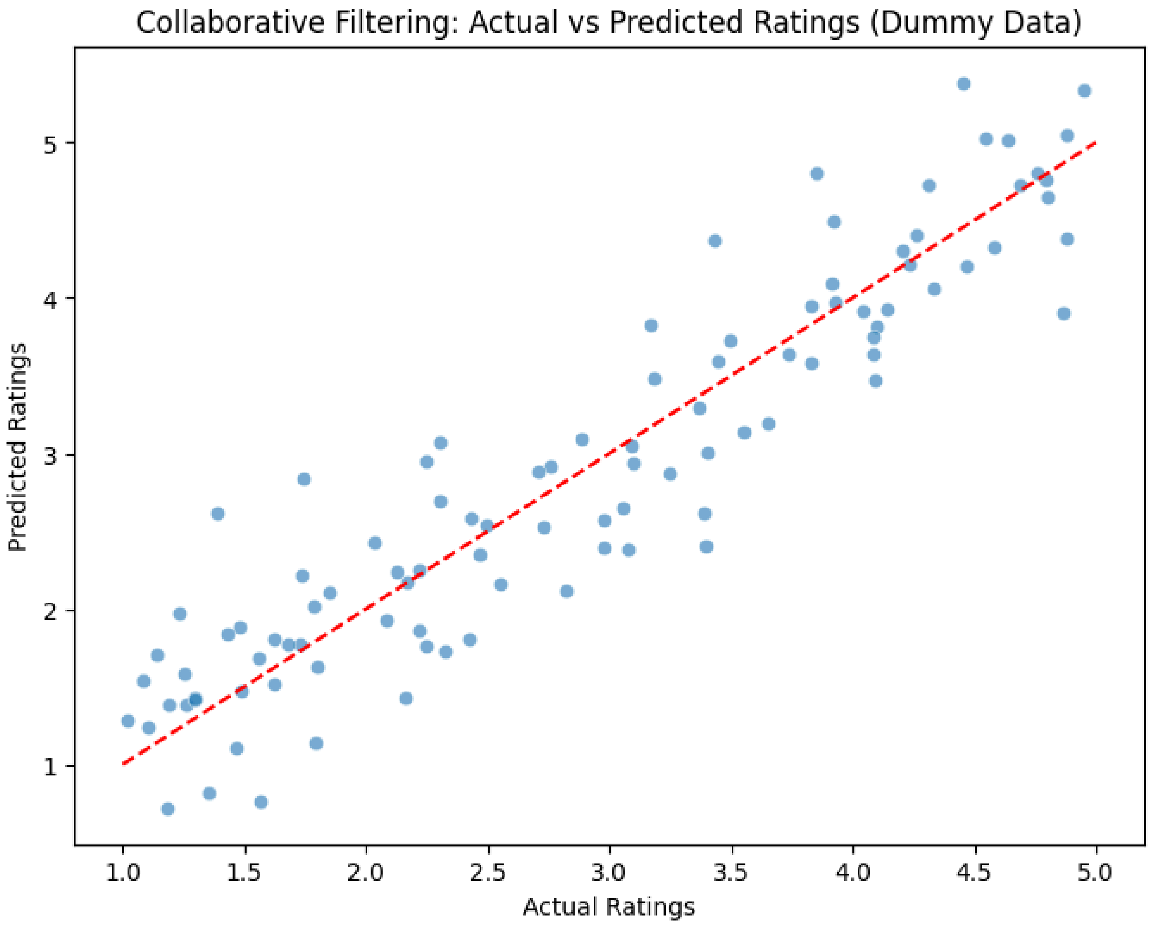
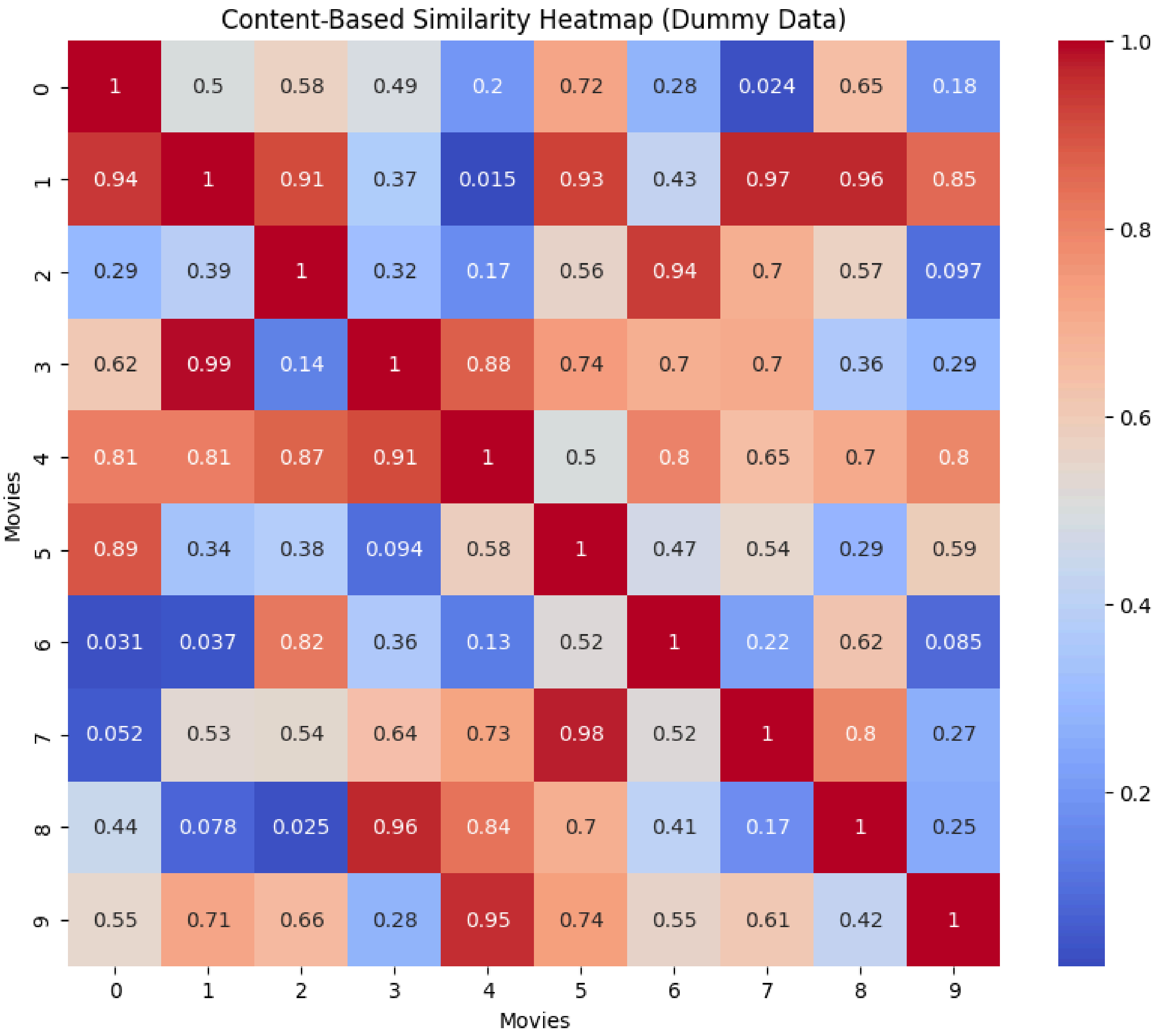
```
pip install pandas scikit-learn scikit-surprise
```

```
Requirement already satisfied: pandas in /usr/local/lib/python3.11/dist-packages (2.2.2)
Requirement already satisfied: scikit-learn in /usr/local/lib/python3.11/dist-packages (1.6.1)
Collecting scikit-surprise
  Downloading scikit_surprise-1.1.4.tar.gz (154 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 154.4/154.4 kB 2.6 MB/s eta 0:00:00
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Preparing metadata (pyproject.toml) ... done
Requirement already satisfied: numpy>=1.23.2 in /usr/local/lib/python3.11/dist-packages (from
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packag
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from p
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from
Requirement already satisfied: scipy>=1.6.0 in /usr/local/lib/python3.11/dist-packages (from s
Requirement already satisfied: joblib>=1.2.0 in /usr/local/lib/python3.11/dist-packages (from
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.11/dist-packages
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from pytho
Building wheels for collected packages: scikit-surprise
  Building wheel for scikit-surprise (pyproject.toml) ... done
  Created wheel for scikit-surprise: filename=scikit_surprise-1.1.4-cp311-cp311-linux_x86_64.w
  Stored in directory: /root/.cache/pip/wheels/2a/8f/6e/7e2899163e2d85d8266daab4aa1cdabec7a6c5
Successfully built scikit-surprise
Installing collected packages: scikit-surprise
Successfully installed scikit-surprise-1.1.4
```

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

# Dummy Data: Actual vs Predicted Ratings
np.random.seed(42)
actual_ratings = np.random.uniform(1, 5, 100)
predicted_ratings = actual_ratings + np.random.normal(0, 0.5, 100)

# Scatter Plot: Actual vs Predicted Ratings
plt.figure(figsize=(8, 6))
sns.scatterplot(x=actual_ratings, y=predicted_ratings, alpha=0.6)
plt.plot([1, 5], [1, 5], color='red', linestyle='--')
plt.xlabel('Actual Ratings')
plt.ylabel('Predicted Ratings')
plt.title('Collaborative Filtering: Actual vs Predicted Ratings (Dummy Data)')
plt.show()
```

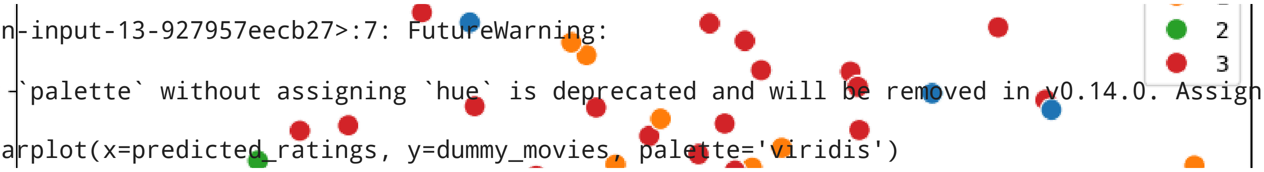


```
# Dummy Similarity Matrix (e.g., genres similarity)
similarity_matrix = np.random.rand(10, 10)
np.fill_diagonal(similarity_matrix, 1)  # Perfect similarity with self

# Heatmap of Similarities
plt.figure(figsize=(10, 8))
sns.heatmap(similarity_matrix, annot=True, cmap='coolwarm', cbar=True)
plt.title('Content-Based Similarity Heatmap (Dummy Data)')
plt.xlabel('Movies')
plt.ylabel('Movies')
plt.show()
```


Content-Based Similarity Heatmap (Dummy Data)
Movies
Movies
0 1 2 3 4 5 6 7 8 9
0: 1 0.5 0.58 0.49 0.2 0.72 0.28 0.024 0.65 0.18
1: 0.94 1 0.91 0.37 0.015 0.93 0.43 0.97 0.96 0.85
2: 0.29 0.39 1 0.32 0.17 0.56 0.94 0.7 0.57 0.097
3: 0.62 0.99 0.14 1 0.88 0.74 0.7 0.7 0.36 0.29
4: 0.81 0.81 0.87 0.91 1 0.5 0.8 0.65 0.7 0.8
5: 0.89 0.34 0.38 0.094 0.58 1 0.47 0.54 0.29 0.59
6: 0.031 0.037 0.82 0.36 0.13 0.52 1 0.22 0.62 0.085
7: 0.052 0.53 0.54 0.64 0.73 0.98 0.52 1 0.8 0.27
8: 0.44 0.078 0.025 0.96 0.84 0.7 0.41 0.17 1 0.25
9: 0.55 0.71 0.66 0.28 0.95 0.74 0.55 0.61 0.42 1
1.0
0.8
0.6
0.4
0.2

```
from sklearn.cluster import KMeans
from sklearn.decomposition import PCA

# Dummy User-Item Ratings Matrix (100 users, 20 movies)
user_item_matrix = np.random.rand(100, 20)

# KMeans Clustering
kmeans = KMeans(n_clusters=4, random_state=42)
clusters = kmeans.fit_predict(user_item_matrix)

# Reduce to 2D with PCA for Visualization
pca = PCA(n_components=2)
pca_result = pca.fit_transform(user_item_matrix)

# Scatter Plot of Clusters
plt.figure(figsize=(10, 7))
```

```
# Dummy Recommended Movies and Predicted Ratings
dummy_movies = [f"Movie {i}" for i in range(1, 11)]
predicted_ratings = np.random.uniform(3.0, 5.0, 10)

# Bar Plot of Recommendations
plt.figure(figsize=(10, 6))
sns.barplot(x=predicted_ratings, y=dummy_movies, palette='viridis')
plt.xlabel('Predicted Rating')
plt.ylabel('Recommended Movies')
plt.title('Hybrid Recommendation Results (Dummy Data)')
plt.show()
```



```
<ipython-input-13-927957eecb27>:7: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in v0.14.0. Assign

  sns.barplot(x=predicted_ratings, y=dummy_movies, palette='viridis')
```