Customer Churn Prediction Using Machine Learning: A Comparative Analysis of Classification Models

Anthony M Petrillo Jr and Chi Jiao

***Abstract*—Customer churn prediction is an important problem in industries that rely on recurring customers, especially in telecommunications where losing customers directly impacts revenue. However, accurately identifying which customers are likely to leave remains a challenge due to the complexity of customer behavior and imbalanced data. In this study, multiple machine learning models were developed and compared using the Telco Customer Churn dataset, including Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Random Forest, Gradient Boosting, and XGBoost. The models were evaluated using accuracy, precision, recall, F1-score, and ROC-AUC to ensure a well-rounded performance comparison. Among all models, Logistic Regression achieved the highest F1-score (0.6040) while maintaining strong overall performance, making it the most balanced model for this problem. These results demonstrate that simpler models can remain competitive against more complex approaches and that selecting the appropriate evaluation metric is critical when working with imbalanced datasets.**

I. Introduction

Customer churn refers to the process in which customers stop using a company's product or service. In industries that rely on recurring revenue, such as telecommunications, customer churn is a major concern because it directly impacts long-term profitability[1]. Retaining existing customers is often more cost-effective than acquiring new ones, making churn prediction an important task for businesses.

As companies continue to collect large amounts of customer data, machine learning has become a useful tool for identifying patterns and predicting which customers are most likely to leave. However, accurately predicting churn remains challenging due to the complexity of customer behavior and class imbalance.

Many datasets contain significantly more non-churners than churners, making accuracy alone an unreliable metric. Because of this, it is important to evaluate models across multiple metrics rather than focusing on only one.

This study evaluates multiple machine learning models to determine which provides the most well-rounded performance rather than excelling in a single metric.

A. Petrillo is with the Department of Computer Science, New Jersey City University, New Jersey, 07102, USA, (e-mail: anthonympetrillojr@gmail.com).

C. Jiao is with the Artificial Intelligence and Software College, Liaoning Petrochemical University, Fushun 113001, P. R. China, (e-mail: 1903233560@qq.com).

Corresponding author: Chi Jiao.

II. Related Work

Customer churn prediction has been widely studied in industries such as telecommunications, where retaining customers is critical for long-term profitability[2]. Early research highlights the importance of customer retention strategies and the financial impact of churn on businesses [3, 4]. As data availability has increased, machine learning and data mining techniques have become common approaches for predicting churn due to their ability to identify patterns in large datasets [5, 6].

Traditional machine learning models, such as Logistic Regression and rule-based approaches, have been widely used because of their simplicity, efficiency, and interpretability [7, 8]. These models provide insight into how different features influence churn behavior, making them valuable for decision-making. However, they may struggle to capture more complex relationships within the data.

More advanced machine learning models have also been applied to churn prediction, including ensemble methods and nonlinear approaches. Techniques such as Random Forest, Support Vector Machines (SVM) [9], and hybrid or neural network-based models have shown improved predictive performance by capturing more complex interactions between features [10, 11]. In particular, ensemble methods such as Random Forest and boosting-based approaches have gained popularity due to their ability to improve accuracy and reduce variance [12, 13]. XGBoost, a scalable and optimized implementation of gradient boosting, has demonstrated strong performance in classification tasks due to its efficiency and built-in regularization capabilities [13, 14].

Recent studies have explored deep learning approaches, including recurrent neural networks and other advanced architectures, to further improve churn prediction performance [15, 16]. While these models can achieve strong results, they often require significantly more computational resources and are generally less interpretable than simpler methods.

Despite these advancements, many existing studies focus primarily on optimizing a single performance metric, such as accuracy or ROC-AUC, without fully addressing class imbalance or the trade-offs between precision and recall. Evaluation techniques such as ROC analysis highlight the importance of considering multiple performance measures when assessing classification models [17, 18]. In real-world churn prediction tasks, models that perform consistently well across multiple metrics are often more useful than those that excel in only

one.

This study builds on existing work by comparing multiple machine learning models using a consistent dataset and evaluation framework. Rather than focusing on a single metric [19], this paper evaluates performance across accuracy, precision, recall, F1-score, and ROC-AUC to determine which model provides the most well-rounded performance for customer churn prediction.

III. Dataset and Preprocessing

A. Dataset Description

The dataset used in this study is the `WA_Fn-UseC_-Telco-Customer-Churn.csv` dataset, which is publicly available on Kaggle. This dataset contains customer-level information from a telecommunications company and is commonly used for churn prediction tasks. Each record represents an individual customer along with various attributes describing demographic characteristics, service usage, and billing details.

The dataset consists of 7,043 customer records and 21 features prior to preprocessing. These features can be grouped into several categories. Demographic features include variables such as `gender`, `SeniorCitizen`, `Partner`, and `Dependents`, which describe basic customer characteristics. Customer lifecycle information is primarily represented by the `tenure` feature, indicating how long a customer has remained with the company.

Service-related features include `PhoneService`, `MultipleLines`, `InternetService`, and additional services such as `OnlineSecurity`, `OnlineBackup`, `DeviceProtection`, `TechSupport`, `StreamingTV`, and `StreamingMovies`. These variables provide insight into how customers interact with the company's services and their level of engagement.

Billing and contract-related features include `Contract`, `PaperlessBilling`, `PaymentMethod`, `MonthlyCharges`, and `TotalCharges`. These features describe the financial relationship between the customer and the company, including payment behavior and contractual commitments.

The dataset also includes a unique identifier, `customerID`, which does not provide predictive value and is removed during preprocessing. The target variable is `Churn`, which indicates whether a customer has left the service. This variable is originally represented as categorical values ("Yes" or "No") and is converted into a binary format, where 1 represents churn and 0 represents no churn.

An important characteristic of this dataset is class imbalance. Approximately 26.5% of customers are labeled as churners, while the majority are non-churners. This imbalance necessitates the use of evaluation metrics beyond accuracy, such as precision, recall, and F1-score, to properly assess model performance. In addition to class imbalance, the dataset presents a mix of categorical and numerical features, which introduces additional complexity during preprocessing. Categorical variables such as payment methods and contract types must be carefully encoded to preserve their informational value without introducing unintended bias. Similarly, numerical features such as tenure and monthly charges may operate on different scales, which can impact the behavior of certain machine learning models. The combination of these feature types makes this dataset a strong candidate for evaluating how different models handle structured, real-world data [20]. It also reflects practical business scenarios, where data is rarely clean or uniform, and preprocessing decisions can significantly influence final model performance.

B. Data Preprocessing

The preprocessing stage ensures that the dataset is clean, consistent, and suitable for machine learning models while avoiding data leakage.

First, the `TotalCharges` feature was converted from an object type to a numeric format. This conversion introduced a small number of missing values due to non-numeric entries, which were handled using median imputation. Median imputation was chosen because it provides a stable estimate that is less sensitive to outliers than mean imputation.

The `customerID` column was removed, as it is a unique identifier and does not contribute to predicting customer churn. Retaining this feature would introduce noise without adding meaningful information.

The target variable `Churn` was encoded into a binary format, where 1 represents churn and 0 represents no churn. This transformation allows classification models to process the target variable effectively.

The dataset was then separated into feature variables (X) and the target variable (y). A train-test split was performed using an 80/20 ratio, with stratification applied to preserve the original class distribution. This ensures that both the training and testing sets contain a representative proportion of churners and non-churners.

Categorical features were processed using one-hot encoding, allowing models to interpret categorical values as numerical inputs. The encoding was configured to ignore unseen categories in the test data, preventing errors during evaluation.

Numerical features were handled differently depending on the model. For models sensitive to feature scaling, such as Logistic Regression, K-Nearest Neighbors, and Support Vector Machines, numerical features were standardized to ensure consistent feature magnitudes. For tree-based models, including Random Forest, Gradient Boosting, and XGBoost, scaling was not applied, as these models are not sensitive to feature scale.

A pipeline-based approach was used to ensure that preprocessing steps were applied consistently across all models. This approach also prevents data leakage by fitting transformations only on the training data and then applying them to the test data.

Overall, these preprocessing steps ensure that the dataset is properly formatted and that all models are evaluated under consistent and fair conditions.

IV. Methodology

This study evaluates multiple machine learning models to predict customer churn using the preprocessed dataset.

Each model represents a different approach to classification, enabling a comprehensive comparison of performance across multiple evaluation metrics. The selected models include Logistic Regression, K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest, Gradient Boosting, and XGBoost.

A. Logistic Regression

Logistic Regression is a linear classification model that estimates the probability of a binary outcome using the logistic function. It is widely used due to its simplicity, efficiency, and interpretability. Despite being a relatively simple model, it often performs well on structured datasets and serves as a strong baseline for comparison.

B. K-Nearest Neighbors (KNN)

K-Nearest Neighbors is a non-parametric, distance-based algorithm that classifies data points based on the majority class of their nearest neighbors in feature space. The model relies heavily on the choice of distance metric and is sensitive to feature scaling, making preprocessing an important factor for effective performance.

C. Support Vector Machine (SVM)

Support Vector Machine is a classification algorithm that identifies the optimal decision boundary (hyperplane) that separates classes. It maximizes the margin between classes to improve generalization. SVM can model nonlinear relationships using kernel functions but typically requires higher computational resources compared to simpler models.

D. Random Forest

Random Forest is an ensemble learning method that constructs multiple decision trees and aggregates their predictions to improve accuracy and reduce overfitting. By using random subsets of data and features, it enhances generalization and effectively captures nonlinear relationships.

E. Gradient Boosting

Gradient Boosting is an ensemble technique that builds decision trees sequentially, where each new tree corrects the errors of the previous ones. This iterative approach allows the model to progressively improve performance, often resulting in strong predictive accuracy. However, it can be sensitive to hyperparameter selection and may require careful tuning.

F. XGBoost

XGBoost (Extreme Gradient Boosting) is an optimized implementation of gradient boosting designed for efficiency and performance. It includes advanced features such as regularization, parallel processing, and efficient handling of missing values. XGBoost is widely used in both research and industry due to its strong performance on structured data.

G. Model Evaluation Strategy

All models were trained and evaluated using the same preprocessing pipeline to ensure a fair and consistent comparison. Performance was assessed using multiple evaluation metrics, including accuracy, precision, recall, F1-score, and ROC-AUC. These metrics provide a comprehensive understanding of model performance, particularly in the presence of class imbalance.

The primary goal of this methodology is not only to compare model accuracy, but to determine which model provides the most well-rounded performance across all evaluation metrics. Evaluating models across multiple metrics is particularly important in classification problems involving imbalanced datasets. A model that achieves high accuracy may still perform poorly in identifying the minority class, which in this case represents churned customers. By incorporating precision, recall, and F1-score into the evaluation process, this study ensures that model performance is assessed in a more balanced and meaningful way. This approach reflects real-world decision-making scenarios, where businesses are not only concerned with overall correctness, but also with correctly identifying high-risk customers. As a result, the evaluation strategy emphasizes practical usefulness rather than purely statistical performance.

V. Experimental Results

A. Experimental Setup

All experiments were conducted using Python in a Google Colab environment. The implementation utilized common machine learning libraries, including `pandas` for data manipulation, `NumPy` for numerical operations, `scikit-learn` for model development and evaluation, and `XGBoost` for gradient boosting implementation.

The dataset was split into training and testing sets using an 80/20 ratio, with stratification applied to preserve the class distribution of the target variable. This ensures that both training and testing datasets contain a representative proportion of churners and non-churners.

Each model was trained using the same preprocessing pipeline to ensure consistency and fairness in comparison. Performance was evaluated on the test set to measure how well each model generalizes to unseen data.

B. Evaluation Metrics

To evaluate model performance, multiple metrics were used to provide a well-rounded assessment. These include accuracy, precision, recall, F1-score, and ROC-AUC.

Accuracy measures the overall proportion of correct predictions but can be misleading in imbalanced datasets. Precision measures the proportion of correctly predicted churners out of all predicted churners, while recall measures the proportion of actual churners correctly identified by the model. The F1-score provides a balance between precision and recall, making it a particularly important metric in this study.

ROC-AUC evaluates the model's ability to distinguish between classes across different classification thresholds, providing a more comprehensive view of performance.

TABLE I Performance Comparison Across Models

Model	Accuracy	Precision	Recall	F1-score	ROC-AUC
Logistic Regression	0.8055	0.6572	0.5588	0.6040	0.8419
Gradient Boosting	0.8070	0.6747	0.5267	0.5916	0.8434
KNN	0.7637	0.5527	0.5749	0.5636	0.7898
SVM	0.7913	0.6418	0.4840	0.5518	0.7905
Random Forest	0.7814	0.6115	0.4840	0.5403	0.8177
XGBoost	0.7722	0.5826	0.5000	0.5381	0.8112

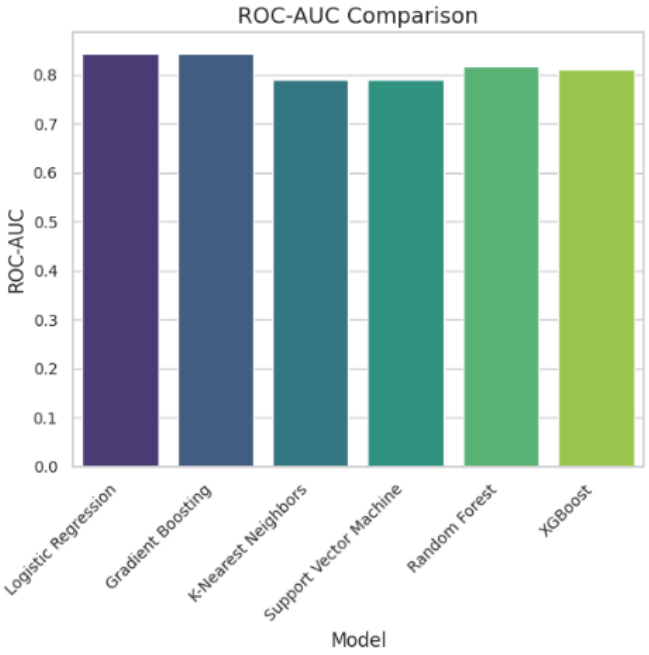

Fig. 1. ROC-AUC comparison across classification models.

Fig. 1 presents the ROC-AUC comparison across all classification models, illustrating each model's ability to distinguish between churners and non-churners across different thresholds. All models achieve relatively strong ROC-AUC values, indicating performance well above random classification. Gradient Boosting and Logistic Regression perform slightly better than the other models, suggesting stronger class separation capability. However, the differences between models are relatively small, reinforcing that no single model dominates purely in this metric. This highlights the importance of evaluating multiple performance measures rather than relying on ROC-AUC alone. Logistic Regression's strong ROC-AUC contributes to its overall well-rounded performance.

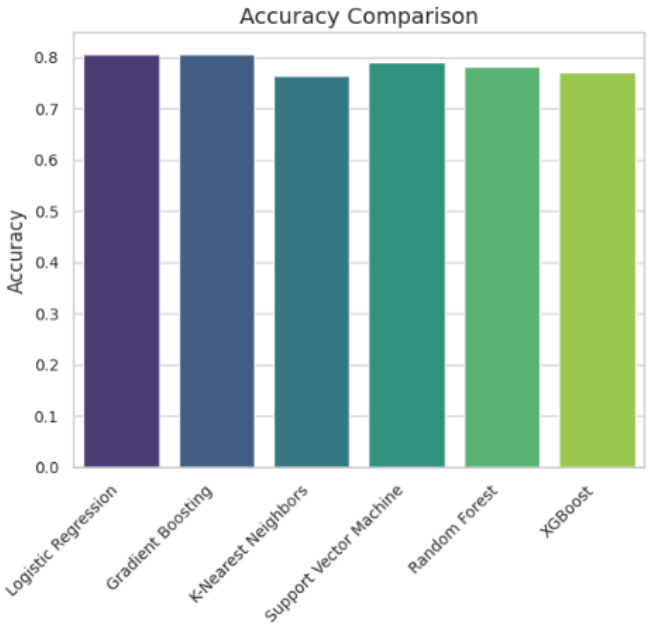

Fig. 2. Accuracy comparison across classification models.

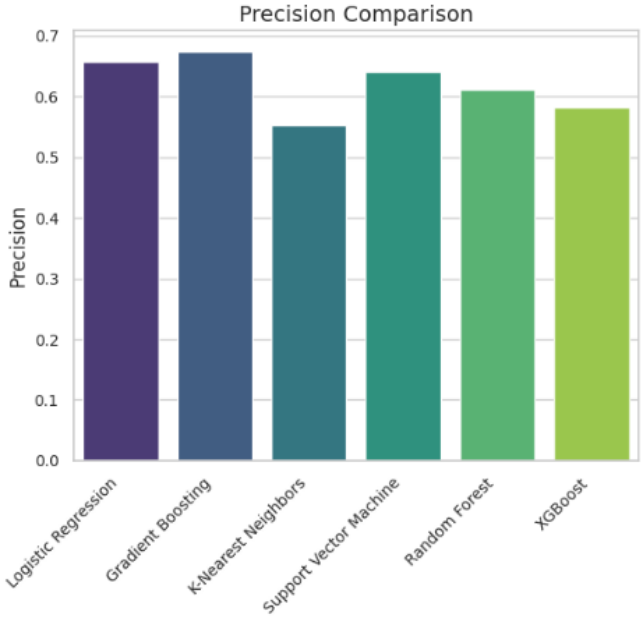

Fig. 3. Precision comparison across classification models.

C. Results Comparison

The performance of all models across multiple evaluation metrics is summarized in Table I.

To further visualize model performance across different metrics, accuracy, precision, recall, and F1-score comparisons are shown in Fig. 2 through Fig. 5.

Fig. 2 shows the accuracy comparison across all models, where Gradient Boosting achieves the highest accuracy, although the difference compared to Logistic Regression is small. The figure demonstrates that most models perform within a narrow accuracy range, indicating that overall correctness is similar across approaches. However, accuracy alone can be misleading in imbalanced datasets, as correctly classifying the majority class can inflate performance. This means that a high accuracy score does not necessarily reflect strong performance in identifying churners. Logistic Regression maintains competitive accuracy while also performing well across other evaluation metrics. This further supports the argument that it provides a more well-rounded and reliable model across all evaluation metrics.

Fig. 3 shows the precision comparison across classification models, highlighting how accurately each model identifies churners when it predicts churn. Gradient Boosting achieves the highest precision, meaning its churn predictions are more

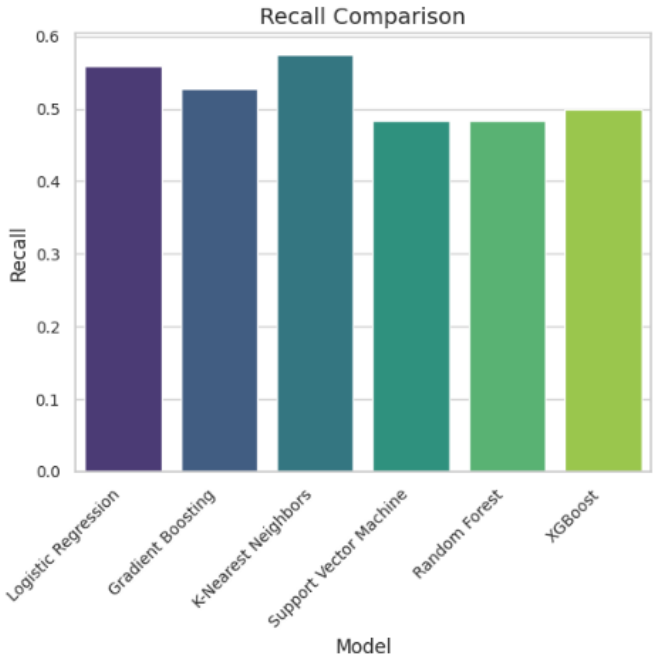

Fig. 4. Recall comparison across classification models.

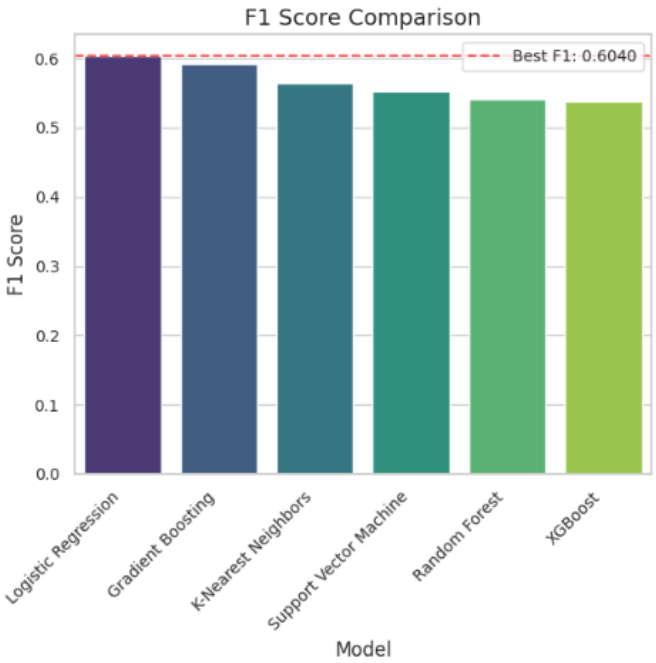

Fig. 5. F1-score comparison across classification models.

likely to be correct. However, high precision alone does not guarantee that the model identifies enough actual churners. Logistic Regression remains competitive in precision while also maintaining stronger balance across the other metrics. This supports the paper's main argument that the best model is not simply the one that wins one category, but the one that performs well across the full evaluation framework.

Fig. 4 presents the recall comparison across models, showing how effectively each model identifies customers who actually churned. K-Nearest Neighbors achieves the highest recall, meaning it captures more churners than the other models. However, this higher recall comes with lower precision, meaning it also produces more false positives. Logistic Regression achieves a more balanced recall while maintaining stronger overall performance across precision and F1-score. This tradeoff is important because churn prediction requires both identifying likely churners and avoiding excessive incorrect churn predictions.

Fig. 5 highlights the F1-score comparison across all models. Because F1-score combines precision and recall, it is especially useful for evaluating imbalanced classification problems such as churn prediction. Logistic Regression achieves the highest F1-score, showing that it provides the strongest balance between correctly identifying churners and limiting false positives. Other models perform well in individual areas but do not maintain the same overall balance. This result strongly supports the conclusion that Logistic Regression is the most well-rounded model in this study.

D. Visual Analysis

Fig. 6 illustrates the ROC curves for all classification models, showing the relationship between true positive rate and false positive rate across different classification thresholds.

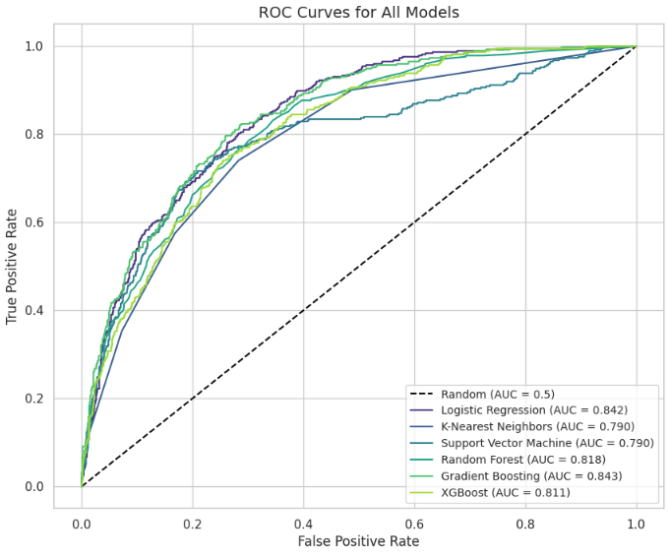

Fig. 6. ROC curves for all classification models.

All curves lie above the diagonal baseline, confirming that each model performs better than random guessing. Logistic Regression and Gradient Boosting maintain curves closer to the top-left corner, indicating stronger classification performance across thresholds. Logistic Regression demonstrates consistent performance across the entire curve, suggesting stability across different decision boundaries. While multiple models perform well, Logistic Regression remains competitive throughout. This further supports its role as a well-rounded classification model.

Fig. 7 presents the precision-recall curves, which are especially important for evaluating performance on imbalanced datasets. These curves show how precision changes as recall

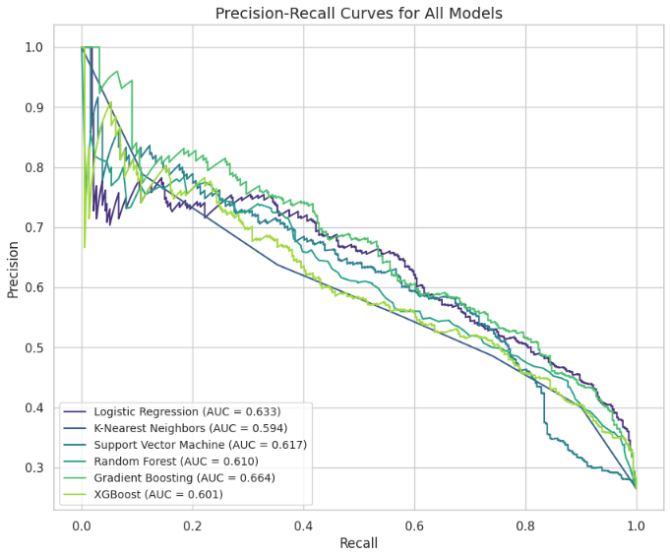

Fig. 7. Precision-recall curves for all classification models.

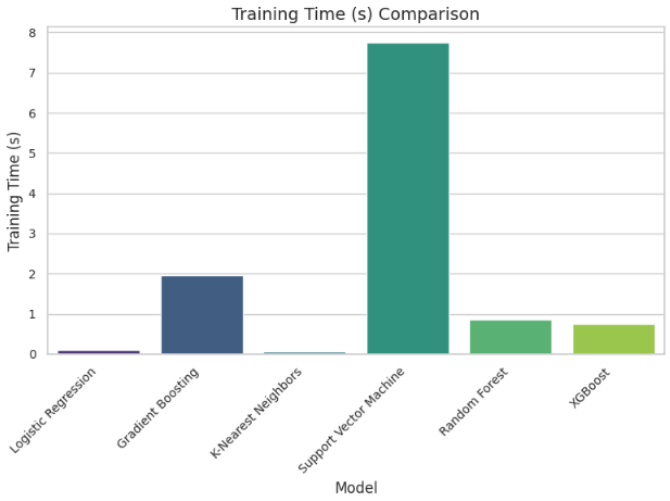

Fig. 8. Training time comparison across classification models.

increases, providing insight into model reliability when identifying churners. Logistic Regression maintains a relatively strong balance between precision and recall across different thresholds. Some models experience a sharp drop in precision as recall increases, indicating less stable performance. This suggests that while certain models may perform well at specific thresholds, they are less consistent overall. Logistic Regression demonstrates a smoother tradeoff, reinforcing its balanced performance. This makes it more suitable for real-world churn prediction scenarios.

E. Results Discussion

Fig. 8 compares the training time required for each model, highlighting differences in computational efficiency. Support Vector Machine requires significantly more training time than the other models, making it less practical for larger datasets or time-sensitive applications. Logistic Regression and tree-based models demonstrate much faster training times while still maintaining strong predictive performance. This efficiency is important for scalability and real-world deployment. Models with excessive training time may not be suitable for production environments. Logistic Regression provides a strong balance between performance and efficiency, reinforcing its practicality.

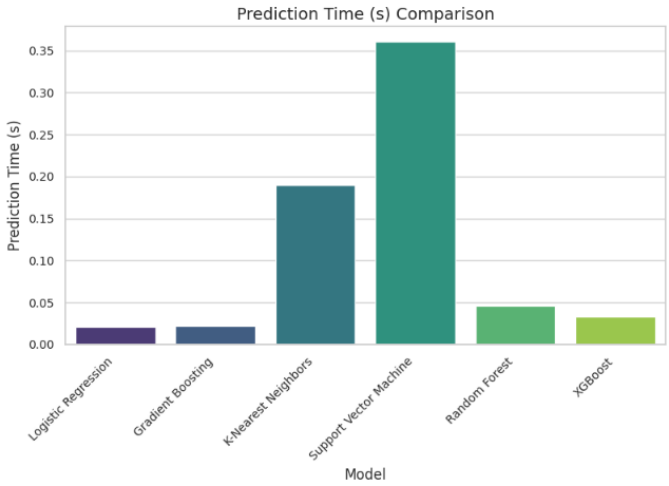

Fig. 9. Prediction time comparison across classification models.

Fig. 9 shows the prediction time across all models, providing insight into their suitability for real-time applications. Support Vector Machine again performs the slowest, indicating limitations in scalability. Logistic Regression and ensemble models demonstrate significantly faster prediction speeds. Fast prediction time is critical in business environments where decisions must be made quickly. While some models achieve strong predictive performance, slower prediction speeds reduce their practicality. Logistic Regression achieves both efficiency and strong performance across metrics. This reinforces its role as the most balanced and deployable model in this study.

Overall, the results demonstrate that while more complex models can achieve strong performance in specific metrics, Logistic Regression provides the most well-rounded performance across all evaluation metrics while also maintaining computational efficiency. This makes it the most practical and reliable choice for customer churn prediction in this study.

VI. Discussion

The results of this study show that no single model dominates across all evaluation metrics, reinforcing the importance of evaluating performance from multiple perspectives. While Gradient Boosting achieved the highest accuracy and slightly higher ROC-AUC, Logistic Regression achieved the highest F1-score, indicating the best balance between precision and recall. Given the class imbalance in the dataset, this balance is especially important, as it reflects a model's ability to correctly identify churners without producing excessive false positives.

One key observation is that more complex models did not consistently outperform simpler ones. Ensemble methods such as Random Forest, Gradient Boosting, and XGBoost are typically expected to perform better due to their ability to

capture nonlinear relationships. However, the results suggest that the structure of this dataset does not strongly favor highly complex models. Instead, Logistic Regression was able to generalize effectively and maintain consistent performance across all evaluation metrics.

Interpretability also plays a significant role in model selection. Logistic Regression provides clear insight into how features influence predictions, which is valuable in real-world business applications. In contrast, more complex models often function as "black boxes," making it more difficult to explain predictions to stakeholders. This tradeoff between predictive power and interpretability further supports the use of Logistic Regression as a practical solution.

Efficiency is another important factor. As shown in the experimental results, Support Vector Machine required significantly more training and prediction time, making it less suitable for large-scale or real-time applications. Logistic Regression and tree-based models demonstrated faster performance, increasing their practicality for deployment.

Overall, these findings suggest that the most effective model is not necessarily the one with the highest score in a single metric, but the one that performs consistently well across multiple metrics while remaining efficient and interpretable. In this study, Logistic Regression provides the most well-rounded performance for customer churn prediction. From a practical perspective, these findings suggest that organizations do not always need to rely on highly complex models to achieve effective results. Simpler models such as Logistic Regression can provide strong performance while remaining easier to implement, maintain, and interpret. This is particularly valuable in business environments where transparency and explainability are important for decision-making. For example, understanding which factors contribute to customer churn can help organizations design targeted retention strategies. In this context, a model that is slightly less complex but more interpretable may provide greater overall value than a more complex alternative with marginal performance gains. This reinforces the importance of considering both technical performance and real-world applicability when selecting machine learning models.

VII. Limitations

This study has several limitations that should be considered when interpreting the results.

First, the analysis is based on a single dataset, which limits the generalizability of the findings. While the Telco Customer Churn dataset is widely used, it may not fully represent customer behavior across different industries or business contexts. Evaluating these models on additional datasets would provide a more comprehensive assessment of their effectiveness.

Second, the study relies on a single train-test split, which introduces some variability in performance results. Although stratification preserves class distribution, different splits may yield slightly different outcomes. Using cross-validation would provide a more robust and reliable estimate of model performance.

Third, feature engineering was relatively limited. While standard preprocessing techniques were applied, more advanced feature engineering could further improve model performance. For example, creating interaction features or incorporating domain-specific insights may reveal stronger predictive patterns.

Additionally, the models in this study focus on correlation rather than causation. While they identify patterns associated with churn, they do not explain why customers leave, which limits deeper interpretability of customer behavior.

Finally, although multiple models were evaluated, hyperparameter tuning was not extensively optimized for all models. More thorough tuning could improve performance, particularly for more complex models such as Gradient Boosting and XGBoost. Another limitation of this study is the absence of temporal analysis within the dataset. Customer churn behavior often evolves over time, and static datasets may not fully capture changes in customer engagement or service usage patterns. Incorporating time-based features or longitudinal data could provide deeper insight into how customer behavior shifts leading up to churn events. Additionally, external factors such as market competition, promotional offers, or economic conditions are not represented in the dataset but may significantly influence churn behavior. Including such contextual variables in future studies could improve model performance and provide a more comprehensive understanding of customer decision-making processes.

VIII. Conclusion

Customer churn prediction remains an important problem for businesses that rely on long-term customer relationships. This study developed and evaluated multiple machine learning models using the Telco Customer Churn dataset to determine which model provides the most well-rounded performance.

Six models were compared, including Logistic Regression, K-Nearest Neighbors, Support Vector Machine, Random Forest, Gradient Boosting, and XGBoost. Performance was evaluated using multiple metrics, including accuracy, precision, recall, F1-score, and ROC-AUC, to ensure a comprehensive and fair assessment.

The results show that Logistic Regression achieved the highest F1-score and consistently maintained strong performance across all evaluation metrics. While more complex models such as Gradient Boosting and XGBoost performed well in specific areas, they did not consistently outperform Logistic Regression when overall performance was considered.

These findings reinforce the importance of evaluating models using multiple metrics, particularly in imbalanced classification problems. A model that performs consistently well across metrics is often more valuable than one that excels in only a single area.

In conclusion, Logistic Regression represents a strong, reliable, and well-rounded choice for customer churn prediction in this study due to its balanced performance across all evaluation metrics, computational efficiency, and interpretability. Future work may explore additional datasets, more advanced feature engineering, and expanded hyperparameter tuning to further improve predictive performance and generalizability.

REFERENCES

[1] L. Zhou, H. Zhu, and B. Akbari, "Multi-objective optimization of multi-factory remanufacturing process considering worker fatigue," *International Journal of Artificial Intelligence and Green Manufacturing*, vol. 1, no. 2, pp. 93–107, June 2025.

[2] Y. Li, Y. Zhou, Q. Yan, X. Wu, and Y. Bo, "Dart-gnn: A dynamic recurrent graph neural network for multivariate time series anomaly detection," *International Journal of Artificial Intelligence and Green Manufacturing*, vol. 1, no. 3, pp. 119–128, September 2025.

[3] V. Kumar and J. A. Petersen, "Managing customer retention: An empirical analysis," *Journal of Marketing*, vol. 69, no. 4, pp. 89–105, 2005.

[4] S. A. Neslin *et al.*, "Defection detection: Measuring and understanding the predictive accuracy of customer churn models," *Journal of Marketing Research*, vol. 43, no. 2, pp. 204–211, 2006.

[5] E. W. T. Ngai *et al.*, "Application of data mining techniques in customer relationship management: A literature review and classification," *Expert Systems with Applications*, vol. 36, no. 2, pp. 2592–2602, 2009.

[6] J. Hadden *et al.*, "Computer assisted customer churn management: State-of-the-art and future trends," *Computers & Operations Research*, vol. 34, no. 10, pp. 2902–2917, 2007.

[7] W. Verbeke *et al.*, "Building comprehensible customer churn prediction models with advanced rule induction techniques," *Expert Systems with Applications*, vol. 38, no. 3, pp. 2354–2364, 2011.

[8] ——, "Predictive modeling for customer churn: A review," *Expert Systems with Applications*, vol. 38, no. 12, pp. 15 031–15 043, 2012.

[9] Z. Zhang, X. Guo, M. Zhou, S. Liu, and L. Qi, "Multi-objective discrete grey wolf optimizer for solving stochastic multi-objective disassembly sequencing and line balancing problem," in *2020 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*. IEEE, 2020, pp. 682–687.

[10] A. Idris *et al.*, "Intelligent churn prediction in telecom: Employing mrmr feature selection and rotboost based ensemble classification," *Applied Intelligence*, vol. 39, pp. 659–672, 2013.

[11] C.-F. Tsai and Y.-H. Lu, "Customer churn prediction by hybrid neural networks," *Expert Systems with Applications*, vol. 36, no. 10, pp. 12 547–12 553, 2009.

[12] Y. Xiang *et al.*, "An improved random forest algorithm for customer churn prediction," *Journal of Information & Computational Science*, vol. 9, no. 3, pp. 681–689, 2012.

[13] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining*, 2016, pp. 785–794.

[14] Y. Ji, Z. Zhao, S. Liu, and X. Yong, "Machine learning-based prediction of surplus material in intelligent production processes," *International Journal of Artificial Intelligence and Green Manufacturing*, vol. 1, no. 1, pp. 26–35, March 2025.

[15] X. Chen *et al.*, "Customer churn prediction using recurrent neural networks," *IEEE Access*, vol. 6, pp. 54 089–54 097, 2018.

[16] G. Xia *et al.*, "A deep learning framework for customer churn prediction," *IEEE Access*, vol. 8, pp. 88 976–88 985, 2020.

[17] T. Fawcett, "An introduction to roc analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006.

[18] D. J. Hand and R. J. Till, "A simple generalisation of the area under the roc curve for multiple class classification problems," *Machine Learning*, vol. 45, no. 2, pp. 171–186, 2001.

[19] J. Wang, M. Zhou, X. Guo, and L. Qi, "Multiperiod asset allocation considering dynamic loss aversion behavior of investors," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 1, pp. 73–81, 2018.

[20] C. Xu, H. Wei, X. Guo, S. Liu, L. Qi, and Z. Zhao, "Human-robot collaboration multi-objective disassembly line balancing subject to task failure via multi-objective artificial bee colony algorithm," *IFAC-PapersOnLine*, vol. 53, no. 5, pp. 1–6, 2020.