

Practical No. 02

Aim : To compute similarity and dissimilarity measures for a given dataset using Euclidean distance and Jaccard similarity.

Software Used : IDLE

Theory :

Dataset: File name (customers.csv)

id,age,income,gender

1,25,50000,Male

2,30,60000,Female

3,22,45000,Male

4,35,80000,Female

5,28,52000,Male

6,40,70000,Female

7,50,90000,Male

8,45,85000,Female

9,33,62000,Male

10,27,48000,Female

Code :

```
import pandas as pd
```

```
from sklearn.metrics import pairwise_distances
```

```
from sklearn.preprocessing import LabelEncoder
```

```
import numpy as np
```

```
# Load dataset
```

```
df = pd.read_csv('customers.csv')
```

```
# Convert categorical data to numerical
```

```
le = LabelEncoder()
```

```
df['gender'] = le.fit_transform(df['gender']) # Male: 1, Female: 0
```

```
# Compute Euclidean distance
```

```
numeric_data = df[['age', 'income', 'gender']].values
```

```
euclidean_distances = pairwise_distances(numeric_data, metric='euclidean')
```

```
print("Euclidean Distances:")
```

```
print(euclidean_distances)
```

```
# Compute Jaccard similarity for categorical data (gender)
```

```
gender_data = df[['gender']].values
```

```
jaccard_similarity = 1 - pairwise_distances(gender_data, metric='jaccard')
```

```
print("Jaccard Similarity:")
```

```
print(jaccard_similarity)
```

```
# Dissimilarity matrix (1 - similarity)
```

```
jaccard_dissimilarity = 1 - jaccard_similarity
```

```
print("Jaccard Dissimilarity:")
```

```
print(jaccard_dissimilarity)
```

Result :

```
Python 3.11.4 (tags/v3.11.4:d2340ef, Jun 7 2023, 05:45:37) [MSC v.1934 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\User\Desktop\Data-Warehousing\Mining\FracNo2.py
Euclidean Distances:
[[ 0.          10000.0013  5000.0009  30000.00160333
  2000.00225  20000.00565  40000.0078125  35000.00572857
 12000.00266667  2000.00125 ]
 [10000.0013  0.          15000.00216667  20000.000625
 8000.000125  10000.005  30000.00668333  25000.0045
 2000.0025  12000.000375 ]
 [ 5000.0009  15000.00216667  0.          35000.00242857
 7000.00257143  25000.0065  45000.00071111  40000.006625
 17000.00355882  3000.00433333 ]
 [30000.00160333  20000.000625  35000.00242857  0.
 28000.0009286  10000.00125  10000.01129999  5000.00999999
 18000.00013889  32000.001 ]
 [ 2000.00125  8000.000125  7000.00257143  28000.0009286
 18000.00402778  38000.00636842  33000.00435354
 0.
 10000.00125  4000.00025 ]
 [20000.00565  10000.005  25000.0065  10000.00125
 18000.00402778  0.          20000.002525  15000.00083333
 8000.003125  22000.00384091 ]
 [40000.0078125  30000.00668333  45000.00071111  10000.01129999
 38000.00636842  20000.002525  0.          5000.00026
 28000.0009286  42000.00630952 ]
 [35000.00572857  25000.0045  40000.006625  5000.00999999
 33000.00435354  15000.00083333  5000.00026  0.
 23000.00315217  37000.00437898 ]
 [12000.00266667  2000.0025  17000.00355882  18000.00013889
 18000.00125  8000.000125  28000.00316071  23000.00315217
 0.          14000.00132143 ]
 [ 2000.00125  12000.000375  3000.00433333  32000.001
 4000.0025  22000.00384091  42000.00630952  37000.00437898
 14000.00132143  0.          ]]
```

```
Warning (from warnings module):
  File "D:\python 3.11\Lib\site-packages\sklearn\metrics\pairwise.py", line 2361
    warnings.warn(msg, DataConversionWarning)
DataConversionWarning: Data was converted to boolean for metric jaccard
Jaccard Similarity:
[[[ 0.  0.  1.  0.  1.  0.  1.  0. ]
 [ 0.  1.  0.  1.  0.  1.  0.  1. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 0.  1.  0.  1.  0.  1.  0.  1. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 0.  1.  0.  1.  0.  1.  0.  1. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 0.  1.  0.  1.  0.  1.  0.  1. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 0.  1.  0.  1.  0.  1.  0.  1. ]]]
Jaccard Dissimilarity:
[[[ 0.  1.  0.  1.  0.  1.  0.  1. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]
 [ 1.  0.  1.  0.  1.  0.  1.  0. ]]]]
```

Conclusion : In this practical , We have performed to compute similarity and dissimilarity measures for a given dataset using Euclidean distance and Jaccard similarity.