CodeTantra

5.1.1. Stacked Plot

05:36

Create a stacked area plot to visualize the temperature variations for three different cities (City A, City B, and City C) across the months of the year. The temperature data is provided for each city in the editor.

Your task is to:

- Create a stacked area plot using the data.
- Label the x-axis as "Month", the y-axis as "Temperature", and provide the title "Temperature Variation" for the plot.
- Display the plot showing the temperature variation for each city throughout the months of the year.

Sample Test Cases +

Explorer

stackedplo...

Submit

Debugger

```
1  import matplotlib.pyplot as plt
2  import pandas as pd
3
4  # Data for Months and Temperature
   for three cities
5  data = {
6      'Month': ['January', 'February',
   'March', 'April', 'May', 'June',
   'July', 'August', 'September',
   'October', 'November', 'December'],
7      'City_A_Temperature': [5, 7, 10,
   13, 17, 20, 22, 21, 18, 12, 8, 6],
8      'City_B_Temperature': [2, 3, 5,
   6, 10, 14, 16, 17, 12, 9, 5, 3],
9      'City_C_Temperature': [3, 4, 6,
   8, 9, 12, 15, 14, 10, 7, 4, 2]
10 }
11 df = pd.DataFrame(data)
12 plt.stackplot(df['Month'],df['City_A_
   Temperature'],df['City_B_Temperature'
   ], df['City_C_Temperature'])
13 plt.title('Temperature Variation')
14 plt.xlabel('Month')
15 plt.ylabel('Temperature')
16 plt.show()
17 # Write your code...
18
```

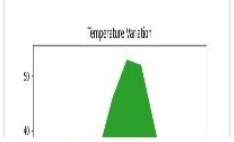
Average time
0.989 s
989.00 ms

Maximum time
0.989 s
989.00 ms

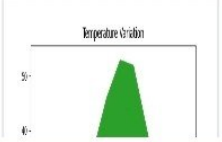
1 out of 1 shown test case(s) passed

Test case 1 989 ms Debug

Expected output



Actual output



‹ Prev Reset Submit Next ›

5.2.1. Titanic Dataset

39:47

Write a Python program to analyze and visualize data from the Titanic dataset based on the following instructions:

Dataset Information:

The dataset is stored in a CSV file named `titanic.csv` and has been loaded using the `pandas` library. It contains the following columns:

- `Pclass`: Passenger class (1 = First, 2 = Second, 3 = Third).
- `Gender`: Gender of the passenger (male/female).
- `Age`: Age of the passenger.
- `Survived`: Survival status (0 = Did not survive, 1 = Survived).
- `Fare`: Ticket fare paid by the passenger.

Visualization:

To represent these trends, you will create 5 visualizations using Matplotlib. The visualizations should be arranged in a 3x2 grid (3 rows and 2 columns).

Visualization Details:

Write the code to create a series of visualizations as follows:

Bar Plot (Pclass Distribution):

- Create a bar plot to show the distribution of passengers across the different passenger classes (`Pclass`).
- Use the color `skyblue` for the bars.
- Title the plot as "**Passenger Class Distribution**".
- Label the x-axis as "**Pclass**" and the y-axis as "**Count**".

Pie Chart (Gender Distribution):

- Create a pie chart to display the distribution of male and female passengers.
- Use `lightblue` for males and `lightcoral` for females.
- Include percentages on the slices (use `autopct='%1.1f%%'`).
- Title the plot as "**Gender Distribution**".

Histogram (Age Distribution):

- Create a histogram to visualize the distribution of passengers' ages.
- Use `lightgreen` for the bars with black edges (`edgecolor = 'black'`).
- Set the number of bins to **8** for the histogram.
- Title the plot as "**Age Distribution**".
- Label the x-axis as "**Age**" and the y-axis as "**Frequency**".

Bar Plot (Survival Count):

- Create a bar plot to show the count of passengers who survived and those who did not, based on the `Survived` column.
- Use the colors `lightblue` for survivors (1) and `lightcoral` for non-survivors (0).
- Title the plot as "**Survival Count**".
- Label the x-axis as "**Survived (0 = No, 1 = Yes)**" and the y-axis as "**Count**".

Scatter Plot (Fare vs Age):

- Create a scatter plot to visualize the relationship between the `Fare` and Age of passengers.
- Use orange for the data points.
- Title the plot as "**Fare vs Age**".
- Label the x-axis as "**Age**" and the y-axis as "**Fare**".

Note: Refer to the displayed plot in the sample test cases for better understanding.

Sample Test Cases +

Explorer

titanicData...

Submit

Debugger

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the Titanic dataset from the CSV file
df = pd.read_csv('titanic.csv')

# Set up the figure for 5 subplots
fig, axes = plt.subplots(3, 2, figsize=(12, 12))

# write the code..
#
df = pd.read_csv('titanic.csv')

# Set up the figure for 5 subplots
fig, axes = plt.subplots(3, 2, figsize=(12, 12))

# Plot 1: Count of passengers by class
axes[0, 0].bar(df['Pclass'].value_counts().index, df['Pclass'].value_counts(), color='skyblue')
axes[0, 0].set_title("Passenger Class Distribution")
axes[0, 0].set_xlabel("Pclass")
axes[0, 0].set_ylabel("Count")

# Plot 2: Gender distribution
axes[0, 1].pie(df['Gender'].value_counts(), labels=df['Gender'].value_counts().index, autopct='%1.1f%%', colors=['lightblue', 'lightcoral'])
axes[0, 1].set_title("Gender Distribution")

# Plot 3: Age distribution
axes[1, 0].hist(df['Age'].dropna(), bins=8, color='lightgreen', edgecolor='black')
axes[1, 0].set_title("Age Distribution")
axes[1, 0].set_xlabel("Age")
axes[1, 0].set_ylabel("Frequency")

# Plot 4: Survival count
axes[1, 1].bar(df['Survived'].value_counts().index, df['Survived'].value_counts(), color=['lightblue', 'lightcoral'])
axes[1, 1].set_title("Survival Count")
axes[1, 1].set_xlabel("Survived (0 = No, 1 = Yes)")
axes[1, 1].set_ylabel("Count")

# Plot 5: Fare vs Age
axes[2, 0].scatter(df['Age'], df['Fare'], color='orange', edgecolors='black')
axes[2, 0].set_title("Fare vs Age")
axes[2, 0].set_xlabel("Age")
axes[2, 0].set_ylabel("Fare")

plt.tight_layout()
plt.show()
```

5.2.2. Histogram of passenger information o... 01:36

Write a Python code to plot a histogram for the distribution of the 'Age' column from the Titanic dataset. The histogram should display the frequency of different age ranges with the following specifications:

1. Use **30 bins** for the histogram.
2. Set the **edge color** of the bars to **black** (k).
3. Label the x-axis as **'Age'** and the y-axis as **'Frequency'**.
4. Add the title **"Age Distribution"** to the histogram.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

Histogram...

Submit

Debugger

```
1  import pandas as pd
2  import matplotlib.pyplot as plt
3
4  # Load the Titanic dataset
5  data = pd.read_csv('Titanic-
   Dataset.csv')
6
7  # Data Cleaning
8  data['Age'].fillna(data['Age'].median
   (), inplace=True)
9  data['Embarked'].fillna(data['Embarke
   d'].mode()[0], inplace=True)
10 data.drop('Cabin', axis=1,
   inplace=True)
11
12 # Convert categorical features to
   numeric
13 data['Sex'] =
   data['Sex'].map({'male': 0,
   'female': 1})
14 data = pd.get_dummies(data, columns=
   ['Embarked'], drop_first=True)
15
16 # Write your code here for Histogram
17 # Write your code here for Histogram
18 plt.hist(data['Age'], bins=30,
   edgecolor='k')
19 plt.xlabel('Age')
20 plt.ylabel('Frequency')
21 plt.title('Age Distribution')
22 plt.show()
```

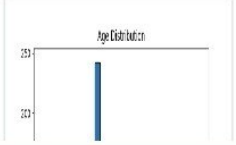
Average time
1.411 s
1411.00 ms

Maximum time
1.411 s
1411.00 ms

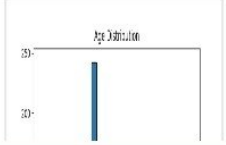
1 out of 1 shown test case(s) passed

Test case 1 1411 ms Debug

Expected output



Actual output



< Prev Reset Submit Next >

5.2.3. Bar plot of survival rate of passengers

01:21

Write a Python code to plot a bar chart that shows the count of passengers who survived and did not survive in the Titanic dataset. The chart should display the following specifications:

1. Use the '**Survived**' column to show the count of survivors (0 = Did not survive, 1 = Survived).
2. Set the chart type to '**bar**'.
3. Add the title "**Survival Count**" to the chart.
4. Label the x-axis as '**Survived**' and the y-axis as '**Count**'.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

BarPlotOfS...

Submit

Debugger

```
1  import pandas as pd
2  import matplotlib.pyplot as plt
3
4  # Load the Titanic dataset
5  data = pd.read_csv('Titanic-
   Dataset.csv')
6
7  # Data Cleaning
8  data['Age'].fillna(data['Age'].median
   (), inplace=True)
9  data['Embarked'].fillna(data['Embarke
   d'].mode()[0], inplace=True)
10 data.drop('Cabin', axis=1,
   inplace=True)
11
12 # Convert categorical features to
   numeric
13 data['Sex'] =
   data['Sex'].map({'male': 0,
   'female': 1})
14 data = pd.get_dummies(data, columns=
   ['Embarked'], drop_first=True)
15 survival_counts =
   data['Survived'].value_counts()
16 survival_counts.plot(kind='bar')
17 plt.title('Survival Count')
18 plt.xlabel('Survived')
19 plt.ylabel('Count')
20 plt.show()
21 # Write your code here for Bar Plot
   for Survival Rate
22
23
```

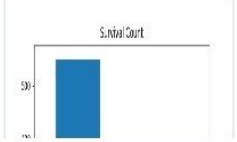
Average time
1.440 s
1440.00 ms

Maximum time
1.440 s
1440.00 ms

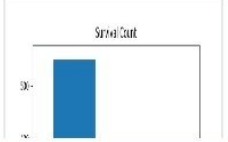
1 out of 1 shown test case(s) passed

Test case 1 1440 ms Debug

Expected output



Actual output



‹ Prev Reset Submit Next ›

5.2.4. Bar Plot for Survival by Gender

02:50

Write a Python code to plot a stacked bar chart that shows the count of passengers who survived and did not survive, grouped by gender, in the Titanic dataset. The chart should display the following specifications:

1. Group the data by the **'Sex'** column, then use the **value_counts()** function to count the occurrences of survivors (0 = Did not survive, 1 = Survived) for each gender.
2. Use a **stacked bar chart** to display the survival counts.
3. Add the title "**Survival by Gender**" to the chart.
4. Label the x-axis as **'Gender'** and the y-axis as **'Count'**.
5. The legend should indicate **'Not Survived'** and **'Survived'**.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

BarPlotOfS...

Submit

Debugger

```
1  import pandas as pd
2  import matplotlib.pyplot as plt
3
4  # Load the Titanic dataset
5  data = pd.read_csv('Titanic-
   Dataset.csv')
6
7  # Data Cleaning
8  data['Age'].fillna(data['Age'].median
   (), inplace=True)
9  data['Embarked'].fillna(data['Embarke
   d'].mode()[0], inplace=True)
10 data.drop('Cabin', axis=1,
   inplace=True)
11
12 # Convert categorical features to
   numeric
13 data['Sex'] =
   data['Sex'].map({'male': 0,
   'female': 1})
14 data = pd.get_dummies(data, columns=
   ['Embarked'], drop_first=True)
15
16
17 survival_by_gender =
   data.groupby('Sex')
   ['Survived'].value_counts().unstack()
   .fillna(0)
18 survival_by_gender.columns = ['Not
   Survived', 'Survived']
19 survival_by_gender.index = ['0', '1']
20 survival_by_gender.plot(kind='bar',
   stacked=True)
21 plt.title('Survival by Gender')
22 plt.xlabel('Gender')
23 plt.xlabel('Count')
24 plt.ylabel('Count')
25 plt.legend(title=None)
26 plt.show()
```

Average time
1.616 s
1616.00 ms

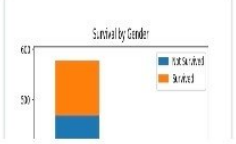
Maximum time
1.616 s
1616.00 ms

1 out of 1 shown test case(s) passed

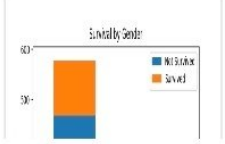
Test case 1 1616 ms

Debug

Expected output



Actual output



‹ Prev Reset Submit Next ›

5.2.5. Bar Plot for Survival by Pclass

10:58

Write a Python code to plot a stacked bar chart that shows the count of passengers who survived and did not survive, grouped by passenger class (**Pclass**), in the Titanic dataset. The chart should display the following specifications:

1. Group the data by the **Pclass** column and count the number of survivors (0 = Did not survive, 1 = Survived) for each class using **value_counts()**.
2. Use a **stacked bar chart** to display the survival counts.
3. Add the title "**Survival by Pclass**" to the chart.
4. Label the x-axis as '**Pclass**' and the y-axis as '**Count**'.
5. The legend should indicate '**Not Survived**' and '**Survived**'.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note:

- Refer to the visible test case for better reference.
- Ensure you use the **groupby()** function with **value_counts()** to count the survivors and non-survivors for each **Pclass**.
- Do **not** manually use **size()** or **unstack()** without **value_counts()**. Use the **value_counts()** method for counting survival status directly.

Sample Test Cases +

Explorer

BarPlotOfS...

Submit

Debugger

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the Titanic dataset
data = pd.read_csv('Titanic-Dataset.csv')

# Data Cleaning
data['Age'].fillna(data['Age'].median(), inplace=True)
data['Embarked'].fillna(data['Embarked'].mode()[0], inplace=True)
data.drop('Cabin', axis=1, inplace=True)

# Convert categorical features to numeric
data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
data = pd.get_dummies(data, columns=['Embarked'], drop_first=True)

# Write your code here for Bar Plot for Survival by Pclass

survival_by_class = data.groupby('Pclass')['Survived'].value_counts().unstack().fillna(0)
survival_by_class.columns = ['Not Survived', 'Survived']
survival_by_class.plot(kind='bar', stacked=True)
plt.title('Survival by Pclass')
plt.xlabel('Pclass')
plt.ylabel('Count')
plt.legend(title=None)
plt.show()

```

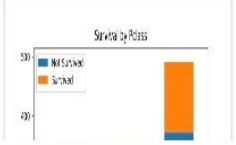
Average time
1.631 s
1631.00 ms

Maximum time
1.631 s
1631.00 ms

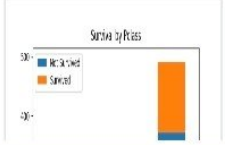
1 out of 1 shown test case(s) passed

Test case 1 1631 ms Debug

Expected output



Actual output



< Prev Reset Submit Next >

5.2.6. Bar Plot for Survival by Embarked

03:37

Write a Python code to plot a stacked bar chart showing the survival count for passengers based on their embarkation location in the Titanic dataset.
The chart should display the following specifications:

1. Use the **Embarked** column to determine the embarkation location. After converting this column into dummy variables (using **pd.get_dummies()**), plot the survival count based on the **Embarked_Q** column (representing passengers who embarked from Queenstown) in relation to survival.
2. Set the chart type to 'bar' and make it stacked.
3. Add the title "**Survival by Embarked** " to the chart.
4. Label the x-axis as **'Embarked'** and the y-axis as **'Count'**.
5. Include a legend to distinguish between survivors and non-survivors (label the legend as **'Survived'** and **'Not Survived'**).

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

BarPlotOfS...

Submit

Debugger

```
1  import pandas as pd
2  import matplotlib.pyplot as plt
3
4  # Load the Titanic dataset
5  data = pd.read_csv('Titanic-
   Dataset.csv')
6
7  # Data Cleaning
8  data['Age'].fillna(data['Age'].median
   (), inplace=True)
9  data['Embarked'].fillna(data['Embarke
   d'].mode()[0], inplace=True)
10 data.drop('Cabin', axis=1,
   inplace=True)
11
12 # Convert categorical features to
   numeric
13 data['Sex'] =
   data['Sex'].map({'male': 0,
   'female': 1})
14 data = pd.get_dummies(data, columns=
   ['Embarked'], drop_first=True)
15
16 # Write your code here for Bar Plot
   for Survival by Embarked
17
18
19 grouped= data.groupby('Embarked_Q')
   ['Survived'].value_counts().unstack()
   .fillna(0)
20 grouped.columns = ['Not Survived',
   'Survived']
21 grouped.plot(kind='bar',stacked=True)
22 plt.title('Survival by Embarked')
23 plt.xlabel('Embarked')
24 plt.ylabel('Count')
25 plt.legend(title=None)
26 plt.show()
```

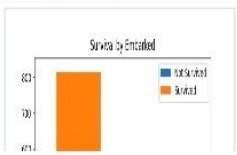
Average time
1.710 s
1710.00 ms

Maximum time
1.710 s
1710.00 ms

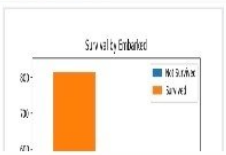
1 out of 1 shown test case(s) passed

Test case 1 1710 ms Debug

Expected output



Actual output



< Prev Reset Submit Next >

CodeTantra

5.2.7. Box plot for Age Distribution

06:57

Write a Python code to plot a boxplot that shows the distribution of the 'Age' column from the Titanic dataset across different passenger classes. The boxplot should display the following specifications:

1. Use the **Pclass** column to group the data for the boxplot.
2. Set the title of the plot to "**Age by Pclass**".
3. Remove the default subtitle with **plt.suptitle('')**.
4. Label the x-axis as '**Pclass**' and the y-axis as '**Age**'.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

BoxPlotFor...

Submit

Debugger

```
1  import pandas as pd
2  import matplotlib.pyplot as plt
3
4  # Load the Titanic dataset
5  data = pd.read_csv('Titanic-
   Dataset.csv')
6
7  # Data Cleaning
8  data['Age'].fillna(data['Age'].median
   (), inplace=True)
9  data['Embarked'].fillna(data['Embarke
   d'].mode()[0], inplace=True)
10 data.drop('Cabin', axis=1,
   inplace=True)
11
12 # Convert categorical features to
   numeric
13 data['Sex'] =
   data['Sex'].map({'male': 0,
   'female': 1})
14 data = pd.get_dummies(data, columns=
   ['Embarked'], drop_first=True)
15
16
17 # Write your code here for Box Plot
   for Age by Pclass
18 data.boxplot(column = 'Age',
   by='Pclass')
19 plt.title('Age by Pclass')
20 plt.suptitle('')
21 plt.xlabel('Pclass')
22 plt.ylabel('Age')
23 plt.show()
24 #
```

Average time
1.630 s
1630.00 ms

Maximum time
1.630 s
1630.00 ms

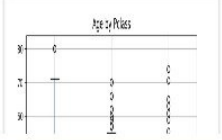
1 out of 1 shown test case(s) passed

Test case 1 1630 ms

Debug

Expected output

Actual output



‹ Prev Reset Submit Next ›

5.2.8. Box Plot for Age by Survived

02:17

Write a Python code to plot a boxplot that shows the distribution of the 'Age' column from the Titanic dataset based on whether passengers survived or not. The boxplot should display the following specifications:

1. Use the **Survived** column to group the data for the boxplot (0 = Did not survive, 1 = Survived).
2. Set the title of the plot to "**Age by Survival**".
3. Remove the default subtitle with **plt.suptitle('')**.
4. Label the x-axis as '**Survived**' and the y-axis as '**Age**'.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

BoxPlotFor...

Submit

Debugger

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the Titanic dataset
data = pd.read_csv('Titanic-Dataset.csv')

# Data Cleaning
data['Age'].fillna(data['Age'].median(), inplace=True)
data['Embarked'].fillna(data['Embarked'].mode()[0], inplace=True)
data.drop('Cabin', axis=1, inplace=True)

# Convert categorical features to numeric
data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
data = pd.get_dummies(data, columns=['Embarked'], drop_first=True)

# Write your code here for Box Plot for Age by Survived


plt.figure(figsize=(8, 6))
data.boxplot(column='Age', by='Survived')
plt.suptitle('')
plt.title('Age by Survival')
plt.xlabel('Survived')
plt.ylabel('Age')
plt.show()
```

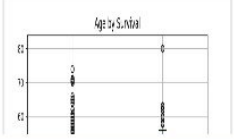
Average time
1.522 s
1522.00 ms

Maximum time
1.522 s
1522.00 ms

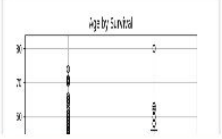
1 out of 1 shown test case(s) passed

Test case 1 1522 ms Debug

Expected output



Actual output



< Prev Reset Submit Next >

5.2.9. Box Plot for Fare by Pclass

01:18

Write a Python code to plot a boxplot that shows the distribution of the 'Fare' column from the Titanic dataset based on the passenger class (Pclass). The boxplot should display the following specifications:

1. Use the **Pclass** column to group the data for the boxplot.
2. Set the title of the plot to "**Fare by Pclass**".
3. Remove the default subtitle with **plt.suptitle('')**.
4. Label the x-axis as '**Pclass**' and the y-axis as '**Fare**'.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

BoxPlotFor...

Submit

Debugger

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the Titanic dataset
data = pd.read_csv('Titanic-Dataset.csv')

# Data Cleaning
data['Age'].fillna(data['Age'].median(), inplace=True)
data['Embarked'].fillna(data['Embarked'].mode()[0], inplace=True)
data.drop('Cabin', axis=1, inplace=True)

# Convert categorical features to numeric
data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
data = pd.get_dummies(data, columns=['Embarked'], drop_first=True)

# Write your code here for Box Plot for Fare by Pclass


plt.figure(figsize=(8, 6))
data.boxplot(column='Fare',by='Pclass')
plt.suptitle('')
plt.title('Fare by Pclass')
plt.xlabel('Pclass')
plt.ylabel('Fare')
plt.show()
```

Average time
1.504 s
1504.00 ms

Maximum time
1.504 s
1504.00 ms

1 out of 1 shown test case(s) passed

Test case 1 1504 ms Debug

Expected output

Actual output

‹ Prev Reset Submit Next ›

5.2.10. Scatter Plot for Age vs. Fare

02:33

Write a Python code to plot a scatter plot showing the relationship between the 'Age' and 'Fare' columns in the Titanic dataset. The scatter plot should display the following specifications:

1. Use the **Age** column for the x-axis and the **Fare** column for the y-axis.
2. Set the title of the plot to "**Age vs. Fare**".
3. Label the x-axis as '**Age**' and the y-axis as '**Fare**'.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

AgeFareSc...

Submit

Debugger

```python
import pandas as pd
import matplotlib.pyplot as plt

# Load the Titanic dataset
data = pd.read_csv('Titanic-Dataset.csv')

# Data Cleaning
data['Age'].fillna(data['Age'].median(), inplace=True)
data['Embarked'].fillna(data['Embarked'].mode()[0], inplace=True)
data.drop('Cabin', axis=1, inplace=True)

# Convert categorical features to numeric
data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
data = pd.get_dummies(data, columns=['Embarked'], drop_first=True)

# Write your code here for Box Plot for Fare by Pclass

# Write your code here for Box Plot for Fare by Pclass
plt.figure()
plt.scatter(data['Age'],data['Fare'])
plt.title('Age vs. Fare')
plt.xlabel('Age')
plt.ylabel('Fare')
plt.show()
#
```

Average time
1.235 s
1235.00 ms

Maximum time
1.235 s
1235.00 ms

1 out of 1 shown test case(s) passed

Test case 1 1235 ms Debug

Expected output

Actual output

5.2.11. Scatter Plot for Age vs. Fare by Survi... 09:02

Write a Python code to plot a scatter plot showing the relationship between the 'Age' and 'Fare' columns in the Titanic dataset, with points color-coded by survival status. The scatter plot should display the following specifications:

1. Use the **Age** column for the x-axis and the **Fare** column for the y-axis.
2. Color the points based on the **Survived** column: **Red** for passengers who did not survive (**Survived = 0**). **Blue** for passengers who survived (**Survived = 1**).
3. Set the title of the plot to "**Age vs. Fare by Survival**".
4. Label the x-axis as '**Age**' and the y-axis as '**Fare**'.

The Titanic dataset contains columns as shown below,

PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked

Sample Data:

```
PassengerId,Survived,Pclass,Name,Sex,Age,SibSp,Parch,Ti
1,0,3,"Braund, Mr. Owen Harris",male,22,1,0,A/5 21171,7
2,1,1,"Cumings, Mrs. John Bradley (Florence Briggs Thay
3,1,3,"Heikkinen, Miss. Laina",female,26,0,0,STON/O2. 3
4,1,1,"Futrelle, Mrs. Jacques Heath (Lily May Peel)",fe
5,0,3,"Allen, Mr. William Henry",male,35,0,0,373450,8.0
6,0,3,"Moran, Mr. James",male,,0,0,330877,8.4583,,Q
7,0,1,"McCarthy, Mr. Timothy J",male,54,0,0,17463,51.86
8,0,3,"Palsson, Master. Gosta Leonard",male,2,3,1,34990
9,1,3,"Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg
10,1,2,"Nasser, Mrs. Nicholas (Adele Achem)",female,14,
```

Note: Refer to the visible test case for better reference.

Sample Test Cases +

Explorer

AgeFareSc... Submit

Debugger

```
1  import pandas as pd
2  import matplotlib.pyplot as plt
3
4  # Load the Titanic dataset
5  data = pd.read_csv('Titanic-
   Dataset.csv')
6
7  # Data Cleaning
8  data['Age'].fillna(data['Age'].median
   (), inplace=True)
9  data['Embarked'].fillna(data['Embarke
   d'].mode()[0], inplace=True)
10 data.drop('Cabin', axis=1,
   inplace=True)
11
12 # Convert categorical features to
   numeric
13 data['Sex'] =
   data['Sex'].map({'male': 0,
   'female': 1})
14 data = pd.get_dummies(data, columns=
   ['Embarked'], drop_first=True)
15
16 # Write your code here for Scatter
   Plot for Age vs. Fare by Survived
17
18
19
20 # Write your code here for Scatter
   Plot for Age vs. Fare by Survived
21 colors = data['Survived'].map({0:
   'red',1: 'blue'})
22
23 plt.scatter(data['Age'],data['Fare'],
   c=colors)
24 plt.title('Age vs. Fare by Survival')
25 plt.xlabel('Age')
26 plt.ylabel('Fare')
27 plt.show()
28 #
29
30
```

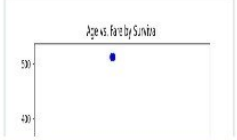
Average time 1.315 s 1315.00 ms

Maximum time 1.315 s 1315.00 ms

1 out of 1 shown test case(s) passed

Test case 1 1315 ms Debug

Expected output



Actual output

Prev Reset Submit