JKBC

Volume – 7, December 2025

Ensemble of Machine Learning Algorithm for Customer Churn in Telecom Industry

Anand Kumar Sah[1]; Pratap Sapkota[2]

[1]Department of Electronics and Computer Engineering, Tribhuvan University, Institute of Engineering, Pulchowk Campus, Lalitpur, Nepal

[2] Nepal Telecom, Kathmandu, Nepal

Received: July 27, 2025; Revised: November 23, 2025; Accepted: December 12, 2025

Corresponding Email: anand.sah@pcampus.edu.np

Abstract

Churn occurs when customers switch providers due to dissatisfaction or competitor offers, causing major losses since retention is cheaper than acquisition. With rising global competition, customer retention is vital for sustainability. Telecom companies now analyze massive CDR data to detect churn patterns and predict likely churners early. This enables effective retention strategies, service improvements, and targeted marketing. The proposed model preprocesses CDR data, rebalances it, applies machine learning for classification, and uses ensemble learning to enhance accuracy and generate reliable churn insights.

Keywords: Classification, Customer Churn, Customer Retention, Machine Learning.

DOI: https://doi.org/10.3126/jkbc.v7i1.88366

1. Introduction

Churn prediction is vital in the telecom sector due to intense competition and rising customer turnover driven by rapid technological advancement and wider service choices. Customer churn causes significant financial losses, making retention strategies essential for business continuity. While Telcos adopt approaches like acquiring new customers and promoting existing ones, research shows retention is most profitable, as maintaining current customers costs less than acquiring new ones (Garcia et al., 2017, Ahmad et al., 2019 and Xu et al., 2021). To reduce churn, Telcos use CRM systems to analyze customer patterns, predict potential churners, and identify reasons behind churn. Based on this, they apply targeted advertisements, packages, and campaigns to retain customers. Thus, Customer Churn Prediction (CCP) is a key element of CRM, helping not only in preventing churn but also in attracting new customers and sustaining competitiveness.

2. Problem Definition and Objectives

The telecom market is highly competitive, where retaining existing customers is more cost-effective than acquiring new ones. Early churn prediction is therefore crucial. The study in (Pustokhina et al., 2021) used ISMOTE (Improved Synthetic Minority Over-Sampling Technique) with Rain Optimization for sampling rate determination and WELM-based classification, though it lacked feature selection, limiting efficiency. Similarly, (Ullah et al., 2019) employed Bayesian networks and decision trees using AUC and Gini indicators but omitted clustering, which could improve accuracy and reduce false positives. This thesis proposes an improvised ensemble classification model to predict churn with minimal false positives and to identify the most influential churn attributes for CRM applications. Objectives include:

- Predicting churn with high accuracy.
- Identifying dominant churn attributes.

Machine learning (ML) methods like Random Forest, Decision Tree, GBM, and XGBoost have been widely used for churn prediction, while Social Network Analysis has enhanced feature engineering (Ahmad et al., 2019). Ensemble methods such as stacking and soft voting further improve accuracy on large datasets (Xu et al., 2021). Advanced approaches like ISMOTE-OWELM integrate class balancing and parameter optimization (Pustokhina et al., 2021), while Random Forest–based models identify key churn attributes in GSM datasets (Ullah et ai. 2019). Hybrid models (decision trees with neural networks (Hu et al., 2020)) and clustering-based techniques like ClusGBDT (Tang et al., 2020) also show strong performance. Comparative studies emphasize ensemble learning, hyperparameter tuning, and dimensionality reduction (Wang et al., 2020 and Lalwani et al., 2022). Recently, XGBoost has been applied to Nepalese telecom datasets, bridging global and local research contexts (Shrestha & Shakya, 2022). Collectively, these works highlight ensemble-based predictive modeling as a powerful tool for churn reduction strategies.

3. Methodology

3.1 Theoretical Formulations

Telecommunication companies face intense competition due to the growing number of operators and rapid technological progress. While customers benefit from more choices, churn has become a persistent challenge for telecom providers. Since retaining existing customers is far cheaper than acquiring new ones, companies must predict churn early, understand its causes, and design effective retention plans. One common strategy is targeted retention campaigns, where promotions or advantages are offered to at-risk customers identified through data analysis.

Traditionally, telecom firms used data mining to detect churn patterns, but this approach mainly provides quantitative insights. ML offers a more effective solution by learning churn behaviors from historical data and predicting future churners with higher accuracy. In the proposed model, raw telecom datasets undergo preprocessing, including handling missing values, type conversion, and balancing class distributions. Feature selection and engineering are then applied before classification using multiple ML algorithms. Finally, predictions from different classifiers are combined through ensemble learning, improving accuracy, balancing performance, and ensuring robust churn prediction. Evaluation metrics are then applied to validate the model's effectiveness.

3.2 System Block Diagram

The ensemble model for telecom churn prediction (Figure 1) includes **data preprocessing, classification, prediction ensemble, and evaluation blocks**, using telecom CDR data as input. The preprocessing stage handles missing values, data conversion, imbalance correction, and feature engineering to prepare data for ML models. The classification block applies algorithms such as **KNN, Logistic Regression, SVM, Random Forest, and Gradient Boosting** for churn prediction.

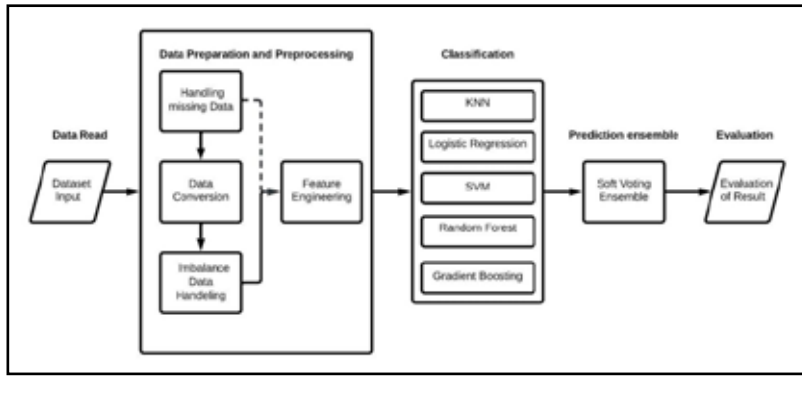

Figure 1: *System Block Diagram*

The main purpose of using multiple algorithms is to increase the prediction accuracy, make the model robust to incorporate the diverse nature of data, and balance the model performance.

The next block of the model is the prediction ensemble block where the system proposes an ensemble learning algorithm. The ensemble block handles the output from the different models from the classification block. The last block is the evaluation block which evaluates the model performance and performs the analysis of the overall system.

3.3 Instrumentation Requirements

The paper uses the free version of google **colab** which is an online cloud-based service for training and testing purposes, whereas python programming is used which contains a large set of libraries and frameworks for machine learning development.

Along with the python language, the Scikit-learn library which is considered a robust and most useful library for machine learning is used which provides the models that serve as a building block for the models.

3.4 Description of Algorithms

Algorithm selection is a challenge in many ML problems since a single algorithm is not perfect for all scenario. This is why a set of potential algorithms is selected for further evaluation.

KNN

KNN k-nearest neighbor is a supervised ML algorithm that can be used on both classification and regression problems.

KNN algorithm can be explained in the following steps:

1. Data load
2. Initialize K
3. To find the predicted class, iterate for the total number of training data points
   - 3.1. Calculate the distance between the test and each row of train data.
   - 3.2. Sort the distances in ascending order of distance values
   - 3.3. Get the highest k rows from the sorted array
4. Get the most frequent class of k rows
5. Return predict class

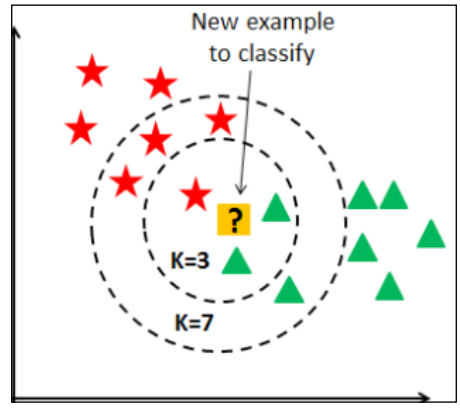

Figure 2: *KNN Classification and Selection of K (Note. Adapted from Photo by Sarang Anil Gotke on Kdnuggets)*

Logistic Regression

Logistic regression is a supervised ML algorithm for classification. This can be used simply and efficiently for classifying binary and linear classification problems. It could be easy to realize and can achieve better performance for linearly separable classes. Logistic regression smooth classifier is defined by sigmoid function as:

$$y = \text{Sigmoid}(x) = \frac{1}{1+e^{-x}}$$

$$y = \text{Sigmoid}(x) = \frac{e^{x}}{1+e^{x}}$$

15

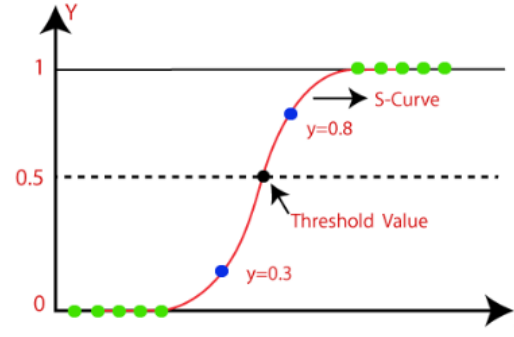

Figure 3: *logistic regression function (Note. Adapted from https://encord.com/blog/what-is-logistic-regression/)*

The sigmoid function maps the predicted values into probabilities within the ranges 0 and 1. Here the threshold is set to define the probability of either 0 or 1 such that values above the threshold move to 1 and values below the threshold move toward 0.

SVM

SVM is a supervised ML algorithm used mainly for binary classification. It represents data points in an n-dimensional space and finds the hyperplane that best separates the classes. The chosen hyperplane maximizes the margin, determined by **support vectors**—data points near the hyperplane that define its orientation.

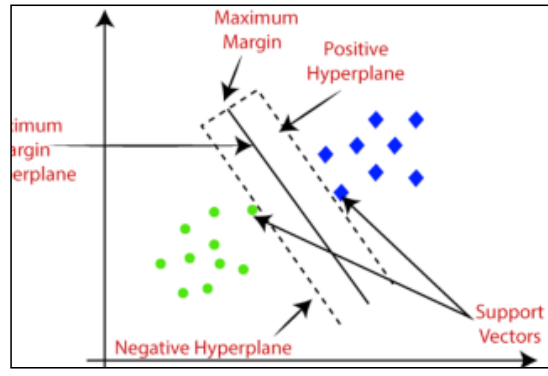

Figure 4: *Support Vectors and Hyperplane of SVM (Note. Adapted from https://python-geeks.org/support-vector-machine/)*

Random Forest

Random forest is a supervised ML algorithm it is used for classification as well as regression problems. In random forest decision trees are built for a random sample and majority voting was calculated for the classification problem, whereas averaging was done for the regression problem.

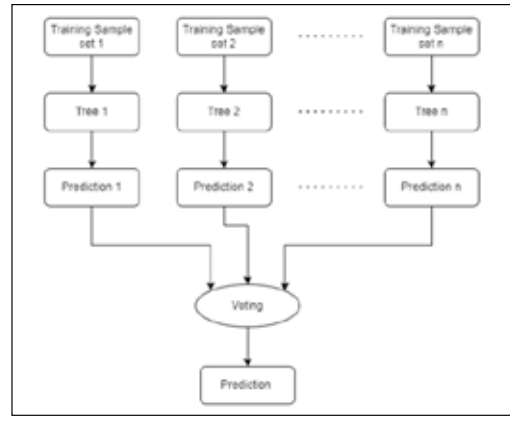

Figure 5: *Random Forest Algorithm Flow (Note. Adapted from https://www.geeksforgeeks.org/machine-learning/what-are-the-advantages-and-disadvantages-of-random-forest/)*

The Random Forest classifier consists of several decision trees on random subsets of the given dataset. It takes the average of those for the improvement of the predictive accuracy of data. It is based on the assumption that instead of relying on a single decision tree, taking a prediction from the number of decision trees and relying on the majority of votes from subsets it simply predicts the final output.

Gradient Boosting

Gradient boosting is one of the powerful ML algorithms which can be used for regression as well as classification problems. It consists of Gradient Descent and Boosting. Gradient

boosting represents a prediction model by combining considerably weak prediction models like decision trees.

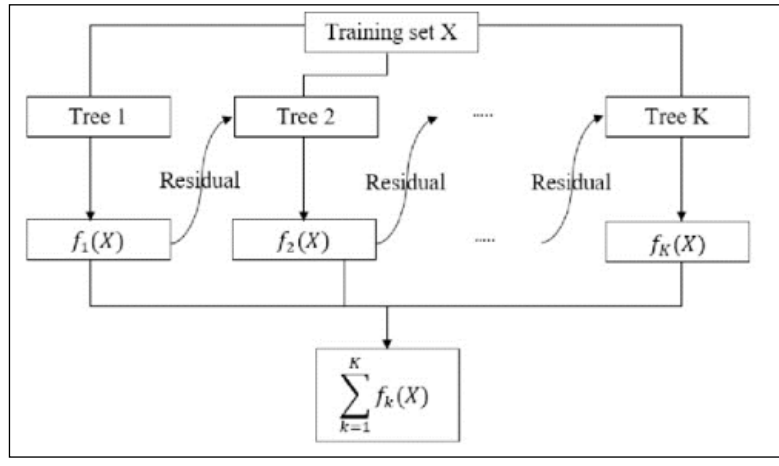

Figure 6: *Flow chart of Gradient Boosting (Note. Addapted from https://medium.com/@dishantkharkar9/about-boosting-and-gradient-boosting-algorithm-98dd4081ec18)*

The Gradient Boosting algorithm consists of the following 7 steps:

1. Average of the target label calculation.
2. Residual calculation.
3. Decision tree Construction.
4. Target level prediction using all decision trees within the ensemble.
5. New residuals computation.
6. Iterate steps 3 to 5 to a number specified by the hyperparameter.
7. After training, ensemble all of the trees to make a final prediction.

SMOTE

There arises a problem with imbalance classification as there are only a few examples of minority classes on the model to effectively learn the decision boundary. One of the ways to overcome this type of problem is oversampling the minority class which can be achieved by duplicating examples from the minority class in the training dataset before fitting the model, which balances the class distribution without any additional information to the model.

SMOTE is Synthetic Minority Oversampling Technique where the synthetic samples are generated for the minority class, which is a widely used approach to balance the imbalance present in the data (Chawla et al., 2002). The SMOTE algorithm also helps to overcome the overfitting problem posed by random oversampling. It also focuses on the feature space to generate new instances with the help of interpolation between the positive instances that lie together.

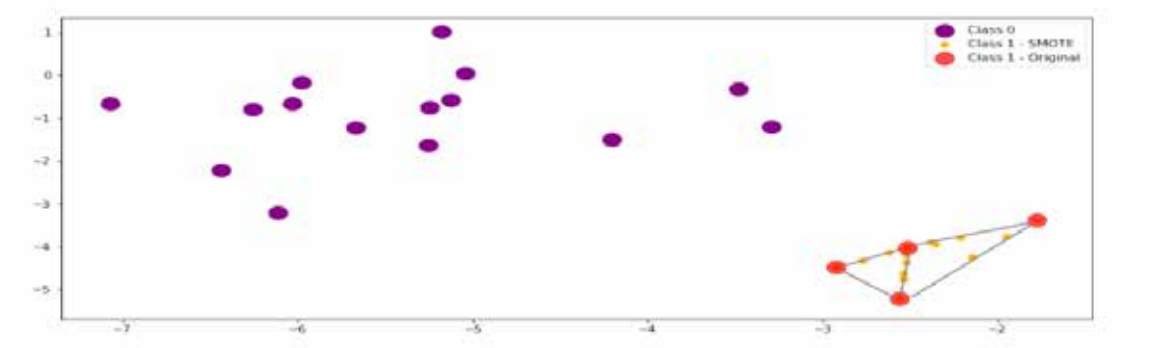

Figure 7: *Data balancing using SMOTE (Note. Adapted from https://neptune.ai/blog/how-to-deal-with-imbalanced-classification-and-regression-data)*

The SMOTE algorithm works in 4 steps:

- Choose a minority class of input data.
- Find KNN as specified by argument in SMOTE.
- Choose one neighbor and place synthetic points on a line that lies between that neighbor and the point of consideration.
- Until data get balanced, repeat.

Ensemble learning

An ensemble of classifiers combines individual classifier decisions using weighted or unweighted voting for classification or regression tasks. Ensemble methods improve performance, especially with unstable algorithms, by balancing bias and variance. Instead of relying on a single model, ensembles strategically combine multiple learners to enhance generalization and predictive accuracy. Weak learners can be boosted into strong learners through this approach.

Diversity among base models is crucial, achieved through different learning methods, datasets, or hybrid techniques. Ensemble learning has been studied for decades, with significant contributions from Hansen, Salamon, and Schapire. Common techniques for combining learners include: Average, Weight average, Dynamic soft voting, Majority voting, Winner takes all, Adaboost, Bagging

Among these ensemble learning techniques, the soft voting approach which assigns a larger weight to the dominant classifier that leads to the highest category being selected by summing up all the probabilities that are predicted by the model is used for the application, mathematically it can be represented as (Xu et al., 2021):

$$\hat{y} = argmax_i \sum_{j=1}^{m} w_j \, p_{ij}$$

Where argmax function outputs maximum, $\mathbf{w}_j$ is the weight which is associated with the prediction of the classifier, and $\mathbf{p}_{ij}$ is the probability associated with the respective classifier.

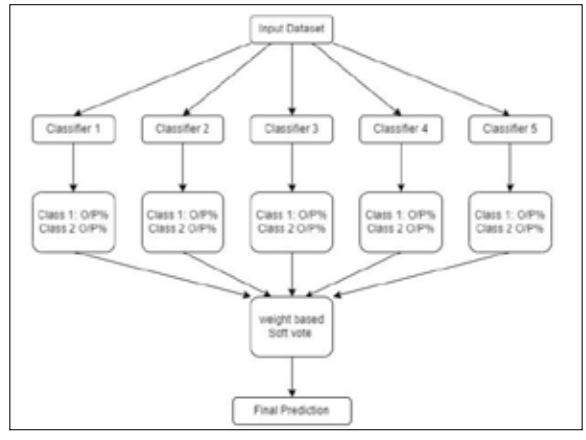

Figure 8: *Ensemble of multiple ML algorithms (Note. Adapted from https://www.intechopen.com/chapters/79087)*

The weight associated with the classifier can be assigned by using the validation scores of the model while the probability of the classifier can be assigned by evaluating the score on the validation set, once the weight and probability of the classifier are calculated, using those weight and probability of the classifier the ensemble prediction $\hat{\mathbf{y}}$ can be implemented.

Feature Engineering

Feature engineering is the process of selecting, altering, and transforming raw data into features suitable for supervised learning. It helps build better features so ML models perform well on new tasks by extracting characteristics and converting them into acceptable formats.

Since ML models process only numerical values, categorical data must be converted into numbers, a process called **categorical encoding**. Common methods include:

- **Label encoding**: replaces each category with a unique integer.
- **One-hot encoding**: creates dummy variables for each unique category.

Dataset Explanation: The dataset used by the model for Customer Churn Prediction in the telecom industry takes standard historical telecommunication datasets publicly available on Kaggle. The dataset is originally from IBM Cogonos Analytics where the data module is named Telco Customer Churn as a base sample. The data consists of information about a fictional telecommunication company from California, from where 7043 customers' data was taken in CVS format. The data contains information about the customers like those who have left or stayed on the service. The data also contain multiple demographic information, payment information, and many more which are present in Table 1 and 2.

index	customerID	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBackup	Devic
0	7590-VHVEG	Female	0	Yes	No	1	No	No phone service	DSL	No	Yes	No
1	5575-GNVDE	Male	0	No	No	34	Yes	No	DSL	Yes	No	Yes
2	3668-QPYBK	Male	0	No	No	2	Yes	No	DSL	Yes	Yes	No
3	7795-CFOCW	Male	0	No	No	45	No	No phone service	DSL	Yes	No	Yes
4	9237-HQITU	Female	0	No	No	2	Yes	No	Fiber optic	No	No	No

Table 1: *The sample data in the dataset*

The Dataset plot shows the distribution of the non-Churn variable is (72%) while the Churn variable is (27%) representing an asymmetric and unbalanced dataset. Also, the data set shows the following:

- The dataset has 7043 data with 21 significant attributes.
- The dataset is labeled dataset with the target: Churn Label (Yes/ No).
- The sample of the data is shown in table 1.

Elaboration of Working Principle: The proposed Customer Churn Prediction model in telecom consists of five blocks: **dataset input, data preparation and preprocessing, classification, prediction ensemble, and evaluation** (Figure 1).

Data is first collected from CDR, billing, and customer status databases to create the input dataset. The **preprocessing block** handles missing values, data conversion, imbalance correction, and feature engineering.

The **classification block** applies multiple ML algorithms (KNN, Logistic Regression, SVM, Random Forest, Gradient Boosting) to improve accuracy and stability compared to a single model. Outputs are then combined in the **ensemble block** for better prediction. Finally, the **evaluation block** verifies and validates model performance.

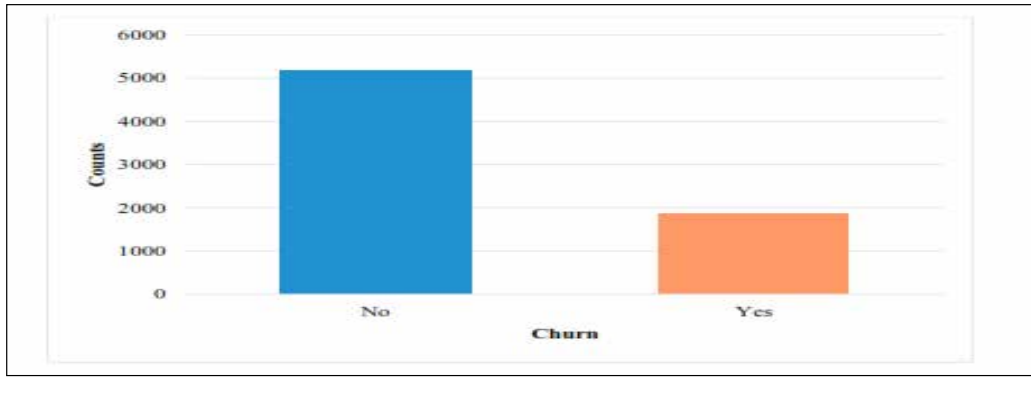

Figure 9: *Churn attribute distribution*

Table 2: *Attributes used in the dataset*

Attributes	Data types
customerID	object
gender	object
SeniorCitizen	int64
Partner	object
Dependents	object
tenure	int64
PhoneService	object
MultipleLines	object
InternetService	object
OnlineSecurity	object
OnlineBackup	object
DeviceProtection	object
TechSupport	object
StreamingTV	object
StreamingMovies	object
Contract	object
PaperlessBilling	object

4. Results and Discussion

Data Exploration

The first step in problem-solving is examining the data. Since a model is only as good as its data, preprocessing and analysis are crucial. Thus, this thesis begins by evaluating the data structure (Figure 10).

On evaluating the data structure, it is observed that the data consists of the 21 attributes of columns of each 7043 rows of customers. The 21 columns are shown in table 3.

	gender	SeniorCitizen	Partner	Dependents	tenure	PhoneService	MultipleLines	InternetService	OnlineSecurity	OnlineBackup	DeviceProtection
0	Female	0	Yes	No	1	No	No phone service	DSL	No	Yes	No
1	Male	0	No	No	34	Yes	No	DSL	Yes	No	Yes
2	Male	0	No	No	2	Yes	No	DSL	Yes	Yes	No
3	Male	0	No	No	45	No	No phone service	DSL	Yes	No	Yes
4	Female	0	No	No	2	Yes	No	Fiber optic	No	No	No

Figure 10: *Data Structure*

Table 3: *Data attributes column*

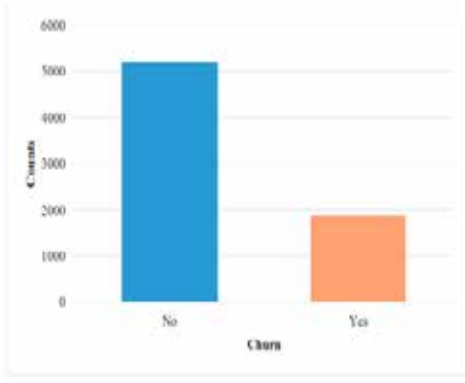

Table 4: ***Data Structure column and data type***

S.N.	Attributes	Non Null Count	Data types
1	customerID	7043	object
2	gender	7043	object
3	SeniorCitizen	7043	int64
4	Partner	7043	object
5	Dependents	7043	object
6	tenure	7043	int64
7	PhoneService	7043	object
8	MultipleLines	7043	object
9	InternetService	7043	object
10	OnlineSecurity	7043	object
11	OnlineBackup	7043	object
12	DeviceProtection	7043	object
13	TechSupport	7043	object
14	StreamingTV	7043	object
15	StreamingMovies	7043	object
16	Contract	7043	object
17	PaperlessBilling	7043	object
18	PaymentMethod	7043	object
19	MonthlyCharges	7043	float64
20	TotalCharges	7043	object
21	Churn	7043	object

Tables 4 and 5 show that the dataset has 21 attributes, including 19 independent variables. Most attributes are binary (e.g., Yes/No, Male/Female), while columns like tenure, monthly charges, and total charges are continuous with wide distributions. Hence, different attribute types require separate processing and evaluation.

Data Analysis and Cleaning

The dataset has 21 columns and 7043 observations with no null values, though total charges was incorrectly stored as an object and converted to numeric. After conversion, 11 missing values were found and dropped, leaving 7032 entries and 20 attributes. Final distribution shows 5163 non-churners and 1869 churners. For analysis, attributes are grouped into: demographics (gender, senior citizen, partner, dependents), account info (tenure, contract, billing, payment, charges), and services (phone, internet, security, backup, protection, support, streaming).

Table 5: ***Unique value of attributes***

S.N.	Attributes column	Unique Value
1	customerID	['7590-VHVEG' '5575-GNVDE' '3668-QPYBK' ... '4801-JZAZL' '8361-LTMKD''3186-AJIEK']
2	gender	['Female' 'Male']
3	SeniorCitizen	[0 1]
4	Partner	['Yes' 'No']
5	Dependents	['No' 'Yes']
6	tenure	[1 34 2 45 8 22 10 28 62 13 16 58 49 25 69 52 71 21 12 30 47 72 17 27 5 46 11 70 63 43 15 60 18 66 9 3 31 50 64 56 7 42 35 48 29 65 38 68 32 55 37 36 41 6 4 33 67 23 57 61 14 20 53 40 59 24 44 19 54 51 26 0 39]
7	PhoneService	['No' 'Yes']
8	MultipleLines	['No phone service' 'No' 'Yes']
9	InternetService	['DSL' 'Fiber optic' 'No']
10	OnlineSecurity	['No' 'Yes' 'No internet service']
11	OnlineBackup	['Yes' 'No' 'No internet service']
12	DeviceProtection	['No' 'Yes' 'No internet service']
13	TechSupport	['No' 'Yes' 'No internet service']
14	StreamingTV	['No' 'Yes' 'No internet service']
15	StreamingMovies	['No' 'Yes' 'No internet service']
16	Contract	['Month-to-month' 'One year' 'Two year']
17	PaperlessBilling	['Yes' 'No']
18	PaymentMethod	['Electronic check' 'Mailed check' 'Bank transfer (automatic)' 'Credit card (automatic)']
19	MonthlyCharges	[29.85 56.95 53.85 ... 63.1 44.2 78.7]
20	TotalCharges	['29.85' '1889.5' '108.15' ... '346.45' '306.6' '6844.5']
21	Churn	['No' 'Yes']

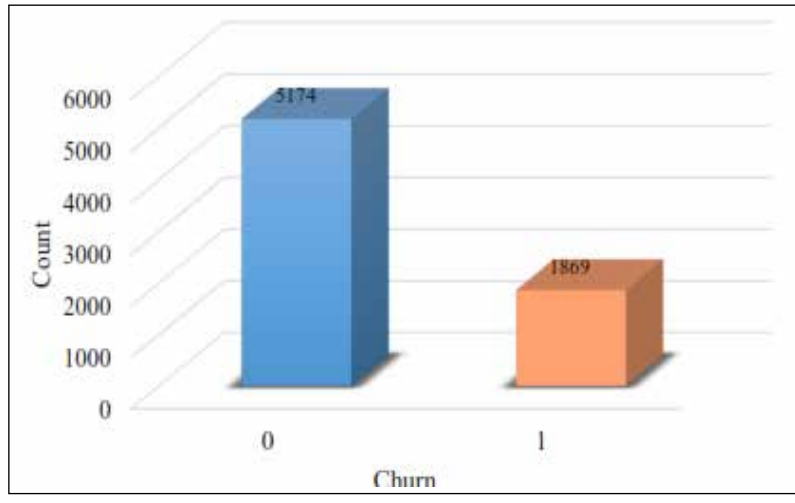

Figure 11: *Target variable distribution*

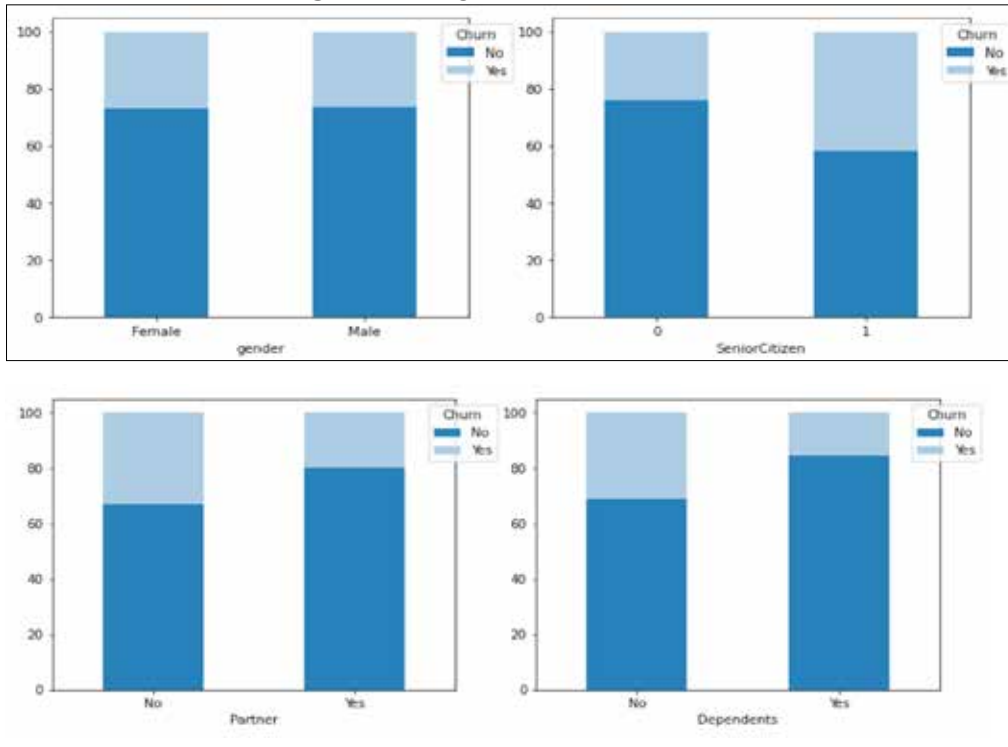

Figure 12: *Demographic Information Plot*

Figure 12 shows churn distribution across demographic attributes. Key findings:

1. Senior citizens churn at twice the rate of younger customers.
2. Gender has little predictive power, as churn is similar for males and females.
3. Customers with partners or dependents churn less than those without.

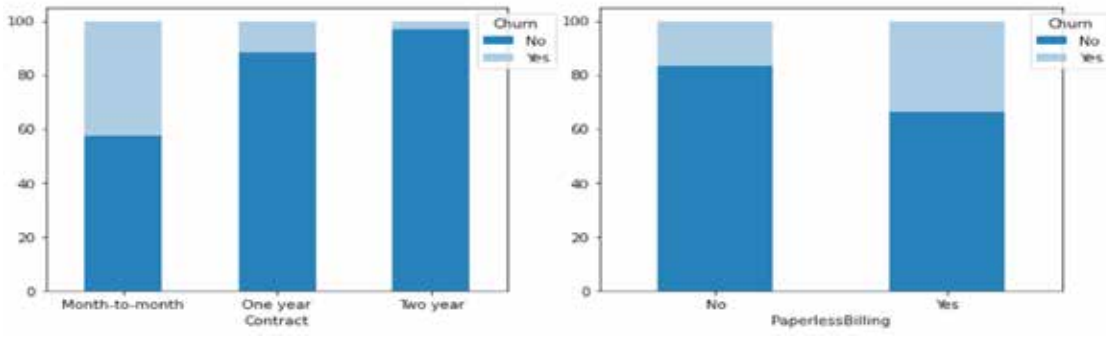

Figure 13: *Customer account information plot*

For the customer account information, we divide it into 2 groups for better visualization. The first group presents the portion of churn for attributes contract, paperless billing, and payment method as shown in figure 13.

Analysis of account attributes shows:

1. Monthly contract customers churn more than yearly or longer contracts.
2. Electronic check users are more likely to churn.
3. Paperless billing customers churn more than others.

Figures 14 – 16 further show churn patterns by tenure, monthly charge, and total charge.

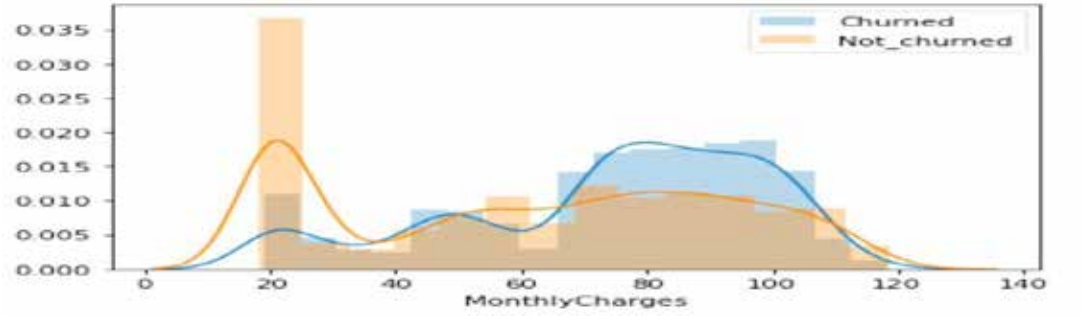

Figure 14: ***Histogram plot for monthly Charges***

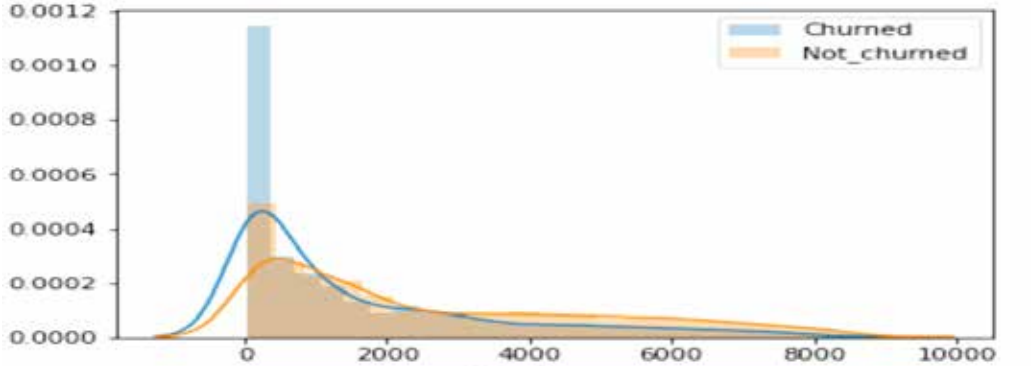

Figure 15: *Histogram plot for Total Charges*

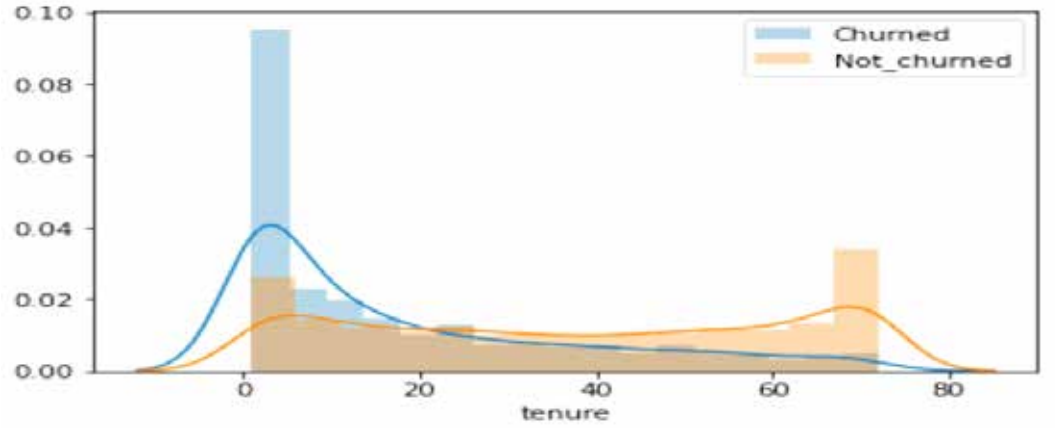

Figure 16: *Histogram plot for Tenure*

A pair plot (Figure 17) visualizes relationships among continuous customer account variables, displaying a matrix of their correlations.

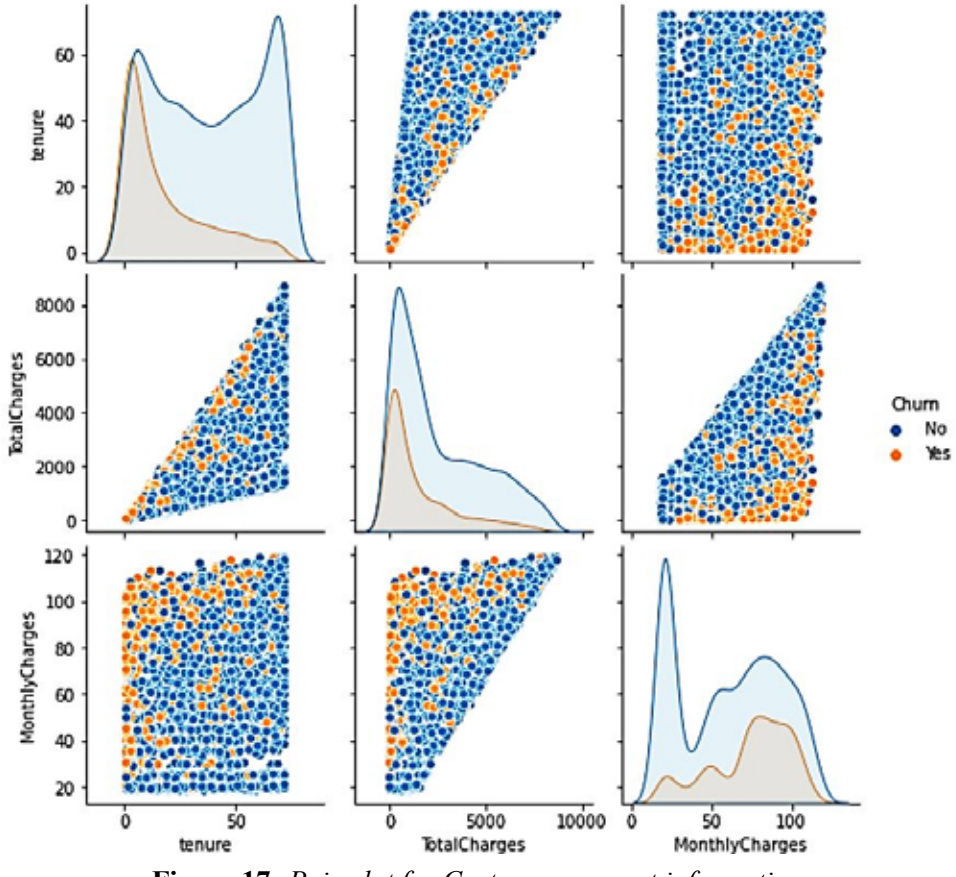

Figure 17: *Pair plot for Customer account information*

Histogram analysis of account attributes shows:

1. Churn decreases with longer tenure—new customers churn more.
2. Higher monthly charges lead to higher churn.
3. High total charges imply long contracts, so churn probability is lower.

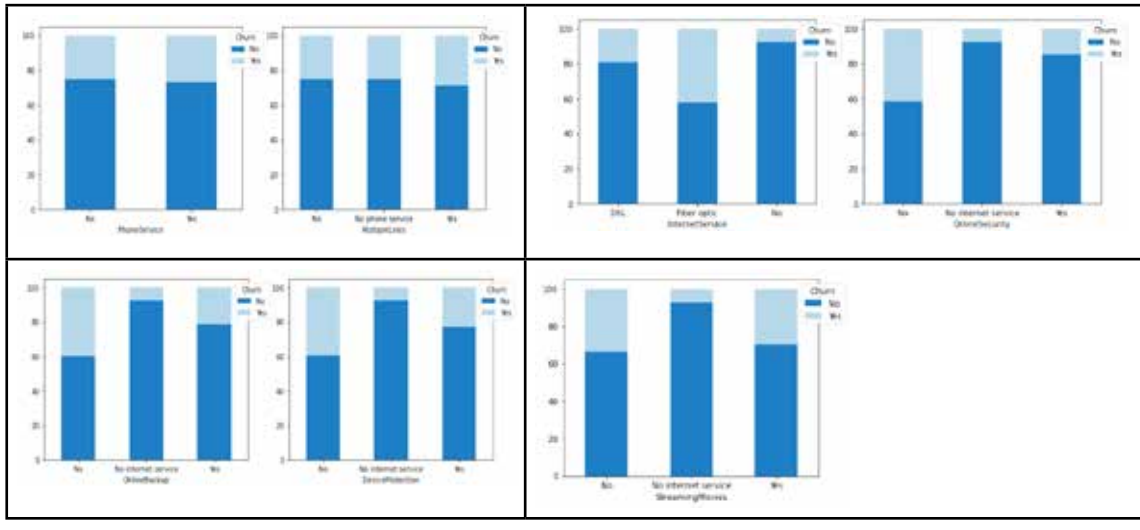

Figure 18: *Customer service information plot*

From Figure 18, service analysis shows:

1. Online security reduces churn.
2. Phone service and multiple lines have little predictive value.
3. Fiber optic internet users churn more.
4. Lack of tech support increases churn.
5. These plots help identify key churn attributes.

Feature Engineering

Feature engineering transforms raw data into useful features for supervised learning. Label encoding converts binary categorical variables (e.g., Gender, Partner, Churn) into 0/1 values. One-hot encoding handles multi-level categorical variables (e.g., Internet Service, Contract, Payment Method) by creating dummy variables as additional features.

Feature Importance

Mutual information measures dependency between variables using entropy. Higher values indicate stronger predictive power, while weak variables can be excluded to reduce model complexity. Table 6 presents the feature importance based on mutual information.

Table 6: *Feature importance*

FEATURE	Feature Importance
Contract	0.124351
OnlineSecurity	0.079717
TechSupport	0.077575
InternetService	0.067459
OnlineBackup	0.056784
DeviceProtection	0.053282
PaymentMethod	0.045434
StreamingMovies	0.038657
StreamingTV	0.038518
PaperlessBilling	0.030362
Dependents	0.023895
SeniorCitizen	0.020462
Partner	0.017794
MultipleLines	0.000863
PhoneService	0.000005
gender	0.000054

Data Balancing

The telecom dataset faces an imbalance issue, with churners being fewer than active customers. Two common techniques address this: under-sampling and oversampling. Since the dataset is small, synthetic oversampling using the SMOTE technique is applied. Table 7 shows the balanced data after SMOTE.

Table 7: *Distribution of training data before and after balancing*

BEFORE DATA BALANCING	
ATTRIBUTE	NUMBER
0	4113
1	1512
AFTER DATA BALANCING	
ATTRIBUTE	NUMBER
0	4113
1	4113

Model Selection

The dataset is split into dependent (churn/non-churn) and independent variables, then divided into training (80%) and testing (20%). Five ML algorithms are compared for churn prediction (Table 8 and figure 19). Results show SVM performs better without balancing, while Random Forest and Gradient Boosting give the best accuracy after applying data balancing.

Table 8: *Different ML Algorithm Accuracy*

ALGORITHM	ACCURACY (Without data balance)	ACCURACY (With data balance)
Logistic Regression	0.72	0.76
Support Vector Machine	0.79	0.76
K-Nearest Neighbours	0.76	0.76
Random Forest	0.78	0.77
Gradient Boosting	0.76	0.77

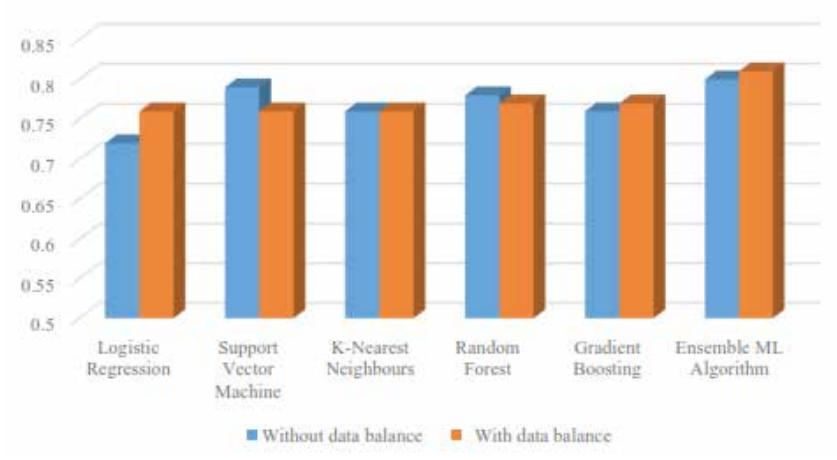

Figure 19: *Different ML Accuracy performance Ensemble learning*

Ensemble learning combines multiple ML models to improve accuracy. A soft voting ensemble is used, slightly enhancing performance (Table 8 and figure 20).

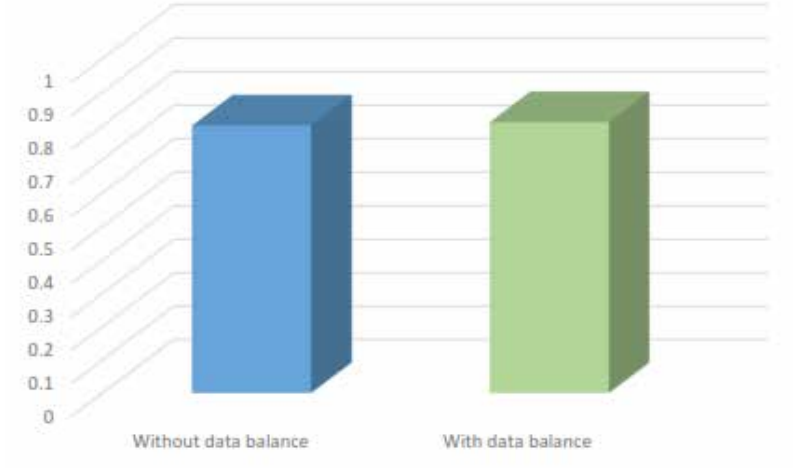

Figure 20: *Ensemble ML Output*

Table 8: ***Ensemble ML Algorithm Output***

ALGORITHM	ACCURACY (Without data balance)	ACCURACY (With data balance)
Ensemble ML Algorithm	0.80	0.81

Output discussion

The ensemble model achieved 81% churn prediction accuracy using an 80/20 train-test split, as shown in Table 9 and figure 21.

Table 9: *Output of Churn Prediction*

INDEX	ACTUAL	PREDICTION
2481	0	0
6784	0	0
6125	1	1
3052	0	0
4099	0	0
.....	.....	.....
1733	0	0
5250	0	1
5465	0	0
5851	0	0
3984	0	0

5. Discussion and Analysis

The model for telecom churn prediction followed the process of descriptive analysis, feature importance evaluation, train-test split, and implementation of multiple ML models. Their performance was compared using accuracy, precision, recall, F1-score, and confusion matrix. Data balancing was applied with SMOTE, and finally, an ensemble learning model was built, which outperformed individual models. Results showed SVM had the best individual performance, while the ensemble model achieved the highest overall accuracy and F1 score (Table 10).

Table 10: *Quantitative output of verification without data balancing*

Algorithms	Accuracy	Precision	Recall	F1 Score	Confusion Matrix
Logistic Regression	0.72	0.69	0.74	0.70	[[732,301],[79,295]]
Support Vector Machine	0.79	0.73	0.71	0.72	[[917,116],[174,200]]
K-Nearest Neighbours	0.76	0.70	0.68	0.69	[[890,143],[182,192]]
Random Forest	0.76	0.69	0.71	0.70	[[843,190],[145,229]]
Gradient Boosting	0.76	0.70	0.67	0.68	[[906,127],[198,176]]
Ensemble ML Algorithm	0.80	0.74	0.71	0.72	[[929,104],[176,198]]

Figure 21: *Output of verification without data balancing*

Figure 22 shows evaluation metrics without data balancing, where non-churn customers ('0') achieve higher precision, recall, and F1 scores than churn customers ('1'). Using Ensemble improves minority class prediction. To address imbalance, SMOTE was applied to generate synthetic minority samples, and the balanced dataset was used to train models, while testing was done on the original unbalanced data.

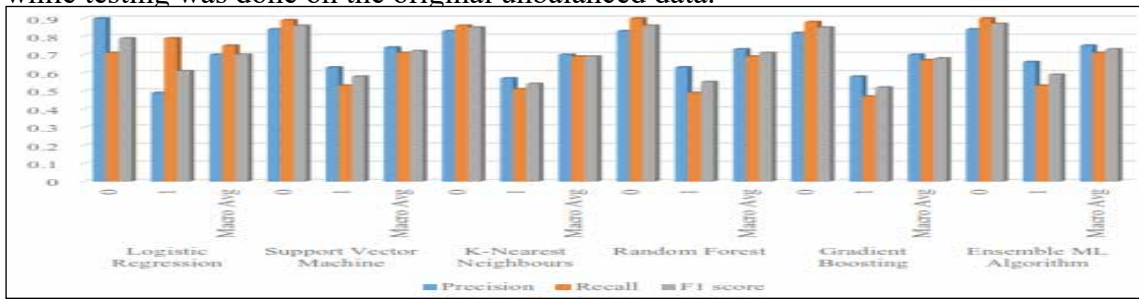

Figure 22: *Evaluation matrices class output without data balancing*

Table 11 shows that data balancing improved minority class prediction and F1 scores across all models. Random forest achieved the highest accuracy, while gradient boosting had a higher F1 score. The ensemble model further enhanced accuracy, precision, recall, and F1 score, highlighting the benefits of balancing data and ensemble learning.

Table 11: Quantitative output of verification after data balancing

Algorithms	Accuracy	Precision	Recall	F1 Score	Confusion Matrix
Logistic Regression	0.76	0.71	0.74	0.72	[[807,226],[106,268]]
Support Vector Machine	0.76	0.71	0.74	0.72	[[818,251],[112,262]]
K-Nearest Neighbours	0.75	0.71	0.76	0.72	[[770,263],[77,297]]
Random Forest	0.78	0.72	0.69	0.70	[[919,114],[189,185]]
Gradient Boosting	0.77	0.71	0.74	0.72	[[827,206],[117,257]]
Ensemble ML Algorithm	0.81	0.76	0.79	0.77	[[852,181],[86,288]]

Figure 23 shows that the ensemble model achieves the best overall performance, while data balancing reduces variation across models and evaluation metrics.

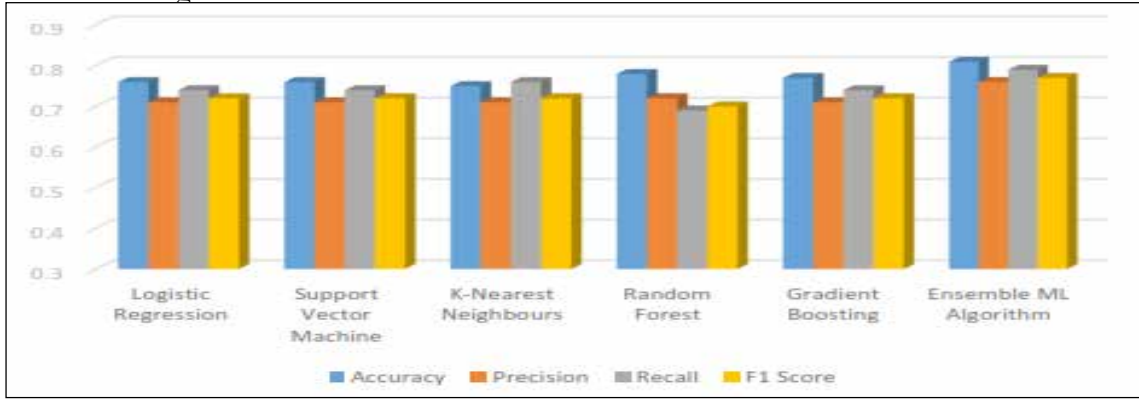

Figure 23: *Output of verification after data balancing*

Figure 24 shows that data balancing reduces variation in precision, recall, and F1 scores, while the ensemble model improves minority class prediction.

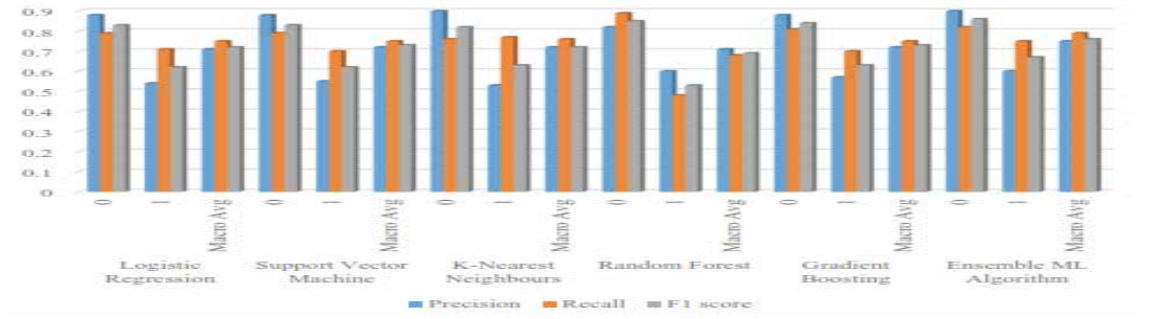

Figure 24: *Evaluation matrices class output after data balancing*

Error analysis and possible sources of error

The ensemble model did not significantly improve performance, likely due to data distribution. While accuracy, precision, recall, and F1 scores remained similar, the confusion matrix shows a slight gain in minority class prediction at the cost of majority class performance. Thus, ensemble learning can be useful when prioritizing minority class improvement.

6. Future Works

Different ML algorithms were applied to classify customers from CDR data, and were combined using ensemble learning for better accuracy and stability. The study identifies dominant churn attributes, suggesting further work on customer profiling and retention. Customers can be clustered using algorithms for group-based profiling, enabling tailored packages. Additionally, advanced ML can support personalized profiling for individual retention policies. Uplift modeling, an individual treatment model, may also be explored for improved churn analysis and industry impact.

7. Conclusion

Customer churn is when customers switch providers due to competitor offers or other issues, posing a major risk for businesses. Retaining customers is more cost-effective than acquiring new ones, but churn causes are complex, often linked to increased consumer choice and bargaining power. This thesis applies an Ensemble Machine Learning approach on classification models to accurately predict telecom churn and identify key churn attributes. The model is trained on CDR data to classify potential churners. Churn prediction is widely used across industries like telecom, banking, education, and healthcare to help firms take preventive measures.

Conflict of Interest / Ethical Approval

The authors declare that there is no conflict of interest regarding the publication of this paper.

References

García, D. L., Nebot, À. & Vellido, A. (2017). "Intelligent data analysis approaches to churn as a business problem: a survey," *Knowl. Inf. Syst.*, 51(3), 719–774, doi: 10.1007/S10115-016-0995-Z.

Ahmad, A.K., Jafar, A. & Aljoumaa, K. (2019). Customer churn prediction in telecom using machine learning in big data platform. J Big Data 6, 28

Xu, T., Ma, Y. & Kim, K. (2021). "Telecom Churn Prediction System Based on Ensemble Learning Using Feature Grouping," Applied Sciences, 11(11), 4742

Pustokhina, I. V., Pustokhin, D. A., Nguyen, P. T., Elhoseny, M. & Shankar, K. (2021). Multi-objective rain optimization algorithm with WELM model for customer churn prediction in telecommunication sector, Complex & Intelligent Systems,https://doi.org/10.1007/s40747-021-00353-6

Ullah, I., Raza, B., Malik, A. K., Imran, M., Islam, S. U. & Kim, S. W. (2019). "A Churn Prediction Model Using Random Forest: Analysis of Machine Learning Techniques for Churn Prediction and Factor Identification in Telecom Sector," *IEEE Access*, 7, 60134–60149, doi: 10.1109/ACCESS.2019.2914999.

Hu, X., Yang, Y., Chen, L. & Zhu, S. (2020). "Research on a Customer Churn Combination Prediction Model Based on Decision Tree and Neural Network," *2020 IEEE 5th Int. Conf. Cloud Comput. Big Data Anal. ICCCBDA,* 129–132, doi: 10.1109/ICCCBDA49378.2020.9095611.

Tang, Q., Xia, G. & Zhang, X. (2020). "A hybrid classification model for churn prediction based on customer clustering," *J. Intell. Fuzzy Syst.*, 39(1), 69–80, doi: 10.3233/JIFS-190677

Rahman, M. & Kumar, V. (2020). "Machine Learning Based Customer Churn Prediction In Banking," 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), 1196-1201, doi:10.1109/ICECA49313.2020.9297529.

Chauhan, S., Saini, S., Bathla, R. & Rana, A. (2020). "Application of Machine Learning to Predict Hospital Churning," ICRITO 2020 - IEEE 8th Int. Conf. Reliab.Infocom Technol. Optim., 33–372020, doi: 10.1109/ICRITO48877.2020.9197766.

Aulck, L., Velagapudi, N., Blumenstock, J. & West, J. (2022). "Predicting Student Dropout in Higher Education," Available: http://arxiv.org/abs/1606.06364.

Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. (2002). "SMOTE: Synthetic Minority Over-sampling Technique," J. Artif. Intell. Res., 16, 321–357

Huang, F., Xie, G. & Xiao, R. (2009). "Research on ensemble learning," 2009 Int. Conf. Artif. Intell. Comput. Intell. AICI 2009, 3, 249–252, doi: 10.1109/AICI.2009.235.

Wang, X., Nguyen, K., Hanoi, I. V., Nguyen, V. P. B. & Nguyen, B. P. (2020). "Churn Prediction using Ensemble Learning,", doi: 10.1145/3380688.3380710.

Lalwani, P., Mishra, M. K., Chadha, J. S. & Sethi, P. (2022). "Customer churn prediction system: a machine learning approach," Computing, 104(2), 271–294, doi: 10.1007/S00607-021-00908-Y.

Shrestha, S.M. & Shakya, A. (2022). "A customer churn prediction model using XGBoost for the telecommunication industry in Nepal", doi: 10.1016/j.procs.2022.12.067