Predict Customer Churn and Build Insights for Retention

Customer Churn Prediction: EDA

1. Dataset Overview:

The dataset consists of 200 records and 11 features, including customer demographics, subscription details, and churn status, which are key for churn prediction.

2. Handling Missing Values:

Missing values in the dataset were handled through **imputation** for numerical features like totalcharges and monthlycharges. **Row deletion** was considered but imputation was preferred to maintain data continuity.

3. Univariate Analysis:

- **Churn Distribution**: A count plot visualized the proportion of churned vs. non-churned customers. The churn variable is binary: "Yes" (churned) and "No" (not churned).
- **Feature Distributions**: tenure, monthlycharges, and totalcharges were analyzed using histograms and KDE plots.
    - monthlycharges and totalcharges were skewed, while tenure showed a more uniform distribution.
- **Outliers**: Boxplots revealed outliers, especially in totalcharges, indicating a need for data preprocessing.

4. Bivariate Analysis:

- **Correlation Matrix**: A heatmap indicated moderate correlation between monthlycharges and totalcharges.
- **Scatter Plots**: Plots between tenure vs. monthlycharges and totalcharges vs. monthlycharges suggested a relationship where longer tenures generally correlate with lower charges.
- **Average Charges by Churn**: Bar charts showed that churned customers had shorter tenures and higher monthly charges.

5. Customer Segmentation:

- **Pie Chart**: Visualized the churn distribution, showing the proportion of churned and non-churned customers.
- **KDE Plot**: The KDE plot showed that churned customers tend to have higher monthlycharges, suggesting that pricing could be a factor influencing churn.

6. Key Insights:

- **Churned Customers**: Customers with shorter tenures and higher monthly charges are more likely to churn. This highlights the importance of pricing and retention strategies.

- **Outliers**: Outliers in totalcharges and other numerical features need to be addressed through data cleaning techniques such as capping or transformation.

Feature Engineering and Data Preprocessing:

1. Feature Creation:

- **Tenure Adjustments**: The tenure column was modified to handle zero values, replacing them with NaN to prevent potential issues during calculations.
- **New Feature - AvgMonthlySpend**: A new feature, AvgMonthlySpend, was created by dividing totalcharges by tenure. This represents the average amount spent per month by a customer.
- **New Feature - EngagementScore**: An EngagementScore was derived as the product of tenure and monthlycharges. This score indicates the level of customer engagement, with higher values representing customers who are more engaged (i.e., have been with the service longer and are paying higher monthly charges).

2. Feature Encoding:

- **Label Encoding**: The binary categorical variable churn was label-encoded to convert the "Yes"/"No" values into binary values (0 and 1). This encoding helps in the machine learning models that require numeric input.
- **One-Hot Encoding**: Multi-class categorical variables such as contract, internetservice, and paymentmethod were one-hot encoded to create dummy variables. This ensures that the models can process these variables without assuming an ordinal relationship.

3. Scaling Numerical Features:

- **Standardization**: Numerical features, including tenure, monthlycharges, totalcharges, AvgMonthlySpend, and EngagementScore, were standardized using the **StandardScaler** to ensure that all features have a mean of 0 and a standard deviation of 1. This step is essential for many machine learning algorithms that are sensitive to feature scaling.
- **Alternative Scaling (Min-Max)**: Although **Min-Max scaling** was considered as an alternative, it was not used in this implementation. Min-Max scaling rescales the features to a range of [0, 1], which may be useful for certain models.

Predictive Modeling and Evaluation:

1. Dataset Split:

- The dataset was divided into **training** and **testing** sets using train_test_split.
    - **Training Set**: 70% of the data
    - **Testing Set**: 30% of the data
- The **stratify** parameter ensured that the distribution of the target variable churn was preserved across both sets.

2. Model Training:

Three models were trained and evaluated:

- **Logistic Regression**
- **Decision Tree Classifier**
- **Random Forest Classifier**

A preprocessing pipeline was applied to handle categorical encoding and numerical scaling:

- **Categorical Encoding**: OneHotEncoding was used for categorical variables.
- **Numerical Scaling**: StandardScaler was applied to normalize numerical features.

Cross-validation (5-fold) was performed on the training set to evaluate the models' performance based on accuracy:

- **Logistic Regression**: The model achieved a high accuracy, showing good potential for binary classification.
- **Decision Tree**: The Decision Tree showed competitive accuracy but may overfit without proper tuning.
- **Random Forest**: The Random Forest classifier generally outperformed the other models, demonstrating robustness and high accuracy.

3. Model Evaluation:

The models were evaluated on the test set using the following metrics:

- **Accuracy**
- **Precision**
- **Recall**
- **F1-Score**

The performance metrics were calculated for each model, providing insights into their effectiveness for predicting customer churn:

- **Logistic Regression**: Balanced precision and recall, with a good F1-score.
- **Decision Tree**: The decision tree model showed good accuracy, but its precision and recall were lower, indicating potential overfitting.
- **Random Forest**: This model consistently performed best, with high precision, recall, and F1-score, making it the most reliable for churn prediction.

4. Confusion Matrices:

Confusion matrices were generated for each model to visualize true positives, false positives, true negatives, and false negatives. The confusion matrix for each model revealed how well the models classified churned vs. non-churned customers.

5. Best-Performing Model:

The **Random Forest Classifier** emerged as the best model, with the highest F1-score, indicating its superior ability to balance precision and recall for churn prediction. This model was then trained on the entire training dataset and evaluated on the test set.

6. Visualization:

The confusion matrix for the best-performing model (Random Forest) was visualized using a heatmap, which helped in understanding the model's classification accuracy and errors.

Summary of Business Insights and Recommendations:

Key Business Insights:

1. **Churn Prediction Accuracy**: The Random Forest Classifier has shown high accuracy in predicting customer churn, providing a reliable tool for identifying at-risk customers and implementing preventive actions.
2. **Feature Importance**: Key features such as tenure, monthlycharges, totalcharges, AvgMonthlySpend, and EngagementScore are crucial in predicting churn. Customers with longer tenure and lower monthly charges are less likely to churn, while higher charges and shorter tenure may indicate higher churn risk.
3. **Customer Segmentation**: By utilizing churn prediction results, customers can be segmented into high-risk and low-risk categories, allowing for more targeted retention and marketing strategies.

Recommendations:

1. **Targeted Retention Campaigns**: Use the churn prediction model to identify high-risk customers early and offer personalized retention strategies like discounts, loyalty rewards, or customized service packages.
2. **Customer Engagement**: Leverage the EngagementScore to measure and enhance customer engagement through loyalty programs, regular interactions, and value-added services to retain customers longer.
3. **Monitor At-Risk Customers**: Closely monitor high-risk customers for behavior changes (e.g., missed payments, reduced usage) and initiate proactive outreach to resolve concerns before they lead to churn.
4. **Pricing Strategy**: Consider revising pricing strategies, especially for customers with high monthly charges, offering flexible pricing or discounts for long-term customers to improve retention.
5. **Improve Customer Support**: Prioritize quick and effective customer support for high-risk customers to prevent dissatisfaction and churn.
6. **Feature Optimization**: Regularly track and adjust features like AvgMonthlySpend and EngagementScore to improve the churn prediction model and optimize retention strategies.

Conclusion:-

The customer churn prediction analysis provided valuable insights into the key factors affecting churn in a subscription-based service. Through Exploratory Data Analysis (EDA), we identified critical

patterns, such as the strong correlation between customer tenure and monthly charges, which are important predictors of churn. By engineering new features, such as average spend per month and customer engagement scores, we enhanced the model's predictive power.

We compared multiple models, including Logistic Regression, Decision Trees, and Random Forest. The Random Forest model emerged as the best performer, offering high accuracy and precision in predicting churn. Feature importance analysis revealed that tenure, monthly charges, and total charges were the most significant factors contributing to churn.

Additionally, a collaborative filtering-based recommendation engine was developed to suggest relevant services, improving customer engagement and satisfaction. Finally, a Flask API was created to deploy the churn prediction model, allowing seamless integration into real-world applications.

By utilizing these insights and models, businesses can implement targeted retention strategies, such as personalized offers, improved pricing, and engagement initiatives. This approach not only helps in reducing churn but also boosts customer loyalty, ultimately driving long-term growth and profitability.