Assignment No: 9

Aim/Problem Statement:- Data Visualization II

Use the inbuilt dataset 'titanic' as used in the above problem. Plot a box plot for distribution of age with respect to each gender along with the information about whether they survived or not. (Column names : 'sex' and 'age')

Write observations on the inference from the above statistics.

Theory:-

Why data visualization is important

It's hard to think of a professional industry that doesn't benefit from making data more understandable. Every STEM field benefits from understanding data—and so do fields in government, finance, marketing, history, consumer goods, service industries, education, sports, and so on. While we'll always wax poetically about data visualization (you're on the Tableau website, after all) there are practical, real-life applications that are undeniable. And, since visualization is so prolific, it's also one of the most useful professional skills to develop. The better you can convey your points visually, whether in a dashboard or a slide deck, the better you can leverage that information. The concept of the citizen data scientist is on the rise. Skill sets are changing to accommodate a data-driven world. It is increasingly valuable for professionals to be able to use data to make decisions and use visuals to tell stories of when data informs the who, what, when, where, and how. While traditional education typically draws a distinct line between creative storytelling and technical analysis, the modern professional world also values those who can cross between the two: data visualization sits right in the middle of analysis and visual storytelling.

The different types of visualizations

When you think of data visualization, your first thought probably immediately goes to simple bar graphs or pie charts. While these may be an integral part of visualizing data and a common baseline for many data graphics, the right visualization must be paired with the right set of information. Simple graphs are only the tip of the iceberg. There's a whole selection of visualization methods to present data in effective and interesting ways. **Common general types of data visualization:**

- Charts
- Tables
- Graphs
- Maps
- Infographics

- Dashboards

More specific examples of methods to visualize data:

- Area Chart
- Bar Chart
- Box-and-whisker Plots
- Bubble Cloud
- Bullet Graph
- Cartogram
- Circle View
- Dot Distribution Map
- Gantt Chart
- Heat Map
- Highlight Table
- Histogram
- Matrix
- Network
- Polar Area
- Radial Tree
- Scatter Plot (2D or 3D)

Algorithm:-

Step 1: Download the data set of Titanic

Step 2: Importing Libraries

```
import math #library for mathematical calculation
import numpy as np #library for scientific computing
import pandas as pd #library for easy-to-use data structures and data analysis tools
import matplotlib.pyplot as plt #library for plotting
import seaborn as sns #library for plotting
```

Step 3: Reading of dataset

```
df = pd.read_csv('train.csv')
```

Step 4: Print details of the dataset

```
print('_'*50)
print('*'*50)
print(df.info(memory_usage=False))
print('_'*50)
```

Step 5: Finding Null values

```
df.isnull().sum()
```

Step 6: Dealing with missing values

```
#drop passengerID, Name, Cabin columns
df = df.drop(['PassengerId', 'Name', 'Cabin'], axis=1)

#drop rows which has nan value
df = df.dropna(axis=0, how='any')
```

Step 7: map 0 to not survived and 1 to survived in survived column

```
df['Survived'] = df['Survived'].map({0: 'Not Survived', 1: 'Survived'})
```

Step 8: group passengers into bins of children, teenager, adult, senior citizen…

```
df['Age'] = pd.cut(df['Age'], bins=[-1, 5, 19, 60, 150], labels=['Children','Teenager','Adult','Senior Citizen'],right=True)
```

Step 9: Set the plot

```
sns.set(context='notebook',style='whitegrid',font_scale=1.5)
```

Step 10: The two methods below is for displaying the count value on top of the bar graph

```
def assignAxis(df,g):
    # Get current axis on current figure
    for i in range(0,g.axes.size):
        ax = g.fig.get_axes()[i]
        displayCount(df,ax)

def displayCount(df,ax):
    # ylim max value to be set
    y_max = df.value_counts().max() + 75
    ax.set_ylim(top=y_max)

    # Iterate through the list of axes' patches
    for p in ax.patches:
        #checks if there index has 0 count
        if(math.isnan(p.get_height())):
            ax.text(p.get_x() + p.get_width()/2, 0, 0,
                fontsize=12, color='red', ha='center', va='bottom')
            continue
        else:
            ax.text(p.get_x() + p.get_width()/2, p.get_height(), int(p.get_height()),
```

```
        fontsize=12, color='red', ha='center', va='bottom')#number

        of males and females
        g = sns.factorplot(x='Sex', data=df,kind='count',size=4, aspect=.8,alpha=0.7,
                palette='muted').set(xlabel='Gender',ylabel='Count',title='Number of Male and
                Female')assignAxis(df['Sex'],g)
```

Step 11: #number of males and females based on Age

```
        g = sns.factorplot(x='Age', data=df,kind='count',size=4, aspect=1.8, hue='Sex',alpha=0.7,
                palette='muted').set(xlabel='Age',ylabel='Count',title='Gender Distribution by Age')

        assignAxis(df['Age'],g)
```

Step 12: People Survived based on gender and age

```
        g = sns.factorplot(x='Survived', data=df,kind='count',size=4, aspect=1.2,hue='Age',col='Sex',alpha=0.7,
                palette='muted',order=['Survived','Not
        Survived'],legend_out=True).set(xlabel='Survival',ylabel='Count')
        g.fig.subplots_adjust(wspace=.3)

        assignAxis(df['Survived'],g)
```

Conclusion: Implemented successfully Simple Data visualization techniques using Python on Titanic dataset.