IMDB_Movies-Data Analysis

October 23, 2024

[1]:
```
import pandas as pd
```

[2]:
```
movies = pd.read_csv(r"C:\Users\91983\Downloads\archive\movie.csv", sep=',')
print(type(movies))
movies.head(20)
```

```
<class 'pandas.core.frame.DataFrame'>
```

[2]:

	movieId	title	genres
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	2	Jumanji (1995)	Adventure|Children|Fantasy
2	3	Grumpier Old Men (1995)	Comedy|Romance
3	4	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	5	Father of the Bride Part II (1995)	Comedy
5	6	Heat (1995)	Action|Crime|Thriller
6	7	Sabrina (1995)	Comedy|Romance
7	8	Tom and Huck (1995)	
8	9	Sudden Death (1995)	
9	10	GoldenEye (1995)	
10	11	American President, The (1995)	
11	12	Dracula: Dead and Loving It (1995)	
12	13	Balto (1995)	
13	14	Nixon (1995)	
14	15	Cutthroat Island (1995)	
15	16	Casino (1995)	
16	17	Sense and Sensibility (1995)	
17	18	Four Rooms (1995)	
18	19	Ace Ventura: When Nature Calls (1995)	
19	20	Money Train (1995)	

```
7                            Adventure|Children
8                                        Action
9                     Action|Adventure|Thriller
10                         Comedy|Drama|Romance
11                               Comedy|Horror
12                Adventure|Animation|Children
13                                       Drama
14                    Action|Adventure|Romance
15                              Crime|Drama
16                             Drama|Romance
17                                     Comedy
18                                     Comedy
19           Action|Comedy|Crime|Drama|Thriller
```

[3]:
```
tags = pd.read_csv(r"C:\Users\91983\Downloads\archive\tag.csv", sep=',')
tags.head()
```

[3]:

	userId	movieId	tag	timestamp
0	18	4141	Mark Waters	2009-04-24 18:19:40
1	65	208	dark hero	2013-05-10 01:41:18
2	65	353	dark hero	2013-05-10 01:41:19
3	65	521	noir thriller	2013-05-10 01:39:43
4	65	592	dark hero	2013-05-10 01:41:18

[6]:
```
ratings = pd.read_csv(r"C:\Users\91983\Downloads\archive\rating.csv", sep=',',␣
↪parse_dates=['timestamp'])
ratings.head()
```

[6]:

	userId	movieId	rating	timestamp
0	1	2	3.5	2005-04-02 23:53:47
1	1	29	3.5	2005-04-02 23:31:16
2	1	32	3.5	2005-04-02 23:33:39
3	1	47	3.5	2005-04-02 23:32:07
4	1	50	3.5	2005-04-02 23:29:40

[7]:
```
del ratings['timestamp']
del tags['timestamp']
```

1 Data Structures:

Series

[8]:
```
row_0 = tags.iloc[0]
type(row_0)
```

[8]:
```
pandas.core.series.Series
```

[9]:
```
print(row_0)
```

```
userId              18
movieId           4141
tag        Mark Waters
Name: 0, dtype: object
```

[10]:
```
row_0.index
```

[10]:
```
Index(['userId', 'movieId', 'tag'], dtype='object')
```

[11]:
```
row_0['userId']
```

[11]:
```
18
```

[12]:
```
'rating' in row_0
```

[12]:
```
False
```

[13]:
```
row_0.name
```

[13]:
```
0
```

[14]:
```
row_0 = row_0.rename('firstRow')
row_0.name
```

[14]:
```
'firstRow'
```

2 DataFrames

[15]:
```
tags.head()
```

[15]:

	userId	movieId	tag
0	18	4141	Mark Waters
1	65	208	dark hero
2	65	353	dark hero
3	65	521	noir thriller
4	65	592	dark hero

[16]:
```
tags.index
```

[16]:
```
RangeIndex(start=0, stop=465564, step=1)
```

[17]:
```
tags.columns
```

[17]:
```
Index(['userId', 'movieId', 'tag'], dtype='object')
```

[18]:
```
tags.iloc[ [0,11,500] ]
```

[18]:

	userId	movieId	tag
0	18	4141	Mark Waters
11	65	1783	noir thriller
500	342	55908	entirely dialogue

3 Descriptive Statistics¶

3.1 Let's look how the ratings are distributed!

[19]:
```
ratings['rating'].describe()
```

[19]:
```
count    2.000026e+07
mean     3.525529e+00
std      1.051989e+00
min      5.000000e-01
25%      3.000000e+00
50%      3.500000e+00
75%      4.000000e+00
max      5.000000e+00
Name: rating, dtype: float64
```

[20]:
```
ratings.describe()
```

[20]:

	userId	movieId	rating
count	2.000026e+07	2.000026e+07	2.000026e+07
mean	6.904587e+04	9.041567e+03	3.525529e+00
std	4.003863e+04	1.978948e+04	1.051989e+00
min	1.000000e+00	1.000000e+00	5.000000e-01
25%	3.439500e+04	9.020000e+02	3.000000e+00
50%	6.914100e+04	2.167000e+03	3.500000e+00
75%	1.036370e+05	4.770000e+03	4.000000e+00
max	1.384930e+05	1.312620e+05	5.000000e+00

[21]:
```
ratings['rating'].mean()
```

[21]:
```
3.5255285642993797
```

[22]:
```
ratings.mean()
```

[22]:
```
userId     69045.872583
movieId     9041.567330
rating         3.525529
dtype: float64
```

[23]:
```
ratings['rating'].min()
```

[23]:
```
0.5
```

[24]:
```
ratings['rating'].max()
```

[24]:
```
5.0
```

[25]:
```
ratings['rating'].std()
```

[25]:
```
1.051988919275684
```

[26]:
```
ratings['rating'].mode()
```

[26]:
```
0    4.0
Name: rating, dtype: float64
```

[27]:
```
ratings.corr()
```

[27]:
```
            userId   movieId    rating
userId    1.000000 -0.000850  0.001175
movieId  -0.000850  1.000000  0.002606
rating    0.001175  0.002606  1.000000
```

[28]:
```
filter1 = ratings['rating'] > 10
print(filter1)
filter1.any()
```

```
0           False
1           False
2           False
3           False
4           False
            …
20000258    False
20000259    False
20000260    False
20000261    False
20000262    False
Name: rating, Length: 20000263, dtype: bool
```

[28]:
```
False
```

[29]:
```
filter2 = ratings['rating'] > 0
filter2.all()
```

[29]:
```
True
```

4 Data Cleaning: Handling Missing Data

[30]:
```
movies.shape
```

[30]:
```
(27278, 3)
```

[31]:
```
movies.isnull().any().any()
```

[31]:
```
False
```

[32]:
```
ratings.shape
```

[32]:
```
(20000263, 3)
```

[33]:
```
ratings.isnull().any().any()
```

[33]:
```
False
```

[34]:
```
tags.shape
```

[34]:
```
(465564, 3)
```

[35]:
```
tags.isnull().any().any()
```

[35]:
```
True
```

[36]:
```
tags=tags.dropna()
```

[37]:
```
tags.isnull().any().any()
```

[37]:
```
False
```

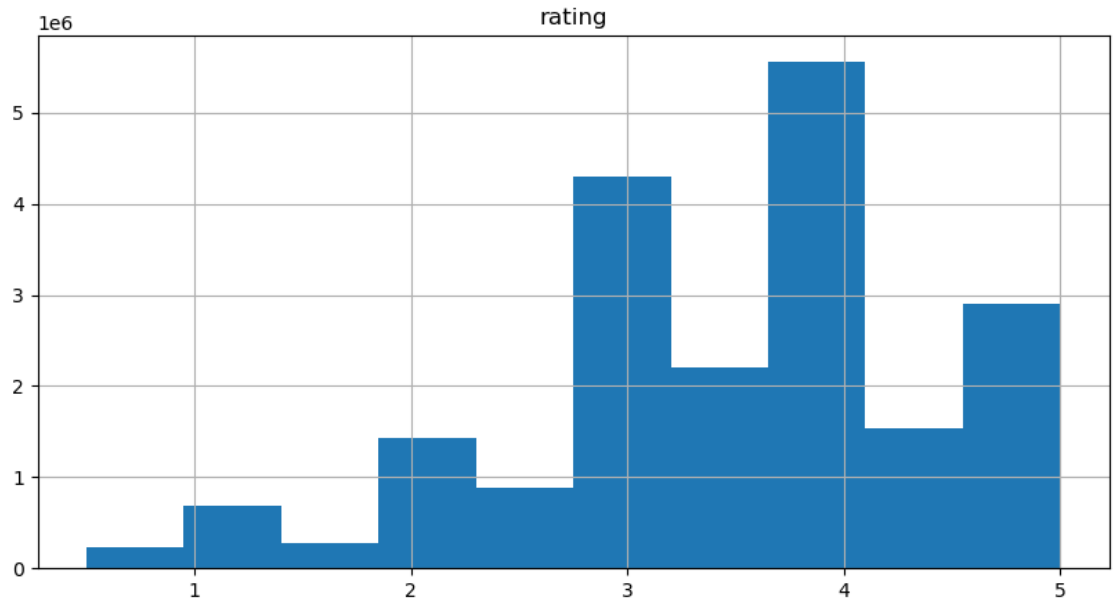
[38]:
```
tags.shape
```

[38]:
```
(465548, 3)
```

5 Data Visualization

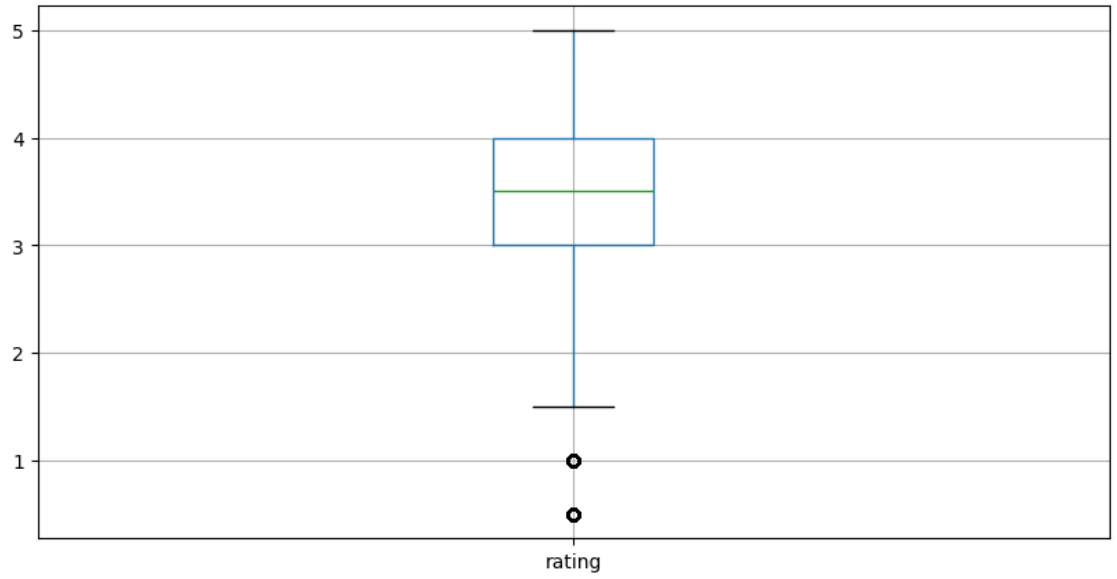
[39]:
```
%matplotlib inline

ratings.hist(column='rating', figsize=(10,5))
```

[39]:
```
array([[<Axes: title={'center': 'rating'}>]], dtype=object)
```



[40]:
```
ratings.boxplot(column='rating', figsize=(10,5))
```

[40]: <Axes: >



[41]:
```
tags['tag'].head()
```

[41]:
```
0      Mark Waters
1        dark hero
2        dark hero
3    noir thriller
4        dark hero
Name: tag, dtype: object
```

[42]:
```
movies[['title','genres']].head()
```

[42]:

	title	genres
0	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy
1	Jumanji (1995)	Adventure|Children|Fantasy
2	Grumpier Old Men (1995)	Comedy|Romance
3	Waiting to Exhale (1995)	Comedy|Drama|Romance
4	Father of the Bride Part II (1995)	Comedy

[43]:
```
ratings[-10:]
```

[43]:

	userId	movieId	rating
20000253	138493	60816	4.5
20000254	138493	61160	4.0
20000255	138493	65682	4.5
20000256	138493	66762	4.5
20000257	138493	68319	4.5
20000258	138493	68954	4.5
20000259	138493	69526	4.5
20000260	138493	69644	3.0
20000261	138493	70286	5.0
20000262	138493	71619	2.5

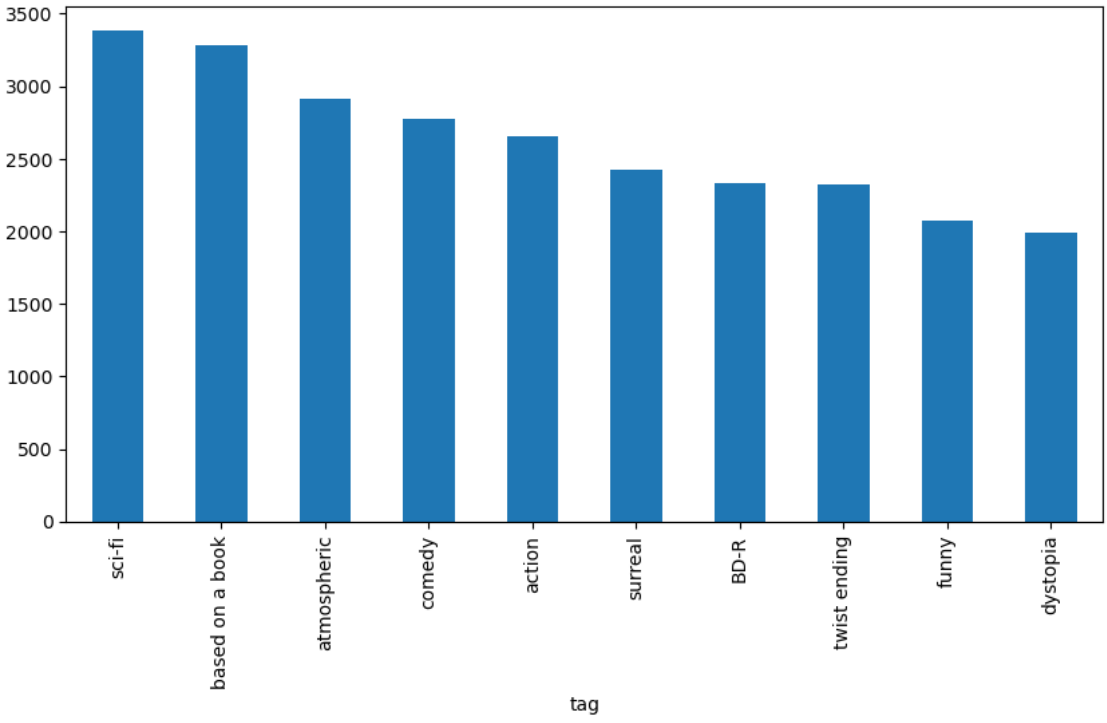
[44]:
```
tag_counts = tags['tag'].value_counts()
tag_counts[-10:]
```

[44]:
```
tag
missing child          1
Ron Moore              1
Citizen Kane           1
mullet                 1
biker gang             1
Paul Adelstein         1
the wig                1
```

```
killer fish                         1
genetically modified monsters       1
topless scene                       1
Name: count, dtype: int64
```

[45]:
```
tag_counts[:10].plot(kind='bar', figsize=(10,5))
```

[45]:
```
<Axes: xlabel='tag'>
```



6 Filters for Selecting Rows

[46]:
```
is_highly_rated = ratings['rating'] >= 5.0
ratings[is_highly_rated][30:50]
```

[46]:

	userId	movieId	rating
239	3	50	5.0
242	3	175	5.0
244	3	223	5.0
245	3	260	5.0
246	3	316	5.0
247	3	318	5.0
248	3	329	5.0
252	3	457	5.0

```
253       3      480     5.0
254       3      490     5.0
256       3      541     5.0
258       3      593     5.0
263       3      858     5.0
264       3      904     5.0
267       3      924     5.0
268       3      953     5.0
271       3     1060     5.0
272       3     1073     5.0
275       3     1084     5.0
276       3     1089     5.0
```

[47]:
```
is_action= movies['genres'].str.contains('Action')
movies[is_action][5:15]
```

[47]:
```
    movieId                                      title  \
22       23                          Assassins (1995)
41       42                     Dead Presidents (1995)
43       44                       Mortal Kombat (1995)
50       51                      Guardian Angel (1994)
65       66  Lawnmower Man 2: Beyond Cyberspace (1996)
69       70                  From Dusk Till Dawn (1996)
70       71                            Fair Game (1995)
75       76                            Screamers (1995)
77       78                  Crossing Guard, The (1995)
85       86                         White Squall (1996)

                          genres
22          Action|Crime|Thriller
41             Action|Crime|Drama
43       Action|Adventure|Fantasy
50          Action|Drama|Thriller
65         Action|Sci-Fi|Thriller
69  Action|Comedy|Horror|Thriller
70                          Action
75         Action|Sci-Fi|Thriller
77    Action|Crime|Drama|Thriller
85         Action|Adventure|Drama
```

[48]:
```
movies[is_action].head(15)
```

[48]:
```
    movieId                                      title  \
5        6                               Heat (1995)
8        9                      Sudden Death (1995)
9       10                         GoldenEye (1995)
14      15                   Cutthroat Island (1995)
```

```
19       20                          Money Train (1995)
22       23                            Assassins (1995)
41       42                        Dead Presidents (1995)
43       44                          Mortal Kombat (1995)
50       51                         Guardian Angel (1994)
65       66  Lawnmower Man 2: Beyond Cyberspace (1996)
69       70                  From Dusk Till Dawn (1996)
70       71                            Fair Game (1995)
75       76                            Screamers (1995)
77       78                  Crossing Guard, The (1995)
85       86                         White Squall (1996)

                                genres
5                 Action|Crime|Thriller
8                               Action
9             Action|Adventure|Thriller
14             Action|Adventure|Romance
19  Action|Comedy|Crime|Drama|Thriller
22                Action|Crime|Thriller
41                   Action|Crime|Drama
43             Action|Adventure|Fantasy
50                Action|Drama|Thriller
65               Action|Sci-Fi|Thriller
69       Action|Comedy|Horror|Thriller
70                              Action
75               Action|Sci-Fi|Thriller
77         Action|Crime|Drama|Thriller
85               Action|Adventure|Drama
```

7 Group By and Aggregate

[49]:
```
ratings_count = ratings[['movieId','rating']].groupby('rating').count()
ratings_count
```

[49]:
```
        movieId
rating
0.5      239125
1.0      680732
1.5      279252
2.0     1430997
2.5      883398
3.0     4291193
3.5     2200156
4.0     5561926
4.5     1534824
5.0     2898660
```

[50]:
```
average_rating = ratings[['movieId','rating']].groupby('movieId').mean()
average_rating.head()
```

[50]:
```
           rating
movieId
1        3.921240
2        3.211977
3        3.151040
4        2.861393
5        3.064592
```

[51]:
```
movie_count = ratings[['movieId','rating']].groupby('movieId').count()
movie_count.head()
```

[51]:
```
         rating
movieId
1         49695
2         22243
3         12735
4          2756
5         12161
```

[52]:
```
movie_count = ratings[['movieId','rating']].groupby('movieId').count()
movie_count.tail()
```

[52]:
```
         rating
movieId
131254        1
131256        1
131258        1
131260        1
131262        1
```

8 Merge Dataframes

[53]:
```
tags.head()
```

[53]:
```
   userId  movieId           tag
0      18     4141    Mark Waters
1      65      208      dark hero
2      65      353      dark hero
3      65      521  noir thriller
4      65      592      dark hero
```

[54]:
```
movies.head()
```

[54]:
```
   movieId                               title  \
0        1                    Toy Story (1995)
1        2                      Jumanji (1995)
2        3             Grumpier Old Men (1995)
3        4            Waiting to Exhale (1995)
4        5  Father of the Bride Part II (1995)

                                        genres
0  Adventure|Animation|Children|Comedy|Fantasy
1                   Adventure|Children|Fantasy
2                               Comedy|Romance
3                         Comedy|Drama|Romance
4                                       Comedy
```

[55]:
```
t = movies.merge(tags, on='movieId', how='inner')
t.head()
```

[55]:
```
   movieId            title                                       genres  \
0        1  Toy Story (1995)  Adventure|Animation|Children|Comedy|Fantasy
1        1  Toy Story (1995)  Adventure|Animation|Children|Comedy|Fantasy
2        1  Toy Story (1995)  Adventure|Animation|Children|Comedy|Fantasy
3        1  Toy Story (1995)  Adventure|Animation|Children|Comedy|Fantasy
4        1  Toy Story (1995)  Adventure|Animation|Children|Comedy|Fantasy

   userId                                     tag
0    1644                                 Watched
1    1741                    computer animation
2    1741                Disney animated feature
3    1741                         Pixar animation
4    1741  TÃ©a Leoni does not star in this movie
```

9 Combine aggreagation, merging, and filters to get useful analytics

[56]:
```
avg_ratings= ratings.groupby('movieId', as_index=False).mean()
del avg_ratings['userId']
avg_ratings.head()
```

[56]:
```
   movieId    rating
0        1  3.921240
1        2  3.211977
2        3  3.151040
3        4  2.861393
4        5  3.064592
```

[57]:
```
box_office = movies.merge(avg_ratings, on='movieId', how='inner')
box_office.tail()
```

[57]:

	movieId	title	genres	rating
26739	131254	Kein Bund für's Leben (2007)	Comedy	4.0
26740	131256	Feuer, Eis & Dosenbier (2002)	Comedy	4.0
26741	131258	The Pirates (2014)	Adventure	2.5
26742	131260	Rentun Ruusu (2001)	(no genres listed)	3.0
26743	131262	Innocence (2014)	Adventure|Fantasy|Horror	4.0

[58]:
```
is_highly_rated = box_office['rating'] >= 4.0
box_office[is_highly_rated][-5:]
```

[58]:

	movieId	title	genres	rating
26737	131250	No More School (2000)	Comedy	4.0
26738	131252	Forklift Driver Klaus: The First Day on the Jo…	Comedy|Horror	4.0
26739	131254	Kein Bund für's Leben (2007)	Comedy	4.0
26740	131256	Feuer, Eis & Dosenbier (2002)	Comedy	4.0
26743	131262	Innocence (2014)	Adventure|Fantasy|Horror	4.0

[59]:
```
is_Adventure = box_office['genres'].str.contains('Adventure')
box_office[is_Adventure][:5]
```

[59]:

	movieId	title	genres	rating
0	1	Toy Story (1995)	Adventure|Animation|Children|Comedy|Fantasy	3.921240
1	2	Jumanji (1995)	Adventure|Children|Fantasy	3.211977
7	8	Tom and Huck (1995)	Adventure|Children	
9	10	GoldenEye (1995)	Action|Adventure|Thriller	
12	13	Balto (1995)	Adventure|Animation|Children	

```
7   3.142049
9   3.430029
12  3.272416
```

[60]:
```
box_office[is_Adventure & is_highly_rated][-5:]
```

[60]:
```
       movieId                                              title  \
26611   130586                Itinerary of a Spoiled Child (1988)
26655   130996                           The Beautiful Story (1992)
26667   131050  Stargate SG-1 Children of the Gods - Final Cut…
26736   131248                                 Brother Bear 2 (2006)
26743   131262                                      Innocence (2014)

                                        genres  rating
26611                          Adventure|Drama     4.5
26655                   Adventure|Drama|Fantasy     5.0
26667                 Adventure|Sci-Fi|Thriller     5.0
26736  Adventure|Animation|Children|Comedy|Fantasy     4.0
26743                 Adventure|Fantasy|Horror     4.0
```

10 Vectorized String Operations

[61]:
```
movies.head()
```

[61]:
```
   movieId                               title  \
0        1                    Toy Story (1995)
1        2                      Jumanji (1995)
2        3             Grumpier Old Men (1995)
3        4            Waiting to Exhale (1995)
4        5  Father of the Bride Part II (1995)

                                        genres
0  Adventure|Animation|Children|Comedy|Fantasy
1                   Adventure|Children|Fantasy
2                               Comedy|Romance
3                         Comedy|Drama|Romance
4                                       Comedy
```

11 Split 'genres' into multiple columns

[63]:
```
movie_genres = movies['genres'].str.split('|', expand=True)
```

[64]:
```
movie_genres[:10]
```

[64]:
```
           0          1         2       3        4     5     6     7     8  \
0  Adventure  Animation  Children  Comedy  Fantasy  None  None  None  None
```

```
1  Adventure   Children   Fantasy    None     None  None  None  None  None
2     Comedy    Romance      None    None     None  None  None  None  None
3     Comedy      Drama   Romance    None     None  None  None  None  None
4     Comedy       None      None    None     None  None  None  None  None
5     Action      Crime  Thriller    None     None  None  None  None  None
6     Comedy    Romance      None    None     None  None  None  None  None
7  Adventure   Children      None    None     None  None  None  None  None
8     Action       None      None    None     None  None  None  None  None
9     Action  Adventure  Thriller    None     None  None  None  None  None

      9
0  None
1  None
2  None
3  None
4  None
5  None
6  None
7  None
8  None
9  None
```

12 Add a new column for comedy genre flag

[65]:
```
movie_genres['isComedy'] = movies['genres'].str.contains('Comedy')
```

[66]:
```
movie_genres[:10]
```

[66]:
```
           0          1         2       3        4     5     6     7     8  \
0  Adventure  Animation  Children  Comedy  Fantasy  None  None  None  None
1  Adventure   Children   Fantasy    None     None  None  None  None  None
2     Comedy    Romance      None    None     None  None  None  None  None
3     Comedy      Drama   Romance    None     None  None  None  None  None
4     Comedy       None      None    None     None  None  None  None  None
5     Action      Crime  Thriller    None     None  None  None  None  None
6     Comedy    Romance      None    None     None  None  None  None  None
7  Adventure   Children      None    None     None  None  None  None  None
8     Action       None      None    None     None  None  None  None  None
9     Action  Adventure  Thriller    None     None  None  None  None  None

      9  isComedy
0  None      True
1  None     False
2  None      True
3  None      True
4  None      True
```

```
5  None     False
6  None      True
7  None     False
8  None     False
9  None     False
```

13 Extract year from title e.g. (2007)

[68]:
```
movies['year'] = movies['title'].str.extract(r'.*\((.*)\).*', expand=True)
```

[69]:
```
movies.tail()
```

[69]:

	movieId	title	genres	year
27273	131254	Kein Bund für's Leben (2007)	Comedy	2007
27274	131256	Feuer, Eis & Dosenbier (2002)	Comedy	2002
27275	131258	The Pirates (2014)	Adventure	2014
27276	131260	Rentun Ruusu (2001)	(no genres listed)	2001
27277	131262	Innocence (2014)	Adventure|Fantasy|Horror	2014

14 Average Movie Ratings over Time¶

Movie ratings related to the year of launch?

[70]:
```
average_rating = ratings[['movieId','rating']].groupby('movieId',␣
 ↪as_index=False).mean()
average_rating.tail()
```

[70]:

	movieId	rating
26739	131254	4.0
26740	131256	4.0
26741	131258	2.5
26742	131260	3.0
26743	131262	4.0

[ ]:
```

```