Luxottica Churn Prediction Report

1. Introduction

This report details the process and results of predicting customer churn for Luxottica. The project includes steps from data collection, pre-processing, feature selection, model training, evaluation, and deployment, along with detailed statistical analysis and exploratory data analysis (EDA).

2. Data Collection

2.1 Data Source

- **Dataset**: luxottica_eyewear_Master.csv
- **Description**: Contains customer data including demographics, purchase history, customer support interactions, and churn status.

2.2 Data Overview

- **Total Records**: 100,015
- **Total Features**: 35 (including target variable Churn)

3. Data Pre-processing

3.1 Handling Missing Values

- Applied forward fill method to handle missing values ensuring no loss of data.

3.2 Encoding Categorical Variables

- Converted categorical variables into numerical format using one-hot encoding.

3.3 Balancing Target Variable

- Used SMOTE (Synthetic Minority Over-sampling Technique) to balance the target variable Churn and address class imbalance.

4. Statistical Analysis

4.1 ANOVA Test

- Used to determine if there are any statistically significant differences between the means of three or more independent (unrelated) groups.

4.2 T-Test

- Conducted independent t-tests to compare the means of two groups (churn vs non-churn).

4.3 Logit OLS Method

- Applied Logistic Regression (Logit) using the Ordinary Least Squares (OLS) method to assess the relationship between the dependent variable (churn) and independent variables.

4.4 Variance Inflation Factor (VIF)

- Calculated VIF to check for multicollinearity among features.

4.5 Normal Distribution

- Assessed the normality of data distribution using histograms and Q-Q plots.

4.6 Standard Normal Distribution

- Standardized the dataset and checked for normal distribution using Z-scores.

4.7 Pearson Correlation Coefficient

- Calculated Pearson correlation coefficients to assess the strength and direction of relationships between pairs of variables.

4.8 Central Tendency

- Evaluated mean, median, and mode to understand the central tendency of numerical variables.

4.9 Outlier Detection

- Identified outliers using box plots and the Z-score method.

5. Exploratory Data Analysis (EDA)

5.1 Univariate Analysis

- **Distribution Plots**: Visualized the distribution of individual features using histograms and density plots.
- **Box Plots**: Used to identify outliers and understand the spread of data.
- **Central Tendency Measures**: Calculated mean, median, and mode for numerical features.

5.2 Bivariate Analysis

- **Scatter Plots**: Visualized relationships between pairs of numerical variables.

- **Heatmaps**: Displayed correlation matrices to identify strong relationships between features.
- **Box Plots**: Compared distributions of numerical features against the target variable (churn).
- **Bar Plots**: Compared categorical feature distributions against the target variable (churn).

6. Feature Selection

6.1 Methodology

- Used SelectKBest with f_classif to identify the top 10 features most relevant to the target variable Churn.

6.2 Selected Features

- Customer_Support_Interactions
- Customer_Satisfaction
- Purchase_Frequency
- Lifetime_Value
- Average_Order_Value
- Number_of_Product_Categories_Purchased
- Loyalty_Program_Participation_Inactive
- Engagement_with_Promotions_Low
- Engagement_with_Promotions_Medium
- Age

6.3 All Variables in Dataset

- Customer ID
- Age
- Gender
- State
- Store Location
- Income Level
- Date of First Purchase
- Last Purchase Date
- Type of Eyewear
- Brand
- Model
- Price
- Discount Amount
- Last Interaction Type
- Churn
- Customer Satisfaction
- Product Usage
- Return/Exchange History

- Customer Support Interactions
- Social Media Engagement
- Referral Source
- Number of Product Categories Purchased
- Loyalty Program Participation
- Sales Driver Index
- Purchase Frequency
- Subscription Status
- Engagement with Promotions
- Customer Segmentation
- Complaint History
- Product Return Rate
- Cross-Sell/Upsell Success Rate
- Purchase Channel Loyalty
- Lifetime Value
- Average Order Value
- Feedback

7. Model Training

7.1 Data Splitting

- **Training Set**: 75%
- **Testing Set**: 25%

7.2 Applied Algorithms

- Random Forest Classifier
- Logistic Regression
- Decision Tree Classifier
- K-Nearest Neighbors
- Support Vector Machine
- Gradient Boosting Classifier
- Naive Bayes

8. Model Evaluation

8.1 Accuracy Scores

- **Random Forest**: 99.81%
- **Logistic Regression**: 99.68%
- **Decision Tree**: 99.72%

8.2 Confusion Matrix

- **Random Forest**:
    - True Positives: 6,589
    - True Negatives: 17,143

- False Positives: 44
- False Negatives: 0

8.3 ROC-AUC Scores

- **Random Forest**: 0.9973
- **Logistic Regression**: 0.9998
- **Decision Tree**: 0.9968

8.4 Classification Reports

- Detailed precision, recall, and F1-score metrics for each model.

9. Key Findings

- **Customer_Support_Interactions** and **Customer_Satisfaction** are significant predictors of churn.
- Customers with low **Engagement with Promotions** and inactive in **Loyalty Programs** are more likely to churn.
- High **Lifetime Value** and **Average Order Value** correlate with lower churn rates.
- Date consistency verified ensuring that **Last Purchase Date** is always after **First Purchase Date**.

10. Model Deployment

10.1 Model Selection

- **Random Forest Classifier** was selected for deployment due to its highest accuracy and ROC-AUC score.

10.2 Saving the Model

- The trained model was serialized and saved as a pickle file for deployment.

10.3 Deployment Strategy

- The model will be integrated into the customer management system to predict churn probability in real-time, allowing for proactive customer retention strategies.

10.4 Future Enhancements

- Continuous monitoring and retraining of the model with new data to maintain accuracy.
- Integration of more customer interaction data to further improve model performance.

11. Django Web Development

11.1 Web Interface

- A web-based interface was developed using the Django framework to facilitate real-time predictions by customer service teams.

11.2 User Interaction

- Users can input relevant customer data through a web form and receive an immediate prediction on whether the customer is likely to churn.

11.3 Backend Processing

- The backend handles data pre-processing, applying the trained Random Forest model, and scaling the inputs using the saved Min-Max scaler.

11.4 Deployment

- The Django application was deployed on a web server, providing an accessible tool for Luxottica's customer service teams to predict churn classification. This application allows users to input customer data and receive an immediate churn classification, empowering the team to take proactive measures in customer retention.

.