practical-8-piyusha-supe

April 12, 2025

0.0.1 Piyusha Supe 23CO315

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. 2. Write a code to check how the price of the ticket (column name: 'fare') for each passenger is distributed by plotting a histogram.

[2]:

```python
# 1. Downloading the Seaborn Library
import seaborn as sns
import matplotlib.pyplot as plt

# Set the visual theme
sns.set(style="darkgrid")

# 2. The Dataset
titanic = sns.load_dataset("titanic")
```

[3]:

```python
print(titanic.head())
print(titanic.tail())
print(titanic.info())
print(titanic.describe(include="all"))
print(titanic.shape)
print(titanic.size)
print(titanic.ndim)
print(titanic.columns)
```

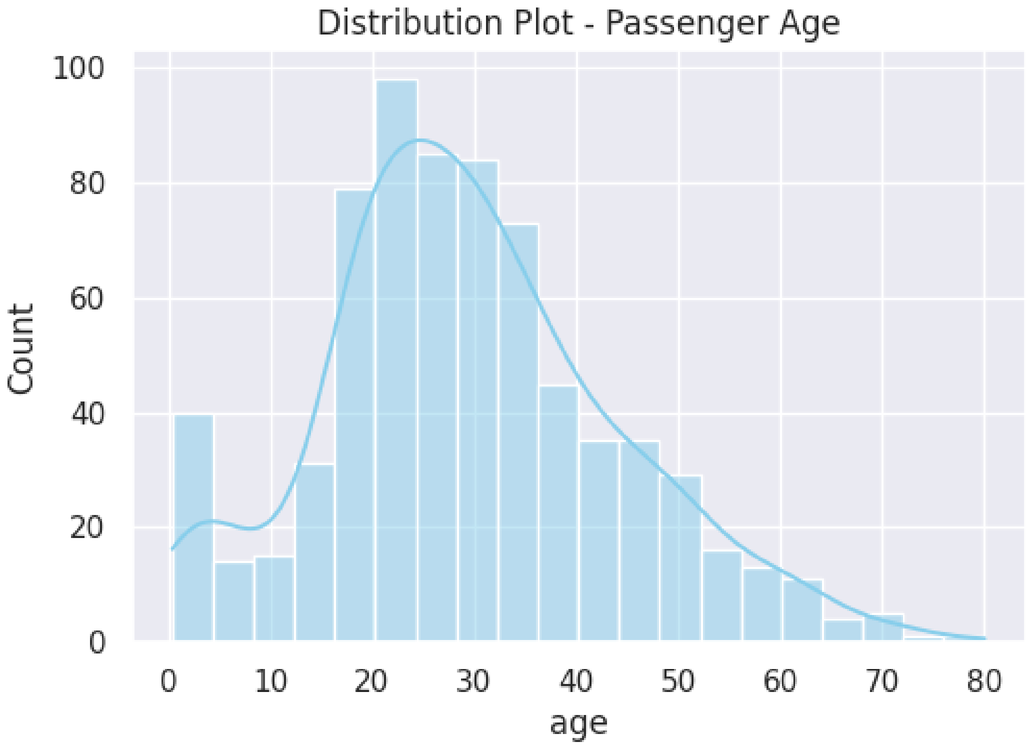
```
   survived  pclass     sex   age  sibsp  parch     fare embarked  class  \
0         0       3    male  22.0      1      0   7.2500        S  Third
1         1       1  female  38.0      1      0  71.2833        C  First
2         1       3  female  26.0      0      0   7.9250        S  Third
3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

```
     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
<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
None
          survived      pclass   sex         age       sibsp       parch  \
count   891.000000  891.000000   891  714.000000  891.000000  891.000000
unique         NaN         NaN     2         NaN         NaN         NaN
top            NaN         NaN  male         NaN         NaN         NaN
freq           NaN         NaN   577         NaN         NaN         NaN
mean      0.383838    2.308642   NaN   29.699118    0.523008    0.381594
std       0.486592    0.836071   NaN   14.526497    1.102743    0.806057
min       0.000000    1.000000   NaN    0.420000    0.000000    0.000000
25%       0.000000    2.000000   NaN   20.125000    0.000000    0.000000
50%       0.000000    3.000000   NaN   28.000000    0.000000    0.000000
75%       1.000000    3.000000   NaN   38.000000    1.000000    0.000000
max       1.000000    3.000000   NaN   80.000000    8.000000    6.000000
```

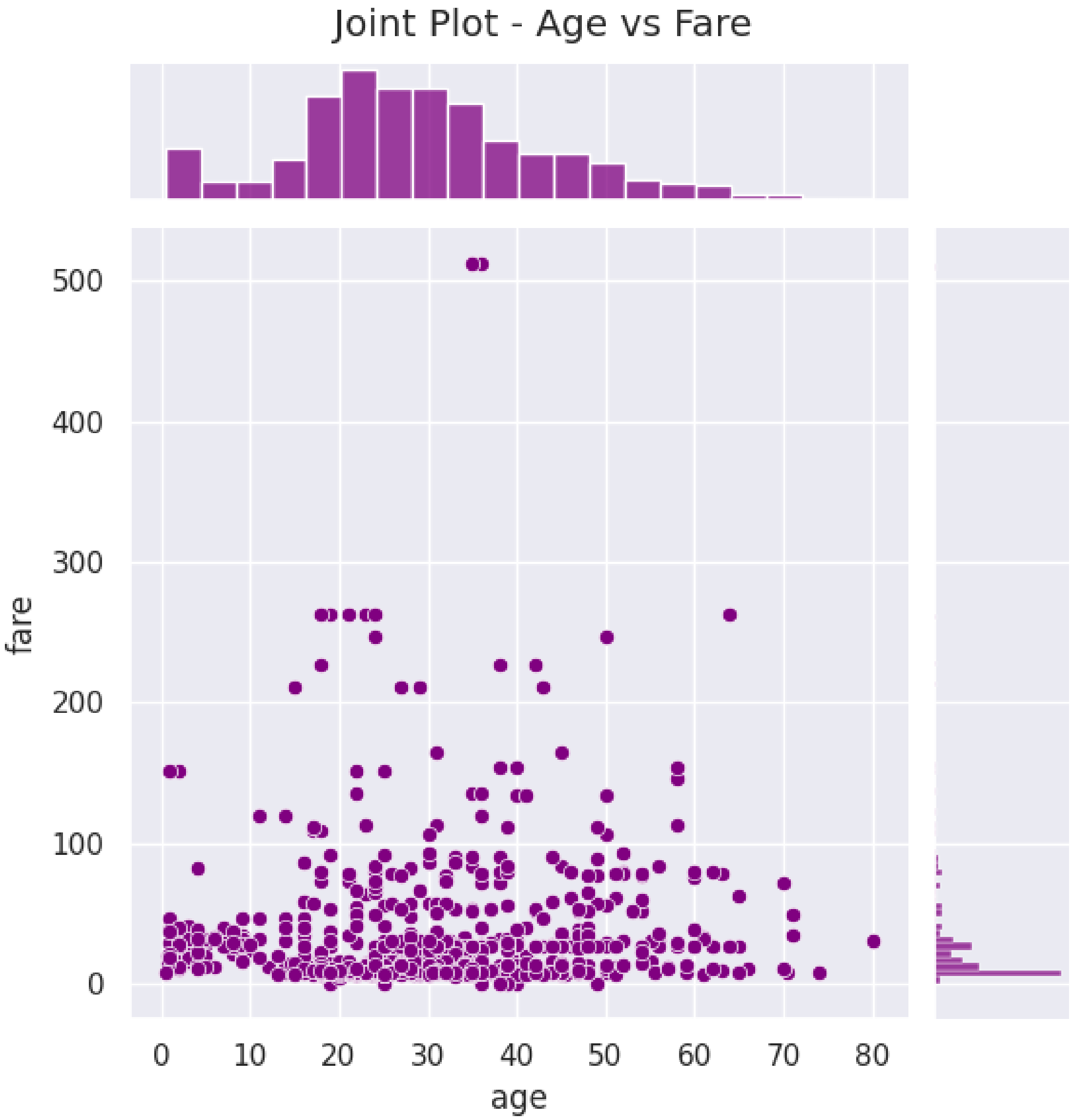
```
              fare embarked  class  who adult_male deck  embark_town alive  \
count   891.000000      889    891  891        891  203          889   891
unique         NaN        3      3    3          2    7            3     2
top            NaN        S  Third  man       True    C  Southampton    no
freq           NaN      644    491  537        537   59          644   549
mean     32.204208      NaN    NaN  NaN        NaN  NaN          NaN   NaN
std      49.693429      NaN    NaN  NaN        NaN  NaN          NaN   NaN
min       0.000000      NaN    NaN  NaN        NaN  NaN          NaN   NaN
25%       7.910400      NaN    NaN  NaN        NaN  NaN          NaN   NaN
50%      14.454200      NaN    NaN  NaN        NaN  NaN          NaN   NaN
75%      31.000000      NaN    NaN  NaN        NaN  NaN          NaN   NaN
max     512.329200      NaN    NaN  NaN        NaN  NaN          NaN   NaN

       alone
count    891
unique     2
top     True
freq     537
mean     NaN
std      NaN
min      NaN
25%      NaN
50%      NaN
75%      NaN
max      NaN
(891, 15)
13365
2
Index(['survived', 'pclass', 'sex', 'age', 'sibsp', 'parch', 'fare',
       'embarked', 'class', 'who', 'adult_male', 'deck', 'embark_town',
       'alive', 'alone'],
      dtype='object')
```

[4]:

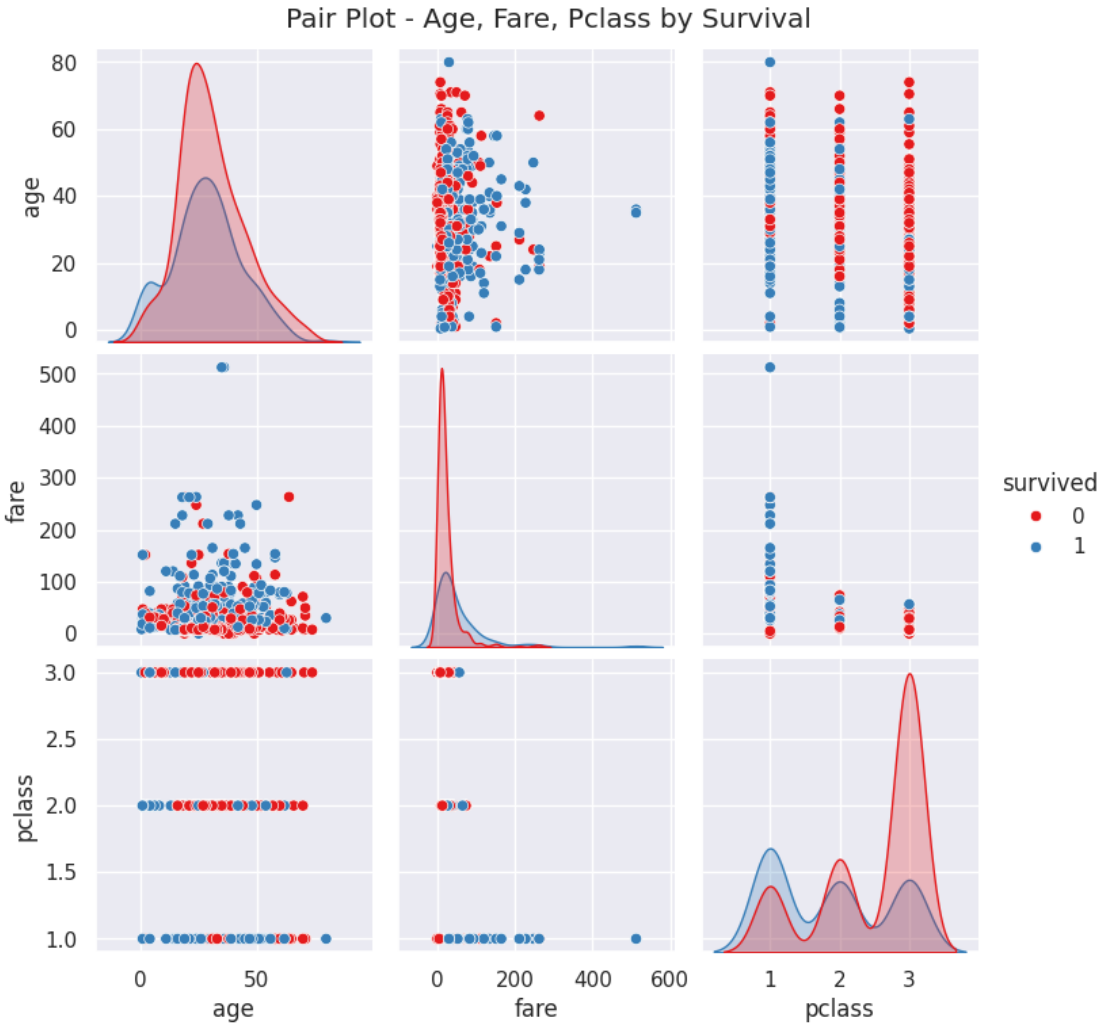
```python
# 3. Distributional Plots

# 3.1 The Dist Plot - Distribution of Passenger Ages
plt.figure(figsize=(6, 4))
sns.histplot(titanic['age'].dropna(), kde=True, color='skyblue')
plt.title("Distribution Plot - Passenger Age")
plt.show()
```



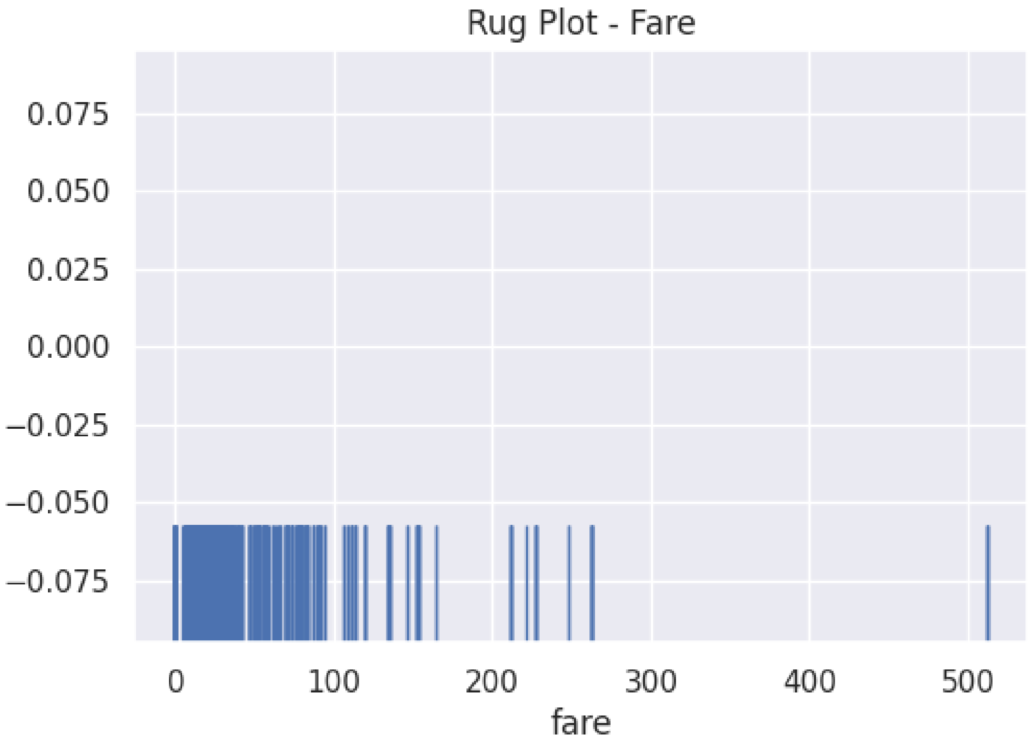
[5]:
```
# 3.2 The Joint Plot - Age vs Fare
sns.jointplot(x='age', y='fare', data=titanic, kind='scatter', color='purple')
plt.suptitle("Joint Plot - Age vs Fare", y=1.02)
plt.show()
```



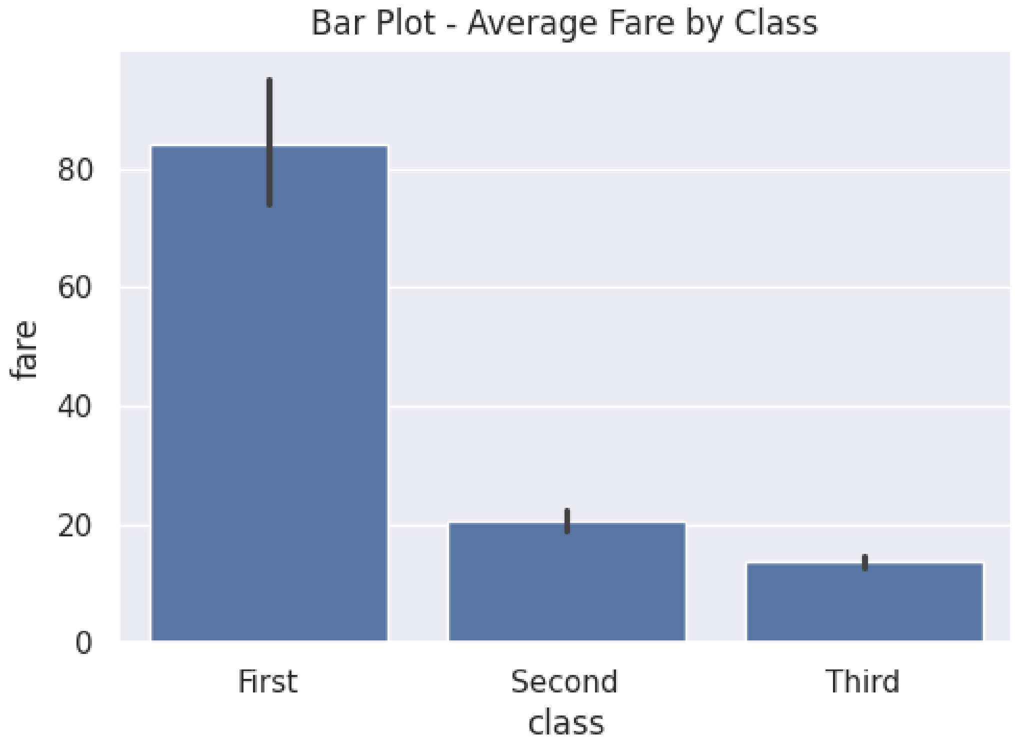
```
[6]: # 3.3 The Pair Plot - Age, Fare, and Pclass (with survival hue)
     sns.pairplot(titanic[['age', 'fare', 'pclass', 'survived']].dropna(),␣
      ↪hue='survived', palette='Set1')
     plt.suptitle("Pair Plot - Age, Fare, Pclass by Survival", y=1.02)
     plt.show()
```



```
[7]: # 3.4 The Rug Plot - Fare
     plt.figure(figsize=(6, 4))
     sns.rugplot(titanic['fare'].dropna(), height=0.2)
     plt.title("Rug Plot - Fare")
     plt.show()
```



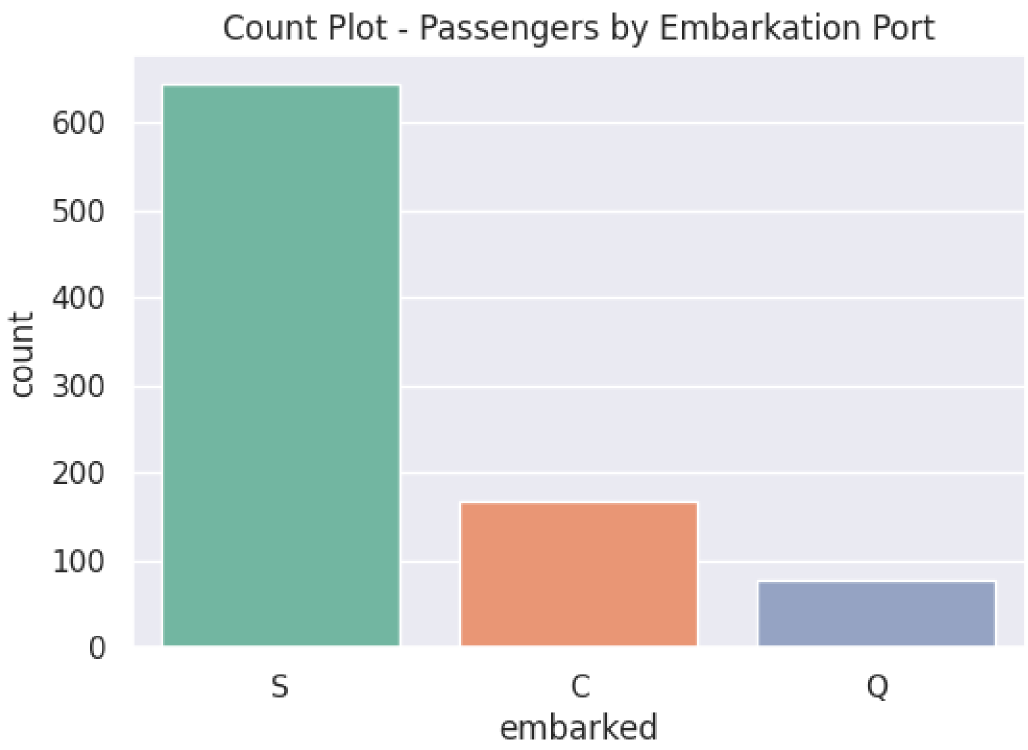
[8]:
```
# 4. Categorical Plots

# 4.1 The Bar Plot - Average Fare by Class
plt.figure(figsize=(6, 4))
sns.barplot(x='class', y='fare', data=titanic)
plt.title("Bar Plot - Average Fare by Class")
plt.show()
```



[9]:

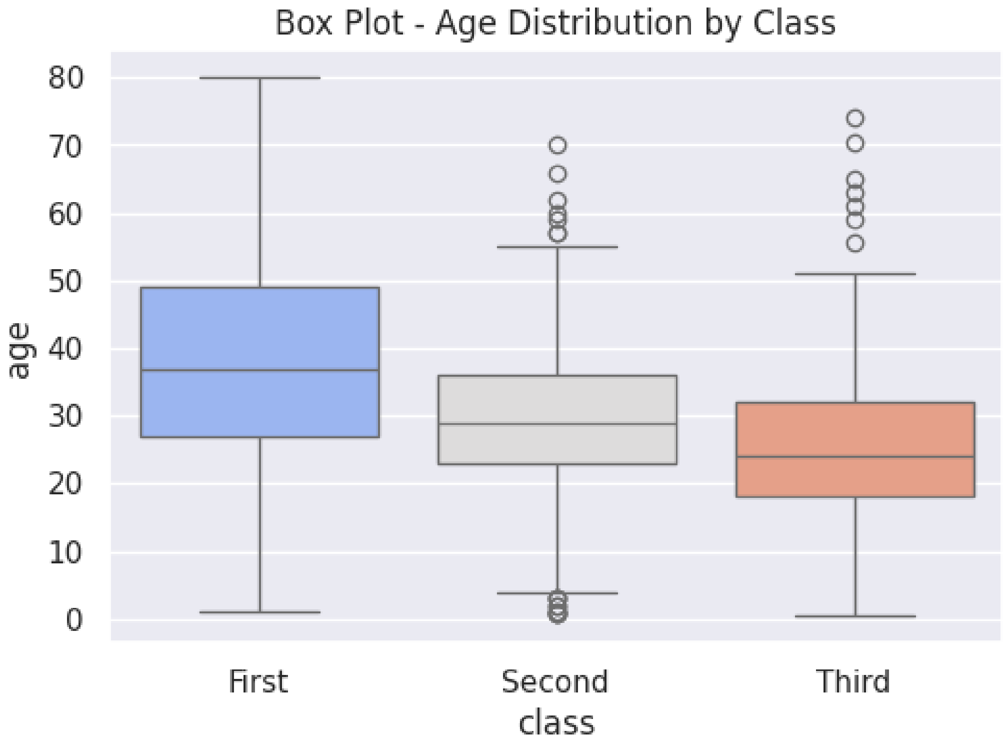
```
# 4.2 The Count Plot - Number of Passengers by Embarkation Port
plt.figure(figsize=(6, 4))
sns.countplot(x='embarked', data=titanic, palette='Set2')
plt.title("Count Plot - Passengers by Embarkation Port")
plt.show()
```

```
<ipython-input-9-44d96e600459>:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.countplot(x='embarked', data=titanic, palette='Set2')
```



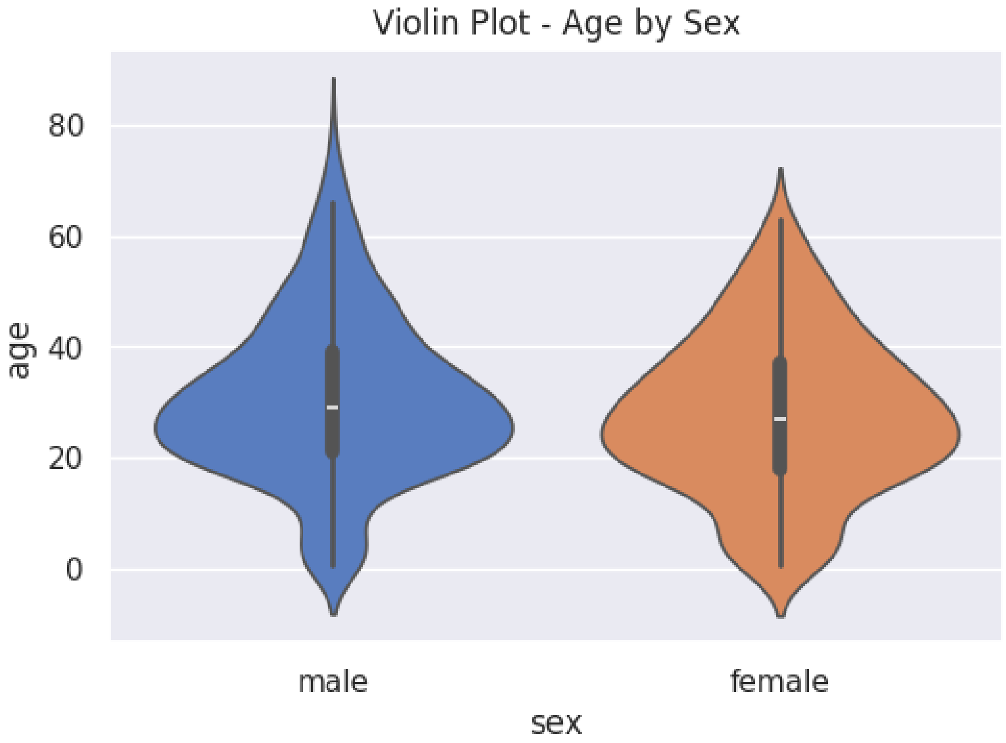
```
[10]: # 4.3 The Box Plot - Age by Class
      plt.figure(figsize=(6, 4))
      sns.boxplot(x='class', y='age', data=titanic, palette='coolwarm')
      plt.title("Box Plot - Age Distribution by Class")
      plt.show()
```

```
<ipython-input-10-b6498d65a393>:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.boxplot(x='class', y='age', data=titanic, palette='coolwarm')
```

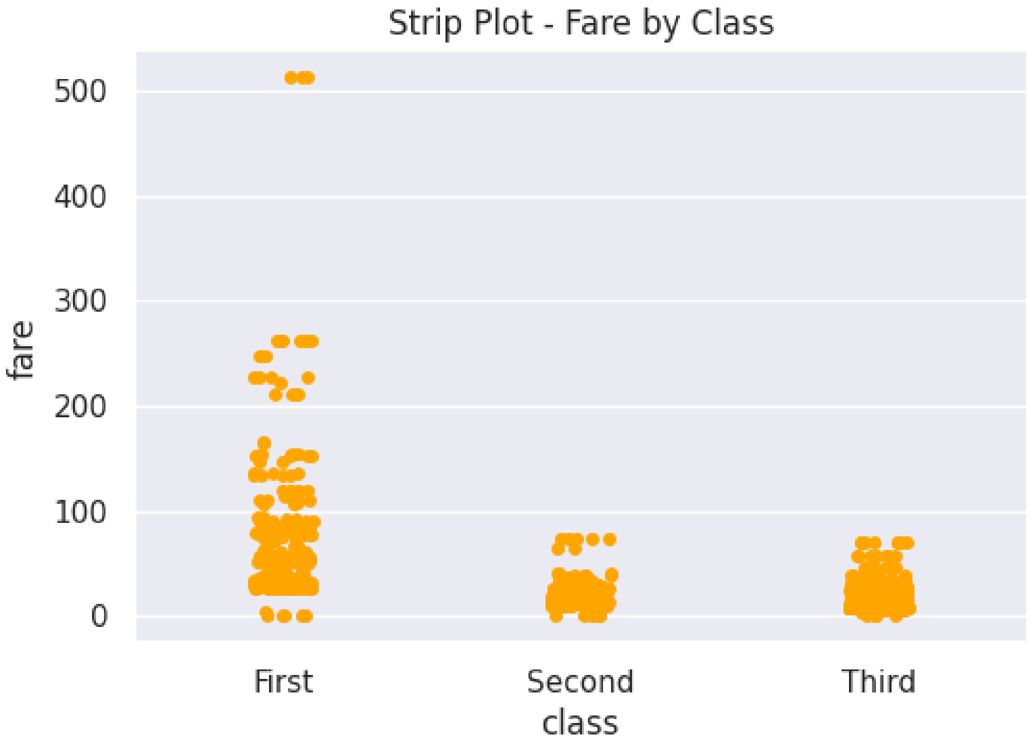


[11]:
```
# 4.4 The Violin Plot - Age by Sex
plt.figure(figsize=(6, 4))
sns.violinplot(x='sex', y='age', data=titanic, palette='muted')
plt.title("Violin Plot - Age by Sex")
plt.show()
```

```
<ipython-input-11-f55bb46b124b>:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.violinplot(x='sex', y='age', data=titanic, palette='muted')
```



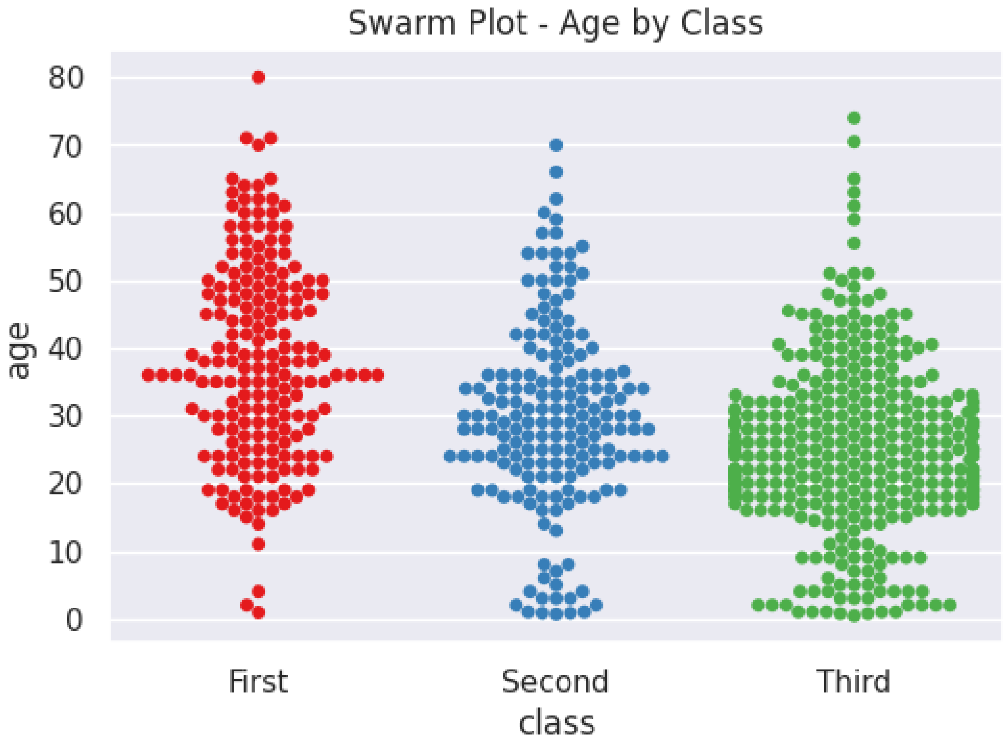
[12]:
```
# 4.5 The Strip Plot - Fare by Class
plt.figure(figsize=(6, 4))
sns.stripplot(x='class', y='fare', data=titanic, jitter=True, color='orange')
plt.title("Strip Plot - Fare by Class")
plt.show()
```



[13]:

```
# 4.6 The Swarm Plot - Age by Class
plt.figure(figsize=(6, 4))
sns.swarmplot(x='class', y='age', data=titanic, palette='Set1')
plt.title("Swarm Plot - Age by Class")
plt.show()
```

```
<ipython-input-13-8b4547937160>:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.swarmplot(x='class', y='age', data=titanic, palette='Set1')
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399:
UserWarning: 6.5% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399:
UserWarning: 16.9% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
```

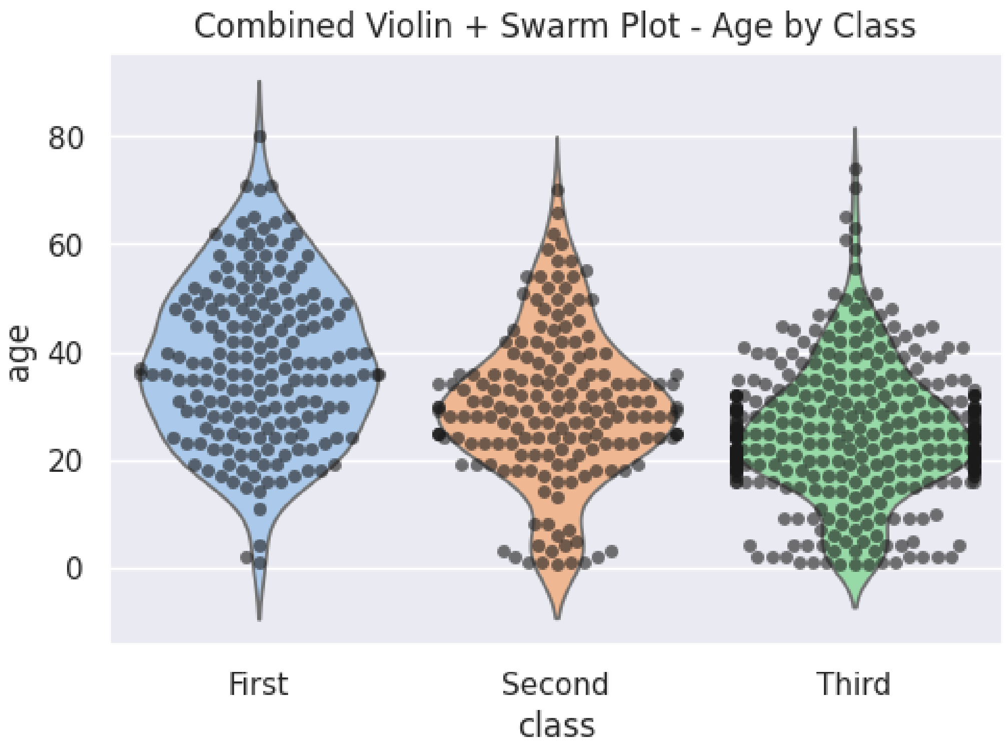


[14]:

```
# 5. Combining Swarm and Violin Plots - Age by Class
plt.figure(figsize=(6, 4))
sns.violinplot(x='class', y='age', data=titanic, inner=None, palette='pastel')
sns.swarmplot(x='class', y='age', data=titanic, color='k', alpha=0.6)
plt.title("Combined Violin + Swarm Plot - Age by Class")
plt.show()
```

```
<ipython-input-14-c624c970ec7d>:3: FutureWarning:

Passing `palette` without assigning `hue` is deprecated and will be removed in
v0.14.0. Assign the `x` variable to `hue` and set `legend=False` for the same
effect.

  sns.violinplot(x='class', y='age', data=titanic, inner=None, palette='pastel')
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399:
UserWarning: 8.7% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399:
UserWarning: 31.8% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
/usr/local/lib/python3.11/dist-packages/seaborn/categorical.py:3399:
```

```
UserWarning: 7.5% of the points cannot be placed; you may want to decrease the
size of the markers or use stripplot.
  warnings.warn(msg, UserWarning)
```



PIYUSHA SUPE 23CO315