Group

AAssignmentNo:

8

TitleoftheAssignment:DataVisualisationI

1. Use the inbuilt dataset 'titanic'. The dataset contains 891 rows and contains informationabout the passengers who boarded the unfortunate Titanic ship. Use the Seaborn library to seeif we can find any patterns in the data.

2. Writeacodetocheckhowthepriceoftheticket(columnname:'fare')foreachpassengeris distributed by plotting a histogram.

ObjectiveoftheAssignment:

Studentsshouldbeabletoperformthedata Visualizationoperationusing Python on any open source dataset

Prerequisite:

1. Basic of Python Programming
2. SeabornLibrary,ConceptofDataVisualisation.

Contents forTheory:

1. **Seaborn Library Basics**
2. **Know yourData**
3. **Finding patterns of data.**

4. **Checkinghowthepriceoftheticket(columnname:'fare')foreachpassengerisdistributed by plotting a histogram.**

Theory:

Data Visualisation plays a very important role in Data mining. Various data scientists spent theirtimeexploringdatathroughvisualisation.Toacceleratethisprocessweneedtohaveawell-documentation of all the plots.

Evenplentyofresourcescan'tbetransformedintovaluablegoodswithoutplanningandarchitecture

1. **Seaborn Library Basics**

   SeabornisaPythondatavisualisationlibrarybasedonmatplotlib.Itprovidesahigh-level interface for drawing attractive and informative statistical graphics.

   For the installation of Seaborn, you may run any of the following in your command line.

   ```
   pip install
   seaborncondainstallse
   aborn
   ```

Toimportseabornyoucanrunthefollowingcommand.

   ```
   importseaborn as sns
   ```

2. **Know yourdata**

The dataset that we are going to use to draw our plots will be the Titanic dataset, which isdownloaded by default with the Seaborn library. All you have to do is use the load_datasetfunction and pass it the name of the dataset.

Let'sseewhattheTitanicdatasetlookslike. Executethefollowingscript:

```
import pandas as
pd import numpy as np
import matplotlib.pyplot as
plt import seaborn as sns
```

```
dataset =

sns.load_dataset('titanic')dataset.he

ad()
```

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True

The dataset contains 891 rows and 15 columns and contains information about the passengerswho boarded the unfortunate Titanic ship. The original task is to predict whether or not thepassengersurviveddependingupondifferentfeaturessuchastheirage,ticket,cabintheyboarded, the class of the ticket, etc. We will use the Seaborn library to see if we can find anypatterns in the data.

3.   **Finding patterns of data.**

Patternsof data canbe find outwith the helpof different typesof plots

Typesofplotsare:

    A. Distribution Plots

       a. Dist-Plot

       b. Joint Plot

       d. Rug Plot

    B. Categorical Plots

       a. Bar Plot

       b. Count Plot

       c. Box Plot

       d. ViolinPlot

C. Advanced Plots

    a. Strip Plot

    b. Swarm Plot

### D. Matrix Plots

    a. Heat Map

    b. Cluster Map

A. Distribution Plots:

Theseplotshelpustovisualisethedistributionofdata.Wecanusetheseplotstounderstandthemean, median, range, variance, deviation, etc of the data.

a. Distplot

- Dist plot gives us the histogram of the selected continuous variable.
- It is an example of a univariate analysis.
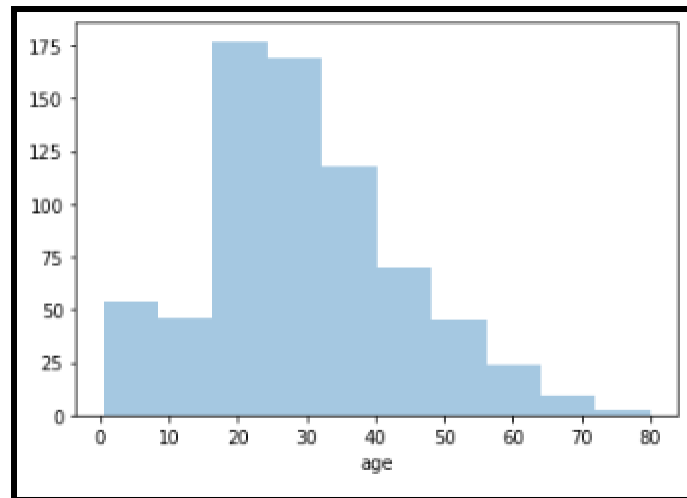- Wecanchangethenumberofbinsi.e.numberofverticalbarsinahistogram

```
import seaborn as sns

sns.distplot(x = dataset['age'], bins = 10)
```
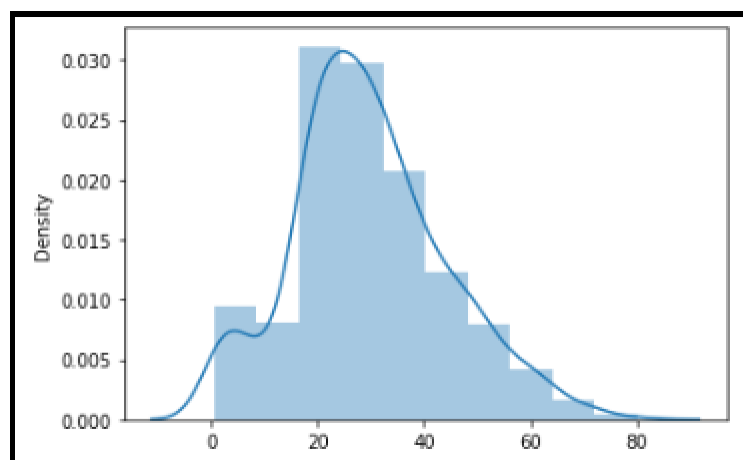
Thelinethatyousee represents the kernel density estimation. You can remove this lineby passing False as the parameter for the kde attribute as shown below

```
sns.distplot(dataset['age'], bins = 10,kde=False)
```


Herethex-axisistheageandthey-axisdisplaysfrequency.Forexample,forbins=10,there are around 50 people having age 0 to 10

b. Joint Plot

- It is the combination of the distplot of two variables.

- It is an example of bivariate analysis.

- We additionally obtain a scatter plot between the variables to reflect their linearrelationship.Wecancustomisethescatterplotintoahexagonalplot,where,themorethe colourintensity, themore willbe thenumber ofobservations.

```
importseabornas sns#
```

```
For Plot 1
```

```
sns.jointplot(x=dataset['age'],y=dataset['fare'],kind='scatter')
```

```
# For Plot 2
```

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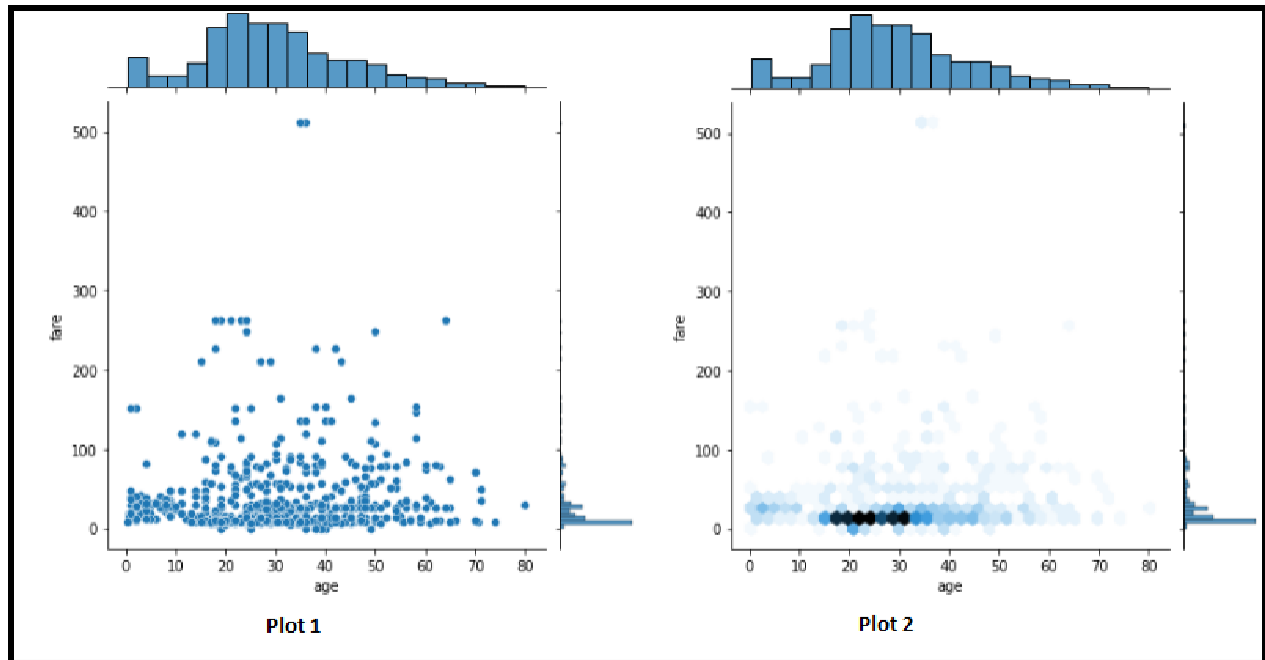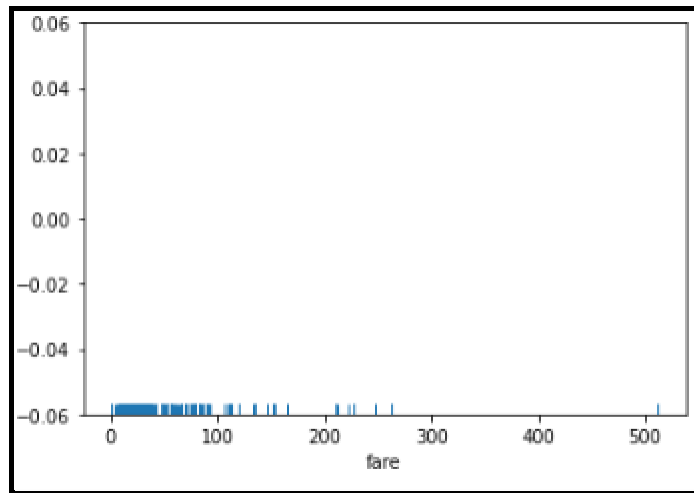

- From the output, you can see that a joint plot has three parts. A distribution plot at the topfor the column on the x-axis, a distribution plot on the right for the column on the y-axisand a scatter plot in between that shows the mutual distribution of data for both thecolumns.Youcanseethatthereisnocorrelationobservedbetweenpricesandthefares.

- You can change the type of the joint plot by passing a value for the kind parameter. Forinstance, if instead of a scatter plot, you want to display the distribution of data in theformof ahexagonal plot, youcan passthe value hexfor thekind parameter.

- In the hexagonal plot, the hexagon with the most number of points gets darker colour. Soifyoulookattheaboveplot,youcanseethatmostofthepassengersarebetweentheages   of   20 and 30 and most of them paid between 10-50 for the tickets.

c. The Rug Plot

Therugplot()isusedtodrawsmallbarsalongthex-axisforeachpointinthedataset.Toplotarug plot, you need to pass the name of the column. Let's plot a rug plot for fare.

```
sns.rugplot(dataset['fare'])
```

From the output, you can see that most of the instances for the fares have values between 0 and100.

These are some of the most commonly used distribution plots offered by the Python's SeabornLibrary.Let'ssee someof thecategorical plotsinthe Seabornlibrary.

## CategoricalPlots

Categoricalplots,asthenamesuggests,arenormallyusedtoplotcategoricaldata.Thecategoricalplotsplotthev aluesinthecategoricalcolumnagainstanothercategoricalcolumnora numeric column. Let's see some of the most commonly used categorical data.

a. The BarPlot

The barplot() is used to display the mean value for each value in a categorical column, against anumericcolumn.Thefirst parameter is the categorical column, the second parameter is thenumeric column while the third parameter is the dataset. For instance, if you want to know themean value of the age of the male and female passengers, you can use the bar plot as follows.

```
sns.barplot(x='sex', y='age', data=dataset)
```
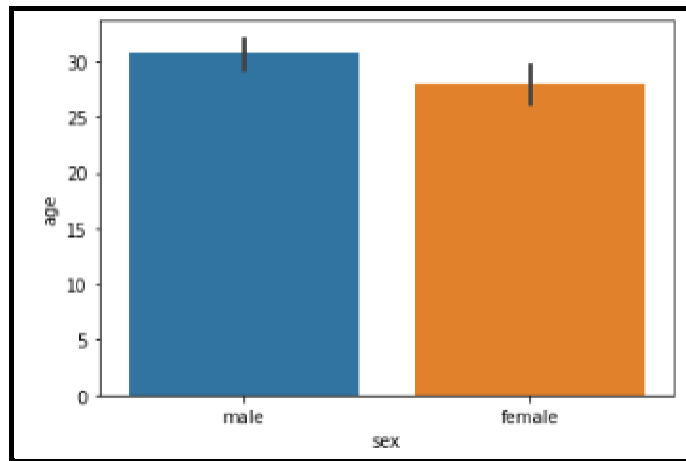
From the output, you can clearly see that the average age of male passengers is just less than 40while the average age of female passengers is around 33.

In addition to finding the average, the bar plot can also be used to calculate other aggregatevalues for each category. To do so, you need to pass the aggregate function to the estimator. Forinstance, you can calculate the standard deviation for the age of each gender as follows:
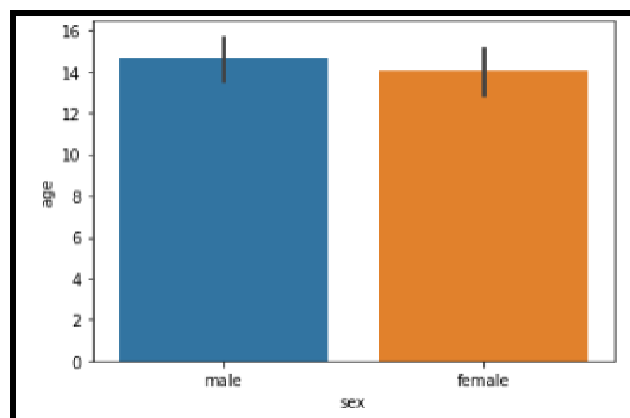
```python
import numpy as np

import matplotlib.pyplot as

plt import seaborn as sns

sns.barplot(x='sex', y='age', data=dataset, estimator=np.std)
```

Notice, in the above script we use the std aggregate function from the numpy library to calculatethe standard deviation for the ages of male and female passengers.The output looks like this:

b. The Count Plot

The count plot is similar to the bar plot, however it displays the count of the categories in aspecific column. For instance, if we want to count the number of males and women passenger wecan do so using count plot as follows:

```
sns.countplot(x='sex', data=dataset)
```
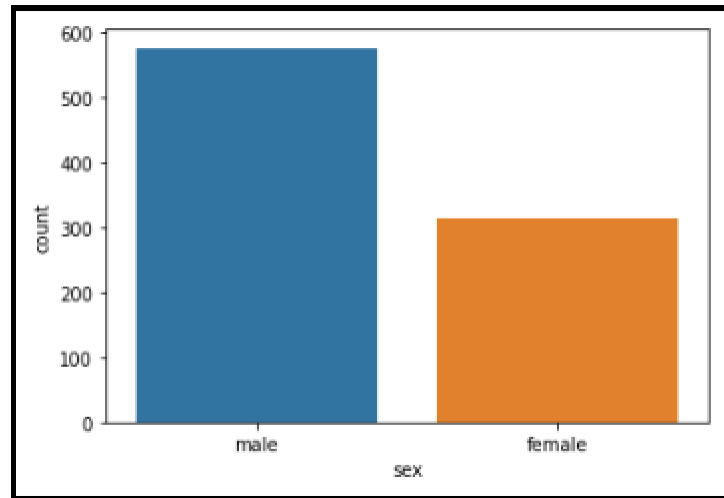

c. The Box Plot

Theboxplotisusedtodisplaythedistributionofthecategoricaldataintheformofquartiles.Thecentreoftheboxs howsthemedianvalue.Thevaluefromthelowerwhiskerto thebottomof the box shows the first quartile. From the bottom of the box to the middle of the box lies thesecond quartile. From the middle of the box to the top of the box lies the third quartile and finallyfrom the top of the box to the top whisker lies the last quartile.

Now let's plot a box plot that displays the distribution for the age with respect to each gender.You need to pass the categorical column as the first parameter (which is sex in our case) and thenumeric column (age in our case) as the second parameter. Finally, the dataset is passed as thethirdparameter, take a look at the following script:

```
sns.boxplot(x='sex', y='age', data=dataset)
```
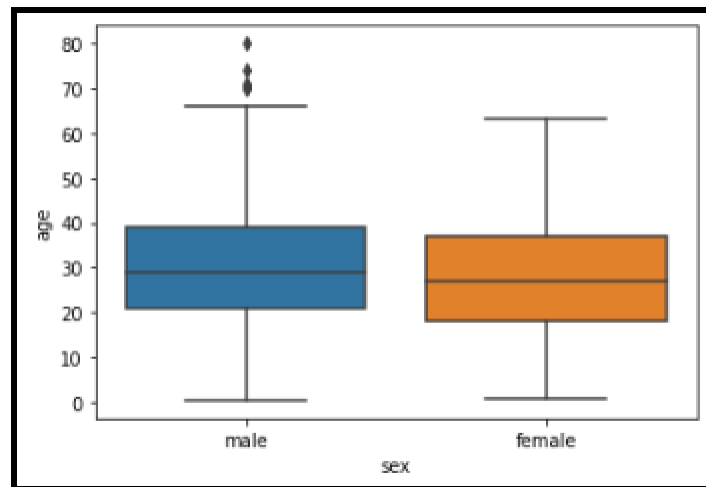
Let'stry to understand the box plot for females. The first quartile starts at around 1 and ends at20whichmeansthat25% of the passengers are aged between 1 and 20. The second quartilestarts at around 20 and ends at around 28 which means that 25% of the passengers are agedbetween20 and 28. Similarly, the third quartile starts and ends between 28 and 38, hence 25%passengers are aged within this range and finally the fourth or last quartile starts at 38 and endsaround 64.

If there are any outliers or the passengers that do not belong to any of the quartiles, they arecalled outliers and are represented by dots on the box plot.

You can make your box plots more fancy by adding another layer of distribution. For instance, ifyou want to see the box plots of forage of passengers of both genders, along with the informationabout whether or not they survived, you can pass the survived as value to the hue parameter asshown below:

```
sns.boxplot(x='sex', y='age', data=dataset, hue="survived")
```
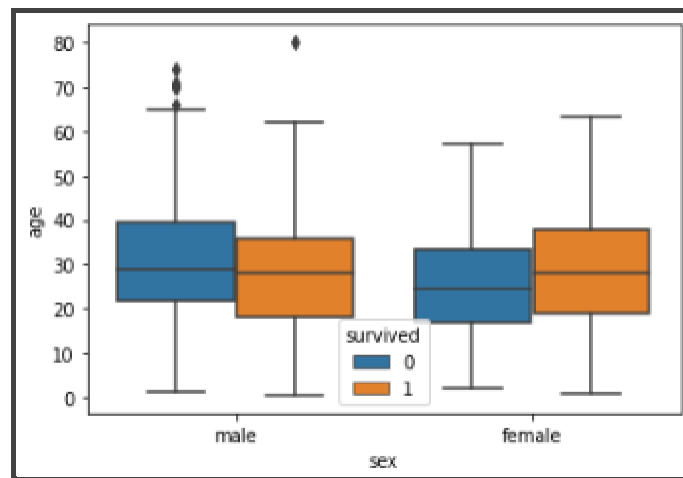
Nowinadditiontotheinformationabouttheageofeachgender,youcanalsoseethedistribution   of   the passengers who survived. For instance, you can see that among the malepassengers, on average more younger people survived as compared to the older ones. Similarly,you can see that the variation among the age of female passengers who did not survive is muchgreater than the age of the surviving female passengers.

d. TheViolinPlot

The violin plot is similar to the box plot, however, the violin plot allows us to display all thecomponentsthatactuallycorrespondtothedatapoint.Theviolinplot()functionisusedtoplottheviolin plot.Likethebox plot, the first parameter is the categorical column, the secondparameter is the numeric column while the third parameter is the dataset.

Let'splotaviolinplotthat displaysthedistributionforthe agewithrespecttoeach gender.

```
sns.violinplot(x='sex', y='age', data=dataset)
```
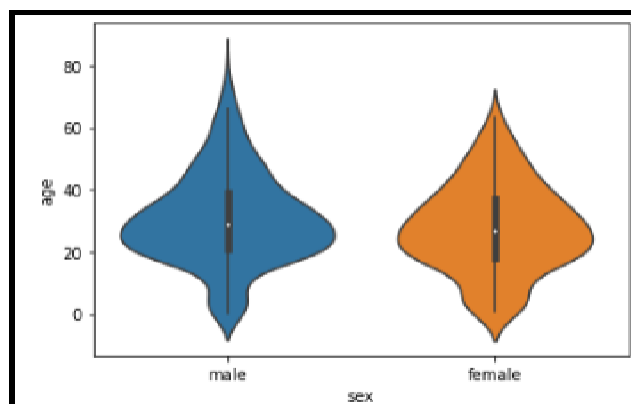
You can see from the figure above that violin plots provide much more information about thedata as compared to the box plot. Instead of plotting the quartile, the violin plot allows us to seeall the components that actually correspond to the data. The area where the violin plot is thickerhas a higher number of instances for the age. For instance, from the violin plot for males, it isclearly evident that the number of passengers with age between 20 and 40 is higher than all therest of the age brackets.

Like box plots, you can also add another categorical variable to the violin plot using the hueparameter as shown below:

```
sns.violinplot(x='sex', y='age', data=dataset, hue='survived')
```
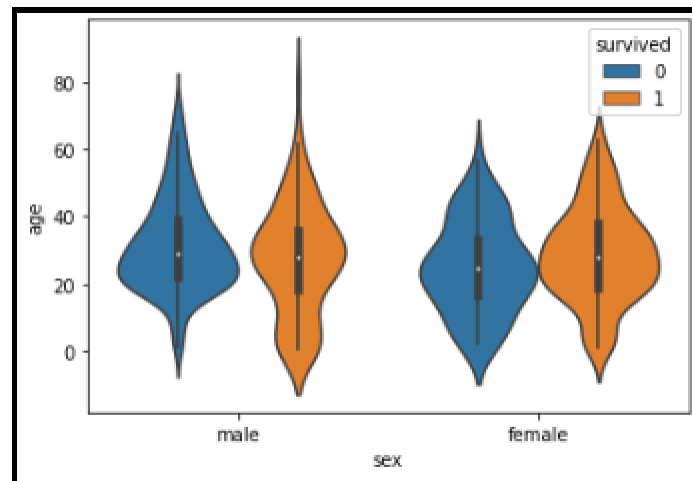

Now you can see a lot of information on the violin plot. For instance, if you look at the bottom ofthe violin plot for the males who survived (left-orange), you can see that it is thicker than thebottom of the violin plot for the males who didn't survive (left-blue). This means that the numberof young male passengers who survived is greater than the number of young male passengerswho did not survive

Advanced Plots:

a. The Strip Plot

The strip plot draws a scatter plot where one of the variables is categorical. We have seen scatterplots in the joint plot and the pair plot sections where we had two numeric variables. The stripplotisdifferentinawaythatoneofthevariablesiscategoricalinthiscase,andforeach

categoryinthecategoricalvariable,you will see a scatter plot with respect to the numericcolumn.

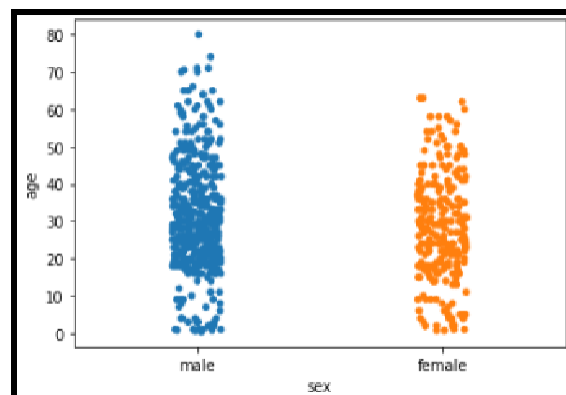The stripplot() function is used to plot the violin plot. Like the box plot, the first parameter is thecategorical column, the second parameter is the numeric column while the third parameter is thedataset. Look at the following script:

```
sns.stripplot(x='sex', y='age', data=dataset, jitter=False)
```
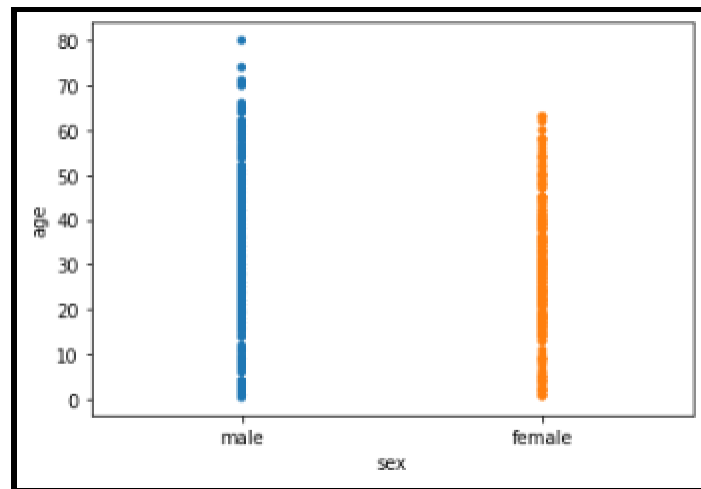

You can see the scattered plots of age for both males and females. The data points look likestrips. It is difficult to comprehend the distribution of data in this form. To better comprehend thedata, pass True for the jitter parameter which adds some random noise to the data. Look at thefollowing script:
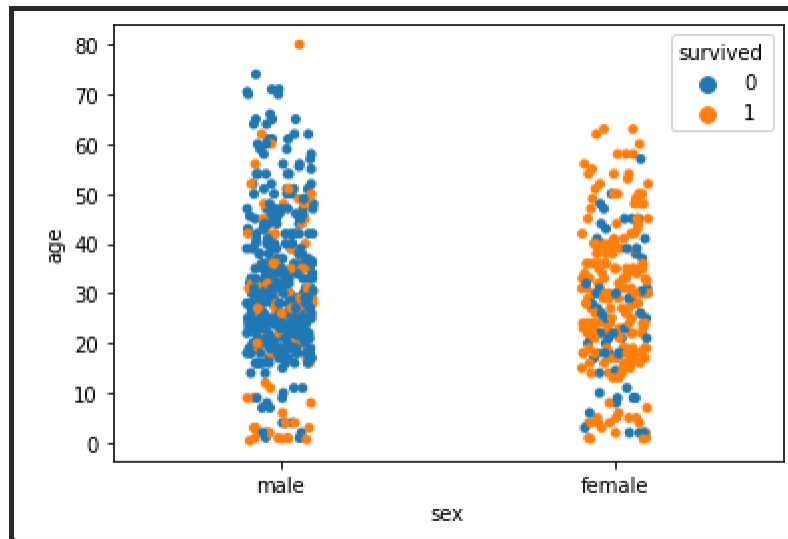
```
sns.stripplot(x='sex', y='age', data=dataset, jitter=True)
```

Now you have a better view for the distribution of age across the genders.

Like violin and box plots, you can add an additional categorical column to strip plot using hueparameter as shown below:

```
sns.stripplot(x='sex', y='age', data=dataset, jitter=True, hue='survived')
```


b. The Swarm Plot

The swarm plot is a combination of the strip and the violin plots. In the swarm plots, the pointsare adjusted in such a way that they don't overlap. Let's plot a swarm plot for the distribution ofageagainstgender.Theswarmplot()functionisusedtoplottheviolinplot.Liketheboxplot,thefirst parameter is the categorical column, the second parameter is the numeric column while thethird parameter is the dataset. Look at the following script:

```
sns.swarmplot(x='sex', y='age', data=dataset)
```
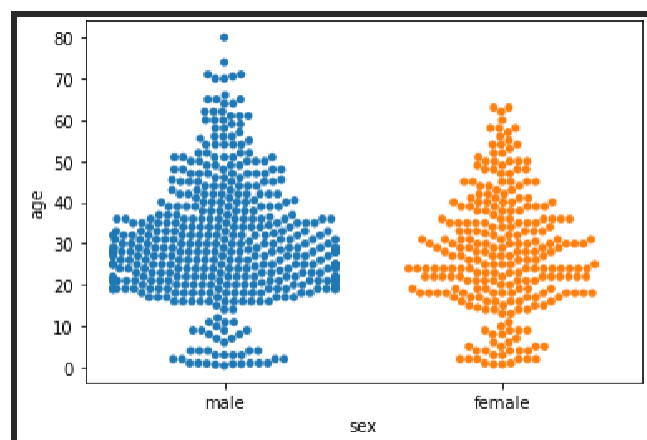
You can clearly see that the above plot contains scattered data points like the strip plot and thedata points are not overlapping. Rather they are arranged to give a view similar to that of a violinplot.

Let'saddanothercategoricalcolumntotheswarmplotusingthehueparameter.

```
sns.swarmplot(x='sex', y='age', data=dataset, hue='survived')
```
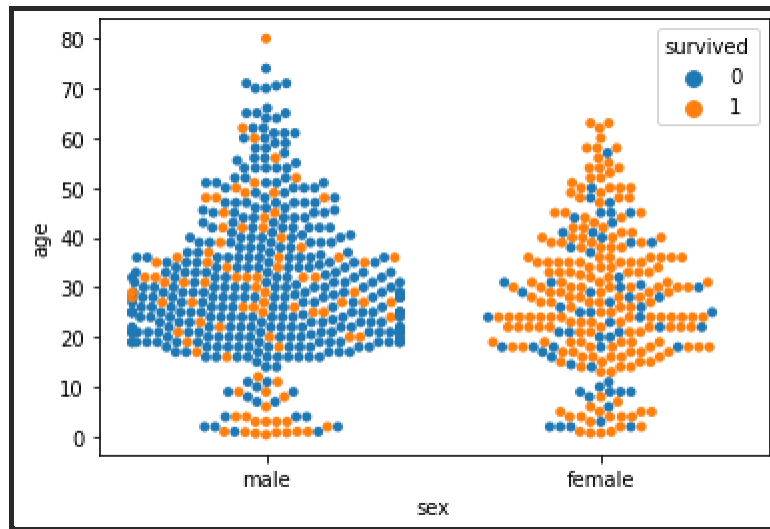
From the output, it is evident that the ratio of surviving males is less than the ratio of survivingfemales. Since for the male plot, there are more blue points and less orange points. On the otherhand, for females, there are more orange points (surviving) than the blue points (not surviving).Another observation is that amongst males of age less than 10, more passengers survived ascompared to those who didn't.

Matrix Plots

Matrix plots are the type of plots that show data in the form of rows and columns. Heat maps arethe prime examples of matrix plots.

a. Heat Maps

Heat maps are normally used to plot correlation between numeric columns in the form of amatrix. It is important to mention here that to draw matrix plots, you need to have meaningfulinformation on rows as well as columns. Let's plot the first five rows of the Titanic dataset to seeif both the rows and column headers have meaningful information. Execute the following script:

```
import pandas as pd
```

```python
import numpy as np

import matplotlib.pyplot as

plt import seaborn as sns

dataset =

sns.load_dataset('titanic')dataset.he

ad()
```

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	who	adult_male	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	man	True	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	woman	False	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	woman	False	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	woman	False	C	Southampton	yes	False
4	0	3	male	35.0	0	0	8.0500	S	Third	man	True	NaN	Southampton	no	True

Fromtheoutput,youcanseethatthecolumnheaderscontainusefulinformationsuchaspassengers surviving, their age, fare etc. However the row headers only contain indexes 0, 1, 2,etc. To plot matrix plots, we need useful information on both columns and row headers. One wayto do this is to call the corr() method on the dataset. The corr() function returns the correlationbetween all the numeric columns of the dataset. Execute the following script:

```python
dataset.corr()
```

Intheoutput,youwillseethatboththecolumnsandtherowshavemeaningfulheaderinformation,   as shown below:

	survived	pclass	age	sibsp	parch	fare	adult_male	alone
survived	1.000000	-0.338481	-0.077221	-0.035322	0.081629	0.257307	-0.557080	-0.203367
pclass	-0.338481	1.000000	-0.369226	0.083081	0.018443	-0.549500	0.094035	0.135207
age	-0.077221	-0.369226	1.000000	-0.308247	-0.189119	0.096067	0.280328	0.198270
sibsp	-0.035322	0.083081	-0.308247	1.000000	0.414838	0.159651	-0.253586	-0.584471
parch	0.081629	0.018443	-0.189119	0.414838	1.000000	0.216225	-0.349943	-0.583398
fare	0.257307	-0.549500	0.096067	0.159651	0.216225	1.000000	-0.182024	-0.271832
adult_male	-0.557080	0.094035	0.280328	-0.253586	-0.349943	-0.182024	1.000000	0.404744
alone	-0.203367	0.135207	0.198270	-0.584471	-0.583398	-0.271832	0.404744	1.000000

Now to create a heat map with these correlation values, you need to call the heatmap() functionand pass it your correlation dataframe. Look at the following script:

```
corr =

dataset.corr()sns.hea

tmap(corr)
```
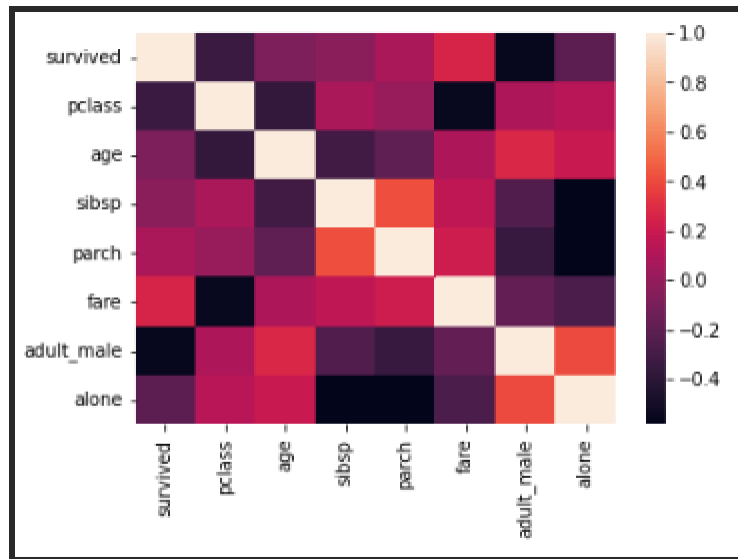

From the output, it can be seen that what heatmap essentially does is that it plots a box for everycombination of rows and column value. The colour of the box depends upon the gradient. Forinstance,intheaboveimageifthereisahighcorrelationbetweentwofeatures,thecorrespondingcello rtheboxiswhite,ontheotherhandifthereisnocorrelation,thecorresponding cell remains black.
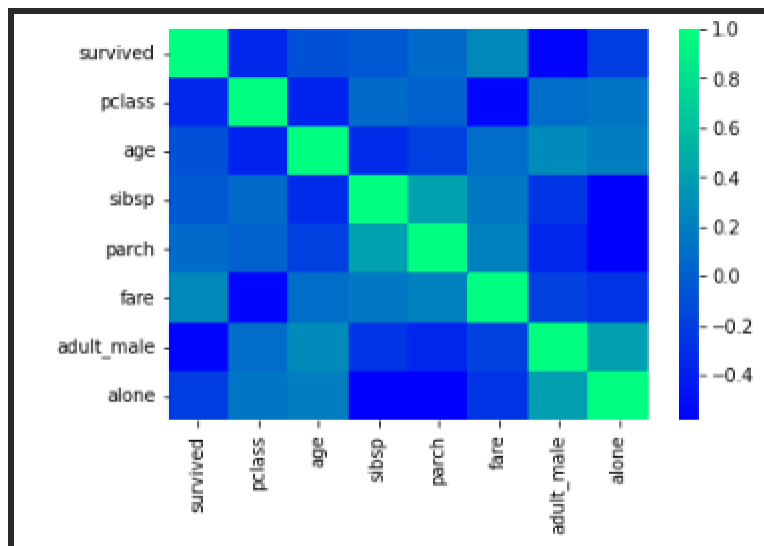
ThecorrelationvaluescanalsobeplottedontheheatmapbypassingTruefortheannotparameter.Execute the following scriptto see this in action:

```
corr =

dataset.corr()sns.heatmap(cor

r,annot=True)
```
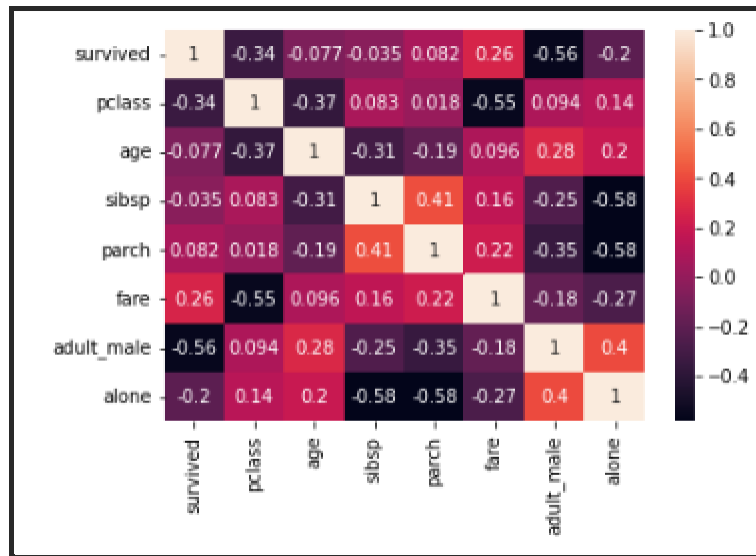
Youcanalsochangethecolouroftheheatmapbypassinganargumentforthecmapparameter.Fornow, just look at thefollowing script:

```
corr =

dataset.corr()sns.hea

tmap(corr)
```


b. **ClusterMap:**

In addition to the heat map, another commonly used matrix plot is the cluster map. Thecluster map basically uses Hierarchical Clustering to cluster the rows and columns of thematrix.

Let'splotaclustermapforthenumberofpassengerswhotravelledinaspecificmonthofaspecific year. Execute the followingscript:
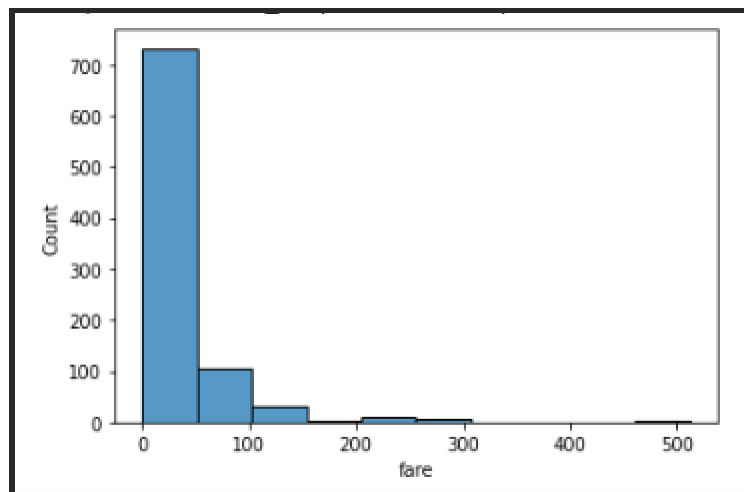
4. **Checkinghowthepriceoftheticket(columnname:'fare')foreachpassengerisdistributed by plotting a histogram.**

```
import seaborn as sns

dataset =

sns.load_dataset('titanic')sns.histplot(dataset['

fare'],kde=False,bins=10)
```


Fromthehistogram,itisseenthatforaround730passengersthepriceofthe ticketis50.For 100 passengers the price of the ticket is 100 and so on.

Conclusion-

Seabornisanadvanceddatavisualisationlibrarybuilt on top of Matplotlib library. In thisassignment, we looked at how we can draw distributional and categorical plots using the Seabornlibrary.WehaveseenhowtoplotmatrixplotsinSeaborn.Wealsosawhowtochangeplotstylesan d use grid functions to manipulate subplots.

Assignment Questions

1. **Listout differenttypes ofplot tofind patterns ofdata**
2. **Explain when you will use distribution plots and when you will use categorical plots.**
3. **Writetheconclusionfromthefollowingswarmplot(considertitanicdataset)**
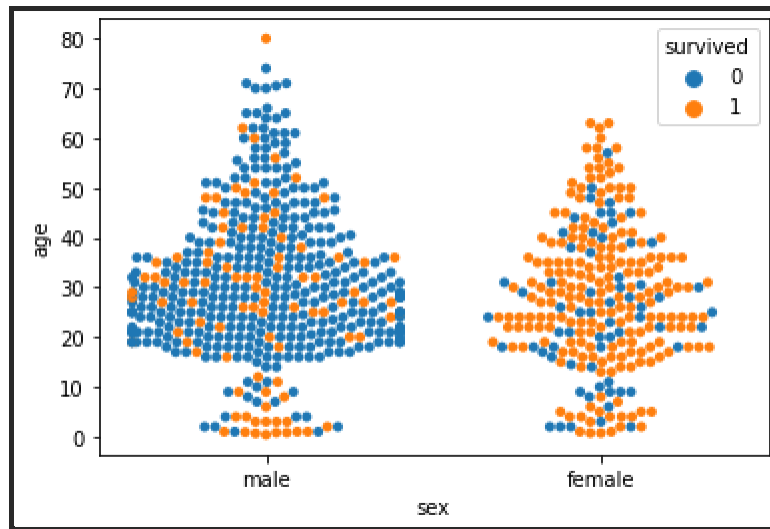
4. **Whichparameterisusedtoaddanothercategoricalvariabletotheviolinplot,Explain with syntax and example.**