p7

March 30, 2024

[4]:
```
import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns

df = sns.load_dataset("titanic")
print(df.head())
print("\n\n", df.dtypes)
```

```
   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          int64
pclass            int64
sex              object
age             float64
sibsp             int64
parch             int64
fare            float64
embarked         object
class          category
who              object
adult_male         bool
deck           category
embark_town      object
alive            object
```

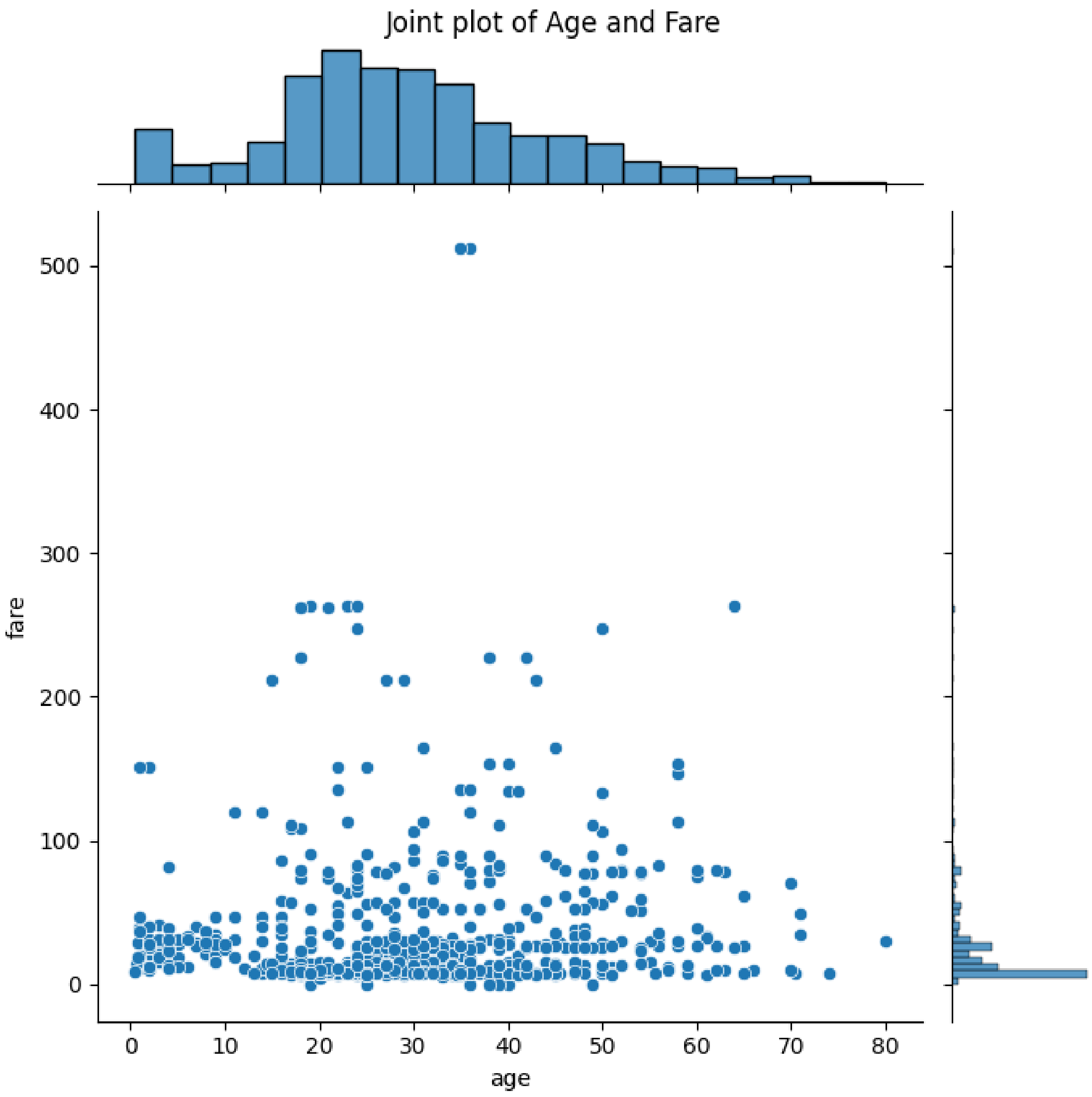
```
alone                bool
dtype: object
```

[6]:
```
df=df[['survived','class','sex','age','fare']]
print(df.head())
print("\n\n", df.dtypes)
```

```
   survived  class     sex   age     fare
0         0  Third    male  22.0   7.2500
1         1  First  female  38.0  71.2833
2         1  Third  female  26.0   7.9250
3         1  First  female  35.0  53.1000
4         0  Third    male  35.0   8.0500


 survived       int64
class       category
sex           object
age          float64
fare         float64
dtype: object
```

[9]:
```
plt.figure(figsize=(10, 6))

# Create a joint plot of 'Age' and 'Fare' with additional histograms and kde␣
 ↪plots
g = sns.jointplot(data=df, x='age', y='fare', kind='scatter', height=7)

# Set the title of the joint plot
g.fig.suptitle('Joint plot of Age and Fare')

# Adjust the position of the title
g.fig.subplots_adjust(top=0.95)

# Display the joint plot
plt.show()
```

```
<Figure size 1000x600 with 0 Axes>
```

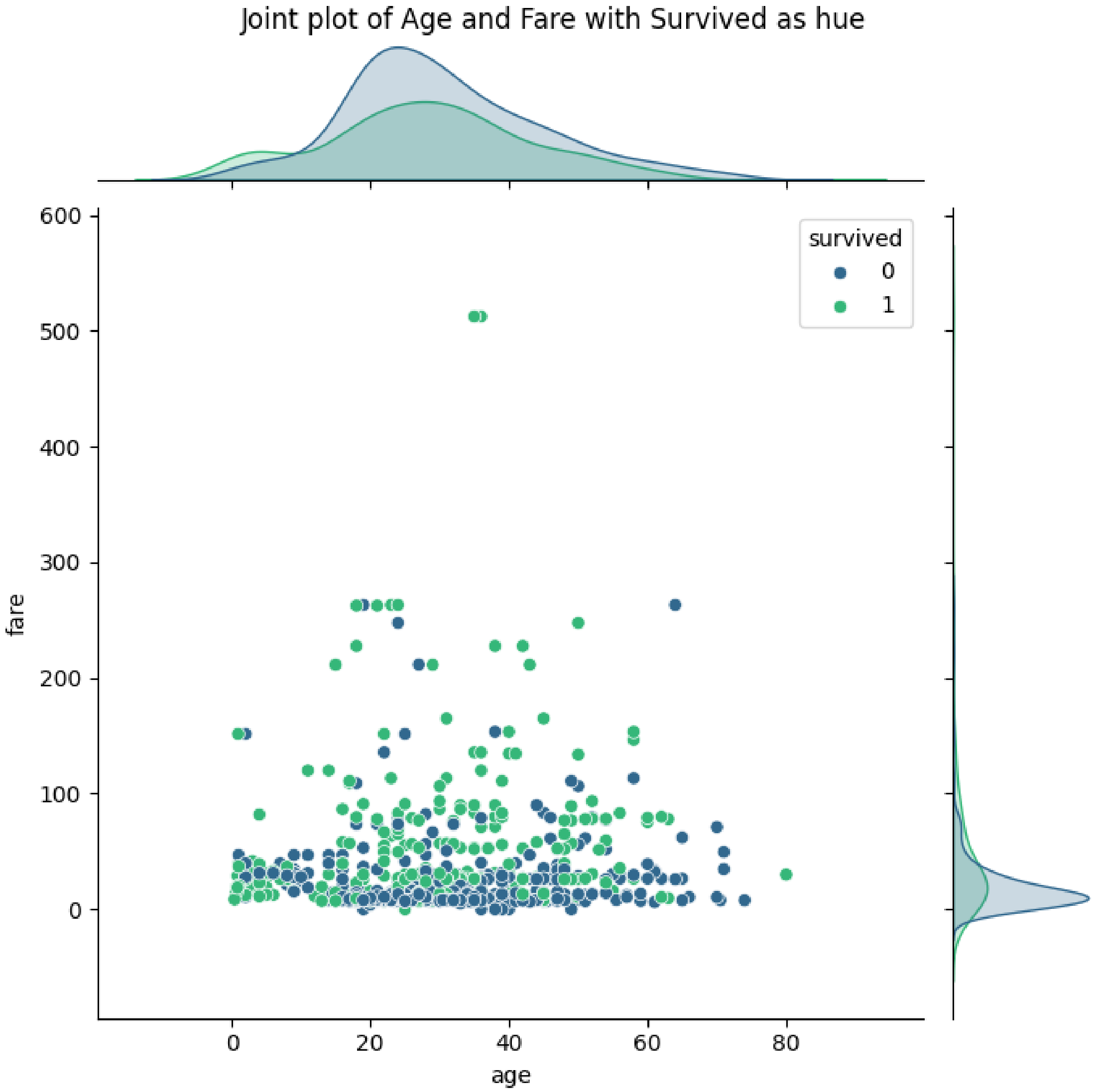


[10]:

```
# Create a new figure and a set of subplots
plt.figure(figsize=(10, 6))

# Create a joint plot of 'Age' and 'Fare' with additional histograms and kde plots
g = sns.jointplot(data=df, x='age', y='fare', hue='survived', palette='viridis', height=7)

# Set the title of the joint plot
g.fig.suptitle('Joint plot of Age and Fare with Survived as hue')

# Adjust the position of the title
g.fig.subplots_adjust(top=0.95)
```

```
# Display the joint plot
plt.show()
```

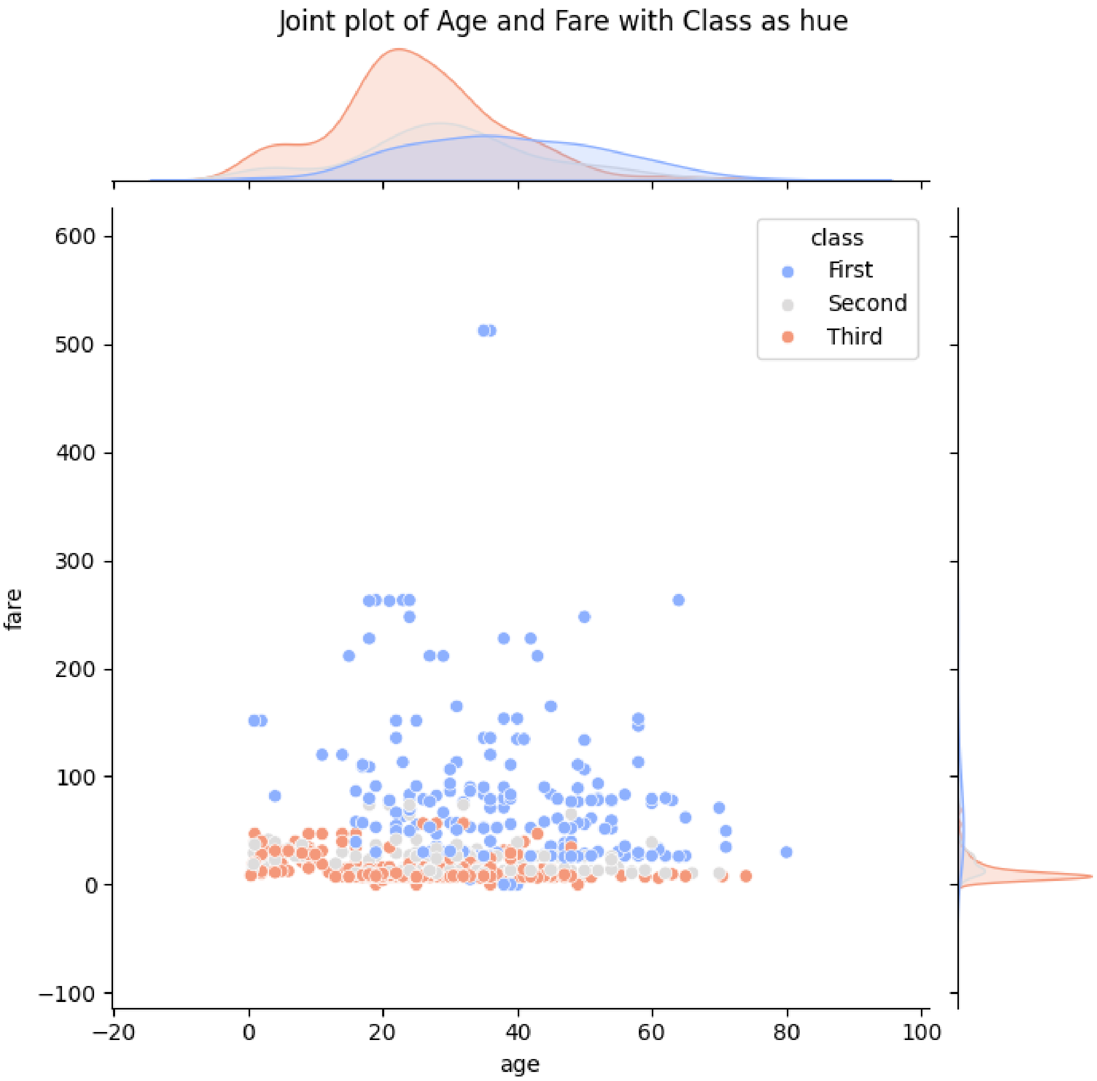
```
<Figure size 1000x600 with 0 Axes>
```



```
[11]: # Create a new figure and a set of subplots
plt.figure(figsize=(10, 6))

# Create a joint plot of 'Age' and 'Fare' with additional histograms and kde␣
 ↪plots
g = sns.jointplot(data=df, x='age', y='fare', hue='class', palette='coolwarm',␣
 ↪height=7)
```

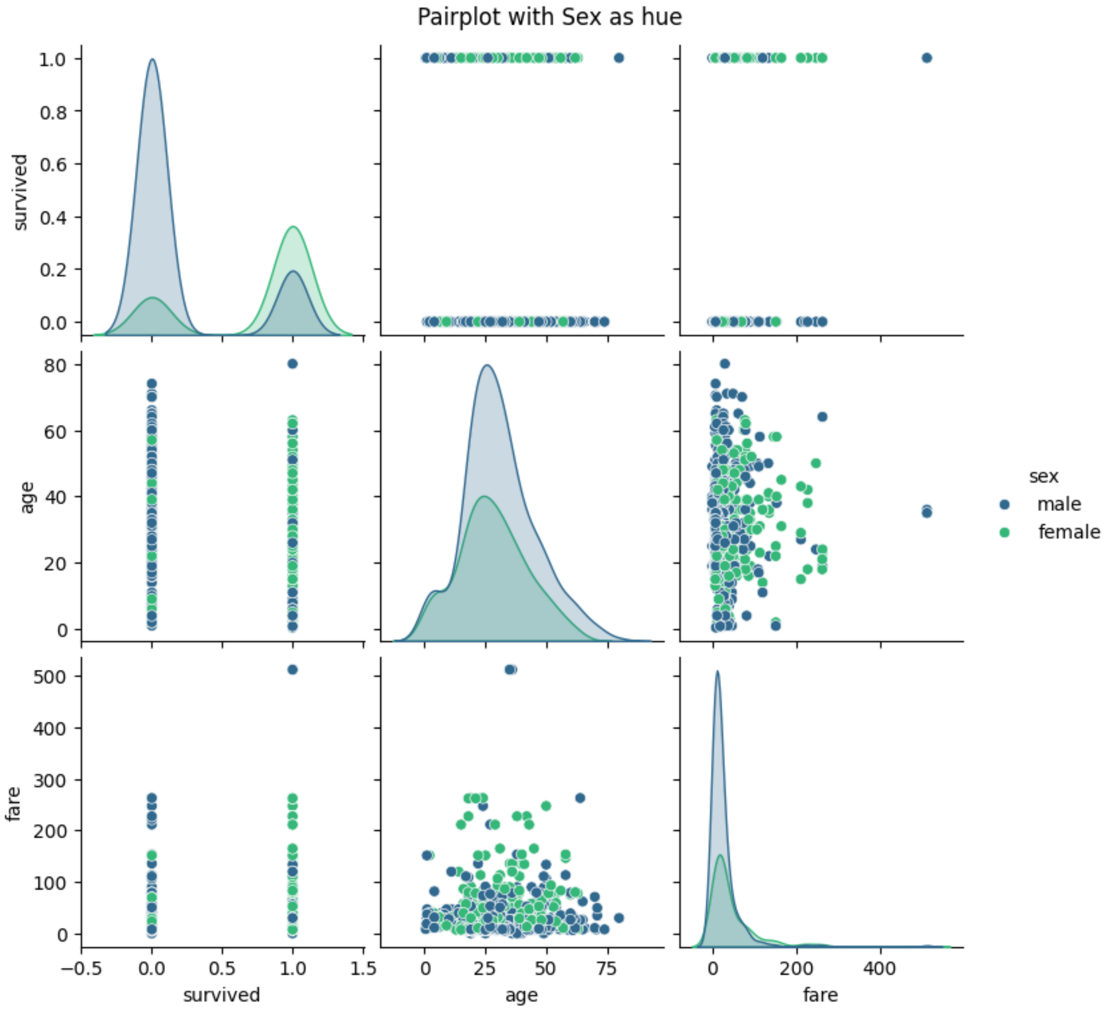
```
# Set the title of the joint plot
g.fig.suptitle('Joint plot of Age and Fare with Class as hue')

# Adjust the position of the title
g.fig.subplots_adjust(top=0.95)

# Display the joint plot
plt.show()
```

```
<Figure size 1000x600 with 0 Axes>
```


Joint plot of Age and Fare with Class as hue

[13]:

```
g = sns.pairplot(df, hue='sex', palette='viridis')

# Set the title of the pairplot
g.fig.suptitle('Pairplot with Sex as hue', y=1.02)

# Display the pairplot
plt.show()
```

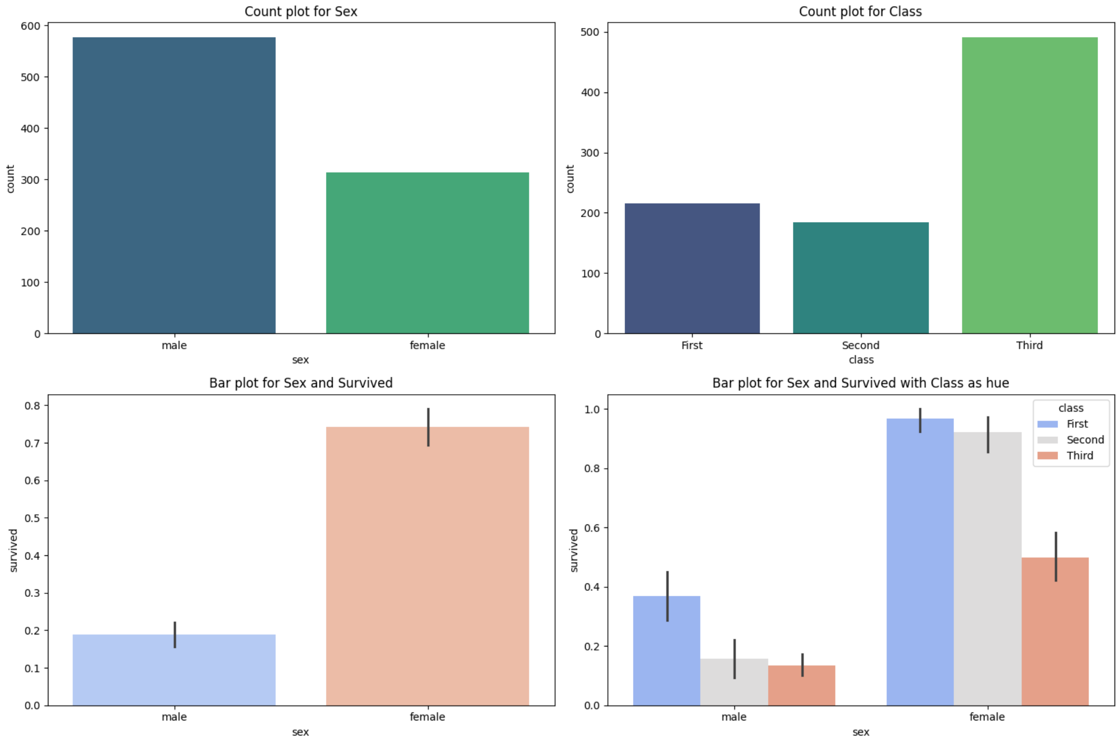


[14]:

```
# Create a new figure and a set of subplots
fig, axs = plt.subplots(2, 2, figsize=(15, 10))

# Create a count plot for 'sex'
sns.countplot(x=df['sex'], palette='viridis', ax=axs[0, 0])
axs[0, 0].set_title('Count plot for Sex')

# Create a count plot for 'class'
```

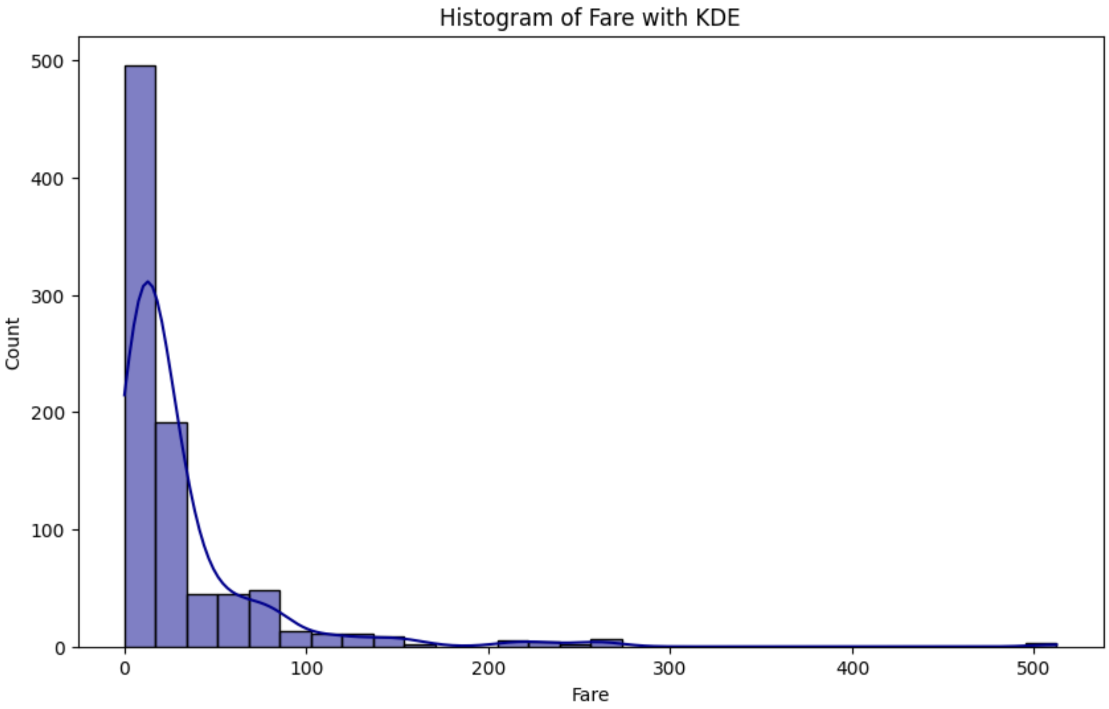
```python
sns.countplot(x=df['class'], palette='viridis', ax=axs[0, 1])
axs[0, 1].set_title('Count plot for Class')

# Create a bar plot for 'sex' and 'survived'
sns.barplot(x='sex', y='survived', data=df, palette='coolwarm', ax=axs[1, 0])
axs[1, 0].set_title('Bar plot for Sex and Survived')

# Create a bar plot for 'sex' and 'survived' with hue as 'class'
sns.barplot(x='sex', y='survived', hue='class', data=df, palette='coolwarm',␣
 ↪ax=axs[1, 1])
axs[1, 1].set_title('Bar plot for Sex and Survived with Class as hue')

# Automatically adjust subplot params so that the subplot fits into the figure␣
 ↪area
plt.tight_layout()

# Display the figure
plt.show()
```

```
<ipython-input-14-9eaf806db171>:5: 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=df['sex'], palette='viridis', ax=axs[0, 0])
<ipython-input-14-9eaf806db171>:9: 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=df['class'], palette='viridis', ax=axs[0, 1])
<ipython-input-14-9eaf806db171>:13: 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.barplot(x='sex', y='survived', data=df, palette='coolwarm', ax=axs[1, 0])
```



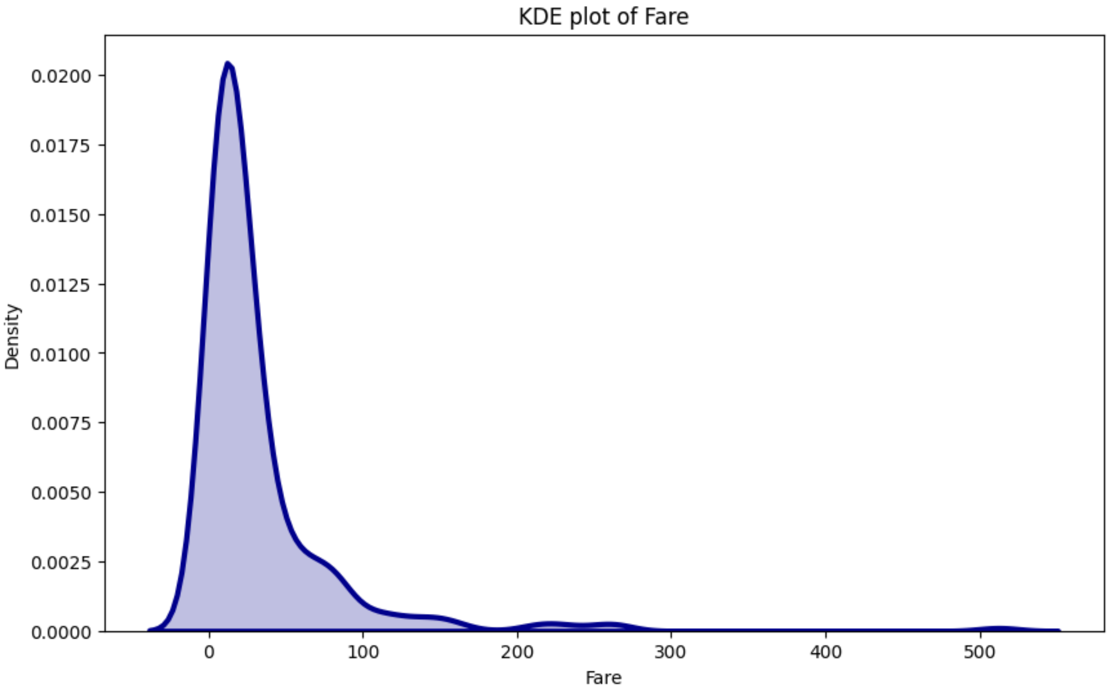
[15]:

```
plt.figure(figsize=(10, 6))

# Create a histogram for 'fare' with additional kde plot and rug plot
sns.histplot(data=df, x='fare', kde=True, color='darkblue', bins=30)

# Set the title of the histogram
plt.title('Histogram of Fare with KDE')

# Set the x and y axis labels
plt.xlabel('Fare')
plt.ylabel('Count')

# Display the histogram
plt.show()
```



[22]:

```
plt.figure(figsize=(10, 6))

# Create a KDE plot for 'fare' with additional shading and a custom color
sns.kdeplot(data=df, x='fare', fill=True, color='darkblue', linewidth=3)

# Set the title of the KDE plot
plt.title('KDE plot of Fare')

# Set the x and y axis labels
plt.xlabel('Fare')
plt.ylabel('Density')

# Display the KDE plot
plt.show()
```


KDE plot of Fare
Density
0.0000
0.0025
0.0050
0.0075
0.0100
0.0125
0.0150
0.0175
0.0200
0
100
200
300
400
500
Fare

```
[ ]:
```