Lecture21

April 18, 2024

[1]:
```
# matplotlib
# import matplotliub.pyplot as plt
# plt.hist(x)
# plt.pie(x)
# plt.boxplot(x)
# plt.bar(x,y)
# plt.barh(x,y)
# plt.scatter(x,y)
# plt.plot(x,y)
```

1 Exercise

[10]:
```
# read movie_metadata.csv as dataframe
import pandas as pd
df=pd.read_csv('datafiles/movie_metadata.csv')
df.head()
```

[10]:

	color	director_name	num_critic_for_reviews	duration	\
0	Color	James Cameron	723.0	178.0	
1	Color	Gore Verbinski	302.0	169.0	
2	Color	Sam Mendes	602.0	148.0	
3	Color	Christopher Nolan	813.0	164.0	
4	NaN	Doug Walker	NaN	NaN	

	director_facebook_likes	actor_3_facebook_likes	actor_2_name	\
0	0.0	855.0	Joel David Moore	
1	563.0	1000.0	Orlando Bloom	
2	0.0	161.0	Rory Kinnear	
3	22000.0	23000.0	Christian Bale	
4	131.0	NaN	Rob Walker	

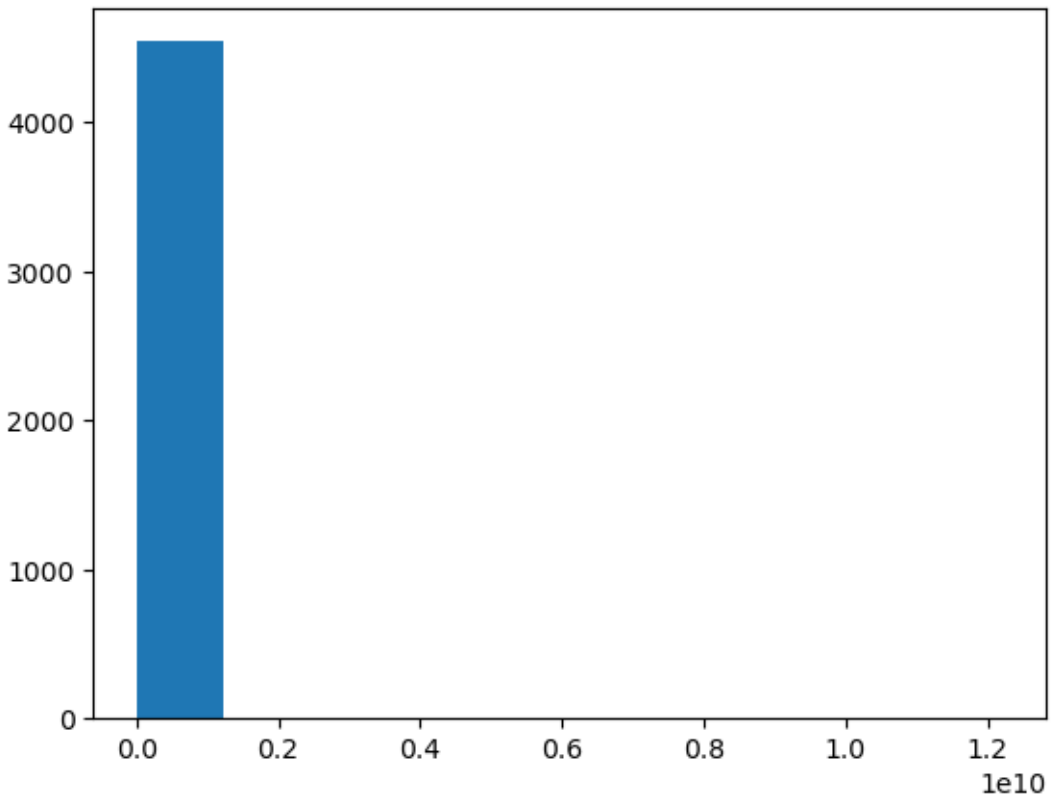
	actor_1_facebook_likes	gross	genres	…	\
0	1000.0	760505847.0	Action|Adventure|Fantasy|Sci-Fi	…	
1	40000.0	309404152.0	Action|Adventure|Fantasy	…	
2	11000.0	200074175.0	Action|Adventure|Thriller	…	
3	27000.0	448130642.0	Action|Thriller	…	
4	131.0	NaN	Documentary	…	

```
  num_user_for_reviews language  country  content_rating       budget  \
0               3054.0  English      USA           PG-13  237000000.0
1               1238.0  English      USA           PG-13  300000000.0
2                994.0  English       UK           PG-13  245000000.0
3               2701.0  English      USA           PG-13  250000000.0
4                  NaN      NaN      NaN             NaN          NaN

   title_year actor_2_facebook_likes imdb_score  aspect_ratio  \
0      2009.0                 936.0        7.9          1.78
1      2007.0                5000.0        7.1          2.35
2      2015.0                 393.0        6.8          2.35
3      2012.0               23000.0        8.5          2.35
4         NaN                  12.0        7.1           NaN

  movie_facebook_likes
0                33000
1                    0
2                85000
3               164000
4                    0

[5 rows x 28 columns]
```

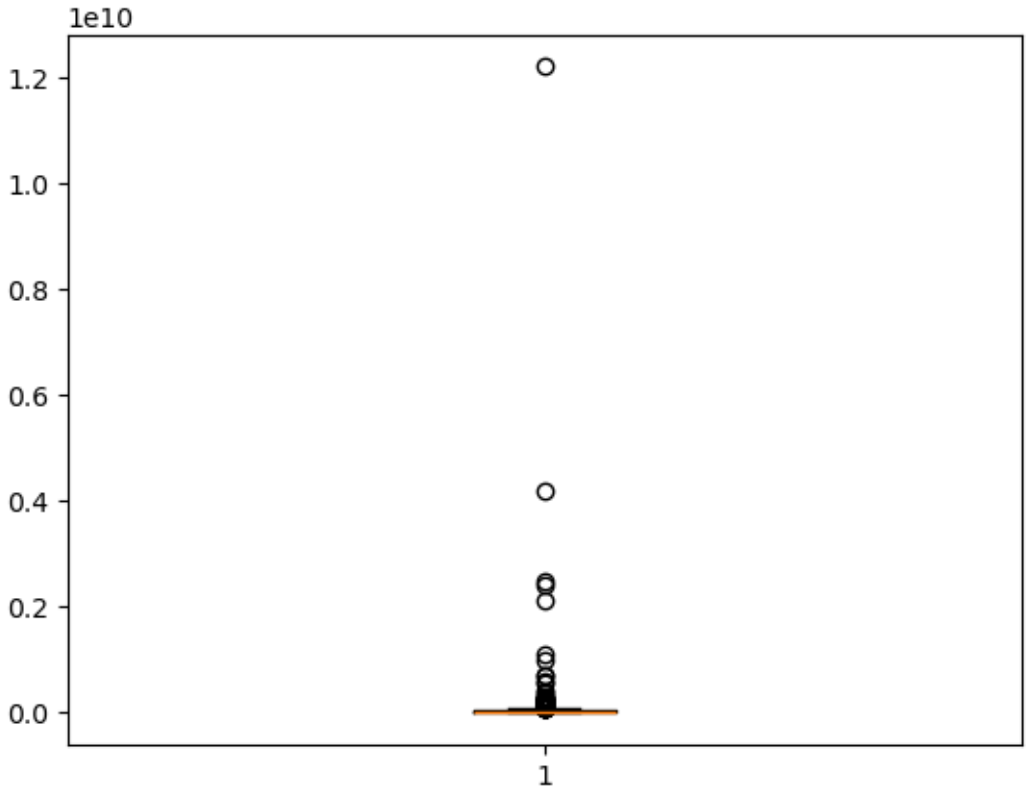
[20]:
```
df.budget.dropna()
```

[20]:
```
0       237000000.0
1       300000000.0
2       245000000.0
3       250000000.0
5       263700000.0
           ...
5035         7000.0
5036         3250.0
5037         9000.0
5040         1400.0
5042         1100.0
Name: budget, Length: 4551, dtype: float64
```

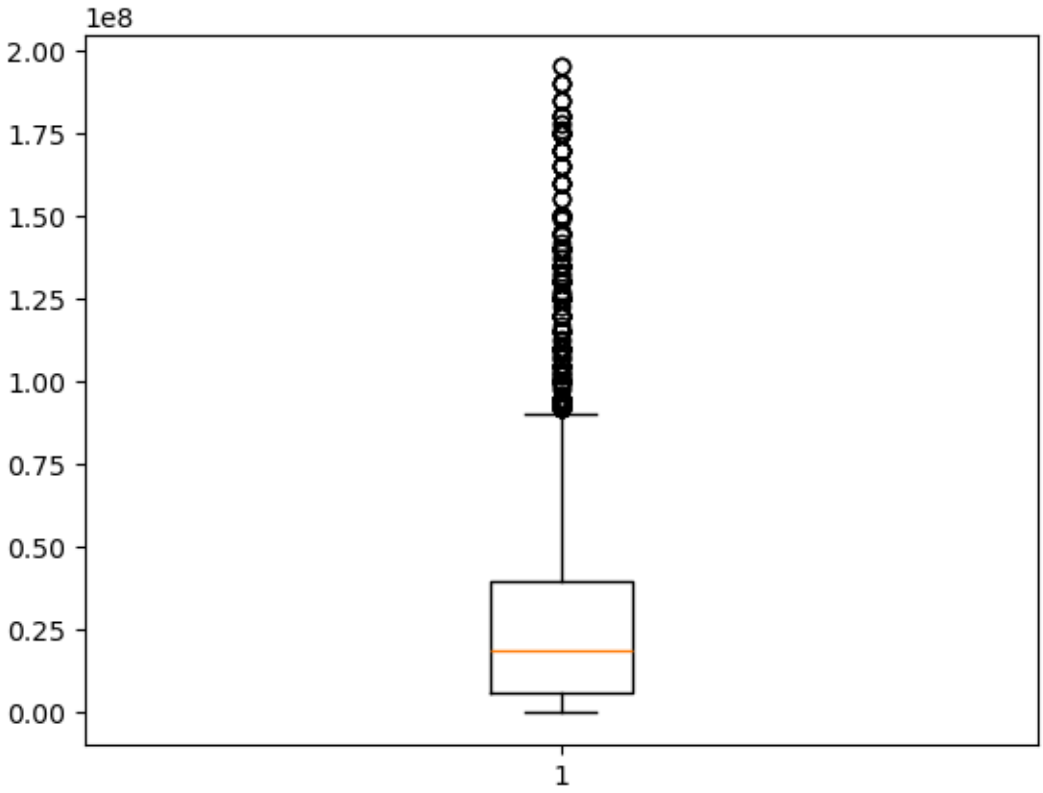
[21]:
```
# create a histogram of the budget
import matplotlib.pyplot as plt
# df.budget.plot(kind='hist')'
plt.hist(df.budget.dropna());
```



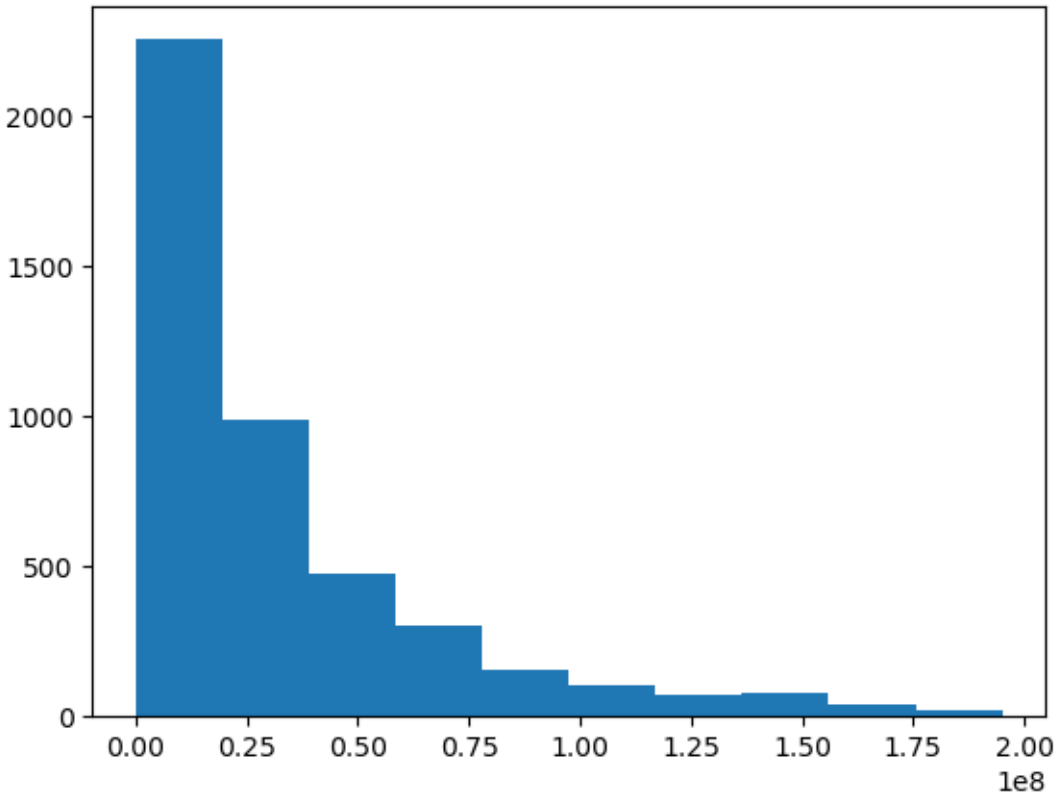
[23]:
```
plt.boxplot(df.budget.dropna());
```



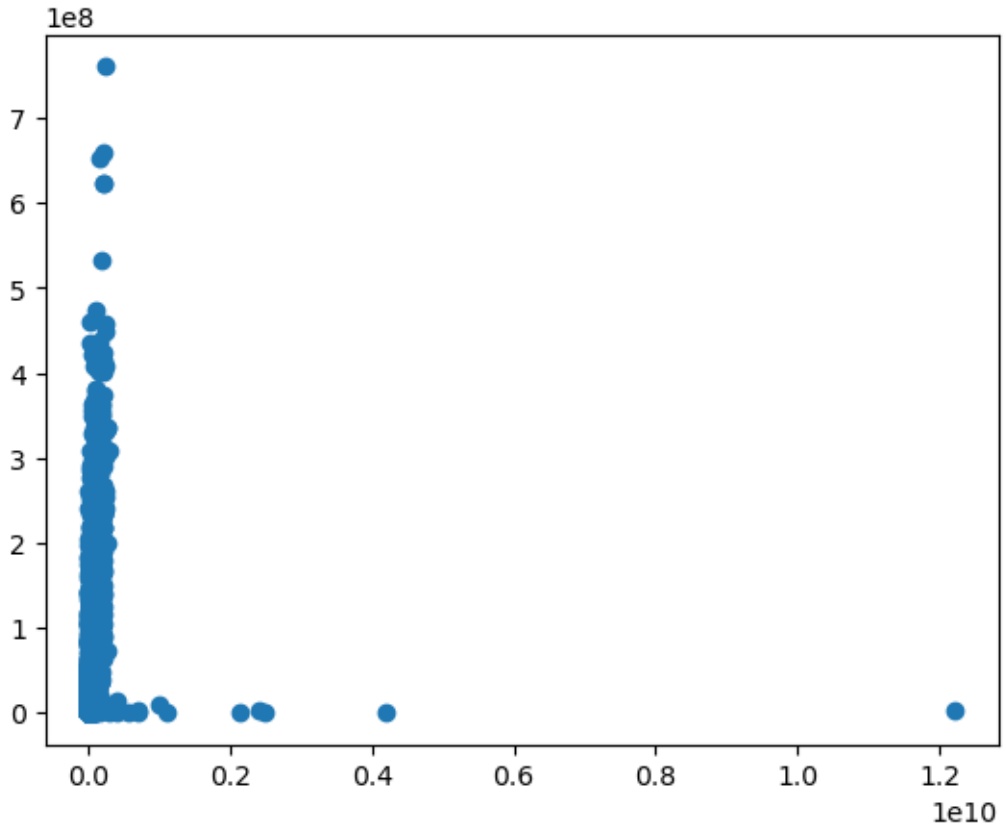
[26]:
```
plt.boxplot(df.loc[df.budget<0.2e9,'budget'].dropna());
```



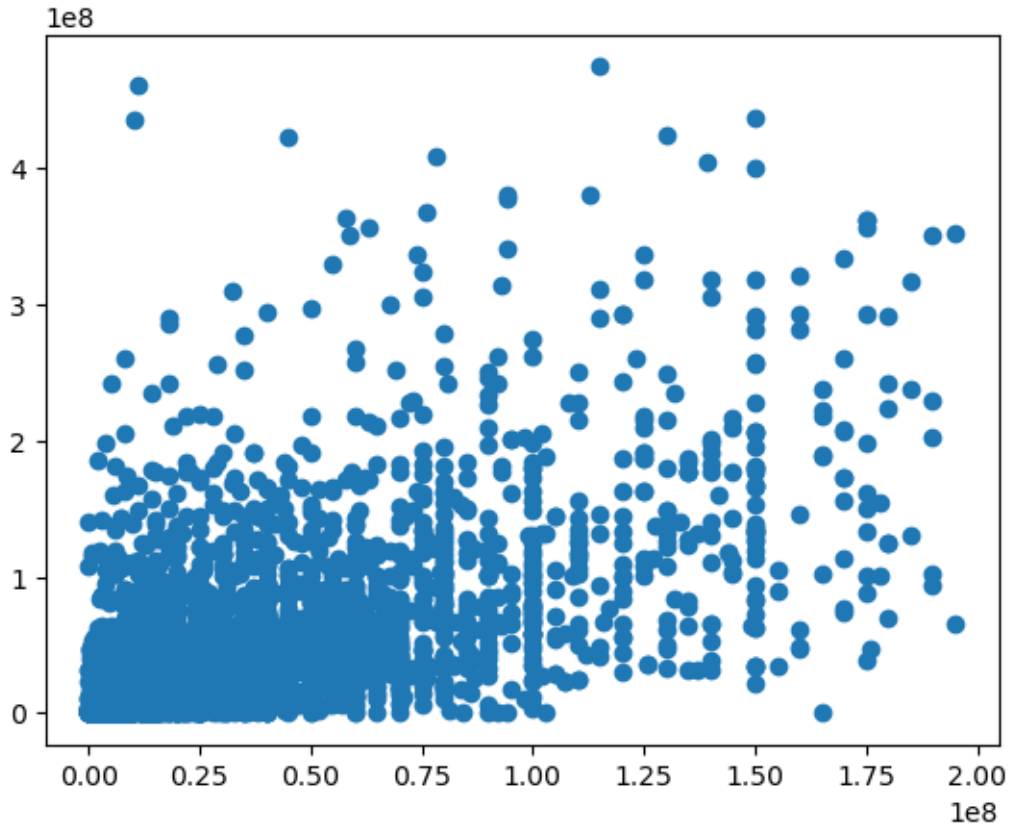
[27]:
```
plt.hist(df.loc[df.budget<0.2e9,'budget'].dropna());
```



[28]:
```
# show the relation between the gross income and the budget
df=df.dropna()
plt.scatter(df.budget, df.gross);
```



[35]:
```
tmpdf=df.loc[(df.budget<0.2e9)&(df.gross<5e8), ['budget','gross']]
plt.scatter(tmpdf.budget, tmpdf.gross);
```



[47]:
```
import numpy as np
mask=df.loc[:,['gross','budget']].apply(lambda x: np.abs(x-x.mean())/x.std()␣
↪<3).all(axis=1)
# mask=df.loc[:,['gross','budget']].apply(lambda x: stats.zscore(x)).all(axis=1)
df.loc[mask,['gross','budget']]
```

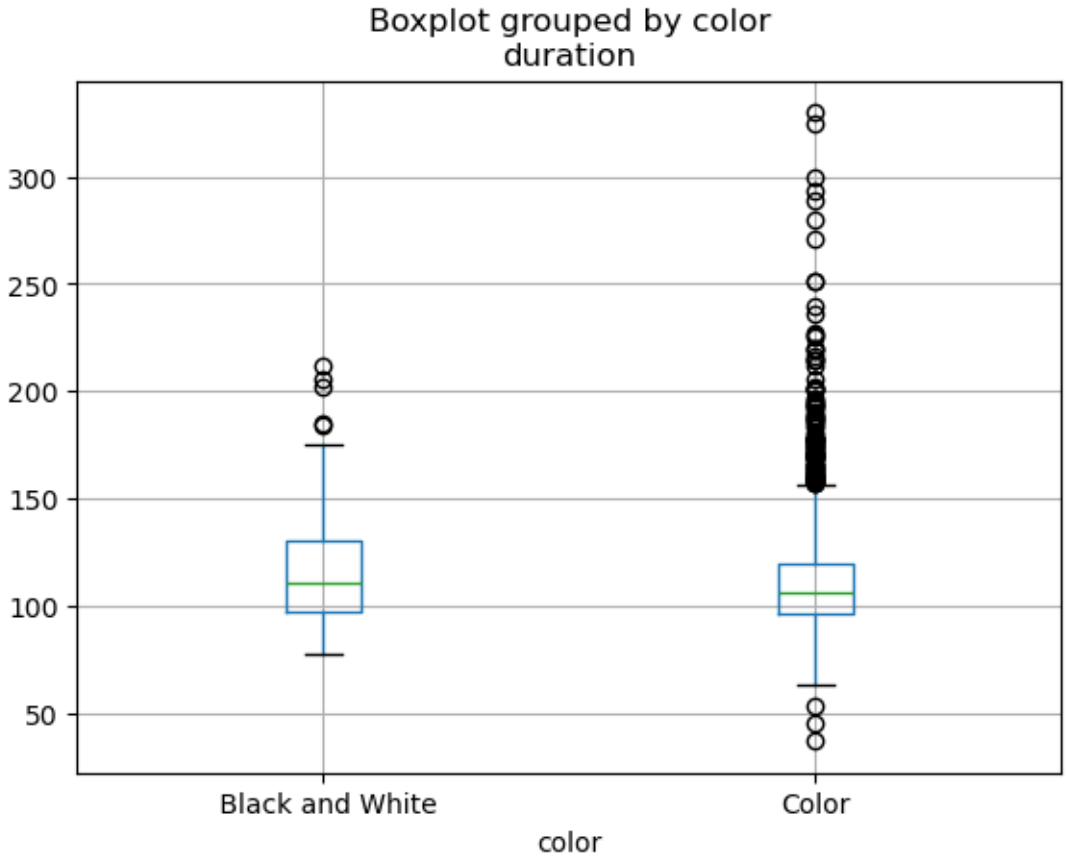
[47]:

	gross	budget
2	200074175.0	245000000.0
5	73058679.0	263700000.0
7	200807262.0	260000000.0
11	200069408.0	209000000.0
12	168368427.0	200000000.0
…	…	…
5026	136007.0	4500.0
5027	673780.0	10000.0
5033	424760.0	7000.0
5035	2040920.0	7000.0
5042	85222.0	1100.0

[3666 rows x 2 columns]

[51]:
```
# what is the duration distrubtion for different kinds of moviews colors
df.loc[:,['color','duration']].boxplot(by='color', column='duration')
```

[51]:
```
<Axes: title={'center': 'duration'}, xlabel='color'>
```



[54]:
```
# pip install seaborn
```

[55]:
```
import seaborn as sns
```

[57]:
```
# displot
df=sns.load_dataset('titanic')
df.head()
```

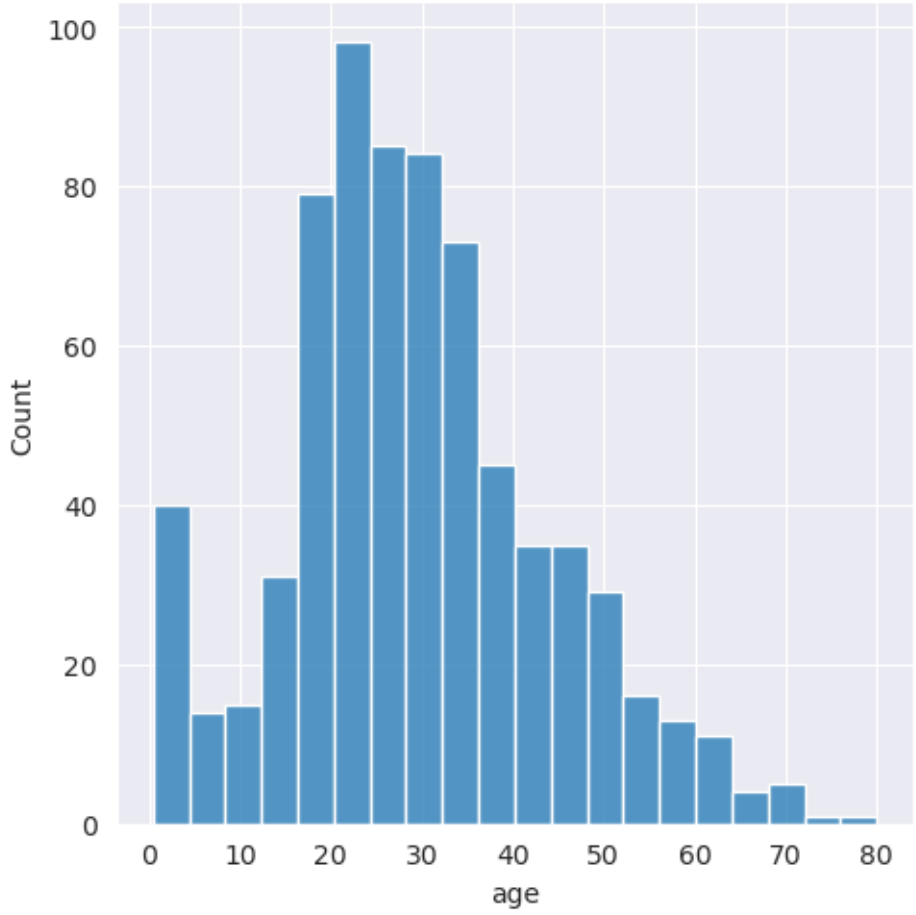
[57]:

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	\
0	0	3	male	22.0	1	0	7.2500	S	Third	
1	1	1	female	38.0	1	0	71.2833	C	First	
2	1	3	female	26.0	0	0	7.9250	S	Third	
3	1	1	female	35.0	1	0	53.1000	S	First	
4	0	3	male	35.0	0	0	8.0500	S	Third	

```
     who  adult_male deck  embark_town alive  alone
0    man        True  NaN  Southampton    no  False
1  woman       False    C    Cherbourg   yes  False
2  woman       False  NaN  Southampton   yes   True
3  woman       False    C  Southampton   yes  False
4    man        True  NaN  Southampton    no   True
```

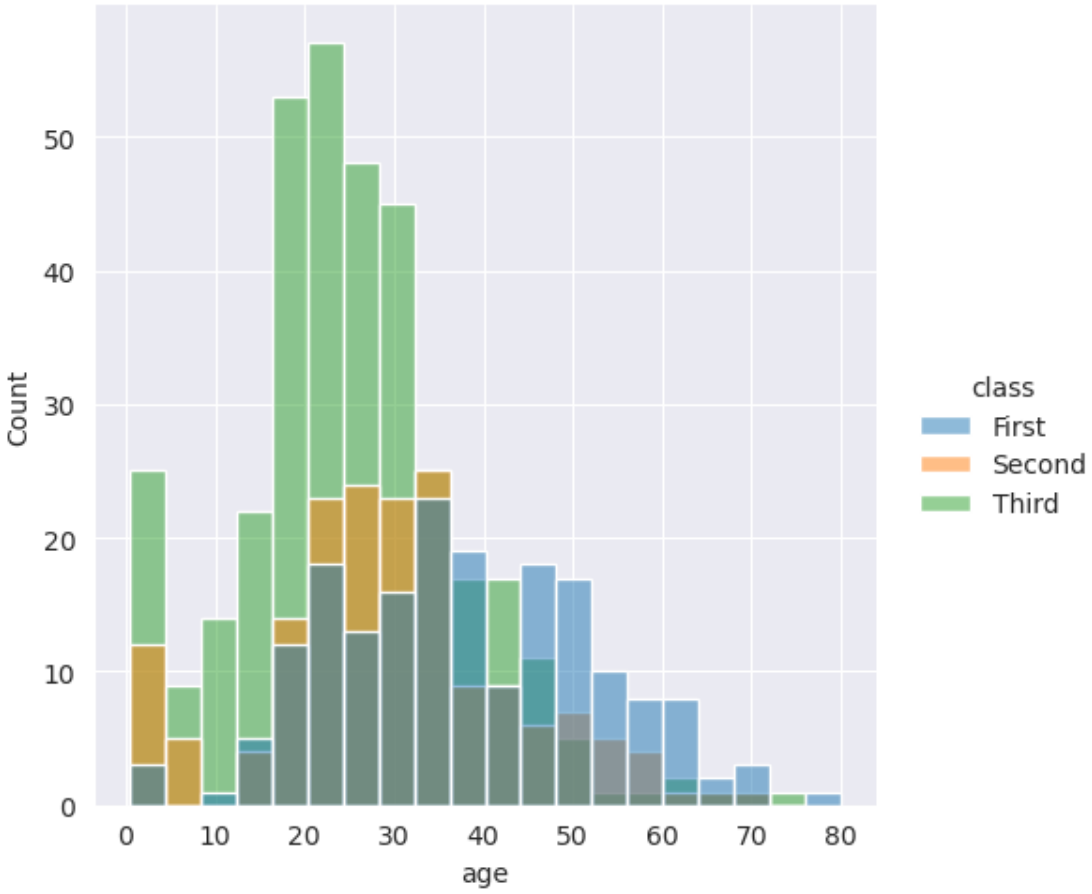
[60]:
```
sns.set_style('darkgrid')
```

[61]:
```
sns.displot(data=df, x='age');
```



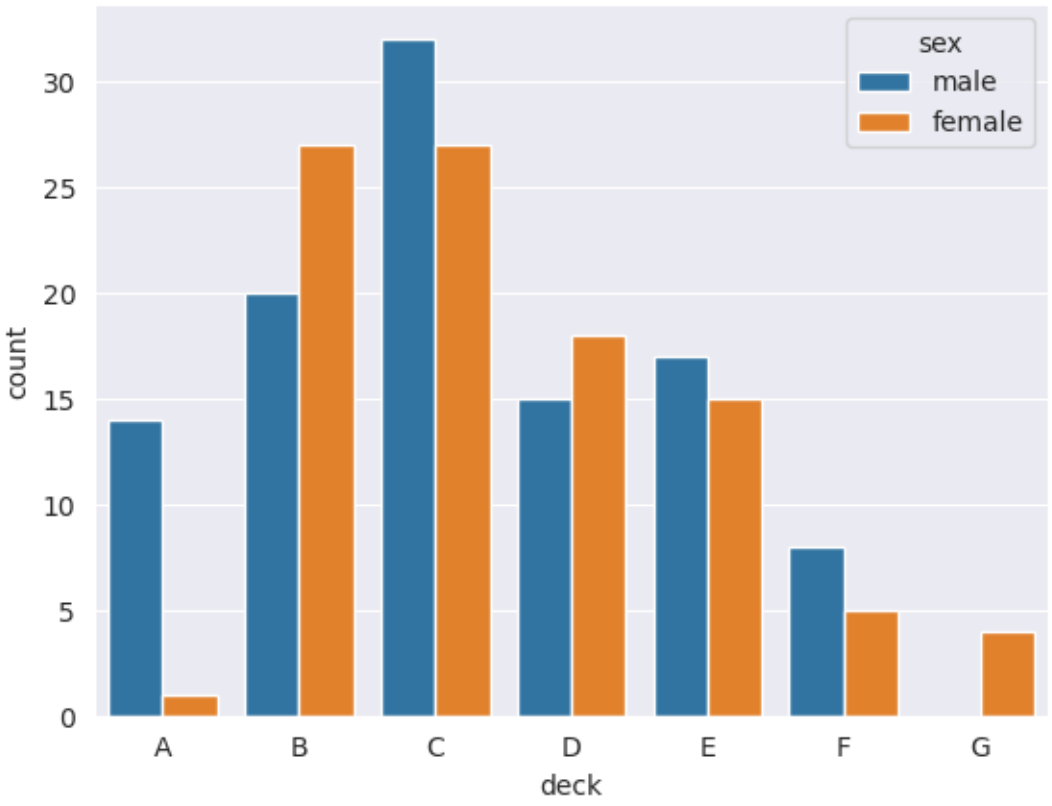
[65]:
```
# displot (cat or cont)
# sns.displot(data=df, x='age')
# sns.displot(data=df, x='deck')
sns.displot(df, x='age', hue='class')
```

[65]: <seaborn.axisgrid.FacetGrid at 0x7fd13512fcd0>

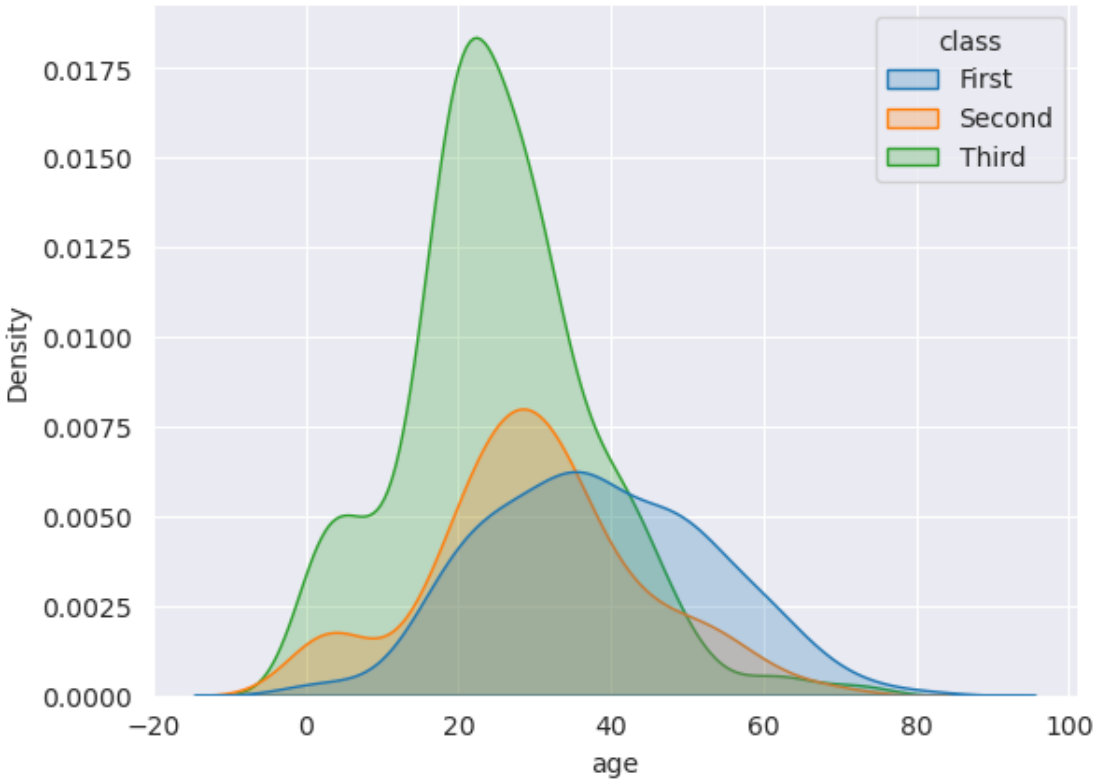


[67]:
```
# countplot (cat or cont)
# sns.countplot(df, x='deck')
sns.countplot(df, x='deck', hue='sex')
```

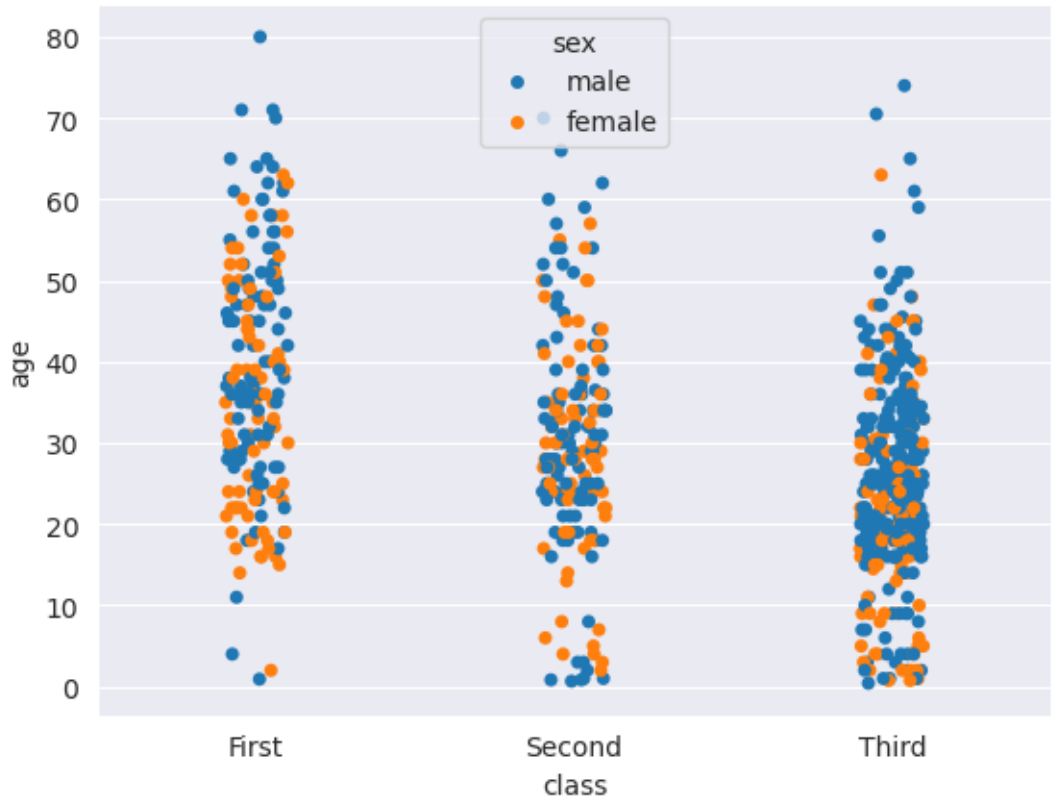
[67]: <Axes: xlabel='deck', ylabel='count'>



[72]:
```
# kdeplot (cont)
# sns.kdeplot(df, x='age', fill=True);
sns.kdeplot(df, x='age', fill=True, hue='class');
```

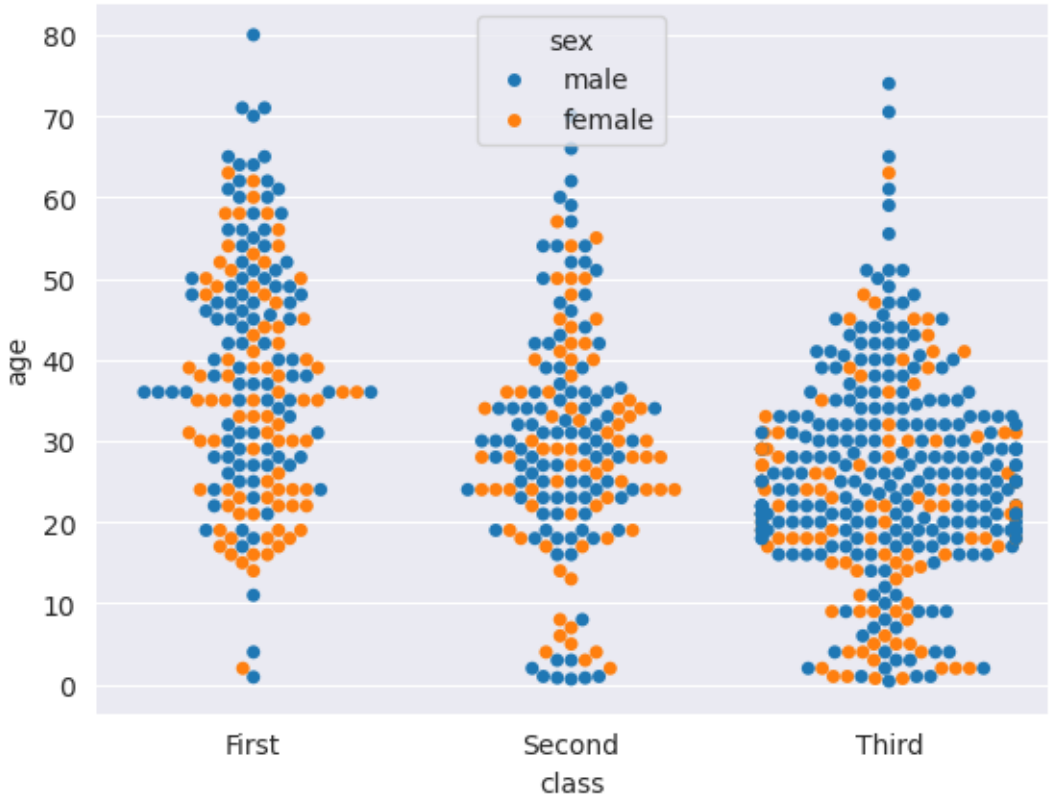


[78]:
```
# Scatterplot (cat vs cont)
# sns.stripplot(data=df, x='class', y='age');
sns.stripplot(data=df, x='class', y='age', hue='sex');
```

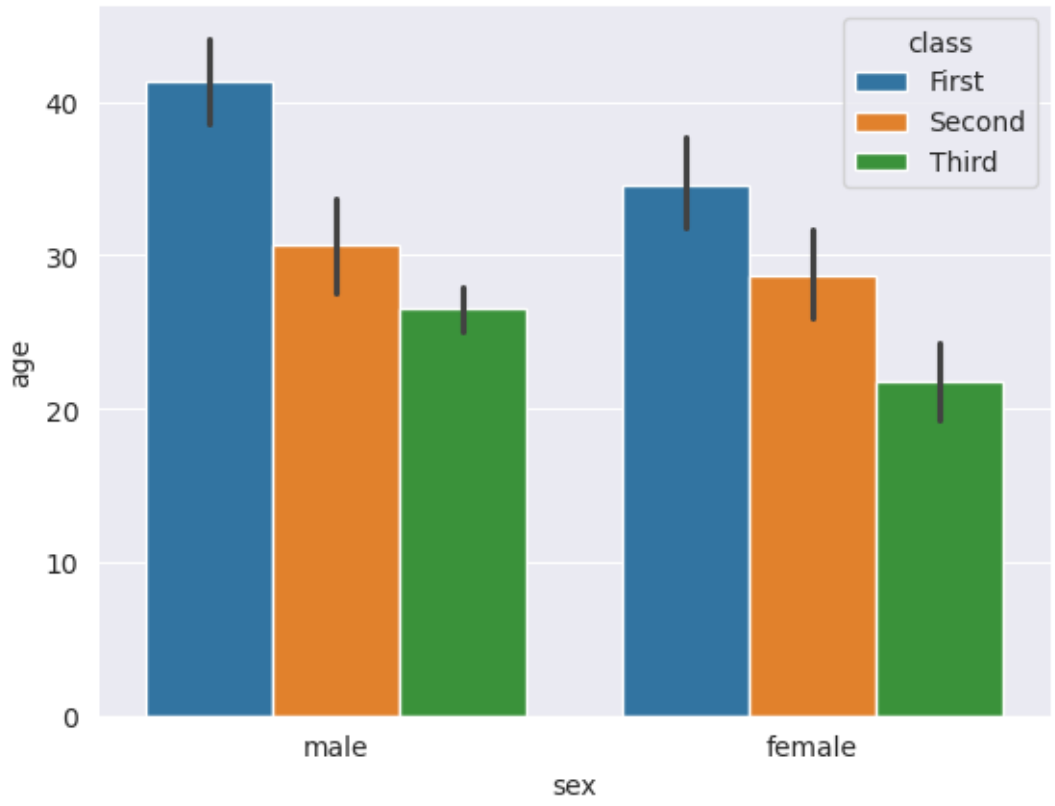


[79]:
```
# sns.swarmplot(data=df, x='class', y='age');
sns.swarmplot(data=df, x='class', y='age', hue='sex');
```

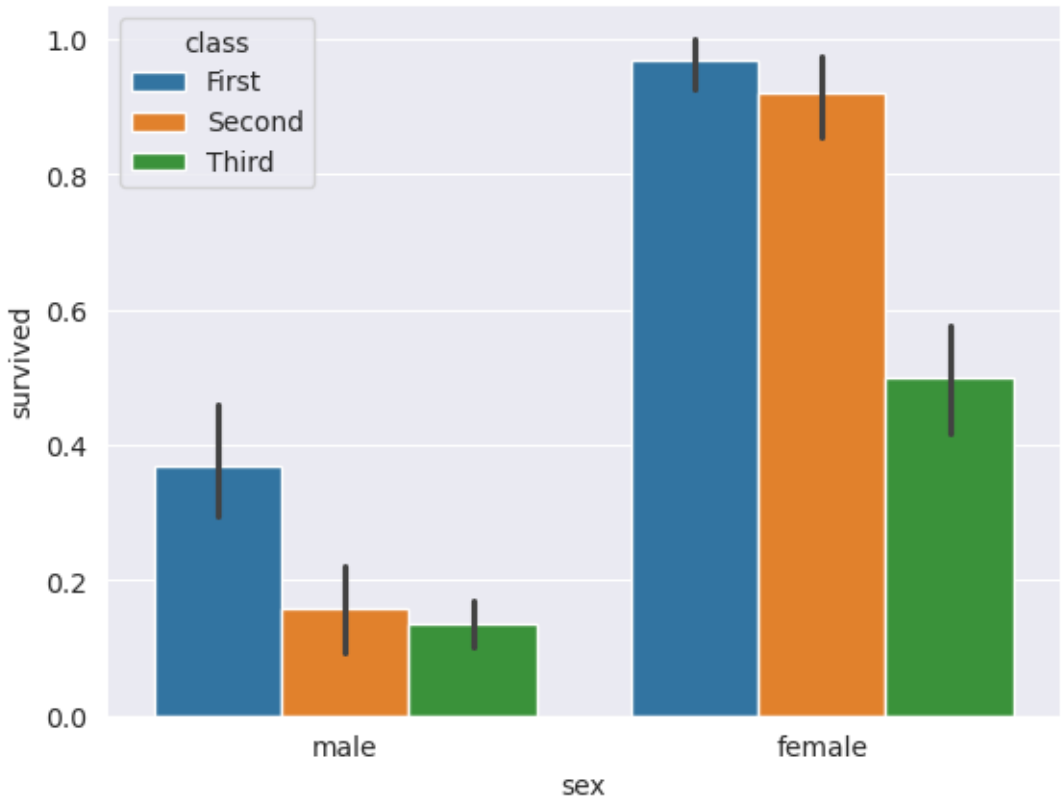
```
/opt/conda/lib/python3.11/site-packages/seaborn/categorical.py:3399:
UserWarning: 15.2% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
```



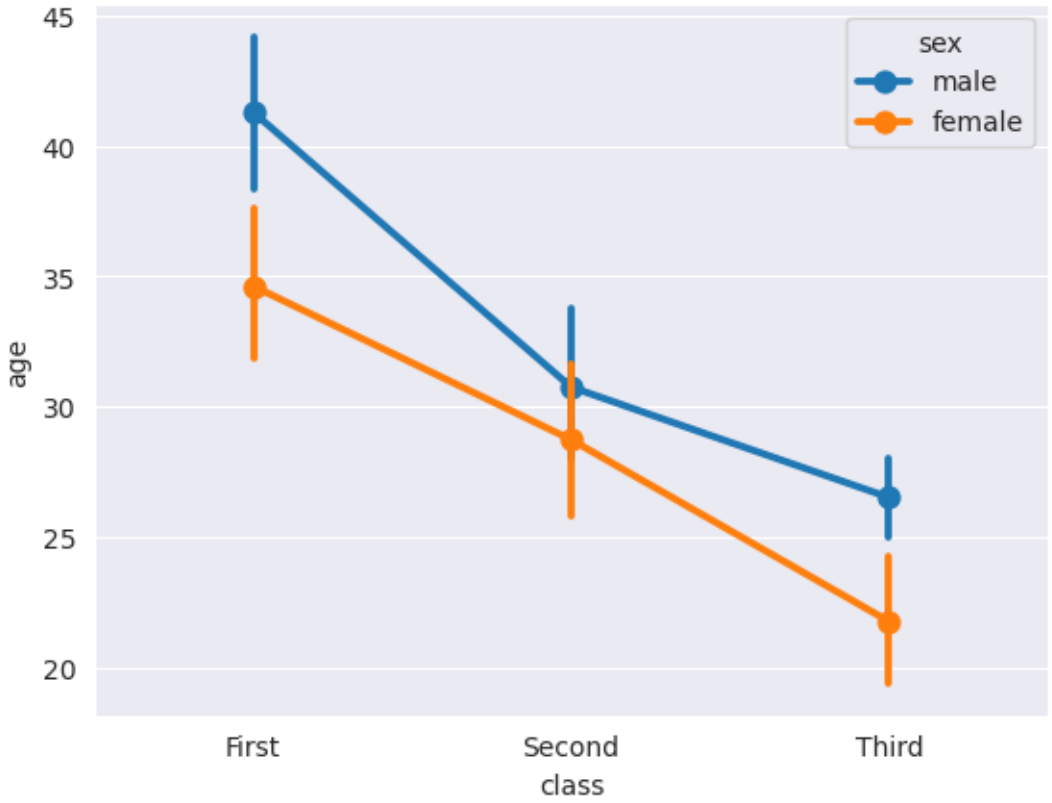
[81]:
```
# barplot (cat vs cont)
# sns.barplot(data=df, x='sex', y='age');
sns.barplot(data=df, x='sex', y='age', hue='class');
```



[83]:
```
# cat vs cat (as int)
# sns.barplot(data=df, x='sex', y='survived');
sns.barplot(data=df, x='sex', y='survived', hue='class');
```



[86]:
```
# pointplot (cat vs cont)
# sns.pointplot(data=df, x='class', y='age');
sns.pointplot(data=df, x='class', y='age', hue='sex');
```


45
40
35
30
25
20
age
sex
male
female
First
Second
Third
class

[ ]: