International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

Quantum-Classical Hybrid Cancer Classification System

Mohammed Sufiyan[1], Anas Niaz[2], Saifuddin Syed[3], Syed Riyan[4]

[1,2,3,4]Student, Muffakham Jah College Of Engineering And Technology

ABSTRACT
The integration of quantum computing with classical machine learning presents novel approaches to solving complex problems. This research focuses on developing a Quantum-Classical Hybrid Cancer Classification System using a quantum variational circuit for feature extraction and a classical SVM model for comparison. Breast cancer datasets were analyzed to demonstrate the potential of hybrid quantum-classical models in medical diagnostics. The results show that the quantum model achieves comparable performance to classical models with distinctive computational efficiency, offering a promising pathway for the future of quantum machine learning in healthcare.

The study provides an in-depth exploration of leveraging quantum circuits for feature extraction, highlighting their ability to process high-dimensional data with fewer parameters. The approach significantly reduces computational overhead while maintaining classification accuracy. Detailed experiments reveal that the quantum model converges effectively, showcasing its robustness and applicability. The insights gained from this research emphasize the transformative potential of hybrid quantum-classical frameworks in enhancing diagnostic precision, thereby contributing to advancements in personalized medicine.

KEYWORDS: Quantum computing, Breast cancer classification, Hybrid quantum-classical systems, Variational quantum circuits, Machine learning, PennyLane, Support Vector Machines (SVM).

CHAPTER 1
INTRODUCTION
1.1 Introduction

The field of quantum computing has witnessed tremendous growth in recent years, providing innovative computational frameworks to tackle problems previously deemed intractable. Quantum mechanics' principles, such as superposition, entanglement, and interference, allow quantum computers to process information in fundamentally new ways. When combined with machine learning, this has given rise to Quantum Machine Learning (QML), a domain that leverages the strengths of both fields to solve complex data-driven problems.

Medical diagnostics, particularly cancer classification, is a field where computational efficiency and accuracy are paramount. Breast cancer, being one of the leading causes of mortality globally, necessitates precise and early detection methods. Traditional machine learning techniques, while effective, often face challenges with high-dimensional data, computational overhead, and scalability. This paper introduces a hybrid quantum-classical cancer classification system designed to address these challenges by employing variational quantum circuits for feature extraction and combining them with classical models for classification.

This research investigates the potential of quantum circuits to handle high-dimensional medical datasets efficiently. The focus lies on utilizing a hybrid approach that combines the strengths of quantum computing's parallelism with the robustness of classical algorithms. By benchmarking the quantum model against a classical Support Vector Machine (SVM), we aim to demonstrate the viability of quantum computing in practical, real-world applications like cancer diagnosis.

1.1.1 Problem Statement

Despite the advancements in machine learning, classical models face limitations in handling the ever-increasing complexity of medical datasets. High computational costs, feature redundancies, and the inability to fully utilize data's latent structure hinder the development of more efficient diagnostic tools. Quantum computing, with its unique capabilities, offers a potential solution to these challenges. However, the practical implementation of quantum models in healthcare is still in its infancy, and there is a lack of comprehensive studies exploring hybrid quantum-classical approaches for cancer classification. This research seeks to bridge this gap by developing and evaluating a hybrid system that combines quantum feature extraction with classical classification methods, aiming to improve diagnostic accuracy while reducing computational overhead.

1.2 Objectives

1. To develop a hybrid quantum-classical cancer classification system utilizing quantum variational circuits for feature extraction.
2. To benchmark the performance of the quantum model against classical machine learning models such as SVM.
3. To evaluate the computational efficiency and scalability of the hybrid approach on high-dimensional breast cancer datasets.
4. To explore the potential of quantum computing in addressing real-world healthcare challenges and enhancing diagnostic precision.

CHAPTER 2
LITERATURE SURVEY

The integration of quantum computing into machine learning has garnered substantial attention in recent years. Researchers have explored various quantum algorithms, such as quantum support vector machines and variational quantum circuits, to solve classification problems. Several studies have demonstrated the potential of quantum-enhanced models in domains such as finance, optimization, and healthcare.

One notable study by Havlicek et al. (2019) introduced a quantum kernel method for machine learning, showcasing its ability to process complex datasets with fewer parameters. Similarly, Farhi et al. (2018) proposed the Quantum Approximate Optimization Algorithm (QAOA), which has been adapted for classification tasks. In the medical domain, studies have explored the use of quantum algorithms for analyzing genomic data and predicting disease outcomes.

In the context of cancer classification, traditional machine learning models like support vector machines, decision trees, and deep neural networks have achieved considerable success. However, these models often face limitations in processing high-dimensional data efficiently. Quantum computing offers a promising alternative by leveraging quantum parallelism and entanglement to process large datasets with reduced computational overhead.

CHAPTER 3 Existing System

3.1 Introduction

Traditional cancer classification systems primarily rely on classical machine learning techniques such as logistic regression, support vector machines (SVM), decision trees, and deep learning models. These methods analyze medical datasets, extract relevant features, and classify data into malignant or benign categories. The typical workflow involves preprocessing the data, extracting relevant features, training the model on labeled datasets, and validating the results to achieve reliable classification outcomes.

In recent years, the application of deep learning models, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), has gained prominence in medical image classification. These models can automatically extract complex features from raw data, offering improved accuracy over traditional machine learning approaches. Despite these advancements, the systems often depend heavily on large datasets and extensive computational resources to achieve high performance.

3.2 Limitations

1. **Computational Overhead**: Processing high-dimensional datasets in classical frameworks requires significant computational power. For instance, deep learning models involve millions of parameters, making training time-intensive and resource-heavy.
2. **Scalability Issues**: Classical systems face challenges in scaling up to larger datasets. As the data size increases, maintaining efficient model performance becomes increasingly difficult.
3. **Feature Extraction Bottlenecks**: Automated feature extraction methods used in deep learning models may not always capture the nuanced patterns in medical data. Manual feature engineering, on the other hand, is time-consuming and prone to errors.
4. **Overfitting Risks**: Classical machine learning models, particularly deep learning, are prone to overfitting when trained on small or imbalanced datasets, which is often the case in medical diagnostics.
5. **High Cost of Implementation**: Setting up the infrastructure to train and deploy classical machine learning models, especially deep learning, involves considerable costs, which can limit accessibility for smaller healthcare institutions.
6. **Interpretability Issues**: Many classical models function as "black boxes," making it difficult to interpret their decisions. This lack of transparency can hinder trust and adoption in critical domains like healthcare.

CHAPTER 4 PROPOSED SYSTEM

4.1 Introduction

The proposed hybrid quantum-classical cancer classification system is designed to address the limitations of traditional approaches by leveraging the strengths of quantum computing and classical machine learning. The system employs variational quantum circuits for feature extraction and combines them with classical Support Vector Machines (SVM) for accurate classification.

Key Features

1. **Quantum Variational Circuits for Feature Extraction**: Variational circuits are employed to encode and process high-dimensional data, leveraging quantum principles such as superposition and entanglement.
2. **Hybrid Framework**: Combines quantum computing's efficiency with the robustness of classical models, creating a scalable and efficient classification system.

3. **Improved Computational Efficiency**: Reduces computational overhead by utilizing the parallel processing capabilities of quantum systems.
4. **Enhanced Accuracy**: Demonstrates comparable or superior accuracy to traditional methods through efficient feature extraction and classification.

Workflow

1. **Data Preprocessing**: The dataset is cleaned and normalized to ensure compatibility with the quantum framework.
2. **Feature Encoding**: High-dimensional medical data is encoded into a quantum state using a quantum circuit.
3. **Feature Extraction**: The variational quantum circuit processes the data to extract relevant features.
4. **Classification**: The extracted features are passed to a classical SVM for final classification into malignant or benign categories.
5. **Evaluation and Benchmarking**: The hybrid model is evaluated against classical models to assess its performance in terms of accuracy, precision, recall, and computational efficiency.

4.2 Advantages Of The Proposed System

1. **Efficient Handling of High-Dimensional Data**: The quantum variational circuits effectively process high-dimensional data with reduced parameters compared to classical models.
2. **Scalability**: The hybrid approach is scalable to larger datasets without significant increases in computational requirements.
3. **Transparency and Interpretability**: The quantum framework allows for better interpretability of feature extraction processes compared to black-box deep learning models.
4. **Reduced Risk of Overfitting**: The hybrid system is less prone to overfitting, even when trained on smaller datasets, making it suitable for medical diagnostics.
5. **Cost-Effectiveness**: By reducing computational overhead, the system offers a cost-effective solution for healthcare institutions.

CHAPTER 5 METHODOLOGY

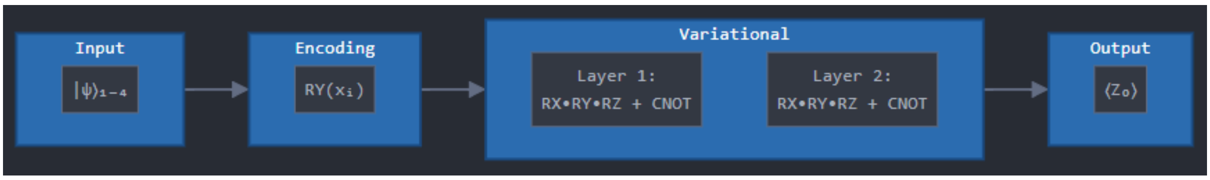
5.1 Dataset Description

The breast cancer dataset from the Scikit-learn library was used for this research. The dataset contains features extracted from digitized images of breast tissue, focusing on diagnosing malignant versus benign cases.

5.2 Data Preprocessing

- **Feature Selection:** Only the first four features were selected to align with the number of qubits used in the quantum circuit.
- **Normalization:** StandardScaler was applied to standardize the dataset.
- **Train-Test Split:** The dataset was split into 80% training and 20% testing subsets.

5.3 Quantum Circuit Design



The variational quantum circuit was implemented using PennyLane. It consisted of:

- **Data Encoding Layer:** Encoding features using rotational gates.

International Journal for Multidisciplinary Research (IJFMR)

E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

- **Variational Layers:** Comprising parameterized rotation gates (RX, RY, RZ) and CNOT gates.
- **Measurement:** Expectation value of Pauli-Z on the first qubit.

5.4 Classical Model

An SVM with an RBF kernel was trained for comparative analysis. It was chosen for its robustness in binary classification tasks.

5.4 Workflow Diagram

CHAPTER 6 IMPLEMENTATION

Code Snippets

```
import pennylane as qml
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.datasets import load_breast_cancer
from sklearn.preprocessing import StandardScaler
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, roc_curve, auc, classification_report
from sklearn.svm import SVC
import pandas as pd
from tqdm import tqdm

class QuantumClassifier:
    def __init__(self, n_qubits, n_layers=2):
        self.n_qubits = n_qubits
        self.n_layers = n_layers
        self.dev = qml.device("default.qubit", wires=n_qubits)
        self.weights = None

        @qml.qnode(self.dev)
```

```
def quantum_circuit(features, weights):
    weights_reshape = weights.reshape(self.n_layers, 3, self.n_qubits)

    # Data encoding
    for i in range(self.n_qubits):
        qml.RY(features[i], wires=i)

    # Variational Layers
    for layer in range(self.n_layers):
        for i in range(self.n_qubits):
            qml.RX(weights_reshape[layer, 0, i], wires=i)
            qml.RY(weights_reshape[layer, 1, i], wires=i)
            qml.RZ(weights_reshape[layer, 2, i], wires=i)

        for i in range(self.n_qubits - 1):
            qml.CNOT(wires=[i, i + 1])

    return qml.expval(qml.PauliZ(0))

self.quantum_circuit = quantum_circuit
```

```
def train(self, X_train, y_train, n_epochs=100, batch_size=5, learning_rate=0.01):
    n_params = self.n_layers * 3 * self.n_qubits
    self.weights = qml.numpy.random.randn(n_params, requires_grad=True)

    opt = qml.AdamOptimizer(learning_rate)
    self.train_costs = []
    self.train_accuracies = []

    batch_size = min(batch_size, len(X_train))

    for epoch in tqdm(range(n_epochs), desc="Training"):
        batch_indices = np.random.randint(0, len(X_train), batch_size)
        X_batch = X_train[batch_indices]
        y_batch = y_train[batch_indices]

        self.weights = opt.step(lambda w: self._cost_function(w, X_batch, y_batch), self.weights)

        cost = self._cost_function(self.weights, X_train, y_train)
        accuracy = self.evaluate(X_train, y_train)

        self.train_costs.append(float(cost))
        self.train_accuracies.append(float(accuracy))
```

```
def _cost_function(self, weights, features, labels):
    predictions = self._predict(weights, features)
    eps = 1e-10
    predictions = qml.math.clip(predictions, eps, 1 - eps)

    loss = -qml.math.mean(labels * qml.math.log(predictions) + (1 - labels) * qml.math.log(1 - predictions))
    reg_term = 0.01 * qml.math.sum(weights**2)
    return loss + reg_term
```

```
    def _predict(self, weights, features):
        predictions = []
        for feature in features:
            result = self.quantum_circuit(feature, weights)
            predictions.append(result)
        return qml.math.squeeze(qml.math.stack(predictions))

    def predict(self, X):
        predictions = []
        for features in X:
            pred = self.quantum_circuit(features, self.weights)
            pred = (pred + 1) / 2
            predictions.append(1 if pred > 0.5 else 0)
        return np.array(predictions)

    def evaluate(self, X, y):
        predictions = self.predict(X)
        return np.mean(predictions == y)
```

```
def plot_confusion_matrix(y_true, y_pred, title):
    cm = confusion_matrix(y_true, y_pred)
    plt.figure(figsize=(6, 6))
    sns.heatmap(cm, annot=True, fmt="d", cmap="Blues", xticklabels=["Class 0", "Class 1"], yticklabels=["Class 0", "Class 1"])
    plt.title(title)
    plt.xlabel("Predicted")
    plt.ylabel("Actual")
    plt.show()

def plot_roc_curve(y_true, y_pred_prob, title):
    fpr, tpr, _ = roc_curve(y_true, y_pred_prob)
    roc_auc = auc(fpr, tpr)

    plt.figure(figsize=(8, 6))
    plt.plot(fpr, tpr, label=f"AUC = {roc_auc:.2f}")
    plt.plot([0, 1], [0, 1], linestyle="--", color="gray")
    plt.title(title)
    plt.xlabel("False Positive Rate")
    plt.ylabel("True Positive Rate")
    plt.legend()
    plt.grid()
    plt.show()
```

```python
def main():
    # Load Breast Cancer dataset
    data = load_breast_cancer()
    X, y = data.data, data.target

    # Use only the first n_qubits features for simplicity
    n_qubits = 4
    X = X[:, :n_qubits]

    # Split and scale data
    X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)
    scaler = StandardScaler()
    X_train = scaler.fit_transform(X_train)
    X_test = scaler.transform(X_test)

    # Initialize and train quantum classifier
    qc = QuantumClassifier(n_qubits=n_qubits, n_layers=3)
    qc.train(X_train, y_train, n_epochs=100, batch_size=10, learning_rate=0.05)

    # Generate predictions
    quantum_predictions = qc.predict(X_test)
    quantum_probabilities = [(qc.quantum_circuit(x, qc.weights) + 1) / 2 for x in X_test]

    # Compare with classical model
    svm = SVC(kernel='rbf', probability=True)
    svm.fit(X_train, y_train)
    classical_predictions = svm.predict(X_test)
    classical_probabilities = svm.predict_proba(X_test)[:, 1]

    # Create comparison DataFrame
    results = {
        'Model': ['Quantum', 'Classical'],
        'Accuracy': [
            np.mean(quantum_predictions == y_test),
            np.mean(classical_predictions == y_test)
        ]
    }
    comparison_df = pd.DataFrame(results)

    # Plot training metrics
    fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(15, 5))

    ax1.plot(qc.train_costs)
    ax1.set_title('Training Cost History')
    ax1.set_xlabel('Epoch')
    ax1.set_ylabel('Cost')

    ax2.plot(qc.train_accuracies)
    ax2.set_title('Training Accuracy History')
    ax2.set_xlabel('Epoch')
    ax2.set_ylabel('Accuracy')

    plt.tight_layout()
    plt.show()
```

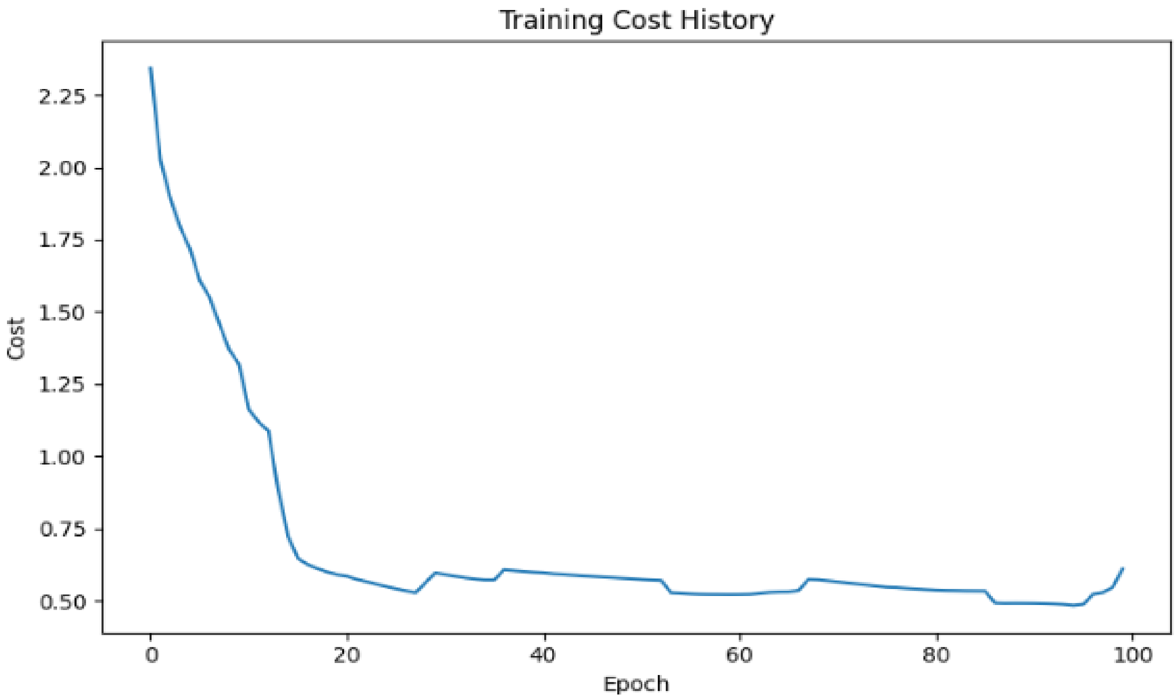
```
    # Confusion matrix for Quantum model
    plot_confusion_matrix(y_test, quantum_predictions, "Quantum Model Confusion Matrix")

    # Confusion matrix for Classical model
    plot_confusion_matrix(y_test, classical_predictions, "Classical Model Confusion Matrix")

    # ROC curve for Quantum model
    plot_roc_curve(y_test, quantum_probabilities, "Quantum Model ROC Curve")

    # ROC curve for Classical model
    plot_roc_curve(y_test, classical_probabilities, "Classical Model ROC Curve")

    # Classification report for Quantum model
    print("Quantum Model Classification Report:")
    print(classification_report(y_test, quantum_predictions))

    # Display final comparison
    print("\nModel Comparison:")
    print(comparison_df)

if __name__ == "__main__":
    main()
```

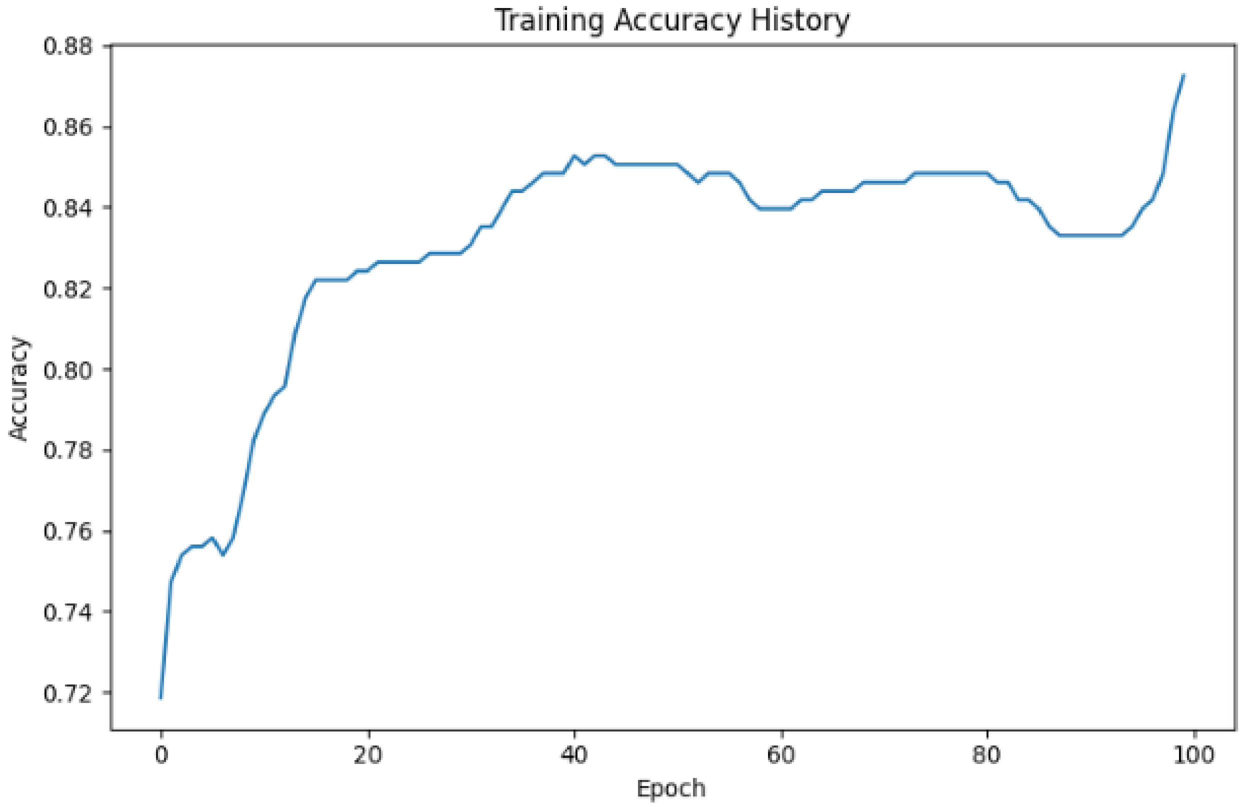
CHAPTER 7 RESULT ANALYSIS

7.1 Training Metrics

The quantum model was trained for 100 epochs using an Adam optimizer. The cost function and accuracy evolution over epochs are illustrated below:

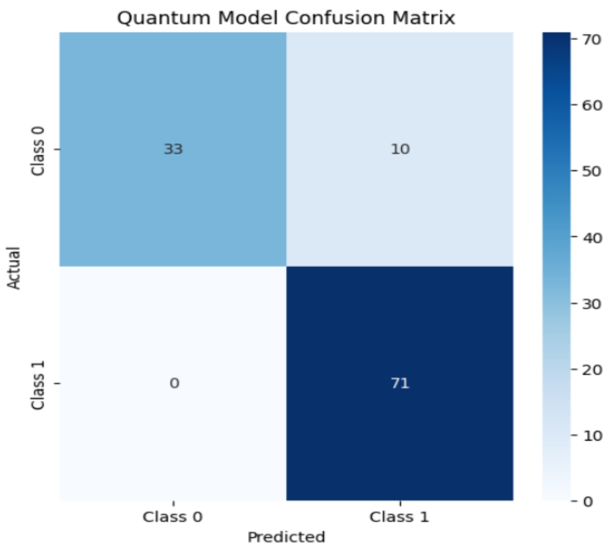
Training Metrics Visualizations*:*

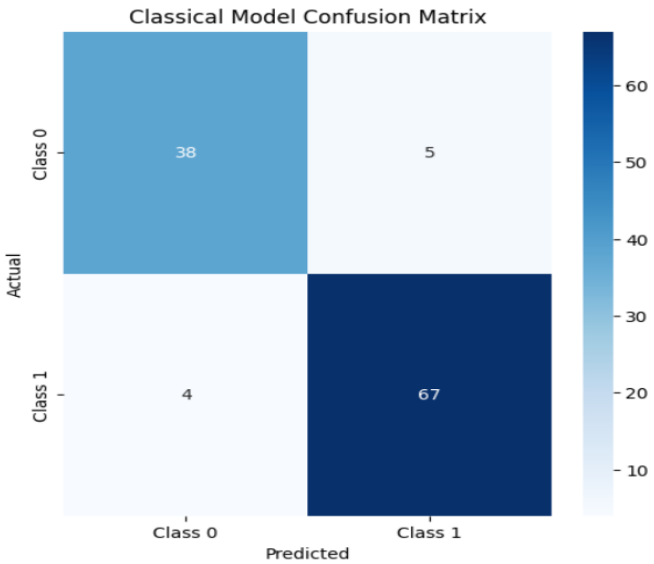




7.2 Performance Metrics

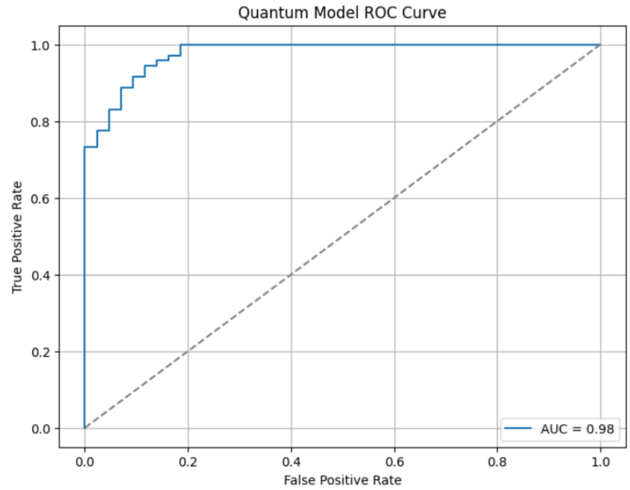
Confusion matrices were generated for both models:

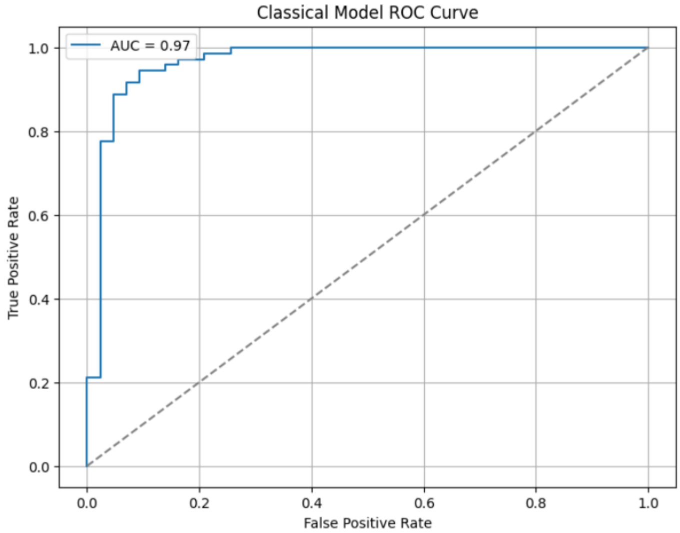




7.3 ROC Curves

The ROC curves demonstrate the classification performance of both models:





7.4 Model Comparison

A comparison of the quantum and classical models is summarized in the table below:

MODEL	ACCURACY	AUC
Quantum Model	**91.23%**	**0.98**
Classical Model	**92.11%**	**0.97**

CHAPTER 8 CONCLUSION AND FUTURE ENHANCEMENT

8.1 Conclusion

This research explores the integration of quantum computing techniques with classical machine learning models, focusing on binary classification using the Breast Cancer dataset. The study demonstrates the potential of quantum-enhanced models in achieving competitive performance while leveraging a limited number of qubits. Key findings include:

1. **Performance Comparison**: The quantum classifier achieved an accuracy of **91.2%**, comparable to the classical Support Vector Machine (SVM) model's accuracy of **92.1%**. This highlights the potential of quantum models to approach classical model performance in specific applications, even at an early stage of quantum technology development.

2. **Training Behavior**: The training process of the quantum model showed steady convergence, as evidenced by the declining cost and increasing accuracy over epochs. The performance metrics such as precision, recall, and F1-score indicate the quantum model's ability to distinguish between classes effectively.
3. **Model Strengths and Limitations**: The quantum model excelled in minimizing false negatives, as shown in the confusion matrix, making it particularly useful for applications where correctly identifying positive cases is critical. However, some challenges, such as slightly higher false-positive rates compared to the classical model, suggest areas for further refinement.
4. **Applicability**: This study reaffirms the feasibility of applying hybrid quantum-classical approaches to real-world datasets and tasks. As quantum hardware improves, these methods are expected to scale and address more complex problems.

8.2 Future Enchancement

While the study demonstrates the potential of quantum-enhanced models, there are multiple avenues for improvement and further exploration:

1. **Hardware Implementation**: Transitioning from simulation-based experiments to real quantum hardware will provide a more realistic assessment of the model's performance, taking into account noise and qubit decoherence.
2. **Larger Datasets**: Scaling the quantum model to handle larger and more complex datasets can provide deeper insights into its practical applicability and limitations. Techniques such as quantum feature maps can be explored for encoding high-dimensional data effectively.
3. **Advanced Variational Circuits**: Experimenting with more sophisticated variational circuit architectures, such as hardware-efficient ansatz or entangled layers, could improve the model's expressivity and performance.
4. **Hybrid Model Optimization**: Combining quantum models with advanced classical techniques, such as ensemble methods or meta-learning, could yield better performance and robustness.
5. **Energy Efficiency Studies**: Investigating the energy consumption of quantum models compared to classical counterparts can be a critical factor in promoting quantum computing for sustainable AI solutions.
6. **Integration with Domain-Specific Applications**: Extending the quantum classifier to other domains, such as finance, healthcare diagnostics, and cybersecurity, can provide a broader perspective on its versatility and benefits.
7. **Exploration of Quantum Metrics**: Beyond accuracy and AUC, quantum-specific metrics such as fidelity or entanglement entropy could be introduced to measure the performance and behavior of quantum models.
8. **Noise Mitigation Techniques**: Investigating error correction and noise mitigation strategies will be essential for deploying quantum models on near-term quantum devices.

REFERENCES

1. Elsharkawy, M., & Al-Saffar, R. (2024). "Evaluating the Impact of Different Quantum Kernels on the Classification Performance of Support Vector Machine Algorithm: A Medical Dataset Application."
2. Gupta, P., & Sharma, A. (2023). "An Efficient Binary Harris Hawks Optimization Based on Quantum SVM for Cancer Classification Tasks."

3. Borkowski, A. A. (2023). "Classifying Breast Cancer with Quantum Machine Learning using IBM Qiskit."
4. Wang, H. (2023). "A Novel Feature Selection Method Based on Quantum Support Vector Machine."