

DSBDAL_08

Assignment No.8

Title: **Data Visualization I** 1. Use the inbuilt dataset 'titanic'. The dataset contains 891 rows and contains information about the passengers who boarded the unfortunate Titanic ship. Use the Seaborn library to see if we can find any patterns in the data or not. 2. Write a code to check how the price of the ticket (column name: 'fare') for each passenger is distributed by plotting a histogram.

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

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

```
[ ]: ds=sns.load_dataset('titanic')
```

```
[ ]: ds
```

```
[ ]:      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
..      ...
886          0         2   male  27.0     0     0  13.0000         S  Second
887          1         1  female  19.0     0     0  30.0000         S   First
888          0         3  female   NaN     1     2  23.4500         S   Third
889          1         1   male  26.0     0     0  30.0000         C   First
890          0         3   male  32.0     0     0   7.7500         Q   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
..      ...
```

```
886     man         True  NaN  Southampton    no   True
887  woman        False    B  Southampton   yes   True
888  woman        False  NaN  Southampton    no  False
889     man         True    C   Cherbourg   yes   True
890     man         True  NaN  Queenstown    no   True
```

[891 rows x 15 columns]

```
[ ]: ds.head()
```

```
[ ]:      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
```

```
[ ]: ds.tail()
```

```
[ ]:      survived  pclass    sex  age  sibsp  parch    fare embarked  class \
886          0         2   male  27.0     0     0   13.00         S  Second
887          1         1  female  19.0     0     0   30.00         S   First
888          0         3  female   NaN     1     2   23.45         S   Third
889          1         1   male  26.0     0     0   30.00         C   First
890          0         3   male  32.0     0     0    7.75         Q   Third
```

```
      who  adult_male deck  embark_town alive alone
886     man         True  NaN  Southampton    no   True
887  woman        False    B  Southampton   yes   True
888  woman        False  NaN  Southampton    no  False
889     man         True    C   Cherbourg   yes   True
890     man         True  NaN  Queenstown    no   True
```

```
[ ]: ds.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 15 columns):
#   Column                Non-Null Count  Dtype
---

```

```

0  survived      891 non-null    int64
1  pclass        891 non-null    int64
2  sex           891 non-null    object
3  age           714 non-null    float64
4  sibsp         891 non-null    int64
5  parch         891 non-null    int64
6  fare          891 non-null    float64
7  embarked      889 non-null    object
8  class         891 non-null    category
9  who           891 non-null    object
10 adult_male    891 non-null    bool
11 deck         203 non-null    category
12 embark_town   889 non-null    object
13 alive         891 non-null    object
14 alone         891 non-null    bool
dtypes: bool(2), category(2), float64(2), int64(4), object(5)
memory usage: 80.7+ KB

```

```
[ ]: ds.isnull().sum()
```

```

[ ]: survived      0
    pclass         0
    sex            0
    age           177
    sibsp          0
    parch          0
    fare           0
    embarked       2
    class          0
    who            0
    adult_male     0
    deck           688
    embark_town     2
    alive          0
    alone          0
dtype: int64

```

Distribution Plot

```
[ ]: sns.distplot(ds['fare'])
```

<ipython-input-8-811d9a59ceff>:1: UserWarning:

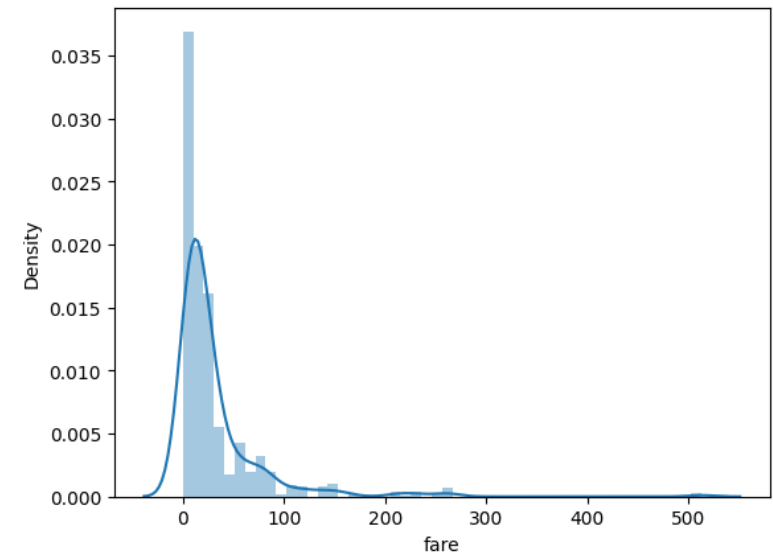
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see <https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>

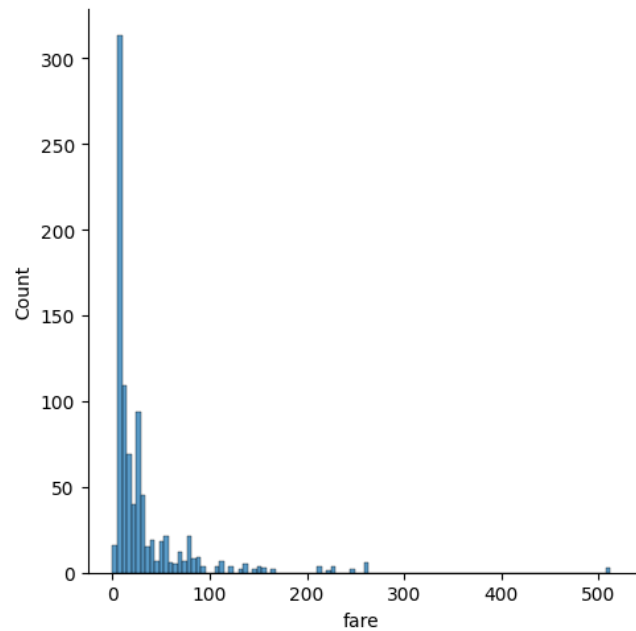
```
sns.distplot(ds['fare'])
```

```
[ ]: <Axes: xlabel='fare', ylabel='Density'>
```



```
[ ]: sns.displot(ds['fare'],kde=False)
```

```
[ ]: <seaborn.axisgrid.FacetGrid at 0x7fef055844c0>
```



Histogram

```
[ ]: sns.distplot(ds['fare'], kde=False, bins=10)
```

<ipython-input-10-477c88ba67b1>:1: UserWarning:

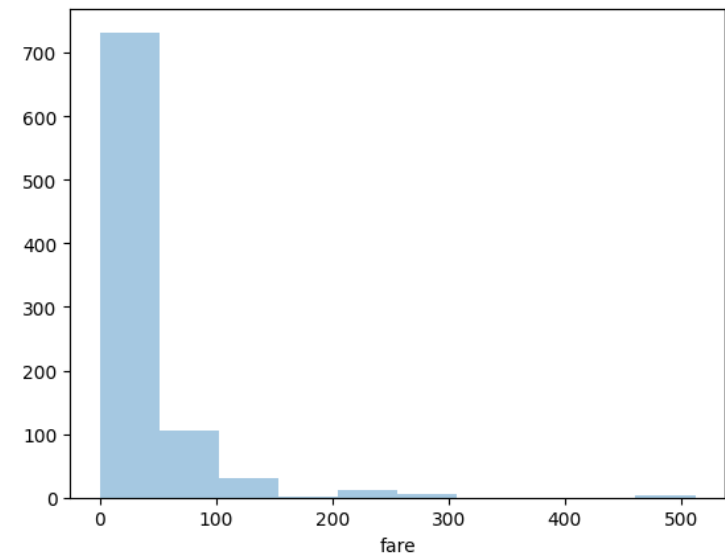
`distplot` is a deprecated function and will be removed in seaborn v0.14.0.

Please adapt your code to use either `displot` (a figure-level function with similar flexibility) or `histplot` (an axes-level function for histograms).

For a guide to updating your code to use the new functions, please see <https://gist.github.com/mwaskom/de44147ed2974457ad6372750bbe5751>

```
sns.distplot(ds['fare'], kde=False, bins=10)
```

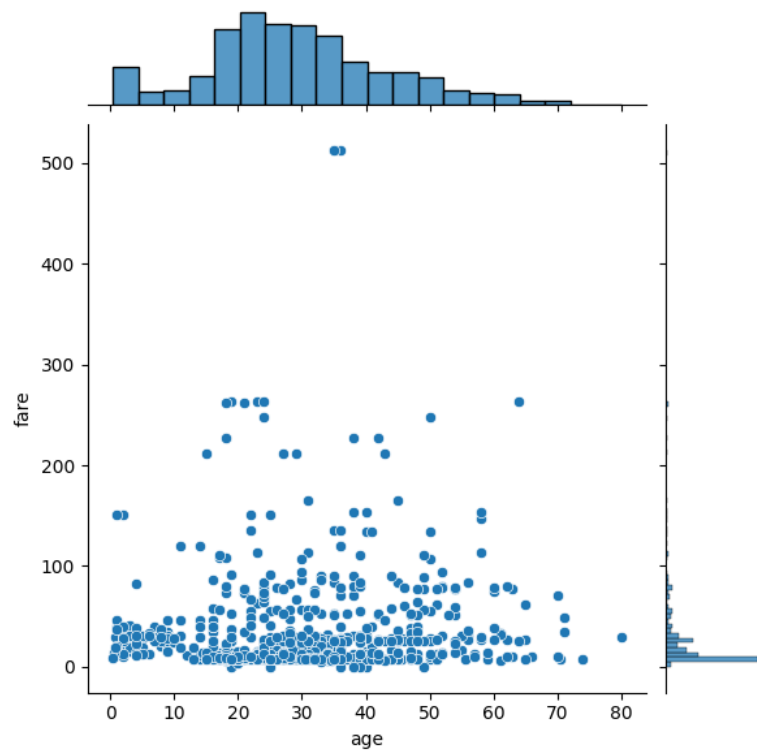
```
[ ]: <Axes: xlabel='fare'>
```



Joint Plot

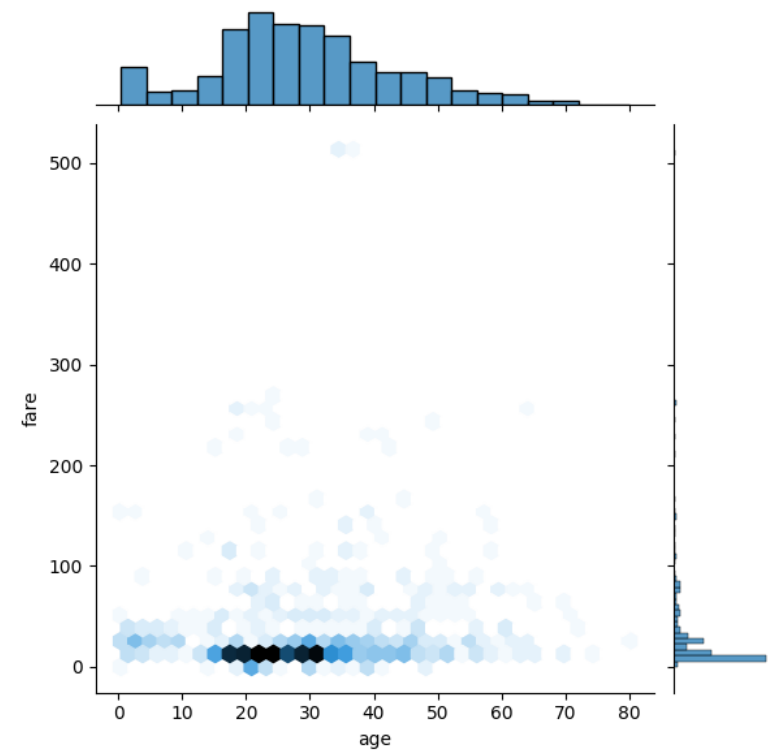
```
[ ]: sns.jointplot(x='age', y='fare', data=ds)
```

```
[ ]: <seaborn.axisgrid.JointGrid at 0x7fef05d202e0>
```



```
[ ]: sns.jointplot(x='age', y='fare', data=ds, kind='hex')
```

```
[ ]: <seaborn.axisgrid.JointGrid at 0x7fef151e02e0>
```



Pair Plot

```
[ ]: nds=ds
```

```
[ ]: nds=nds.dropna()
```

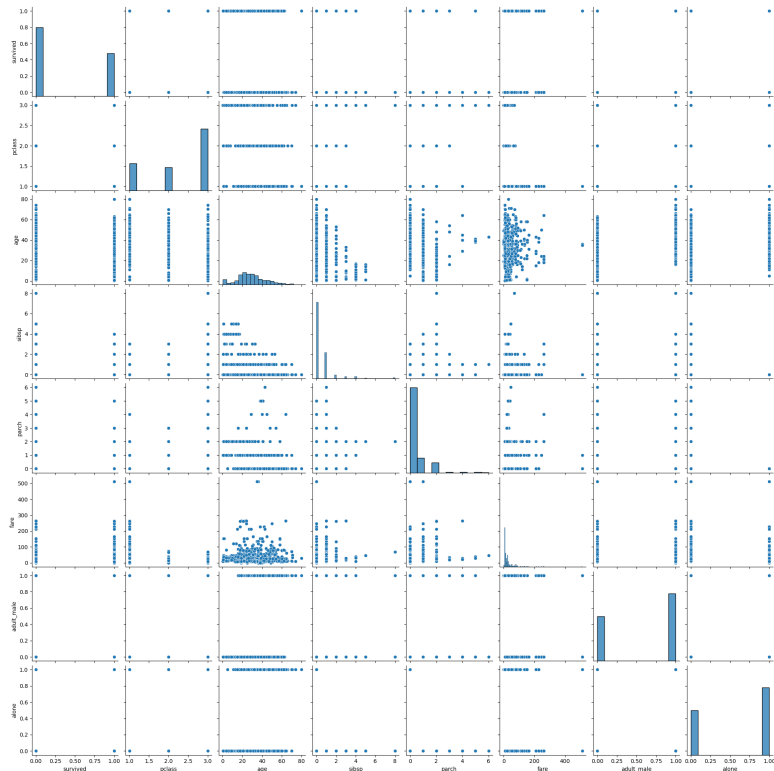
```
[ ]: nds.isnull().sum()
```

```
[ ]: survived    0
      pclass     0
      sex        0
      age        0
      sibsp      0
```

```
parch      0
fare       0
embarked   0
class      0
who        0
adult_male 0
deck       0
embark_town 0
alive      0
alone      0
dtype: int64
```

```
[ ]: sns.pairplot(ds)
```

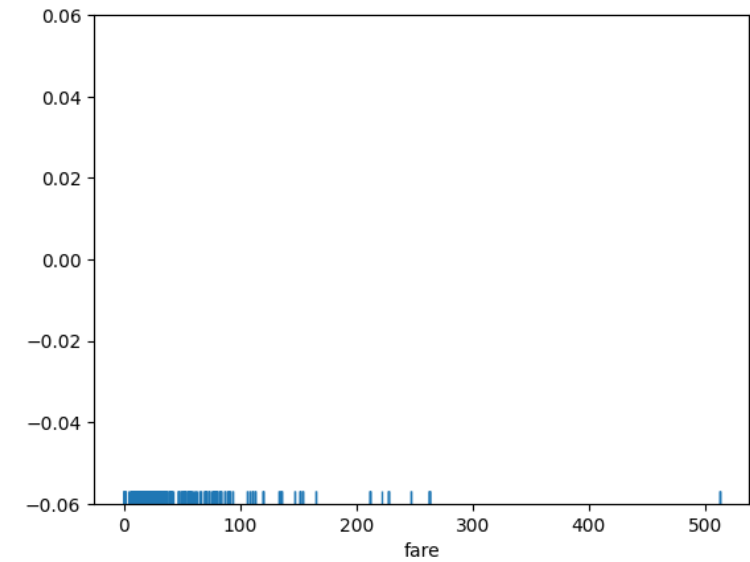
```
[ ]: <seaborn.axisgrid.PairGrid at 0x7fef00eaec20>
```



Rug Plot

```
[ ]: sns.rugplot(ds['fare'])
```

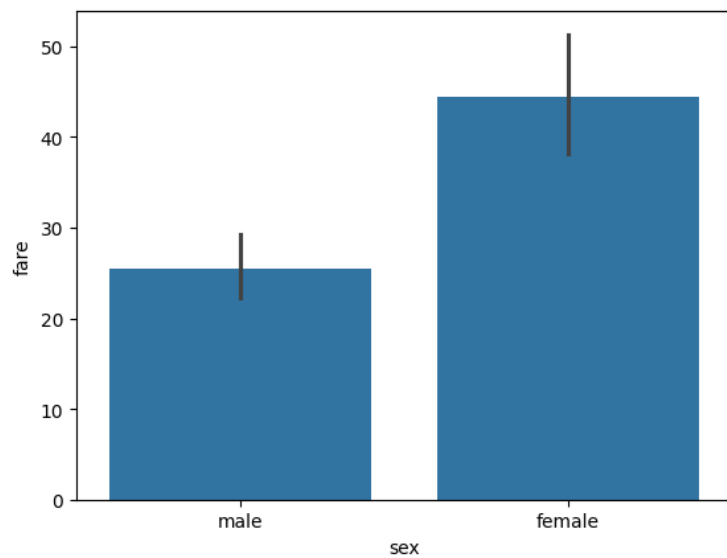
```
[ ]: <Axes: xlabel='fare'>
```



Bar Plot

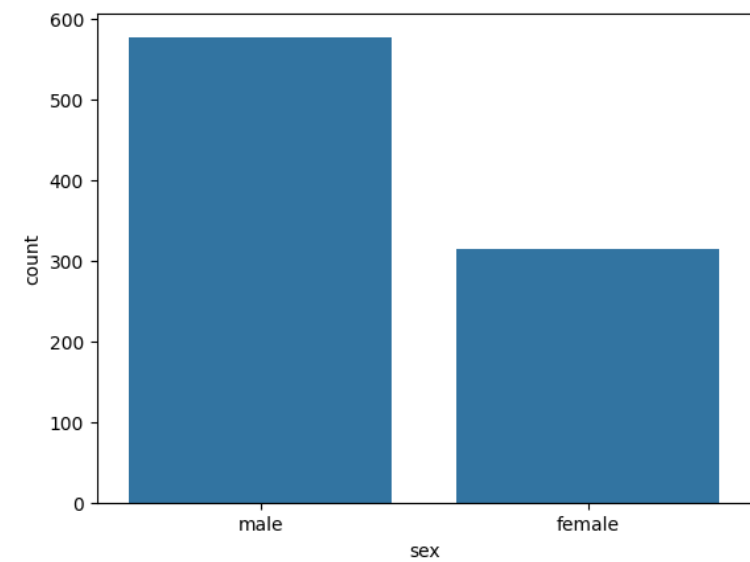
```
[ ]: sns.barplot(x='sex', y='fare', data=ds)
```

```
[ ]: <Axes: xlabel='sex', ylabel='fare'>
```



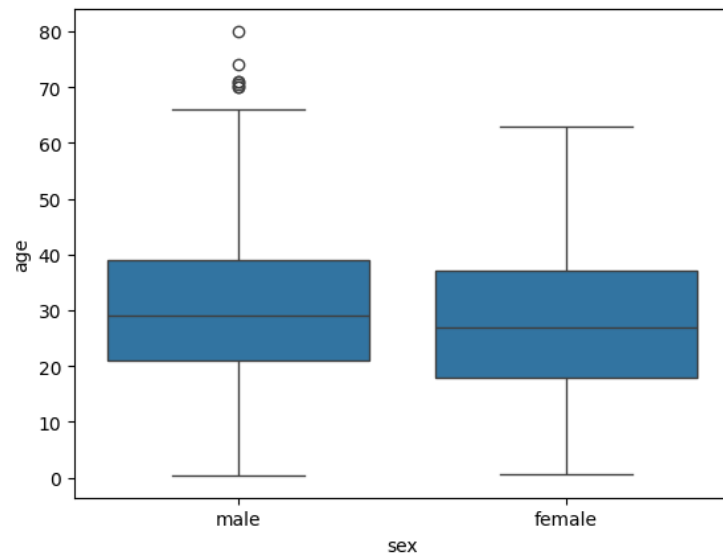
Count Plot

```
[ ]: sns.countplot(x='sex', data=ds)  
[ ]: <Axes: xlabel='sex', ylabel='count'>
```



Boxplot

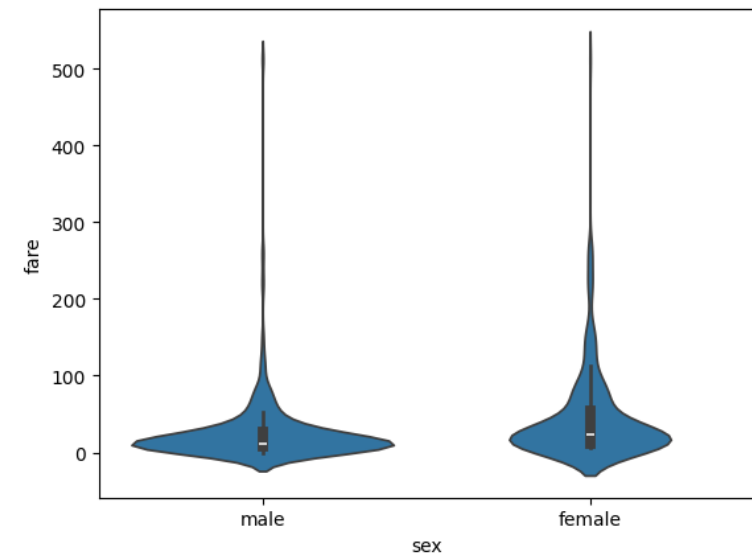
```
[ ]: sns.boxplot(x='sex', y='age', data=ds)  
[ ]: <Axes: xlabel='sex', ylabel='age'>
```



Violin Plot

```
[ ]: sns.violinplot(x='sex',y='fare',data=ds)
```

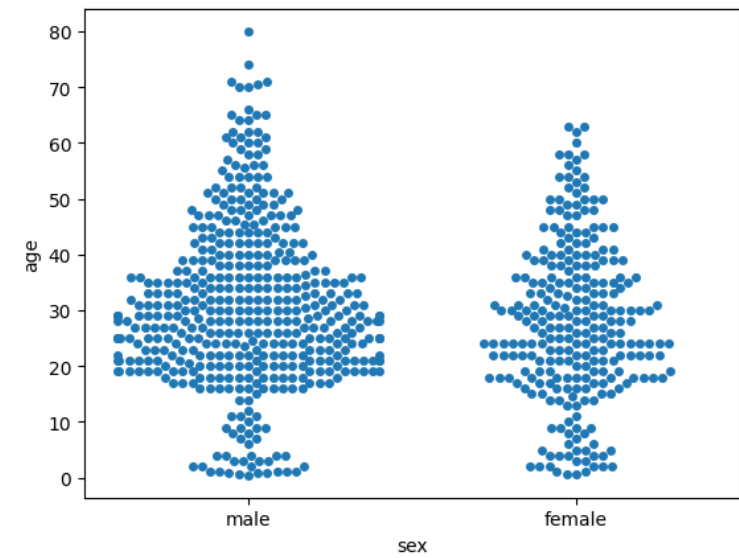
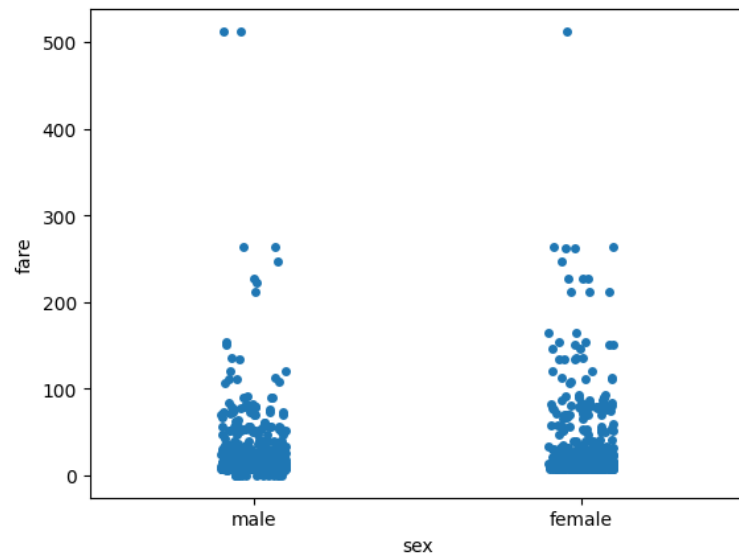
```
[ ]: <Axes: xlabel='sex', ylabel='fare'>
```



Strip Plot

```
[ ]: sns.stripplot(x='sex',y='fare',data=ds,jitter=True)
```

```
[ ]: <Axes: xlabel='sex', ylabel='fare'>
```



Swarm Plot

```
[ ]: sns.swarmplot(x='sex', y='age', data=ds)
```

```
[ ]: <Axes: xlabel='sex', ylabel='age'>
```