

Research Paper

A Quantum Support Vector Machine-Based Approach for Customer Churn Prediction

P. Ratna Kumari¹, S. Mohan Krishna²

**¹Assistant Professor (CSE), Sanketika Vidya Parishad Engineering College,
Vishakhapatnam, Andhra Pradesh, India**

**²Student, M-Tech (CST) Final Semester, Sanketika Vidya Parishad Engineering College,
Vishakhapatnam, Andhra Pradesh, India**

ABSTRACT

Customer churn prediction is important in the banking sector for improving customer retention and reducing revenue loss. Traditional Support Vector Machines (SVM) face scalability challenges when processing high-dimensional and non-linear customer data. This study proposes a Quantum Support Vector Machine (QSVM) for churn prediction by combining classical preprocessing with quantum machine learning. The data is cleaned, encoded, scaled, and reduced using Principal Component Analysis (PCA). The resulting features are encoded into quantum states, and a quantum kernel is used for classification. The model is implemented using Python and Qiskit and evaluated on the Qiskit Aer simulator. Performance is assessed using accuracy, precision, recall, and F1-score. The proposed QSVM achieves approximately 80% accuracy, demonstrating strong non-churn prediction and moderate churn identification. The results indicate the potential of quantum machine learning for banking churn prediction.

Keywords : Customer Churn Prediction, Quantum Support Vector Machine (QSVM), Quantum Machine Learning, Banking, Principal Component Analysis (PCA), Quantum Kernel, Qiskit, Qiskit Aer, Machine Learning, Customer Retention

INTRODUCTION

Customer churn prediction is an important task in the banking sector, as retaining existing customers helps reduce financial losses and improve long-term profitability. Traditional machine learning methods, including Support Vector Machines (SVM), face challenges when handling high-dimensional and complex customer data. To address these limitations, this study proposes a Quantum Support Vector Machine (QSVM) for customer churn prediction. The proposed approach combines classical preprocessing techniques, including Principal Component Analysis (PCA), with quantum feature mapping and quantum kernel computation. The model is implemented using Python and Qiskit and evaluated using the Qiskit Aer simulator. Experimental results show that the proposed QSVM achieves approximately 80% classification accuracy, demonstrating the feasibility of quantum machine learning for banking churn prediction and its potential for future applications in complex customer behavior analysis.

A. Problem Statement

Customer churn is a significant challenge in the banking sector, where accurately identifying customers likely to discontinue services is important for customer retention. Traditional

methods such as Support Vector Machines (SVM) face difficulties with high-dimensional and complex customer data due to expensive kernel computation, scalability issues, and limited representation of non-linear relationships. These limitations can reduce the effectiveness of churn prediction when processing large and complex datasets. This project proposes a Quantum Support Vector Machine (QSVM) approach that combines classical preprocessing, PCA-based dimensionality reduction, quantum feature mapping, and quantum kernel computation to improve customer churn classification.

B. Purpose

The primary purpose of this project is to develop a Quantum Support Vector Machine (QSVM) model for customer churn prediction in the banking sector. The study aims to address the limitations of classical SVM by applying quantum computing techniques to complex and high-dimensional customer data. PCA is used to reduce the number of features, while quantum feature mapping and quantum kernels are applied to represent and classify the processed data. The proposed model is implemented using Qiskit and evaluated using a quantum simulator with metrics such as accuracy, precision, recall, and F1-score.

C. Overview

This project presents a hybrid classical-quantum approach for predicting customer churn using a Quantum Support Vector Machine. The system first preprocesses customer data and applies feature scaling and Principal Component Analysis (PCA) to reduce dimensionality. The reduced features are then encoded into quantum states using quantum feature mapping techniques such as ZZFeatureMap. A quantum kernel is computed to measure similarity between customer data points in the quantum feature space. The QSVM classifier uses this quantum kernel to classify customers into churn and non-churn categories. Qiskit Aer is used to simulate the quantum computations without requiring real quantum hardware. The model is evaluated using accuracy, precision, recall, and F1-score to examine its classification performance.

II. DESIGN

A. System Design

The proposed system is designed to predict customer churn using a hybrid classical-quantum approach. The system processes customer data through data preprocessing, dimensionality reduction, quantum feature mapping, quantum kernel computation, and QSVM classification. PCA is applied to reduce the number of features to match the available qubits, while Qiskit Aer is used to execute the quantum computations through simulation. The system is organized into modular layers to support data processing, classification, evaluation, and integration between classical and quantum components.

B. System Architecture

The system architecture consists of input, preprocessing, dimensionality reduction, quantum feature encoding, quantum kernel, QSVM classifier, evaluation, and simulation layers. The input layer receives the customer dataset, while the preprocessing layer removes irrelevant attributes, encodes categorical features, and applies feature scaling. PCA reduces the feature dimensions before the data is mapped into quantum states using ZZFeatureMap. The quantum kernel computes similarity between quantum-encoded data points, and the QSVM classifier

predicts churn or non-churn customers. Qiskit Aer provides the simulation environment for the quantum operations.

C. System Modules

The system contains modules for customer data input, preprocessing, PCA-based dimensionality reduction, quantum feature mapping, quantum kernel computation, QSVM classification, evaluation, and simulation. The preprocessing module prepares the customer data for further processing, while PCA reduces the dimensionality of the dataset. The quantum feature mapping module encodes the reduced features into quantum states, and the quantum kernel module calculates similarity between data points. The QSVM module performs classification, and the evaluation module measures performance using accuracy, precision, recall, and F1-score.

D. Data Flow

The data flow begins with the customer dataset and proceeds through classical preprocessing and dimensionality reduction. The processed features are then encoded into quantum states using quantum feature mapping, followed by quantum kernel computation. The resulting quantum kernel is used by the QSVM classifier to classify customers into churn and non-churn categories. The predictions are evaluated using classification metrics, while Qiskit Aer provides the simulation environment for executing the quantum components.

E. Data Preprocessing

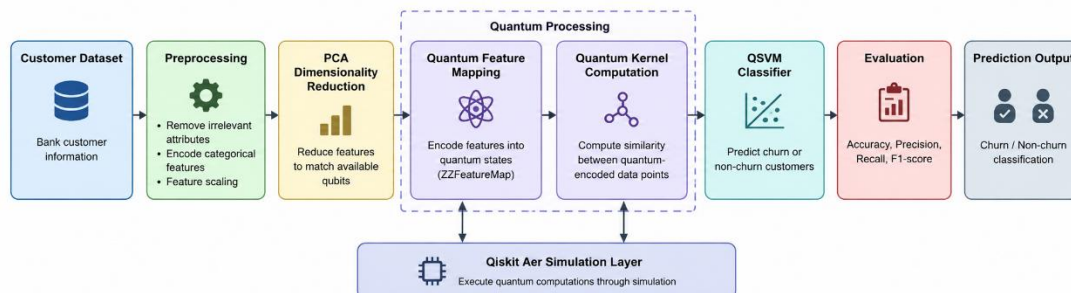


Fig. 1. QSVM System Architecture for Customer Churn Prediction

The attributes RowNumber, CustomerId, and Surname are removed because they identify individual records but do not represent customer behavior. The dataset is checked for missing values using the null-value count and for duplicate records using duplicate-row detection. No missing values or duplicate records are identified; therefore, no imputation or duplicate removal is required. The target attribute Exited is separated from the predictors before model training.

The categorical attributes Geography and Gender are converted into numerical form using one-hot encoding. The first category of each attribute is removed to avoid redundant dummy variables. After encoding, the predictor matrix contains 11 numerical features.

F. Feature Selection / Dimensionality Reduction

To compress the 11-dimensional preprocessed feature space into a four-dimensional representation, Principal Component Analysis (PCA) is employed. Let $X \in \mathbb{R}^{n \times 11}$ denote the scaled feature matrix and let $\tilde{X}$ denote its mean-centered form. PCA determines an orthogonal

basis whose directions are ordered according to the variance captured from the training data. The covariance matrix is computed as

$$C = \frac{1}{n-1} \tilde{X}^T \tilde{X}.$$

Spectral decomposition of C produces the eigenvalue–eigenvector pairs satisfying

$$Cv_i = \lambda_i v_i,$$

where the eigenvalues are ordered as $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_{11}$. The projection matrix is formed from the four eigenvectors associated with the largest eigenvalues:

$$W_4 = [v_1, v_2, v_3, v_4].$$

The four-dimensional feature matrix is then obtained as

$$Z = \tilde{X}W_4, Z \in \mathbb{R}^{n \times 4}.$$

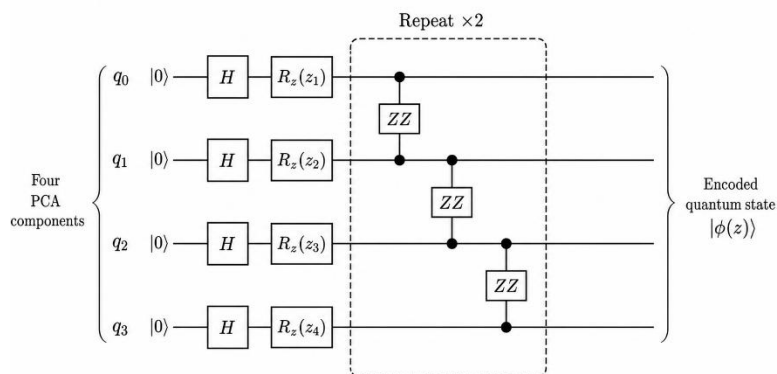
PCA is fitted using only the training data, and the same fitted transformation is applied to the test data. Each of the four resulting principal components is subsequently assigned to one qubit in the quantum feature-mapping circuit.

G. Quantum Feature Mapping

After PCA, each customer record is represented by the four-dimensional vector $\mathbf{z} = [z_1, z_2, z_3, z_4]^T$. A parameterized quantum circuit maps this vector to the state

$$|\phi(\mathbf{z})\rangle = U_{ZZ}(\mathbf{z}) |0\rangle^{\otimes 4},$$

where one qubit corresponds to each principal component. The implementation uses Qiskit's ZZFeatureMap with feature_dimension=4, reps=2, and entanglement='linear'. Within each repetition, data-dependent parameterized gates encode the PCA components, while ZZ interactions entangle adjacent qubits in the sequence $q_0 \leftrightarrow q_1 \leftrightarrow q_2 \leftrightarrow q_3$. Repeating the encoding block twice allows both individual components and interactions between connected components to influence the resulting quantum state. This circuit does not directly predict customer churn; instead, it defines the quantum feature space used to calculate similarities between customer records during quantum-kernel estimation.



H. Quantum Kernel Construction

The quantum kernel measures the similarity between two PCA-transformed customer records after they have been mapped to quantum states. For the reduced vectors $\mathbf{z}_i$ and $\mathbf{z}_j$, the feature map prepares the states $|\phi(\mathbf{z}_i)\rangle$ and $|\phi(\mathbf{z}_j)\rangle$. Their kernel value is calculated from the squared state overlap:

$$K(\mathbf{z}_i, \mathbf{z}_j) = |\langle \phi(\mathbf{z}_i) | \phi(\mathbf{z}_j) \rangle|^2.$$

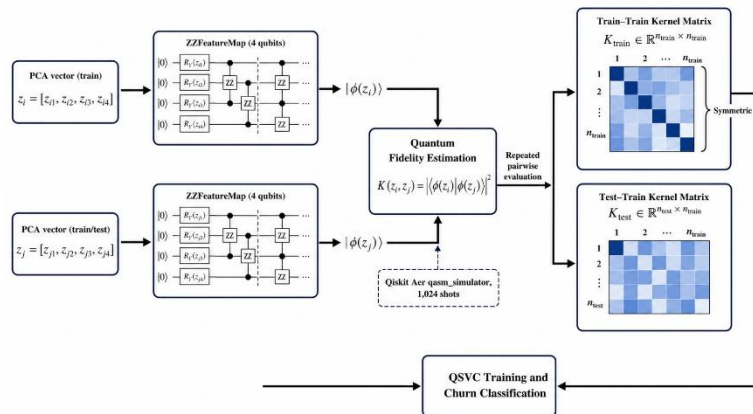
A value approaching one represents a high degree of similarity in the quantum feature space, whereas a value approaching zero represents low similarity. Pairwise evaluation of the training observations produces the square training-kernel matrix

$$[K_{\text{train}}]_{ij} = K(\mathbf{z}_i^{\text{train}}, \mathbf{z}_j^{\text{train}}),$$

while comparison of each test observation with the training observations produces

$$[K_{\text{test}}]_{ij} = K(\mathbf{z}_i^{\text{test}}, \mathbf{z}_j^{\text{train}}).$$

In the documented implementation, the kernel matrices are estimated using Qiskit Machine Learning's QuantumKernel, configured through a QuantumInstance running on the Qiskit Aer qasm_simulator with 1,024 shots. The resulting numerical matrices are supplied to QSVC, allowing support-vector optimization to be performed using similarities generated by the quantum feature map rather than a conventional classical kernel.

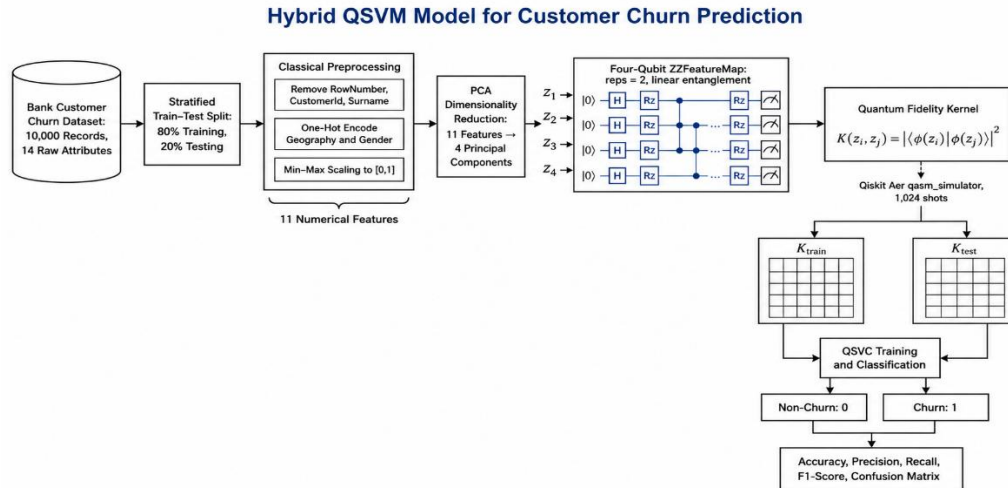


I. QSVM Model

The proposed QSVM is designed as a hybrid classical–quantum model for binary customer-churn classification. The classical stage divides the customer data into training and testing subsets, removes the non-predictive attributes RowNumber, CustomerId, and Surname, converts Geography and Gender through one-hot encoding, and normalizes the resulting predictors to $[0,1]$. PCA, fitted using the training data, transforms the 11-dimensional numerical feature space into four principal components, with each component assigned to one qubit. A four-qubit ZZFeatureMap with two repetitions and linear entanglement then maps each reduced vector $\mathbf{z}$ to the quantum state $|\phi(\mathbf{z})\rangle$. Similarity between two encoded records is calculated using

$$K(\mathbf{z}_i, \mathbf{z}_j) = |\langle \phi(\mathbf{z}_i) | \phi(\mathbf{z}_j) \rangle|^2.$$

Pairwise evaluation of the training observations produces K_{train} , which is supplied to QSVC to learn a maximum-margin decision boundary between churn and non-churn customers. During testing, K_{test} is generated by comparing the encoded test observations with the training observations. The trained classifier uses these kernel values to assign each customer either the non-churn label 0 or churn label 1



J. Training and Classification

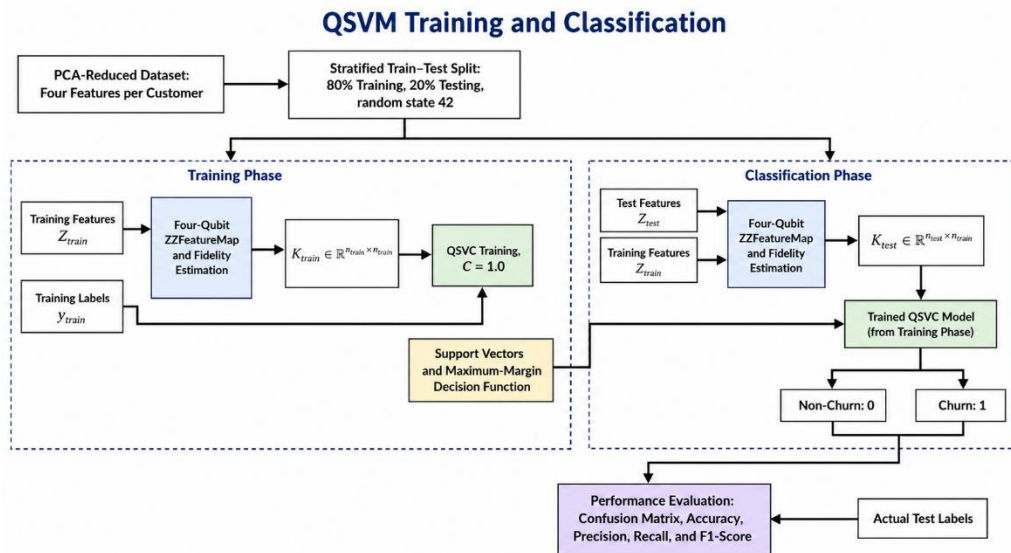
The processed dataset is divided into training and testing subsets using an 80%–20% stratified split with `random_state=42`. For the complete 10,000-record dataset, this division produces 8,000 training observations and 2,000 test observations while preserving the original churn-class distribution. Quantum-kernel values are first calculated between every required pair of training observations to construct the matrix

$$[K_{\text{train}}]_{ij} = K(z_i^{\text{train}}, z_j^{\text{train}}), K_{\text{train}} \in \mathbb{R}^{n_{\text{train}} \times n_{\text{train}}}.$$

The training-kernel matrix and its corresponding churn labels are supplied to QSVC, which identifies the support vectors and determines a maximum-margin decision boundary in the quantum-induced feature space. During testing, each test observation is compared with the training observations to construct

$$[K_{\text{test}}]_{ij} = K(z_i^{\text{test}}, z_j^{\text{train}}), K_{\text{test}} \in \mathbb{R}^{n_{\text{test}} \times n_{\text{train}}}.$$

The trained QSVC uses K_{test} to assign each observation either the non-churn label 0 or churn label 1. The regularization parameter is set to $C = 1.0$. No additional QSVC hyperparameter optimization is reported; parameters not explicitly configured retain their library-default values. The predicted labels are stored and subsequently compared with the actual test labels to calculate the confusion matrix, accuracy, precision, recall, and F1-score.



III. IMPLEMENTATION

SVM comparison, and performance evaluation.

The QSVM system is implemented as a hybrid Python-based workflow that integrates classical data-processing libraries with the Qiskit quantum-computing framework. The implementation begins by loading the customer churn dataset and removing the non-predictive attributes RowNumber, CustomerId, and Surname. The categorical variables Geography and Gender are converted through one-hot encoding, the resulting predictors are normalized to [0,1] using Min-Max scaling, and PCA reduces the 11-dimensional feature space to four principal components. Each reduced customer record is represented by a four-qubit ZZFeatureMap with two repetitions and linear entanglement. The training and testing quantum-kernel matrices are estimated using Qiskit Machine Learning on the Qiskit Aer qasm_simulator with 1,024 shots and are supplied to QSVC to generate churn predictions. Pandas and NumPy support data handling, while scikit-learn performs preprocessing, PCA, stratified dataset division, classical SVM implementation, and performance evaluation

IV. EXPERIMENTAL RESULTS AND DISCUSSION

The QSVM was evaluated on the preprocessed bank customer churn dataset using an **80%–20% stratified data division** with random_state=42. Four PCA components were encoded through a four-qubit ZZFeatureMap, and the quantum-kernel matrices were estimated using the Qiskit Aer qasm_simulator with **1,024 shots**. The model was evaluated using accuracy, precision, recall, F1-score, and a confusion matrix; however, the final metric values must be obtained from the same verified prediction run because the available project documents contain inconsistent performance records. These metrics must be interpreted together with the confusion matrix because the dataset contains 7,963 non-churn records and 2,037 churn records. A controlled comparison with a classical SVM has not yet been verified; therefore, no claim of improved QSVM performance is made. The experiment demonstrates the implementation feasibility of quantum-kernel classification for customer churn analysis but does not establish quantum advantage without repeated evaluation, runtime analysis, and statistically validated classical baselines

V. CONCLUSION

This study developed a hybrid Quantum Support Vector Machine framework for bank customer churn prediction by integrating classical preprocessing with quantum-kernel classification. Customer records were cleaned, one-hot encoded, normalized using Min–Max scaling, and reduced from 11 numerical features to four principal components through PCA. The reduced features were represented using a four-qubit ZZFeatureMap with two repetitions and linear entanglement, while pairwise quantum-state similarities were estimated on the Qiskit Aer qasm_simulator with 1,024 shots to construct the training and testing kernel matrices. Model performance was assessed using accuracy, precision, recall, F1-score, and a confusion matrix. Because a verified classical SVM baseline and one internally consistent set of QSVM results are not currently available, no claim of improved performance or quantum advantage is made. Nevertheless, the study establishes the implementation feasibility of applying a small, NISQ-compatible quantum-kernel model to customer churn classification and provides a foundation for controlled comparisons on quantum simulators and physical quantum hardware.

VI. FUTURE SCOPE

Future research should examine the QSVM on physical quantum processors to assess the influence of hardware noise, limited qubit connectivity, circuit transpilation, sampling error, and error-mitigation methods. Controlled experiments may compare the current ZZFeatureMap with angle encoding, Pauli-based feature maps, different circuit depths, and alternative entanglement structures. As larger quantum systems become available, additional principal components may be retained to investigate the relationship between feature dimensionality, qubit count, kernel quality, and computational cost. Because the dataset contains fewer churn observations than non-churn observations, class weighting, cost-sensitive learning, and SMOTE may also be evaluated using training data only. The QSVM should further be compared with classical SVM, Random Forest, and gradient-boosting classifiers using identical preprocessing steps, data partitions, and cross-validation folds. Future evaluations may include runtime, precision–recall AUC, Matthews correlation coefficient, statistical uncertainty, and the financial cost of false predictions. The framework may subsequently be adapted to related banking applications such as loan-default prediction, credit-risk assessment, and customer-product recommendation.

ACKNOWLEDGMENT

Mrs. P. Ratna Kumari, M.Tech., is working as an Assistant Professor in the Department of Computer Science and Engineering (CSE) at Sanketika Vidya Parishad Engineering College, Visakhapatnam, Andhra Pradesh, which is accredited with an A Grade by NAAC. Her areas of interest include C, Data Warehousing and Data Mining, Design and Analysis of Algorithms, Python, Formal Languages and Automata Theory, and Compiler Design.

Siriyala Mohan Krishna is pursuing his final semester M.Tech. at Sanketika Vidya Parishad Engineering College, Visakhapatnam, Andhra Pradesh, which is accredited with an A Grade by NAAC, affiliated with Andhra University, and approved by AICTE. With a keen interest in Quantum Computing, he has undertaken his postgraduate project titled “Quantum Support Vector Machine for Customer Churn Prediction” and published a research paper based on the project under the guidance of Mrs. P. Ratna Kumari, Assistant Professor, Sanketika Vidya Parishad Engineering College.

REFERENCES

- [1] C. Cortes and V. Vapnik, "Support-vector networks," *Machine Learning*, vol. 20, no. 3, pp. 273–297, 1995, doi: 10.1007/BF00994018.
- [2] P. Rebentrost, M. Mohseni, and S. Lloyd, "Quantum support vector machine for big data classification," *Physical Review Letters*, vol. 113, no. 13, Art. no. 130503, 2014, doi: 10.1103/PhysRevLett.113.130503.
- [3] J. Biamonte, P. Wittek, N. Pancotti, P. Rebentrost, N. Wiebe, and S. Lloyd, "Quantum machine learning," *Nature*, vol. 549, pp. 195–202, 2017, doi: 10.1038/nature23474.
- [4] M. Schuld and N. Killoran, "Quantum machine learning in feature Hilbert spaces," *Physical Review Letters*, vol. 122, no. 4, Art. no. 040504, 2019, doi: 10.1103/PhysRevLett.122.040504.
- [5] 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, 2019, doi: 10.1038/s41586-019-0980-2.
- [6] I. T. Jolliffe, *Principal Component Analysis*, 2nd ed. New York, NY, USA: Springer, 2002, doi: 10.1007/b98835.
- [7] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. [Online]. Available: <https://jmlr.org/papers/v12/pedregosa11a.html>.
- [8] J. Hadden, A. Tiwari, R. Roy, and D. Ruta, "Computer assisted customer churn management: State-of-the-art and future trends," *Computers & Operations Research*, vol. 34, no. 10, pp. 2902–2917, 2007, doi: 10.1016/j.cor.2005.11.007.
- [9] S. A. Neslin, S. Gupta, W. Kamakura, J. Lu, and C. H. Mason, "Defection detection: Measuring and understanding the predictive accuracy of customer churn models," *Journal of Marketing Research*, vol. 43, no. 2, pp. 204–211, 2006, doi: 10.1509/jmkr.43.2.204.
- [10] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.
- [11] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *The Annals of Statistics*, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.
- [12] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information*, 10th anniversary ed. Cambridge, U.K.: Cambridge University Press, 2010, doi: 10.1017/CBO9780511976667.
- [13] IBM, "Qiskit documentation," *IBM Quantum*. [Online]. Available: <https://docs.quantum.ibm.com/>. [Accessed: Sep. 24, 2026].
- [14] Qiskit Community, "Qiskit Machine Learning documentation." [Online]. Available: <https://qiskit-community.github.io/qiskit-machine-learning/>. [Accessed: Sep. 24, 2026].
- [15] S. Badole, "Banking Customer Churn Prediction Dataset," *Kaggle*, ver. 2, 2024. [Online]. Available: <https://www.kaggle.com/datasets/saurabhbadole/bank-customer-churn-prediction-dataset>. [Accessed: Sep. 24, 2026].