

CODETANTRA

Home

202401070179@mitaoe.ac.inSupportLogout

5.1.1. Stacked Plot

08:30

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

stackedpl...

```
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',
7             'June', 'July', 'August', 'September', 'October', 'November',
8             'December'],
9     'City_A_Temperature': [5, 7, 10, 13, 17, 20, 22, 21, 18,
10                          12, 8, 6],
11     'City_B_Temperature': [2, 3, 5, 6, 10, 14, 16, 17, 12, 9,
12                          5, 3],
13     'City_C_Temperature': [3, 4, 6, 8, 9, 12, 15, 14, 10, 7,
14                          4, 2]
15 }
16
17 # Write your code...
18 df = pd.DataFrame(data)
19 plt.stackplot(df['Month'], df['City_A_Temperature'], df['City_B_Temperature'], df['City_C_Temperature'])
20 plt.title('Temperature Variation')
```

TerminalTest cases

Activate Windows
Go to Settings to activate Windows.

PrevResetSubmitNext

5.1.1. Stacked Plot08:30

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

stackedpl...

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

TerminalTest cases

Activate WindowsGo to Settings to activate Windows.

< PrevResetSubmitNext >

CODETANTRA

Home

202401070179@mitabe.ac.inSupportLogout

5.2.1. Titanic Dataset39:05

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).

Sample Test Cases

titanicDat...

1import pandas as pd

2import matplotlib.pyplot as plt

3

4# Load the Titanic dataset from the CSV file

5df = pd.read_csv('titanic.csv')

6

7# Set up the figure for 5 subplots

8fig, axes = plt.subplots(3, 2, figsize=(12, 12))

9

10# write the code..

11df = pd.read_csv('titanic.csv')

12

13fig, axes = plt.subplots(3, 2, figsize=(12, 12))

14

15axes[0, 0].bar(df['Pclass'].value_counts().index,

16df['Pclass'].value_counts(), color='skyblue')

17axes[0, 0].set_title("Passenger Class Distribution")

18axes[0, 0].set_xlabel('Pclass')

19axes[0, 0].set_ylabel('Count')

20axes[0, 1].pie(df['Gender'].value_counts(),

Terminal

Test cases

Activate Windows
Go to Settings to activate Windows.

Prev

Reset

Submit

Next

5.2.1. Titanic Dataset

39:05

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).

Sample Test Cases

+

```
20 axes[0, 1].pie(df['Gender'].value_counts(),
21 labels=df['Gender'].value_counts().index, autopct='%1.1f%%',
22 colors=['lightblue', 'lightcoral'])
23 axes[0, 1].set_title("Gender Distribution")
24
25 axes[1, 0].hist(df["Age"].dropna(), bins=8,
26 color="lightgreen", edgecolor='black')
27 axes[1, 0].set_title("Age Distribution")
28 axes[1, 0].set_xlabel("Age")
29 axes[1, 0].set_ylabel("Frequency")
30
31 axes[1, 1].bar(df["Survived"].value_counts().index,
32 df['Survived'].value_counts(), color=['lightblue',
33 'lightcoral'])
34 axes[1, 1].set_title("Survival Count")
35 axes[1, 1].set_xlabel("Survived (0 = No, 1 = Yes)")
36 axes[1, 1].set_ylabel("Count")
37
38 axes[2, 0].scatter(df['Age'], df['Fare'], color='orange',
39 edgecolors='black')
40 axes[2, 0].set_title("Fare vs Age")
41
```

Terminal Test cases

Activate Windows

Go to Settings to activate Windows.

CODETANTRA

Home

202401070179@mitace.ac.in

Support

Logout

5.2.1. Titanic Dataset

39:05

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).

Sample Test Cases

titanicDat...

Submit

23

24

25

26

27

28

29

30

31

32

33

34

35

36

37

38

39

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")

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")

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()

Terminal

Test cases

Activate Windows

Go to Settings to activate Windows.

Debugger

< Prev

Reset

Submit

Next >

5.2.2. Histogram of passenger information of Titanic 02:25

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 Test Cases +

Histogram... Submit

```
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['Embarked'].mode()[0],
10                          inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16                        drop_first=True)
17
18 # Write your code here for Histogram
19
20 plt.hist(data['Age'], bins=30, edgecolor='k')
21 plt.xlabel('Age')
```

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

< Prev

Reset

Submit

Next >

5.2.2. Histogram of passenger information of Titanic 02:25

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:

- Use **30 bins** for the histogram.
- Set the **edge color** of the bars to **black (k)**.
- Label the x-axis as **'Age'** and the y-axis as **'Frequency'**.
- 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 Test Cases +

Histogram... Submit

```
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['Embarked'].mode()[0],
10                           inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16                           drop_first=True)
17
18 # Write your code here for Histogram
19
20 plt.hist(data['Age'], bins=30, edgecolor='k')
21 plt.xlabel('Age')
22 plt.ylabel('Frequency')
23 plt.title('Age Distribution')
24 plt.show()
```

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

< Prev

Reset

Submit

Next >

5.2.3. Bar plot of survival rate of passengers

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 Test Cases

+

BarPlotOf...

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['Embarked'].mode()[0],
10 inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16 drop_first=True)
17
18 # Write your code here for Bar Plot for Survival Rate
19
20 survival_counts = data['Survived'].value_counts()
21 survival_counts.plot(kind='bar')

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

Submit

5.2.3. Bar plot of survival rate of passengers03:51

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 Test Cases+

BarPlotOf...Submit

```
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['Embarked'].mode()[0],
10                           inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16                           drop_first=True)
17
18 # Write your code here for Bar Plot for Survival Rate
19
20 survival_counts = data['Survived'].value_counts()
21 survival_counts.plot(kind='bar')
22 plt.title('Survival Count')
23 plt.xlabel('Survived')
24 plt.ylabel('Count')
25 plt.show()
```

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

< Prev

Reset

Submit

Next >

5.2.4. Bar Plot for Survival by Gender

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 Test Cases

+

BarPlotOf... Submit

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['Embarked'].mode()[0],
10 inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16 drop_first=True)
17
18 # Write your code here for Bar Plot for Survival by Gender
19
20 survival_by_gender = data.groupby('Sex')

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

< Prev Reset Submit Next >

5.2.4. Bar Plot for Survival by Gender

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 Test Cases

+

BarPlotOf... Submit

```
10 inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16 drop_first=True)
17
18 # Write your code here for Bar Plot for Survival by Gender
19
20 survival_by_gender = data.groupby('Sex')
21 ['Survived'].value_counts().unstack().fillna(0)
22 survival_by_gender.columns = ["Not Survived", 'Survived']
23 survival_by_gender.index = ['0', '1']
24 survival_by_gender.plot(kind='bar', stacked=True)
25 plt.title('Survival by Gender')
26 plt.xlabel('Gender')
27 plt.ylabel('Count')
28 plt.legend(title=None)
29 plt.show()
```

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

Prev Reset Submit Next

5.2.5. Bar Plot for Survival by Pclass

00:18

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 Test Cases

+

BarPlotOf...

Submit

```

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['Embarked'].mode()[0],
10                          inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16                        drop_first=True)
17
18 # Write your code here for Bar Plot for Survival by Pclass
19
20 survival_by_class = data.groupby('Pclass')
21 ['Survived'].value_counts().unstack().fillna(0)

```

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

< Prev Reset Submit Next >

5.2.5. Bar Plot for Survival by Pclass06:18

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:

- 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()**.
- Use a **stacked bar chart** to display the survival counts.
- Add the title "**Survival by Pclass**" to the chart.
- Label the x-axis as '**Pclass**' and the y-axis as '**Count**'.
- 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 Test Cases+

BarPlotOf...Submit

```
8 data['Age'].fillna(data['Age'].median(), inplace=True)
9 data['Embarked'].fillna(data['Embarked'].mode()[0],
10                           inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16                           drop_first=True)
17
18 # Write your code here for Bar Plot for Survival by Pclass
19
20 survival_by_class = data.groupby('Pclass')
21 ['Survived'].value_counts().unstack().fillna(0)
22 survival_by_class.columns = ['Not Survived', 'Survived']
23 survival_by_class.plot(kind='bar', stacked=True)
24 plt.title('Survival by Pclass')
25 plt.xlabel('Pclass')
26 plt.ylabel('Count')
27 plt.legend(title=None)
28 plt.show()
```

TerminalTest cases

Activate WindowsGo to Settings to activate Windows.

< PrevResetSubmitNext >

5.2.6. Bar Plot for Survival by Embarked

07:58

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 Test Cases

+

```
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['Embarked'].mode()[0],
10                          inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16                        drop_first=True)
17
18 # Write your code here for Bar Plot for Survival by Embarked
19
20 grouped = data.groupby('Embarked_Q')
21 ['Survived'].value_counts().unstack().fillna(0)
```


07:58

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.

Pa ss	Su pl	Pcl	Na	Se	Ad	Si	Ba	Tic	Ea	Ca	E mb
----------	----------	-----	----	----	----	----	----	-----	----	----	---------

Sample Test Cases

+

Explorer

BarPlotOf...

Submit

Debugger

```

8 data['Age'].fillna(data['Age'].median(), inplace=True)
9 data['Embarked'].fillna(data['Embarked'].mode()[0],
10 inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16 drop_first=True)
17
18 # Write your code here for Bar Plot for Survival by Embarked
19
20 grouped = data.groupby('Embarked_Q')
21 ['Survived'].value_counts().unstack().fillna(0)
22 grouped.columns = ['Not Survived', 'Survived']
23 grouped.plot(kind='bar', stacked=True)
24 plt.title('Survival by Embarked')
25 plt.xlabel('Embarked')
26 plt.ylabel('Count')
27 plt.legend(title=None)
28 plt.show()

```

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

5.2.7. Box plot for Age Distribution 02:34

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 Test Cases +

BoxPlotF... Submit

Explorer

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['Embarked'].mode()[0],
10 inplace=True)
11 data.drop('Cabin', axis=1, inplace=True)
12
13 # Convert categorical features to numeric
14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
15 data = pd.get_dummies(data, columns=['Embarked'],
16 drop_first=True)
17
18 # Write your code here for Box Plot for Age by Pclass
19
20 plt.figure(figsize=(8, 6))
21 data.boxplot(column='Age', by='Pclass')

Debugger

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

< Prev Reset Submit Next >

5.2.7. Box plot for Age Distribution02:34

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:

- Use the **Pclass** column to group the data for the boxplot.
- Set the title of the plot to **"Age by Pclass"**.
- Remove the default subtitle with **plt.suptitle("")**.
- 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 Test Cases+

BoxPlotF...Submit

```
6
7
8 # Data Cleaning
9 data['Age'].fillna(data['Age'].median(), inplace=True)
10 data['Embarked'].fillna(data['Embarked'].mode()[0],
11                          inplace=True)
12 data.drop('Cabin', axis=1, inplace=True)
13
14 # Convert categorical features to numeric
15 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})
16 data = pd.get_dummies(data, columns=['Embarked'],
17                          drop_first=True)
18
19 # Write your code here for Box Plot for Age by Pclass
20
21 plt.figure(figsize=(8, 6))
22 data.boxplot(column='Age', by='Pclass')
23 plt.suptitle('')
24 plt.title('Age by Pclass')
25 plt.xlabel('Pclass')
26 plt.ylabel('Age')
27 plt.show()
```

TerminalTest cases

Activate Windows
Go to Settings to activate Windows.

< Prev

Reset

Submit

Next >

5.2.8. Box Plot for Age by Survived03:04

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:

- Use the **Survived** column to group the data for the boxplot (0 = Did not survive, 1 = Survived).
- Set the title of the plot to **"Age by Survival"**.
- Remove the default subtitle with **plt.suptitle("")**.
- 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 Test Cases

BoxPlotF...Submit

1import pandas as pd

2import matplotlib.pyplot as plt

3

4# Load the Titanic dataset

5data = pd.read_csv('Titanic-Dataset.csv')

6

7# Data Cleaning

8data['Age'].fillna(data['Age'].median(), inplace=True)

9data['Embarked'].fillna(data['Embarked'].mode()[0],

10inplace=True)

11data.drop('Cabin', axis=1, inplace=True)

12

13# Convert categorical features to numeric

14data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})

15data = pd.get_dummies(data, columns=['Embarked'],

16drop_first=True)

17

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

19

20plt.figure(figsize=(8, 6))

21data.boxplot(column='Age', by='Survived')

22

TerminalTest cases

Activate Windows

Go to Settings to activate Windows.

PrevResetSubmitNext

5.2.8. Box Plot for Age by Survived 03:04

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:

- Use the **Survived** column to group the data for the boxplot (0 = Did not survive, 1 = Survived).
- Set the title of the plot to **"Age by Survival"**.
- Remove the default subtitle with **plt.suptitle("")**.
- 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 Test Cases +

BoxPlotF... Submit

6

Data Cleaning

7

data['Age'].fillna(data['Age'].median(), inplace=True)

8

data['Embarked'].fillna(data['Embarked'].mode()[0],

9

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'],

15

drop_first=True)

16

Write your code here for Box Plot for Age by Survived

17

18

plt.figure(figsize=(8, 6))

19

data.boxplot(column='Age', by='Survived')

20

plt.suptitle('')

21

plt.title('Age by Survival')

22

plt.xlabel('Survived')

23

plt.ylabel('Age')

24

plt.show()

Terminal Test cases

Activate Windows
Go to Settings to activate Windows.

< Prev

Reset

Submit

Next >

5.2.9. Box Plot for Fare by Pclass

02:42

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 Test Cases

+

BoxPlotF...

Submit

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['Embarked'].mode()[0],

10 inplace=True)

11 data.drop('Cabin', axis=1, inplace=True)

12

13 # Convert categorical features to numeric

14 data['Sex'] = data['Sex'].map({'male': 0, 'female': 1})

15 data = pd.get_dummies(data, columns=['Embarked'],

16 drop_first=True)

17

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

19

20 plt.figure(figsize=(8,6))

21 data.boxplot(column='Fare', by='Pclass')

Terminal Test cases

Activate Windows

Go to Settings to activate Windows.