Analysing Effect of tags on movie ratings

Akanchha Choudhary

This is a mini project for week 6 of course python for data science from micromaster of Ucsandiego in data science. We were given this practice project to prepare us for next two big project in this course. I am supposed to perform following steps for this mini project that has been mentioned below:

Step 1: Select a dataset we’ve already seen

Step 2: Continue to explore the dataset(s)

Step 3: Identify one research question

Step 4: Use appropriate methods to explore your data.

Step 5: Present your findings

Step 6: Present your work!

Dataset(s)

As a first step of this project. I am asked to choose a one of three dataset that has already been discussed in this course.I am choosing Movielens dataset which is a IMBD dataset and make analysis of movies domain.
The dataset is available for download here - https://grouplens.org/datasets/movielens/20m/

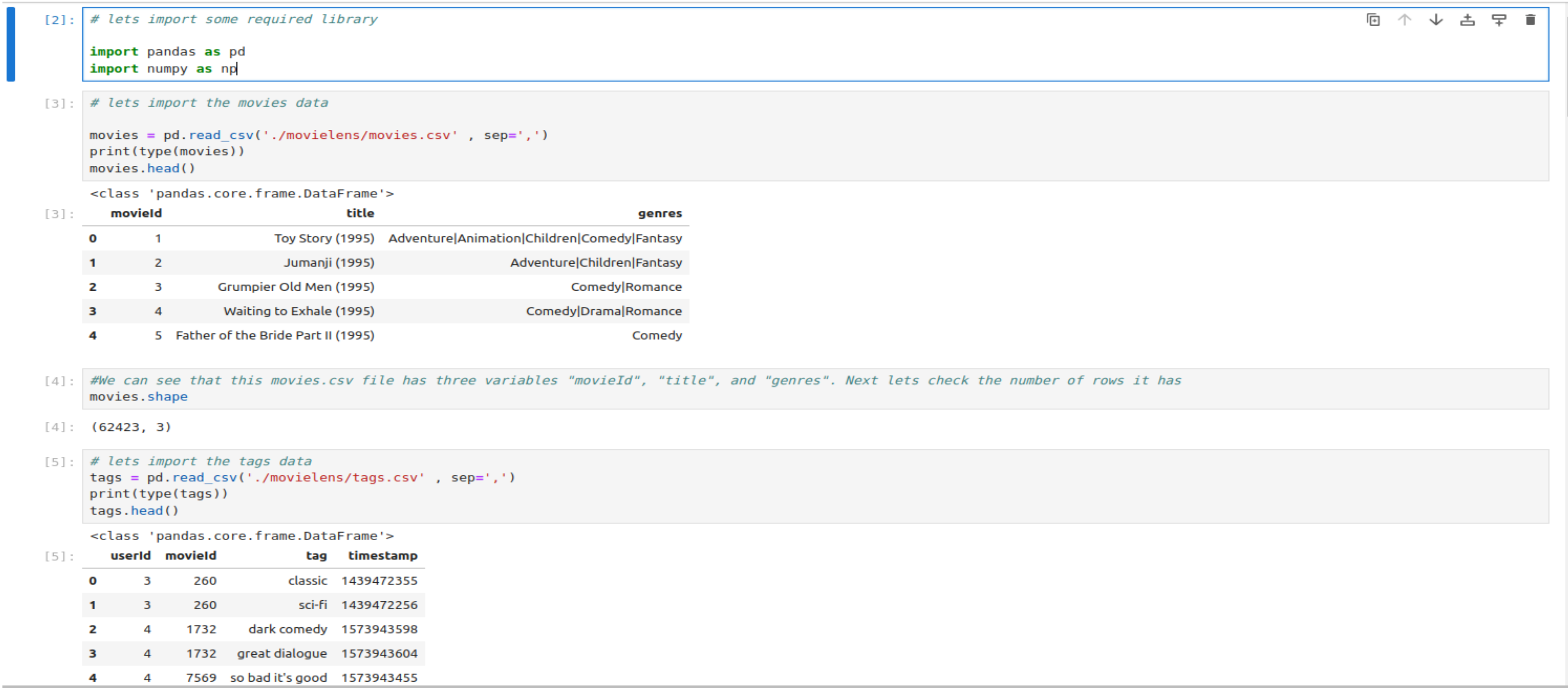
Description about the dataset, as shown on the website is below:

This dataset (ml-20m) describes 5-star rating and free-text tagging activity from MovieLens, a movie recommendation service. It contains 20000263 ratings and 465564 tag applications across 27278 movies. These data were created by 138493 users between January 09, 1995 and March 31, 2015. This dataset was generated on October 17, 2016.

Users were selected at random for inclusion. All selected users had rated at least 20 movies. No demographic information is included. Each user is represented by an id, and no other information is provided.

The data are contained in six files, genome-scores.csv, genome-tags.csv, links.csv, movies.csv, ratings.csv and tags.csv. More details about the contents and use of all these files follows.

Lets perform some analysis . I have loaded movies.csv, ratings.csv and tags.csv and did some initial analysis with command shown below.



```
[2]: # lets import some required library

     import pandas as pd
     import numpy as np
```

```
[3]: # lets import the movies data

     movies = pd.read_csv('./movielens/movies.csv' , sep=',')
     print(type(movies))
     movies.head()
```

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

[3]:

	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

```
[4]: #We can see that this movies.csv file has three variables "movieId", "title", and "genres". Next lets check the number of rows it has
     movies.shape
```

```
[4]: (62423, 3)
```

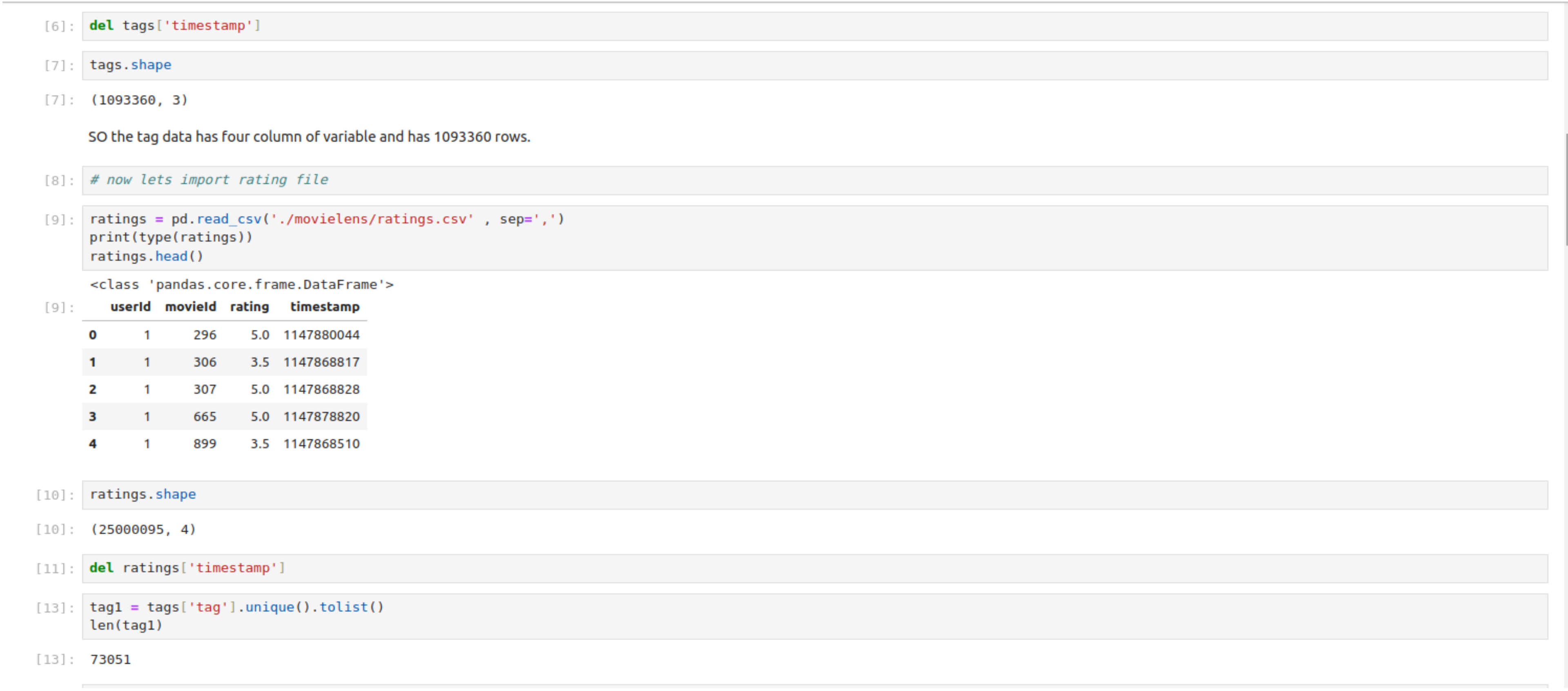
```
[5]: # lets import the tags data
     tags = pd.read_csv('./movielens/tags.csv' , sep=',')
     print(type(tags))
     tags.head()
```

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

[5]:

	userId	movieId	tag	timestamp
0	3	260	classic	1439472355
1	3	260	sci-fi	1439472256
2	4	1732	dark comedy	1573943598
3	4	1732	great dialogue	1573943604
4	4	7569	so bad it's good	1573943455

Lets perform some analysis . I have loaded movies.csv, ratings.csv and tags.csv and did some initial analysis with command shown below.



```
[6]: del tags['timestamp']
```

```
[7]: tags.shape
```

```
[7]: (1093360, 3)
```

SO the tag data has four column of variable and has 1093360 rows.

```
[8]: # now lets import rating file
```

```
[9]: ratings = pd.read_csv('./movielens/ratings.csv' , sep=',')
     print(type(ratings))
     ratings.head()
```

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

[9]:

	userId	movieId	rating	timestamp
0	1	296	5.0	1147880044
1	1	306	3.5	1147868817
2	1	307	5.0	1147868828
3	1	665	5.0	1147878820
4	1	899	3.5	1147868510

```
[10]: ratings.shape
```

```
[10]: (25000095, 4)
```

```
[11]: del ratings['timestamp']
```

```
[13]: tag1 = tags['tag'].unique().tolist()
      len(tag1)
```

```
[13]: 73051
```

Research Question

Based on the above exploratory commands, I believe that the following questions can be answered using the dataset for example

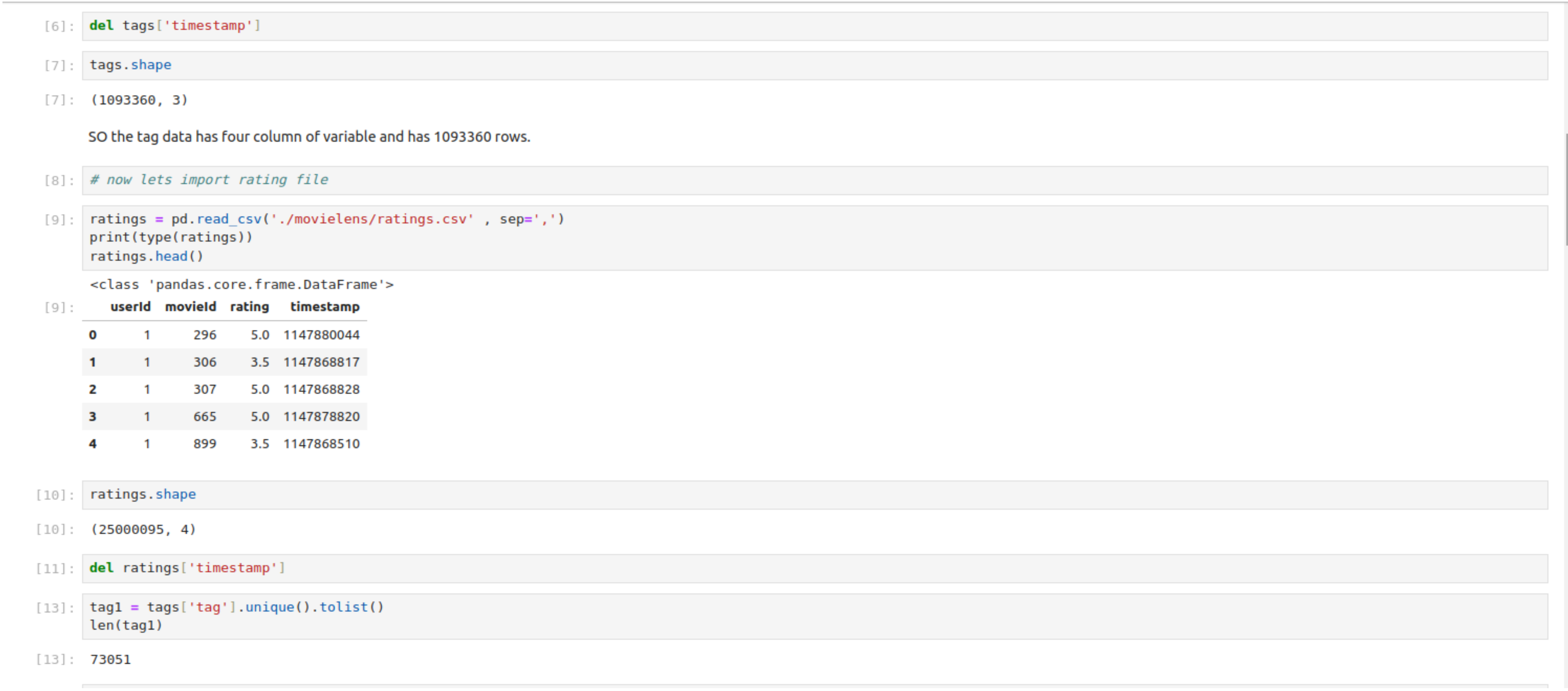
1.highly rated movie by year

2.Is there any correlation between rating and frequency of tagging.

3. Most watch genres of all time.

For the analysis I will go with 3rd research question

Performing more analysis using pandas and matplotlib to answer the research question



```
[6]: del tags['timestamp']
```

```
[7]: tags.shape
```

```
[7]: (1093360, 3)
```

SO the tag data has four column of variable and has 1093360 rows.

```
[8]: # now lets import rating file
```

```
[9]: ratings = pd.read_csv('./movielens/ratings.csv' , sep=',')
     print(type(ratings))
     ratings.head()
```

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

[9]:

	userId	movieId	rating	timestamp
0	1	296	5.0	1147880044
1	1	306	3.5	1147868817
2	1	307	5.0	1147868828
3	1	665	5.0	1147878820
4	1	899	3.5	1147868510

```
[10]: ratings.shape
```

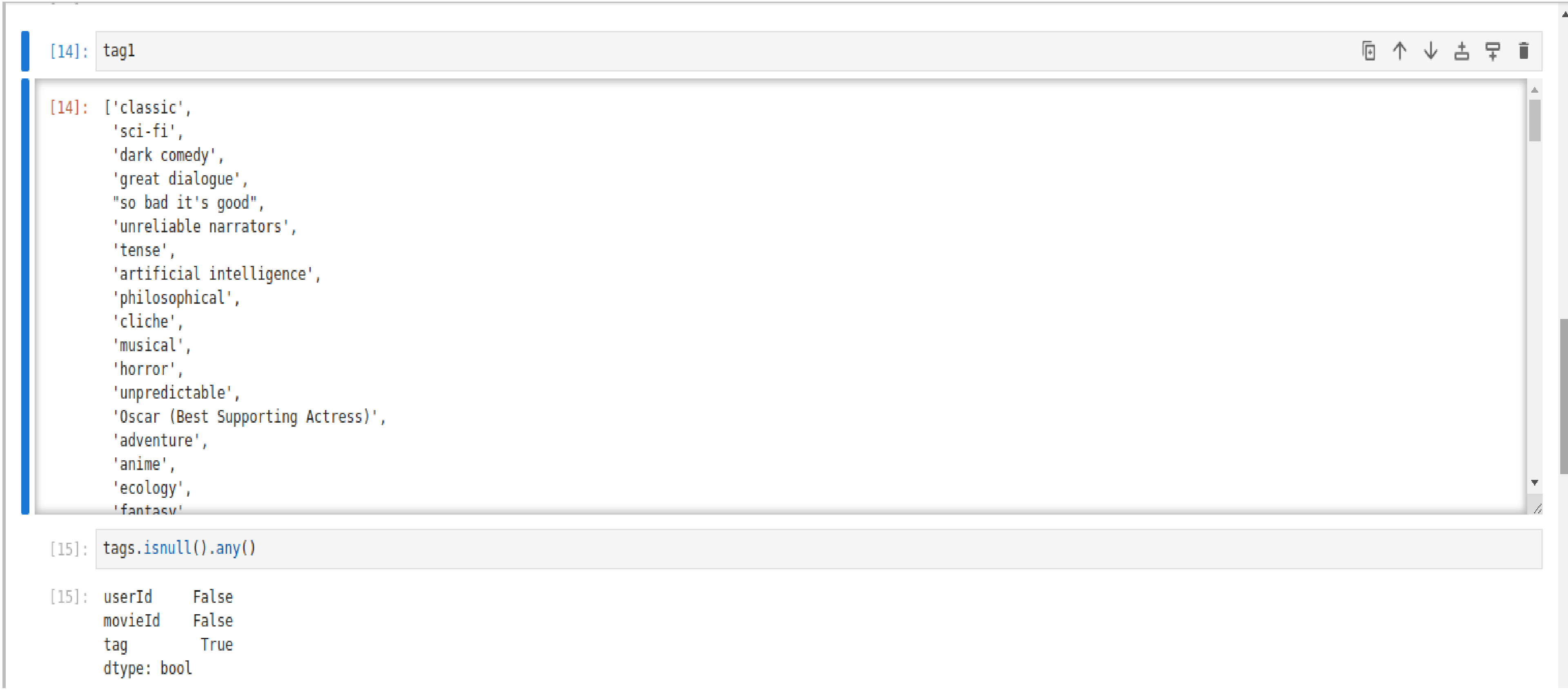
```
[10]: (25000095, 4)
```

```
[11]: del ratings['timestamp']
```

```
[13]: tag1 = tags['tag'].unique().tolist()
      len(tag1)
```

```
[13]: 73051
```

Performing more analysis using pandas and matplotlib to answer the research question



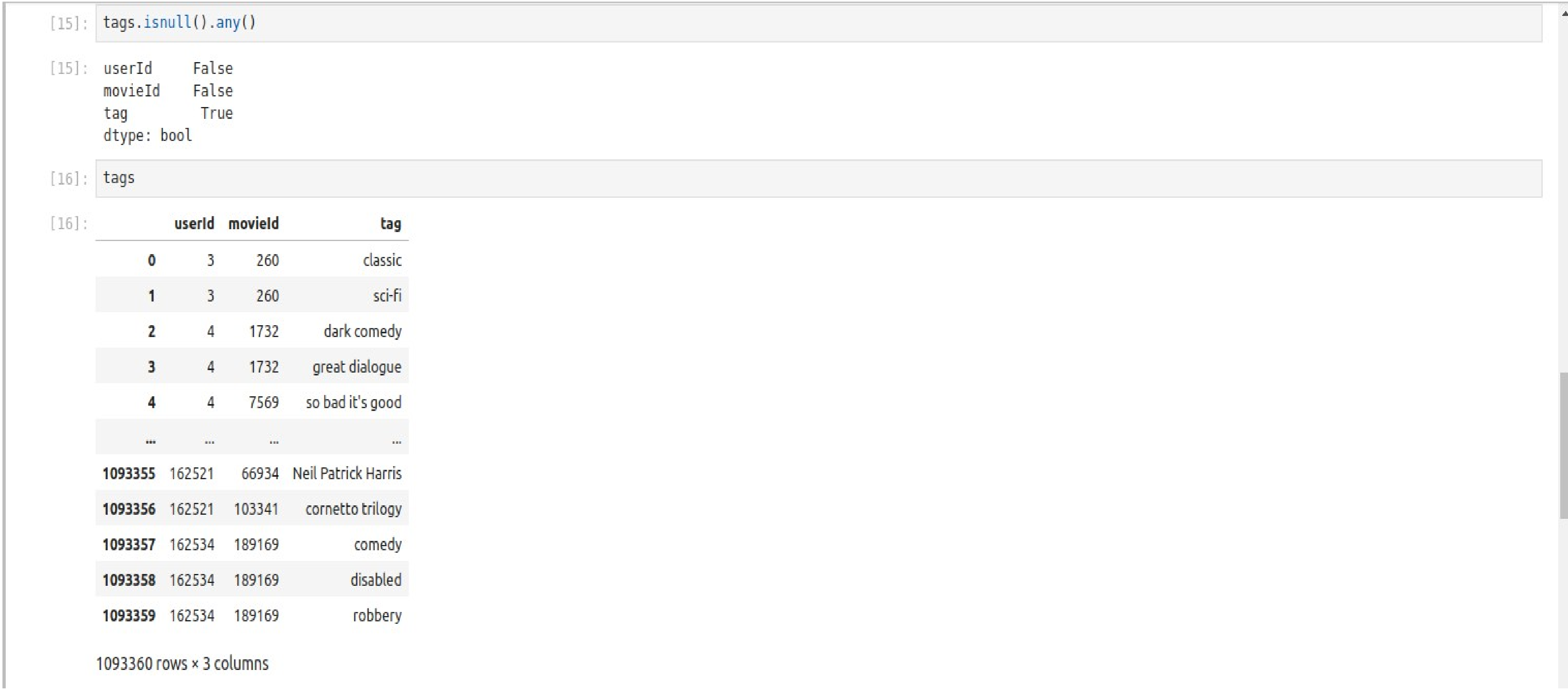
```
[14]: tag1
```

```
[14]: ['classic',
       'sci-fi',
       'dark comedy',
       'great dialogue',
       "so bad it's good",
       'unreliable narrators',
       'tense',
       'artificial intelligence',
       'philosophical',
       'cliche',
       'musical',
       'horror',
       'unpredictable',
       'Oscar (Best Supporting Actress)',
       'adventure',
       'anime',
       'ecology',
       'fantasy'
```

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

```
[15]: userId     False
      movieId    False
      tag         True
      dtype: bool
```

Performing more analysis using pandas and matplotlib to answer the research question



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

```
[15]: userId     False
      movieId    False
      tag         True
      dtype: bool
```

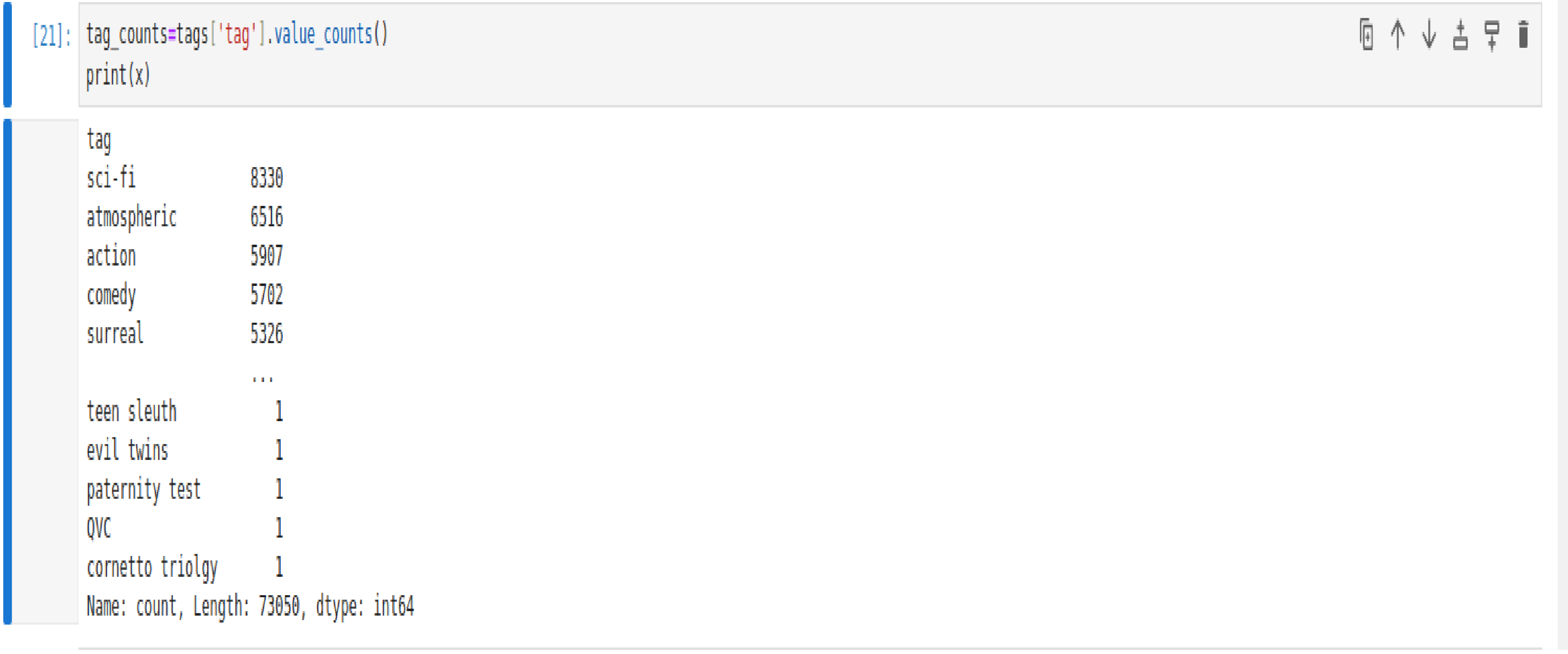
```
[16]: tags
```

[16]:

	userId	movieId	tag
0	3	260	classic
1	3	260	sci-fi
2	4	1732	dark comedy
3	4	1732	great dialogue
4	4	7569	so bad it's good
...	...	...	...
1093355	162521	66934	Neil Patrick Harris
1093356	162521	103341	cornetto trilogy
1093357	162534	189169	comedy
1093358	162534	189169	disabled
1093359	162534	189169	robbery

1093360 rows × 3 columns

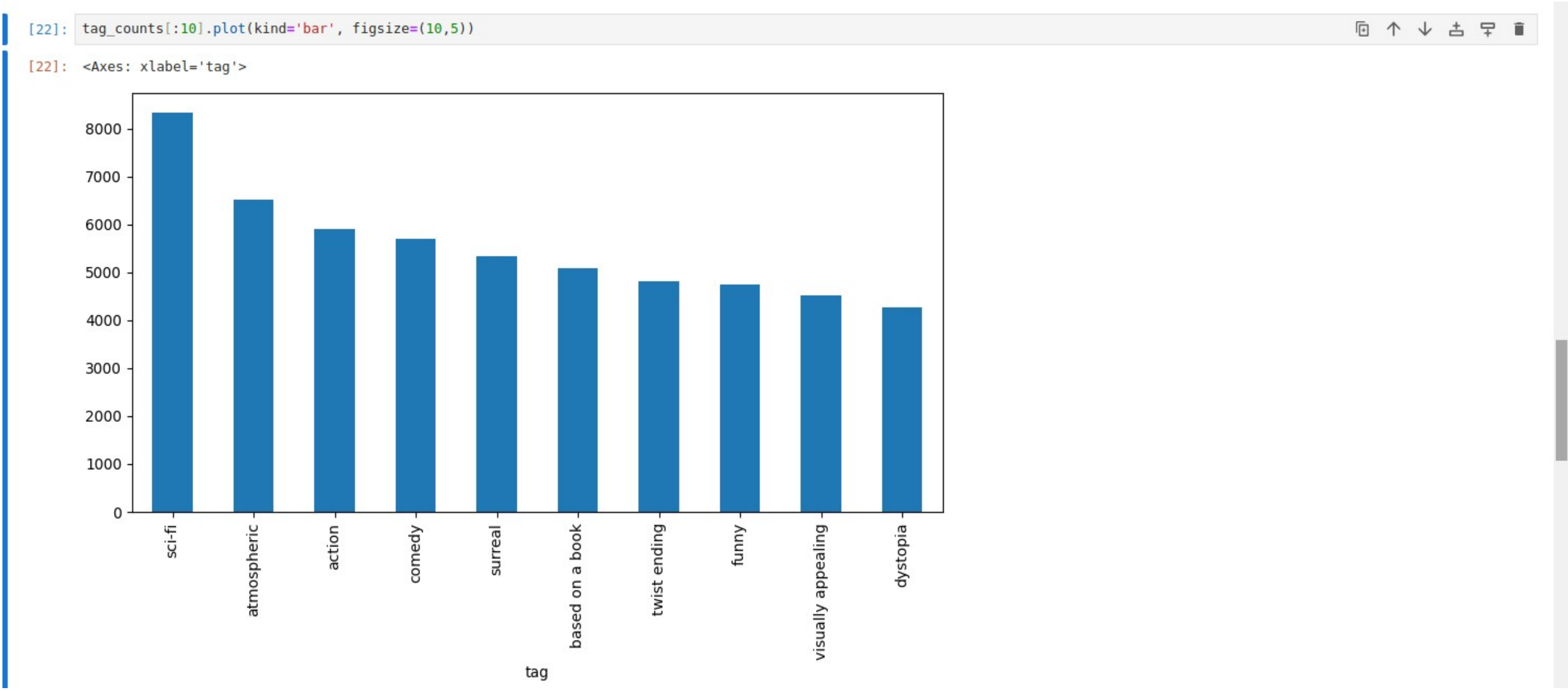
Performing more analysis using pandas and matplotlib to answer the research question



```
tag_counts=tags['tag'].value_counts()
print(x)
```

```
tag
sci-fi            8330
atmospheric       6516
action            5907
comedy            5702
surreal           5326
                  ...
teen sleuth          1
evil twins           1
paternity test       1
QVC                  1
cornetto triolgy     1
Name: count, Length: 73050, dtype: int64
```

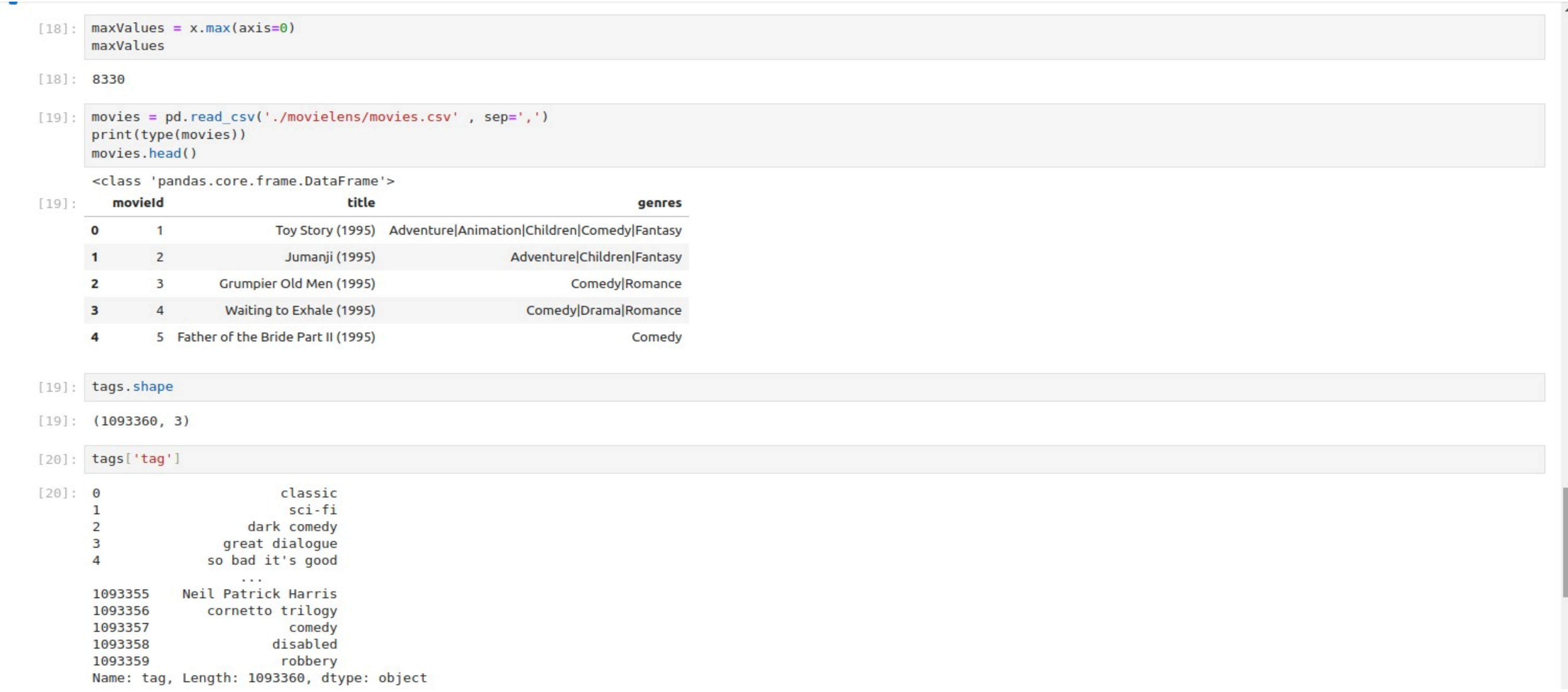
Performing more analysis using pandas and matplotlib to answer the research question



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

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

Performing more analysis using pandas and matplotlib to answer the research question



```
[18]: maxValues = x.max(axis=0)
      maxValues
```

```
[18]: 8330
```

```
[19]: movies = pd.read_csv('./movielens/movies.csv' , sep=',')
      print(type(movies))
      movies.head()
```

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

[19]:

	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

```
[19]: tags.shape
```

```
[19]: (1093360, 3)
```

```
[20]: tags['tag']
```

```
[20]: 0                      classic
      1                       sci-fi
      2                  dark comedy
      3               great dialogue
      4             so bad it's good
                         ...
      1093355    Neil Patrick Harris
      1093356        cornetto trilogy
      1093357                 comedy
      1093358               disabled
      1093359                robbery
      Name: tag, Length: 1093360, dtype: object
```

Reporting findings/analyses

As per the analysis and as shown in graph the most watch genres is SCIFI with 8330 counts.