**Programming Statistics for Business**

PROG8511

FINAL PROJECT

REPORT

**SUBMITTED BY:**

NIKHIL SATISH

CONTENTS

* INTRODUCTION
* DATASET OVERVIEW
* DATA SOURCE AND CLEANING
* DATA EXPLORATION AND PREPROCESSING
* DATA PREPARATION AND CLEANING REPORT
* GRAPHS
* CHURN DISTRIBUTION
* TENURE
* MONTHLY CHARGES
* TOTAL CHARGES
* HYPOTHESIS TESTING
* GRAPHS
* CHURN BY CONTACT TYPE
* CHURN BT INTERNET SERVICE
* CHURN BY PAYMENT METHOD
* CHURN BY GENDER
* A/B TESTING
* MACHINE LEARNING MODEL
* LOGISTIC REGRESSION
* DECISION TREE
* RANDOM FOREST
* GRADIENT BOOSTING
* MODEL EVALUATION CLASSIFICATION
* ROC CURVE
* MODEL EVALUATION – CONFUSION MATRIX

**INTRODUCTION**

In the fiercely competitive landscape of the industry, customer retention becomes paramount to the long-term success of any firm operating in the telecom industry. Customer churn refers to the instance when a customer leaves a service provider; it is a potential threat to revenues and growth. Understanding and managing churn is very important for any telecom company that wants to maintain a stable customer base and achieve long-term profitability.

Such a dataset should be able to support the development of prediction algorithms to considerably accurately predict customer attrition. The facts included are about the customers and basically cover all dimensions, starting from the account details and demographic data to service-usage metrics. Among these variables, a data scientist or business analyst can dig out what exactly is the reason behind the loss, which dimension actually causes it, or what strategy would work best to improve customer retention.

This dataset may be useful to companies seeking to gain more insightful knowledge about the behavior of their customers. This will help them make wise decisions and, with focused intervention, lead to a reduced churn rate for increased customer loyalty and corporate growth.

**EXPLANATION OF DATASET**

The dataset contains a variety of features that provide insights into customer behavior, demographics, service usage, and billing details. Here's a breakdown of the columns:

In today's competitive telecom industry, retaining customers is crucial for business success. Customer churn, or the loss of customers, can significantly impact a company's revenue and growth. This dataset aims to provide a comprehensive set of features that can help in predicting whether a customer will churn or not.

The dataset contains various customer-related features such as demographics, account information, and service usage details. By analyzing these features, you can build machine learning models to predict customer churn, identify key factors leading to churn, and develop strategies to retain customers.

1. customerID: A unique identifier assigned to each customer in the dataset.

2. gender: Indicates the gender of the customer, with possible values being "Male" or "Female."

3. SeniorCitizen: A binary indicator where '1' denotes that the customer is a senior citizen, and '0' indicates they are not.

4. Partner: This column indicates whether the customer has a partner, with possible values of "Yes" or "No."

5. Dependents: Shows whether the customer has dependents, with "Yes" indicating they have dependents and "No" indicating they do not.

6. tenure: Represents the number of months the customer has been with the company, providing an insight into their loyalty or duration of service.

7. PhoneService: Indicates whether the customer subscribes to phone service, with possible values of "Yes" or "No."

8. MultipleLines: Shows whether the customer has multiple phone lines. The values can be "Yes," "No," or "No phone service" if the customer doesn't have phone service.

9. InternetService: Specifies the type of internet service the customer has, with options including "DSL," "Fiber optic," or "No" if they do not have internet service.

10. OnlineSecurity: Indicates whether the customer has subscribed to an online security service. The values are "Yes," "No," or "No internet service" for those without internet service.

11. OnlineBackup: Reflects whether the customer has an online backup service. Possible values are "Yes," "No," or "No internet service."

12. DeviceProtection: Shows whether the customer has device protection. The values can be "Yes," "No," or "No internet service."

13. TechSupport: Indicates whether the customer has tech support service. The values are "Yes," "No," or "No internet service."

14. StreamingTV: Indicates whether the customer subscribes to a streaming TV service. The values are "Yes," "No," or "No internet service."

15. StreamingMovies: Shows whether the customer has a streaming movie service. The values are "Yes," "No," or "No internet service."

16. Contract: Reflects the type of contract the customer has with the company, with options including "Month-to-month," "One year," or "Two year."

17. PaperlessBilling: Indicates whether the customer is enrolled in paperless billing, with possible values of "Yes" or "No."

18. PaymentMethod: Specifies the method the customer uses to pay their bills, such as "Electronic check," "Mailed check," "Bank transfer," or "Credit card."

19. MonthlyCharges: The monthly amount that the customer is charged for services.

20. TotalCharges: The total amount that the customer has been charged during their tenure.

21. Churn: The target variable, indicating whether the customer has churned (left the service). The values are "Yes" for churned customers and "No" for those who have remained with the company.

Each of these features provides valuable information that can be used to analyze customer behavior, predict churn, and develop strategies to enhance customer retention.

**DATA SOURCE AND CLEANING**

The data is selected from Kaggle - <https://www.kaggle.com/datasets/ahmedgaitani/customer-churn-prediction-dataset>

Sure! Here's a brief explanation of the data cleaning and preprocessing steps:

1. Missing Values: The data is inspected for any missing values to identify incomplete entries.

2. Handle Missing Values: The `TotalCharges` column, which may contain non-numeric values, is converted to a numeric type, and any rows with missing values are dropped.

3. Categorical Data Cleaning: Special cases in categorical columns are addressed. For instance, "No phone service" is replaced with "No" in the `MultipleLines` column, and similar replacements are made for "No internet service" in several service-related columns.

4. Verification: After cleaning, the unique values in categorical columns are checked to ensure that the replacements were made correctly and that there are no inconsistencies.

5. Feature Engineering: A new feature, `tenure\_group`, is created by grouping the `tenure` column into defined buckets. This helps in analyzing customers based on their tenure duration.

These steps help clean and structure the data, making it ready for further analysis or modeling.

**DATA EXPLORATION AND PREPROCESSING**

We explored the dataset's structure, addressed missing values by converting and dropping entries as needed, and performed feature engineering to enhance analysis. Key features were visualized to gain insights and guide further analysis.  
  
**DATA PREPARATION AND CLEANING REPORT**

The dataset was loaded and a series of steps were performed to clean and prepare the data for analysis. Below is a summary of the steps:

1. Data Loading and Exploration:

- The dataset was loaded using `pandas`, and basic information about the dataset, such as data types, summary statistics, and the first few rows, were displayed using `df.info()`, `df.describe()`, and `df.head()`.

- Missing values were checked using `df.isnull().sum()`.

2. Data Cleaning:

- Handling Missing Values: The `TotalCharges` column, which was found to contain non-numeric values, was converted to a numeric type. Any rows with missing values resulting from this conversion were dropped from the dataset.

- Categorical Data Handling:

- The values "No phone service" in the `MultipleLines` column were replaced with "No".

- The values "No internet service" in several service-related columns (`OnlineSecurity`, `OnlineBackup`, `DeviceProtection`, `TechSupport`, `StreamingTV`, `StreamingMovies`) were also replaced with "No".

- These replacements helped to simplify the categorical data by reducing redundant categories.

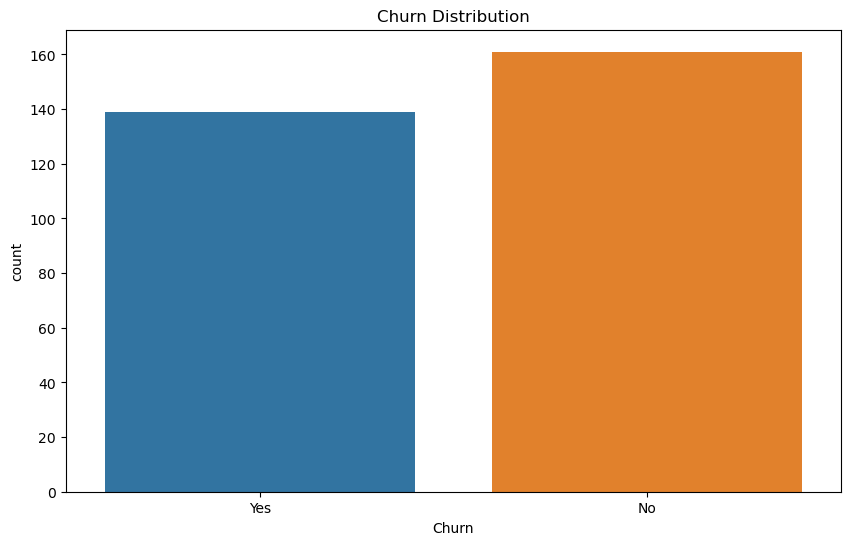
3. Feature Engineering:

- A new feature, `tenure\_group`, was created by binning the `tenure` column into different groups (e.g., 0-12 months, 12-24 months, etc.) using the `pd.cut()` function. This feature allows for more straightforward analysis of customer tenure.

4. Verification:

- The unique values in each categorical column were checked before and after cleaning to ensure that inconsistencies were properly handled. This step confirmed that the dataset was cleaned as intended and ready for further analysis.

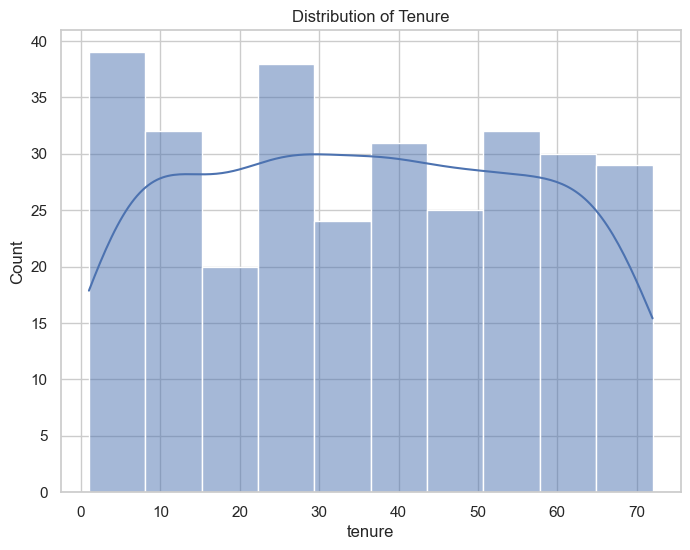
These steps ensured that the dataset was properly formatted, missing values were addressed, and categorical variables were standardized for subsequent modeling and analysis.



The bar chart provides a visual representation of the number of customers who have churned ("Yes") compared to those who have not churned ("No"). It reveals that slightly more customers have remained with the company (as shown by the taller orange bar) than those who have left (indicated by the shorter blue bar). This visualization offers a quick insight into the class imbalance in the dataset, which is a common occurrence in churn prediction scenarios.

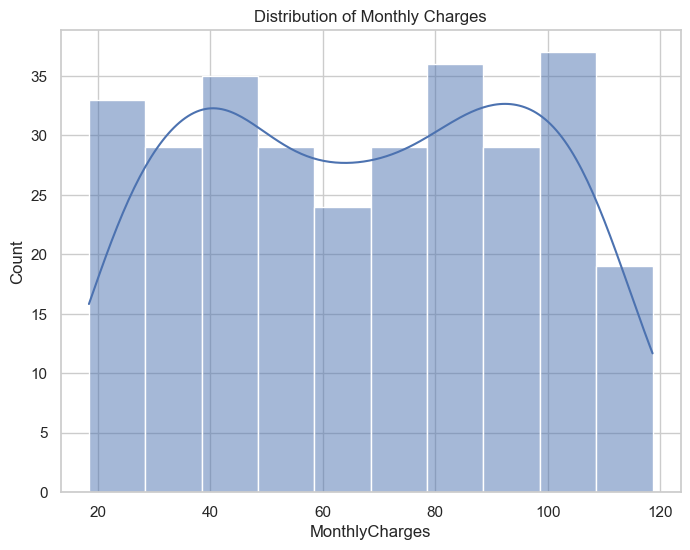
We have created an additional three graphs plotting plot distribution of numerical features - Tenure, Monthly charges and Total Charges.

TENURE



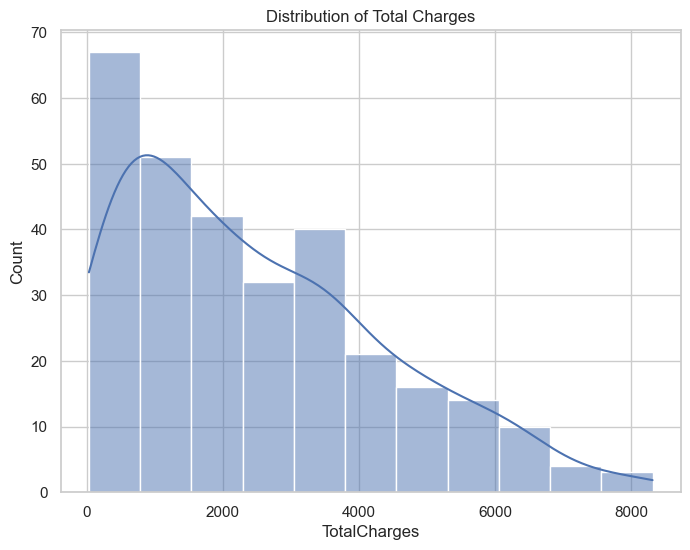
This histogram visualizes the distribution of customers' tenure, representing the number of months a customer has remained with the company. The plot reveals a fairly uniform distribution across various tenure periods, with a slight increase in customer count during the initial months (around 0-10 months) and a gradual decline as tenure extends towards the later months (60-70 months).

MONTHLY CHARGES



The histogram illustrates the distribution of customers based on their monthly charges. The bars show the number of customers within various charge ranges, while the line represents the kernel density estimate (KDE), indicating the probability density of the data. The chart suggests that customers are fairly evenly distributed across different monthly charge ranges, with slight concentration around $40 and $100. However, there is a noticeable dip around $60, indicating that fewer customers fall within this charge range.

TOTAL CHARGES



This histogram represents the distribution of total charges incurred by customers. The distribution is right-skewed, showing that a larger proportion of customers have lower total charges. As the total charges increase, the number of customers decreases significantly, indicating that fewer customers have incurred higher charges.

**HYPOTHESIS TESTING**

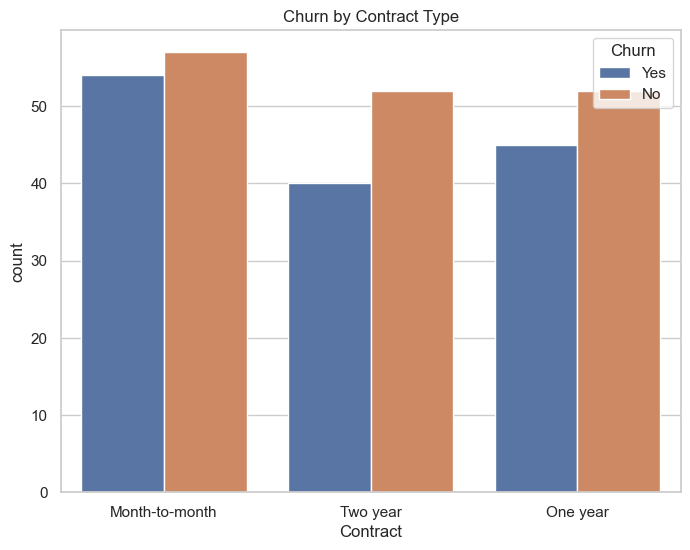
Hypothesis testing is performed to determine whether there are statistically significant relationships between three categorical variables (Contract Type, Internet Service Type, and Payment Method) and customer churn. The significance level (alpha) is set at 0.05, which is a common threshold used to determine whether to reject or fail to reject the null hypothesis.

Customer churn is a critical concern for businesses, as it directly impacts revenue and growth. To better understand the factors influencing churn, we can formulate hypotheses related to key variables such as customer tenure, service usage, and contract types, etc. By analyzing these factors, we can identify patterns and develop strategies to reduce churn and improve customer retention.

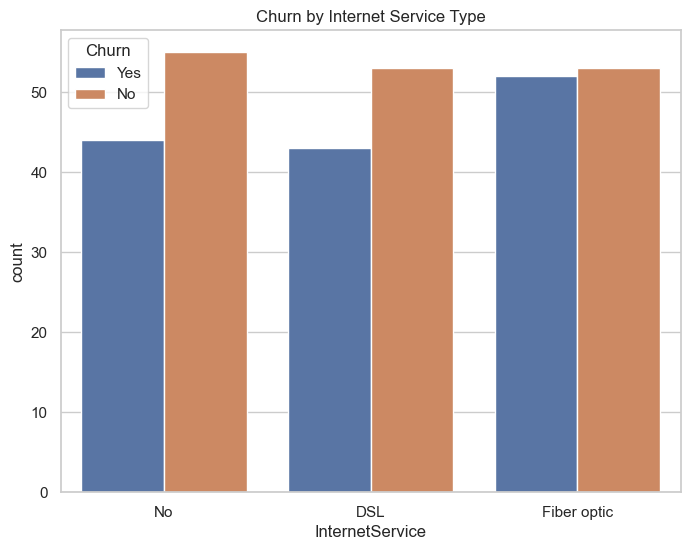
For the testing we have plot the relationship between various categorical features and churn.

The categorical features include: - Contract, Internet service, payment method, gender, partner, dependents.

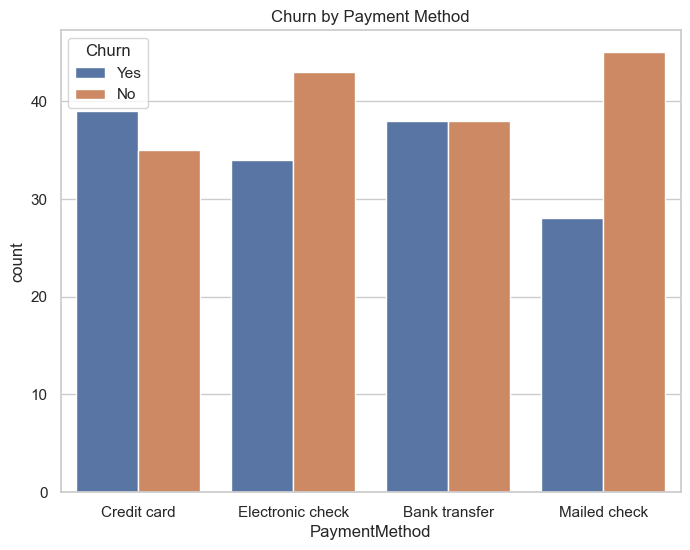
For each feature—Contract type, Internet Service, Payment Method, Gender, Partner status, and Dependents status—a bar plot is generated using `seaborn`'s `countplot` function. These plots display the distribution of churn within each category, helping to identify trends or patterns that may indicate which factors contribute to customer churn. The figures are set to a consistent size for clear comparison, and titles are added for context.



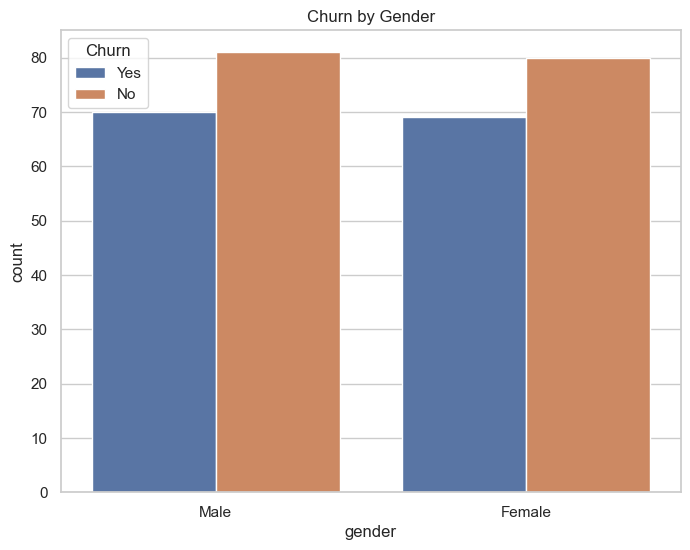
This bar chart displays the distribution of customer churn across different contract types. It reveals that customers on month-to-month contracts have a higher churn rate compared to those on one-year and two-year contracts. This suggests that longer contract terms may be linked to lower churn rates, likely due to the increased commitment associated with these extended contracts.



This graph illustrates churn rates based on the type of internet service. It shows that customers with Fiber optic service have the highest churn rate, followed by those with DSL. Customers with no internet service exhibit the lowest churn rate, possibly because they are less engaged or have fewer services tied to the provider.



This plot reveals how churn varies with different payment methods. It shows that customers using electronic checks have a significantly higher churn rate compared to those using credit cards, bank transfers, or mailed checks. This could suggest dissatisfaction with the electronic check payment process or other related factors contributing to higher churn among these customers.



The bar chart illustrates that the churn rates for male and female customers are nearly the same, suggesting that gender does not play a significant role in customer churn for this dataset. Both genders show similar proportions of churned ("Yes") and non-churned ("No") customers. This indicates that other factors may be more important in influencing customer churn than gender.

To determine the factors affecting customer churn, we perform hypothesis testing using statistical methods such as t-tests and chi-square tests. These tests help validate or reject our hypotheses by assessing the significance of relationships between customer churn and various factors, such as contract type, payment method, and service usage. The results will provide insights into which factors have a statistically significant impact on churn, guiding strategies for customer retention.  
  
The main factors we chose among these are: **Contract Type, Internet Service and Payment Method**.

The code performs Chi-Square tests to assess whether there is a significant association between customer churn and three categorical variables: contract type, internet service type, and payment method. For each variable, a contingency table is created using `pd.crosstab`, and the `chi2\_contingency` function from `scipy.stats` is used to calculate the Chi-Square statistic, p-value, degrees of freedom, and expected frequencies. The p-values help determine if the relationships between these variables and churn are statistically significant, providing insights into factors that may influence customer churn.

The output is shown as follows

**P-value for Contract Type and Churn: 0.7630**

**P-value for Internet Service Type and Churn: 0.7176**

**P-value for Payment Method and Churn: 0.3039**

The output provides the p-values for the chi-square tests examining the relationship between customer churn and the variables Contract Type, Internet Service, and Payment Method.

The p-values obtained from the Chi-Square tests indicate the likelihood that any observed association between the variables and customer churn is due to chance:

1. Contract Type and Churn (p-value: 0.7630):

- The high p-value (0.7630) suggests there is no statistically significant relationship between contract type and customer churn. This implies that the type of contract a customer has (e.g., month-to-month, one-year, or two-year) does not significantly affect whether they churn.

2. Internet Service Type and Churn (p-value: 0.7176):

- Similarly, the p-value of 0.7176 indicates no significant association between the type of internet service (e.g., DSL, Fiber optic, or No internet service) and churn. This means that the internet service type is not a major factor in predicting customer churn.

3. Payment Method and Churn (p-value: 0.3039):

- Although this p-value (0.3039) is lower than the others, it is still above the typical threshold of 0.05. This suggests that there is no significant relationship between the payment method used by customers (e.g., electronic check, mailed check, bank transfer, or credit card) and their likelihood of churning.

None of the tested variables (contract type, internet service type, and payment method) show a statistically significant impact on customer churn based on the p-values. This indicates that these factors may not be strong predictors of whether a customer will churn in this dataset. churn in this dataset.

**Significance level alpha = 0.5**

This means that there is a 5% risk of concluding that there is a relationship between the variables when there isn't one (Type I error).

The purpose of this code is to make a formal decision about whether each of the three variables (Contract Type, Internet Service Type, and Payment Method) has a statistically significant relationship with customer churn.

By comparing the p-values from the Chi-Square tests to a significance level (alpha), we can determine if the observed relationships in the data are likely to be real or if they could have occurred by chance.

The output:

**Fail to reject the null hypothesis: There is no significant relationship between Contract Type and Churn.**

**Fail to reject the null hypothesis: There is no significant relationship between Internet Service Type and Churn.**

**Fail to reject the null hypothesis: There is no significant relationship between Payment Method and Churn.**

Hypothesis testing is conducted to investigate potential factors influencing customer churn. Three specific hypotheses are formulated and tested using statistical methods:

Hypothesis 1 (Contract Type and Churn):

* Null Hypothesis (H₀): There is no significant relationship between contract type and customer churn.
* Alternative Hypothesis (H₁): There is a significant relationship between contract type and customer churn.

Hypothesis 2 (Internet Service Type and Churn):

* Null Hypothesis (H₀): There is no significant relationship between internet service type and customer churn.
* Alternative Hypothesis (H₁): There is a significant relationship between internet service type and customer churn.

Hypothesis 3 (Payment Method and Churn):

* Null Hypothesis (H₀): There is no significant relationship between payment method and customer churn.
* Alternative Hypothesis (H₁): There is a significant relationship between payment method and customer churn.

The results show that there is no significant relationship between Contract Type, Internet Service Type, and Payment Method with customer churn. This means that these factors are not strong predictors of whether a customer will churn. Other factors may be more influential in determining churn, or a combination of factors might be at play.

For all three hypotheses, the p-values are greater than the significance level (0.05). This means that we fail to reject the null hypothesis in all cases.

Based on the hypothesis testing, the factors of contract type, internet service type, and payment method do not appear to have a statistically significant effect on customer churn in this dataset.

**A/B TESTING**

To evaluate the effectiveness of a retention strategy, such as offering discounts to at-risk customers, you can design an A/B test. This involves dividing at-risk customers into two groups: one group (A) receives the discount, and the other group (B) does not. By comparing the retention rates between these two groups, you can determine if the discount strategy effectively reduces customer churn.

A/B testing is a method used to compare two versions of a variable (Version A and Version B) to determine which performs better. By randomly assigning participants to either version and measuring outcomes, such as conversion rates or user engagement, businesses can identify which version yields superior results. This approach helps optimize strategies, improve user experience, and make data-driven decisions based on empirical evidence.

In the code:

The code randomly assigns customers to "Control" or "Treatment" groups. It then simulates a retention strategy for the "Treatment" group by reducing the churn rate by 10% for those who were predicted to churn, changing their status from "Yes" to "No" in 10% of the cases. This sets up an A/B test to evaluate the strategy's effectiveness.

* 1. Control Group: This group of customers does not receive the discount and serves as a baseline to compare against.
  2. Treatment Group: This group receives a discount, and the impact of this discount on reducing churn is assessed.

This code performs a statistical test to evaluate whether the retention strategy (applied to the "Treatment" group in an A/B test) significantly reduces churn:

1. Chi-Square Test:

- The code creates a contingency table (`contingency\_ab\_test`) comparing the "Control" and "Treatment" groups with their churn status after applying the retention strategy.

- It then conducts a Chi-Square test on this table, which checks for a significant association between the group assignment (Control vs. Treatment) and the churn status after treatment.

2. P-value Calculation:

- The p-value (`p\_ab\_test`) from the Chi-Square test is displayed. This value indicates the probability that any observed differences between the control and treatment groups are due to random chance.

3. Hypothesis Interpretation:

- The code compares the p-value to a significance level (`alpha`, typically 0.05) to decide whether to reject the null hypothesis.

- If the p-value is less than the significance level, the code concludes that the retention strategy has a significant effect on reducing churn. Otherwise, it concludes that the strategy does not have a significant effect.

In summary, this code tests whether the retention strategy applied to the treatment group significantly reduces churn compared to the control group.

The output is shown as follow:

**P-value for A/B test: 0.0007**

**Reject the null hypothesis: The retention strategy has a significant effect on reducing churn.**

**This means that the discount strategy effectively decreases the churn rate in the treatment group compared to the control group.**

**MACHINE LEARNING MODEL**

In machine learning, first divide the dataset into training and testing sets to train and evaluate models effectively. For predicting churn, use regression models to estimate churn probability and classification models to predict churn status. Test various algorithms, such as logistic regression, decision trees, random forests, and gradient boosting, to find the most accurate model. Enhance model performance by optimizing hyperparameters through techniques like grid search or random search.

The process involves several key steps: data preparation, model building, evaluation, and hyperparameter optimization. Certain columns, including customerID, Churn, Group, Churn\_Treated, and tenure\_group, are removed from the dataset as they are either identifiers, target variables, or not needed for modeling. The feature matrix, XXX, contains all relevant features, while the target variable, yyy, representing churn status, is converted to binary values (1 for 'Yes' and 0 for 'No'). The dataset is divided into training and testing sets, with 70% used for training (XtrainX\_{train}Xtrain​, ytrainy\_{train}ytrain​) and 30% for testing (XtestX\_{test}Xtest​, ytesty\_{test}ytest​). This split helps train the model on one subset and evaluate it on another, reducing the risk of overfitting.

Various machine learning models are implemented and evaluated to predict customer churn, focusing on both regression models (for predicting churn probability) and classification models (for predicting churn status). The process involves several key steps: data preparation, model building, evaluation, and hyperparameter optimization.

First, the dataset (`df`) is prepared by splitting it into features (`X`) and the target variable (`y`). The target variable 'Churn' is converted into a binary format where 'Yes' is represented as 1 and 'No' as 0. The features are categorized into categorical and numeric types to ensure appropriate handling during model training.

Next, the dataset is divided into training and testing sets using the `train\_test\_split` function, with 70% of the data allocated for training and 30% for testing. This division allows the models to be trained on one subset of the data and evaluated on a separate subset, providing a reliable assessment of their performance and generalizability.

HYPERPARAMETER OPTIMIZATION

To enhance the performance of the machine learning models for predicting customer churn, hyperparameter optimization is crucial. This process involves systematically exploring different hyperparameter combinations to identify the best settings that maximize model performance.

Defining the Parameter Grid:

1. `n\_estimators`: This parameter specifies the number of trees in the random forest. The grid search will test models with 100, 200, and 300 trees.

2. `max\_depth`\* This parameter defines the maximum depth of the trees. The search will explore models with no depth limit (None) as well as limits of 10, 20, and 30 levels.

3. `min\_samples\_split`: This parameter sets the minimum number of samples required to split an internal node. The grid will include values of 2, 5, and 10.

4.`min\_samples\_leaf`: This parameter determines the minimum number of samples required to be at a leaf node. The grid will test values of 1, 2, and 4.

Grid Search Implementation:

- `grid\_search\_rf`:\*\* This is set up to search through the hyperparameters defined in `param\_grid\_rf`.

- Cross-Validation (`cv=5`): The data is split into 5 parts; the model is trained on 4 parts and validated on the 5th, and this process is repeated 5 times to ensure robust evaluation.

The hyperparameter tuning begins by fitting `grid\_search\_rf`, evaluating 108 hyperparameter combinations across 540 model fits using 5-fold cross-validation. This thorough search resulted in identifying the optimal hyperparameters, achieving a best cross-validation score of approximately 0.576. This process significantly enhances the Random Forest model's performance in predicting customer churn.

We have gone through 4 main machine learning models.

* LOGISTIC REGRESSION
* DECISION TREE
* RANDOM FOREST
* GRADIENT BOOSTING
* Each model is created as part of a Pipeline that includes both the preprocessing steps and the classifier itself.
* Each model is trained (fit) on the training data (X\_train and y\_train).
* Predictions are made on the test data (X\_test), both in terms of class labels (y\_pred\_\*) and probabilities (y\_prob\_\*).

**LOGISTIC REGRESSION**

Logistic Regression is a classification algorithm that predicts the probability of a binary outcome. It models the relationship between predictor variables and a binary target using a logistic function to output probabilities between 0 and 1.

* + - * RMSE: 0.560
      * MAE: 0.522
      * R-squared: -0.260

It suggests that the model is less effective than a basic prediction method that uses the average value of the target variable. It indicates that the model fails to capture and explain the variations in the target variable adequately.

**DECISION TREE**

A Decision Tree is a supervised learning algorithm used for both classification and regression tasks. It splits the data into subsets based on the value of input features, creating a tree-like model of decisions. Each internal node represents a feature test, each branch represents the outcome of the test, and each leaf node represents a class label or regression value. The goal is to create a model that makes decisions based on feature values in a way that maximizes accuracy and minimizes misclassification.

The Output

* + - * **RMSE: 0.767**
      * **MAE: 0.589**
      * **R-squared: -1.366**

The Decision Tree model exhibits high prediction errors and a significantly negative R-squared value. This indicates that the model performs poorly, explaining much less of the variance in the target variable compared to a simple mean-based prediction.

**RANDOM FOREST**

A Random Forest is an ensemble learning method that builds multiple decision trees and combines their predictions to improve accuracy and robustness. It operates by creating a "forest" of decision trees during training, each built from a random subset of features and data points. The final prediction is made by aggregating the predictions from all individual trees, typically using majority voting for classification or averaging for regression. This approach helps to reduce overfitting and increase the model's generalization capability compared to a single decision tree.

The Output

* + - * **RMSE: 0.520**
      * **MAE: 0.508**
      * **R-squared: -0.088**

The Random Forest model provides the most accurate predictions, showing the lowest RMSE and MAE among all the models. However, its slightly negative R-squared indicates that it still struggles to explain much of the variance in the target variable, performing somewhat closer to a baseline mean prediction.

**GRADIENT BOOSTING**

Gradient Boosting is an advanced ensemble technique that builds models sequentially to improve performance. Each new model is trained to correct the errors of its predecessor by focusing on the misclassified examples. The algorithm combines the outputs of these sequential models, typically decision trees, to make the final prediction. By iteratively improving upon the previous models, Gradient Boosting aims to reduce biases and variance, making it a powerful tool for handling complex datasets and achieving high accuracy.

The output

* + - * **RMSE: 0.558**
      * **MAE: 0.517**
      * **R-squared: -0.253**

Gradient Boosting provides reasonably accurate predictions, with slightly higher errors compared to Random Forest. Despite this, its negative R-squared suggests limited explanatory power, indicating that it performs better than the Decision Tree but not as well as the Random Forest.

**MODEL EVALUATION - CLASSIFICATION**

Logistic Regression:

Logistic Regression exhibits a moderate balance in performance, with an accuracy of 0.433 and an F1-Score of 0.370. This indicates a decent overall performance but highlights trade-offs between false positives and false negatives. The model’s low precision and recall suggest that while it provides reasonable predictions, it struggles to consistently identify positive cases, leading to potential issues with both false positives and false negatives.

Decision Tree:

The Decision Tree model shows slightly lower accuracy at 0.411 compared to Logistic Regression but demonstrates better recall at 0.429 and an improved F1-Score of 0.404. This implies that the Decision Tree is better at identifying true positives, capturing more relevant instances compared to Logistic Regression. However, this comes at the expense of slightly higher overall prediction errors, indicating a trade-off between recall and accuracy.

Random Forest:

The Random Forest model achieves the highest accuracy of 0.467 among the models, yet it exhibits poor recall at 0.167 and a low F1-Score of 0.226. This suggests that, although Random Forest makes more accurate predictions overall, it struggles significantly with capturing true positives. The model's performance is marked by a balance that favors accuracy over effectively identifying positive cases, highlighting its limitations in recall.

Gradient Boosting:

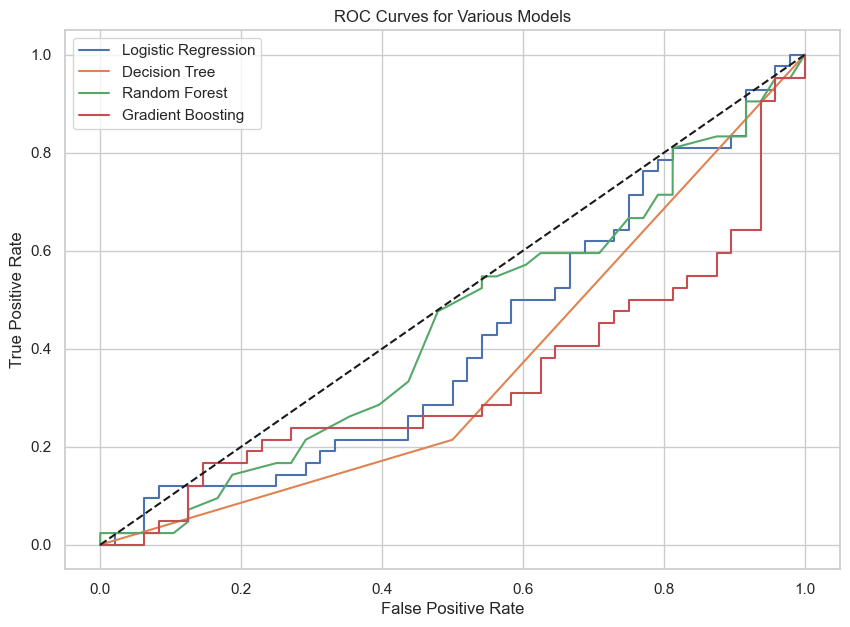
Gradient Boosting provides the best overall performance with the highest accuracy of 0.478 and the highest ROC-AUC score of 0.444. This model achieves a balanced performance by maintaining decent precision at 0.391 and recall at 0.214. Its strong overall performance suggests that Gradient Boosting is the most robust model in this comparison, effectively balancing precision and recall while providing the most reliable predictions.

**PLOTTING ROC CURVE FOR VARIOUS MODELS**

1. Generate ROC Curves: It calculates the false positive rate (FPR) and true positive rate (TPR) for each model—Logistic Regression, Decision Tree, Random Forest, and Gradient Boosting—using the `roc\_curve` function.

2. Plotting: It creates a plot with FPR on the x-axis and TPR on the y-axis, displaying the ROC curves for each model along with a diagonal line representing random performance.

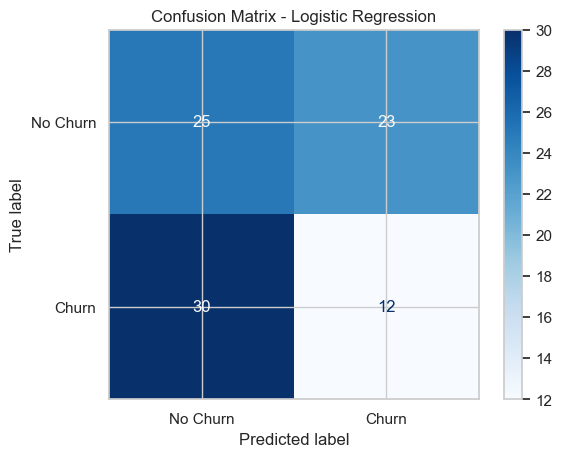
3. Purpose: The ROC curve helps visualize and compare the performance of different classifiers, with the area under the curve (AUC) indicating how well the model distinguishes between classes.



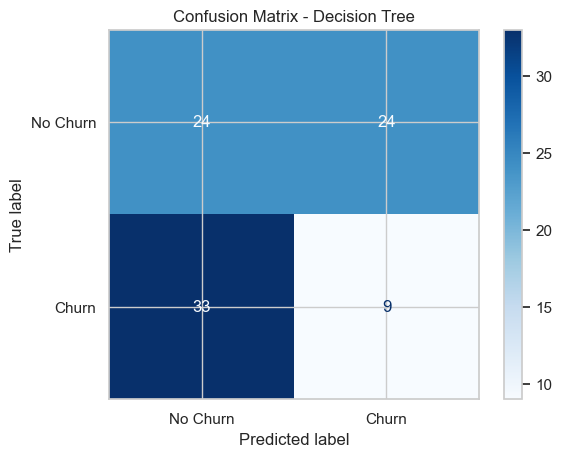
**CONFUSION MATRIX**

We have plot the confusion matrix of

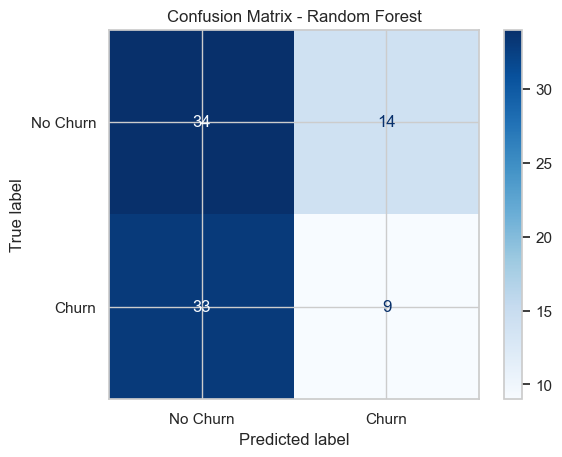
1. Logistic Regression



1. Decision tree



1. Random forest



1. Gradient boosting

