In [1]:
```
import numpy as np
import pandas as pd
```

In [3]:
```
columns_names = ['user_id','item_id', 'rating','timestamp']
```

In [5]:
```
df = pd.read_csv('u.data', sep='\t', names=columns_names)
```

In [7]:
```
df.head()
```

Out[7]:

	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 [9]:
```
movies_titles = pd.read_csv('Movie_Id_Titles')
```

In [11]:
```
movies_titles.head()
```

Out[11]:

	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 [13]:
```
df = pd.merge(df, movies_titles, on='item_id')
```

In [15]:
```
df.head()
```

Out[15]:

	user_id	item_id	rating	timestamp	title
0	0	50	5	881250949	Star Wars (1977)
1	0	172	5	881250949	Empire Strikes Back, The (1980)
2	0	133	1	881250949	Gone with the Wind (1939)
3	196	242	3	881250949	Kolya (1996)
4	186	302	3	891717742	L.A. Confidential (1997)

In [17]:
```
import matplotlib.pyplot as plt
import seaborn as sns
```

```
sns.set_style('white')
%matplotlib inline
```

In [19]:
```
df.groupby('title')['rating'].mean().sort_values(ascending=False).head()
```

Out[19]:
```
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 [21]:
```
df.groupby('title')['rating'].count().sort_values(ascending=False).head()
```

Out[21]:
```
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 [23]:
```
ratings = pd.DataFrame(df.groupby('title')['rating'].mean())
```

In [25]:
```
ratings.head()
```

Out[25]:

title	rating
'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 [29]:
```
ratings['num of ratings'] = pd.DataFrame(df.groupby('title')['rating'].count())
```

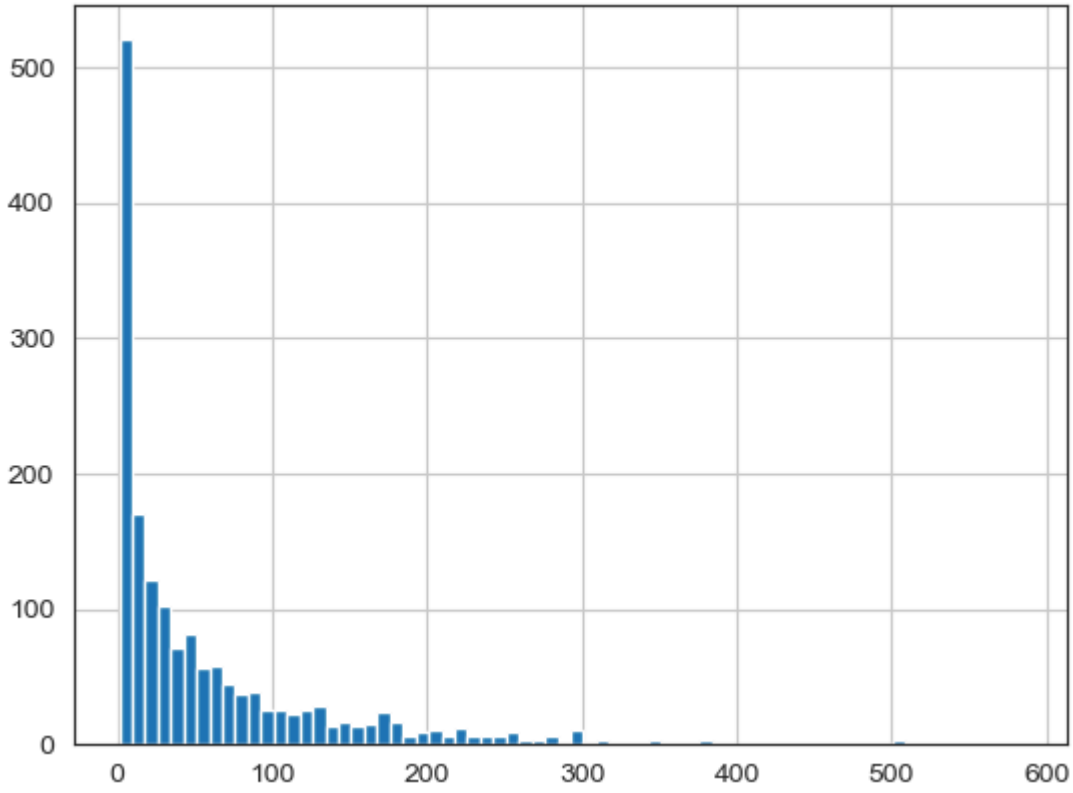
In [31]:
```
ratings.head()
```

Out[31]:

title	rating	num of ratings
'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

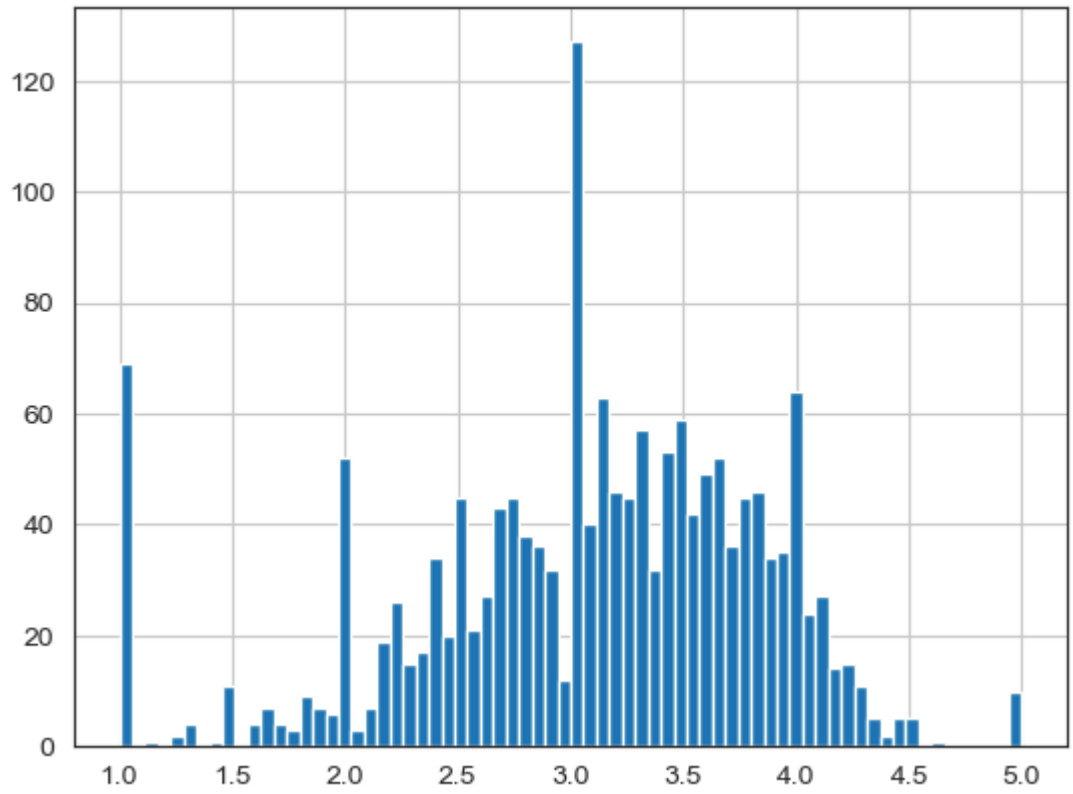
In [33]:
```
ratings['num of ratings'].hist(bins=70)
```

Out[33]: <Axes: >



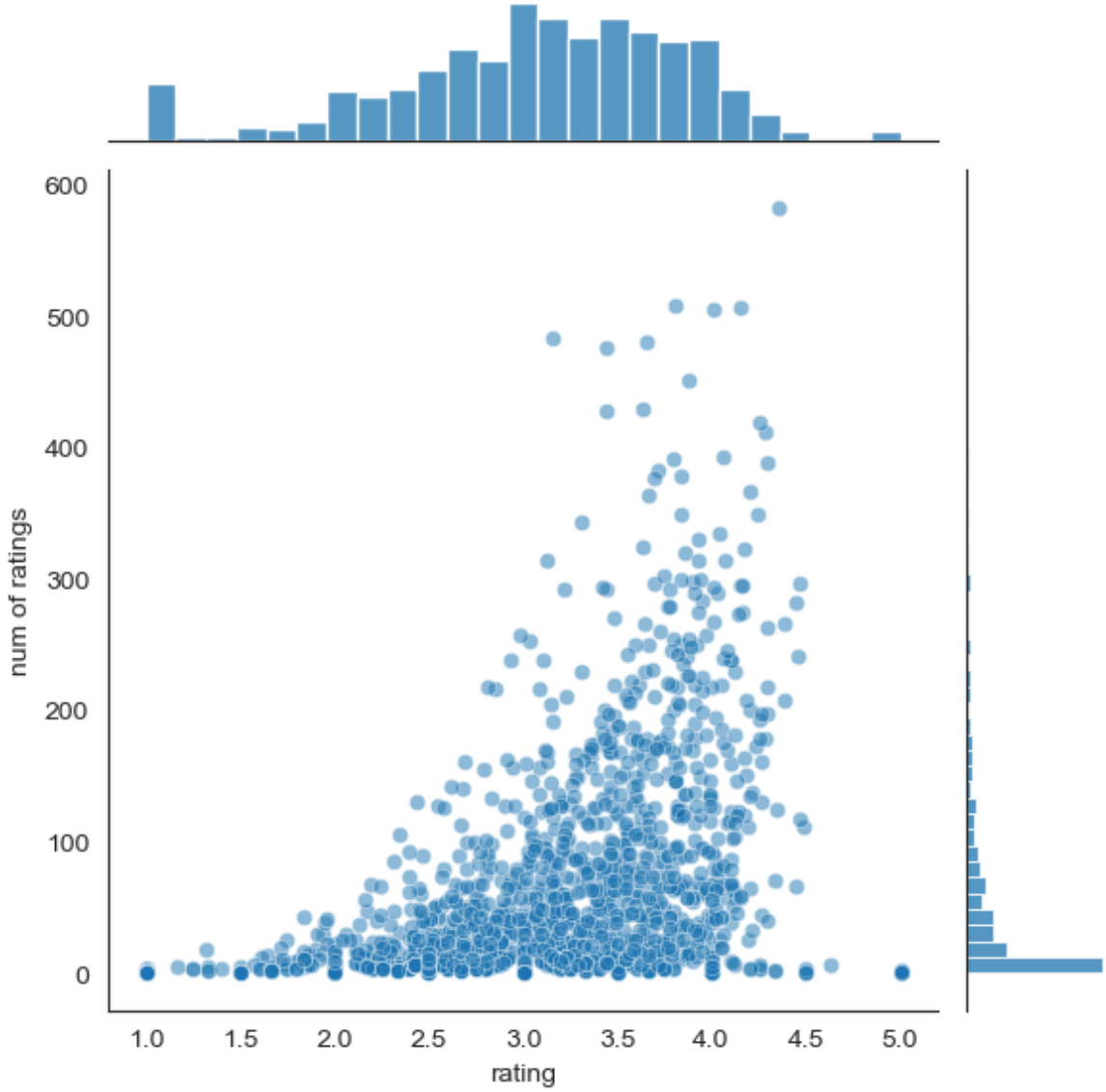
In [35]:
```
ratings['rating'].hist(bins=70)
```

Out[35]: <Axes: >



In [37]:
```
sns.jointplot(x='rating', y='num of ratings', data=ratings, alpha=0.5)
```

Out[37]: <seaborn.axisgrid.JointGrid at 0x272df97c260>



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

Out[39]:

	user_id	item_id	rating	timestamp	title
0	0	50	5	881250949	Star Wars (1977)
1	0	172	5	881250949	Empire Strikes Back, The (1980)
2	0	133	1	881250949	Gone with the Wind (1939)
3	196	242	3	881250949	Kolya (1996)
4	186	302	3	891717742	L.A. Confidential (1997)

In [41]:
```
moviemat = df.pivot_table(index='user_id', columns='title', values='rating')
```

In [43]:
```
moviemat.head()
```

Out[43]:

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 Ninj Hi Noon Me Mounta (199
user_id									
0	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N
1	NaN	NaN	2.0	5.0	NaN	NaN	3.0	4.0	N
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
3	NaN	NaN	NaN	NaN	2.0	NaN	NaN	NaN	N
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	N

5 rows × 1664 columns

In [45]:
```
ratings.sort_values('num of ratings', ascending=False).head(10)
```

Out[45]:

title	rating	num of ratings
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 [47]:
```
star_wars_user_ratings = moviemat['Star Wars (1977)']
liarliar_user_ratings = moviemat['Liar Liar (1997)']
```

In [49]:
```
star_wars_user_ratings.head()
```

Out[49]:
```
user_id
0    5.0
1    5.0
2    5.0
3    NaN
4    5.0
Name: Star Wars (1977), dtype: float64
```

In [53]:
```
similar_to_starwars = moviemat.corrwith(star_wars_user_ratings)
```

```
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2897: RuntimeWarning: i
nvalid value encountered in divide
  c /= stddev[:, None]
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2898: RuntimeWarning: i
nvalid value encountered in divide
  c /= stddev[None, :]
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2889: RuntimeWarning: D
egrees of freedom <= 0 for slice
  c = cov(x, y, rowvar, dtype=dtype)
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2748: RuntimeWarning: d
ivide by zero encountered in divide
  c *= np.true_divide(1, fact)
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2748: RuntimeWarning: i
nvalid value encountered in multiply
  c *= np.true_divide(1, fact)
```

In [55]:
```
similar_to_liarliar = moviemat.corrwith(liarliar_user_ratings)
```

```
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2889: RuntimeWarning: D
egrees of freedom <= 0 for slice
  c = cov(x, y, rowvar, dtype=dtype)
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2748: RuntimeWarning: d
ivide by zero encountered in divide
  c *= np.true_divide(1, fact)
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2748: RuntimeWarning: i
nvalid value encountered in multiply
  c *= np.true_divide(1, fact)
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2897: RuntimeWarning: i
nvalid value encountered in divide
  c /= stddev[:, None]
C:\learnings\Lib\site-packages\numpy\lib\function_base.py:2898: RuntimeWarning: i
nvalid value encountered in divide
  c /= stddev[None, :]
```

In [57]:
```
corr_star_wars = pd.DataFrame(similar_to_starwars, columns=['Correlation'])
corr_star_wars.dropna(inplace=True)
corr_star_wars.head()
```

Out[57]:

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 [59]:
```
corr_star_wars.sort_values('Correlation', ascending=False).head()
```

Out[59]:

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

In [61]:
```
corr_star_wars = corr_star_wars.join(ratings['num of ratings'])
corr_star_wars.head()
```

Out[61]:

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 [63]:
```
corr_star_wars[corr_star_wars['num of ratings']>100].sort_values('Correlation',
```

Out[63]:

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 [65]:
```
corr_liarliar = pd.DataFrame(similar_to_liarliar, columns=['Correlation'])
```

In [67]:
```
corr_liarliar.dropna(inplace=True)
```

In [69]:
```
corr_liarliar = corr_liarliar.join(ratings['num of ratings'])
```

In [71]:
```
corr_liarliar[corr_liarliar['num of ratings']>100].sort_values('Correlation', as
```

Out[71]:

title	Correlation	num of ratings
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
...	...	...
Hoop Dreams (1994)	-0.184503	117
Ed Wood (1994)	-0.199481	133
Dr. Strangelove or: How I Learned to Stop Worrying and Love the Bomb (1963)	-0.238092	194
Welcome to the Dollhouse (1995)	-0.254231	112
Raging Bull (1980)	-0.308129	116

334 rows × 2 columns

In [ ]: