Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Heart failure prediction using Pegasos Quantum Support Vector Classifier

Bagoun Yassine[1], Ahmed Zinedine[1]
[1]*Faculty of Sciences Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, Fez, Morocco*
Corresponding Author: Yassine Bagoun
Faculty of Sciences Dhar El Mahraz, Sidi Mohamed Ben Abdellah University, Fez, Morocco.
Email: yassine.bagoun@usmba.ac.ma

ARTICLE INFO

Received: 18 Dec 2024

Revised: 10 Feb 2025

Accepted: 28 Feb 2025

ABSTRACT

Heart failure (HF) is a critical global health problem, and early diagnosis with accuracy is highly essential for effective treatment. the combination of machine learning (ML) with quantum physics has driven research into quantum algorithms for data processing, offering new methods in novel computational environments. While quantum computing is still in its infancy, with quantum algorithms being actively tested and developed, this study proposes a Quantum Machine Learning (QML) approach for HF prediction using the Pegasos Quantum Support Vector Classifier (QSVC) algorithm. Addressing a critical research gap—the lack of direct comparisons between this algorithm and classical ML algorithms in HF — this study conducts a comprehensive evaluation of Pegasos (Primal Estimated sub-GrAdient SOlver for SVM) QSVC against classical models, including Decision Tree (DT), Random Forest (RF), Support Vector Machine (SVM), and k-Nearest Neighbors (K-NN), using a real-world electronic health dataset. The proposed work utilizes quantum feature maps and kernels to improve HF prediction. the results showed that Pegasos QSVC leveraging quantum principles has potential in handling high dimensional data. The results highlight complementary strengths of the quantum and classical approaches. This work establishes the foundation for further QML research highlighting the technologies potential to transform healthcare predictive modeling.

Keywords: Heart failure, Pegasos QSVC, Decision Tree, Random Forest , Support Vector Machine, K-Nearest Neighbors

1. INTRODUCTION

The advent of quantum computing (QC) presents a transformative opportunity for computational science. Leading companies like Google and IBM Systems are at the forefront of developing quantum processors that use quantum mechanical processes to do computations at speeds that are exponentially faster than those of classical computers [1]. This would change everything from cryptography to medical diagnosis [2]. Quantum computers can analyze complicated datasets in novel ways thanks to the principles of quantum mechanics which also make it possible to create quantum algorithms that can improve ML procedures [3]. In genomics for example it has been demonstrated that quantum algorithms increase the precision of predictive analytics [4]. There are already some real-world uses for QC such as IBMs Qiskit an open-source tool for creating quantum algorithms [5].

Classical ML algorithms continue to be widely applied across many domains, including healthcare, in spite of the promise of QC. Techniques such as SVM, kNN, LR, and decision trees have been applied to datasets incorporating genetic profiles, clinical characteristics, and lifestyle factors in predicting disease progression and outcome [6]. However, these algorithms stay limited. They cannot properly handle large-scale data and don't capture complex nonlinear interactions required for a good predictive model [7]. These difficulties are especially noticeable in the healthcare where datasets are frequently heterogeneous and high-dimensional. With these limitations of classical ML, QML has emerged as a promising solution. Complex datasets can be analyzed more quickly and accurately thanks to QML algorithm's ability to treat information in methods that conventional systems can't by utilizing quantum mechanics [8]. The application of QML to healthcare has shown great promise in enhancing predictive

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

modeling. When using QML techniques on healthcare data especially in predictive modeling for chronic diseases several studies have shown notable gains in performance metrics [9], [10]. In addition to finding new patterns in intricate medical datasets that traditional ML might overlook QML can produce more precise diagnoses [11]. To successfully incorporate QC into clinical practice computer scientists and medical professionals will need to work together as the technology advances [12].

Predicting HF a condition marked by the hearts incapacity to pump enough blood into the bodys organs is one area where QML shows great promise. HF affects around 64 million people worldwide and is a leading cause of morbidity and mortality [13]. Although more common in older individuals, HF can affect people of any age. Early prediction of HF is critical for identifying at-risk patients and optimizing therapeutic strategies. HF prediction has made use of classical machine learning algorithms. The prediction of HF has made extensive use of classical ML algorithms. with researchs like that conducted by F. Alotaibi [14] shows how models like RF, SVM and DT are effective in reaching high prediction accuracy. Using the UCI heart disease dataset Alotaibis study produced excellent results: DT accuracy was 93.19%, SVM was 92.30%, RF was 89.14%, LR was 87.36% and Naïve Bayes was 87.27%. The goal of the study is to help with the prompt and precise diagnosis of HF by integrating ML models into medical information systems.

Recent studies have explored the application of QML in healthcare diagnostics, including HF prediction. Alsubai et al. [15] present a study on HF detection using an instance quantum circuit approach combined with conventional predictive analysis. To improve prediction accuracy the study combines deep learning and quantum learning algorithms. It will apply the classifiers like SVM, DT, and RF, followed by their pre-processing in handling the missing values, by applying the Cleveland heart disease dataset. This approach represents notable improvements with respect to the classic methods and therefore is an interesting approach in exploring possibilities opened by quantum computing for healthcare diagnostics. Similarly, a groundbreaking work on the application of QML for the multi-class classification of cardiovascular diseases using electrocardiogram images is presented by Prabhu et al. Their study [16] achieves notable gains over classical models by introducing a unique Pegasos QSVC and a Quanvolutional Neural Network (QNN). The QSVC and Pegasos QSVC outperform the traditional SVC in terms of accuracy. The QNN demonstrates how promising QML is for enhancing the classification of medical images. Beyond healthcare, QML has shown potential in other domains. Bhavsar et al. [17] present a study concerning the application of supervised QML for classifying potentially hazardous asteroids. The authors have proposed a novel technique that makes use of the Pegasos QSVC and Variational Quantum Circuits (VQC) algorithms to improve the accuracy and precision of asteroid hazard prediction. The proposed QML-based results reveal better results compared to classical ML techniques by yielding 92.69% and accuracy of 98.11% average F1-scores. It shows how this is a really powerful space through which the identification and classification of hazardous asteroids might improve space science and consequently planetary defense. In genomic data analysis, N. Singh et al. present [18] an independent implementation of QML algorithms in Qiskit for genomic data classification. Using feature mapping approaches such as ZFeatureMap, ZZFeatureMap, and PauliFeatureMap, they assess QSVC, Pegasos QSVC, VQC, and Quantum Neural Networks. Their research shows significant improvements in accuracy, precision, recall, F1-Score, and AUROC for genomic sequence classification. In order to fully realize these gains, the authors stress the need for additional study with larger datasets, while also highlighting the promise of QML models in genomic data analysis.

The reviewed papers reflect the advances concerning novel approaches and predictive accuracy within the field of ML and QC applications. Studies [14] and [15], using conventional ML models and quantum-inspired approaches, respectively, indicate a significant improvement in the prediction of heart disease. The articles [16] and [17] investigate QML application for classification of cardiovascular diseases, and asteroid hazards prediction using methods such as VQC and Pegasos QSVC respectively. QML algorithms were explored in the study [18] for genomic data classification.These studies illustrate the potential of QML algorithms to enhance predictive modeling in domains requiring the analysis of large and complex datasets.

Despite these advancements, a critical research gap persists: the direct comparison of quantum algorithm Pegasos QSVC with classical ML algorithms for HF prediction using real-world datasets. Existing studies have largely focused on exploring the independent capabilities of QML algorithms in different domains, but the performance of Pegasos

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

QSVC versus conventional methods in the context of HF prediction has not been studied before. Addressing this gap is essential to understand the practical feasibility, computational efficiency, and clinical relevance of QML in medical diagnostics. This study aims to bridge this gap by explore Pegasos QSVC algorithm and conducting a comprehensive comparative analysis of it with classical ML algorithms, including DT, RF, SVM, and KNN, for HF prediction. We utilize F1-score, recall, accuracy, and precision to assess performance. By using this comparative analysis the study offers important insights into how QML can transform predictive modeling in the healthcare laying the groundwork for its incorporation into clinical decision-making procedures and enhancing patient outcomes.

2. METHOD

In this study, the methodology was designed to systematically compare the performance of Pegasos QSVC with traditional ML models, including DT, RF, SVC, and KNN. The methodology followed a structured approach, beginning with data collection and preprocessing, followed by feature encoding, model implementation, and performance evaluation. Below, we detail each step of the process, including the dataset description, preprocessing techniques, and the algorithms used for classification.

2.1. Dataset description

Our goal is to create a prediction model that will help us identify people who are at risk of developing HF. In order to do this, we utilize the retrieved dataset [19] from Kaggle and contains electronic health record (EHR) data of individuals who have HF. Table 1 shows the description of the dataset attributes for this study. 12-clinical variables are included in each patient's profile inside the dataset, and these traits are critical markers for predicting and understanding patterns in HF.

Table 1. Description of dataset attributes

Var	Description
Age	The patient's age (years)
Anaemia	Decline in hemoglobin or red blood cells (boolean)
Creatinine phosphokinase (CPK)	CPK enzyme level (mcg/L)
Diabetes	Is diabeet (boolean)
Ejection fraction	the proportion of blood that leaves the heart during each contraction (%)
Elevated blood pressure	is hypertensive (boolean)
Platelets	Blood platelets (kiloplatelets/mL)
Sex	Male or female (binary)
Serum creatinine	Serum creatinine concentration in the bloodstream (mg/dL)
Serum sodium	Serum sodium concentration in blood (mEq/L)
Smoking	The smoking status of the patient (boolean)
Time	Follow-up period (in days)
Death event	Patient died during the monitoring process (boolean)

2.2. Approach followed

This part of the paper talks about the methodology used to make both classical and QML algorithms work. Figure 1 depicts a workflow for the implementation of QML algorithms. It displays a flow of the main steps in dataset preparation, preprocessing, quantum circuit design, model training, and evaluation. While the diagram emphasizes the Pegasos QSVC process, the structure of the classical models is similar but without those quantum-specific steps,

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

feature encoding and quantum circuit. Below, we detail the specific processes involved in each stage of the workflow.

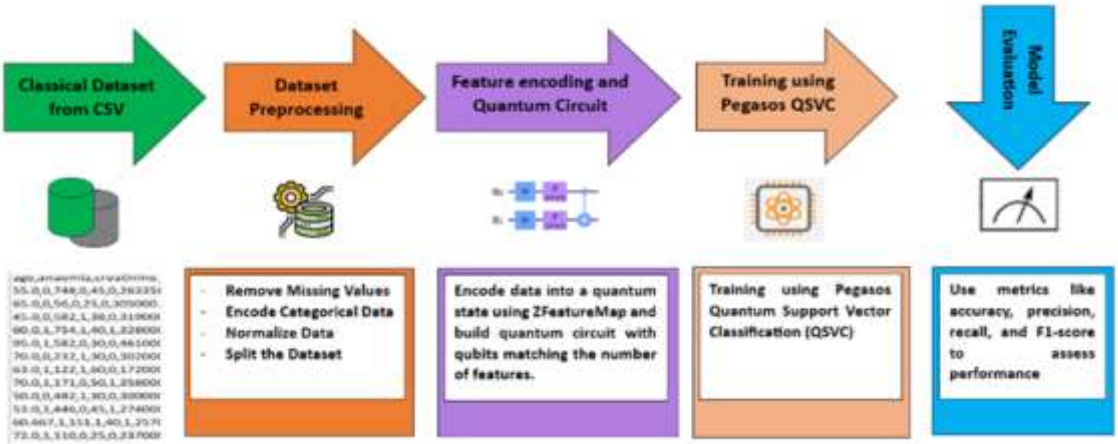

Figure 1. Workflow diagram

1) The dataset was gathered over follow-up periods, and the dataset description section has already covered its specifics.

2) Preprocessing the received dataset involved converting any string values to float and looking for NULL values, as ML methods cannot use string values directly.

3) Many libraries have been used for implementation, including sklearn, matplotlib, time, copy, math, NumPy, and most significantly, qiskit for QC jobs, ZFeature Map. These libraries offer the necessary resources and features for creating and using quantum algorithms.

4) The min-max scaler was used to modify the data, scaling each characteristic to a common range. Regardless of its initial scale, this normalization guarantees that every feature contributes equally during modeling.

5) The dataset was divide into: 80% for train and 20% for test. This is because it is important to have a large dataset to effectively capture the underlying pattern in the data while at the same time ensuring a sufficiently large test set for the evaluation of the model's generalization ability. The 80% ensures variety in the data the model will train on, while 20% serves as a test set to give a right, unbiased estimate of its performance on unseen data. This is one of the standard ratios that many consider to be the accepted standard in ML, and for most mid-sized datasets helps avoid overfitting yet provides a reliable estimate of performance.

6) The qubit count was given, and the general guideline for doing so is as follows: qubit count = features count. There are 12 qubits in our dataset

7) To demonstrate the encoding process, the first row of the dataset was considered as an example. The feature values for this row before encoding are: [50.0, 1, 111, 0, 20, 0, 211000.0, 1.9, 137, 1, 0, 7]. These represent clinical attributes of a patient like age, anaemia, creatinine phosphokinase, and serum sodium, among others. Preprocessing normalized them into the range [0, 1] to obtain the following features: [0.18181818, 1.0, 0.01122735, 0.0, 0.09090909, 0.0, 0.22536065, 0.15730337, 0.68571429, 1.0, 0.0, 0.01067616]. These now-normalized values were encoded onto the quantum circuit using ZFeatureMap, which rotates each qubit by a rotation gate RZ(2×normalized feature value). It starts with the Hadamard gates to set all qubits in superposition, then the rotation gates that perform the encoding. Figure 2 shown the obtained quantum circuit encoding all 12 features of the first row. The described circuit maps normalized data into the quantum Hilbert space in such a way that the Pegasos QSVC algorithm can process high-dimensional representations of data.

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

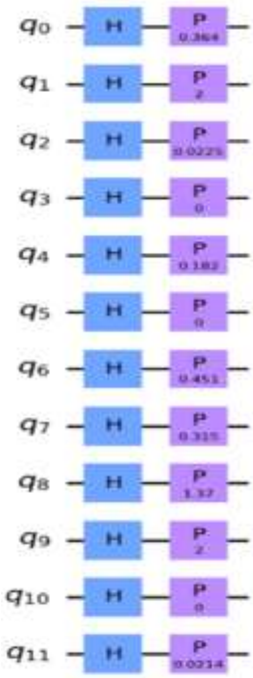

Figure 2. Quantum circuit representation for encoding all 12 features from the first row

8) After generating the quantum circuit for ML algorithm and completing the model training, performance evaluation is carried out using quality metrics such as accuracy, precision, recall, and F1-score.

2.3. Algorithms used

2.3.1. Pegasos Quantum Support Vector Classifier

The Pegasos algorithm is an iterative gradient-based technique for resolving the SVM primal optimization issue. The objective function is minimized by using stochastic gradient descent (SGD), which strikes a balance between regularization and hinge loss. The definition of the fundamental problem is given in (1), as shown below:

$$\min_{w\in\mathbb{R}^s}\left(\frac{\lambda}{2}|w|^2 + \sum_{i=1}^{M} max(0,1-y_i\langle w,\phi(x_i)\rangle)\right) \quad (1)$$

Where the feature map is $(\phi(x_i))$□ the class labels are(y_i), and the regularization parameter is (λ).

Pegasos is appropriate for online and large-scale learning applications since it iteratively updates the weight vector (w) using the gradient of the loss function [20]. This technique works especially well with non-stationary data because it enables ongoing adaptation and unlearning of out-of-date knowledge. The Pegasos QSVC represents an exciting intersection of QC and classical ML. The training complexity of this method is not influenced by the size of the training set and takes advantage of the kernel trick. Therefore, for sufficiently large training sets, the Pegasos QSVC train quicker than QSVC [21]. It encodes classical data into quantum states, computes quantum kernels, and optimizes the separation hyperplane in a quantum-enhanced feature space. Figure 3 shows the algorithm of the proposed approach. The algorithm begins by preprocessing the input dataset df, extracting the target variable y (DEATH_EVENT) and scaling the features using MinMaxScaler. A quantum feature map, specifically the ZFeatureMap, is employed to transform the data into a quantum state space, followed by the creation of a fidelity-based quantum kernel. The PegasosQSVC model is initialized with the quantum kernel, a regularization parameter C=100, and a fixed number of training steps tau=100. The model is trained on the training data and subsequently used to predict labels for the test set. Finally, the algorithm evaluates the model's performance by generating a classification report comparing the predicted labels (predic) with the true labels (y_test). This approach leverages quantum computing principles to enhance the classification task, providing a robust framework for predictive modeling in QML.

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Algorithm X: Procedure for Prediction Using Pegasos QSVC

Inputs: df: Preprocessed input dataset **Output:** Classification report for the PegasosQSVC model

---

```
procedure PREDICT_USING_PEGASOS_QSVC(df)
    y ← df['DEATH_EVENT']
    X ← MinMaxScaler().fit_transform(df)
    X_train, X_test, y_train, y_test ← train_test_split(X, y, test_size=0.2, random_state=1)
    feature_map ← ZFeatureMap(feature_dimension=12, reps=1)
    qkernel ← FidelityQuantumKernel(feature_map=feature_map)
    pegasos_qsvc ← PegasosQSVC(quantum_kernel=qkernel, C=100, num_steps=100)
    pegasos_qsvc.fit(X_train, y_train)
    predic ← pegasos_qsvc.predict(X_test)
    result ← classification_report(y_test, predic)
    print(result)
end procedure
```

Figure 3. Suggested method

2.3.2. Decision Tree

Because DT are simple to understand and have characteristics that are akin to human reasoning, they are frequently used to create categorization models [22]. These models combine to build a structure like a tree, with each leaf node representing a predicted value and each inside node representing a choice based on a particular attribute. DT have a number of benefits, such as their simplicity, their capacity to handle category and numerical data, and their assistance for multi-output issues. DT frequently employ the entropy formula to divide nodes. The entropy (E(D)) of a dataset D for a binary classification problem with classes {0, 1} is computed as in (2):

$$E(D) = -\sum_{i=1}^{n} p_i log_2(p_i) \quad (2)$$

Where p_i represents the proportion of class (i) in the dataset.

2.3.3. Random Forest

By combining many DT, the ensemble learning method known as RF improves performance in both classification and regression. In order to determine the final output, a large number of DT are built during the training phase [23].

2.3.4. Support Vector Classifier

The SVC is a supervised ML algorithm specifically developed for classification purposes. Its main goal is to determine the optimal hyperplane that separates data points from various classes while ensuring the maximum possible margin between them. To achieve this, the SVC model leverages a training dataset consisting of input-output pairs, represented by$\{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$, where $\boldsymbol{x}_i$ denotes the feature vector and $y_i \in \{-1,1\}$ stands for the corresponding class label.

2.3.5. K-Nearest Neighbors

The k-NN algorithm is used for both regression and classification applications is the [24]. When classifying an item, k-NN considers the (k) nearest training samples in a dataset and classifies it based on the majority vote of its neighbors, given a positive integer (k). The following is the conditional probability formula, as shown in (3).

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

$$Pr(Y = j \mid X = x_0) = \frac{1}{k}\sum_{i\in N_0} I(y_i = j) \quad (3)$$

Where $Pr(Y = j \mid X = x_0)$ illustrates the probability that, given that the feature vector X is x_0 and the class label Y is equal to j. N_0: The indices of the k nearest neighbors to the data point x_0 are represented by this set. k: The number of elements in N_0, or the number of nearest neighbors, was taken into consideration. $I(y_i = j)$: The indicator function is equal to 1 if the neighbor's class label (y_i) is equal to j, and 0 otherwise.

2.4. Basic Quantum Gates

2.4.1. Pauli Gates

- Pauli-X Gate (X): The classical NOT gate and this gate are comparable. It allows a qubit's state to flip.

X|0⟩=|1⟩ and X|1⟩=|0⟩

- Pauli-Y Gate (Y): This gate adds a bit flip in addition to a phase flip.

- Pauli-Z Gate (Z): A phase flip is applied via this gate.

2.4.2. Hadamard Gate (H)

Superposition states are produced by this gate. Basis states are changed into equal superpositions by it, as shown in (4).

$$H \mid 0\rangle = \frac{1}{\sqrt{2}}(\mid 0\rangle + \mid 1\rangle) \quad (4)$$

2.4.3. Controlled-NOT Gate (CNOT)

This is two qubits gate that, in the event that the first qubit (control) is |1⟩, flips the second qubit.

2.5. Data encoding and Feature map

Key ideas in Qiskit-based QML are data encoding and feature maps. They explain how classical data can be processed by quantum algorithms since it is incorporated within a quantum state.

2.5.1. data encoding

Data encoding is the conversion of classical data into quantum states. It is necessary to encode classical data so that quantum circuits can comprehend it. Quantum bits or qubits are the building blocks of quantum computers [25].

2.5.2. Feature map

A quantum circuit that encodes conventional data into a quantum state is known as a feature map [26] in the context of QML. Data may be made linearly separable for quantum algorithms by using feature maps to create a higher-dimensional quantum Hilbert space [27].

3. RESULTS AND DISCUSSION

The experiments have been executed on a classical device; the 12-feature HF dataset has been taken for the results of suggested methods. The models are implemented in Python with ML and Qiskit libraries. Scaling techniques have been carried out to normalize the data so that it becomes more feasible for ML and quantum ML models.

3.1. Pegasos Quantum Support Vector Classifier

The binary classification model's performance metrics, which assess the model's capacity to discriminate between two classes denoted as 0 and 1, are presented in the classification report Figure 4. (class 0) exhibits good recall (0.81), F1-score (0.79), and precision (0.77), suggesting that the model is effective at recognizing occurrences of this class. The precision (0.48), recall (0.42), and F1-score (0.45) for (class 1) are noticeably lower, indicating that the model has difficulty with this class. The overall accuracy is (0.69), however as there are more instances of (class 0) than (class 1), the class imbalance may make this figure misleading. The model's performance is more accurately represented by the weighted averages, which take this imbalance into account.

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

	precision	recall	f1-score	support
0	0.77	0.81	0.79	186
1	0.48	0.42	0.45	78
accuracy			0.69	264
macro avg	0.62	0.61	0.62	264
weighted avg	0.68	0.69	0.69	264

Figure 4. Classification report of Pegasos QSVC

3.2. Decision Tree

With a 90% accuracy, the classification model performs admirably as illustrated in Figure 5. While (class 1) has a lower precision of 81%, indicating some potential for improvement, (class 0) has a high precision of 94%, indicating solid predictions. (class 1) has a recall of 86%, which is greater than its precision and indicates that most cases are accurately identified when they occur. This difference is reflected in the f1-scores, where (class 0) is at 93% and (class 1) is at 83%. The macro and weighted averages, which are approximately 89%, show balanced performance in both classes.

	precision	recall	f1-score	support
0	0.94	0.91	0.93	186
1	0.81	0.86	0.83	78
accuracy			0.90	264
macro avg	0.87	0.89	0.88	264
weighted avg	0.90	0.90	0.90	264

Figure 5. Classification report of DT

3.2. Random Forest

Figure 6 shows that the classification model performed well, as evidenced by its high overall accuracy of 93% in the evaluation results. In comparison to (class 1), which has precision of 89%, recall of 86%, and f1-score of 88%, (class 0) has greater precision 94%, recall 96%, and f1-score 95%. This implies that the model predicts (class 0) more accurately than (class 1). This disparity is further highlighted by the support values, which show 186 instances for (class 0) and 78 for (class 1). Resolving this discrepancy could enhance the model's efficacy in practical scenarios where balanced forecasts are crucial.

	precision	recall	f1-score	support
0	0.94	0.96	0.95	186
1	0.89	0.86	0.88	78
accuracy			0.93	264
macro avg	0.92	0.91	0.91	264
weighted avg	0.93	0.93	0.93	264

Figure 6. Classification report RF

3.2. Support Vector Classifier

Figure 7 shows the performance of the classification SVC model. (class 0) has a low recall (0.54) and a high precision (0.93), suggesting that the model is conservative in its prediction of this class and frequently misses real occurrences. (class 1) on the other hand has a high recall (0.90) but low precision (0.45), indicating that while the model correctly detects the majority of instances of this class, it also includes a large number of false positives. A more balanced view is shown by the weighted averages and macros, where the model's inconsistent performance in both classes is reflected in the precision, recall, and F1-scores.

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

	precision	recall	f1-score	support
0	0.93	0.54	0.68	186
1	0.45	0.90	0.60	78
accuracy			0.65	264
macro avg	0.69	0.72	0.64	264
weighted avg	0.79	0.65	0.66	264

Figure 7. Classification report of SVC

3.2. K-Nearest Neighbors

The performance measures of the classification model in Figure 8 show a notable imbalance in its capacity to distinguish between the two classes. The model performs well for (class 0), with high recall (0.94), f1-score (0.85), and precision (0.78). (class 1), on the other hand, shows a decline in precision to (0.72), recall to (0.36), and f1-score to (0.48), indicating that the model has difficulty with this class. Although the accuracy is (0.77) overall, the difference in the classes suggests that there may be a problem with the feature representation or data imbalance.

	precision	recall	f1-score	support
0	0.78	0.94	0.85	186
1	0.72	0.36	0.48	78
accuracy			0.77	264
macro avg	0.75	0.65	0.67	264
weighted avg	0.76	0.77	0.74	264

Figure 8. Classification report of KNN

This study used a real-world clinical dataset to compare the Pegasos QSVC with traditional ML algorithms such as DT, RF, SVM and KNN for the prediction of HF. In terms of predictive accuracy, the results showed that classical models performed better than Pegasos QSVC in most cases with DT and RF showing the best results. In contrast to SVM Pegasos QSVC demonstrated competitive performance suggesting that quantum algorithms may be able to compete with classical methods in some situations even when simulated on conventional hardware. The performance disparity between classes suggests that class imbalance may have influenced the results, with the model performing better on the majority class. These findings highlight the complementary strengths of quantum and classical approaches, with classical models excelling in accuracy and robustness, while quantum ML offers promising avenues for future exploration. This study underscores the transformative potential of quantum ML in healthcare predictive modeling and emphasizes the need for further research to address current limitations, such as improving quantum hardware and developing hybrid quantum-classical frameworks.

4. CONCLUSION

In this work, we used a real-world dataset to compare the performance of QML algorithms, namely the Pegasos QSVC, with classical ML algorithms—DT, RF, SVC, and KNN—for the classification of HF. The models were assessed for precision, recall, and overall accuracy using Qiskit, Python, and feature mapping approaches. The results demonstrate the complementary advantages of quantum and classical methods with QML providing exciting new directions for further research while classical models excel in accuracy and robustness. The transformative potential of QML in healthcare predictive modeling is highlighted in this study along with the need for additional research to overcome existing constraints. These include developing hybrid quantum-classical frameworks optimizing feature encoding and improving quantum hardware in order to fully realize the potential of quantum-enhanced ML. We will expand the suggested methodology in subsequent research to evaluate Pegasos QSVC using datasets with over 12 features further investigating its scalability and high-dimensional performance.

REFERENCES

[1] J. Preskill, "Quantum Computing in the NISQ era and beyond," Quantum, vol. 2, p. 79, Aug. 2018, doi: 10.22331/q-2018-08-06-79.
[2] B. Basha, "Enhancing Healthcare Data Security Using Quantum Cryptography for Efficient and Robust Encryption," Journal of Electrical Systems, vol. 20, pp. 1993–2000, Apr. 2024, doi: 10.52783/jes.2535.
[3] V. P. Gopal, C. N, H. S, M. Padhma, and P. Suhail, "Implement Quantum Machine Learning Classifier using MNIST Dataset," IJARCCE, vol. 13, Apr. 2024, doi: 10.17148/IJARCCE.2024.134146.
[4] N. Jeyaraman, M. Jeyaraman, S. Yadav, S. Ramasubramanian, and S. Balaji, "Revolutionizing Healthcare: The Emerging Role of Quantum Computing in Enhancing Medical Technology and Treatment," Cureus, vol. 16, no. 8, p. e67486, Aug. 2024, doi: 10.7759/cureus.67486.
[5] Gadi Aleksandrowicz, « Qiskit: An Open-source Framework for Quantum Computing ». Zenodo, janv. 23, 2019. doi: 10.5281/zenodo.2562111.
[6] D. S. Watson, "Interpretable machine learning for genomics," Human Genetics, vol. 141, no. 9, pp. 1499-1513, Sep. 2022. doi: 10.1007/s00439-021-02387-9.
[7] G. Nguyen, S. Dlugolinsky, M. Bobák, V. Tran, Á. L. García, I. Heredia, P. Malík, and L. Hluchý, "Machine Learning and Deep Learning frameworks and libraries for large-scale data mining: a survey," Artificial Intelligence Review, vol. 52, no. 1, pp. 77-124, Jun. 2019. doi: 10.1007/s10462-018-09679-z.
[8] M. Benedetti, E. Lloyd, S. Sack, and M. Fiorentini, "Parameterized quantum circuits as machine learning models," Quantum Science and Technology, vol. 4, no. 4, p. 043001, Nov. 2019, doi: 10.1088/2058-9565/ab4eb5.
[9] U. Ullah and B. Zapirain, "Quantum Machine Learning Revolution in Healthcare: A Systematic Review of Emerging Perspectives and Applications," IEEE Access, vol. PP, pp. 1–1, Jan. 2024, doi: 10.1109/ACCESS.2024.3353461.
[10] L. Wei et al., "Quantum machine learning in medical image analysis: A survey," Neurocomputing, vol. 525, pp. 42–53, 2023, doi: 10.1016/j.neucom.2023.01.049.
[11] R. Ur Rasool, H. F. Ahmad, W. Rafique, A. Qayyum, J. Qadir, and Z. Anwar, "Quantum Computing for Healthcare: A Review," Future Internet, vol. 15, no. 3, p. 94, Mar. 2023, doi: 10.3390/fi15030094.
[12] M. Javaid, A. Haleem, and R. P. Singh, "Health informatics to enhance the healthcare industry's culture: An extensive analysis of its features, contributions, applications and limitations," Informatics and Health, vol. 1, no. 2, pp. 123–148, 2024, doi: 10.1016/j.infoh.2024.05.001.
[13] World Heart Federation, "Heart Failure," available at: https://world-heart-federation.org/what-we-do/heart-failure/. Accessed: Dec. 28, 2024.
[14] F. Alotaibi, "Implementation of Machine Learning Model to Predict Heart Failure Disease," International Journal of Advanced Computer Science and Applications, vol. 10, Jan. 2019. doi: 10.14569/IJACSA.2019.0100637.
[15] S. Alsubai, A. Alqahtani, A. Binbusayyis, M. Sha, A. Gumaei, and S. Wang, "Heart Failure Detection Using Instance Quantum Circuit Approach and Traditional Predictive Analysis," Mathematics, vol. 11, Mar. 2023. doi: 10.3390/math11061467.
[16] S. Prabhu, S. Gupta, G. Prabhu, A. Dhanuka, and K. Bhat, "QuCardio: Application of Quantum Machine Learning for Detection of Cardiovascular Diseases," IEEE Access, vol. PP, Jan. 2023, pp. 1-1. doi: 10.1109/ACCESS.2023.3338145.
[17] R. Bhavsar, N. Jadav, U. Bodkhe, R. Gupta, S. Tanwar, G. Sharma, P. Bokoro, and R. Sharma, "Classification of Potentially Hazardous Asteroids Using Supervised Quantum Machine Learning," IEEE Access, vol. PP, May 2023. doi: 10.1109/ACCESS.2023.3297498.
[18] N. Singh and S. R. Pokhrel, "An Independent Implementation of Quantum Machine Learning Algorithms in Qiskit for Genomic Data," arXiv preprint arXiv:2405.09781, May 2024.
[19] Kaggle, "Heart Failure Prediction Clinical Records," Kaggle, available at: https://www.kaggle.com/datasets/aadarshvelu/heart-failure-prediction-clinical-records. Accessed: Dec. 28,

Journal of Information Systems Engineering and Management
2025, 10(46s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/
Research Article

2024.

[20] S. Shalev-Shwartz, Y. Singer, and N. Srebro, "Pegasos: Primal Estimated Sub-GrAdient sOlver for SVM," in ACM International Conference Proceeding Series, vol. 227, Jun. 2007, pp. 807-814. doi: 10.1145/1273496.1273598.

[21] Qiskit Community, "Pegasos QSVC: Primal Estimated Sub-GrAdient Solver for Quantum SVM," available at: https://qiskit-community.github.io/qiskit-machine-learning/tutorials/07_pegasos_qsvc.html. Accessed: Dec. 28, 2024.

[22] S. B. Kotsiantis, "Decision trees: a recent overview," Artificial Intelligence Review, vol. 39, no. 4, pp. 261-283, Apr. 2013. doi: 10.1007/s10462-011-9272-4.

[23] G. Louppe, "Understanding Random Forests: From Theory to Practice," Ph.D. dissertation, Oct. 2014. doi: 10.13140/2.1.1570.5928.

[24] K. P. Murphy, Machine Learning: A Probabilistic Perspective. Cambridge, MA: MIT Press, 2012.

[25] M. Rath and H. Date, "Quantum Data Encoding: A Comparative Analysis of Classical-to-Quantum Mapping Techniques and Their Impact on Machine Learning Accuracy," arXiv preprint arXiv:2311.10375, 2023.Q. Song, H. Ge, J. Caverlee, and X. Hu, “Tensor completion algorithms in big data analytics,” *arXiv*, vol. 13, no. 1, 2017.

[26] M. Schuld and N. Killoran, "Quantum Machine Learning in Feature Hilbert Spaces," Phys. Rev. Lett., vol. 122, no. 4, p. 040504, Feb. 2019, doi: 10.1103/PhysRevLett.122.040504.

[27] B.-S. Chen and J.-L. Chern, Genetically Auto-Generated Quantum Feature Maps. 2022.