Churn Prediction for Telecom Subscription

Shin Le - Florida State University

1. Problem Statement

Churn, the rate at which customers potentially cancel a service, poses a risk to long-term revenue. Predicting churn enables businesses to identify key factors influencing cancellations, allowing them to enhance their subscription-based services. This proactive approach helps in retaining customers and improving overall customer experience, considering the lower cost of service improvement compared to acquiring new customers.

Understanding the dynamics of customer behavior leading to cancellations is crucial. By leveraging churn prediction models, businesses can pinpoint specific features or factors to focus on for improvement, ensuring customer satisfaction and loyalty. This knowledge informs strategic planning, guiding the development of new initiatives and enhancements to existing services, thereby minimizing cancellations.

2. Dataset

For this project, we will utilize a dataset that contains historical customer data associated with a telecommunication. This dataset contains information about customer subscriptions and their interaction with the service. The data includes various features such as subscription type, payment method, viewing preferences, customer support interactions, and other relevant attributes. It will serve as the basis for training and evaluating our churn prediction model.

These are the columns that are contained in the dataset:

	Column_name	Column_type	Data_type	Description
0	AccountAge	Feature	integer	The age of the user's account in months.
1	MonthlyCharges	Feature	float	The amount charged to the user on a monthly basis.
2	TotalCharges	Feature	float	The total charges incurred by the user over the account's lifetime.
3	SubscriptionType	Feature	object	The type of subscription chosen by the user (Basic, Standard, or Premium).
4	PaymentMethod	Feature	string	The method of payment used by the user.
5	PaperlessBilling	Feature	string	Indicates whether the user has opted for paperless billing (Yes or No).
6	ContentType	Feature	string	The type of content preferred by the user (Movies, TV Shows, or Both).
7	MultiDeviceAccess	Feature	string	Indicates whether the user has access to the service on multiple devices (Yes or No).
8	DeviceRegistered	Feature	string	The type of device registered by the user (TV, Mobile, Tablet, or Computer).
9	ViewingHoursPerWeek	Feature	float	The number of hours the user spends watching content per week.
10	AverageViewingDuration	Feature	float	The average duration of each viewing session in minutes.
11	ContentDownloadsPerMonth	Feature	integer	The number of content downloads by the user per month.
12	GenrePreference	Feature	string	The preferred genre of content chosen by the user.
13	UserRating	Feature	float	The user's rating for the service on a scale of 1 to 5.
14	SupportTicketsPerMonth	Feature	integer	The number of support tickets raised by the user per month.
15	Gender	Feature	string	The gender of the user (Male or Female).
16	WatchlistSize	Feature	float	The number of items in the user's watchlist.
17	ParentalControl	Feature	string	Indicates whether parental control is enabled for the user (Yes or No).
18	SubtitlesEnabled	Feature	string	Indicates whether subtitles are enabled for the user (Yes or No).
19	CustomerID	Identifier	string	A unique identifier for each customer.
20	Churn	Target	integer	The target variable indicating whether a user has churned or not (1 for churned, 0 for not churned).

3. Description of the methodology used

- **Resampling Strategy:** To mitigate bias in accuracy calculations, implement oversampling (Ex: SMOTE) to balance the class distribution before splitting the dataset into training and test sets. This ensures a more equitable representation of classes during model training and evaluation.

- **Logistic Regression:** We started with the Sequential Feature Selection method to choose some significant variables for our model. In research, logistic regression is a common method to use for churn prediction since it is simple and easy to interpret.

- **Decision Tree:** The goal of the decision tree is to take our many data features, learn simple decision rules inferred by those features, and predict our target variable, in this case, our car price. A Pruned Tree can be used to reduce the size of tree that is not overfitting. Decision trees are the most susceptible out of all the machine learning algorithms to overfitting and effective pruning can reduce this likelihood.

- **Random Forest:** A potent supervised machine-learning algorithm is an amalgamation of decision trees. Leveraging the bagging technique, it assembles diverse samples, generating multiple trees from bootstrap samples. These individual trees collectively contribute to the robust framework of the Random Forest model, enhancing its predictive capabilities.

- **Gradient Boosting:** A dynamic variant of Ensemble Models harnesses the strength of multiple weak models to enhance overall performance. In the realm of classification, the iterative process involves leveraging metrics such as balanced accuracy and F1 score. With each iteration, the model is fine-tuned based on these metrics, continually refining its predictive capabilities to achieve superior classification accuracy.

- **AdaBoost:** A compelling ensemble learning technique, operates by combining the strengths of multiple weak classifiers to achieve heightened classification performance. In the classification context, AdaBoost excels by iteratively adjusting the model based on misclassified instances. This adaptive learning process empowers AdaBoost to incrementally improve its classification accuracy, making it a robust and effective tool in the realm of machine learning.

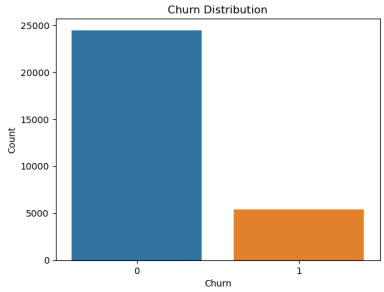
- **Support Vector Machine (SVM):** A formidable, supervised learning algorithm in machine learning, finds applications in both classification and regression tasks. Operating in an n-dimensional space, where n represents the number of features, the SVM endeavors to establish a hyperplane that effectively segregates distinct feature groups. In our dataset, we opted for the linear kernel, a strategic choice given the larger volume of records and features at our disposal.

- **Neural Network**: A multi-layer neural network, a pivotal architecture in deep learning, consists of interconnected layers of neurons. Comprising an input layer, hidden layers, and an output layer, this network excels at capturing intricate patterns and relationships within data. Through forward and backward propagation, the network refines its weights, progressively enhancing its ability to extract complex features and make nuanced predictions.

4. Description of the implementation, potential problems, and assumptions

- **Data collection**:
  Our data was sourced from the "Predictive Analytics for Customer Churn: Dataset" by Safrin S published on Kaggle. The dataset contains sales data from 70,000 records, with 21 features related to information about the customer and their behavior on the service.
- **Data Preprocessing:**

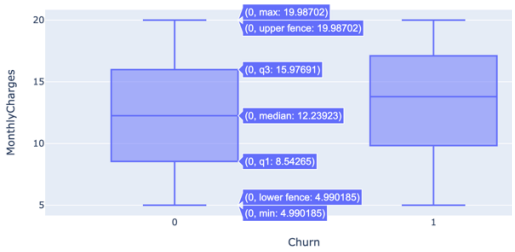
Our dataset stands prepared for analysis, devoid of null or missing values, ensuring a solid foundation for our exploration. Although outliers may be present, their influence on our results is deemed negligible. To facilitate optimal model performance, our preprocessing strategy involves labeling categorical features and standardizing the data.



By examining the response/target variable 'Churn' in the dataset, we can observe that the dataset has **an imbalance issue**, with '**Not Churn'** being the majority class. This means that evaluating accuracy alone might not be as informative, as it could be skewed by the dominance of the majority class. Accuracy would be calculated based on the majority class, while the minority class would have minimal impact on the accuracy measurement.

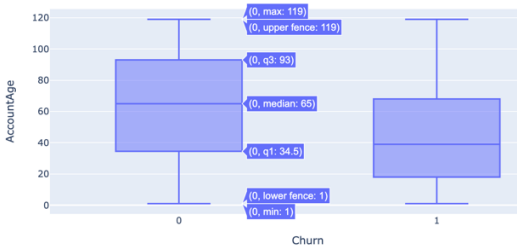
- **Exploratory data analysis (EDA):** Using visualization to obtain the first understand about the dataset and gain the trend of data.

Monthly Charges vs. Churn



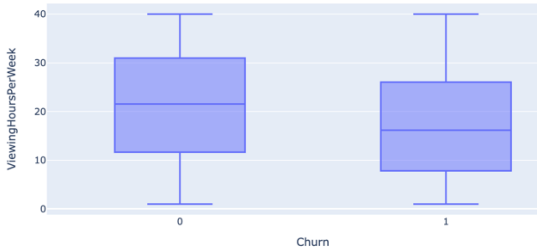
The higher the plan's premium, the higher the churn rate tends to be. That means, the possibility of cancellation is higher if customers spend more money on the service.

Account Age vs. Churn



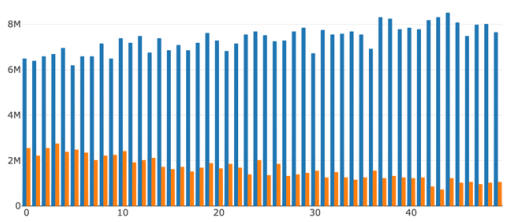
Customers become loyal connections to the business service after staying for at least 34.5 months. However, churn begins to increase after they have been using the service for at least 20 months.

Viewing Hours Per Week vs. Churn

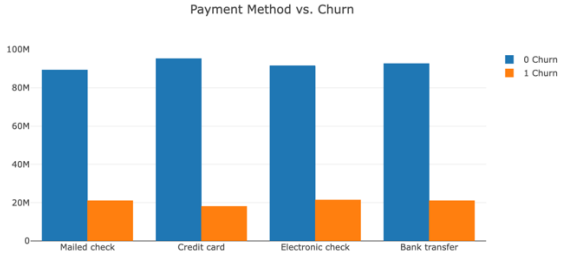


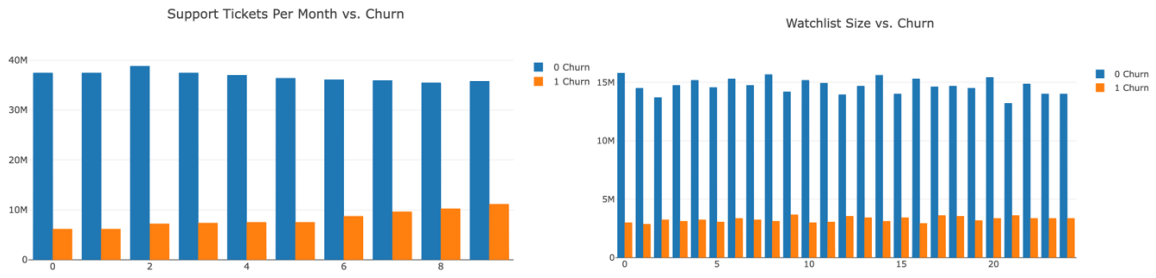
The churn rate tends to be higher when customers spend more time watching content per week. However, on average, this rate does not differ significantly from the non-churn rate.

Content Downloads Per Month vs. Churn



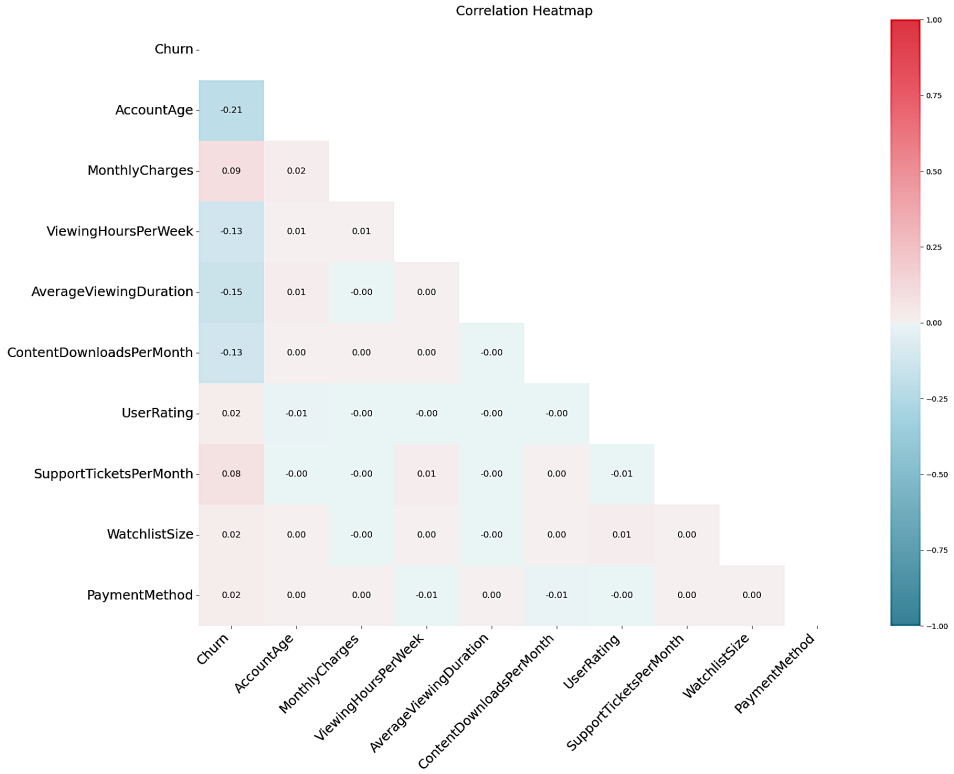
Payment Method vs. Churn





It seems that **content downloads per month, payment methods, support tickets per month, watchlist size** show comparable patterns between churn and non-churn rates in their respective categories. This suggests that cancellations could occur equally across various factors, indicating a lack of significant correlation with any specific factor.

- A **correlation matrix** can be used to detect and minimize the multilinearity (dependence) among features and select features to fit the models.



- These are the features, such as **AccountAge, MonthlyCharges, ViewingHoursPerWeek, AverageViewingDuration, ContentDownloadsPerMonth, UserRating, SupportTicketsPerMonth, WatchlistSize, PaymentMethod** that we employ to fit the models.

- **Data Splitting**:
  We also need to apply **resampling** (**oversampling/undersampling**) method to balance our dataset and improve the accuracy calculation.
  Before creating our models, we create proper training and testing sets for our models. For this purpose, we used scikit-learn's train_test_split to divide our dataset into 70% training data and 30% test data.

5. Model Description:

a. *Logistic Regression*: We employed the top 7 significant features, selected through Forward-Sequential Feature Selection, to train logistic regression and other models. This approach, aimed at mitigating overfitting and multicollinearity, involves reducing the dataset's size. Our model training utilized LogisticRegressionCV with lasso regularization, employing a 5-fold cross-validation. The solver "liblinear" was employed for training and testing the dataset.

b. *Decision Tree*: Employing the entropy criterion and restricting the maximum depth to 4, a decision tree classifier undergoes refinement through cost complexity pruning. Through cross-validation utilizing 5 folds and a grid search spanning the cost-complexity alpha, we meticulously explore the parameter space. This process is geared towards identifying the optimal pruning parameter, seeking a model that strikes the optimal balance between complexity and performance, evaluated through accuracy scores.

c. *Random Forest*: Conducting a thorough exploration, we implement a 5-fold cross-validation coupled with grid search, considering various parameters such as the number of estimators, max depth, cost complexity pruning alpha, and more. This comprehensive approach aims to identify the best estimator, optimizing for enhanced prediction accuracy while mitigating the risk of overfitting.

d. *Gradient Boosting*: Following a methodology akin to Random Forest, this approach provides a distinct advantage through meticulous data assessment. The ability to set a learning rate empowers us to control the learning process, ultimately enhancing the model's performance. Notably, this method excels as an optimal strategy for mitigating overfitting, ensuring a refined and well-calibrated model.

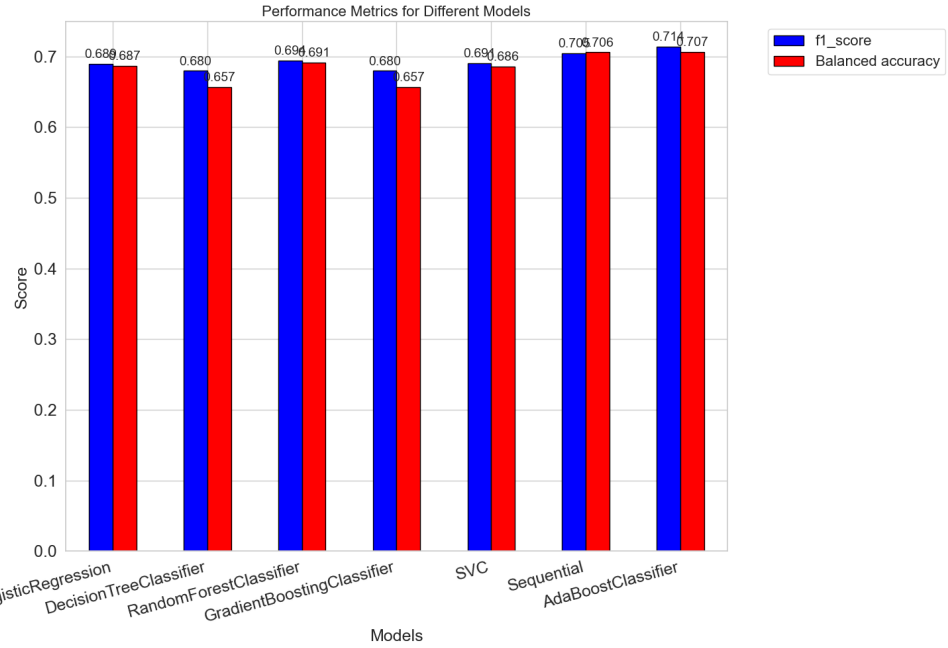
e. *AdaBoost*: using a decision tree as the base estimator (best_tree). Configured with 15 weak learners, the model strikes a balance between complexity and computational efficiency. This AdaBoost model, combining multiple weak learners, aims to enhance performance and adaptability in predictive tasks.

f. *SVM*: We standardized the data before fitting the SVM model. As with all of our other models, we conducted 5-fold cross-validation and utilized grid search to identify the estimator that provides the optimal hyperplane for each specific kernel with multiple parameters. This approach is appropriate to deal with high-dimensional feature space.

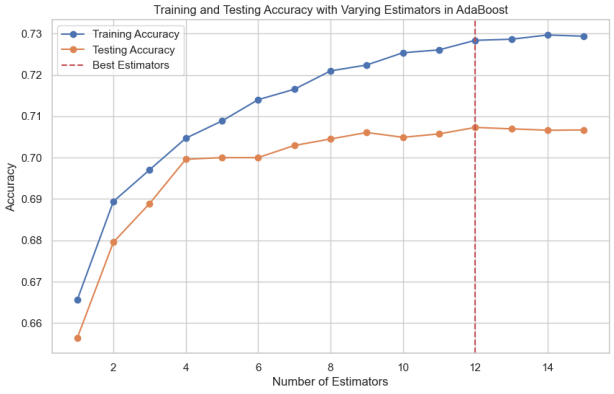
g. *Neural Network:* The provided code constructs a neural network model using TensorFlow's Keras API. The model architecture includes three layers: an input layer with 64 neurons and (Rectified Linear Unit) ReLU activation, a dropout layer with a dropout rate of 0.5, a hidden layer with 32 neurons and ReLU activation, another dropout layer, and a final output layer with one neuron and sigmoid activation for binary classification. The model is compiled using the *Adam optimizer* and *binary crossentropy loss*, with accuracy as the evaluation metric. It is then trained for 10 epochs on the training data, with a batch size of 64 and validation on the test data. Finally, the model's performance is evaluated on the test set, and the test accuracy is printed. This neural network aims to learn patterns within the data for binary classification tasks.

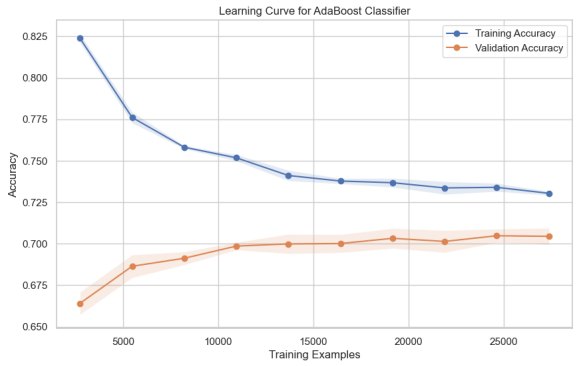
6. Presentation and analysis of results

- Evaluate and compare the models using **F1-score**, and **balance accuracy** for an imbalanced dataset.



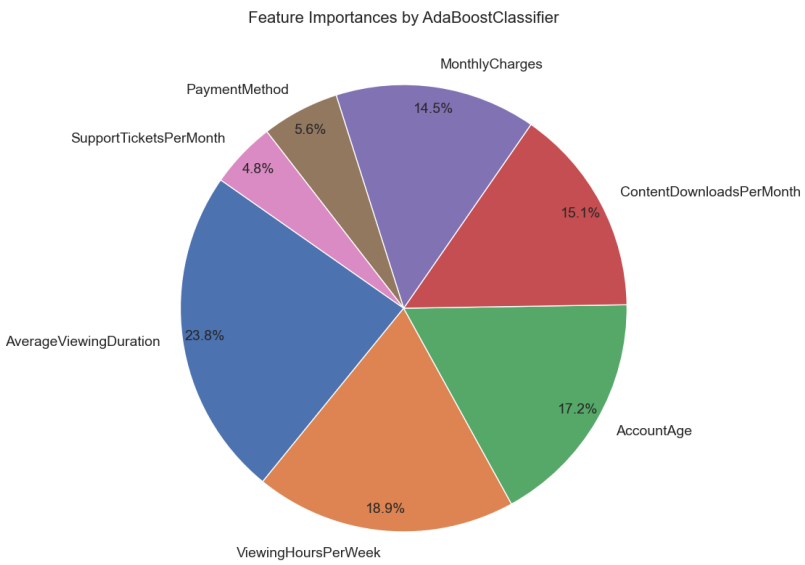
This accuracy plot shows that AdaBoost is the most optimal model for managing our imbalanced dataset in the context of churn prediction, though the balanced accuracies and F1- scores among models are not different much at all.



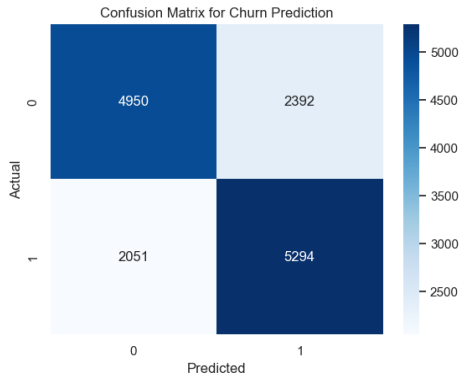


These plots illustrate the convergence of accuracy scores with an increasing number of training examples (training data size). As the number of estimators grows, both testing and training accuracies stabilize, indicating that the model's performance becomes more consistent and reliable. This convergence suggests that the model benefits from additional training examples, leading to improved generalization and a more robust predictive capability.

- Let’s see the feature importance from the best model (AdaBoost) coefficients.



The pie chart highlights the influential features in churn prediction, with 'AverageViewingDuration' emerging as the most impactful, contributing 23.8% to the AdaBoost model's generalization on unseen data. Additionally, 'ViewingHoursPerWeek,' 'AccountAge,' 'ContentDownloadsPerMonth,' and 'MonthlyCharges' are identified as key factors in determining churn rate. Together, these features collectively contribute to an impressive 90% success rate in predicting churn, underlining their crucial roles in the model's overall predictive accuracy.



The confusion matrix provides insights into the model's predictions, showcasing a balanced accuracy across different classes after addressing the initial imbalance issue. The overall accuracy stands at a solid 70%. However, it's noteworthy that the mismatch rate between actual and predicted classes remains relatively high. This suggests the need for further exploration and potentially fine-tuning the model to enhance its performance in capturing class-specific patterns and improving overall prediction accuracy.

7. Insights and discussions about the results

- The Feature Importance plot generated by AdaBoost indicates that Average Viewing Duration, Viewing Hours Per Week, Account Age, Content Downloads Per Month, and Monthly Charges exert the most significant influence on churn prediction. It is well-known that the cost of acquiring new customers is higher than retaining existing ones. Therefore, focusing on improving these features could potentially lead to a reduction in the churn rate.

	Model	precision_score	recall_score	f1_score	Balanced accuracy
0	LogisticRegression	0.680224	0.698671	0.689324	0.686561
1	DecisionTreeClassifier	0.634025	0.732502	0.679715	0.656707
2	RandomForestClassifier	0.684414	0.704287	0.694208	0.691201
3	GradientBoostingClassifier	0.634025	0.732502	0.679715	0.656707
4	SVC	0.677564	0.704013	0.690535	0.685973
5	Sequential	0.704698	0.704698	0.704698	0.706001
6	AdaBoostClassifier	0.694261	0.734009	0.713582	0.706809

- The tables presented showcase results obtained from fitting the balanced dataset, addressing imbalanced data through the oversampling method, specifically using SMOTE in Python. This approach assumes near equality between churn and non-churn rates, interpreting an increasing number of cancellations as an indicator of unsatisfactory service.

- Analyzing the table and accuracy plot are evident that all models perform well, achieving balanced accuracies within the range of 68% to 72%. Notably, there is minimal variance in balanced accuracies across models. Hence, the preference leans towards simpler and more interpretable methods. Logistic Regression emerges as a robust choice for churn prediction, especially evident when fitting the model with selected features, reinforcing the notion that a few features hold negligible impact. This underlines the potential for optimal logistic regression as an effective tool for interpreting and predicting churn.

- The Neural Network (Sequential Model in Python) is an advanced machine learning model that has the potential to improve the predictive accuracy of a churn model. However, we observed similar results when using the AdaBoost method.

8. Future research directions:

We integrate our models to enhance predictive accuracy, aiming to discern more noticeable differences between various methods. The Neural Network stands out as a more considerate choice due to its advanced nature and ease of adjustment, such as the addition of hidden layers, modification of activation functions, or alteration of batch sizes, offering improved outcomes.

Furthermore, we emphasize the development of models that not only deliver accurate predictions but also provide interpretable explanations for anticipating customer churn. Employing Explainable AI techniques facilitates a better understanding of the factors influencing churn, enabling businesses to make more informed decisions.

Our focus lies in crafting models capable of adapting to evolving customer behavior over time. Real-time churn prediction systems empower businesses to proactively address potential churn events as they unfold, facilitating timely intervention strategies.