

Review

Quantum Machine Learning—An Overview

Kyriaki A. Tychola , Theofanis Kalampokas and George A. Papakostas * 

MLV Research Group, Department of Computer Science, International Hellenic University, 65404 Kavala, Greece; kytzcho@cs.ihu.gr (K.A.T.); theokala@cs.ihu.gr (T.K.)

* Correspondence: gpapak@cs.ihu.gr; Tel.: +30-2510462321

Abstract: Quantum computing has been proven to excel in factorization issues and unordered search problems due to its capability of quantum parallelism. This unique feature allows exponential speed-up in solving certain problems. However, this advantage does not apply universally, and challenges arise when combining classical and quantum computing to achieve acceleration in computation speed. This paper aims to address these challenges by exploring the current state of quantum machine learning and benchmarking the performance of quantum and classical algorithms in terms of accuracy. Specifically, we conducted experiments with three datasets for binary classification, implementing Support Vector Machine (SVM) and Quantum SVM (QSVM) algorithms. Our findings suggest that the QSVM algorithm outperforms classical SVM on complex datasets, and the performance gap between quantum and classical models increases with dataset complexity, as simple models tend to overfit with complex datasets. While there is still a long way to go in terms of developing quantum hardware with sufficient resources, quantum machine learning holds great potential in areas such as unsupervised learning and generative models. Moving forward, more efforts are needed to explore new quantum learning models that can leverage the power of quantum mechanics to overcome the limitations of classical machine learning.

Keywords: quantum machine learning; quantum classifier; quantum computer; quantum support vector machine



Citation: Tychola, K.A.; Kalampokas, T.; Papakostas, G.A. Quantum Machine Learning—An Overview. *Electronics* **2023**, *12*, 2379. <https://doi.org/10.3390/electronics12112379>

Academic Editors: Shi-Lei Su, He-Liang Huang, Chu Guo and Zu-En Su

Received: 27 March 2023

Revised: 19 May 2023

Accepted: 23 May 2023

Published: 24 May 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The term “quantum” in physics refers to the smallest discrete unit of a “physical quantity”. Quantum particles are dualistic (wave-particle) and fall under quantum theory, which deals with finding the probability of a quantum particle at a given point in space [1]. In recent decades, Quantum Machine Learning (QML) has been developing and evolving in computer science, as it is linked to Machine Learning (ML), where data is processed and analyzed using various decision-making models. With data volume increasing by around 20% per year, it is necessary to manage it properly [2].

The idea of optimizing a limited multivariable function without programming was proposed by A. Samuel in 1959 [3]. Since then, this concept has been the basis of machine learning, where algorithms are implemented to match the input and output points in order to create decision functions. In QML, the most common supervised algorithm is the Quantum Support Vector Machine (QSVM) [4], which uses the vector space optimization limit of the higher dimension to classify labeled data categories. QSVM creates patterns of unlabeled data and reduces them for easier analysis [5].

Machine learning techniques create patterns in data, while quantum systems produce informal patterns and investigate how to create and implement quantum software for faster machine learning [2]. In the 20th century, computers were built to analyze mathematical models using several techniques and methods. Artificial neural networks were implemented in the 1950s [6], and from 1960 to 1990, deep learning was proposed (Hopfield networks, Boltzmann machines) using the backpropagation method [7]. Recently,

significant concerns have been defined for the security and robustness of machine learning models in critical applications [8–10].

Quantum computers excel at solving complex algebraic expressions, such as the factorization of large integers and the computation of discrete logarithms, leveraging their computational power [11]. These impressive capabilities stem from the unique axioms and characteristics of quantum mechanics, such as quantum bits (qubits), interpolation, superposition, and entanglement. Quantum computers process information using qubits, enabling breakthroughs in various scientific domains [12,13], including cryptography, big data analysis, machine learning, optimization, the Internet of Things (IoT), and Blockchain.

The simultaneous storage of several qubit states forms the basis of quantum parallelism. The entanglement and interference of quantum states combined with the above accelerate quantum computation. Quantum computation has evolved rapidly since Feynman first claimed it as an effective means of simulating complex quantum systems [14,15]. The first QML application was presented in 2014, where the quantum version of the SVM algorithm, QSVM, sorted large amounts of data [4]. Another learning algorithm based on superposition and quantum operator developed was the Superposition-based Architecture Learning algorithm (SAL) [16]. Search algorithms such as QKNN (Quantum K-Nearest Neighbors) [17] and decision tree classification [18] have also been proposed.

In addition to the distinction between classical machine learning (ML) and quantum machine learning (QML), researchers have also explored the hybrid approach, which combines both quantum and classical algorithms to achieve optimal performance. Extensive research has been conducted in this exciting area, yielding promising results. A notable study has been conducted in this area, where a classifier was presented for encoding data in N-dimensions with the help of the training algorithm [19]. In another study, the authors introduced the Hybrid Quantum Feature Selection Algorithm (HQFSA), which is based on subroutines for the selection of features [20]. Another approach is adiabatic QML, which is suitable for certain classes of optimization problems [21–23].

Quantum technologies also include quantum cryptography, with the most famous being Quantum Key Distribution (QKD) (BB84 protocol) [24,25]. With the appearance of quantum computers, researchers started to focus on post-quantum cryptography (PQC), which is the pre-design and development of quantum encryption algorithms [26,27]. With the Shors algorithm [28] and a powerful quantum computer, every known encryption algorithm like RSA or ECC could be exposed since they were designed based on known mathematical problems like elliptic-curve discrete logarithm problems or integer factorization. The algorithms implemented in quantum machine learning are not entirely quantum but correspond to classical methods adapted by differentiations.

The purpose of this paper is two-fold. Firstly, it offers an overview of Quantum Machine Learning and its connection to classical Machine Learning. Secondly, it presents a series of experiments conducted on three datasets, including one tested for the first time, to compare the classical and quantum versions of the SVM ML model. The experimental part is conducted to highlight QML and CML connections from a more technical perspective and verify their differences and limitations through experimental results.

The remainder of this paper is structured as follows: The introduction section establishes the research scope, performs a literature analysis, and outlines the main contributions of this study. Section 2 describes the basic concepts of Quantum Machine Learning. Section 3 presents quantum algorithms and their applications, focusing on the Support Vector Machine and Quantum Support Vector Machine. Section 4 introduces quantum learning methods, and Section 5 provides a critical analysis of their advantages and limitations. Section 6 presents the conducted experiments, describes the applied methods, and evaluates the results. Finally, Section 7 concludes the study with potential conclusions and future directions.

1.1. Motivation and Contribution

Quantum machine learning has become an increasingly popular research topic in the computer science field. While the roots of this field date back to earlier days, recent research has focused on various techniques and methods of quantum machine learning, including supervised and unsupervised algorithms. These studies aim to combine and compare classical and quantum machine learning algorithms through experiments on different datasets with complex features.

One notable theoretical study by Ciliberto et al. [29] examines the computational costs and the need for data transformation in quantum machine learning. Other research has investigated quantum algorithms, such as Quantum SVM, which offers quantum speed-up in machine learning applications [30]. Schuld et al. [31] explain quantum machine learning algorithms for handling big data, while Harenstein et al. [32] compare the performance of machine learning algorithms executed on traditional and quantum computers. The latter concludes that quantum multi-class SVM classifiers offer advantages for future quantum computers with large numbers of qubits.

In data classification, Support Vector Machines (SVM) and Quantum SVM [2] are the most common methods used in classical and quantum machine learning, respectively. Many classical and quantum SVM algorithms have been developed to solve classification problems, as shown in various benchmark studies. For instance, [33] implements a quantum support vector machine (QSVM) algorithm with the MNIST dataset of handwritten digits and compares classical and quantum SVM algorithms in terms of execution time and accuracy. Saini et al. [34] implement a QSVM-based classification model on a breast cancer dataset and compare QSVM accuracy to SVM. Havlicek et al. [35] experiment with quantum algorithms on noisy quantum computers, while Tang [36] designs a QML-based recommendation algorithm that can achieve exponential improvement.

Other researchers propose new algorithms and methods to improve the accuracy and efficiency of classical and quantum machine learning. For example, Shan et al. [37] designed a QSVM algorithm and proposed a quantum kernel estimation method with measurement error mitigation using the breast cancer dataset. Kumar et al. [38] implement classification models using QSVM and classical SVM on quantum and classical computational backends on three constructed datasets. In [39], the authors examine the execution speed and accuracy of quantum support vector machines compared to classical SVM by proper quantum feature mapping selection using IBMQ quantum computer. Batra et al. [40] experiment with a large dataset, such as the drug dataset, which is hard for classical computations, to compare quantum and classical computations using the SVM algorithm. Finally, Suzuki et al. [41] propose a method to analyze the feature map for the kernel-based quantum classifier with two qubits using the SVM algorithm, while Bay et al. [42] propose a quadratic kernel-free least squares support vector machine (QLSSVM) to solve binary classification problems using the heart disease dataset.

The aforementioned works have produced interesting results. Some of these studies focus on the comparison between classical and quantum machine learning algorithms only theoretically, while others include experiments implementing classical and quantum classifiers such as SVM and QSVM to solve binary classification problems. However, all researchers evaluate and focus on the computational cost and accuracy, and they conclude that machine learning can harness the advantages offered by a quantum algorithm.

Motivated by the growing interest in quantum machine learning and its potential to revolutionize the field of machine learning, this paper aims to provide an overview of the current state of QML from a historical, theoretical, and technical perspective. Specifically, we focus on the implementation and comparison of classical machine learning algorithms, particularly Support Vector Machines, with their quantum counterparts, known as Quantum Support Vector Machines (QSVM). By comparing classical and quantum machine learning models, we aim to evaluate the current state of the QML field while highlighting the corresponding challenges and limitations.

To achieve this goal, we address several critical research questions, including:

- RQ1: To what extent is classical computing combined with quantum computing?
- RQ2: Does quantum machine learning provides speed increases compared to classical machine learning?
- RQ3: Does combining classical and quantum approaches lead to increased accuracy, or are there cases where classical machine learning performs better?
- RQ4: What are the advantages and limitations of quantum machine learning?

By providing answers to these questions, we aim to provide a better understanding of the current state of the QML field and the potential benefits and limitations of using quantum approaches in machine learning.

1.2. Literature Analysis

This paper presents a systematic literature review that explores the current state of Quantum Machine Learning (QML) from historical, theoretical, and technical perspectives. To ensure a thorough analysis, specific criteria were applied based on relevant keywords found in the titles of publications. The research methodology employed for this review was conducted in March 2023, and it included an initial search on Scopus with the following query: TITLE-ABS-KEY (quantum AND machine AND learning) OR TITLE-ABS-KEY (quantum AND svm) OR TITLE-ABS-KEY (quantum AND classifiers) (6171 papers). We limited the results to journal articles, conference papers, and book chapters only in English (5079).

Scopus includes the most important digital libraries (Elsevier, Springer, IEEE), provides a refined search, and facilitates the export of files, focusing on the years 2000–2023.

We conducted more detailed searches on Scopus specific to our paper, such as QSVM and SVM classifiers, evaluated them, and gathered any relevant works referenced in our paper. This process yielded around 84 papers that are included in this systematic review.

Quantum computing is a rapidly advancing field of research. Scientists are exploring ways to solve complex problems quickly, efficiently, and accurately using quantum systems. Many different quantum algorithms have been developed to address a wide range of computational challenges. Some key milestones in the history of quantum computing include Feynman's observation in the early 1980s that classical computers cannot simulate quantum phenomena [43], Deutsch's demonstration in 1985 of the universality of quantum computing [44], Shor's 1994 development of an algorithm for prime factorization that is exponentially faster than classical algorithms [28], and Grover's 1996 algorithm for quantum cryptography [45]. One of the major challenges of quantum computing is achieving faster computation speeds, and significant research efforts have been dedicated to this goal. In 2019, Google announced Quantum Supremacy, a significant milestone that demonstrated the superior computational power of quantum systems compared to classical computers [46].

The number of publications on quantum computing has increased significantly since 2011, as illustrated in Figure 1, depicting the articles published in the Scopus database until early 2022. Specifically, the first decade, from 2000 to 2010 exhibited a comparatively lower number of publications. However, from 2011 onwards, the interest of researchers has noticeably increased, reaching its peak in 2022, which recorded the highest number of publications to date. However, there has been a slight decrease in publications in 2016.

Below are the first ten countries that have published articles on QML, with the United States and China singling out almost 50% (Figure 2), and finally, all countries that published articles from 2000 to 2023 (Figure 3).

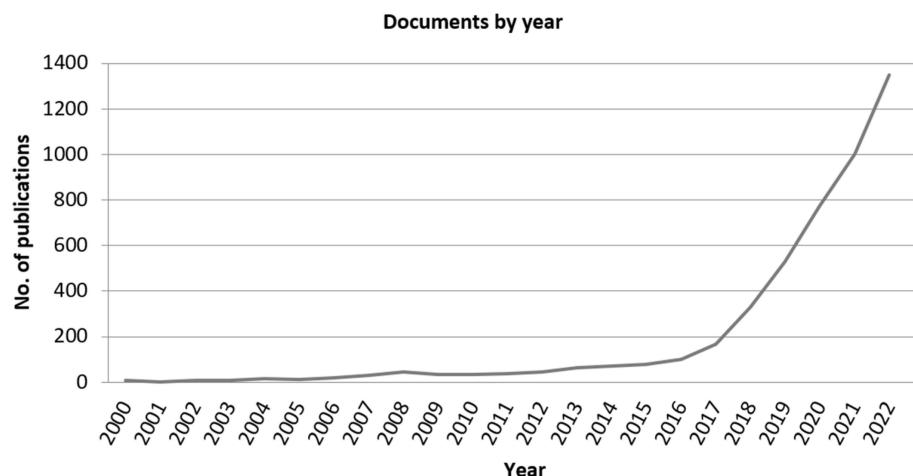


Figure 1. Number of relevant publications per year on the subject of QML (Statistics March 2023).

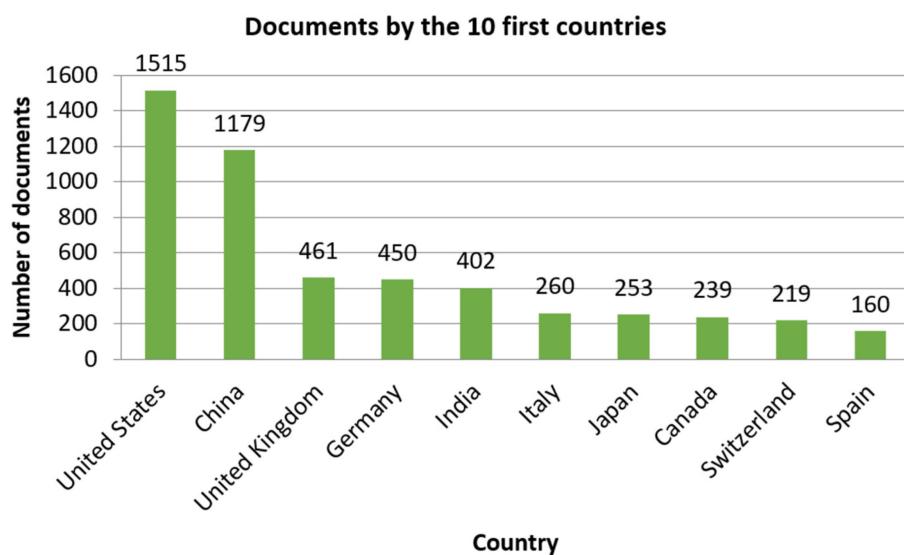


Figure 2. Countries that have published (Scopus source) papers related to QML.

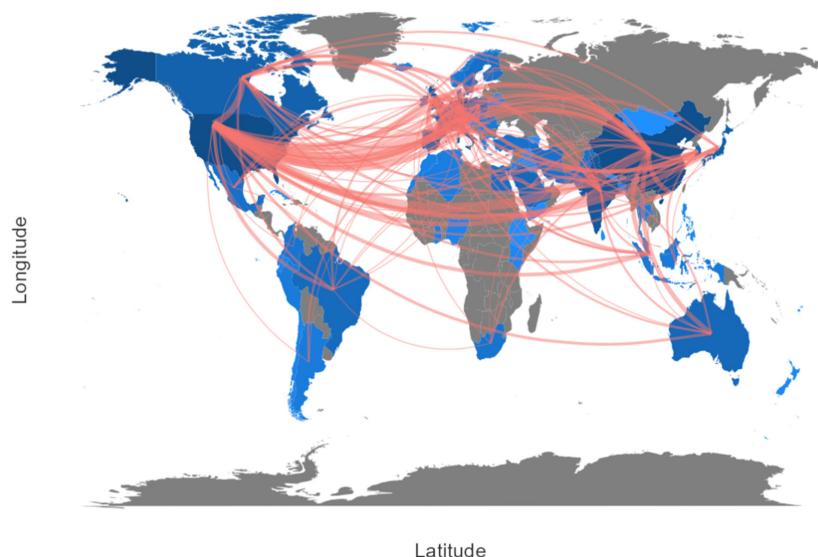


Figure 3. Countries and collaboration network that have published researchers in Scopus related to QML.

Figures 2 and 3 demonstrate that the United States and China are at the forefront of QML contributions. This finding is justified by prominent quantum hardware providers in these countries.

Quantum machine learning has a long and fascinating history that runs almost parallel to classical machine learning. QML began to take shape before the advent of quantum computers in 1900, and since then, it has been growing steadily, with several landmark contributions that led to the development of quantum computing.

In 1981, physicist Richard Feynman gave a lecture on the possible benefits of computation with quantum systems and the simulation of the physical properties of matter [15,43]. Four years later, physicist David Deutsch published the idea of a universal quantum computer. In 1994, mathematician Peter Shor developed an algorithm for finding the prime factors of large numbers, which demonstrated the immense potential of quantum computers [11,28]. In 1996, mathematician Lov Grover introduced an algorithm that optimizes search in an unstructured database [45]. In 1998 [46], Jones, Mosca, and Hansen of Oxford University executed Grover's algorithm on a 2-qubit quantum computer.

In 2001, IBM partnered with Stanford University to publish and implement the Shor algorithm on a 7-qubit processor [47]. In 2012, physicist John Preskill described the moment when controlled quantum systems perform tasks that transcend classical ones [48]. In 2019, Google announced the achievement of quantum supremacy, a milestone in which a quantum computer can perform a task that is beyond the capability of classical computers [49].

Today, QML has evolved into a new concept known as quantum brain-inspired machine learning, which aims to develop artificial algorithms that use interactions and dynamics of a quantum system as a direct resource of learning, mimicking the computation of the brain (Figure 4).

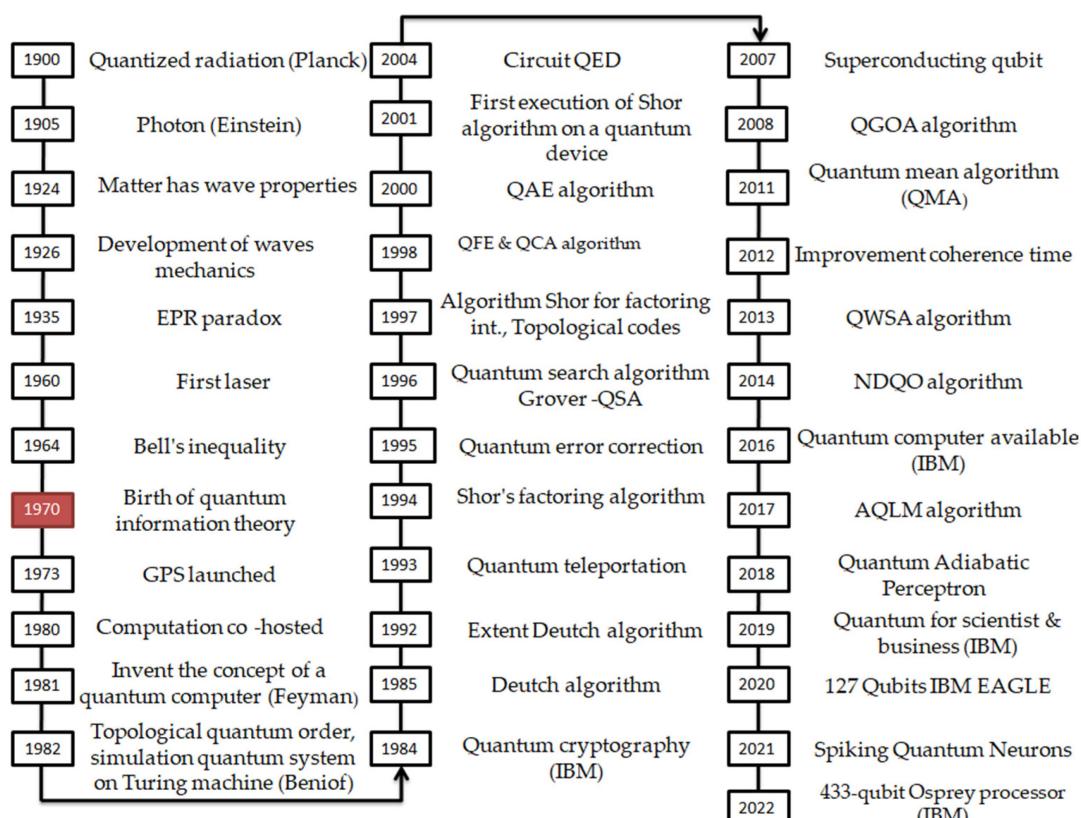


Figure 4. Timeline progression of QML milestones. Diagram highlighting major advancements in quantum machine learning.

2. Quantum Machine Learning—The Basic Concept

Quantum Machine Learning, classical Machine Learning, and Quantum Computing are interconnected, with the shared goal of developing more accurate and reliable models. Learning algorithms are designed to identify patterns in data for making predictions and decisions. The power of quantum computing lies in its qubits, which cannot be copied and have no ramifications or feedback loops [50]. Quantum computation is represented by quantum registers, gates, and circuits that consist of qubits and denote the chronological order and mode of action of the gates and registers [51]. A quantum register comprises a set of unordered qubits that simultaneously store all their states, with the elements numbered clockwise [52]. The SWAP gate is central to designing networks for the quantum computation of qubits, which carries out multiple qubit gates to create a base. The multi-qubit and CNOT gates are typical features of quantum computation [53], with the SWAP gate being crucial in the network design of Shor's algorithm [54,55]. Recently, it has been suggested that generalizing quantum computation to higher-dimensional systems may offer advantages [56]. At the heart of Machine Learning is the extraction of information from data distributions without being explicitly programmed. Therefore, harnessing quantum phenomena is necessary [2], which is accomplished by developing quantum algorithms that implement classical algorithms using a quantum computer. In this way, data can be classified and analyzed by supervised and unsupervised learning methods using Quantum Neural Networks (QNNs) [57,58]. Variational Quantum Circuit (VQC) is a quantum gate circuit with free parameters that approximate, optimize, and classify various arithmetic tasks. The VQC-based algorithm is known as the Variational Quantum Algorithm (VQA), a classical quantum hybrid algorithm where parameter optimization typically occurs on classical computers [59]. The VQA approaches the target function using learning parameters with quantum characteristics, such as reversible linear gate operations and multi-layer structures that use layers of engagement. VQC has been used to replace existing Convolutional Neural Networks (CNNs) [60,61], with QNNs being defined as a subset of VQA, and a general expression of the QNN quantum circuit is presented in Figure 5 [62,63].

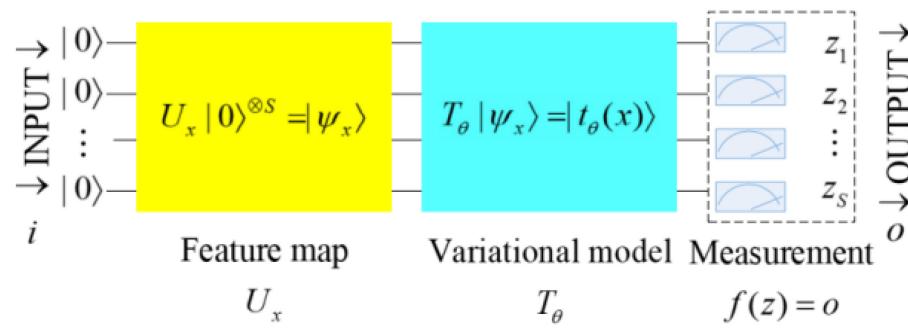


Figure 5. QNN based on VQA framework [63].

While VQA continues to be a prominent approach for designing QNNs, it also inherits some of its drawbacks. For instance, the QNN framework currently faces the issue of the barren plateau, but specific solutions to this problem have yet to be proposed. Additionally, the measurement efficiency of quantum circuits has not been thoroughly investigated, which remains a challenge for QNN designers.

3. Quantum Machine Learning—Algorithms and Applications

Quantum machine learning algorithms are applied in the domains of Supervised Learning, Unsupervised Learning, and Reinforcement Learning (RL). In Supervised Learning, patterns are learned by observing training data, whereas in Unsupervised Learning, the structure is recognized from a set of grouped data. In RL, the algorithm learns from direct interaction with the environment [64]. Additionally, a technique called deep-supervised learning trains QNN to recognize patterns and images. It is a feed-forward network that

employs circuits with qubits (based on neurons) and rotational gates (proportional to weights). On the other hand, Classical deep learning (CDL) uses complex multi-level neural networks, and a deep learning algorithm constructs multiple levels of abstraction from large datasets. Boltzmann machines (BM) are a well-known class of deep-learning networks where the nodes of graphs and connections are established by the Gibbs distribution [65] (Figure 6). The method aims to minimize the maximum distribution probability using the gradient descent method, which ensures consistency between the model and the training data [66].

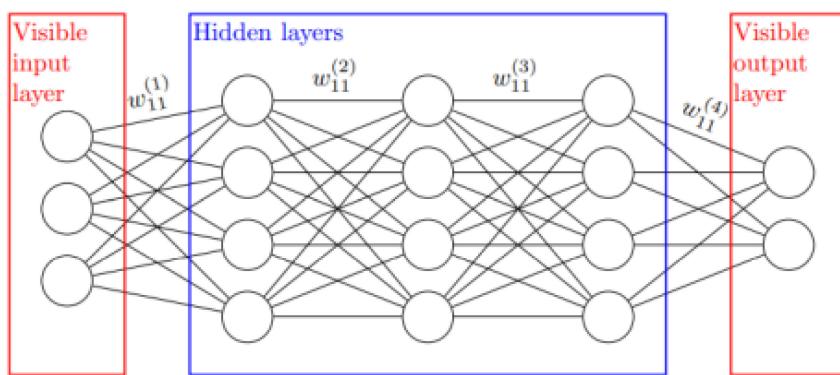


Figure 6. Boltzmann Neural Network. A deep learning neural network with three hidden layers. Each layer is specified as a vector of binary components, and the edges between the vectors are defined as a matrix of weight values. The configuration space of the graph is given by a Gibbs distribution with an Ising-spin Hamiltonian [65].

Recently, Wiebe et al. [67] have developed two quantum algorithms, namely BM and QDL, which efficiently calculate the distribution of data. In BM, the state is initially approximated using the classical mean-field method before being fed into the quantum computer and applying sampling to calculate the required gradient. On the other hand, the second algorithm performs a similar process but requires access to the superposition training data (via QRAM) and provides more accurate solutions but not acceleration. This procedure is equivalent to an attribute map that assigns data to vectors in Hilbert space [68,69]. The inner products of such quantum states encode data and create kernels [35].

Fuzzy cognitive maps (FCMs) have been introduced as a quantum-inspired machine learning model belonging to the category of expert systems. Quantum Fuzzy Cognitive Maps (QFCM) were initially introduced in 2009, presenting a quantum-based approach to cognitive maps [2,70]. In this framework, each concept is represented by a single qubit, and the concept value is computed through qubit superposition. In 2015, the QFCM model was further developed as an ensemble classifier [71], which outperformed other models such as AdaBoost and Neural Networks, demonstrating increased robustness against noise. Additionally, in the same year, a variant of FCM called *bipolar quantum FCMs* was proposed [72]. In 2018, the authors of bipolar quantum FCM explored the application of the quantum cryptography problem [73], where bipolar quantum FCMs performed well in comparison with other methods. Although QFCMs may not strictly fall within the domain of QML, most implementations are inspired by quantum principles, even though explicit proposals for execution on quantum computers are not prevalent.

A well-defined example of QML is the QSVM algorithm, which utilizes a quantum processor to estimate the kernel directly in the quantum space. This method involves a training phase where the kernel is computed, and support vectors are determined. Subsequently, the unlabeled data is classified based on the solution obtained during the training phase [74]. The algorithm is capable of performing binary or multi-class classification based on the data classes and can even be utilized for data clustering and exploration.

Machine learning classifications are versatile tools with applications in diverse fields, such as computer vision, medical imaging, drug discovery, handwriting recognition, geo-

statistics, and more. Quantum computers hold the potential to help overcome the challenges of support vector machine (SVM) and kernel learning, as previous surveys have shown that quantum computation can exponentially accelerate SVM training. Quantum SVM and kernels can efficiently explore high-dimensional spaces, creating maps and decision boundaries (Figure 7a) for specialized datasets in line with their design objective; this task is difficult for classical kernel functions to match [75]. This mapping of classical data to the Hilbert space is illustrated using the Bloch sphere, as shown in Figure 7b.

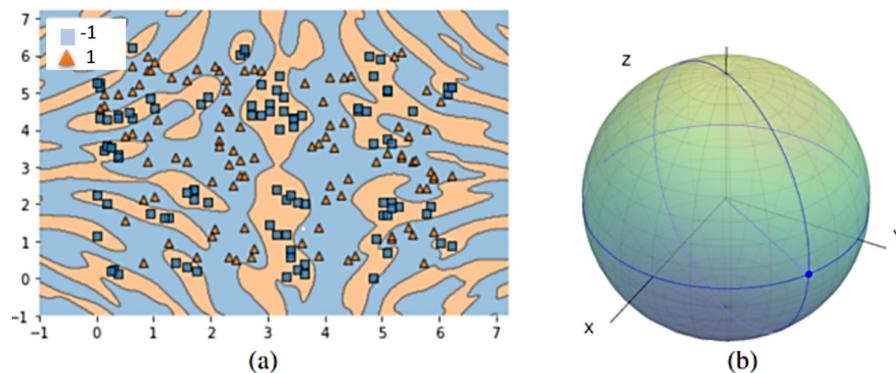


Figure 7. (a) QSVM decision boundary (b) The Bloch sphere [75].

SVM Kernels and Quantum SVM

The SVM algorithm can be implemented in two ways: using a kernel or using a quantum processor (QSVM). In cases where the data set is non-linear and cannot be handled by a linear classifier, the distance between each point and the center is calculated to create a new feature, which enables classification in a higher-dimensional space (see Figure 8).

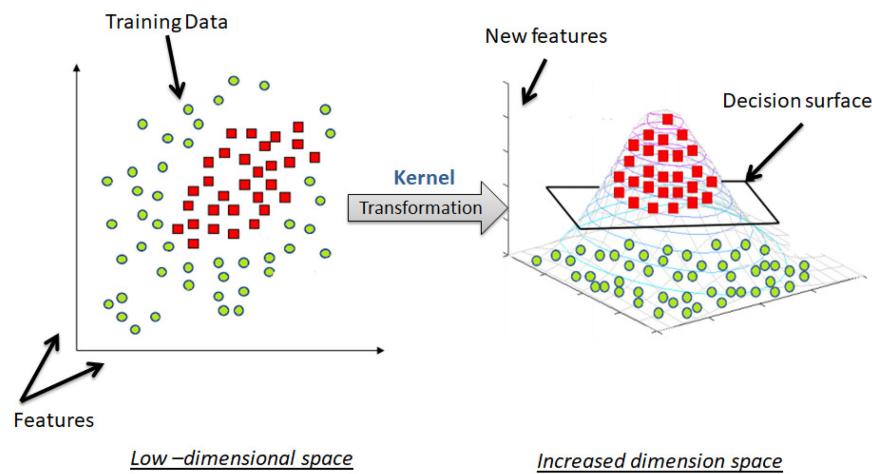


Figure 8. SVM kernel method for the case of 2 classes (1st class: red squares, 2nd class green circles).

The kernel function maps an input feature space into a new, possibly higher-dimensional space where the training dataset can be better separated. For SVMs, the first step involves preparing the training dataset and mapping features into the range $[-1, 1]$, followed by kernel optimization to minimize the cost function. The final step is to test the model. QSVMs follow a similar process, but instead of using a classical computer to evaluate the kernel function, qubits are used to encode the feature space, and the quantum computer performs the kernel evaluation. The attribute mapping is done by encoding data onto the quantum state of each qubit, allowing for the efficient calculation of the kernel matrix. Figure 9 illustrates the nonlinear features classification to higher dimensions to observe

new feature differences. In CML, these features are clustered. However, when they move in 3D space, they are actually scattered.

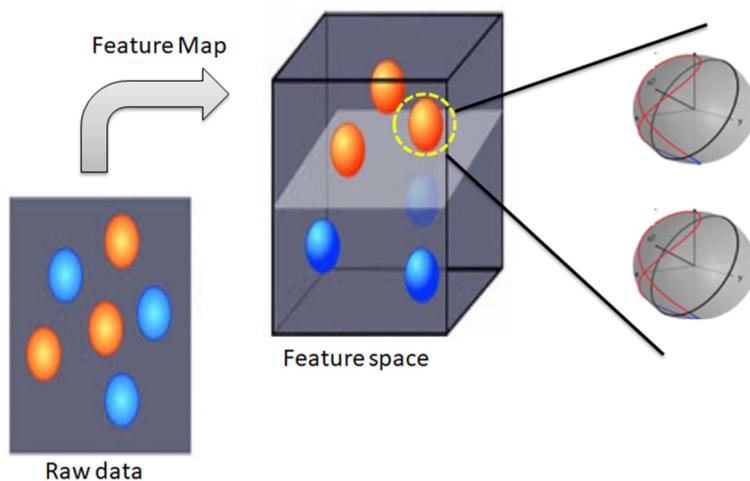


Figure 9. Quantum feature map: Feature classification using nonlinear kernel for the case of 2 classes (1st class: orange balls, 2nd class: blue balls).

Mapping is achieved with a single gate, and this leads to the creation of a quantum circuit. The kernel is computed by selecting a reference point x and encoding the remaining data points relative to it using quantum states. The resulting circuit is then reversed to return to the zero state with a wider width that depends on the distance between the x and z states, giving rise to the kernel value. The weights of the QSVM model are optimized by minimizing a cost function using classical optimization techniques, similar to the classical SVM algorithm [76].

The second approach involves utilizing the Variational Quantum Eigensolver (VQE), a hybrid quantum-classical computational method designed to define the eigenvalues of a Hamiltonian. However, there are two main limitations associated with this approach. Firstly, the complexity of the quantum circuits used in VQE can be challenging. Secondly, the classical optimization problem that relies on the variational ansatz [77] introduces additional complexity. The Quantum Variational Circuit (QVC), which is applied in this method, enables a weighted rotation of L (a hyperparameter of the variational circuit) times on the Bloch sphere. As the vectors are already encoded as linear angles in the sphere, this technique provides a detailed description and the ability to search for optimal weights θ . The results are output as a distribution of 0 s and 1 s, mapped to +1 and -1 [78].

4. Quantum Learning Methods

Currently, there is no comprehensive quantum learning theory, and the primary approach is to adapt classical machine learning algorithms or their costly subroutines to run on a quantum computer or within the framework of quantum computing theory. As a result, a hybrid solution has emerged, leaving unanswered questions about what a fully quantum learning procedure will look like. A significant challenge in quantum learning procedures is the presence of noise, which increases as the quantum circuits become more complex and the number of measurements increases. To mitigate this issue, a new method called noise learning has emerged, which has roots in classical machine learning theory. In the context of learning problems, noise plays diverse roles, and sometimes it can produce favorable outcomes. While noise in the inputs and gradients is typically undesirable in classical machine learning, it can be useful in quantum computing. Current quantum computers have few qubits, which limits the implementations of methodologies and applications that will give quantum machine learning the field to be evolved. With more qubits, the capacity of information encoding will increase. Another challenge is error correction; notable solutions have been proposed in recent years in order to solve it [79]. Quantum

computers, without much error, will give the advantage of stabilizing the computational results and increasing the resources like available qubits. Furthermore, a simple quantum error model is used to simulate noisy quantum devices numerically, which involves a weighted combination of two types of errors and phase reversals. Many researchers have investigated how noise affects the ability to learn a function in the quantum setting. Bshouty and Jackson [80] demonstrated that Disjunctive Normal Form (DNF) types could effectively learn under a uniform distribution. Both the classical and quantum problems can be solved quickly without noise, but the existence of noise cannot be ignored.

5. Quantum Machine Learning—Advantages and Limitations

Quantum computers can perform the same tasks as classical computers but with the potential for much faster speeds due to the phenomenon of quantum parallelism. However, this requires the development of new quantum algorithms since classical algorithms may not be sufficient. For example, Shor's algorithm uses quantum parallelism to efficiently factorize large numbers. In order to fully harness the power of quantum computing, it needs to be combined with other technologies such as machine learning, AI, big data, and cloud computing.

The benefits of quantum computing include potential improvements in computational speed, exponential acceleration in certain problems, and the ability to learn from fewer data with complex structures and handle noise. Additionally, quantum computing can increase correlation capacity and achieve results with less training information or simpler models. However, the high cost and sensitivity of the machines are significant disadvantages, as is the fact that they must be operated at extremely low temperatures. The results produced by quantum algorithms can be difficult for humans to comprehend and require experienced users. Furthermore, the possibility of quantum computers easily breaking encryption codes is a concern for internet security.

6. A Preliminary Experimental Study

As described in the sections above, quantum machine learning has achieved significant breakthroughs over its classical counterpart, despite the fact that quantum hardware is still in its early stages of development. This success has raised questions about the concept of quantum supremacy and whether quantum computers hold an advantage over classical computers for machine learning tasks. However, one major limitation is the number of physical qubits available, which makes it challenging to execute quantum machine learning models for large datasets. This creates a paradox where quantum machine learning can outperform classical models in some cases but cannot handle large amounts of data that would demonstrate clear quantum supremacy over classical computers. As a result, a common solution is to simulate quantum machine learning models on classical hardware, where they still demonstrate their superiority over classical models.

Our experiments utilized Qiskit [81] Aqua, a library specifically designed for building and developing quantum algorithms. This versatile framework also provides pre-implemented quantum machine learning algorithms. Our study employed two quantum machine learning algorithms: Support Vector Machine (SVM) and the quantum kernel-based method for SVM-based supervised classification. While the direct kernel-based method for SVM was run on a classical computer that utilized a regular CPU, we employed SVM and Quantum Support Vector Machine (QSVM) learning methods for binary classification problems using three benchmark datasets. It should be noted that quantum machine learning algorithms are only simulated in this study, and therefore the comparison is not perfectly fair; hence, it is a limitation.

To measure the performance of the two models in the classification problems, three known metrics were applied, ROC_AUC, F1-Score, and accuracy. ROC_AUC stands for Compute Area Under the Receiver Operating Characteristic Curve and is a statistical measure for classification problems. In general, it shows the performance of a classifier to distinguish the different classes in a dataset. The two terms that form this performance

curve are True Positive Rate (TPR) or recall and False Positive Rate (FPR) or fall-out where their mathematical formulas are:

$$\text{TPR} = \frac{\text{TP}}{\text{TP} + \text{FN}} \quad (1)$$

$$\text{FPR} = \frac{\text{FP}}{\text{FP} + \text{TN}} \quad (2)$$

In Equation (1), the term TP stands for True Positive and indicates that a model correctly estimates the positive class. The term FN stands for False Negative and indicates that a model incorrectly estimates the negative class. In Equation (2), the term FP stands for False Positive and indicates that a model incorrectly estimated the positive class. The term TN stands for True Negative and indicates that a model correctly estimated the negative class. Based on these two formulas, the ROC_AUC curve is calculated. F1-score is also a performance metric that shows the balance of a model to distinguish the positive classes from the negatives. Its mathematical formula is:

$$\text{F}_1\text{-score} = \frac{2\text{TP}}{2\text{TP} + \text{FP} + \text{FN}} \quad (3)$$

In Equation (3), its terms are the same as TPR and FPR, and the results are the harmonic mean of precision and sensitivity. Similarly to the previous metrics, accuracy is also a performance metric calculated based on the previously mentioned terms. Its mathematical formula is:

$$\text{Accuracy} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{TN} + \text{FP} + \text{FN}} \quad (4)$$

Equation (4) is a fraction between the number of correct predictions and the total number of predictions. With these metrics, the overall performance of a classification model is covered, even if it is quantum or classical, since only the model predictions are considered.

6.1. Datasets Acquisition

In this study, we utilized three datasets from the UCI Machine Learning Repository. The first dataset is the Breast Cancer Wisconsin (Diagnosis) dataset, which contains 699 observations with ten features used to predict the diagnosis value of a breast cancer cell nucleus as either benign (represented by the default diagnosis value of 0) or malignant (represented by a diagnosis value of 1). This dataset is available from scikit learn [82].

The second dataset is the UCI ML Ionosphere dataset, collected by a system in Goose Bay, Labrador, and consists of a phased array of 16 high-frequency antennas with total transmitted power on the order of 6.4 kilowatts. This dataset contains 351 observations and 34 features that are used to predict whether radar returns show evidence of some type of structure in the ionosphere (represented as “good” returns) or not (represented as “bad” returns) [83].

The third dataset contains 4601 observations and 57 features related to the concept of spam, such as advertisements for products/websites, make-money-fast schemes, chain letters, etc. The dataset includes spam and non-spam emails from fieldwork and personal emails [84]. All three datasets are suitable for quantum machine learning algorithms and were used for supervised binary classification problems.

6.2. Experimental Results and Discussion

To provide a more technical review, we conducted an experimental study to compare the performance of QSVM and classical SVM on three datasets, two of which were used for the first time in this study. The experiments were conducted in several phases, including data preprocessing, quantum and classical model training, and prediction using a 10-fold cross-validation approach. Cross-validation is the sampling by no overlap test sets, while

“fold” is the number of resulting subsets by randomly sampling cases from the learning set without replacement. Then, the model is applied to the remaining subset, denoted as the validation set, and the performance is measured. This procedure is repeated until each of the k subsets has served as the validation set. In the 10-fold cross-validation, the first fold serves as the validation set, and the remaining nine folds serve as the training set. In the second fold, the second subset is the validation set, the remaining subsets are the training set, and so on. The cross-validated accuracy, for example, is the average of all ten accuracies achieved on the validation sets [85,86]. Figure 10 shows a diagram of the experimental setup:

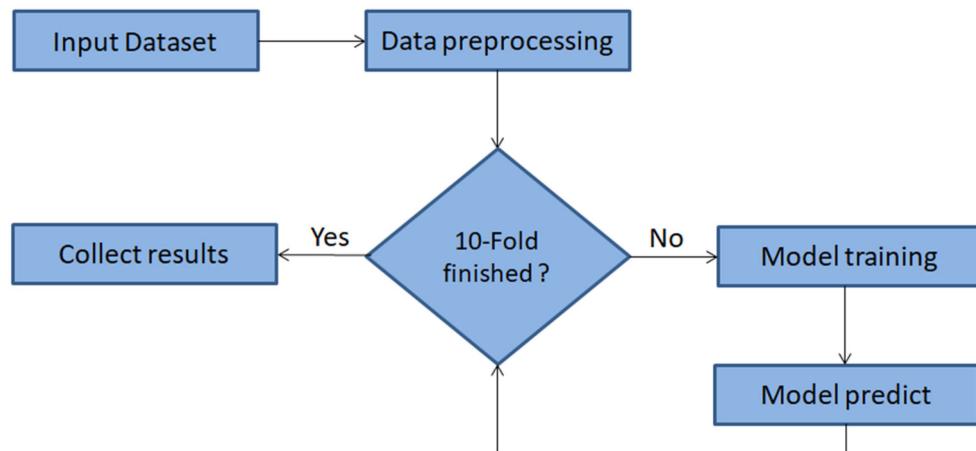


Figure 10. Experimental setup scheme.

Based on Figure 10, the data preprocessing block differs between quantum and classical approaches. On the quantum side, each feature of the dataset is represented by a qubit to create the quantum state of a data point and the values are normalized and scaled to the range $(-1, 1)$ on the Bloch sphere. Then, PCA is applied to reduce the dimensionality of the data features to meet the qubit simulation requirements. Next, a hybrid process occurs, where each fold is created on a classical computer, and then quantum training and prediction are performed. Finally, the measured labels are used to calculate the performance metrics. On the classical side, the same preprocessing steps are followed to ensure a fair comparison. As mentioned above, three datasets are used: Breast Cancer, Ionosphere, and Spam Base. Table 1 below presents some characteristics of these datasets.

Table 1. Dataset characteristics.

Dataset Name	Number of Samples	Number of Features	Number of Classes
Breast cancer	699	10	2
Ionosphere	351	34	2
Spam Base	4601	57	2

The datasets presented above have different characteristics that affect the performance of quantum and classical machine learning models. The Breast Cancer dataset is well-constructed, with a balanced number of features and samples for the two classes. It is a well-known dataset in the quantum machine learning community, and most models, both quantum and classical, can perform well without further processing.

The Ionosphere dataset is a two-class problem with fewer samples and more features than the Breast Cancer dataset. Furthermore, the dataset only contains numerical values, not categorical ones like the Breast Cancer dataset. These factors make Ionosphere a more challenging dataset than Breast Cancer.

The Spam Base dataset is a newly applied dataset in quantum machine learning, and it is the most challenging dataset in this experimental setup due to a large number of samples and features. The selection of this dataset is to demonstrate the superior performance of QSVM over classical SVM with a quantum simulator as the execution environment.

A fundamental difference between QSVM and classical SVM is the type of kernel used for data representation in order to find the hyperplane that separates the data into different classes. In the quantum version, quantum kernels can be constructed using Pauli operations along the X, Y, and Z axes and their combinations, creating higher-dimensional quantum kernels in a multidimensional Hilbert space. The other parameters of QSVM and SVM are the same and kept at their default values in the experimental comparison. It is worth noting that even with dimensionality reduction and simulation on a classical computer, quantum kernels perform equally or better than classical kernels in most cases. However, despite the fact that QSVM performs better than classical SVM in many cases, the supremacy of quantum over classical is not yet clear, especially in the absence of real quantum hardware due to limited access.

It is also worth noting that even with simulation, the execution of quantum circuits is faster than classical operations, with the time-consuming process being the encoding and decoding of classical data into quantum states.

Table 2 illustrates the performance of the QSVM and SVM for the case of the first dataset, whereas Figures 11 and 12 illustrate the same results graphically.

Table 2. QSVM and SVM performance for the Breast cancer dataset (highest performance in bold).

Folds	QSVM			SVM		
	Roc_Auc	Accuracy	F1-Score	Roc_Auc	Accuracy	F1-Score
Fold 1	0.97	0.97	0.97	0.97	0.97	0.97
Fold 2	0.96	0.95	0.96	0.93	0.94	0.95
Fold 3	0.96	0.95	0.96	0.94	0.95	0.96
Fold 4	0.96	0.97	0.97	0.93	0.94	0.95
Fold 5	0.96	0.95	0.96	0.98	0.98	0.98
Fold 6	0.92	0.94	0.95	0.96	0.97	0.97
Fold 7	0.91	0.91	0.93	0.97	0.98	0.98
Fold 8	0.92	0.89	0.91	1.0	1.0	1.0
Fold 9	0.92	0.92	0.94	0.96	0.97	0.97
Fold 10	0.95	0.94	0.95	0.92	0.91	0.92
Mean ± Std	0.94 ± 0.0226	0.94 ± 0.0256	0.95 ± 0.0189	0.96 ± 0.0255	0.96 ± 0.0260	0.97 ± 0.0217

QSVM performance on breast cancer data

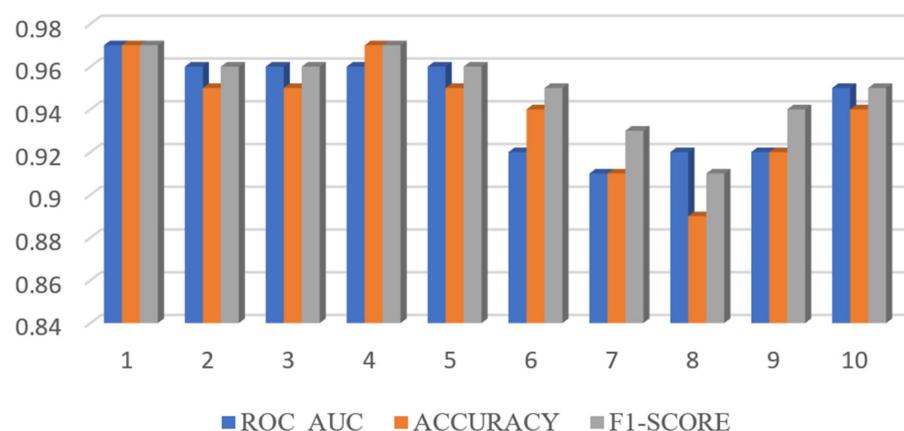


Figure 11. Performance of the Quantum Support Vector Machine (QSVM) on breast cancer data, assessed by three different methods: ROC_AUC, Accuracy, and F1-Score.

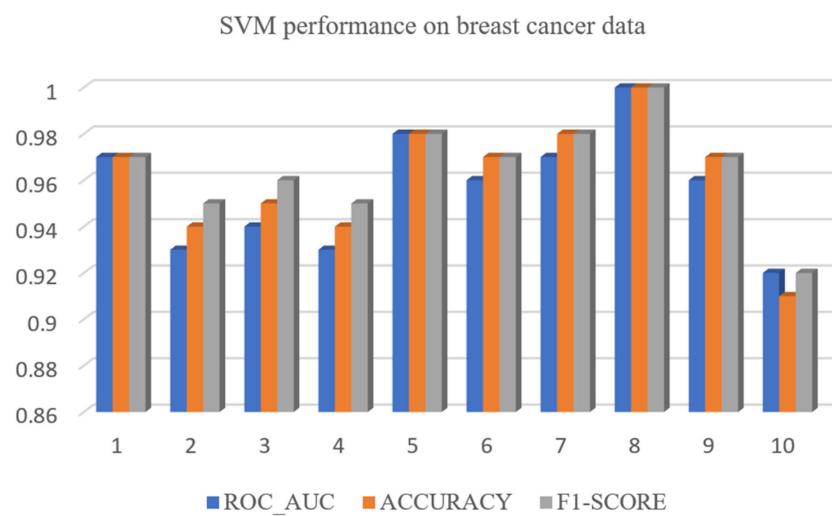


Figure 12. Performance of the Support Vector Machine (SVM) on breast cancer data, assessed by three different methods: ROC_AUC, Accuracy, and F1-Score.

Regarding the first dataset, the quantum advantage is not readily apparent, largely due to how the data is distributed. Both SVM and QSVM can approximate the hyperplane with relative ease. In fact, classical SVM outperforms QSVM with a 2% higher accuracy in both the training and testing samples for each of the folds, which encompass 614 training and 69 testing samples.

Table 3 illustrates the performance of the QSVM and SVM for the case of the second dataset, whereas Figures 13 and 14 illustrate the same results graphically.

Regarding the second dataset, it is noteworthy that the Ionosphere dataset comprises a relatively small number of samples, and in this scenario, the results of QSVM are better than SVM. In 6 out of 10 folds QSVM brings better results than SVM. This implies that QSVM can effectively learn even with a limited amount of data. Furthermore, it should be noted that QSVM has the added ability to differentiate between classes based on numerical values, which can be advantageous in certain applications.

Table 4 illustrates the performance of the QSVM and SVM for the case of the third dataset, whereas Figures 15 and 16 illustrate the same results graphically.

Table 3. QSVM and SVM performance for the Ionosphere dataset (highest performance in bold).

Folds	QSVM			SVM		
	Roc_Auc	Accuracy	F1-Score	Roc_Auc	Accuracy	F1-Score
Fold 1	0.87	0.86	0.94	0.81	0.88	0.89
Fold 2	0.89	0.88	0.95	0.91	0.91	0.95
Fold 3	0.66	0.80	0.86	0.73	0.77	0.71
Fold 4	0.87	0.88	0.90	0.70	0.74	0.73
Fold 5	0.82	0.85	0.84	0.75	0.82	0.79
Fold 6	0.90	0.91	0.86	0.80	0.77	0.65
Fold 7	0.89	0.88	0.94	0.76	0.80	0.83
Fold 8	0.72	0.74	0.81	0.84	0.85	0.82
Fold 9	0.83	0.82	0.88	0.73	0.85	0.86
Fold 10	0.72	0.77	0.80	0.84	0.88	0.88
Mean ± Std	0.82 ± 0.0862	0.84 ± 0.0549	0.88 ± 0.0539	0.78 ± 0.0646	0.83 ± 0.0562	0.81 ± 0.0921

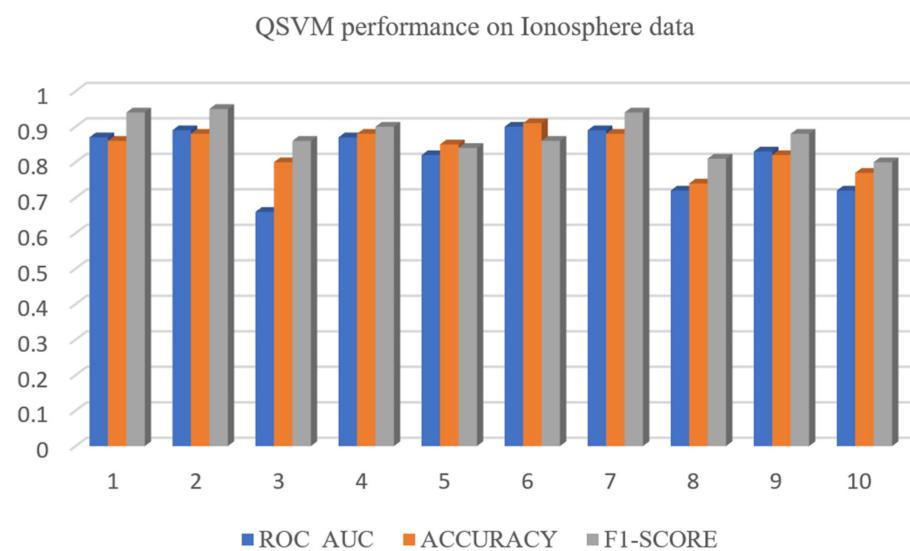


Figure 13. Performance of the Quantum Support Vector Machine (QSVM) on Ionosphere data, assessed by three different methods: ROC_AUC, Accuracy, and F1-Score.

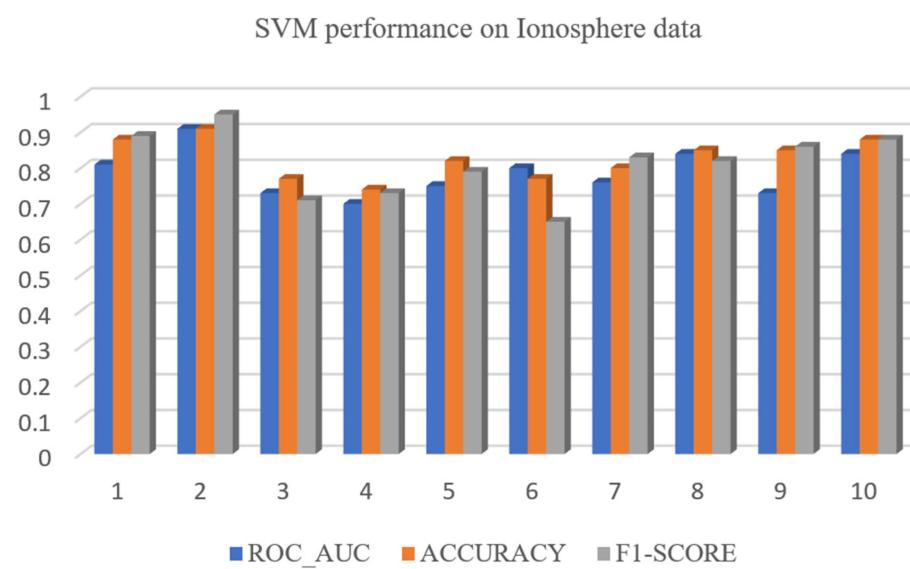


Figure 14. Performance of the Support Vector Machine (SVM) on Ionosphere data, assessed by three different methods: ROC_AUC, Accuracy, and F1-Score.

Table 4. QSVM and SVM performance for the Spam Base dataset (highest performance in bold).

Folds	QSVM			SVM		
	Roc_Auc	Accuracy	F1-Score	Roc_Auc	Accuracy	F1-Score
Fold 1	0.80	0.80	0.79	0.78	0.77	0.71
Fold 2	0.79	0.81	0.81	0.84	0.84	0.86
Fold 3	0.86	0.86	0.86	0.85	0.85	0.91
Fold 4	0.81	0.81	0.84	0.81	0.81	0.83
Fold 5	0.85	0.85	0.84	0.81	0.81	0.84
Fold 6	0.85	0.85	0.89	0.82	0.82	0.84
Fold 7	0.87	0.87	0.84	0.83	0.83	0.82
Fold 8	0.85	0.85	0.83	0.79	0.79	0.74
Fold 9	0.81	0.82	0.83	0.83	0.82	0.90
Fold 10	0.84	0.84	0.87	0.80	0.80	0.81
Mean ± Std	0.83 ± 0.0279	0.83 ± 0.0241	0.83 ± 0.0287	0.81 ± 0.0222	0.81 ± 0.0237	0.82 ± 0.0626

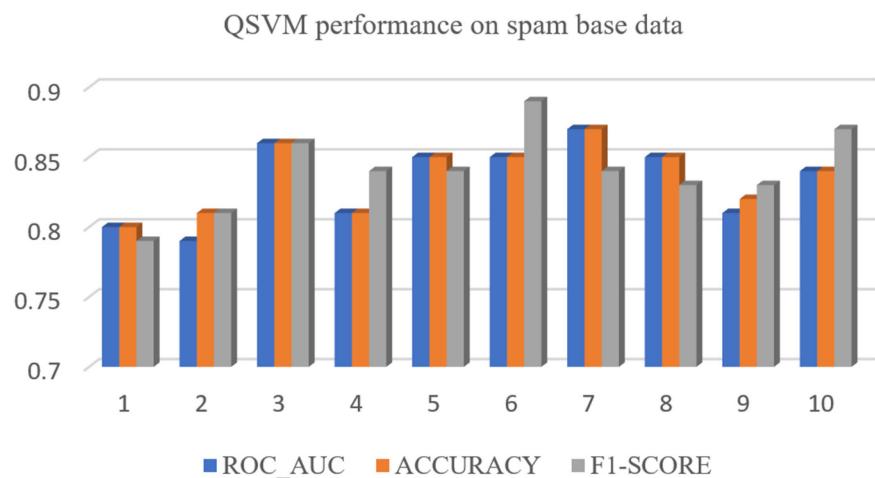


Figure 15. Performance of the Quantum Support Vector Machine (QSVM) on Spam Base data, assessed by three different methods: ROC_AUC, Accuracy, and F1-Score.

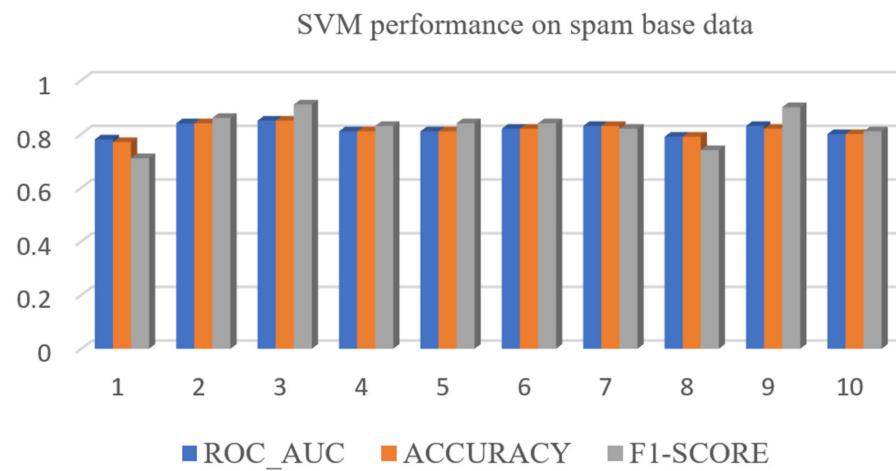


Figure 16. Performance of the Support Vector Machine (SVM) on Spam Base data, assessed by three different methods: ROC_AUC, Accuracy, and F1-Score.

In this dataset, QSVM performs better than SVM in 9 out of 10 folds without a large difference. These findings suggest that QSVM has the potential to effectively handle complex datasets, such as the Spam Base dataset, and therefore can be considered as a promising solution in this domain. Based on the overall results, QSVM performs better in two of three datasets without having a large difference from classical SVM.

Regarding efficiency and speed, QSVM is computationally more demanding and slower than classical SVM. QSVM requires at least 25 GB of memory and over 3 h of training time for each dataset, while classical SVM operates with less than 1 GB of memory and completes training within 1 h for each dataset.

From an efficiency perspective, QSVM falls behind SVM, especially when executed on a classical machine where extensive computational resources are required and not all quantum computing characteristics can be fully harnessed. However, from a performance standpoint, QSVM demonstrates potential, as it outperforms SVM in two of three datasets. It is worth noting that in this study, the QSVM algorithm is compared with its classical counterpart, although other quantum-inspired classical SVM approaches that have been proposed recently [87] can also be considered. In the latter cases, improved accuracy in an accelerated way seems feasible without implementing the SVM algorithm in a fully quantum form. However, when quantum computers are established as alternative computing devices, any classical ML model is expected to be outperformed by their QML counterparts.

In conclusion, quantum models exhibit promising potential for achieving superior performance compared to classical models, but there are numerous challenges and limitations that need to be addressed and resolved.

7. Conclusions and Future Works

Quantum computing is still in its early stages, and building a functional and efficient computer with enough qubits takes years. While quantum machine learning (QML) methodologies produce spectacular results, the quantum material currently available is not sufficient. Researchers need access to quantum computers with more physical resources to expand the scope and power of QML. As quantum hardware and computing continue to evolve, QML is likely to become a leading approach in applications such as unsupervised learning and generative models, which outperform classical versions.

Many classical machine learning (ML) methods and techniques can be transformed into QML schemes to increase the domain's capabilities, such as expert systems. In this paper, we experimentally compared three datasets using classical and quantum machine learning methods. We used three datasets (Breast Cancer, Ionosphere, and Spam Base) and implemented SVM and QSVM algorithms to predict accuracy. The Spam Base dataset was used for the first time and was the most challenging of the experiments, where QSVM outperformed classical SVM. It is important to note that even though the quantum execution environment was simulated by a classical computer, QSVM achieved higher scores in Spam Base dataset, where each fold contained 1527 training samples and 712 testing samples.

Another important observation is that the higher the complexity of the dataset, the wider the performance gap between quantum and classical models. This is because quantum machine learning operations can generalize much better without losing performance, in contrast to classical models, where complex datasets and simple models often overfit. In the future, there are some open issues that need to be addressed. For instance, the quantum simulator was executed on a local classical machine without high computing power, while execution on a real quantum computer could provide more realistic outcomes. The QML domain should also target designing new quantum learning models that will observe patterns under quantum mechanics schemes, not classical statistical theory. This will provide an opportunity to explore new model architectures that might overcome classical machine learning limitations.

Author Contributions: Conceptualization, K.A.T., T.K. and G.A.P.; methodology, K.A.T. and T.K.; software, K.A.T. and T.K.; validation, K.A.T. and T.K.; formal analysis, K.A.T. and T.K.; investigation, K.A.T. and T.K.; resources, K.A.T.; data curation, K.A.T.; writing—original draft preparation, K.A.T. and T.K.; writing—review and editing, G.A.P.; visualization, K.A.T. and T.K.; supervision, G.A.P.; project administration, G.A.P.; funding acquisition, G.A.P. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: Not applicable.

Acknowledgments: This work was supported by the MPhil program "Advanced Technologies in Informatics and Computers", hosted by the Department of Computer Science, International Hellenic University, Kavala, Greece.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Hu, W. Comparison of Two Quantum Nearest Neighbor Classifiers on IBM's Quantum Simulator. *Nat. Sci.* **2018**, *10*, 87–98. [[CrossRef](#)]
2. Biamonte, J.; Wittek, P.; Pancotti, N.; Rebentrost, P.; Wiebe, N.; Lloyd, S. Quantum Machine Learning. *Nature* **2017**, *549*, 195–202. [[CrossRef](#)] [[PubMed](#)]
3. Samuel, A.L. Some Studies in Machine Learning Using the Game of Checkers. *IBM J. Res. Dev.* **1959**, *3*, 210–229. [[CrossRef](#)]
4. Rebentrost, P.; Mohseni, M.; Lloyd, S. Quantum Support Vector Machine for Big Data Classification. *Phys. Rev. Lett.* **2014**, *113*, 130503. [[CrossRef](#)] [[PubMed](#)]

5. Abdi, H.; Williams, L.J. Principal Component Analysis: Principal Component Analysis. *WIREs Comp. Stat.* **2010**, *2*, 433–459. [[CrossRef](#)]
6. Rosenblatt, F. The Perceptron: A Probabilistic Model for Information Storage and Organization in the Brain. *Psychol. Rev.* **1958**, *65*, 386–408. [[CrossRef](#)]
7. LeCun, Y.; Bengio, Y.; Hinton, G. Deep Learning. *Nature* **2015**, *521*, 436–444. [[CrossRef](#)]
8. Mozaffari-Kermani, M.; Sur-Kolay, S.; Raghunathan, A.; Jha, N.K. Systematic Poisoning Attacks on and Defenses for Machine Learning in Healthcare. *IEEE J. Biomed. Health Inform.* **2015**, *19*, 1893–1905. [[CrossRef](#)]
9. Qayyum, A.; Qadir, J.; Bilal, M.; Al-Fuqaha, A. Secure and Robust Machine Learning for Healthcare: A Survey. *IEEE Rev. Biomed. Eng.* **2021**, *14*, 156–180. [[CrossRef](#)]
10. Wang, B.; Yao, Y.; Shan, S.; Li, H.; Viswanath, B.; Zheng, H.; Zhao, B.Y. Neural Cleanse: Identifying and Mitigating Backdoor Attacks in Neural Networks. In Proceedings of the IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 19–23 May 2019; pp. 707–723.
11. Shor, P.W. Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer. *SIAM Rev.* **1999**, *41*, 303–332. [[CrossRef](#)]
12. Abohashima, Z.; Elhosien, M.; Houssein, E.H.; Mohamed, W.M. Classification with Quantum Machine Learning: A Survey. *arXiv* **2020**, arXiv:2006.12270.
13. Farouk, A.; Tarawneh, O.; Elhoseny, M.; Batle, J.; Naseri, M.; Hassanien, A.E.; Abedl-Aty, M. Quantum Computing and Cryptography: An Overview. In *Quantum Computing: An Environment for Intelligent Large Scale Real Application*; Hassanien, A.E., Elhoseny, M., Kacprzyk, J., Eds.; Studies in Big Data; Springer International Publishing: Cham, Switzerland, 2018; Volume 33, pp. 63–100.
14. Duan, B.; Yuan, J.; Yu, C.-H.; Huang, J.; Hsieh, C.-Y. A Survey on HHL Algorithm: From Theory to Application in Quantum Machine Learning. *Phys. Lett. A* **2020**, *384*, 126595. [[CrossRef](#)]
15. Feynman, R.P. *Feynman Lectures on Computation*, 1st ed.; Hey, T., Allen, R.W., Eds.; CRC Press: Boca Raton, FL, USA, 2018.
16. da Silva, A.J.; Ludermir, T.B.; de Oliveira, W.R. Quantum Perceptron over a Field and Neural Network Architecture Selection in a Quantum Computer. *Neural Netw.* **2016**, *76*, 55–64. [[CrossRef](#)] [[PubMed](#)]
17. Li, J.; Lin, S.; Yu, K.; Guo, G. Quantum K-Nearest Neighbor Classification Algorithm Based on Hamming Distance. *Quantum Inf. Process.* **2022**, *21*, 18. [[CrossRef](#)]
18. Lu, S.; Braunstein, S.L. Quantum Decision Tree Classifier. *Quantum Inf. Process.* **2014**, *13*, 757–770. [[CrossRef](#)]
19. Adhikary, S.; Dangwal, S.; Bhowmik, D. Supervised Learning with a Quantum Classifier Using Multi-Level Systems. *Quantum Inf. Process.* **2020**, *19*, 89. [[CrossRef](#)]
20. Chakraborty, S.; Shaikh, S.H.; Chakrabarti, A.; Ghosh, R. A Hybrid Quantum Feature Selection Algorithm Using a Quantum Inspired Graph Theoretic Approach. *Appl. Intell.* **2020**, *50*, 1775–1793. [[CrossRef](#)]
21. Neven, H.; Denchev, V.S.; Rose, G.; Macready, W.G. Training a Large Scale Classifier with the Quantum Adiabatic Algorithm. *arXiv* **2009**, arXiv:0912.0779.
22. Pudenz, K.L.; Lidar, D.A. Quantum Adiabatic Machine Learning. *Quantum Inf. Process.* **2013**, *12*, 2027–2070. [[CrossRef](#)]
23. Neigozzen, R.; Neves, J.L.; Sollacher, R.; Glaser, S.J. Quantum Pattern Recognition with Liquid-State Nuclear Magnetic Resonance. *Phys. Rev. A* **2009**, *79*, 042321. [[CrossRef](#)]
24. Bennett, C.H.; Brassard, G.; Breidbart, S.; Wiesner, S. Quantum Cryptography, or Unforgeable Subway Tokens. In *Advances in Cryptology*; Chaum, D., Rivest, R.L., Sherman, A.T., Eds.; Springer: Boston, MA, USA, 1983; pp. 267–275.
25. Bennett, C.H.; Bessette, F.; Brassard, G.; Salvail, L.; Smolin, J. Experimental Quantum Cryptography. *J. Cryptol.* **1992**, *5*, 3–28. [[CrossRef](#)]
26. Tian, J.; Wu, B.; Wang, Z. High-Speed FPGA Implementation of SIKE Based on an Ultra-Low-Latency Modular Multiplier. *IEEE Trans. Circuits Syst. I Regul. Pap.* **2021**, *68*, 3719–3731. [[CrossRef](#)]
27. Mozaffari-Kermani, M.; Azarderakhsh, R. Reliable hash trees for post-quantum stateless cryptographic hash-based signatures. In Proceedings of the IEEE International Symposium on Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFTS), Amherst, MA, USA, 12–14 October 2015; pp. 103–108.
28. Shor, P.W. Algorithms for Quantum Computation: Discrete Logarithms and Factoring. In Proceedings of the 35th Annual Symposium on Foundations of Computer Science, Santa Fe, NM, USA, 20–22 November 1994; pp. 124–134.
29. Ciliberto, C.; Herbster, M.; Ialongo, A.D.; Pontil, M.; Rocchetto, A.; Severini, S.; Wossnig, L. Quantum Machine Learning: A Classical Perspective. *Proc. R. Soc. A* **2008**, *474*, 20170551. [[CrossRef](#)] [[PubMed](#)]
30. Dunjko, V.; Briegel, H.J. Machine Learning & Artificial Intelligence in the Quantum Domain: A Review of Recent Progress. *Rep. Prog. Phys.* **2018**, *81*, 074001. [[PubMed](#)]
31. Schuld, M.; Sinayskiy, I.; Petruccione, F. An Introduction to Quantum Machine Learning. *Contemp. Phys.* **2015**, *56*, 172–185. [[CrossRef](#)]
32. Havenstein, C.; Thomas, D.; Chandrasekaran, S. Comparisons of performance between quantum and classical machine learning. *SMU Data Sci. Rev.* **2018**, *1*, 11.
33. Singh, G.; Kaur, M.; Singh, M.; Kumar, Y. Implementation of Quantum Support Vector Machine Algorithm Using a Benchmarking Dataset. *IJPAP* **2022**, *60*, 407–414.
34. Saini, S.; Khosla, P.; Kaur, M.; Singh, G. Quantum Driven Machine Learning. *Int. J. Theor. Phys.* **2020**, *59*, 4013–4024. [[CrossRef](#)]

35. Havlíček, V.; Córcoles, A.D.; Temme, K.; Harrow, A.W.; Kandala, A.; Chow, J.M.; Gambetta, J.M. Supervised Learning with Quantum-Enhanced Feature Spaces. *Nature* **2019**, *567*, 209–212. [[CrossRef](#)]
36. Tang, E. A Quantum-Inspired Classical Algorithm for Recommendation Systems. In Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, ACM, Phoenix, AZ, USA, 23–26 June 2019; pp. 217–228.
37. Shan, Z.; Guo, J.; Ding, X.; Zhou, X.; Wang, J.; Lian, H.; Gao, Y.; Zhao, B.; Xu, J. Demonstration of Breast Cancer Detection Using QSVM on IBM Quantum Processors. *Res. Sq.* **2022**; preprint, in review.
38. Kumar, T.; Kumar, D.; Singh, G. Performance Analysis of Quantum Classifier on Benchmarking Datasets. *IJEER* **2022**, *10*, 375–380. [[CrossRef](#)]
39. Kavitha, S.S.; Kaulgud, N. Quantum Machine Learning for Support Vector Machine Classification. *Evol. Intel.* **2022**, *1*–10. [[CrossRef](#)]
40. Batra, K.; Zorn, K.M.; Foil, D.H.; Minerali, E.; Gawriljuk, V.O.; Lane, T.R.; Ekins, S. Quantum Machine Learning Algorithms for Drug Discovery Applications. *J. Chem. Inf. Model* **2021**, *61*, 2641–2647. [[CrossRef](#)] [[PubMed](#)]
41. Suzuki, Y.; Yano, H.; Gao, Q.; Uno, S.; Tanaka, T.; Akiyama, M.; Yamamoto, N. Analysis and Synthesis of Feature Map for Kernel-Based Quantum Classifier. *Quantum Mach. Intell.* **2020**, *2*, 9. [[CrossRef](#)]
42. Bai, Y.; Han, X.; Chen, T.; Yu, H. Quadratic Kernel-Free Least Squares Support Vector Machine for Target Diseases Classification. *J. Comb. Optim.* **2015**, *30*, 850–870. [[CrossRef](#)]
43. Feynman, R.P.; Kleinert, H. Effective Classical Partition Functions. *Phys. Rev. A* **1986**, *34*, 5080–5084. [[CrossRef](#)]
44. Deutsch, D. Quantum theory, the Church–Turing principle and the universal quantum computer. *Proc. R. Soc. London A Math. Phys. Sci.* **1985**, *400*, 97–117.
45. Grover, L.K. A Fast Quantum Mechanical Algorithm for Database Search. In Proceedings of the 28th Annual ACM Symposium on Theory of Computing (STOC ‘96), Philadelphia, PA, USA, 22–24 May 1996; ACM: New York, NY, USA, 1996; pp. 212–219.
46. Jones, J.A.; Mosca, M.; Hansen, R.H. Implementation of a quantum search algorithm on a quantum computer. *Nature* **1998**, *393*, 344–346. [[CrossRef](#)]
47. IBM. DOcplex Examples. Available online: <https://prod.ibmdocs-production-dal-6099123ce774e592a519d7c33db8265e-0000.us-south.containers.appdomain.cloud/docs/en/icos/12.9.0?topic=api-docplex-examples> (accessed on 16 April 2023).
48. Preskill, J. Quantum Comput. Entanglement Frontier. *arXiv* **2012**, arXiv:1203.5813.
49. Gibney, E. Hello quantum world! Google publishes landmark quantum supremacy claim. *Nature* **2019**, *574*, 461–463. [[CrossRef](#)]
50. Milburn, G.J.; Leggett, A.J. The Feynman Processor: Quantum Entanglement and the Computing Revolution. *Phys. Today* **1999**, *52*, 51–52. [[CrossRef](#)]
51. Nielsen, M.A.; Chuang, I.L. *Quantum Computation and Quantum Information*, 10th anniversary ed.; Cambridge University Press: Cambridge, UK; New York, NY, USA, 2010.
52. Lewin, D.I. Searching for the elusive qubit. *Comput. Sci. Eng.* **2001**, *3*, 4–7. [[CrossRef](#)]
53. Barenco, A.; Bennett, C.H.; Cleve, R.; DiVincenzo, D.P.; Margolus, N.; Shor, P.; Sleator, T.; Smolin, J.A.; Weinfurter, H. Elementary Gates for Quantum Computation. *Phys. Rev. A* **1995**, *52*, 3457–3467. [[CrossRef](#)] [[PubMed](#)]
54. Fowler, A.G.; Devitt, S.J.; Hollenberg, L.C.L. Implementation of Shor’s Algorithm on a Linear Nearest Neighbour Qubit Array. *arXiv* **2004**, arXiv:quant-ph/0402196. [[CrossRef](#)]
55. Liang, L.; Li, C. Realization of Quantum SWAP Gate between Flying and Stationary Qubits. *Phys. Rev. A* **2005**, *72*, 024303. [[CrossRef](#)]
56. Grassl, M.; Roetteler, M.; Beth, T. Efficient Quantum Circuits for Non-Qubit Quantum Error-Correcting Codes. *Int. J. Found. Comput. Sci.* **2002**, *14*, 757–775. [[CrossRef](#)]
57. Mishra, N.; Kapil, M.; Rakesh, H.; Anand, A.; Mishra, N.; Warke, A.; Sarkar, S.; Dutta, S.; Gupta, S.; Prasad Dash, A.; et al. Quantum Machine Learning: A Review and Current Status. In *Data Management, Analytics and Innovation. Advances in Intelligent Systems and Computing*; Springer: Singapore, 2019; pp. 101–145.
58. Sharma, N.; Chakrabarti, A.; Balas, V.E. Data Management, Analytics and Innovation. In Proceedings of the Third International Conference on Data Management, Analytics and Innovation—ICDMAI, Kuala Lumpur, Malaysia, 18–20 January 2019; Volume 2.
59. McClean, J.R.; Romero, J.; Babbush, R.; Aspuru-Guzik, A. The Theory of Variational Hybrid Quantum-Classical Algorithms. *New J. Phys.* **2016**, *18*, 023023. [[CrossRef](#)]
60. Abbas, A.; Sutter, D.; Zoufal, C.; Lucchi, A.; Figalli, A.; Woerner, S. The Power of Quantum Neural Networks. *Nat. Comput. Sci.* **2021**, *1*, 403–409. [[CrossRef](#)]
61. Cong, I.; Choi, S.; Lukin, M.D. Quantum Convolutional Neural Networks. *Nat. Phys.* **2018**, *15*, 1273–1278. [[CrossRef](#)]
62. Bausch, J. Recurrent Quantum Neural Networks. *arXiv* **2020**, arXiv:2006.14619.
63. Zhao, R.; Wang, S. A Review of Quantum Neural Networks: Methods, Models, Dilemma. *arXiv* **2021**, arXiv:2109.01840.
64. Lloyd, S.; Mohseni, M.; Rebentrost, P. Quantum Algorithms for Supervised and Unsupervised Machine Learning. *arXiv* **2013**, arXiv:1307.0411.
65. Adcock, J.; Allen, E.; Day, M.; Frick, S.; Hinchliff, J.; Johnson, M.; Morley-Short, S.; Pallister, S.; Price, A.; Stanisic, S. Advances in quantum machine learning. *arXiv* **2015**, arXiv:1512.02900.
66. Ackley, D.H.; Hinton, G.E.; Sejnowski, T.J. A Learning Algorithm for Boltzmann Machines. *Cogn. Sci.* **1985**, *9*, 147–169. [[CrossRef](#)]
67. Wiebe, N.; Kapoor, A.; Granade, C.; Svore, K.M. Quantum Inspired Training for Boltzmann Machines. *arXiv* **2015**, arXiv:1507.02642.

68. Schuld, M.; Sinayskiy, I.; Petruccione, F. The Quest for a Quantum Neural Network. *Quantum Inf. Process.* **2014**, *13*, 2567–2586. [[CrossRef](#)]
69. Schuld, M.; Killoran, N. Quantum Machine Learning in Feature Hilbert Spaces. *Phys. Rev. Lett.* **2019**, *122*, 040504. [[CrossRef](#)]
70. Huang, Y.; Ni, L.; Miao, Y. A Quantum Cognitive Map Model. In Proceedings of the 2009 Fifth International Conference on Natural Computation, Tianjin, China, 14–16 August 2009; pp. 28–31.
71. Nan, M.; Hamido, F.; Yun, Z.; Shupeng, W. Ensembles of Fuzzy Cognitive Map Classifiers Based on Quantum Computation. *APH* **2015**, *12*, 7–26.
72. Zhang, W.-R. Information Conservational YinYang Bipolar Quantum-Fuzzy Cognitive Maps—Mapping Business Data to Business Intelligence. In Proceedings of the 2016 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), Vancouver, BC, Canada, 24–29 July 2016; pp. 2279–2286.
73. Zhang, W.-R. From Equilibrium-Based Business Intelligence to Information Conservational Quantum-Fuzzy Cryptography—A Cellular Transformation of Bipolar Fuzzy Sets to Quantum Intelligence Machinery. *IEEE Trans. Fuzzy Syst.* **2018**, *26*, 656–669. [[CrossRef](#)]
74. Bologna, G.; Hayashi, Y. QSVM: A Support Vector Machine for Rule Extraction. In *Advances in Computational Intelligence*; Rojas, I., Joya, G., Catala, A., Eds.; Lecture Notes in Computer Science; Springer International Publishing: Cham, Switzerland, 2015; Volume 9095, pp. 276–289.
75. Park, J.-E.; Quanz, B.; Wood, S.; Higgins, H.; Harishankar, R. Practical Application Improvement to Quantum SVM: Theory to Practice. *arXiv* **2020**, arXiv:2012.07725.
76. Vashisth, S.; Dhall, I.; Aggarwal, G. Design and Analysis of Quantum Powered Support Vector Machines for Malignant Breast Cancer Diagnosis. *J. Intell. Syst.* **2021**, *30*, 998–1013. [[CrossRef](#)]
77. Fedorov, D.A.; Peng, B.; Govind, N.; Alexeev, Y. VQE Method: A Short Survey and Recent Developments. *Mater. Theory* **2022**, *6*, 2. [[CrossRef](#)]
78. Cerezo, M.; Arrasmith, A.; Babbush, R.; Benjamin, S.C.; Endo, S.; Fujii, K.; McClean, J.R.; Mitarai, K.; Yuan, X.; Cincio, L.; et al. Variational Quantum Algorithms. *Nat. Rev. Phys.* **2021**, *3*, 625–644. [[CrossRef](#)]
79. Acharya, R.; Aleiner, I.; Allen, R.; Andersen, T.I.; Ansmann, M.; Arute, F.; Arya, K.; Asfaw, A.; Atalaya, J.; Babbush, R.; et al. Suppressing quantum errors by scaling a surface code logical qubit. *arXiv* **2022**, arXiv:2207.06431.
80. Bshouty, N.H.; Jackson, J.C. Learning DNF over the Uniform Distribution Using a Quantum Example Oracle. *SIAM J. Comput.* **1998**, *28*, 1136–1153. [[CrossRef](#)]
81. Qiskit Aqua Algorithms Documentation. Available online: https://qiskit.org/documentation/stable/0.19/apidoc/qiskit_aqua.html (accessed on 12 April 2023).
82. Nick Street. Scikit Learn Dataset Load. Available online: http://scikit-learn.org/stable/modules/generated/sklearn.datasets.load_breast_cancer.html (accessed on 12 April 2023).
83. Available online: <https://archive.ics.uci.edu/ml/datasets/ionosphere> (accessed on 12 April 2023).
84. Available online: <https://archive.ics.uci.edu/ml/datasets/spambase> (accessed on 12 April 2023).
85. Simon, R. Resampling strategies for model assessment and selection. In *Fundamentals of Data Mining in Genomics and Proteomics*; Dubitzky, W., Granzow, M., Berrar, B., Eds.; Springer: Berlin/Heidelberg, Germany, 2007; pp. 173–186.
86. Berrar, D. Cross-Validation. In *Encyclopedia of Bioinformatics and Computational Biology*; Ranganathan, S., Gribskov, M., Nakai, K., Schönbach, C., Eds.; Academic Press: Oxford, UK, 2019; pp. 542–545.
87. Ding, C.; Bao, T.-Y.; Huang, H.-L. Quantum-Inspired Support Vector Machine. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *33*, 7210–7222. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.