*import seaborn as sns*

*import matplotlib.pyplot as plt*

*sns.histplot(df['rating'], bins=10, kde=True)*

*plt.title('Distribution of Movie Ratings')*

*plt.xlabel('Rating')*

*plt.ylabel('Frequency')*

*plt.show()*

*Copy code*

*# Boxplot for Rating Distribution*

*sns.boxplot(x=df['rating'])*

*plt.title('Boxplot of Movie Ratings')*

*plt.xlabel('Rating')*

*plt.show()*

*# Distribution of number of ratings per movie*

*movie\_rating\_count = df.groupby('movieId')['rating'].count()*

*sns.histplot(movie\_rating\_count, bins=50, kde=False)*

*plt.title('Distribution of Number of Ratings per Movie')*

*plt.xlabel('Number of Ratings')*

*plt.ylabel('Movie Count')*

*plt.show()*

*# Number of ratings per user*

*user\_rating\_count = df.groupby('userId')['rating'].count()*

*sns.histplot(user\_rating\_count, bins=50, kde=False)*

*plt.title('Distribution of Number of Ratings per User')*

*plt.xlabel('Number of Ratings')*

*plt.ylabel('User Count')*

*plt.show()*

*# Correlation Matrix*

*correlation\_matrix = df.corr()*

*sns.heatmap(correlation\_matrix, annot=True, cmap='coolwarm', fmt='.2f')*

*plt.title('Correlation Heatmap')*

*plt.show()*

*# Pairplot to check relationship between features*

*sns.pairplot(df[['rating', 'userId', 'movieId']])*

*plt.title('Pairplot of Ratings, User ID, and Movie ID')*

*plt.show()*

*# Scatterplot between Number of Ratings and Average Rating per User*

*avg\_user\_rating = df.groupby('userId')['rating'].mean()*

*user\_data = pd.DataFrame({'userId': user\_rating\_count.index, 'num\_ratings': user\_rating\_count.values, 'avg\_rating': avg\_user\_rating.values})*

*sns.scatterplot(x='num\_ratings', y='avg\_rating', data=user\_data)*

*plt.title('Number of Ratings vs. Average Rating per User')*

*plt.xlabel('Number of Ratings')*

*plt.ylabel('Average Rating')*

*plt.show()*

*# Ratings by Genre*

*genre\_avg\_rating = df.groupby('genre')['rating'].mean()*

*sns.barplot(x=genre\_avg\_rating.index, y=genre\_avg\_rating.values)*

*plt.title('Average Ratings by Genre')*

*plt.xlabel('Genre')*

*plt.ylabel('Average Rating')*

*plt.xticks(rotation=45)*

*plt.show()*