Churn Prediction Model Report

# 1. Introduction

This project aims to predict which users are most likely to churn (i.e., stop returning or purchasing) based on their interaction with the platform. We utilize a variety of data science techniques, including feature engineering, model training, and interpretation using SHAP values. The goal is to not only predict churn but also provide actionable business insights on why users are churning and how to retain them.

# 2. Approach

2.1 Data Exploration and Cleaning  
1. Dataset Loading: Load the CSV data into a Pandas DataFrame.  
2. Missing Values: Handle missing values appropriately (if any).  
3. Feature Engineering: Create new features such as price sums, averages, and categories based on user interactions.  
4. Churn Definition: Define churn as users who have not interacted with the platform in the last 30 days.  
  
 2.2 Data Visualization  
1. Bar Plots: Visualize the distribution of event types, top brands, and top categories.  
2. Line Plot: Show the trend of events over time.  
3. Scatter Plot: Show the relationship between price and event activity for churned vs. non-churned users.  
4. Pie & Donut Charts: Visualize the distribution of churned and non-churned users.  
  
 2.3 Model Training  
1. Feature Selection: Choose relevant features such as user interactions and price information.  
2. Train-Test Split: Split the data into training and testing sets (70-30).  
3. Model: Use XGBoost (eXtreme Gradient Boosting) to train a classification model on the data.  
4. Evaluation: Evaluate the model using classification metrics (accuracy, AUC score) and SHAP for interpretability.  
  
 2.4 Insights & Recommendations  
1. Churn Insights: Provide actionable recommendations based on the analysis of features contributing to churn.  
2. Model Interpretability: Use SHAP to explain the impact of features on model predictions.

# 3. Installation

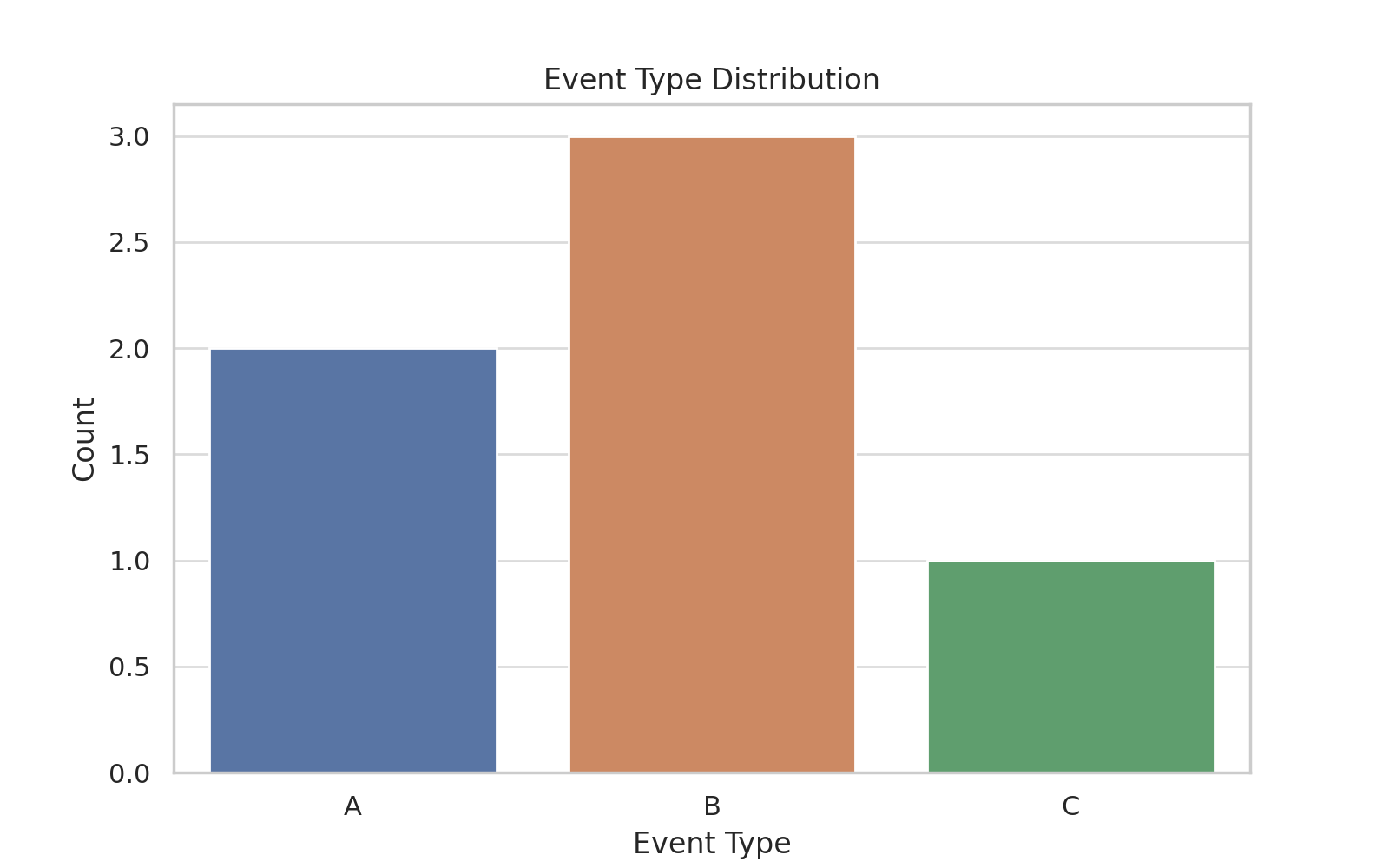
To get started with the project, follow the installation instructions below.  
  
 Prerequisites  
You’ll need the following installed on your system:  
- Python 3.x  
- Google Colab or Jupyter Notebook (optional, but recommended for interactive development)  
  
 3.1 Install Required Libraries  
You can install the required libraries using `pip`. Run the following command:  
```bash  
pip install pandas numpy matplotlib seaborn scikit-learn xgboost shap  
```  
  
Alternatively, if you're using Google Colab or Jupyter Notebook, you can run the command inside a notebook cell (uncomment the first line if you're on Google Colab):  
```python  
 !pip install pandas numpy matplotlib seaborn scikit-learn xgboost shap  
```

# 4. Execution

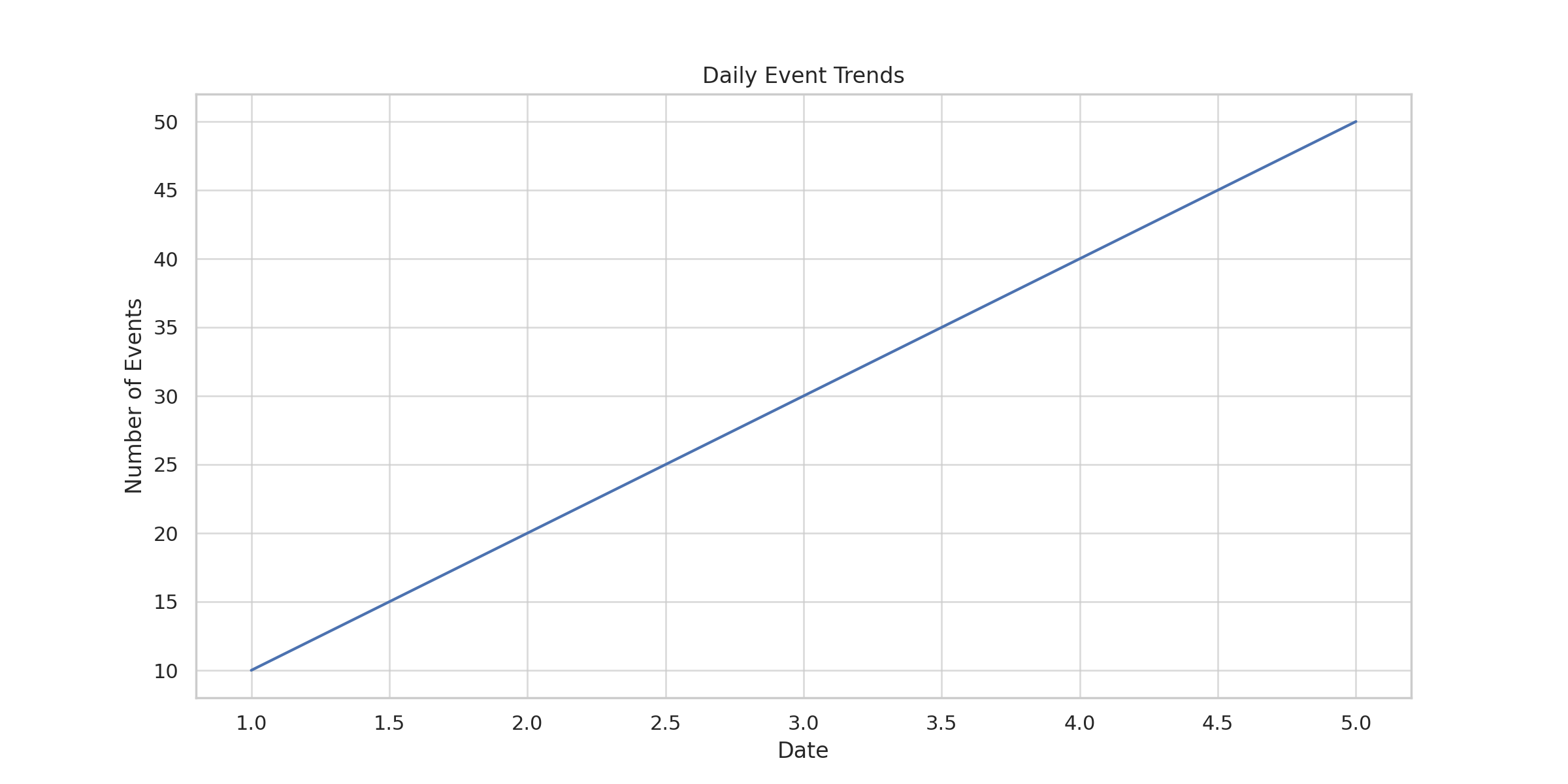
4.1 Loading CSV Data  
Ensure you have a CSV file containing user event data. The file should have columns like:  
- `user\_id`: Unique identifier for each user.  
- `event\_type`: Type of interaction/event by the user.  
- `event\_time`: Timestamp of the event.  
- `price`: Price associated with the event (if applicable).  
- `brand`: Brand associated with the event.  
- `category\_code`: Category related to the event.  
  
In the code, replace the file path with the correct path to your CSV file:  
```python  
file\_path = '/path/to/your/events.csv' Update this path  
data = pd.read\_csv(file\_path)  
```

# 5. Visualizations

Event Type Distribution (Bar Plot)



Daily Event Trends (Line Plot)



# 6. Model Evaluation and Insights

The model evaluation is conducted using the AUC score and other classification metrics. SHAP (Shapley Additive Explanations) values are used to interpret the model’s predictions, highlighting the most influential features for churn prediction.