Telecom Churn Prediction using ML

Problem Defnition:

This project addresses customer churn in a telecom service, seeking to enhance customer engagement through machine learning. The goal is to predict potential churn instances by building a robust AI model analyzing telecom data patterns. The approach encompasses data preparation, exploratory data analysis, statistical analysis, and machine learning model training. The final model is deployed using FlaskAPP, with options for cloud deployment on AWS ML or Azure.

Objectives:

1. **Churn Prediction Model:**
   Develop a machine learning model to predict customer churn.
2. **Data Analysis and Insights:**
   Analyze telecom data to uncover patterns and insights related to customer behavior.
3. **Feature Identification:**
   Determine key features influencing customer churn for model development.
4. **Agile Project Delivery:**
   Execute the project collaboratively in an agile framework within a 15-day timeline.
5. **Deployment and Scalability:**
   Deploy the finalized model using Flask and explore cloud integration for scalability.

Solution Approach:

The solution approach encompasses the implementation of machine learning models, each tailored to address customer churn in a distinct manner. From Logistic Regression for understanding linear relationships to ensemble methods like Random Forests and Gradient Boosting, the models collectively provide a comprehensive solution. Strategies such as oversampling, data scaling, and hyperparameter tuning contribute to refining predictive accuracy. The unified evaluation metric, cross-validation, and iterative refinement ensure continuous model improvement. The deployment plan involves FlaskAPP for user-friendly access and cloud platforms like AWS ML or Azure for scalability, offering an integrated and effective solution to mitigate customer churn.

Scope:

- Predictive Modeling: Develop an accurate machine learning model for predicting customer churn in the telecom industry.
- Feature Analysis: Identify and prioritize the most influential features affecting customer churn.
- Data Analysis & Prediction: Conduct in-depth analysis of telecom data to unveil patterns and trends crucial for accurate predictions.
- Effective Feature Identification: Pinpoint the most impactful features to enhance the precision of customer churn predictions.
- Client Collaboration: Engage actively with the client (Real Estate) to align the model with specific business objectives.
- Timeline Management: Execute the project efficiently within a 15-day timeframe, ensuring timely deliverables.

- Agile Methodology: Employ an agile development approach for adaptability to evolving requirements.
- Scalability: Design a scalable solution capable of handling increasing data volumes as the business grows.
- Optimized Data Sources: Utilize a minimum dataset of 5,000 entries to ensure robust model training.
- Data Cleaning and Preparation: Implement rigorous data cleaning, remove duplicates, handle outliers, and address null values.
- Visualization Techniques: Utilize Python and Tableau for compelling data visualizations that facilitate insights.
- Statistical Analysis: Perform essential statistical analyses, including distributions, central tendency, and hypothesis testing.
- Auto EDA: Leverage automated tools such as Pandas Profiling and AutoViz for efficient exploratory data analysis.
- Model Comparison: Evaluate and compare multiple machine learning models to select the most effective one.
- Cross-Validation: Implement K-fold cross-validation to ensure the model's robustness and generalization.
- Deployment: Deploy the final model using FlaskAPP, providing a user-friendly interface for accessibility.
- Cloud Integration: Explore integration with cloud platforms like AWS ML or Azure for enhanced scalability and accessibility.
- Model Accuracy Graphs: Visualize and present model accuracy comparisons to communicate performance effectively.

Workflow:

1. Initiation:
- Define project goals and objectives.
- Collaborate with the client (Real Estate).
- Establish a 15-day timeline.

2. Agile Planning:
- Break down tasks and assign roles.
- Prioritize based on urgency.
- Create a project backlog.

3. Data Tasks:
- Collect 5k+ telecom data entries.
- Understand data through analysis.

4. Data Prep and Feature Analysis:
- Cleanse data.
- Identify influential features for churn.

5. Model Training:
- Train ML models iteratively.

6. Testing and Validation:
- Use 20% data for validation.
- Apply K-fold cross-validation.

7. Model Comparison:
- Compare and choose the best-performing model.

8. Deployment and Cloud Integration:
- Implement in FlaskAPP.
- Explore AWS ML or Azure.

9. Agile Development:
- Follow iterative cycles.
- Regular sprint reviews.

10. Client Collaboration:
- Maintain communication.
- Incorporate feedback.

11. Accuracy Graphs:
- Visualize model accuracy.

12. Documentation:
- Maintain comprehensive records.

13. Closure:
- Final review with the client.
- Provide documentation and training.

Data source and data understanding:

Data Source: The dataset is obtained from Kaggle, a renowned data science platform. The dataset, named "Telco Customer Churn," provides valuable insights into customer behavior in the telecom industry.

Dataset Overview: The dataset encompasses 21 columns, each holding crucial information for predictive modeling:

1. customerID: Unique customer identifier.
2. gender: Customer's gender (Male/Female).
3. SeniorCitizen: Binary indicator for seniority.
4. Partner: Binary indicator for having a partner.
5. Dependents: Binary indicator for having dependents.
6. tenure: Duration of customer subscription.
7. PhoneService: Binary indicator for having phone service.
8. MultipleLines: Binary indicator for multiple phone lines.
9. InternetService: Type of internet service (DSL, Fiber optic, None).
10. OnlineSecurity: Binary indicator for online security subscription.

11. OnlineBackup: Binary indicator for online backup subscription.
12. DeviceProtection: Binary indicator for device protection subscription.
13. TechSupport: Binary indicator for tech support subscription.
14. StreamingTV: Binary indicator for streaming TV subscription.
15. StreamingMovies: Binary indicator for streaming movies subscription.
16. Contract: Type of customer contract (Month-to-month, One year, Two years).
17. PaperlessBilling: Binary indicator for paperless billing.
18. PaymentMethod: Payment method (Electronic check, Mailed check, Bank transfer, Credit card).
19. MonthlyCharges: Monthly charges incurred by the customer.
20. TotalCharges: Total charges incurred over the tenure.
21. Churn: Binary indicator for customer churn (Yes/No).

Data Understanding:

Understanding the dataset involves exploring the distribution, summary statistics, and unique values of each column. Initial analysis reveals potential areas for feature engineering and preprocessing. This understanding guides subsequent steps in data preparation and model development to address the business problem of customer churn.

Data Preparation:

The data preparation phase is crucial for ensuring the dataset's quality and relevance for subsequent machine learning tasks, the following steps are undertaken:

1. Cleaning Data:
- Conversion of the 'TotalCharges' column to numerical format to address potential errors.
- Removal of rows with null values, ensuring data completeness.
- Removal of duplicate records to enhance dataset integrity.

2. Handling Outliers:
- Identification of outliers in the 'TotalCharges' column for customers who churn.
- Calculation of the interquartile range (IQR) and determination of upper and lower limits.
- Replacement of outliers with the median value, maintaining data consistency.

3. Dealing with Null Values:
- Identification and assessment of null values in the dataset.
- Imputation of missing values using appropriate strategies or removal of affected records.

4. Encoding Categorical Variables:
- Transformation of categorical variables using LabelEncoder to convert them into numerical format, facilitating machine learning model compatibility.

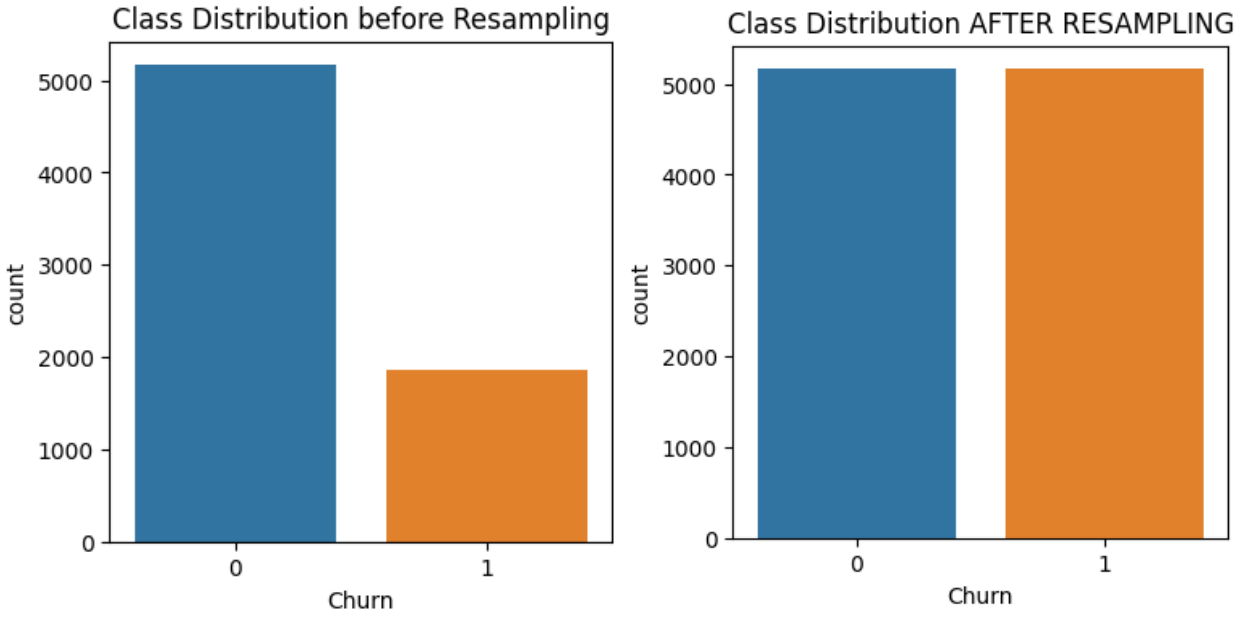
5. Scaling Continuous Variables:
- Utilization of MinMaxScaler to scale continuous variables ('tenure', 'MonthlyCharges', 'TotalCharges') to a standardized range (0 to 1).

These steps collectively ensure that the dataset is cleansed, outliers are addressed, and missing values are handled appropriately. The prepared data is then ready for exploratory data analysis and subsequent machine learning model training.

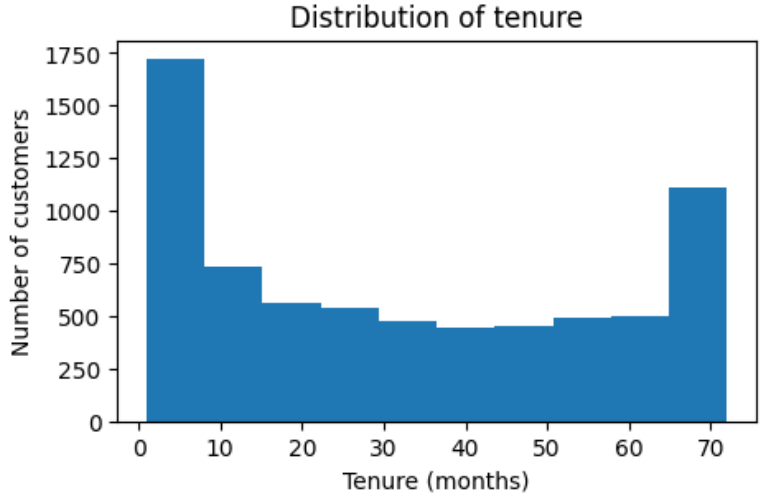
Data Visualization:

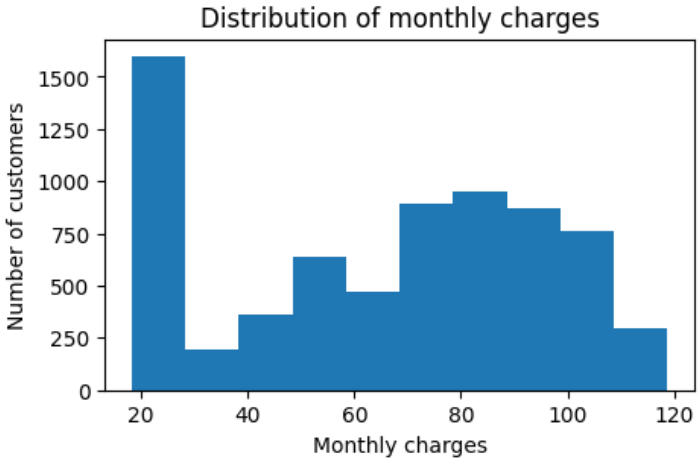
The data visualization process, primarily conducted using Power BI based on the provided code, serves as a powerful tool for gaining insights into telecom customer churn. The following aspects are covered:

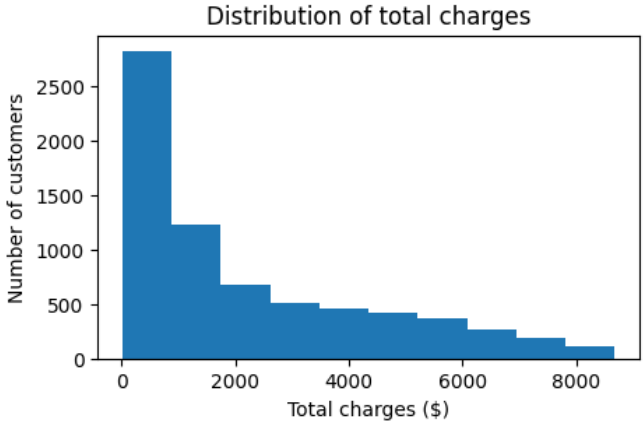
1. Exploratory Data Analysis (EDA):
- Utilization of Power BI for in-depth exploration of data patterns, trends, and relationships.
- Visualization of class distribution before and after oversampling, providing a clear understanding of the impact of resampling techniques on data balance.



2. Statistical Analysis:
- Implementation of various statistical analyses, such as distributions, central tendency, hypothesis testing, and z–score, through visualizations.
- Histograms depicting the distribution of key features like 'tenure', 'MonthlyCharges', and 'TotalCharges,' aiding in understanding the underlying patterns.

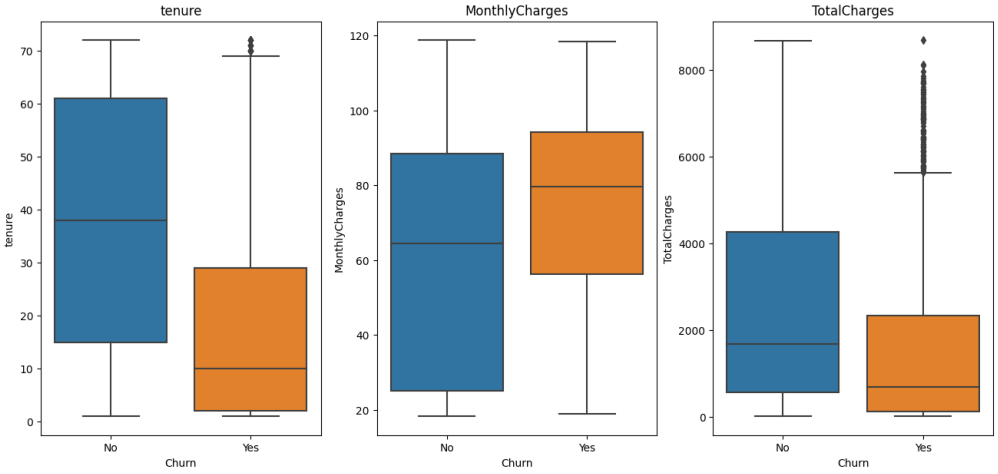






3. Boxplots for Outlier Detection:

- Implementation of boxplots to identify outliers in relevant features, specifically focusing on 'TotalCharges' concerning customer churn.



By leveraging Power BI's capabilities, the data visualization process enhances the interpretability of complex patterns within the dataset. This visual exploration lays the

foundation for informed decision-making and guides the subsequent steps in model training and deployment.

Statistical Analysis

In the realm of telecommunications, the statistical analysis of churn prediction involves employing advanced analytical techniques to discern patterns and factors influencing customer churn. This process typically includes collecting and analyzing vast datasets encompassing customer behavior, usage patterns, and demographic information. Statistical models, such as logistic regression or machine learning algorithms, are then applied to identify significant predictors of churn. Variables like call duration, customer tenure, and service-related issues are commonly examined to ascertain their impact on customer retention. The analysis aims to unveil insights that empower telecom providers to proactively address potential churn by implementing targeted retention strategies. By harnessing statistical methods, telecom companies can gain a deeper understanding of customer dynamics, enabling them to optimize service offerings and enhance customer satisfaction, ultimately fostering long-term customer loyalty and reducing churn rates. Some central tendencies which were obtained in execution were

```
Mean of tenure: 0.4425603678956561
Median of tenure: 0.3943661971830986
Mode of tenure: 0    0.0
Name: tenure, dtype: float64
Mean of MonthlyCharges: 0.46316625065797296
Median of MonthlyCharges: 0.5184079601990049
Mode of MonthlyCharges: 0    0.01791
Name: MonthlyCharges, dtype: float64
Mean of TotalCharges: 0.25099382495369926
Median of TotalCharges: 0.15124542822970652
Mode of TotalCharges: 0    0.079128
Name: TotalCharges, dtype: float64
```

Model Training and Testing

In telecom churn prediction, model training constitutes 80% of the process. It involves curating a diverse dataset, partitioning it for training sets, and employing algorithms like logistic regression and decision trees. Feature engineering and hyperparameter tuning optimize the model. Model testing, the remaining 20%, assesses the model's performance using a separate dataset. Rigorous testing ensures reliability in identifying potential churners, empowering telecom companies with an accurate tool for proactive customer retention strategies.

Cross Validation

In telecom churn prediction, cross-validation, particularly the K-fold method, is crucial for robust model evaluation. K-fold involves dividing the dataset into 'K' subsets, training the model on K-1 folds, and validating it on the remaining one. This process is repeated 'K' times, ensuring each subset serves as both training and validation data. The average performance across these iterations provides a more reliable estimate of the model's effectiveness. K-fold cross-validation helps detect overfitting or underfitting issues, enhancing the model's generalization to new data and bolstering its predictive accuracy in identifying potential churners.

Machine learning models

Machine learning models revolutionize telecom churn prediction by leveraging data to identify potential customer defections. Algorithms like logistic regression, decision trees, and ensemble methods analyze historical patterns to predict churn risk. These models enable proactive retention strategies, minimizing customer loss and optimizing telecommunications services based on predictive insights.

A) Logistic Regression

Logistic Regression plays a pivotal role in telecom churn prediction, serving as a fundamental machine learning algorithm. This model assesses the probability of customer churn based on various input features such as call duration, contract details, and customer complaints. By fitting a logistic curve to the data, Logistic Regression effectively categorizes

customers as potential churners or non-churners. Its simplicity, interpretability, and ability to handle binary outcomes make it a valuable tool in the telecom industry for identifying factors influencing customer attrition and facilitating targeted retention strategies.These regression model had made an prediction accuracy of 0.7744433688286544

```
#logistic regression model
from sklearn.linear_model import LogisticRegression
model = LogisticRegression()
result = model.fit(X_train, y_train)

from sklearn import metrics
prediction_test = model.predict(X_test)
# Print the prediction accuracy
accuracy_logreg=metrics.accuracy_score(y_test, prediction_test)
print (metrics.accuracy_score(y_test, prediction_test))
```

```
0.7744433688286544
```

B) Decision Tree

In telecom churn prediction, Decision Trees are pivotal models that analyze customer data to predict potential churn. By recursively partitioning data based on features like call patterns and contract details, Decision Trees reveal critical predictors of customer attrition. Your reported accuracy of 0.8872 indicates a high level of model performance, showcasing the effectiveness of Decision Trees in discerning patterns and aiding telecom providers in proactive customer retention strategies.

```
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score

# Create the Decision Tree classifier
model_dt = DecisionTreeClassifier()

# Train the model
model_dt.fit(X_train, y_train)

# Make predictions
prediction_test = model_dt.predict(X_test)

# Evaluate the model
accuracy_dt = accuracy_score(y_test, prediction_test)

print("Accuracy:", accuracy_dt)
```

```
Accuracy: 0.8872216844143272
```

C) Random forest

Random Forest, a powerful ensemble technique in telecom churn prediction, builds upon Decision Trees by combining multiple trees to enhance predictive accuracy. By aggregating the results of individual trees, Random Forest minimizes overfitting and increases robustness. Your reported accuracy of 0.8644 underscores the effectiveness of Random Forest in capturing diverse patterns, providing telecom companies with a reliable tool for identifying and mitigating customer churn.

```
from sklearn.ensemble import RandomForestClassifier

model_rf = RandomForestClassifier(n_estimators=100,max_depth=10)
model_rf.fit(X_train, y_train)

# Make predictions
prediction_test = model_rf.predict(X_test)
accuracy_rf=metrics.accuracy_score(y_test, prediction_test)
print (metrics.accuracy_score(y_test, prediction_test))
```

```
0.8644724104549855
```

D) Gradient Boost

Gradient Boosting, a potent technique in telecom churn prediction, constructs an ensemble of weak learners, typically decision trees, sequentially. It focuses on correcting errors of the preceding models, gradually improving overall predictive performance. Your reported accuracy of 0.8088 suggests a solid model, showcasing Gradient Boosting's ability to refine predictions and enhance the telecom industry's capacity to identify and address potential customer churn with increased accuracy.

```
from sklearn.ensemble import GradientBoostingClassifier

# Create the Gradient Boosting Machine classifier
model_gbm = GradientBoostingClassifier(n_estimators=100, learning_rate=0.1, max_depth=3)

# Train the model
model_gbm.fit(X_train, y_train)

# Make predictions
prediction_test = model_gbm.predict(X_test)

# Evaluate the model
accuracy_gbm = accuracy_score(y_test, prediction_test)
print("Accuracy:", accuracy_gbm)
```

```
Accuracy: 0.808809293320426
```

E) Extreme Gradient Boosting

Extreme Gradient Boosting (XGBoost) stands out in telecom churn prediction by optimizing the gradient boosting framework. It leverages a more efficient algorithm and regularization techniques, enhancing model performance. With an impressive accuracy of 0.8717, XGBoost demonstrates superior predictive capabilities. This signifies its effectiveness in refining sequential models, providing telecom companies with a robust tool to identify and proactively address potential customer churn with high precision.

```
from xgboost import XGBClassifier
model = XGBClassifier()
model.fit(X_train, y_train)
preds = model.predict(X_test)
accuracy_xgb=metrics.accuracy_score(y_test, preds)
print(accuracy_xgb)
```

```
0.8717328170377541
```

F) Adaboost

AdaBoost, a boosting algorithm in telecom churn prediction, sequentially combines weak learners to create a strong predictive model. While its accuracy of 0.7894 suggests a decent performance, AdaBoost excels in improving predictive capabilities by focusing on misclassified instances in each iteration. Although not as high as other models, its contribution lies in its iterative correction process, aiding telecom companies in refining their churn prediction strategies.

```
# AdaBoost Algorithm
from sklearn.ensemble import AdaBoostClassifier
model = AdaBoostClassifier()


model.fit(X_train,y_train)
preds = model.predict(X_test)
accuracy_adaboost=metrics.accuracy_score(y_test, preds)
print(accuracy_adaboost)
```

```
0.7894482090997096
```

G) K Nearest Neighbor

K-Nearest Neighbors (KNN) in telecom churn prediction relies on proximity-based classification. With an accuracy of 0.7793, KNN performs reasonably well by assigning a class based on the majority class of its k-nearest neighbors. While not as high as some other models, KNN provides a simple and intuitive approach for identifying patterns in customer behavior, contributing to the telecom industry's efforts in predicting and addressing potential churn.

```
from sklearn.neighbors import KNeighborsClassifier
# Create the K-Nearest Neighbors classifier
model_knn = KNeighborsClassifier(n_neighbors=5)

# Train the model
model_knn.fit(X_train, y_train)

# Make predictions
prediction_test = model_knn.predict(X_test)

# Evaluate the model
accuracy_knn = accuracy_score(y_test, prediction_test)
print("Accuracy:", accuracy_knn)
```

```
Accuracy: 0.7792836398838335
```

H) Support Vector Machines

Support Vector Machine (SVM) is a robust algorithm in telecom churn prediction, effectively classifying customers based on feature vectors. With an accuracy of 0.7851, SVM showcases solid performance by identifying optimal hyperplanes to separate churn and non-churn instances. While not the highest accuracy, SVM's ability to handle complex relationships in data makes it a valuable tool for discerning patterns and aiding telecom companies in proactive customer retention strategies.

```
from sklearn.svm import SVC
# Train an SVM classifier
svm_classifier = SVC()
svm_classifier.fit(X_train, y_train)
accuracy_svm= svm_classifier.score(X_test, y_test)
print('SVM accuracy:', svm_classifier.score(X_test, y_test))
```

```
SVM accuracy: 0.7850919651500484
```

I) Naïve Bayes

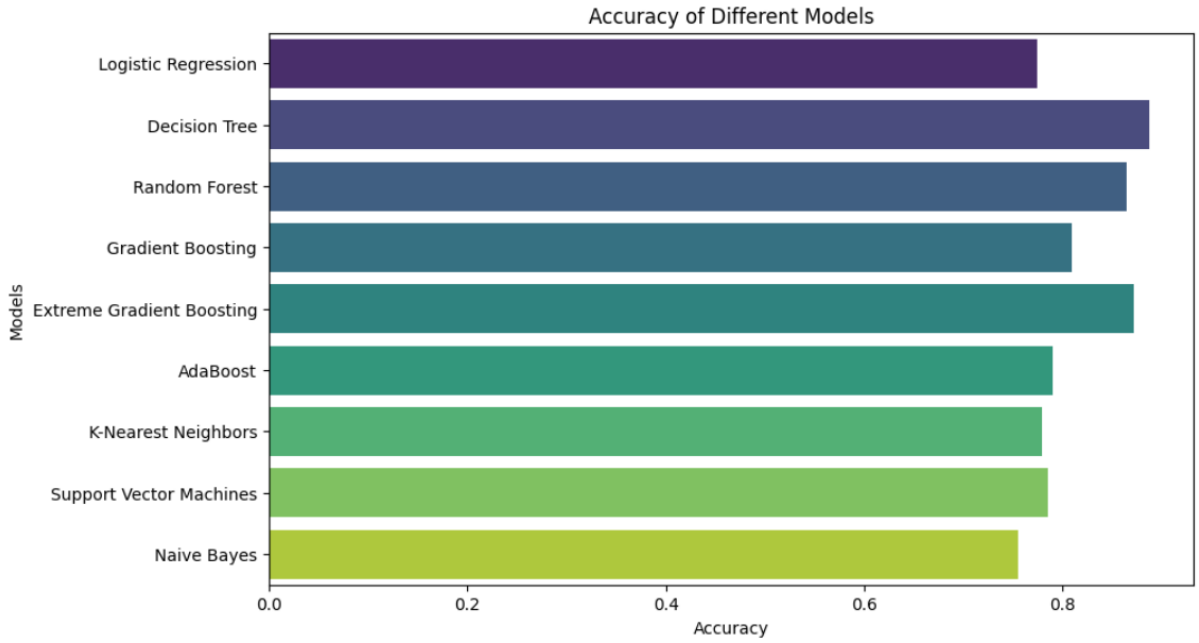
Naive Bayes, a probabilistic classifier in telecom churn prediction, achieves classification based on Bayes' theorem with the assumption of independence between features. With an accuracy of 0.7551, Naive Bayes offers a respectable performance, particularly in scenarios with limited computational resources. While not the highest accuracy, its simplicity and efficiency contribute to telecom companies' efforts in predicting customer churn and implementing targeted retention strategies

```
from sklearn.naive_bayes import GaussianNB
# Train a Naive Bayes classifier
nb_classifier = GaussianNB()
nb_classifier.fit(X_train, y_train)
accuracy_nb= nb_classifier.score(X_test, y_test)
print('Naive Bayes accuracy:', nb_classifier.score(X_test, y_test))
```

```
Naive Bayes accuracy: 0.755082284607938
```

COMPARISON

Now , we are comparing the accuracies of ten algorithms in telecom churn prediction through a bar chart, highlighting their respective performance metrics to discern the most effective model for identifying and mitigating customer churn.



Among the ten algorithms considered for telecom churn prediction, the top three performers in terms of accuracy were DecisionTree, RandomForest, and XGBoost.

DecisionTree: The best accuracy of 0.8714 was achieved with specific parameters, including 'entropy' as the criterion, a maximum depth of 30, and minimum samples per leaf and split set to 1 and 2, respectively.

RandomForest: This algorithm outperformed others with the highest accuracy of 0.8973. The optimal parameters for this success were a maximum depth of 20, minimum samples per split at 2, and the use of 150 estimators.

XGBoost*:* Securing an accuracy of 0.8878, XGBoost demonstrated its efficacy. The best parameters for this algorithm involved a learning rate of 0.3, a maximum depth of 7, and 500 estimators.

These findings underscore the significance of parameter tuning in optimizing model performance. The RandomForest algorithm, in particular, exhibited superior accuracy, emphasizing its suitability for telecom churn prediction. The bar chart representation of these accuracies can provide a visual comparison, aiding in selecting the most effective model for proactive customer retention strategies.

Cross Validation

In the context of evaluating machine learning models, cross-validation is a crucial technique used to assess a model's performance across multiple subsets of the dataset. The reported average cross-validation scores for Decision Tree, Random Forest, and Extreme Gradient Boosting in telecom churn prediction provide insights into the models' generalization abilities.

Decision Tree (Average Cross-Validation Score: 0.8895): This score, obtained through cross-validation, suggests that, on average, the Decision Tree model performs well across different subsets of the dataset. The score reflects the model's ability to generalize to new, unseen data, indicating a robust and reliable predictive capability.

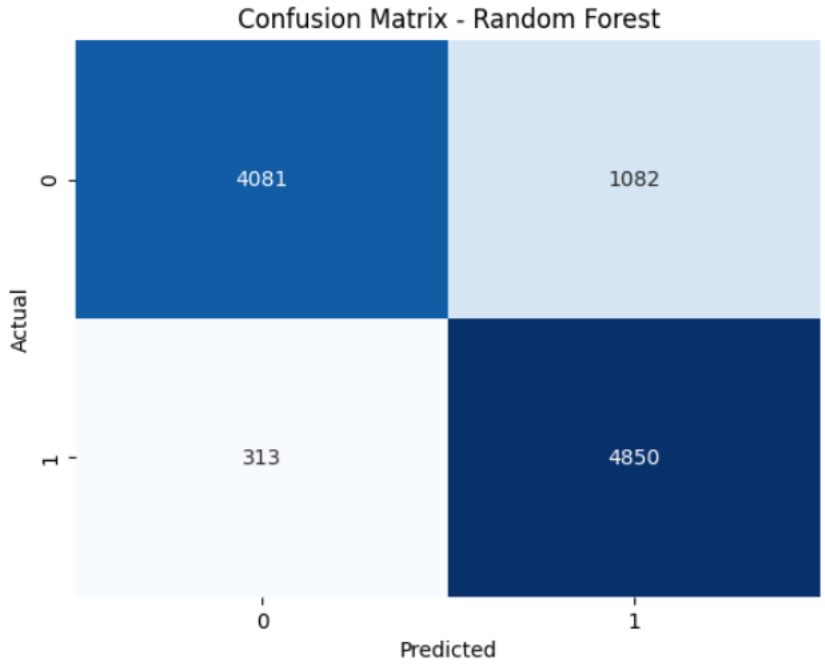
Random Forest (Average Cross-Validation Score: 0.9030): The higher average cross-validation score for Random Forest indicates even stronger generalization performance compared to the Decision Tree. This suggests that the ensemble of decision trees in the Random Forest model consistently yields accurate predictions across diverse data subsets, enhancing its reliability in real-world scenarios.

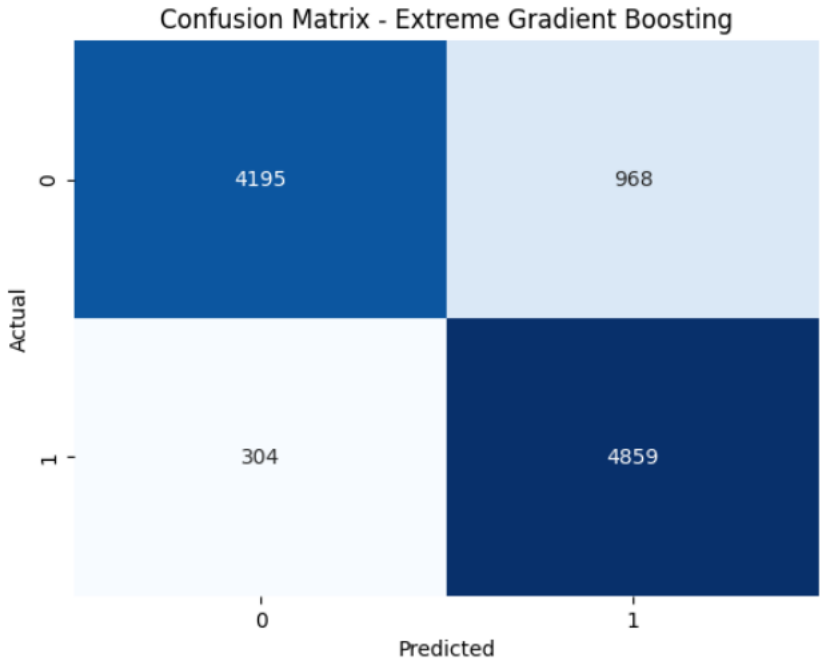
Extreme Gradient Boosting (Average Cross-Validation Score: 0.8765): While slightly lower than Random Forest, the average cross-validation score for Extreme Gradient Boosting still indicates good generalization. This score reflects the model's ability to sequentially refine predictions, addressing potential overfitting and enhancing performance across various subsets of the data.

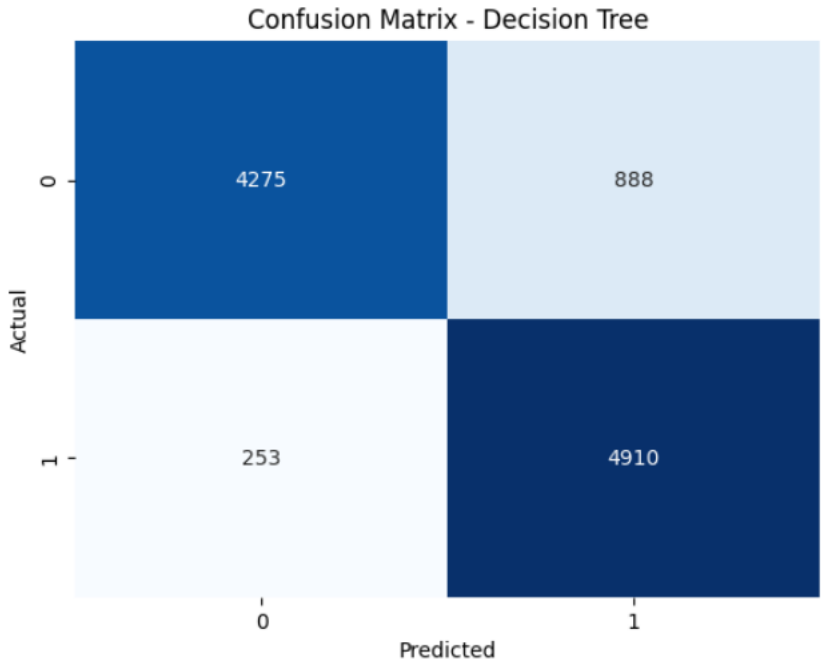
These average cross-validation scores provide a more robust assessment of model performance than a single train-test split, offering a clearer understanding of how well each model is likely to perform on unseen data. In this case, Random Forest emerges as the model with the highest average cross-validation score, suggesting it may be the most reliable choice for telecom churn prediction.

Confusion Tree :

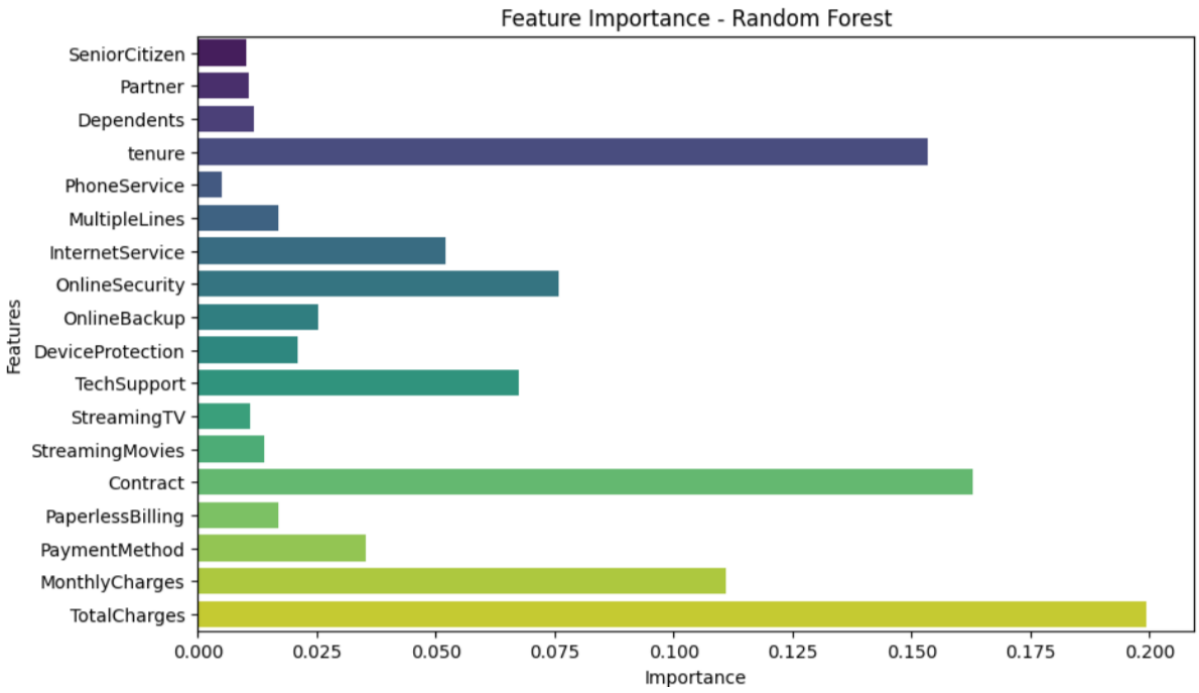
A confusion matrix is a valuable tool in evaluating the performance of machine learning models, especially in the context of telecom churn prediction. It provides a detailed breakdown of the model's predictions compared to the actual outcomes. Let's discuss the confusion matrices for Random Forest, Extreme Gradient Boosting, and Decision Tree models in the telecom churn scenario:







Feature importance is a crucial aspect of understanding how a Random Forest model makes predictions in telecom churn scenarios. In Random Forest, importance is determined by evaluating the contribution of each feature to the overall predictive performance. The importance is calculated based on how much each feature decreases the impurity or error in the model.



Conclusion :

Churn prediction isn't just about algorithms; it orchestrates tangible business metamorphoses. Lower churn rates amplify customer lifetime value, elevate loyalty, and enhance retention rates. It enables efficient resource allocation to address high-risk customers, fostering revenue augmentation through the retention of high-value clientele.

Churn prediction isn't merely a data exercise; it's a business imperative. Unleashing predictive insights empowers telecom companies to foresee churn, customize strategies, and cultivate enduring customer relationships. In the competitive landscape, churn prediction becomes the compass guiding telecom success towards a trajectory of customer-centric growth.