  $\lambda_j$ . Now, the system's state is close to

$$\sum_{j=1}^N \beta_j |u_j\rangle |\lambda_j\rangle \text{ and } |b\rangle = \sum_{j=1}^N \beta_j |u_j\rangle.$$

So, a linear map say  $B$  which takes  $|\lambda_j\rangle$  to a multiple of  $\lambda_j^{-1} |\lambda_j\rangle$  is required to undo the phase estimation to get a state proportional to

$$\sum_{j=1}^N \beta_j \lambda_j^{-1} |u_j\rangle = A^{-1} |b\rangle = |x\rangle$$

Finally, the measurement with expectation value  $\langle x|M|x\rangle$  gives the answer.

## V. FUTURE OF QUANTUM MACHINE LEARNING

In IV, we saw a QC technique replacing the entire model. Though it is way more efficient and scalable than classical methods, it is *not feasible* in near future where a quantum device will contain 100-1000 qubits. Even after the exponential speed-up, large-scale models will require millions of qubits and so these algorithms have no practical value (atleast in the near-term).

So for near-term quantum computers, the best way to proceed is for quantum computers to *assist* classical ML models. This is called as Quantum-Assisted Machine Learning (QAML).

There are three things where QAML can focus according to [3] that has the potential for killer applications

- Hard and out of scope problems for ML. E.g., generative models for semi-supervised and unsupervised ML
- Datasets with quantum-like connections, turning quantum computers absolutely necessary. E.g., cognitive sciences
- Hybrid algorithms with a difficult execution step for ML pipeline.

## VI. OPPORTUNITIES IN QUANTUM-ASSISTED MACHINE LEARNING (QAML)

### A. Unsupervised Learning – Quantum Devices for Sampling

Today, there is an incredible amount of unlabelled data available. Courtesy internet, satellite and medical imaging, stock market time series and more due to ever increasing use of computers. There is a need to extract patterns within such data – scientists do not always know what patterns look for, these patterns are essential for the development of science and humanity as a whole. But, it is beyond the human capacity to label all this data. So, a need arises for a machine which is capable of extracting order from disorder.

a) *Generative Model*: Generative Model can learn the joint probability among the variables. If this is achieved, similar data as the training set can be generated. A popular design of such models is using layers of stochastic ‘hidden variables’. It has shown positive results for high-dimensional data, with correct inference of multi-modal distributions over it. Sadly, the exact guess is not possible for non-trivial topologies. Here, the intractable step is computation of expectation values under a complex distribution and this step is a part of each iteration and data point. Markov chain Monte Carlo (MCMC) techniques are being used but they face various issues e.g., slow mixing problem. QAML can possibly solve this problem as they are capable of sampling from probability distributions. Quantum Gibbs distributions are such alternative to MCMC.

### B. Cognitive Sciences - Exploiting Datasets

A quantum model can significantly reduce the computational resources, e.g, memory needed to model a data set. So, real-life data sets where quantum model is simple compared with classical models must be identified.

Cognitive Sciences has candidates to be such dataset, let's discuss one such example

1) *The Two-Stage Gambling Paradigm*: In the first stage, people were required to gamble. If they win they get  $x$  amount of money, if they lose they lose  $y$  amount of money (both are equally likely).

Before learning about the results, participants were asked whether they will ‘plan’ to gamble again (if they win and if they lose). Then after learning the first stage results, final decision was made about playing the second stage was made. The observations are shown in I. ‘Gamble’ column has the payoff. Plan Win (Lose) is the fraction of people planning to take the second stage before learning if they Win (Lose) the first stage. Final Win (Lose) is the fraction of people planning to take the second stage after learning they Win (Lose) the first stage.

As visible in I, the results are dynamically inconsistent. The participants changed their plans; whether they won or lose they thought of rejecting the idea of second stage. This violates the law of total probability!

Gamble		Choice Proportions			
Win	Loss	Plan Win	Plan Loss	Final Win	Final Loss
0.8	1	0.25	0.26	0.2	0.35
0.8	0.4	0.76	0.72	0.69	0.73
2	1	0.68	0.68	0.6	0.75
2	0.4	0.84	0.86	0.76	0.89

TABLE I  
THE TWO-STAGE GAMBLING PARADIGM

Let's see, the way this way resolved using Quantum Model

a) *The Quantum Model*: Unlike the classical models which uses inconsistent utility functions for the explanation of behaviour; the quantum model assumes a consistent utility function is used by participants for both plan and final choices and it also considers the first stage results. Here, the dynamic

inconsistency emerges from *uncertainty* in the first stage results which is solved at the second stage.

Using quantum theory, this game has 4 events  $\{WA, WR, LA, LR\}$  where  $W(L)$  stands for win (lose) in first stage and  $A(R)$  stands for accept (reject) the second round. This represent the four-dimensional vector space with basis vectors as  $\{|WA\rangle, |WR\rangle, |LA\rangle, |LR\rangle\}$ . The person is actually in a superposition of these states  $|\psi\rangle = \psi_{WA}|WA\rangle, \psi_{WR}|WR\rangle, \psi_{LA}|LA\rangle, \psi_{LR}|LR\rangle$  where  $|\psi_{WA}|^2$  is the probability that the person believes he won in the first stage and will accept the second stage.

The initial state is  $\psi_0$  which has some distribution over these 4 amplitudes. As, the first stage result is not known we can take a uniform distribution, i.e.,  $|\psi_{ij}| = \frac{1}{2}$  for all 4  $ij$  pairs. *Uncertainty* in the first stage results is solved at the second stage after learning the result. Now that state is  $\psi_1 = \psi_W = |1100\rangle$  ( $\psi_L = |0011\rangle$ ) if win (lose).

The payoffs can be achieved using a unitary matrix which rotates  $\psi$  towards the gamble or away from it. The final state is

$$\psi_D = U\psi$$

where  $U$  will depend on the utility functions. The inconsistency can be seen using the projection matrix.

Such identification of quantum-like behaviours will be a game changer. Even better, it will make quantum technologies unique in their own applications and thus becoming irreplaceable.

## VII. CHALLENGES IN QUANTUM-ASSISTED MACHINE LEARNING

### A. Compatibility Issues in Hybrid Tech

Information sharing between the classical and quantum might be a challenge, as the samples should from both models should match. In training of restricted Boltzmann machines, stochastic gradient descent algorithm that performs parameter updates requires two major component: ‘positive phase’ which can be estimated efficiently using classical sampling and the ‘negative phase’ has to be assisted with quantum sampling. It is necessary to match and control all the parameters used for describing probability distributions of both the models as the components are part of the same equation and originate from same model. A lot depends on Gibbs distribution temperature as that is the sampling source. This temperature depends on many factors so it is not completely under our control. This challenge can be resolved using hardware that can create many quantum Gibbs states at anytime, but this may open up other issues (See VII-B). A better approach is to carry out a proper estimation of temperature, so that the techniques can be restarted.

### B. Robustness to Noise

Creating quantum Gibbs states with a quantum annealer is a very complicated process due to intrinsic noise in parameters of programming. Dynamical effects and freezing on quantum distributions leads to non-equilibrium distributions away from

required. This intrinsic noise can anyway shift the state away from desired state. Seeding quantum devices using classical Gibbs samplers was a suggested solution, but its major disadvantage is that the model must have a specific form irrespective of the design of the quantum device otherwise there would be a lot of post-processing.

Fully-visible Boltzmann machine (FVBM) is a potential near-term solution. Here, we assume that Gibbs-like distribution sends samples but actually the model is working at first and second moment statistics level. Actually, if there is a positive projection in the direction of actual gradient, stochastic gradient descent’s estimated gradients will work.

A drawback in this approach is that a same device needs to be used for ML tasks.

### C. The Curse of Limited Connectivity

The curse of limited connectivity is a problem of qubit-qubit interactions not on-device which has an additional computational overhead and of setting the parameters to ensure correct sampling and mapping.

### D. Complex Dataset Representation

Common datasets such as images have a large amount of non-binary variables. A simple binarization of data will rapidly eat up 100-1000 qubits. So, instead QAML algorithms do amplitude coding. But this a slow process as just reading all amplitudes will effect speed.

*Semantic Binarization* may be useful, where the encoding is a abstract binary representations of continuous variables. Possibly, implemented using hybrid models.

## VIII. QUANTUM-ASSISTED HELMHOLTZ MACHINE (QAHM)

A hybrid quantum–classical ML which can potentially handle real-world datasets. Its made of a generator network and a recognition network using the notion of stochastic hidden variables. Classification can also be implemented in QAHM.

## REFERENCES

- [1] A. W. Harrow, A. Hassidim, and S. Lloyd, “Quantum algorithm for linear systems of equations,” *Physical Review Letters*, vol. 103, oct 2009.
- [2] R. Cleve, A. Ekert, C. Macchiavello, and M. Mosca, “Quantum algorithms revisited,” *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 454, pp. 339–354, jan 1998.
- [3] A. Perdomo-Ortiz, M. Benedetti, J. Realpe-Gómez, and R. Biswas, “Opportunities and challenges for quantum-assisted machine learning in near-term quantum computers,” *Quantum Science and Technology*, vol. 3, p. 030502, jun 2018.
- [4] C. L. Franklin De Lima Marquezino, Renato Portugal, *A Primer on Quantum Computing*. SpringerBriefs in Computer Science, Springer International Publishing, 1 ed., 2019.
- [5] M. Benedetti, J. Realpe-Gómez, and A. Perdomo-Ortiz, “Quantum-assisted helmholtz machines: A quantum–classical deep learning framework for industrial datasets in near-term devices,” *Quantum Science and Technology*, vol. 3, p. 034007, may 2018.
- [6] J. Busemeyer, Z. Wang, and R. Shiffrin, “Bayesian model comparison favors quantum over standard decision theory account of dynamic inconsistency,” *Decision*, vol. 2, pp. 1–12, 01 2015.