



Machine-Learning Techniques for Customer Retention- A Comparative Study

¹Eneh, Kingsley Monday, ²Chinagolum Ituma, ³Emeka Agwu, & ⁴John Ndubuisi Ngene

^{1,4}Department of Computer Science, Enugu State University of Science and Technology, Nigeria

^{2,3}Department of Computer Science, Ebonyi State University, Nigeria

Accepted: December 8th, 2022

Published: December 12th, 2022

Citations - APA

Eneh, K. M., Ituma, C., Agwu, E., & Ngene, J. N. (2022). Machine-Learning Techniques for Customer Retention- A Comparative Study. *American Journal of Applied Sciences and Engineering*, 3(6), 51-66. DOI: <https://doi.org/10.5281/zenodo.7435601>

Customer retention is the capacity of a corporation, business, product, or service to maintain its customers over a defined period. Customers who stay with a brand are more likely to return, keep buying, or refrain from switching to another service or product altogether. Businesses must establish strategies for customer retention, though, as a result of heightened competition in the telecom sector. According to studies, businesses frequently employ a variety of strategies to lower the number of clients they lose over time and enhance their experiences to keep them coming back. Due to increasing client sophistication, these strategies have become less profitable over time. Because of this, companies continue to lose clients at a rapid rate. This article compares various machine learning methods in order to provide the most effective algorithm for predicting customer attrition. Experimentation serves as the methodology. We plan to construct and train the chosen machine learning model in order to verify the effectiveness of the methods. When compared to surveys and reviews, this approach produces more accurate predictions and results. Support vector machines, the k-nearest neighbor technique, random forests, logistic regression, decision trees, and XGBoost algorithms were all experimented. Using six machine learning methods, we were able to train six models for the prediction of churn in the telecoms service sector. After training and experimenting the models with the IBM telco dataset, the Random Forest model outperformed all others in our trial, with a greater accuracy rate of 80.57%, according to the findings.

←
ABSTRACT

Keywords: Customer Relationship Management (CRM); Customer Retention; Machine Learning Techniques; XGBoost Algorithm; Predicting Customers Attrition

Introduction

Customers are the core product and strength in terms of achievable success and revenue generation for businesses therefore, such businesses must do all they can to gain customer trust and satisfaction. Customer Relationship Management (CRM) provides for marketing by identifying and predicting both future customer needs, and understanding existing customer needs, in order to customer retention. Understanding the behavior patterns of customers is the basis of providing high-efficiency customer service, which translates into a high rate of customer retention. The management of customer relationships is a process that integrates human behavior as it relates to customer service. As a result, Customer Relationship Management systems use both business intelligence models as well as analytical to determine the most profitable customer audience group and attain greater customer retention rates.

By using online marketing strategies, these models are capable of identifying and predicting customers who are likely to subscribe based on analysis of such customers' behavioral, personal, and demographic data. Such data are used to provide tailored and customer-centric advertising campaigns to meet customer expectation.

There are four stages in business to customer relationship lifecycle. This includes:

- 1) identification; 2) attraction; 3) retention; and 4) development (Sabbeh, 2018).

Customer identification: The success and effectiveness of any marketing or advertising and sales endeavors depends largely on customer identification – ability to know the audience group. The target customers are those who are most likely to make purchases or subscribe to certain businesses service(s). Such group are usually given more attention.

Clustering algorithms can be used to aid customer segmentation - a process for grouping customers with similar attributes and interests. Clustering algorithm makes it easy for businesses to have good understanding of their customers in terms of dynamic behaviors and demography. Customers with similar attributes are usually in constant interaction with such business similarly, thus, by creating custom marketing strategy, this technique benefits businesses. (Gong, 2021)

Customer attraction: To determine the common characteristics that set customers apart within a segment, the identified customer segments and subgroups are examined. Targeted advertising and/or direct marketing are two examples of different marketing strategies that can be used to target different customer segments. (Kazemi & Babaei, 2011).

Customer retention: The ability of a business to keep customers as repeat customers and keep them from switching to a rival is known as customer retention. It shows whether the quality of your service and product is satisfactory to your current customers. Most subscription-based businesses and service providers depend on it to stay alive (Olson, 2020). Generally speaking, it is less expensive to keep your existing customers satisfied than to attract new ones. Gallo (2014) of The Harvard Business Review, claimed that acquiring new customers can cost five to twenty-five times more than keeping a current one. Therefore, businesses may adapt strategies to prevent customer churn (customers moving to other business or service competitors).

The percentage of customers who stopped being your customers at any given time is known as your customer churn rate. Your customer churn rate could rise for a number of reasons. (McEachern, 2022)

Churn rate is a key indicator most business organizations try to minimize. As a result, predicting churn a vital component of proactive measures for customer retention. (Vadakattu et. al, 2015). It's important to ensure your customer churn rate should be kept as low as possible. However, you must first determine your rate, though. The formular for the churn rate is given as

$$\text{Churn rate} = \frac{\# \text{ of customer at start of a given period} - \# \text{ of customer at the end of period}}{\# \text{ of customer at start of that period}} \times 100$$

One of the most significant data science applications in the business world is probably churn prediction. Its popularity stems from the fact that it has more noticeable effects and has a significant impact on the overall profits realized by the company. (Naik, 2021). Churn prediction can be performed using machine learning predictive model and data analysis. The churn model predicts, at the level of specific clients, their tendency (or predisposition) to exit. It tells us how likely it is that we will lose a particular customer in the future to competitors. (Votava, 2021).

Customer development: The Economic Development Collaborative (2021) states that customer development aids companies in obtaining distinctive insights that they can use to enhance customer service. The data can be used to ensure that you are making the right investments as well as to help you identify when customer needs have changed so that you can create new products. With customer development, you expand your clientele while creating a good or service that addresses their particular issues. Customer development is a separate process that you carry out in addition to product development; it does not replace it. Alvarez (2014). A customer base is only one aspect of customer development, though. One of the three pillars of a lean startup, along with customer development and agile engineering, is business model design. (Malsam, 2019). But building a customer base is only one aspect of customer development. Designing a business model, using agile engineering, and developing customers are the three pillars of a lean startup. Ries (2011) listed agile development and customer development as essential components of the Lean Startup. Following this, a larger audience accepted the term, and now widely used.

The Customer Development Process ensures that you create a product or service for which there is a real demand from the market.

The Customer Development Process is divided into 4 phases:

- i. Customer Discovery: test the customer’s problem, your solution and your business model.
- ii. Customer Validation: set up a scalable sales funnel.
- iii. Customer Creation: create demand for your product or service.
- iv. Company Building: evolve into a thriving company.

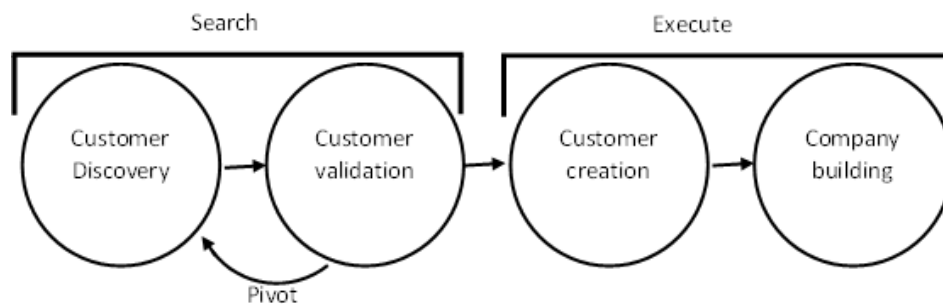


Fig 1. Customer development process

Customer retention and churn prediction is increasingly becoming a deliberate issue. Some studies have tried to show how machine learning can be used to predict customer churn. However, in this study, a comparative analysis of the machine learning algorithms for predicting churn and identify which of the algorithms provides the best churn prediction accuracy to help businesses strategize for customer retention.

Machine Learning Techniques for Customer Churn Prediction

Machine learning automates the creation of an analytical model. Using algorithms that iteratively study data allows machine learning systems to explore hidden patterns without explicitly planning where to look (Saran and Chandrakala, 2016). These methods are widely used to estimate customer likelihood (Jadhav and Pawar, 2011).

The algorithms used in this study include 1) The Support Vector Machine (SVM), 2) The k-nearest neighbors (KNN), 3) Random Forest, 4) Logistic Regression, 5) the Decision tree, and 6) XGBoost. They were all selected based on reviews of the literature on churn prediction. The following section describes the algorithms used in this study.

The Support Vector Machine (SVM)

The support vector machine algorithm is a popular supervised machine learning algorithm because it achieves a high level of accuracy while using less processing power. Although used for classification goals, it can also be helpful for tasks involving regression (Gandhi, 2018). Every data point is denoted by a point in n-dimensional space using the SVM algorithm, with values of each being the value of a specific coordinate (Sunil, 2017). Finding the hyper-plane that effectively divides the two classes is the next step in classification. Particular types of SVMs can be used for specific machine learning problems, for example, support vector regression or SVR (McGregor, 2020).

When creating applications that use predictive modeling, SVMs are essential. Understanding and using SVMs is simple. They provide a sophisticated machine learning algorithm that uses kernels to process linear and non-linear data. Every domain and real-world situation where handling data involve adding higher dimensional spaces finds applications for SVMs. This entail taking into account aspects like tuning hyper-parameters, choosing the kernel to run on, and allocating time and resources to the training phase, all of which contribute to the development of supervised learning models (Kanade, 2022).

The K-Nearest Neighbors (KNN)

The k-nearest neighbor (KNN) is a straightforward and simple supervised machine learning algorithm. The KNN is known to solve classification and regression issues. It assumes that related things are nearby. Alternatively put, related items are close to one another. (Onel, 2018).

The algorithms calculate the distances between a given data point and all other K numbers of data points close to it in the dataset, then cast their votes for the category with the highest frequency for that particular data point (Saji, 2021).

KNN is not a parametric learning algorithm. KNN does not assume anything about hidden data (Cássia Sampaio (n.d.)). Its' ability to not make assumptions concerning the underlying data is a feature since most real-world data doesn't follow any theoretical assumption, for example, linear separability, uniform distribution, etc. Usually, the Euclidean distance here is the distance separating two points. Thus, the resulting model is the labeled data placed in a space. This algorithm is widely known for various applications like genetics, forecasting, etc. The algorithm is best when more features are available and labeled.

The Random Forest

Random forest algorithm is a machine-learning technique used in predictive modeling and behavioral analysis based on decision trees. It is made up of several decision trees that are unique examples of classifying the input data in a random forest. The random algorithm treats each instance on an individual basis. It picks the instance with the most votes as the nominated prediction (CFI Team, 2022). It makes decisions based on predictions made by the decision trees. It makes predictions by averaging the output of different trees. The prediction accuracy increases by increasing the number of its trees (Mbaabu, 2020). The algorithm's ability to handle data sets with continuous variables, as with regression, and categorical variables, as with classification, is one of its most crucial features. It delivers better results when confronted with classification issues (Sruthi, 2021). Due to its use of a rule-based approach, data normalization is not a requirement. Findings by Song et. al (2021) indicate that the algorithm is more efficient in processing classification problems.

Logistic Regression

One of the best and most frequently used machine learning algorithms for binary classification problems is logistic regression. In light of how easily model parameters; can be fitted computationally and how easily the model is understood, logistic regression is common. IRLS is a popular method for fitting; logistic regression models because it is effective if the number of predictor variables is reasonable. Logistic regression models are significantly more accurate than black-box algorithms like artificial neural networks.

Interpretability enables the use of the model in situations where prediction accuracy and model interpretation are equally important (Makalic and Schmidt, 2010). The likelihood that a specific customer among a group of customers will stop using the service; is predicted using a model built using logistic regression and customer churn data (Sharma, 2021).

Decision Tree

One well-liked supervised machine learning algorithm is the decision tree. Decision Tree algorithms have many applications, including choosing a flight to take, the best hotel to stay at, forecasting employee turnover, identifying high occupancy dates for hotels, and identifying factors that improve gross margins for a chain of stores. (Qomariyah et al. 2020). It follows the conditions-based principle and has a flow akin to a tree structure. It is potent and has powerful algorithms in predictive analysis. According to Dwivedi (2020), its characteristics include internal nodes, branches, and a terminal node. They begin at one root, move toward decision nodes, and end in labeled leaves (Hajjej et al., 2022). Figure 2 depicts the decision tree's structure. Jiao and associates Due to its simple and understandable model, decision trees, a traditional machine learning algorithm, are used extensively. Although continuous variables are more common, decision trees can categorize data.

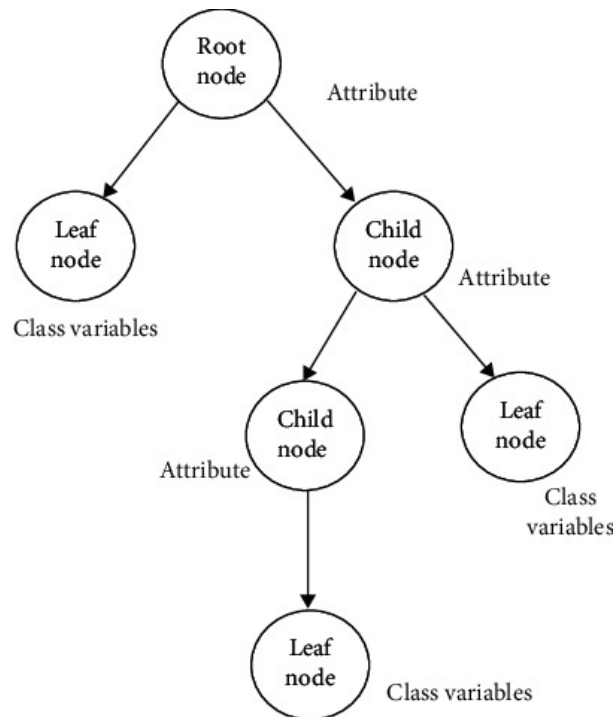


Fig. 2 Structure of decision tree

Researchers like decision trees because of their many benefits. Decision trees don't require much data preparation and are simple to comprehend and interpret, according to Wiryaseputra (2022). Categorical data and numerical data are both handled by decision trees. They function well even when the actual model from which the generated data

partially violates the assumptions. Due to their application to qualitative, quantitative, continuous, and discrete variables, decision trees are frequently used (Kaur and Vasana, 2006). Additionally, they are simple to interpret

The XGBoost

Extreme Gradient Boosting (also known as XGBoost) is a machine learning algorithm that uses boosting to improve performance. Since its inception, it has emerged as one of the most successful machine-learning algorithms, consistently outperforming most other algorithms like regular decision trees, random forest models, and logistic regression (Clarke, 2021). A weak learner is a term used to describe the basic model. They work on the assumption that poor learners make below-average predictions when acting alone but the best predictions when acting as a team. XGBoost sequentially adds models while building a strong learner off of weak learners. As a result, subsequent models in the chain can fix any errors introduced by models that aren't robust to create an optimized solution. Wasike 2021 refers to it as the ensemble. According to Jain (2016), advantages include regularization, parallelism, high flexibility, missing value handling, tree pruning, built-in cross-validation, and continuation of existing models.

Methodology

In this study, we used experimentation to determine which technique is best for predicting customer churn and to recommend action plans for customer retention. The approach was to first perform an exploratory data analysis (EDA) using the dataset obtained from Kaggle, followed by data preparation (preprocessing). Finally, we selected a few algorithms for evaluation. The best-performing algorithm was selected based on the results of the evaluation.

Experimental Environment

The environment we used for the research is PyCharm. It offers a wide range of essential tools for developers, tightly integrated to create a convenient environment for productive programming, web development, and data science projects. Python 3.9 was used for writing and interpreting the script in collaboration with libraries (table 1).

Table 1. List of python libraries and description

Library	Description and Use
<i>Pandas</i>	It has number of functions for analysing, exploring, cleaning and manipulating data, therefore, good for working with data sets
<i>Numpy</i>	It has all the mathematical functions for working with domains like linear algebra and matrices. Perfect for working with arrays.
<i>Matplotlib</i>	Graph plotting library used for visualization
<i>Seaborn</i>	Seaborn is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics
<i>Sklearn</i>	Used for machine learning projects. Sklearn comes bundled with functions for machine learning and modelling such as Classification, Clustering, Dimensionality Reduction, and Regression

Data Set

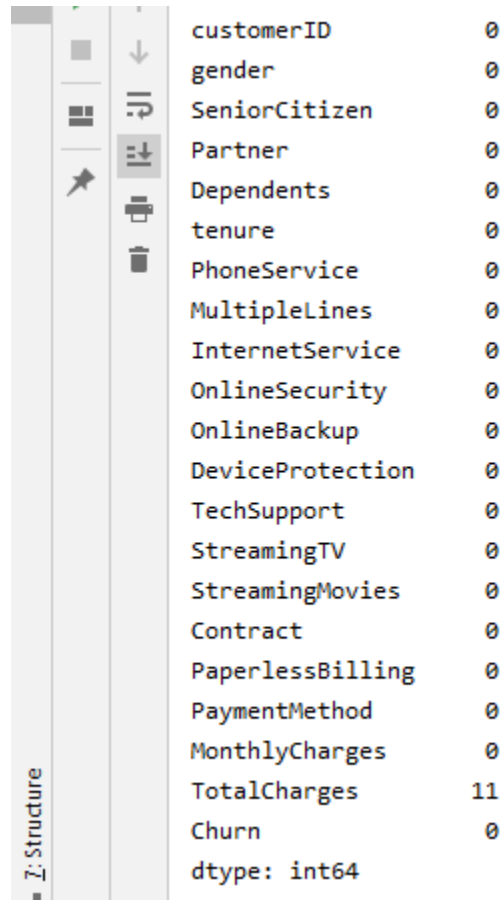
The data used in this study is a churn dataset of telecommunication customers created by IBM. The dataset was made available on Kaggle. The goal is to help researchers predict customer behavior by analyzing all relevant customer data to develop customer-centric loyalty programs.

Exploratory Data Analysis (EDA)

EDA is an approach data scientists use while investigating a given dataset. With EDA, it is simple to analyze aspects of a dataset that machines might not be able to comprehend. It helps with dataset preprocessing from a human perspective to see what the data can tell us beyond formal modeling. It can manipulate the data for better results.

The data in its raw form is full of noise, outliers, and missing values. Therefore, the data needs to be preprocessed before being fitted into a model. The preprocessing of raw data involves the following steps:

1. Data cleansing: Missing and erratic values are a natural part of raw data. Handling this noisy, inconsistent data is what data cleaning entails. Filling in the missing values, ignoring the tuples, clustering, and regression are all methods for handling data cleaning. We start by duplicating telecommunications data for further preprocessing before cleaning up our data. The "Total Charges" were then changed to numeric types.



customerID	0
gender	0
SeniorCitizen	0
Partner	0
Dependents	0
tenure	0
PhoneService	0
MultipleLines	0
InternetService	0
OnlineSecurity	0
OnlineBackup	0
DeviceProtection	0
TechSupport	0
StreamingTV	0
StreamingMovies	0
Contract	0
PaperlessBilling	0
PaymentMethod	0
MonthlyCharges	0
TotalCharges	11
Churn	0
dtype: int64	

Fig. 3 TotalCharge missing values after conversion to numerical type

We observed that TotalCharges had lots of missing values up to 11 from the output n see that there are a lot of missing values in the Total charges' column. Since the missing values constitutes only 0.15% of the entire dataset, they were replaced with other values using the average strategy.

2. Data Transformation: To effectively visualize and use the data, we transformed it into forms suitable for exploration processes. Some examples of data transformation techniques include Normalization, discretization, and feature selection.

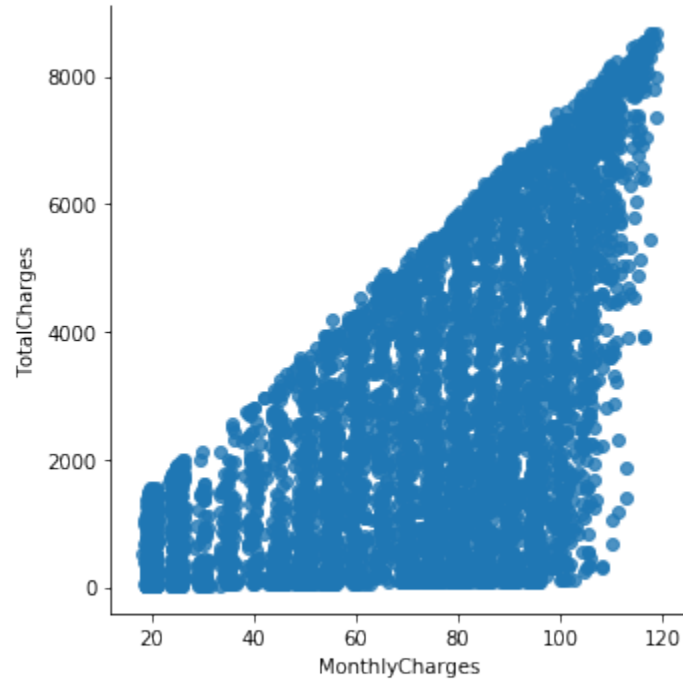


Fig 4. Regression plot between charges.

From the graph, we can see a positive correlation between monthly charge and total charge. It is clear that the total charges increase with every increase in monthly charge.

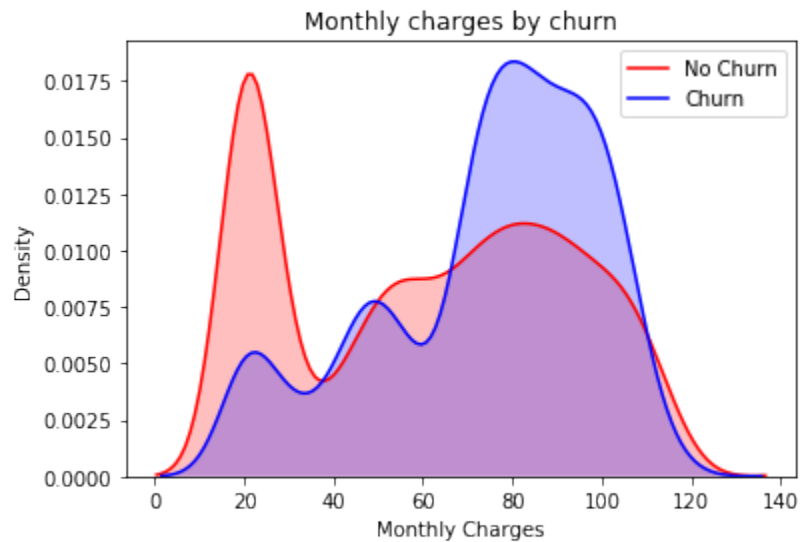


Fig 5. Measure the churn by monthly charges

The graph in figure gives us an insight into customer behavior towards increase in charges

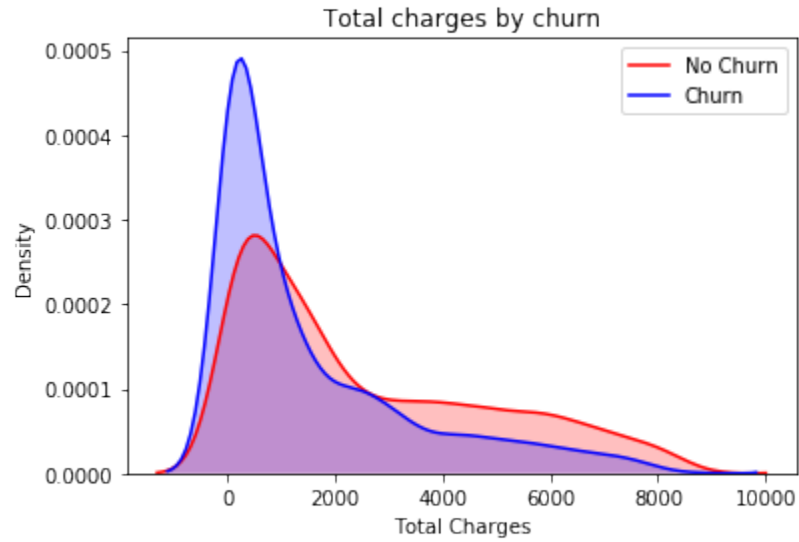


Fig 6. Measure the churn by total charges

Surprisingly, we observed there is more churn even with lower charges. Looking at Monthly Charges, Total Charges, and Tenure, we confirm their links to high churn. It's also clear that we get a low Total Charge when the tenure is lower at higher Monthly Charge.

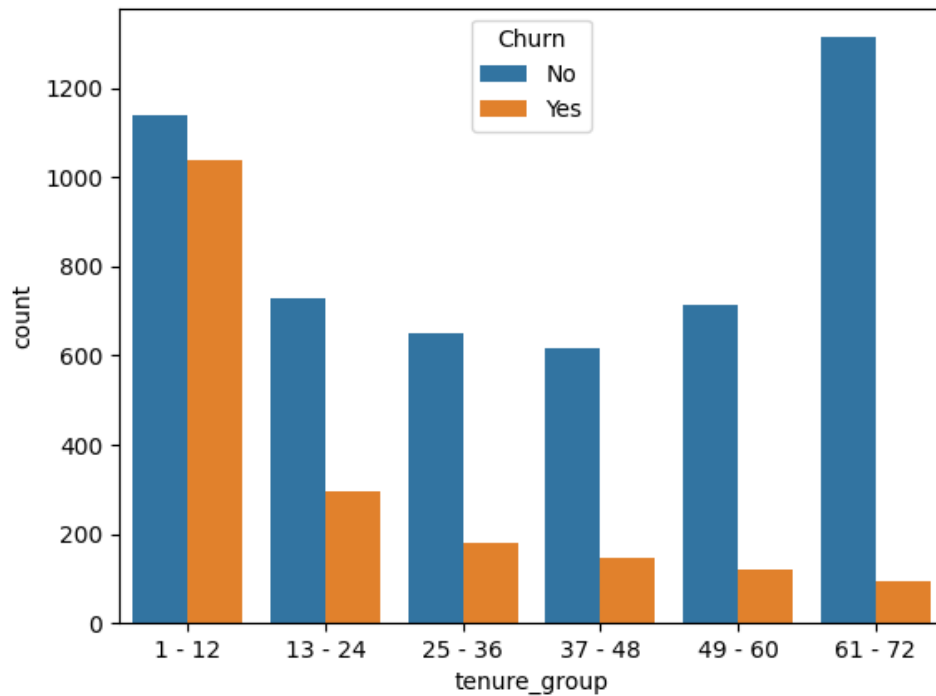


Fig 7. Measure the churn by tenure group

3. Data Reduction: Feature subset selection, dimensionality reduction, and other techniques are used in data reduction to manage massive amounts of data. When modeling, data reduction is selected because working with large amounts of data can be challenging. We chose PCA to reduce the size.

```
----- Null entries are resolved -----
gender                0
SeniorCitizen        0
Partner              0
Dependents           0
tenure               0
PhoneService         0
MultipleLines        0
InternetService      0
OnlineSecurity       0
OnlineBackup         0
DeviceProtection     0
TechSupport          0
StreamingTV          0
StreamingMovies      0
Contract             0
PaperlessBilling     0
PaymentMethod        0
MonthlyCharges       0
TotalCharges         0
Churn                0
dtype: int64
```

Fig 8. Features after missing value treatments

Prediction

The results of model prediction from the preprocessed data after fitting the data set into each selected algorithms is given in this section. Six algorithms namely Support Vector Machine, the k-nearest neighbors, Random Forest, Logistic Regression, the Decision tree, and XGBoost were considered for comparison. The performance metrics of each model is noted and studied to compare the models to find out the best performing algorithm.

```
# SUPPORT VECTOR MACHINE

from sklearn.svm import SVC

svc_model = SVC(random_state = 42)

t0 = time.time()
svc_model.fit(X_train,y_train)
t1 = time.time()

accuracy_svc = svc_model.score(X_test,y_test)

print("\n\n-----")
print("Accuracy of Support Vector Machine: ", accuracy_svc)
print("Execution time: %0.8f seconds" % (t1 - t0))
print("-----")

svc_prediction = svc_model.predict(X_test)

plt.figure(13)
plt.figure(figsize=(4,3))
sns.heatmap(confusion_matrix(y_test, svc_prediction),
            annot=True, fmt = "d", linecolor="k", linewidths=3)

plt.title("Support Vector Machine Confusion Matrix", fontsize=16)
plt.show()

model_comparison = model_comparison.append(
    {'Model': 'Support Vector Machine',
     'Accuracy': accuracy_svc,
     'Execution time': '%0.8f seconds' % (t1 - t0)}, ignore_index = True)

-----
Accuracy of Support Vector Machine:  0.7872037914691943
Execution time: 1.51999807 seconds
-----
```

Fig 9. Code of Support vector machines

```
# K-NEAREST NEIGHBOR (KNN)

from sklearn.neighbors import KNeighborsClassifier

knn_model = KNeighborsClassifier(n_neighbors = 10)

t0 = time.time()
knn_model.fit(X_train,y_train)
t1 = time.time()

accuracy_knn = knn_model.score(X_test,y_test)

print("\n\n-----")
print("Accuracy of K-Nearest Neighbor: ", accuracy_knn)
print("Execution time: %0.8f seconds" % (t1 - t0))
print("-----")

knn_prediction = knn_model.predict(X_test)

plt.figure(14)
plt.figure(figsize=(4,3))
sns.heatmap(confusion_matrix(y_test, knn_prediction),
            annot=True, fmt = "d", linecolor="k", linewidths=3)

plt.title("K-Nearest Neighbor Confusion Matrix", fontsize=16)
plt.show()

model_comparison = model_comparison.append(
    {'Model': 'K-Nearest Neighbor',
     'Accuracy': accuracy_knn,
     'Execution time': '%0.8f seconds' % (t1 - t0)}, ignore_index = True)

-----
Accuracy of K-Nearest Neighbor:  0.7819905213270142
Execution time: 0.00901699 seconds
-----
```

Fig 10. Code of K-Nearest Neighbor

```
# RANDOM FOREST

from sklearn.ensemble import RandomForestClassifier

random_forest_model = RandomForestClassifier(n_estimators=500,
                                           oob_score = True, n_jobs = -1,
                                           random_state=42, max_features = "auto",
                                           max_leaf_nodes = 30)

t0 = time.time()
random_forest_model.fit(X_train, y_train)
t1 = time.time()

accuracy_random_forest = random_forest_model.score(X_test, y_test)

print("\n\n-----")
print("Accuracy of Random Forest: ", accuracy_random_forest)
print("Execution time: %0.8f seconds" % (t1 - t0))
print("-----")

random_forest_prediction = random_forest_model.predict(X_test)

plt.figure(12)
plt.figure(figsize=(4,3))
sns.heatmap(confusion_matrix(y_test, random_forest_prediction),
            annot=True, fmt = "d", linecolor="k", linewidths=3)

plt.title("Random Forest Classifier Confusion Matrix", fontsize=16)
plt.show()

model_comparison = model_comparison.append(
    {'Model': 'Random Forest Classifier',
     'Accuracy': accuracy_random_forest,
     'Execution time': '%0.8f seconds' % (t1 - t0)}, ignore_index = True)

-----
Accuracy of Random Forest:  0.8056872037914692
Execution time: 2.34610057 seconds
-----
```

Fig 11. Code of Random Forest

```
# LOGISTIC REGRESSION
from sklearn.linear_model import LogisticRegression

logistic_regression_model = LogisticRegression(solver='lbfgs', max_iter=1000)

# mesurement of execution time
import time
t0 = time.time()

accuracy_logistic_regression = logistic_regression_model.score(X_test,y_test)
print("\n\n-----")
print("Accuracy of Logistic Regression: ", accuracy_logistic_regression)
print("Execution time: %0.8f seconds" % (t1 - t0))
print("-----")

-----
Accuracy of Logistic Regression:  0.8052132701421801
Execution time: 1.81107426 seconds
-----
```

Fig 12. Code of Logistic Regression

```
# DECISION TREE

from sklearn.tree import DecisionTreeClassifier

decision_tree_model = DecisionTreeClassifier()

t0 = time.time()
decision_tree_model.fit(X_train,y_train)
t1 = time.time()

accuracy_decision_tree = decision_tree_model.score(X_test, y_test)
print("\n\n-----")
print("Accuracy of Decision Tree: ", accuracy_decision_tree)
print("Execution time: %0.8f seconds" % (t1 - t0))
print("-----")

# Decision Tree Classifier gives very low accuracy score.

decision_tree_prediction = decision_tree_model.predict(X_test)

plt.figure(11)
plt.figure(figsize=(4,3))
sns.heatmap(confusion_matrix(y_test, decision_tree_prediction),
            annot=True, fmt="d", linecolor="k", linewidths=3)

plt.title("Decision Tree Classifier Confusion Matrix", fontsize=16)
plt.show()

model_comparison = model_comparison.append(
    {'Model': 'Decision Tree Classifier',
     'Accuracy': accuracy_decision_tree,
     'Execution time': '%0.8f seconds' % (t1 - t0)}, ignore_index = True)

-----
Accuracy of Decision Tree:  0.7488151658767772
Execution time: 0.05465221 seconds
-----
```

Fig 13. Code of Decision Tree

```
# EXTREME GRADIENT BOOSTING (XGBOOST)

from xgboost import XGBClassifier

xgb_model = XGBClassifier()

t0 = time.time()
xgb_model.fit(X_train, y_train)
t1 = time.time()

accuracy_xgb = xgb_model.score(X_test,y_test)

print("\n\n-----")
print("Accuracy of XGBoost Classifier: ", accuracy_xgb)
print("Execution time: %0.8f seconds" % (t1 - t0))
print("-----")

prediction_xgb = xgb_model.predict(X_test)

plt.figure(15)
plt.figure(figsize=(4,3))
sns.heatmap(confusion_matrix(y_test, prediction_xgb),
            annot=True, fmt = "d", linecolor="k", linewidths=3)

plt.title("XGBoost Classifier Confusion Matrix", fontsize=16)
plt.show()

model_comparison = model_comparison.append(
    {'Model': 'XGBoost Classifier',
     'Accuracy': accuracy_xgb,
     'Execution time': '%0.8f seconds' % (t1 - t0)}, ignore_index = True)

-----
Accuracy of XGBoost Classifier:  0.7876777251184834
Execution time: 0.63899517 seconds
-----
```

Fig 14. Code of XGBoost Classifier

Model comparison

Table 2: Model comparisons

<i>Index</i>	<i>Model</i>	<i>Accuracy Score</i>	<i>Execution Time</i>
0	Support Vector Machine	0.7872037914691943	1.51999807 seconds
1	The k-nearest neighbors	0.7819905213270142	0.00901699 seconds
2	Random Forest	0.8056872037914692	2.34610057 seconds
3	Logistic Regression	0.8052132701421801	1.81107426 seconds
4	Decision tree	0.7488151658767772	0.05465221 seconds
5	XGBoost	0.7876777251184834	0.63899517 seconds

Accuracy and Execution time were the only criteria were used to for our model comparisons. Judging by the results of our model execution, some models performed optimally with good execution time. With accuracy score of 0.8056872037914692, the Random Forest clearly outperformed the Support Vector Machine, The k-nearest neighbors, Logistic Regression, Decision tree and the XGBoost. However, the difference between the Random Forest and Logistic Regression is 0.000474. so, we can say that Logistic Regression can comfortably substitute the Random Forest.

There are times where execution time may not be a consideration while high accuracy is needed. It is also, important to note that false positive may not be a problem where quick decisions are needed.

Conclusion

In recent times, more service providers have emerged than ever, so every business must know why customers reject their offerings to maintain business growth in a cutthroat market. Although business loss is inevitable, mitigating such losses by attracting and keeping customers is possible. Any area of business management can perform better with refined methods and improvements to current ones. Churn prediction is one technique for reducing business losses and customer churn. So, in this article, we've chosen to focus on churn prediction in the context of telecommunication services. We sourced data from Kaggle, an open-source data source. EDA examines the information. Following preprocessing, 30% of the data was for testing. The remaining 70% was for training. First, the study conducted a brief review of the relevant literature in our chosen field. We compared machine-learning algorithms, including the support vector machine, the k-nearest neighbors, random forest, logistic regression, decision tree, and XGBoost. The aim was to determine which algorithm would perform the best. The Random Forest model performance was the best, with an accuracy rate of 80.57%, according to the experimental findings.

References

- Alvarez C. (2014). *Lean Customer Development: Building Products Your Customers Will Buy*. 1st Edition. O'Reilly Media, Inc. ISBN: 9781449356354
- CFI Team (2022). Random Forest: A combination of decision trees that can be modeled for prediction and behavior analysis. <https://corporatefinanceinstitute.com/resources/data-science/random-forest/corporatefinanceinstitute>
- Clarke M. (2021). How to create a classification model using XGBoost in Python: Learn how to create a classification model using XGBoost and scikit-learn in Python by classifying wine types from their features. <https://practicaldatascience.co.uk/machine-learning/how-to-create-a-classification-model-using-xgboost>
- Dwivedi R. (2020). Introduction to Decision Tree Algorithm in Machine Learning. <https://www.analyticssteps.com/blogs/introduction-decision-tree-algorithm-machine-learning>
- EDC (2021). Why Customer Development Is Important for Small Businesses <https://edcollaborative.com/blog/why-customer-development-is-important-for-small-businesses/>
- Gallo A. (2014). The Value of Keeping the Right Customers. <https://hbr.org/2014/10/the-value-of-keeping-the-right-customers>

- Gandhi R. (2018). Support Vector Machine — Introduction to Machine Learning Algorithms: SVM model from scratch. <https://towardsdatascience.com/support-vector-machine-introduction-to-machine-learning-algorithms-934a444fca47>
- Gong D. (2021). Clustering Algorithm for Customer Segmentation: A Step-by-Step Guide to K-Means Clustering. In Towards Data Science. <https://towardsdatascience.com/clustering-algorithm-for-customer-segmentation-e2d79e28cbc3>
- Hajjej, F., Alohal, M. A., Badr, M., & Rahman, M. A. (2022). A Comparison of Decision Tree Algorithms in the Assessment of Biomedical Data. *BioMed research international*, 2022, 9449497. <https://doi.org/10.1155/2022/9449497>
- Jain, A. (2016). Complete Guide to Parameter Tuning in XGBoost with codes in Python <https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/>
- Jain, H., Yadav, G., Manoov, R. (2021). Churn Prediction and Retention in Banking, Telecom and IT Sectors Using Machine Learning Techniques. In: Patnaik, S., Yang, X.S., Sethi, I. (eds) *Advances in Machine Learning and Computational Intelligence. Algorithms for Intelligent Systems*. Springer, Singapore. https://doi.org/10.1007/978-981-15-5243-4_12
- Jiao S. R, Song J, and Liu B. (2020). A Review of Decision Tree Classification Algorithms for Continuous Variables. *Journal of Physics: Conference Series*, Volume 1651, The 2020 second International Conference on Artificial Intelligence Technologies and Application (ICAITA) DOI 10.1088/1742-6596/1651/1/012083
- Kanade V. (2022). What Is a Support Vector Machine? Working, Types, and Examples. <https://www.spiceworks.com/tech/big-data/articles/what-is-support-vector-machine/>
- Kaur, H., & Wasan, S. K. (2006). Empirical study on applications of data mining techniques in healthcare. *Journal of Computer science*, 2(2), 194-200. DOI: 10.3844/jcssp.2006.194.200.
- Kazemi, A., & Babaei, M. (2011). Modelling customer attraction prediction in customer relation management using decision tree: A data mining approach. *Journal of Optimization in Industrial Engineering*, 4(9), 37–45.
- Makalic, E., Schmidt, D.F. (2010). Review of Modern Logistic Regression Methods with Application to Small and Medium Sample Size Problems. In: Li, J. (eds) *AI 2010: Advances in Artificial Intelligence*. AI 2010. *Lecture Notes in Computer Science*, 6464. 213–222. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-17432-2_22
- Malsam W. (2019). The Importance of Customer Development for Startups <https://www.projectmanager.com/blog/importance-of-customer-development>
- Mbaabu O. (2020). Introduction to Random Forest in Machine Learning. <https://www.section.io/engineering-education/introduction-to-random-forest-in-machine-learning/>
- McEachern A. (2022). What Is Customer Retention? Definition and Guide. <https://www.shopify.com/ng/blog/customer-retention-strategies>
- McGregor M. (2020). SVM Machine Learning Tutorial – What is the Support Vector Machine Algorithm, Explained with Code Examples. <https://www.freecodecamp.org/news/svm-machine-learning-tutorial-what-is-the-support-vector-machine-algorithm-explained-with-code-examples/>
- Naik S. (2021). Churn Prediction- Commercial use of Data Science. <https://www.analyticsvidhya.com/blog/2021/08/churn-prediction-commercial-use-of-data-science/>
- Olson S. (2020). What is customer retention? Importance, metrics & definition: Customer retention is cheaper than customer acquisition. Here are a few ways to improve your retention strategies. <https://www.zendesk.com/blog/customer-retention/>
- Onel H. (2018). Machine Learning Basics with the K-Nearest Neighbors Algorithm <https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761>
- Qomariyah N. N., Heriyanni E., Fajar A. N. and Kazakov D. (2020). Comparative Analysis of Decision Tree Algorithm for Learning Ordinal Data Expressed as Pairwise Comparisons. In *2020 8th International Conference on Information and Communication Technology (ICoICT)*. pp. 1-4, doi: 10.1109/ICoICT49345.2020.9166341.
- Ries E. (2011). *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. 1st Edition. Currency. ISBN-10 : 9780307887894

- Sabbeh, S.F. (2018). Machine-Learning Techniques for Customer Retention: A Comparative Study. *International Journal of Advanced Computer Science and Applications*, 9(2).
<https://doi.org/10.14569/ijacsa.2018.090238>
- Saji B. (2021). A Quick Introduction to K – Nearest Neighbor (KNN) Classification Using Python
<https://www.analyticsvidhya.com/blog/2021/01/a-quick-introduction-to-k-nearest-neighbor-knn-classification-using-python/>
- Sampaio C. (n.d.) Guide to the K-Nearest Neighbors Algorithm in Python and Scikit-Learn
<https://stackabuse.com/k-nearest-neighbors-algorithm-in-python-and-scikit-learn/>
- Saran K. A. and Chandrakala D. (2016). A Survey on Customer Churn Prediction using Machine Learning Techniques. *International Journal of Computer Applications*, 154(10), 0975 – 8887
- Sharma A. (2021). Customer Churn Data Analysis using Logistic Regression. <https://medium.com/data-science-on-customer-churn-data/customer-churn-data-analysis-using-logistic-regression-3861e2d4d1f3>
- Song J, Gao Y, Yin P, Li Y, Li Y, Zhang J, Su Q, Fu X, Pi H. (2021). The Random Forest Model Has the Best Accuracy Among the Four Pressure Ulcer Prediction Models Using Machine Learning Algorithms. *Risk Manag Healthc Policy*. 14:1175-1187. <https://doi.org/10.2147/RMHP.S297838n>.
<https://www.dovepress.com/the-random-forest-model-has-the-best-accuracy-among-the-four-pressure-peer-reviewed-fulltext-article-RMHP>
- Sruthi E. R. (2021). Understanding Random Forest.
<https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/>
- Sunil R. (2017). Understanding Support Vector Machines (SVM) algorithm from examples (along with code)
<https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/>
- Vadakattu, R.; Panda, B.; Narayan, S.; Godhia, H. (2015). Enterprise subscription churn prediction. IEEE International Conference on Big Data (Big Data).
- Votava A. (2021). Churn prediction model. <https://towardsdatascience.com/churn-prediction-model-8a3f669cc760>
- Wasike B. (2021). Machine Learning with XGBoost and Scikit-learn. <https://www.section.io/engineering-education/machine-learning-with-xgboost-and-scikit-learn/>
- Wiryaseputra M. (2022). Bank Customer Churn Prediction Using Machine Learning
<https://www.analyticsvidhya.com/blog/2022/09/bank-customer-churn-prediction-using-machine-learning/>