

Received December 1, 2021, accepted December 20, 2021, date of publication December 30, 2021, date of current version January 12, 2022.

Digital Object Identifier 10.1109/ACCESS.2021.3139323

# Variational Quantum Classifier for Binary Classification: Real vs Synthetic Dataset

DANYAL MAHESHWARI<sup>1</sup>, DANIEL SIERRA-SOSA<sup>2</sup>,  
AND BEGONYA GARCIA-ZAPIRAIN<sup>1</sup>, (Member, IEEE)

<sup>1</sup>eVIDA Research Group, University of Deusto, 48007 Bilbao, Spain

<sup>2</sup>Computer Science Department, Hood College, Frederick, MD 21701, USA

Corresponding author: Danyal Maheshwari (danyal.maheshwari@deusto.es)

This work was supported by the eVIDA Research Group, University of Deusto, Bilbao, Spain, through the Basque Government's under Grant IT 905-16.

**ABSTRACT** Nowadays, quantum-enhanced methods have been widely studied to solve machine learning related problems. This article presents the application of a Variational Quantum Classifier (VQC) for binary classification. We utilized three datasets: a synthetic dataset with randomly generated values between 0 and 1, the publicly available University of California Intelligence Machine learning (UCI) sonar dataset consisting of mining data, and a proprietary diabetes dataset related to diabetes with acute diseases and diabetes without acute disease. To deal with the limitation of noisy intermediate-scale quantum systems (NISQ), we used a pre-processing method to enhance the prediction rate when applying the VQC method. The process includes feature selection and state preparation. Quantum state preparation is critical for obtaining a functioning pipeline in a quantum machine learning (QML) model. Amplitude encoding is a state preparation approach that enhances the performance of data encoding and the learning of quantum models. As a result, our proposed methods achieved accuracies of 75%, 71.4%, and 68.73% by using VQC model and in contrast, the amplitude encoding-based VQC achieved 98.40%, 67.3%, and 74.50% accuracies on the synthetic, sonar, and diabetes dataset, respectively.

**INDEX TERMS** Quantum machine learning, state preparation, amplitude encoding, variational quantum classifier and T2DM diabetes.

## I. INTRODUCTION

Machine Learning (ML) is predominantly in the artificial intelligence domain, such as computer vision, image recognition, natural language processing, healthcare, and many other applications [1], [2]. Quantum Computing (QC) related research has expanded rapidly in recent years. Despite the fact that, at the present QC is limited by Noise Intermediate-Scale Quantum (NISQ) devices, it could potentially surpass the performance of a classical computer in certain ML applications [3].

Quantum Machine Learning (QML) [4] is an emerging interdisciplinary research field that combines quantum physics and ML [5]. The use of QML improves the performance and speeds up the data processing on QC [6], [7]. However, the limits of the learning capacity of a modern-day machine are solely determined by polynomial computing

time [8]. Thus, it is essential to reduce the complexity of quantum algorithms to obtain a reliable results. There are four possible approaches in QML based on data and the correspondent processing device, whether classical or quantum [9].

ML methods, such as Support Vector Machines (SVM) [10] have been widely explored as a supervised learning task on various datasets and have demonstrated better performance with the kernel-based approaches [11]. The variational quantum classifier (VQC) is widely used for classification problems in the NISQ device [12]. There are several ways to classify well-known supervised QML algorithms, such as QSVM and VQC. Numerous efforts have been taken in this domain based on quantum-inspired neural networks [13] and have been related to the applications, such as hybridized low-depth VQC classification methods with simple error-mitigation [14] and pre-processing methodologies such as Principle Component Analysis (PCA) [15], resulting in an improvement in performance for categorization.

The associate editor coordinating the review of this manuscript and approving it for publication was Joanna Kołodziej<sup>1</sup>.

In QML, state preparation is critical for pre-processing and encoding the classical data into a Quantum state. This process reduces experimental overhead in terms of resources and assists in avoiding of non-linearities in data [5], [16], allowing linear classifiers and kernel-based techniques to perform better in terms of predictions [17], [18], as well as the use of near-term Quantum processors with exponential speedup in such methods like QSVM [19]. One literature has presented the benefits of utilizing quantum algorithms in the ML approach, for instance, being a quadratically reduced query complexity in closest neighbour classification when compared to traditional algorithms [20]. Data encoding is a key component of state preparation, whereas the amplitude encoding technique converts classical data into the amplitude of the essential QC component, better known as a qubit. The qubit is the quantum equivalent of the binary bit and is denoted by  $|\psi\rangle$ . Amplitude encoding as an emerging data encoding technique, which encodes and decodes data without loss.

Quantum Computing methods have been applied on well-known and real datasets. For instance, *D.Sierra-Sosa et al.* [21] employed the state preparation approach on synthetic data for different QML algorithms such as Quantum Conventional Neural Network (QCNN), Hybrid angle encoding, and hybrid amplitude encoding to solve the binary classification problem by utilizing the TensorFlow Quantum system. They compared hybrid angle encoding to hybrid amplitude encoding. The best accuracy of which is 97.4% has been achieved by the utilization of the hybrid amplitude encoding model. *S. Chakraborty et al.* [22] presented a hybrid quantum feature selection algorithm (HQFSA) model that uses quantum parallel amplitude estimation and amplitude amplification to solve the binary classification problem for the UCI sonar dataset and compute the computing power, and they achieved overall 74% accuracy on HQFSA model. *D.Sierra-Sosa et al.* [23] have developed a pre-processing pipeline technique that uses feature scaling, feature selection, ellipsoidal coordinate map, and stoke parameters to use VQC to determine whether diabetes is associated with acute illnesses. In that article, they examined the normalized and zero standard deviation, ellipsoidal transform, and Poincare sphere in the domain of VQC by using the two features and three features diabetes mellitus dataset. They obtained 72% accuracy using the Poincare sphere. *H.Gupta et al.* [24] used the Exploratory data analysis (EDA) and pre-processing approach which was utilized for data scaling, they applied it to the VQC, root mean square propagation (RMSprop) and Deep Learning (DL) models for binary classification with the PIMA diabetes dataset. In that research, they evaluated RMSprop by utilizing back-propagation and VQC model. In terms of over accuracy, the DL model outperformed the VQC model. *D.Maheshwari et al.* [25] employed the classical and quantum algorithms. To use ensemble methods to build a voting model for the prediction of diabetes with acute illnesses and compute the computational time using

DWave System's QPU [26]. In that study, they contrasted the traditional voting model to the hybrid New Model voting method. They obtained approximately the same accuracy on both models, but the hybrid new model was 55 times faster than the classical voting model. Thus, there is a great need to analyze real-world applications of QML, which motivates future studies in this field to utilize quantum characteristics in real-world applications. However, while exploring the potential of QML algorithms, it must be considered that these algorithms may not provide a competitive edge over the classical equivalents. Moreover, knowing the research gaps, this consideration is essential in the development of contemporary quantum technology.

Considering the potential of QML, this article presents binary classification based on the synthetic, sonar, and diabetes datasets. The diabetes dataset is the primary focus of this study. Diabetes is the sixth most prominent cause of death worldwide, but Type 2 diabetes mellitus is a severe public health concern with a significantly growing problem for the world [23], [25], [27]. Around 463 million people have diabetes, 232 million have undiagnosed diabetes, and 4.2 million people died due to the symptoms that stemmed from diabetes in 2019 [25], [28]. There have been various studies conducted to effectively, detect and identify the exact type of diabetes.

This paper aims to assess and compare the performance of two QML methods analyzed using three datasets: a synthetic dataset, a publicly available dataset, and a private dataset. 1) We present a pre-processing approach for mapping data into quantum states, in order to conduct quantum classification. Specifically, this method is focuses on improving data encoding methods outlined by [12], using the IBM Qiskit framework [29]. 3) The amplitude encoding technique assisting to enhance the performance of the VQC model. 4) Compared to the work mentioned above, we perform several experiments using the same features and parameters with VQC, including basis encoding VQC and Amplitude encoding based VQC.

This paper is split into four sections. Section I covers the basics of QC, QML, and VQC. Section II reports Materials and methodology, Pre-processing and state preparation, Section III has briefly described QSVM and VQC, while Section IV discusses the reports and findings. Finally, the conclusion is provided in the final section.

## II. RELATED WORK

We present the materials in this subsection, including datasets and the emphasised traditional ML pre-processing techniques.

### A. MATERIALS

This research is based on three datasets: a synthetic dataset, a publicly available dataset, and a private dataset, namely the Synthetic dataset, the (UCI) Sonar dataset, and the Diabetes dataset. This research aims to effectively perform binary classification using VQC models based on these datasets mentioned in Table 1.

TABLE 1. Datasets attributions.

Datasets	Number of Classes	Number of Features	Train Size	Test Size	Total Size
Synthetic	2	25	400	100	500
Sonar	2	60	166	42	208
Diabetes	2	105	2121	531	2651

The first benchmarking dataset used in this research, a synthetic dataset, is employed to evaluate the performance of the models. The eVIDA research group created the synthetic dataset. The primary motivation for creating a dataset was to have high properties of separable data points, following the probability density function of a Gauss distribution with a variance of 1 and a different mean for every feature. A balanced dataset with 500 samples and 25 attributes, 250 samples of 0 and 250 samples of 1 are used to determine the binary classification.

The second dataset used the publicly available UCI Sonar dataset, introduced by P.R Gorman *et al.* [30] for the binary classification task. The UCI Sonar data was obtained from 208 different rock and metal cylinder shapes and the bouncing sonar frequency at different angles, which are under various conditions. It is a balanced dataset containing 208 samples and has 60 attributes. It carries 111 metal cylinder samples and 97 rock samples used to determine rock and metal.

We have made use of Osakidetza electronic health records (EHRs) of the Diabetes dataset as a significant focus of this study. In addition, the PREST Database (DB) is used to get information on analytical and clinical data for this study.

In the diabetes dataset, classes are separated using the International Classification of Diseases (ICD-9-CM), and the drug encoding scheme is the anatomical, therapeutic, and chemical classification system. The PREST (the Stratification Program's Database) was created in 2010 to categorize Basque citizens using the Johns Hopkins Adjusted Clinical Groups case-mix approach [31]. The preceding bibliography provides a more comprehensive insight [32]. Although type 2 diabetes can develop at any age, it is more common in those over the age of 40. Therefore, individuals under the age of 35 were excluded. There were 12 months of successful periods formed and accomplished (year 1: from September 2007 to August 2008; year 2: from September 2008 to August 2009; year 3: from September 2009 to August 2010; and year 4: from September 2010 to August 2011). A patient was considered to have Type 2 diabetes mellitus (T2DM) throughout the illness on a specific date before the chosen end-point (i.e., prior to 01-09-2007, or preceding 01-09-2008, and so on). Furthermore, the patient was expected to have public insurance at the start of the calendar year but was not required to have it for the entire calendar year. As a result, many individuals have been diagnosed with T2DM. There were 116,295 people in year one, 123,991 in year two, 130,554 in year three, and 134,421 in year four. The total number of T2DM samples investigated was 149,015 [25], [27].

The diabetes dataset contains variables such as the patients' age, hemoglobin, retinopathy, and John Hopkin Aggregated Diagnosis Groups (ADGs). It may be noted that the relationship of these parameters with T2DM is not reflected. To demonstrate the prevalence of complications, a minimum primary dataset and hospital records were used. These variables are included in the diabetes database. Hospital admissions due to acute myocardial infarction (MI), major amputation, or unnecessary hospitalizations, also known as ambulatory care sensitive conditions (ACSC) which were identified independently for each observation period. A list of 52 medical problems was created to evaluate the acute diseases, and precise criteria were set to consider those active diseases between the dates of September 2010 and August 2011, using a methodology previously described by the authors [25], [27].

Diabetes mellitus was not one of the 52 stated health conditions. Furthermore, this set of data was divided into two parts: a) related acute comorbidities, which include seven diseases such as ischemic heart disease, renal failure, stroke, heart failure, peripheral neuropathy, foot ulcers, and diabetic retinopathy, and b) unrelated acute diseases, which follow the other 44 health problems listed. Thus, the raw DB was made up of 321 variables.

## B. METHODS

The limitations of existing NISQ devices create constraints on QML techniques. For example, several proposed QML applications have recently been dependent on exploiting publicly well-known datasets, despite the relatively standard pre-processing methods. These methods are not always appropriate for sufficient data to be prepared by Quantum Classifiers when working with real datasets. In this research, a pre-processing method is proposed, as illustrated in Figure 1, which encodes the data before implementing it into QML algorithms.

### 1) PRE-PROCESSING

The Feature Selection (FS) technique is extensively used in ML and DL [33] to select the best features from the set of features that allows effective prediction of the outputs. Thus, the FS method helps to enhance the performance of the model. The feature realm has increased from a few to several features used in ML applications. There are several methods designed to tackle the problem of reducing irrelevant and excessive features. A reduction of features assists in perceiving the most relevant data and overcoming the excessive feature problem that affects the model's performance [34].

In this manuscript, three ML classifiers are employed:

- 1) Random forest (entropy-based)
- 2) Logistic regression (on a logit basis)
- 3) Support vector machine (Linear Kernel-based)

These three classifiers are widely used in various applications. However, working with different classifiers always presents problems of posed entropy. To avoid the entropy problem, we use entropy-based classifiers [35]. These classifiers select

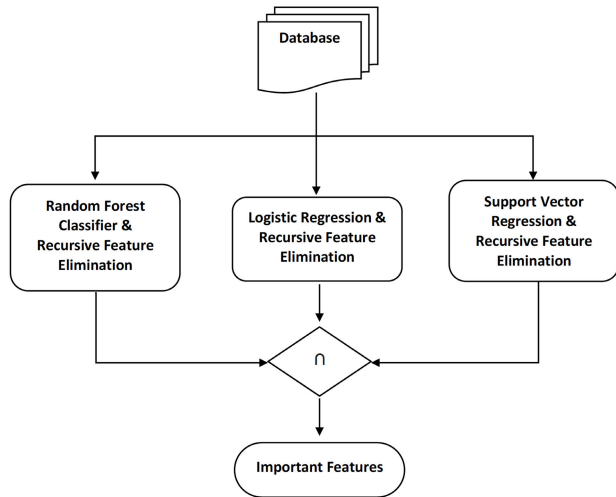


FIGURE 1. Feature Selection Flow diagram.

the best features from an individual classifier’s training and extract the features.

The Recursive Feature Elimination (RFE) selection technique is fundamentally a recursive/cursive method that sorts the features according to their importance [36]. RFE provides an external estimator that assigns weights to the features. A random forest classifier with RFE is used to classifier and tune some parameters, including the number of estimators and maximum depth. Logistic regression with RFE uses the Broyden Fletcher Goldfarb Shannon (LBFGS) solver as an estimator and tunes the number of steps and volume. Support Vector Regression with RFE, uses a linear kernel and a gamma parameter. These classifiers are trained individually and they create a set of features. We have employed the intersection technique, which includes standard features in the individual classifiers provided, these three classifiers extracted the eight important features from all datasets, including the synthetic dataset, the sonar dataset, and the diabetes dataset, as illustrated in Figure 1.

Following the FS method, which transforms the raw data into a practical and straightforward format in subsequent steps, in the next step, the data is normalized using the min-max technique. As a result, the data will be in the range of 0 and 1, which reduces the training time of models. Here, 0 and 1 represent the maximum and minimum values of each feature, respectively [12], [22].

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (1)$$

## 2) STATE PREPARATION

State preparation is a need in QML to prepare data for processing. For example, a typical classification of the function in supervised learning calculates the function  $f$  to map the input data  $x$  and the output labels  $y$  to become  $y = f(x)$ . The fundamental goal of classification is to improve the

accuracy of prediction models. The binary classification  $B = \{c_1, c_2, \dots, c_n\}$ , where  $B$  is the target label and a collection of data in the training phase such that it may be represented in the traditional ML domain as [37].

$$D = \{(x_1, y_1), \dots (x_i, y_i), \dots (x_n, y_n)\} \quad (2)$$

In above  $x_i$  is some of the features ( $n$ ) on the properties of the order of data point  $i$  and  $y_i$  is the corresponding data point. In the case of binary classification  $y_i \in \{c_1, c_2\}$ , where  $x_i \in R^d$  and  $d$  is real-valued features. To illustrate the framework to analyze in the QML domain, we first should transform classical data to quantum data, represented in training data [37].

$$F_n = \{(|\psi_1\rangle, y_1), \dots (|\psi_i\rangle, y_i), \dots, (|\psi_n\rangle, y_n)\} \quad (3)$$

where  $|\psi_i\rangle$  represents the quantum state of  $F_n$ ,  $|\psi_i\rangle \in C^{2^d}$  and  $y_i \in \{c_1, c_2\}$  in the case of data stratification. There are several techniques to embed the classical data into quantum data into high dimensions. There are two ways to implement this viz -a -viz Basis encoding and Amplitude encoding.

### a: BASIS ENCODING

Basis encoding is the most common technique to embed classical data into a quantum state. This technique has a relationship between n-bit classical datapoints and the computational basis of n-qubit datapoints, such as the classical data (1001) encipher to four qubits  $|1001\rangle$  quantum data, via following equation [38].

$$|D\rangle = \frac{1}{\sqrt{M}} \sum_{m=1}^M |x^m\rangle \quad (4)$$

whereas,  $D$  is classical data,  $D = \{x^1, x^2, \dots, x^M\}$  that creates binary vector,  $x^m = \{b_1^m, b_2^m, \dots, b_N^m\}$ ,  $b_i^m \in \{0, 1\}$ ,  $i \in \{1, 2, \dots, M\}$  and  $M$  is the number of attributes.

### b: AMPLITUDE ENCODING

The fundamental concept of amplitude encoding is the coupling of classical data with quantum state amplitudes. A normalized classical vector is shown in the following equation to encode classical data string to quantum amplitude string [38], [39].

$$x = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} \quad (5)$$

whereas,  $x$  is a normalized classical string,  $x \in \Sigma C^{2^n}$ ,  $C$  is complex numbers. Generally, quantum state amplitudes can be encoded by the following equation [38], [39].

$$|\psi_x\rangle = \sum_{i=1}^{2^n} x_i |i\rangle \quad (6)$$

where  $|\psi\rangle \in$  Hilbert space ( $\mathcal{H}$ ) and  $\sum_i |x_i|^2 = 1$ .

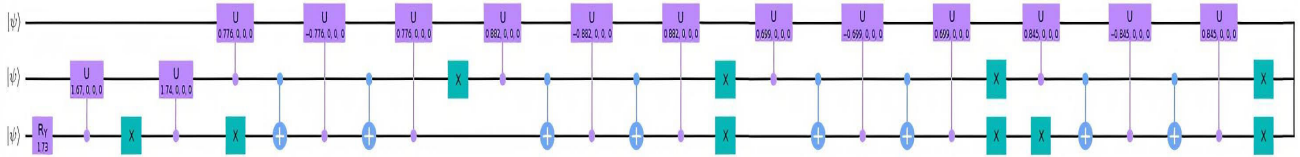


FIGURE 2. Amplitude Encoding VQC Circuit.

For this approach, prior to creating the amplitude encoded states, the data is transformed to their angle depictions using multi-controlled rotations executed. Where the angle  $\theta$  is created, the dataset vector  $x_i$  can be used to denote the  $i^{th}$  attribute, and  $\beta$  is the angle based on the arcsin of the set of parameters in the probability distribution. As a result, there is a correlation between the number of dimensions in the specimen and the angles used to communicate the sample's characteristics [21], [38].

$$|\psi\rangle = R(x_i, \beta) |q_1 \dots q_{s-1}\rangle |q_s\rangle \quad (7)$$

The state  $|\psi\rangle$  is configured in a parallel circuit. The  $R_y$  gates rotations are implemented so that  $nR_y$  operations gates are performed, where  $n$  is the binary influence for encoding a feature vector  $x_i$  [21]. The primary benefit of this encoding is that it requires only  $n$  qubits for an array of  $p = 2^n$  elements. This indicates that if a quantum method is polynomial of  $n$  dimensions and its latency will be polylogarithmic in relation to data dimension. Mottonen *et al.* [39] presented a potential approach for amplitude encoding, which is employed for studies in this research. The objective of this method is to map any state  $|x\rangle$  to the ground  $|0 \dots 0\rangle$  [40]. After obtaining the circuit, all procedures are flipped and executed in reverse order, as shown in Figure 2.

### III. QUANTUM MACHINE LEARNING ALGORITHMS

The quantum support vector machine and the variational quantum classifier are the two most commonly used supervised QML algorithms. In this work, we are considering both of these QML algorithms, which are briefly discussed below.

#### A. QUANTUM SUPPORT VECTOR MACHINE

The Support Vector Machine (SVM)[11], [41] is a supervised ML technique that can perform binary or multiclass classification. SVM separates the two data groups by drawing a line between them and separating the data as accordingly. In some cases, complexity can occur. Sometimes, a line cannot distinguish the data effectively when the line is highly non-linear. SVM aims to discover the relevant variables  $w$  and  $b$ , which precisely fit the data points with tags  $y = 1$  and assured  $wx + b > 1$  and the datapoints  $y = -1$  assured  $wx + b < -1$ . Additionally, the margin between the hyperplane  $\frac{2}{\|w\|}$  is maximized as shown in Figure 3.

A technique called the kernel method [19], [20] is used, which comprehends using a feature map to plot the data into a higher space, where a separable hyperplane can be drawn.

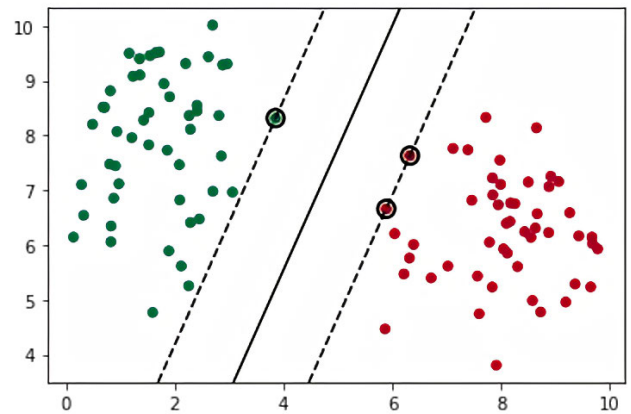


FIGURE 3. Support Vector Machine.

The numerical form of the optimization plight looks identical to the quadratic equation with linear restrictions,  $\min_{w,b} \|w\|^2$  such that  $y_i(wx_i + b) \geq 1, i = 1 \dots N$ . The minimizing of the optimization problem is the same as the minimization of its dual issues. The dual problem has the form with constraints  $0 \leq \alpha_i \leq C$  and  $\sum_i \alpha_i y_i = 0$ . The most significant advantage of this formulation we will see shortly, when its quantum counterpart will be introduced [11], [19].

$$\max_{\alpha_i} \sum_i -\frac{1}{2} \sum_{j,k} \alpha_i \alpha_k y_j y_k (x_k^T x_j) \quad (8)$$

Although data points are not linearly divisible, we implement the kernel method to distinguish the data. First, the data points are deep-seated in 2-dimensional space with the map  $\phi: x \rightarrow (x, x^2)$ , which guides a simple linear segregation problem in this technique. Then, enumerating the scalar product  $x_k^T x_k$  is all that is required in this framework. Furthermore, we encoded the data points that followed the pattern  $\phi(x_k^T) \phi(x_k)$  and elaborated on the advantages of the dual configuration.

The QSVM is another approach that is known as a quantum-enhanced technique [11], [20] because the quantum algorithm is classical, mainly with some operations executed by a quantum processor. However, QSVM works in the same way as SVM does.

The QC converts the classical sample points  $\vec{x}$  into quantum variables  $|\phi(\vec{x})\rangle$ . A unitary gate is appropriate for the job since it spins the qubit to a specific value  $U_{\phi}(\vec{x})|0\rangle$ , where

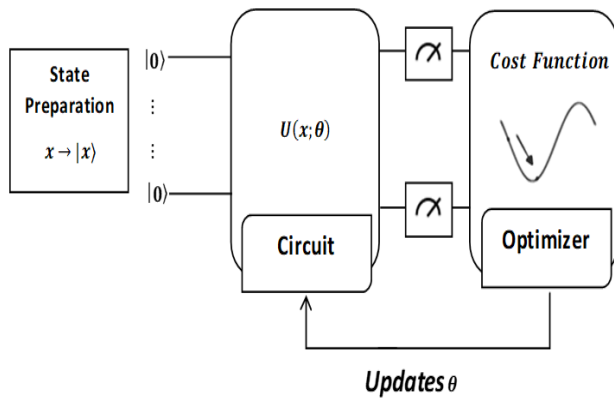


FIGURE 4. Variational Quantum Classifier.

$\phi(\vec{x})$  is an arbitrary classical function applied to the classical data  $x$ . In order to obtain a classical value of 1 or -1 for each classical input, we perform measurement operation, which depends on generalized quantum circuit  $W(\theta)$ . As a result, we can confidently assert that these test datasets are associated with the relevant labelled data. Such procedures define  $W(\theta)U_{\phi}(\vec{x})|0\rangle$  as an ansatz for this type of classification.

**B. VARIATIONAL QUANTUM CLASSIFIER**

The Variational Quantum Classifier (VQC) is a key QML algorithm to classify physics events of interest from background events. It is a supervised QML algorithm that is widely used for classification problems in the NISQ device. Havlicek et al. [12] proposed the VQC model, which enables us to get exploratory findings on NISQ devices without the need for extra error-correction approaches. The cost function is calculated using iterative device measurements, which helps to mitigate errors by integrating noisy data in the optimization computations [21], [23]. This quantum method uses the mapping of classical input data to an increasingly ample quantum feature space, which is based on quantum circuits that are difficult to mimic conventionally.

VQC starts with the initial state preparation of QML problems, in which various feature mapping techniques embed classical data into quantum computing. The variational circuit or ansatz, equal to the number of measurements and dimensions. Finally, the measured value is transmitted to a circuit as feedback to improve the variational circuit’s trainable parameters, as shown in Figure 4. While optimization is not a part of Quantum circuit, classical optimizer is a part of it. The VQC algorithm has a training stage and a testing stage.

**1) FEATURE MAP**

The key concept behind quantum feature mappings [12], [13] is derived from the conventional machine learning kernel technique, in which a dataset is non-linearly mapped into a higher-dimensional space in order to find a hyperplane that

classifies non-linear data [12], [23].

$$U_{\Phi(x)} = \prod_d U_{\Phi(x)} H^{\otimes n} U_{\Phi(x)} H^{\otimes n} \tag{9}$$

For Instance, a quantum feature map  $\phi(\vec{x})$  is a map from the classical feature vector  $|\phi(\vec{x})\rangle \langle\phi(\vec{x})|$  the quantum states, a vector in Hilbert space [12]. By applying the unitary operation on the initial state, we have now blown up the dimension of our feature space ( $Z_i$ ), and the task of our classifier is to find a separating hyperplane in this new space. Which contain the layer of Hadamard gates ( $H$ ) interleaved with entangling blocks encoding the classical data and depth ( $d$ ) of the circuit by the following equation [12], [23].

$$U_{\Phi(x)} = \exp\left(i \sum_{S \subseteq [n]} \phi_S(x) \prod_{k \in S} Z_k\right) \tag{10}$$

The number of qubits required is proportional to the dimension of the data. The data is encoded through the unitary gates  $U_{\Phi(x)}$  by varying the angle to a particular values. We used several feature maps in this classification, such as FirstOrderExpansion, SecondOrderExpansion, and SecondOrderPauliExpansion.

Encoding methods for various feature maps are as follows [12], [17], [19].

FirstOrderExpansion

$$\phi_S : x \mapsto x_i \tag{11}$$

SecondOrderExpansion

$$\phi_S : x \mapsto (\pi - x_i) (\pi - x_j) \tag{12}$$

SecondOrderPauliExpansion

$$\phi_S : x \mapsto \sin(\pi - x_i) \sin(\pi - x_j) \tag{13}$$

The quantum advantage comes into the picture when we use non-classically simulated quantum feature maps over feature maps that can be simulated on classical computers.

**2) VARIATIONAL CIRCUIT**

The main idea in this approach is to optimize the parameters using an objective function as a guide. The quantum and classical phases are the two distinct phases of variational quantum circuits, as shown in Figure 6. That process comprises state preparation, the variational quantum circuit parameterized input  $x$  based on the number of parameters  $\theta$  and measurement [12]. The output of the circuit, the objective function, and the learning algorithm are all part of the classical phase. The optimization techniques such as constrained optimization by linear approximations [42] estimates the VQC as shown in Figure 5. In addition, the variational circuit is utilized to solve complex optimization issues [12], [23]

$$|\psi(x : \theta)\rangle = U(\theta) |\phi(x)\rangle \tag{14}$$

For parameterized variational circuit with interlinking parameters  $R_y, R_z$  gates and entangles with CNOT gate.

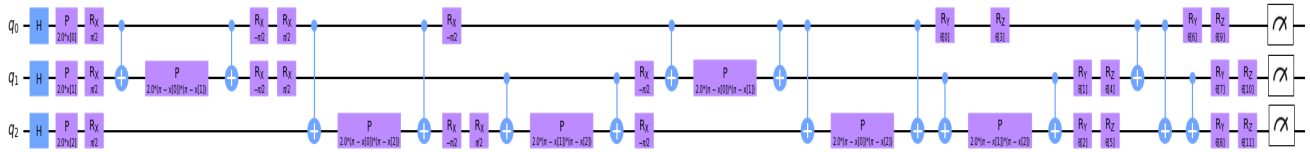


FIGURE 5. A typical VQC full circuit with feature map and variational form.

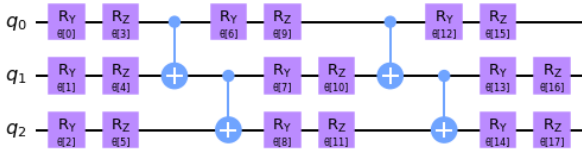


FIGURE 6. An encoding circuit with CNOT, RY and RZ gates.

### 3) MEASUREMENT

The next phase is the measurement stage, which assesses the class possibilities by carrying out a decisive measurement. It is the same as taking many samples from a distribution of potential computational base states and calculating the average value. Then, for an elaboration of the final purpose circuit, we have PauliFeatureMap and a variational circuit, EfficientSU2, with the depth of circuit is 2. Finally, to design a simple schematic diagram of the entire model, as shown in Figure 5.

The goal of training is to determine the values of parameters that will optimize a particular loss function. We can optimize a quantum model similarly to optimize a conventional neural network [43]. In both situations, we run the model forward and determine the loss function. Since a quantum circuit’s gradient [21] can be computed, we may update our trainable parameters using gradient-based optimization methods as a loss function during training. Using this method, we can determine the distance between our predictions and the truth, expressed by a loss function value.

### 4) OPTIMIZATION

The parameters of the quantum variational circuit are updated using a optimization routine once the measurements are ready. The classical loop trains our parameters until the cost function’s value decreases.

#### *α*: CONSTRAINED OPTIMIZATION BY LINEAR APPROXIMATIONS

The Constrained Optimization by Linear Approximations (COBYLA) optimizer [42] generates sequential linear assumptions of the cost function and impediments using an  $n + 1$  fundamental ( $n$  is the number of features) and improves these assumptions in a trusted region at each stage. In addition, the COBYLA scaffolds balance impediments by converting them into two different impediment variations.

### C. EVALUATION OF CLASSIFIERS

The performance of our models is evaluated using conventional evaluation matrices such as gradients mean and variance, precision, recall, accuracy, and f1-score, which are provided in eq [15]–[19].

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t + \epsilon}} \hat{m}_t \quad (15)$$

The optimizer utilized the COBYLA optimizer, with a learning rate of  $\eta = 0.0001$ . Whereas, the previous gradient  $\theta_t$ , the current gradient  $\theta_{t+1}$  of the optimizer. For the loss function, Whereas,  $m_t$  weight and  $v_t$  momentum being approximations of the gradients mean and variance.

$$Precision = \frac{T_P}{T_P + F_P} \quad (16)$$

$$Accuracy = \frac{T_P + T_N}{T_P + T_N + F_P + F_N} \quad (17)$$

$$f1-Score = 2 * \frac{T_P}{2T_P + F_P + F_N} \quad (18)$$

$$Recall = \frac{T_P}{T_P + F_N} \quad (19)$$

In the above equations, true positive is represented by  $T_P$ , false positive by  $F_P$ , true negative by  $T_N$ , and false negative by  $F_N$  [21], [44], [45].

## IV. RESULTS & DISCUSSION

The training data are used to develop each classifier, and the test examples are used to compare the classification model’s predicted labels to known test tags. For example, the labels “0” and “1” represent negative and positive values, respectively. To distribute our testing and training samples, we divided our complete database into 80% for training and 20% for testing, preserving the same subcategory percentage in each subgroup (50 percent of 0 and 50 percent of 1). In the study, ML and QML methods such as SVM, QSVM, and VQC are utilized to solve a binary classification problems on three distinct datasets.

### A. SYNTHETIC DATASET

The performance of our proposed models, including SVM, QSVM, VQC and amplitude encoding VQC on the synthetic dataset, are depicted in Tables 2 and 3.

The classical SVM achieved 100% accuracy in the synthetic dataset, and quantum algorithms such as QSVM and VQC reached 94% and 75% accuracies, respectively.

**TABLE 2. Performance of SVM, QSVM and VQC for all datasets.**

Dataset	Classifiers	Confusion Matrix			Performance Evaluation				Average Values	
		Prediction Class → Actual class ↓	0	1	Precision	Recall	F1-score	Test Size	Accuracy	F1-score
Synthetic (Baseline)	SVM	0	50	0	100%	100%	100%	50	100%	100%
		1	0	50	100%	100%	100%	50		
	QSVM	0	49	1	91%	98%	94%	50	94%	94%
		1	5	45	98%	90%	94%	50		
	VQC	0	33	17	80%	66%	73%	50	75%	75%
		1	8	42	71%	84%	77%	50		
Sonar	SVM	0	17	3	85%	85%	85%	20	85.71%	86%
		1	3	19	86%	86%	86%	22		
	QSVM	0	14	6	78%	70%	74%	20	76.19%	76%
		1	4	18	75%	80%	78%	22		
	VQC	0	15	5	68%	75%	71%	20	71.4%	71%
		1	7	15	75%	68%	71%	22		
Diabetes	SVM	0	201	64	75%	76%	76%	265	75.32%	75.5%
		1	66	200	76%	74%	75%	266		
	QSVM	0	203	62	73%	77%	75%	265	74.19%	74.5%
		1	75	191	75%	72%	74%	266		
	VQC	0	177	88	69%	67%	68%	265	68.73%	69%
		1	78	188	68%	71%	69%	266		

**TABLE 3. The accuracy of the proposed model against the selected number of epochs.**

Dataset	Encoding	Layers	Epochs	Accuracy (%)
Synthetic (Baseline)	Amplitude Encoding	5	100	98.40
Sonar	Amplitude Encoding	5	100	67.30
Diabetes	Amplitude Encoding	5	100	74.50

The VQC Model is implemented using the amplitude encoding method on the synthetic dataset which achieved 98.40% testing accuracy with a depth of 5 layers and 100 epochs, as depicted in Figure 7.

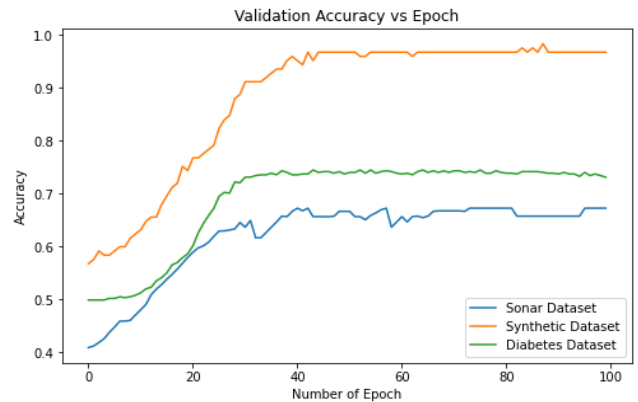
We compared the outcomes of synthetic dataset to previous studies with complicated designs. *D.Sierra-Sosa et al.* [21] used the Amplitude-Hybrid model to predict negative and positive classes on a synthetic dataset and achieved a maximum accuracy 97.4%, respectively. Even though this study used the TensorFlow Quantum Amplitude Hybrid model on the Google Cirq quantum system, the results were positive. However, in contrast to our technique, the VQC method achieved 75% accuracy, whereas the amplitude encoding based VQC method achieved 98.40% accuracy.

In favourably compares SVM, QSVM, VQC and amplitude encoding VQC in terms of accuracy, recall, precision, and F1-score. However, the overall accuracy of SVM is outshined by the QSVM, VQC, and amplitude encoding VQC. The SVM achieved 1.6% more accuracy than amplitude encoding based VQC model. Eventually, amplitude encoding VQC achieved almost the same accuracy as a classical algorithm (SVM).

In conclusion, our Amplitude encoding model produced substantial results on the synthetic dataset and was competitive with recently published research, as shown in table 4.

**TABLE 4. Comparative study of Synthetic dataset.**

Author	Method	Accuracy
D. Sierra-Sosa et al [21]	Amplitude Hybrid Model	97.4%
Proposed	Amplitude Encoding + VQC	98.4%



**FIGURE 7. The accuracy of the proposed model against the selected number of epochs.**

**B. SONAR DATASET**

The performance of the employed models, including SVM, QSVM, VQC and amplitude encoding VQC are summarized in Tables 2 and 3.

In the Sonar dataset, the conventional SVM had an accuracy of 85.71 percent. In contrast, quantum techniques like QSVM and VQC had accuracies of 76.19 percent and 71.4 percent, respectively

The VQC Model is implemented on the sonar dataset using the amplitude encoding approach, with a depth of 5 layers and 100 epochs. As a result, the testing accuracy is achieved as 67.30%, as shown in Figure 7.



**TABLE 5. Comparative study of Sonar dataset.**

Author	Method	Accuracy
S. Chakraborty et al [22]	HQFSA	74%
Proposed	Amplitude Encoding + VQC	67.3%

We compared the competitiveness of our research work conducted on the sonar data to some recent published work using advanced designs. For example, *S. Chakraborty et al.* [22] utilized a hybrid quantum feature selection algorithm (HQFSA) to forecast metal and rocks, achieving an accuracy of 74% on a sonar dataset containing 60 features. Although, using quantum parallel amplitude estimation and amplitude amplification. While our VQC approach obtained 71.43% accuracy and the amplitude encoding VQC technique achieved 67.3% accuracy.

The comparison between SVM, QSVM, VQC and amplitude encoding VQC is favourable and effective in performance evaluation. SVM outperformed the efficacy of QSVM, VQC, and amplitude encoding VQC in terms of precision, recall, F1-score, and overall accuracy. As a result, the SVM was 18.41% more accurate than VQC model based on amplitude encoding.

In conclusion, our Amplitude encoding model produced substantial results on the Sonar dataset and was competitive with recently published research, as shown in Table 5.

### C. DIABETES DATASET

Performance metrics of our models of SVM, QSVM, VQC and amplitude encoding VQC performance metrics are presented in Tables 2 and 3.

Using the diabetes dataset, the classical SVM obtained 75.32% accuracy, while the quantum models, QSVM and VQC recorded 74.19% and 68.73% accuracies, respectively.

The amplitude encoding technique is implemented on VQC using Diabetes dataset, which has depth of 5 layers and 100 epochs, the validation accuracy achieved 74.5% is depicted in Figure 7.

We evaluated the competitiveness of diabetes outcomes to previously presented studies with advanced designs. *D. Sierra-Sosa et al.* [23] utilized the VQC model to predict diabetes with acute disease and reached a maximum accuracy of 72% on the Diabetes dataset with three variables. Despite the use of three separate attention processes in conjunction with the VQC model, the results of this investigation were satisfactory.

*H. Gupta et al.* [24] employed the VQC model to predict related to diabetic disease using the Diabetes dataset, and they obtained a maximum accuracy of 74%. Regardless, the study employed PIMA diabetes data, which only includes one type of pregnancy diabetes. On the other hand, our diabetic dataset covers T2DM along with a variety of acute illnesses, demonstrates its robustness of the data.

*D. Maheshwari et al.* [25] predicted Acute diabetic morbidity on a diabetic dataset using the QBoost and Voting model, with a maximum accuracy of 68.3%. Despite the fact

**TABLE 6. Comparative study of Diabetes dataset.**

Author	Method	Accuracy
D. Sierra-Sosa et al [23]	VQC	72%
H. Gupta et al [24]	VQC	74%
D. Maheshwari et al [25]	Qboost + Voting Model	68.3%
Proposed	Amplitude Encoding + VQC	74.4%

that the DWave system was utilized to merge two different attention processes with the QBoost and voting models in this investigation, the findings were promising. On the other hand, our VQC approach achieved 68.73% accuracy and amplitude encoding based on VQC technique reached 74.50% accuracy.

In order to compare SVM, QSVM, VQC and amplitude encoding VQC in terms of accuracy, recall, precision, and F1-score. SVM has performed slightly better than QSVM, VQC, and amplitude encoding VQC in terms of overall accuracy. The SVM achieved 0.82% more accuracy than amplitude encoding based VQC model. Eventually, amplitude encoding VQC achieved almost the similar accuracy as a classical algorithm (SVM).

In summary, as shown in Table 6, our Amplitude encoding model produced substantial results on a diabetic dataset and was competitive with recently published studies.

### V. CONCLUSION

In this article, we implemented the VQC model using basis and amplitude encoding techniques. We used amplitude encoding, therefore which should not be the only evaluation optimization, we used to improve a quantum framework. Furthermore, state preparation is simply one aspect of QML algorithms to benefit from, whether implemented into a quantum system. We suggested a pre-processing approach for improving the quantum state preparation for VQC. Our results showed VQC achieved 75%, 71.4%, and 68.73% efficiencies, Similarly amplitude encoding VQC performed 98.4%, 67.30% and 74.50% accuracies on the synthetic, Sonar, and Diabetes datasets, respectively. As a consequence, all the databases perform in conformity well with basis and amplitude encoding based VQC. The use of amplitude encoding VQC improved prediction rates for synthetic and diabetic datasets but has little influence performance on sonar dataset. The outcomes obtained using the optimal QML model were compared to state-of-the-art models. The comparative study revealed that the generated QML model beat all other studies.

The future direction will be to use different data encoding techniques such as repeated amplitude encoding, angle encoding, or other encoding methods to enhance the QML models and increase the number of features to enhancing performance relative to the established models and cutting-edge techniques.

### ACKNOWLEDGMENT

This study was funded by the Basque government's IT-905-16 grant to the eVIDA group and was facilitated by Osakidetza, which contributed to the database. In addition, the Clinical Research Ethics Committee of Euskadi (PI2014074) in Spain validated the research protocols. How-

ever, because the patient health data were anonymized and de-identified prior to extrapolation, informed consent was not obtained.

## REFERENCES

- [1] G. Carleo, Y. Nomura, and M. Imada, "Constructing exact representations of quantum many-body systems with deep neural networks," *Nature Commun.*, vol. 9, no. 1, pp. 1–11, Dec. 2018.
- [2] W. Liu, P. Gao, Y. Wang, W. Yu, and M. Zhang, "A unitary weights based one-iteration quantum perceptron algorithm for non-ideal training sets," *IEEE Access*, vol. 7, pp. 36854–36865, 2019.
- [3] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, pp. 195–202, Sep. 2017.
- [4] C. Ciliberto, M. Herbster, A. D. Ialongo, M. Pontil, A. Rocchetto, S. Severini, and L. Wossnig, "Quantum machine learning: A classical perspective," *Proc. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 474, no. 2209, Jan. 2018, Art. no. 20170551.
- [5] M. Schuld and N. Killoran, "Quantum machine learning in feature Hilbert spaces," *Phys. Rev. Lett.*, vol. 122, no. 4, p. 40504, Feb. 2019.
- [6] J. D. Martín-Guerrero and L. Lamata, "Quantum machine learning: A tutorial," *Neurocomputing*, vol. 470, pp. 457–461, Jan. 2022.
- [7] N. Wiebe, A. Kapoor, and K. M. Svore, "Quantum deep learning," *Quantum Inf. Comput.*, vol. 16, nos. 7–8, pp. 541–587, 2016.
- [8] R. Somma, G. Ortiz, E. Knill, and J. Gubernatis, "Quantum simulations of physics problems," *Int. J. Quantum Inf.*, vol. 1, no. 2, pp. 189–206, 2003.
- [9] M. Schuld, I. Sinayskiy, and F. Petruccione, "An introduction to quantum machine learning," *Contemp. Phys.*, vol. 56, no. 2, pp. 172–185, 2015.
- [10] D. Sierra-Sosa, J. Arcila-Moreno, B. Garcia-Zapirain, C. Castillo-Olea, and A. Elmaghraby, "Dementia prediction applying variational quantum classifier," 2020, *arXiv:2007.08653*.
- [11] M. Schuld, "Supervised quantum machine learning models are kernel methods," 2021, *arXiv:2101.11020*.
- [12] V. Havlíček, A. D. Córcoles, K. Temme, A. W. Harrow, A. Kandala, J. M. Chow, and J. M. Gambetta, "Supervised learning with quantum-enhanced feature spaces," *Nature*, vol. 567, no. 7747, pp. 209–212, Mar. 2019.
- [13] E. Farhi and H. Neven, "Classification with quantum neural networks on near term processors," 2018, *arXiv:1802.06002*.
- [14] N. Moll, P. Barkoutsos, L. S. Bishop, J. M. Chow, A. Cross, D. J. Egger, S. Filipp, A. Fuhrer, J. M. Gambetta, M. Ganzhorn, and A. Kandala, "Quantum optimization using variational algorithms on near-term quantum devices," *Quantum Sci. Technol.*, vol. 3, p. 30503, Jun. 2018.
- [15] S. Lloyd, M. Mohseni, and P. Rebentrost, "Quantum principal component analysis," *Nature Phys.*, vol. 10, pp. 631–633, Jul. 2014.
- [16] J. Romero, J. P. Olson, and A. Aspuru-Guzik, "Quantum autoencoders for efficient compression of quantum data," *Quantum Sci. Technol.*, vol. 2, p. 45001, Aug. 2017.
- [17] M. Schuld, A. Bocharov, K. M. Svore, and N. Wiebe, "Circuit-centric quantum classifiers," *Phys. Rev. A, Gen. Phys.*, vol. 101, no. 3, p. 32308, Mar. 2020.
- [18] G. Verdon, M. Broughton, and J. Biamonte, "A quantum algorithm to train neural networks using low-depth circuits," 2017, *arXiv:1712.05304*.
- [19] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Phys. Rev. Lett.*, vol. 113, no. 13, Sep. 2014, Art. no. 130503.
- [20] Y. Ruan, X. Xue, H. Liu, J. Tan, and X. Li, "Quantum algorithm for K-nearest neighbors classification based on the metric of Hamming distance," *Int. J. Theor. Phys.*, vol. 56, no. 11, pp. 3496–3507, Nov. 2017.
- [21] D. Sierra-Sosa, M. Telahun, and A. Elmaghraby, "TensorFlow quantum: Impacts of quantum state preparation on quantum machine learning performance," *IEEE Access*, vol. 8, pp. 215246–215255, 2020.
- [22] S. Chakraborty, S. H. Shaikh, A. Chakrabarti, and R. Ghosh, "A hybrid quantum feature selection algorithm using a quantum inspired graph theoretic approach," *Int. J. Speech Technol.*, vol. 50, no. 6, pp. 1775–1793, Feb. 2020.
- [23] D. Sierra-Sosa, J. D. Arcila-Moreno, B. Garcia-Zapirain, and A. Elmaghraby, "Diabetes type 2: Poincaré Data preprocessing for quantum machine learning," *Comput., Mater. Continua*, vol. 67, no. 2, pp. 1849–1861, 2021.
- [24] H. Gupta, H. Varshney, T. K. Sharma, N. Pachauri, and O. P. Verma, "Comparative performance analysis of quantum machine learning with deep learning for diabetes prediction," *Complex Intell. Syst.*, vol. 2021, pp. 1–15, May 2021.
- [25] *QPU Topology—Ocean Documentation 3.1.1 Documentation*. Accessed: Nov. 23, 2021. [Online]. Available: <https://docs.ocean.dwavesys.com/en/stable/concepts/topology.html>
- [26] D. Maheshwari, B. Garcia-Zapirain, and D. Sierra-Sosa, "Machine learning applied to diabetes dataset using quantum versus classical computation," in *Proc. IEEE Int. Symp. Signal Process. Inf. Technol. (ISSPIT)*, Dec. 2020, pp. 1–6.
- [27] D. Sierra-Sosa, B. Garcia-Zapirain, C. Castillo, I. Oleagordia, R. Nuno-Solinis, M. Urtearan-Laresgoiti, and A. Elmaghraby, "Scalable healthcare assessment for diabetic patients using deep learning on multiple GPUs," *IEEE Trans. Ind. Informat.*, vol. 15, no. 10, pp. 5682–5689, Oct. 2019.
- [28] *International Diabetes Federation—Facts & Figures*. Available: <https://idf.org/aboutdiabetes/what-is-diabetes/facts-figures.html>
- [29] H. Abraham et al. *Qiskit 0.34.0 Documentation*. Accessed: Dec. 30, 2021. [Online]. Available: <https://qiskit.org/documentation/>
- [30] R. P. Gorman and T. J. Sejnowski, "Analysis of hidden units in a layered network trained to classify sonar targets," *Neural Netw.*, vol. 1, no. 1, pp. 75–89, Jan. 1988.
- [31] G.-L. Du, Y.-X. Su, H. Yao, J. Zhu, Q. Ma, A. Tuerdi, X.-D. He, L. Wang, Z.-Q. Wang, S. Xiao, S.-X. Wang, and L.-P. Su, "Metabolic risk factors of type 2 diabetes mellitus and correlated glycemic control/complications: A cross-sectional study between rural and urban Uygur residents in Xinjiang Uygur autonomous region," *PLoS ONE*, vol. 11, no. 9, Sep. 2016, Art. no. e0162611.
- [32] P. C. Austin, B. R. Shah, A. Newman, and G. M. Anderson, "Using the Johns Hopkins' aggregated diagnosis groups (ADGs) to predict 1-year mortality in population-based cohorts of patients with diabetes in Ontario, Canada," *Diabetic Med.*, vol. 29, no. 9, pp. 1134–1141, Sep. 2012.
- [33] Z. Hameed, S. Zahia, B. Garcia-Zapirain, J. J. Aguirre, and A. M. Vanegas, "Breast cancer histopathology image classification using an ensemble of deep learning models," *Sensors*, vol. 20, no. 16, pp. 1–17, 2020.
- [34] D. K. Soother, J. Daudpoto, N. R. Harris, M. Hussain, S. Mehran, I. H. Kalwar, T. Hussain, and T. D. Memon, "The importance of feature processing in deep-learning-based condition monitoring of motors," *Math. Problems Eng.*, vol. 2021, pp. 1–23, May 2021.
- [35] A. Kirshners, S. Parshutin, and H. Gorskis, "Entropy-based classifier enhancement to handle imbalanced class problem," *Proc. Comput. Sci.*, vol. 104, no. 3, pp. 586–591, Jan. 2017.
- [36] D. Albashish, A. I. Hammouri, M. Braik, J. Atwan, and S. Sahran, "Binary biogeography-based optimization based SVM-RFE for feature selection," *Appl. Soft Comput.*, vol. 101, Mar. 2021, Art. no. 107026.
- [37] Z. Abohashima, M. Elhosen, E. H. Houssein, and W. M. Mohamed, "Classification with quantum machine learning: A survey," 2020, *arXiv:2006.12270*.
- [38] I. F. Araujo, D. K. Park, F. Petruccione, and A. J. da Silva, "A divide-and-conquer algorithm for quantum state preparation," *Sci. Rep.*, vol. 11, no. 1, p. 6329, Dec. 2021.
- [39] M. Möttönen, J. J. Vartiainen, V. Bergholm, and M. M. Salomaa, "Transformation of quantum states using uniformly controlled rotations," *Quantum Inf. Comput.*, vol. 5, no. 6, pp. 467–473, 2005.
- [40] A. Macaluso, L. Clissa, S. Lodi, and C. Sartori, "A variational algorithm for quantum neural networks," in *Computational Science—ICCS 2020 (Lecture Notes in Computer Science)*, vol. 12142, V. Krzhizhanovskaya et al., Eds. Cham, Switzerland: Springer, 2020, pp. 591–604, doi: [10.1007/978-3-030-50433-5\\_45](https://doi.org/10.1007/978-3-030-50433-5_45).
- [41] O. Jossa-Bastidas, S. Zahia, A. Fuente-Vidal, N. Sánchez Férez, O. R. Noguera, J. Montane, and B. Garcia-Zapirain, "Predicting physical exercise adherence in fitness apps using a deep learning approach," *Int. J. Environ. Res. Public Health*, vol. 18, no. 20, p. 10769, Oct. 2021.
- [42] M. J. D. Powell, "A view of algorithms for optimization without derivatives," *Math. Today-Bull. Inst.*, vol. 43, no. 5, pp. 1–12, 2007.
- [43] H. Ayaz, E. Rodríguez-Esparza, M. Ahmad, D. Oliva, M. Pérez-Cisneros, and R. Sarkar, "Classification of apple disease based on non-linear deep features," *Appl. Sci.*, vol. 11, no. 14, p. 6422, Jul. 2021.
- [44] Z. Hameed and B. Garcia-Zapirain, "Sentiment classification using a single-layered BiLSTM model," *IEEE Access*, vol. 8, pp. 73992–74001, 2020.
- [45] S. Zahia, B. Garcia-Zapirain, I. Saralegui, and B. Fernandez-Ruanova, "Dyslexia detection using 3D convolutional neural networks and functional magnetic resonance imaging," *Comput. Methods Programs Biomed.*, vol. 197, Dec. 2020, Art. no. 105726.



**DANYAL MAHESHWARI** was born in Hyderabad, Pakistan, in 1993. He received the B.E. and M.E. degrees in biomedical engineering from the Mehran University of Engineering Technology, Jamshoro, Pakistan. He is currently pursuing the Ph.D. degree in engineering with the University of Deusto, Bilbao, Spain. During his bachelor's and master's studies, he was an Erasmus Scholar with the University of Limerick, Ireland. He is also working with the eVida Research Team. His research is focused on quantum machine learning for biomedical and medical data.



**BEGONYA GARCIA-ZAPIRAIN** (Member, IEEE) was born in San Sebastián, Spain, in 1970. She received the degree in telecommunication engineering from the University of Basque Country, Spain, in 1994, and the Ph.D. degree in computer science and artificial intelligence from the University of Deusto, Spain, in 2004. From 2002 to 2008, she worked as the Director of the Telecommunication Department, University of Deusto, where she is currently working as a Full Professor. In 2001, she created the eVida Research Group, which is recognized by the Government of the Basque Country, Spain, and the European Network of Living Labs (ENoLL).

...



**DANIEL SIERRA-SOSA** is currently an Assistant Professor with the Computer Science Department, Hood College, with expertise in mathematical modeling, data analytics, artificial intelligence, machine learning, and quantum computing, his research results have been published in well-known journals. He worked the development of an application for the assessment of patients in health care facilities, a predictive model for patient's outcome, in addition to participating in the development of mobile applications. He is also an IBM's Qiskit Advocate and an IBM's Skills Academy Instructor, he teaches courses in quantum computing, deep learning techniques, programming, signal, and image processing at the undergraduate and graduate level and been an advisor for advanced project courses.