Recommender Systems with Python

In [1]:

```
import numpy as np
import pandas as pd
```

In [2]:

```
columns_names = ['user_id','item_id','rating','timestamp']
```

In [3]:

```
df = pd.read_csv('u.data',sep='\t',names=columns_names)
```

In [4]:

```
df.head()
```

Out[4]:

	user_id	item_id	rating	timestamp
0	0	50	5	881250949
1	0	172	5	881250949
2	0	133	1	881250949
3	196	242	3	881250949
4	186	302	3	891717742

In [5]:

```
movie_titles = pd.read_csv('Movie_Id_Titles')
```

In [6]:

```
movie_titles.head()
```

Out[6]:

	item_id	title
0	1	Toy Story (1995)
1	2	GoldenEye (1995)
2	3	Four Rooms (1995)
3	4	Get Shorty (1995)
4	5	Copycat (1995)

In [7]:

```
df=pd.merge(df,movie_titles,on='item_id')
```

In [8]:

```
df.head()
```

Out[8]:

	user_id	item_id	rating	timestamp	title
0	0	50	5	881250949	Star Wars (1977)
1	290	50	5	880473582	Star Wars (1977)
2	79	50	4	891271545	Star Wars (1977)
3	2	50	5	888552084	Star Wars (1977)
4	8	50	5	879362124	Star Wars (1977)

In [12]:

```
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [13]:

```
sns.set_style('white')
```

In [15]:

```
df.groupby('title')['rating'].mean().sort_values(ascending=False).head()
```

Out[15]:

```
title
They Made Me a Criminal (1939)                5.0
Marlene Dietrich: Shadow and Light (1996)     5.0
Saint of Fort Washington, The (1993)          5.0
Someone Else's America (1995)                 5.0
Star Kid (1997)                               5.0
Name: rating, dtype: float64
```

In [17]:

```
df.groupby('title')['rating'].count().sort_values(ascending=False).head()
```

Out[17]:

```
title
Star Wars (1977)              584
Contact (1997)                509
Fargo (1996)                  508
Return of the Jedi (1983)     507
Liar Liar (1997)              485
Name: rating, dtype: int64
```

In [19]:

```
ratings = pd.DataFrame(df.groupby('title')['rating'].mean())
```

In [20]:

```
ratings.head()
```

Out[20]:

	rating
title	
'Til There Was You (1997)	2.333333
1-900 (1994)	2.600000
101 Dalmatians (1996)	2.908257
12 Angry Men (1957)	4.344000
187 (1997)	3.024390

In [21]:
```
ratings['num of ratings'] = df.groupby('title')['rating'].count()
```

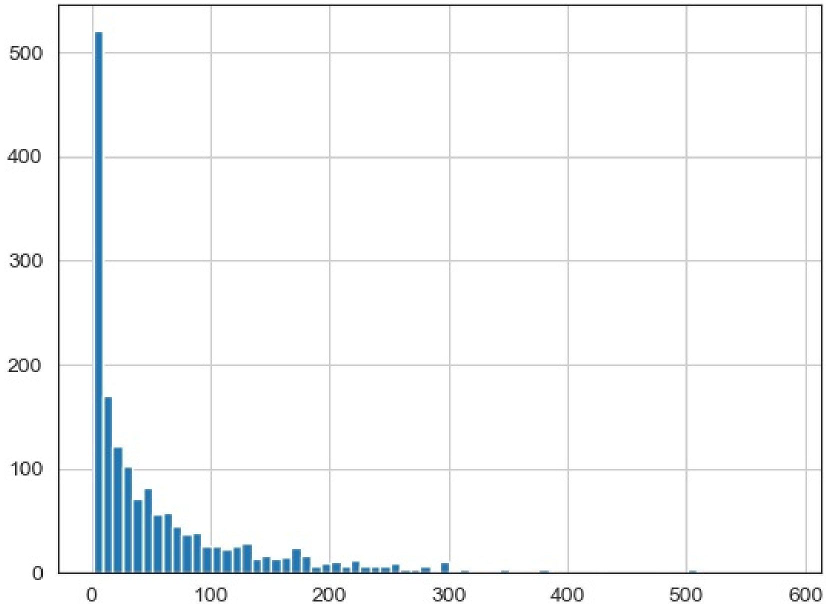
In [22]:
```
ratings.head()
```

Out[22]:

	rating	num of ratings
title		
'Til There Was You (1997)	2.333333	9
1-900 (1994)	2.600000	5
101 Dalmatians (1996)	2.908257	109
12 Angry Men (1957)	4.344000	125
187 (1997)	3.024390	41

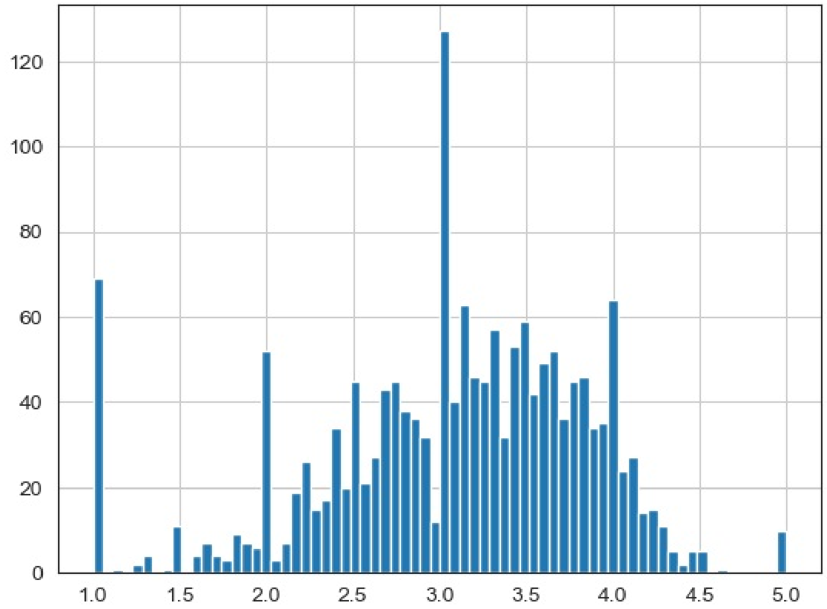
In [23]:
```
ratings ['num of ratings'].hist(bins=70)
```

Out[23]:
```
<AxesSubplot:>
```



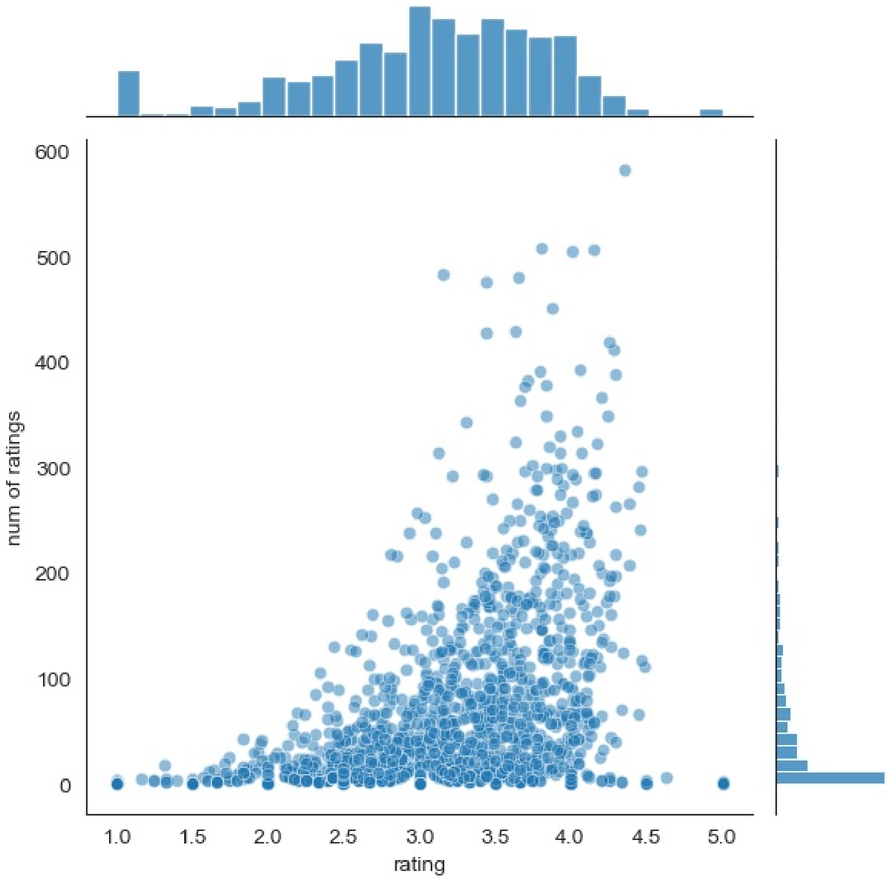
In [24]:
```
ratings['rating'].hist(bins=70)
```

Out[24]:
```
<AxesSubplot:>
```



```
In [25]: sns.jointplot(x='rating',y='num of ratings',data=ratings,alpha=0.5)

Out[25]: <seaborn.axisgrid.JointGrid at 0x1b8a08478e0>
```



In [ ]:

```
```

In [26]:

```
#Create a matrix with the user id on one axis and movie ratings on the other.

moviemat = df.pivot_table(index='user_id',columns='title',values='rating')
```

In [27]:

```
moviemat.head()

#Most people have not seen all the movies
```

Out[27]:

title	'Til There Was You (1997)	1-900 (1994)	101 Dalmatians (1996)	12 Angry Men (1957)	187 (1997)	2 Days in the Valley (1996)	20,000 Leagues Under the Sea (1954)	2001: A Space Odyssey (1968)	3 Ninjas: High Noon At Mega Mountain (1998)	39 Steps, The (1935)	...	Yankee Zulu (1994)	Year of the Horse (1997)	You So Crazy (1994)	Young Frankenstein (1974)	Yo G (1
user_id																
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	
1	NaN	NaN	2.0	5.0	NaN	NaN	3.0	4.0	NaN	NaN	...	NaN	NaN	NaN	5.0	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	1.0	NaN	...	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	2.0	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	

5 rows × 1664 columns

In [28]:

```
#Check most rated movies
ratings.sort_values('num of ratings',ascending=False).head(10)
```

Out[28]:

	rating	num of ratings
title		
Star Wars (1977)	4.359589	584
Contact (1997)	3.803536	509
Fargo (1996)	4.155512	508
Return of the Jedi (1983)	4.007890	507
Liar Liar (1997)	3.156701	485
English Patient, The (1996)	3.656965	481
Scream (1996)	3.441423	478
Toy Story (1995)	3.878319	452
Air Force One (1997)	3.631090	431
Independence Day (ID4) (1996)	3.438228	429

In [29]:

```
#Let's choose 2 movies and grab the user ratings
```

In [30]:

```
starwars_user_ratings = moviemat['Star Wars (1977)']
liarliar_user_ratings = moviemat['Liar Liar (1997)']
```

In [31]:

```
starwars_user_ratings.head()
```

Out[31]:

```
user_id
0    5.0
1    5.0
2    5.0
3    NaN
4    5.0
Name: Star Wars (1977), dtype: float64
```

In [33]:

```
similar_to_starwars = moviemat.corrwith(starwars_user_ratings)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\numpy\lib\function_base.py:2683: RuntimeWarning: Degrees of freedom
<= 0 for slice
  c = cov(x, y, rowvar, dtype=dtype)
C:\ProgramData\Anaconda3\lib\site-packages\numpy\lib\function_base.py:2542: RuntimeWarning: divide by zero enco
untered in true_divide
  c *= np.true_divide(1, fact)
```

In [34]:

```
similar_to_liarliar = moviemat.corrwith(liarliar_user_ratings)
```

```
C:\ProgramData\Anaconda3\lib\site-packages\numpy\lib\function_base.py:2683: RuntimeWarning: Degrees of freedom
<= 0 for slice
  c = cov(x, y, rowvar, dtype=dtype)
C:\ProgramData\Anaconda3\lib\site-packages\numpy\lib\function_base.py:2542: RuntimeWarning: divide by zero enco
untered in true_divide
  c *= np.true_divide(1, fact)
```

In [35]:

```
corr_starwars = pd.DataFrame(similar_to_starwars,columns=['Correlation'])
corr_starwars.dropna(inplace=True)
```

In [36]:

```
corr_starwars.head()
```

Out[36]:

title	Correlation
'Til There Was You (1997)	0.872872
1-900 (1994)	-0.645497
101 Dalmatians (1996)	0.211132
12 Angry Men (1957)	0.184289
187 (1997)	0.027398

In [37]:
```
corr_starwars.sort_values('Correlation',ascending=False).head(10)
```

Out[37]:

title	Correlation
Commandments (1997)	1.0
Cosi (1996)	1.0
No Escape (1994)	1.0
Stripes (1981)	1.0
Man of the Year (1995)	1.0
Hollow Reed (1996)	1.0
Beans of Egypt, Maine, The (1994)	1.0
Good Man in Africa, A (1994)	1.0
Old Lady Who Walked in the Sea, The (Vieille qui marchait dans la mer, La) (1991)	1.0
Outlaw, The (1943)	1.0

In [38]:
```
corr_starwars = corr_starwars.join(ratings['num of ratings'])
```

In [39]:
```
corr_starwars.head()
```

Out[39]:

title	Correlation	num of ratings
'Til There Was You (1997)	0.872872	9
1-900 (1994)	-0.645497	5
101 Dalmatians (1996)	0.211132	109
12 Angry Men (1957)	0.184289	125
187 (1997)	0.027398	41

In [43]:
```
corr_starwars[corr_starwars['num of ratings']>100].sort_values('Correlation',
                                                              ascending=False).head()
```

Out[43]:

title	Correlation	num of ratings
Star Wars (1977)	1.000000	584
Empire Strikes Back, The (1980)	0.748353	368
Return of the Jedi (1983)	0.672556	507
Raiders of the Lost Ark (1981)	0.536117	420
Austin Powers: International Man of Mystery (1997)	0.377433	130

In [45]:
```
corr_liarliar = pd.DataFrame(similar_to_liarliar,columns=['Correlation'])
```

In [47]:
```
corr_liarliar.dropna(inplace=True)
```

In [48]:
```
corr_liarliar = corr_liarliar.join(ratings['num of ratings'])
```

In [49]:
```
corr_liarliar[corr_liarliar['num of ratings']>100].sort_values('Correlation',
                                                              ascending = False).head()
```

Out[49]:

	Correlation	num of ratings
title		
Liar Liar (1997)	1.000000	485
Batman Forever (1995)	0.516968	114
Mask, The (1994)	0.484650	129
Down Periscope (1996)	0.472681	101
Con Air (1997)	0.469828	137

In [ ]:

Loading [MathJax]/jax/output/CommonHTML/fonts/TeX/fontdata.js