Quantum Algorithms for Machine Learning

A practical guide to implementing quantum machine learning algorithms in Python

Abstract: This document aims at defining the outline of a flagship book with Packt Publishing (hands-on series) on Quantum Machine Learning Algorithms

Author: Laurent QUERELLA, PhD, MSc

Senior Data Scientist and Quantum Machine Learning Explorer

Version: draft version 0.1 – 19-Dec-2022

Table of Contents

About the Author

Full name

Laurent QUERELLA

Job title

Senior Data Scientist and Quantum Machine Learning Explorer

Organization

Freelance, owner of *Ad Infinitum BI* company (SME)

Skillset

- Math/Physics: Quantum computation and quantum information
- Statistics/Machine learning: Supervised/Unsupervised/Bayesian learning, high-dimensional probability techniques, model interpretability, hyper-parameter optimization
- Programming languages Python/R
- Quantum computing Python libraries (incl. quantum machine learning): Qiskit (IBM) – PennyLane (Xanadu)

Qualifications

- PhD in Physics (Mathematical Cosmology)
- MSc in Astrophysics and Geophysics
- QHack2022 coding master (Xanadu)
- IBM Quantum Challenge 2021 Achievement – Foundational
- Certificate of project submission to QC Hack 2021 (Quantum Coalition)
- Quantum Computing & Quantum Machine Learning (self-learning, training, attendance to several conferences since 2019)
    - Last conference, in-person: *Quantum Techniques in Machine Learning* (QTML 2022), Naples (Italy), 7-12 Nov 2022

Work history

- Jun 2015 – Present: Senior data scientist at TotalEnergies (oil & gas major)
- 2014 – 2021: Data scientist advisor at Tangent Works (machine learning start-up)
- 2010 – 2011: Member of the advisory committee at Cryptasc (university spin-off, quantum cryptography)

Projects

1. 2021-2022: Quantum hackathons and challenges
    - PennyLane code camp (2022)
    - QHack2022 (Xanadu)
    - IBM Quantum Challenge (2021)
    - QC Hack (2021) (Quantum Coalition)
    - QHack2021 (Xanadu)

2.	2021-2022:	Quantum versions of classical computing paradoxes (ongoing personal R&D project)
3.	2022:	Exploring practical use cases with quantum machine learning (ongoing personal R&D project)

The Book's Goal

What kind of individual would be interested in this book?

This book is intended for professionals (such as data scientists, computer scientists, and developers) or advanced students and researchers in computer science and related fields who want to use quantum computing techniques and algorithms for machine learning to solve specific challenges in various industries.

What will the reader need to know or understand before reading this book?

The reader will need to have a basic understanding of machine learning algorithms and quantum computing (especially the gate model) together with basic programming skills (Python). Basic knowledge in linear algebra is also required. The reader does not need to be a quantum physicist but knowing quantum physics fundamental principles is beneficial.

What are the 3 major challenges faced by the reader which this book will help them overcome?

- They don't know from where to start to implement quantum algorithms for machine learning
- They don't know which algorithm or technique to pick to address a specific use case
- They want to get hands-on quickly

What makes this title unique compared to other books or free material available in the market?

There are very few books in the market on quantum machine learning algorithms since this field is very recent and is mainly addressed by scientists and researchers at an academic level. This is even more true if one considers hands-on with Python.

This book offers a comprehensive and organized presentation of information with the right balance between theoretical concepts and concrete applications that can be implemented on current available quantum computers.

What are the 3 biggest advantages of this book for a reader?

- Provides a clear understanding of how to implement the most important quantum algorithms for machine learning on their laptop and eventually on a quantum computer
- Get up and running with Python libraries for quantum machine learning
- Provides a complementary -- hands-on oriented – perspective to reference textbooks. In that respect, it is important to provide in-depth content that is not a mere collection of already available resources; this is also what makes the writing of this book an ambitious challenge!

How will these 3 advantages be beneficial to the reader?

- Increase your chances of being noticed as a talented professional (or even as a newcomer) in a domain which exhibits a strong shortage of talents
- Increase your skillset and unlock other ways of thinking to address problems that are hard to solve with known classical techniques
- Progress in implementing quantum machine learning techniques faster than by following formal courses or by thoroughly studying reference textbooks.

TARGET AUDIENCE

This book is intended for professionals (such as data scientists, computer scientists, and developers) or advanced students and researchers in computer science and related fields who want to use quantum computing techniques and algorithms for machine learning to solve specific challenges in various industries. It is assumed that readers have a basic understanding of machine learning algorithms, quantum computing (especially the gate model), programming in Python, and linear algebra. By reading this book, readers will acquire the knowledge and expertise to use Python libraries for quantum machine learning and implement essential quantum algorithms for machine learning on either a personal computer or a quantum computer. With this foundation, readers will be able to take advantage of quantum computing to potentially outperform classical methods in their field, setting themselves apart as highly skilled professionals in a field with a shortage of talented individuals and increasing their visibility and opportunities in the industry.

The Book Structure

NB: I have followed the Packt course “Outline Course” and focused on “hands-on” series.

Chapter Outline

[A practical guide to implementing quantum machine learning algorithms in Python]

IMPORTANT REMARK:

There are two options for the PART 1 outline:

1. Option 1:
    a. Starting with two “refresher” chapters:
        i. Quantum computing in a nutshell (concepts that will be used throughout the book with illustrations and simple code snippets)
        ii. Quantum machine learning in a nutshell (overview of QML in the context of quantum computing and NISQ compliant algorithms – no code at this stage!)
    b. Technical setup and QML libraries walkthrough
2. Option 2: The reverse order:
    a. Technical setup and QML libraries walkthrough
    b. The two “refresher” chapters (as above)

Part 1: Getting started with Quantum Machine Learning (QML)	
1	Quantum computing in a nutshell
2	Quantum machine learning in a nutshell
3	Guided walkthrough of QML libraries & workstation configuration

Part 2: A deep-dive into NISQ compliant algorithms	
4	Basic algorithms as subroutines for quantum machine learning
5	Variational quantum algorithms

6	Quantum kernel methods
7	Quantum deep learning

Part 3: Quantum enhanced data science	
8	Quantum enhanced data science
9	Quantum noise simulation and quantum error mitigation
10	Challenges and further prospects

Detailed outline

Chapter 1: **Quantum computing in a nutshell**

[20-25] pages

Description:

This chapter is a refresher on quantum computing key concepts. It is important for the reader to grasp the use of quantum-mechanical phenomena, such as superposition and entanglement, to perform operations on data. In practical terms, this means using quantum computers to perform calculations and solve problems. The reader should know how to use a quantum computer typically by writing a quantum algorithm or program that specifies the operations one wants the quantum computer to perform. This can be done using a quantum computing programming language, such as Qiskit.

Examples: Readers get the fundamentals of quantum computing in the form of cheat sheets for one-qubit and multiple-qubits universal quantum gates. Some code snippets (Qiskit/PennyLane) are provided as examples.

Chapter Headings

1. HEADING 1: Quantum computing fundamentals: gates and circuits
2. HEADING 2: Quantum algorithms – superposition and entanglement
3. HEADING 3: Quantum computing hardware, software and applications

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Memorize quantum computing fundamentals and gate model with cheat sheets
2. SKILL 2: Know how to use superposition and entanglement to gain a quantum advantage
3. SKILL 3: Understand the capabilities and restrictions of using quantum hardware/software and the importance of NISQ compliance before reaching fault-tolerant hardware; understand the place of QML within the application spectrum

Chapter 2: Quantum machine learning in a nutshell
[20-25] pages
Description:

This chapter introduces the reader to quantum machine learning (QML) and its motivations. The reader will gain an understanding of QML applications and how QML algorithms leverage the unique properties of quantum computers, such as superposition and entanglement, to perform tasks faster and more efficiently than classical algorithms. The chapter will also cover the specificities of quantum neural networks, quantum kernel methods, and quantum clustering and classification algorithms, before moving on to the second part of the book.

Examples: Readers will get the fundamentals of quantum machine learning (in the form of cheat sheets for the taxonomy of QML algorithms) with some code snippets (Qiskit/PennyLane).

Chapter Headings (3-5 main chapter headings)

1. HEADING 1: What is quantum machine learning?
2. HEADING 2: Quantum machine learning algorithms and techniques - Taxonomy
3. HEADING 3: Quantum machine learning software and applications

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Understand the motivation behind QML
2. SKILL 2: Get a comprehensive overview of quantum algorithms for machine learning summarized as a taxonomy of QML algorithms cheat sheet
3. SKILL 3: Understand the types of applications that can be addressed using QML and familiarize yourself with the various QML platforms and software available for experimentation

Chapter 3: Guided walkthrough of QML libraries & workstation configuration
[15-20] pages
Description:

In this chapter, we will provide a detailed overview of three popular quantum computing libraries used for QML: Qiskit, PennyLane, and TensorFlow Quantum. We will discuss the unique features and capabilities of each library, as well as provide guidance on setting up a workstation for using them. We will also highlight the types of applications and experiments that can be explored with these libraries.

Examples: Configuration steps for each quantum computing environment. Code snippets exemplifying the unique features of Qiskit, PennyLane, and TensorFlow Quantum for QML applications or experiments.

Chapter Headings

1. HEADING 1: Installation and practical use of Qiskit for QML
2. HEADING 2: Installation and practical use of PennyLane
3. HEADING 3: Installation and practical use of TensorFlow Quantum
4. HEADING 4: Complementary information on other programing languages and libraries

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Learn how to install, configure, and use a Qiskit environment for QML usage
2. SKILL 2: Learn how to install, configure, and use a PennyLane environment for QML usage
3. SKILL 3: Learn how to install, configure, and use a TensorFlow Quantum environment for QML usage
4. SKILL 4: Receive guidance on other programing languages and libraries (e.g., the SDK of the service Amazon Braket, rewritten in Julia)

Chapter 4: Basic algorithms as subroutines for quantum machine learning
[35-40] pages
Description:

In this chapter, we will introduce the reader to the basic algorithms that are commonly used as subroutines in QML. We will focus on algorithms that are compatible with Noisy Intermediate-Scale Quantum (NISQ) computers, which are currently available and able to perform certain quantum computations but are limited in their capabilities due to noise and errors. We will provide a detailed explanation of how each algorithm works and will provide guidance on how to implement these important subroutines in code.

Examples: Explanation and code implementation of each basic quantum subroutine.

Chapter Headings

1. HEADING 1: Fault-tolerant vs NISQ compliant subroutines

2. HEADING 2: Quantum Fourier transform (QFT)
3. HEADING 3: Quantum phase estimation (QPE)
4. HEADING 4: Grover search algorithm
5. HEADING 5: Hamiltonian simulation

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Understand the difference between fault tolerant and NISQ compliant algorithms
2. SKILL 2: Learn and implement in code the QFT subroutine
3. SKILL 3: Learn and implement in code the QPE subroutine
4. SKILL 4: Learn and implement in code the Grover search subroutine
5. SKILL 5: Learn and implement in code the Hamiltonian simulation techniques

Chapter 5: Variational quantum algorithms

[35-40] pages

Description:

This chapter will explore *variational quantum algorithms* (VQAs), which use a classical optimizer to train a parameterized quantum circuit and have become a popular approach for addressing the limitations of current quantum devices, such as a limited number of qubits and noise processes that limit circuit depth. VQAs have been suggested for a variety of applications and are considered a potential means of achieving quantum advantage or uncovering new insights in machine learning. We will explain the principles behind VQAs and demonstrate how to implement them in code with real-world applications.

Examples: Explanation and code implementation of variational quantum eigensolver (VQE).

Chapter Headings

1. HEADING 1: VQA common structure and methodology
2. HEADING 2: Quantum approximate optimization algorithm (QAOA)
3. HEADING 3: Variational quantum eigensolver (VQE)
4. HEADING 4: Variational quantum generator (VQG) or generative modelling (QGANs)

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Understand how VQAs work and learn how to make use of classical optimizers to learn a cost function
2. SKILL 2: Solving optimization problems including route selection
3. SKILL 3: Predicting properties of molecules and materials

4. SKILL 4: Generating representative data such as text, images, audio, etc.

Chapter 6: Quantum kernel methods
[30-35] pages
Description:

In this chapter, we will explore various techniques for using quantum computers to perform kernel-based machine learning algorithms, with a focus on classification tasks and feature mapping. We will examine how quantum computers can be used to directly compute inner products between data points in the feature space or to map the data into a higher-dimensional space for improved separability and classification. It is important to note that quantum kernel methods are still an active area of research and their potential for improving machine learning applications is yet to be fully understood.

Examples: Explanation and code implementation of quantum support vector machine (QSVM).

Chapter Headings

1. HEADING 1: Quantum kernel methods and hybrid computation
2. HEADING 2: Classification with quantum kernels
3. HEADING 3: Clustering with quantum kernels
4. HEADING 4: Regression with quantum kernels

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Understand what quantum kernel methods are and learn how to implement them with various approaches
2. SKILL 2: Solving a classification task with quantum kernels
3. SKILL 3: Finding patterns in data with quantum kernels
4. SKILL 4: Perform regression with quantum kernels

Chapter 7: Quantum deep learning
[30-35] pages
Description:

In this chapter, we will explore how quantum computers can be trained like neural networks. We will also explore the capabilities of differentiable programming and differentiation of quantum circuits. The reader will learn how to implement different types of quantum neural networks (QNNs): e.g., quantum convolution (QCNN), quantum autoencoder, quantum reinforcement learning.

Examples: Code implementation of a quantum autoencoder.

Chapter Headings

1. HEADING 1: Quantum neural network architecture and variational methods
2. HEADING 2: Quantum reinforcement learning
3. HEADING 3: Quantum convolution
4. HEADING 4: Quantum autoencoder

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Understand what quantum deep learning is and learn how to implement QNNs
2. SKILL 2: Implementing a quantum reinforcement learning application
3. SKILL 3: Implementing a quantum convolution application
4. SKILL 4: Implementing a quantum autoencoder application

Chapter 8: **Quantum enhanced data science**
[25-30] pages

Description:

In this chapter, we will delve into the field of *quantum enhanced data science*, which involves using quantum computing techniques to improve data science algorithms. We will focus on the potential of quantum techniques to solve problems that are currently infeasible or impractical with classical methods, such as those requiring the processing of large or complex data sets or the optimization of highly multi-dimensional functions. The reader will also gain an understanding of the challenges involved in quantum data preprocessing and feature engineering.

Examples: Dimension reduction using quantum principal component analysis (QPCA).

Chapter Headings

1. HEADING 1: Quantum data preprocessing techniques
2. HEADING 2: Quantum feature engineering techniques
3. HEADING 3: Quantum dimensionality reduction

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Implementing quantum data preprocessing techniques
2. SKILL 2: Implementing quantum feature engineering techniques
3. SKILL 3: Implementing quantum dimensionality reduction

Chapter 9: Quantum noise simulation and quantum error mitigation

[25-30] pages

Description:

This chapter focuses on how to mitigate quantum noise in order to improve the reliability and accuracy of quantum machine learning algorithms. We will explore various techniques for quantum error mitigation, such as error correction codes, dynamical decoupling, and quantum filtering, and the reader will have the opportunity to implement these techniques and to simulate the noise model for real quantum hardware.

Examples: Zero-noise extrapolation (ZNE) method.

Chapter Headings

1. HEADING 1: Introduction to quantum error mitigation and quantum noise
2. HEADING 2: Techniques for quantum error mitigation
3. HEADING 3: Applications of quantum error mitigation

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Understand the motivations for dealing with quantum noise and quantum error correction
2. SKILL 2: Implement techniques for quantum error mitigation
3. SKILL 3: Examine how this is important for QML

Chapter 10: Challenges and further prospects

[25-30] pages

Description:

This chapter will provide a brief overview of the main challenges and limitations of quantum machine learning (QML). One of the main challenges is the limited availability of large-scale quantum computers, which restricts the size and complexity of problems that can be solved using QML algorithms. Another challenge is the impact of quantum noise and errors on the accuracy of QML algorithms. There are also limitations in the QML algorithms that are currently available, such as trainability issues, expressivity and generalization challenges. To address these limitations, we will examine the use of symmetries of the problem as a potential solution.

Examples: geometric quantum machine learning (GQML).

Chapter Headings

1. HEADING 1: Limitations of the quantum hardware

2. HEADING 2: Limitations of the QML algorithms
3. HEADING 3: Future directions

Skills learned: *For each heading, insert what the reader will learn to DO in this chapter?*

1. SKILL 1: Understand the limitations of the quantum hardware
2. SKILL 2: Understand and experiment the limitations of QML algorithms
3. SKILL 3: Explore future directions

Reviewers

Technical reviewers

Can you recommend peers and members of your community to become technical reviewers?

Full name	Email Address	LinkedIn Profile
1.		
2.		
3.		

Amazon Reviewers

Can you recommend peers and members of your community to leave Amazon reviews?

Full name	Email Address	LinkedIn Profile
1.		
2.		
3.		

Influencers

Can you recommend any influential community members or organisations for Packt to collaborate with on the marketing campaign of your title?

Full name	Email Address	LinkedIn Profile
1.		
2.		
3.		