Exploratory Data Analysis (EDA) is essential for understanding your data and uncovering insights. Here’s a structured approach: --- ### **\*1. Understand the Data\*** #### **\*a. Data Overview\*** - Load the data and inspect it:

print(messages\_df.head()) print(messages\_df.info()) print(messages\_df.describe()) print(messages\_df.isnull().sum())

- Check for: - Data types (object, int, float, datetime). - Missing values. - Basic statistics (mean, median, std, etc.). #### **\*b. Understand the Columns\*** - Review each column's purpose, units, and significance. - Classify columns: - **\*Categorical:\*** e.g., user\_id, campaign\_type. - **\*Numerical:\*** e.g., is\_clicked, purchase\_amount. - **\*Datetime:\*** e.g., date, sent\_at. --- ### **\*2. Explore Individual Columns\*** #### **\*a. Numerical Variables\*** - Plot distributions:

import matplotlib.pyplot as plt import seaborn as sns sns.histplot(messages\_df['is\_opened'], kde=True) plt.show()

- Check for: - Outliers. - Skewness. #### **\*b. Categorical Variables\*** - Check unique values and their frequencies:

print(messages\_df['campaign\_type'].value\_counts())

- Visualize:

sns.countplot(x='campaign\_type', data=messages\_df) plt.show()

#### **\*c. Datetime Variables\*** - Convert and extract parts:

messages\_df['year'] = messages\_df['date'].dt.year messages\_df['month'] = messages\_df['date'].dt.month

- Plot trends over time:

messages\_df.groupby('month').size().plot(kind='line') plt.show()

--- ### **\*3. Relationship Between Variables\*** #### **\*a. Numerical vs Numerical\*** - Scatter plots and correlations:

sns.scatterplot(x='is\_clicked', y='is\_purchased', data=messages\_df) sns.heatmap(messages\_df.corr(), annot=True) plt.show()

#### **\*b. Categorical vs Numerical\*** - Compare distributions:

sns.boxplot(x='campaign\_type', y='is\_clicked', data=messages\_df)

#### **\*c. Categorical vs Categorical\*** - Cross-tabulations:

pd.crosstab(messages\_df['campaign\_type'], messages\_df['is\_clicked'])

--- ### **\*4. Time-Based Analysis\*** #### **\*a. Analyze Trends\*** - Aggregate over time:

messages\_df.groupby('date')['is\_opened'].sum().plot(kind='line') plt.show()

#### **\*b. Seasonality\*** - Break down by time components (day of week, month, etc.). --- ### **\*5. Handle Missing Data\*** - Analyze missing patterns:

sns.heatmap(messages\_df.isnull(), cbar=False) plt.show()

- Options: - **\*Remove:\*** If rows/columns have too many missing values. - **\*Impute:\*** Replace with mean/median for numerical or mode for categorical. --- ### **\*6. Feature Engineering\*** - Create new features: - **\*Ratios:\*** Click-to-open rate. - **\*Durations:\*** Time between sent\_at and opened\_first\_time\_at. --- ### **\*7. Analyze Behavior\*** #### **\*a. User Segmentation\*** - Group by user\_id and aggregate behavior:

user\_behavior = messages\_df.groupby('user\_id').agg({ 'is\_opened': 'sum', 'is\_clicked': 'sum', 'is\_purchased': 'sum' })

#### **\*b. Funnel Analysis\*** - Conversion rates:

total\_sent = messages\_df['is\_opened'].count() total\_opened = messages\_df['is\_opened'].sum() total\_clicked = messages\_df['is\_clicked'].sum() conversion\_rate = (total\_clicked / total\_opened) \* 100 print(f"Conversion Rate: {conversion\_rate}%")

--- ### **\*8. Advanced Techniques\*** #### **\*a. Clustering\*** - Identify user groups based on behavior (see earlier clustering steps). #### **\*b. Correlation & Hypothesis Testing\*** - Determine statistical relationships:

from scipy.stats import chi2\_contingency, ttest\_ind

#### **\*c. Anomaly Detection\*** - Identify unusual behavior or values. --- ### **\*9. Visualization for Insights\*** #### **\*a. Use Pair Plots\*** - Examine relationships between multiple variables:

sns.pairplot(messages\_df[['is\_opened', 'is\_clicked', 'is\_purchased']])

#### **\*b. Highlight Key Metrics\*** - Use bar charts, pie charts, or stacked plots for proportions and counts. --- ### **\*10. Document Insights\*** - Summarize findings with key observations. - Identify actionable steps based on EDA. --- ### Example Insights: 1. High **\*click-to-open rates\*** for certain campaigns might indicate effective targeting. 2. Users who engage consistently over a long **\*engagement duration\*** are more likely to purchase. 3. Low-performing campaigns might have high **\*unsubscribe rates\*** or complaints. EDA involves iterating through these steps to refine insights and guide further analysis