test-copy

June 14, 2023

[36]:
```
import pandas as pd
from sklearn.cluster import KMeans
```

[8]:
```
title_basics = pd.read_csv('title_basics_2018.csv')
title_basics
```

[8]:
```
          tconst                 primaryTitle                originalTitle  \
0      tt0069049   The Other Side of the Wind   The Other Side of the Wind
1      tt0111414                  A Thin Life                  A Thin Life
2      tt0170651           T.G.M. - osvoboditel           T.G.M. - osvoboditel
3      tt0192528                Heaven & Hell              Reverse Heaven
4      tt0253093                 Gangavataran                 Gangavataran
…            …                          …                            …
12810  tt9908960                      Pliusas                      Pliusas
12811  tt9909086            Pheriaa Come Back            Pheriaa Come Back
12812  tt9909650                     Hellbiro                     Hellbiro
12813  tt9914644  9/11: Escape from the Towers  9/11: Escape from the Towers
12814  tt9916132  The Mystery of a Buryat Lama  The Mystery of a Buryat Lama

       year  runtimeMinutes                        genres
0      2018             122                         Drama
1      2018              75                        Comedy
2      2018              60                   Documentary
3      2018             104                         Drama
4      2018             134                            \N
…       …               …                            …
12810  2018              90                        Comedy
12811  2018             137                         Drama
12812  2018              95                        Comedy
12813  2018             120                   Documentary
12814  2018              94  Biography,Documentary,History

[12815 rows x 6 columns]
```

[9]:
```
title_basics.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 12815 entries, 0 to 12814
```

```
Data columns (total 6 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   tconst          12815 non-null  object
 1   primaryTitle    12815 non-null  object
 2   originalTitle   12815 non-null  object
 3   year            12815 non-null  int64
 4   runtimeMinutes  12815 non-null  int64
 5   genres          12815 non-null  object
dtypes: int64(2), object(4)
memory usage: 600.8+ KB
```

[23]:
```
title_basics.isna().sum()
```

[23]:
```
tconst            0
primaryTitle      0
originalTitle     0
year              0
runtimeMinutes    0
genres            0
dtype: int64
```

[10]:
```
title_ratings = pd.read_csv('title_ratings.csv')
title_ratings
```

[10]:
```
            tconst  averageRating  numVotes
0        tt0000001            5.6      1543
1        tt0000002            6.1       186
2        tt0000003            6.5      1201
3        tt0000004            6.2       114
4        tt0000005            6.1      1921
…              …              …        …
985454   tt9916576            5.9         7
985455   tt9916578            9.1        11
985456   tt9916720            5.1        41
985457   tt9916766            6.7        11
985458   tt9916778            6.9        16

 [985459 rows x 3 columns]
```

[11]:
```
title_ratings.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 985459 entries, 0 to 985458
Data columns (total 3 columns):
 #   Column         Non-Null Count   Dtype
---  ------         --------------   -----
 0   tconst         985459 non-null  object
 1   averageRating  985459 non-null  float64
 2   numVotes       985459 non-null  int64
dtypes: float64(1), int64(1), object(1)
memory usage: 22.6+ MB
```

[24]:
```
title_ratings.isna().sum()
```

[24]:
```
tconst           0
averageRating    0
numVotes         0
dtype: int64
```

[7]:
```
title_basics_ratings = pd.merge(title_basics, title_ratings, on='tconst')
title_basics_ratings
```

[7]:
```
         tconst                  primaryTitle                 originalTitle  \
0     tt0069049    The Other Side of the Wind    The Other Side of the Wind
1     tt0170651          T.G.M. - osvoboditel          T.G.M. - osvoboditel
2     tt0192528                 Heaven & Hell                Reverse Heaven
3     tt0253093                  Gangavataran                  Gangavataran
4     tt0262759      Seven Jews from My Class  Siedmiu Zydów z mojej klasy
…           …                           …                            …
7224  tt9903952   BADMEN with a good behavior   BADMEN with a good behavior
7225  tt9904014               Lost in Klessin               Lost in Klessin
7226  tt9904530                Scream Returns                Scream Returns
7227  tt9908960                       Pliusas                       Pliusas
7228  tt9914644  9/11: Escape from the Towers  9/11: Escape from the Towers

      year  runtimeMinutes           genres  averageRating  numVotes
0     2018             122            Drama            6.9      4937
1     2018              60      Documentary            7.5         6
2     2018             104            Drama            3.9        74
3     2018             134               \N            6.6         8
4     2018              40      Documentary            7.0         6
…    …                …              …             …       …
7224  2018              87    Comedy,Horror            7.8         6
7225  2018              90              War            7.5        14
7226  2018              48  Horror,Thriller            3.0         6
7227  2018              90           Comedy            4.1        18
7228  2018             120      Documentary            8.4        24

[7229 rows x 8 columns]
```

[25]:
```
#1.        According to the provided dataset, how many 2018 films were␣
↪categorized as a Comedy?
```

```
title_comedy_2018 = title_basics_ratings[(title_basics_ratings['year'] == 2018)␣
 ↪& (title_basics_ratings['genres'] == 'Comedy')]
title_comedy_2018

#Answer: 566 films
```

[25]:
```
         tconst                        primaryTitle  \
12    tt0432010  The Queen of Sheba Meets the Atom Man
32   tt10112464                    Frank & Fearless
36   tt10131904                     Vse ili nichego
47   tt10178280                          Mangoshake
65   tt10399736                               Fagot
…           …                                   …
7162  tt9652322                         Chief Daddy
7191  tt9798310                        #Odindenleta
7199  tt9837502     Merry Men: The Real Yoruba Demons
7215  tt9870612                  Randy Writes a Novel
7227  tt9908960                             Pliusas

                              originalTitle  year  runtimeMinutes  genres  \
12    The Queen of Sheba Meets the Atom Man  2018             110  Comedy
32                         Frank & Fearless  2018              97  Comedy
36                          Vse ili nichego  2018              85  Comedy
47                               Mangoshake  2018             104  Comedy
65                                    Fagot  2018              61  Comedy
…                                        …    …               …     …
7162                            Chief Daddy  2018              99  Comedy
7191                           #Odindenleta  2018             100  Comedy
7199      Merry Men: The Real Yoruba Demons  2018             106  Comedy
7215                   Randy Writes a Novel  2018              70  Comedy
7227                                Pliusas  2018              90  Comedy

      averageRating  numVotes
12              7.1        48
32              4.8        18
36              5.2        23
47              5.4         5
65              4.7         7
…               …       …
7162            4.8       133
7191            4.7        10
7199            5.2        56
7215            8.7        53
7227            4.1        18

[566 rows x 8 columns]
```

```
[28]: #2.        According to the provided dataset, how many 2018 films got a score␣
      ↪of 8.0 or higher?  (Note that this will require joining the two datasets␣
      ↪together)

      title_score_2018 = title_basics_ratings[(title_basics_ratings['year'] == 2018)␣
      ↪& (title_basics_ratings['averageRating'] >= 8.0)]
      title_score_2018

      #Answer: 780 films
```

```
[28]:        tconst                              primaryTitle  \
      13     tt0825334            Caravaggio and My Mother the Pope
      18    tt10005184                                     Lysis
      27    tt10062150                         Yücel'in Çiçekleri
      29    tt10078502            Stars in the Sky: A Hunting Story
      45    tt10176328  Exteriores: Mulheres Brasileiras na Diplomacia
      …            …                                        …
      7193   tt9805820                                     Caisa
      7206   tt9856680              Puffs: Filmed Live Off Broadway
      7214   tt9869952                         The First Company
      7215   tt9870612                       Randy Writes a Novel
      7228   tt9914644                  9/11: Escape from the Towers

                                          originalTitle  year  runtimeMinutes  \
      13               Caravaggio and My Mother the Pope  2018              90
      18                                           Lysis  2018              98
      27                              Yücel'in Çiçekleri  2018              69
      29               Stars in the Sky: A Hunting Story  2018              75
      45    Exteriores: Mulheres Brasileiras na Diplomacia  2018              52
      …                                             …   …               …
      7193                                         Caisa  2018              84
      7206               Puffs: Filmed Live Off Broadway  2018             118
      7214                             The First Company  2018             100
      7215                          Randy Writes a Novel  2018              70
      7228                  9/11: Escape from the Towers  2018             120

                        genres  averageRating  numVotes
      13          Comedy,Drama            8.8        53
      18             Adventure            8.1        15
      27    Documentary,History            8.4        65
      29           Documentary            9.3        19
      45           Documentary           10.0         5
      …                    …             …       …
      7193         Documentary            8.0        30
      7206    Adventure,Comedy            8.7        15
      7214         Documentary            8.6         7
      7215              Comedy            8.7        53
```

```
7228          Documentary             8.4        24

[780 rows x 8 columns]
```

[35]:
```
#3.         What was the best film of 2018?

max_votes = title_basics_ratings['numVotes'].max()
best_film_2018 = title_basics_ratings[(title_basics_ratings['numVotes']==␣
  ↪max_votes)]
best_film_2018

#Answer: Avengers: Infinity War 2018
```

[35]:
```
        tconst            primaryTitle           originalTitle  year  \
556  tt4154756  Avengers: Infinity War  Avengers: Infinity War  2018

     runtimeMinutes                    genres  averageRating  numVotes
556             149  Action,Adventure,Sci-Fi            8.5    719146
```

[37]:
```
#4.         Do audiences prefer longer films, or shorter films?  You may choose␣
  ↪to simply outline your methodology to approach this problem

df_test = title_basics_ratings[['runtimeMinutes', 'numVotes']]

kmeans = KMeans(n_clusters=2)
kmeans.fit(df_test)

labels = kmeans.labels_

df_test['Cluster'] = labels
df_test
```

```
C:\Users\Renzo\AppData\Local\Packages\PythonSoftwareFoundation.Python.3.11_qbz5n
2kfra8p0\LocalCache\local-packages\Python311\site-
packages\sklearn\cluster\_kmeans.py:870: FutureWarning: The default value of
`n_init` will change from 10 to 'auto' in 1.4. Set the value of `n_init`
explicitly to suppress the warning
  warnings.warn(
C:\Users\Renzo\AppData\Local\Temp\ipykernel_11372\1286142752.py:10:
SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  df_test['Cluster'] = labels
```

```
[37]:       runtimeMinutes  numVotes  Cluster
      0                122      4937        0
      1                 60         6        0
      2                104        74        0
      3                134         8        0
      4                 40         6        0
      …                  …         …        …
      7224              87         6        0
      7225              90        14        0
      7226              48         6        0
      7227              90        18        0
      7228             120        24        0

      [7229 rows x 3 columns]
```

```
[44]: cluster0 = df_test[df_test['Cluster'] == 0]
      cluster0
```

```
[44]:       runtimeMinutes  numVotes  Cluster
      0                122      4937        0
      1                 60         6        0
      2                104        74        0
      3                134         8        0
      4                 40         6        0
      …                  …         …        …
      7224              87         6        0
      7225              90        14        0
      7226              48         6        0
      7227              90        18        0
      7228             120        24        0

      [7204 rows x 3 columns]
```

```
[51]: mean_time_cluster0 = round(cluster0['runtimeMinutes'].mean(),2)
      mean_rating_cluster0 = round(cluster0['numVotes'].mean(),2)
      print("Average time in cluster 0 ",mean_time_cluster0,"minutes and average␣
       ↪Votes in cluster 0",mean_rating_cluster0)
```

```
      Average time in cluster 0  96.17 minutes and average Votes in cluster 0 1627.12
```

```
[43]: cluster1 = df_test[df_test['Cluster'] == 1]
      cluster1
```

```
[43]:       runtimeMinutes  numVotes  Cluster
      103              112    309088        1
      114              119    170924        1
      119              143    306154        1
```

```
123              136    280715        1
131              140    318183        1
135              134    388400        1
145              134    547427        1
254              100    172662        1
259              124    235791        1
266              115    242396        1
274              140    144671        1
418              118    217516        1
466              135    239985        1
546              134    183146        1
556              149    719146        1
715              117    269769        1
822              128    231666        1
835              147    250124        1
925              118    264136        1
951              110    154836        1
1096             119    420985        1
2153              90    329157        1
2559             130    252993        1
3121             135    166589        1
3917             127    177004        1
```

[53]:
```
mean_time_cluster1 = round(cluster1['runtimeMinutes'].mean(),2)
mean_rating_cluster1 = round(cluster1['numVotes'].mean(),2)
print("Average time in cluster 1 ",mean_time_cluster1,"minutes and average␣
 ↪Votes in cluster 1",mean_rating_cluster1)
```

```
Average time in cluster 1  126.16 minutes and average Votes in cluster 1
279738.52
```

[ ]:
```
#Conclusion:

#Although there are many more films in cluster 0, their average duration in␣
 ↪relation to the number of votes,
# where the latter indicates the audience's preferences, is lower than the␣
 ↪other cluster which is formed by
# fewer films but with a higher average number of votes. This concludes that␣
 ↪the audience prefers longer movies.
```