import plotly.express as px

df = px.data.tips()

print(df)

A) Bubble Chart

fig = px.scatter(df, x="size", y="total\_bill",

color="day",

size='tip',

hover\_data=['sex'])

fig.show()

B) Scatter Plot

fig1 = px.scatter(df, x="total\_bill", y="tip", color="time")

C) Heatmaps

hd = []

ld = []

dd = []

tld = []

tdd =[]

fld = []

fdd =[]

sald = []

sadd =[]

suld = []

sudd =[]

for i in range(244):

if df["time"][i] == 'Lunch':

if df["day"][i]== 'Thur':

tld.append(df["size"][i])

elif df["day"][i]== 'Fri':

fld.append(df["size"][i])

elif df["day"][i]== 'Sat':

sald.append(df["size"][i])

else:

suld.append(df["size"][i])

else:

if df["day"][i]== 'Thur':

tdd.append(df["size"][i])

elif df["day"][i]== 'Fri':

fdd.append(df["size"][i])

elif df["day"][i]== 'Sat':

sadd.append(df["size"][i])

else:

sudd.append(df["size"][i])

ld.append(sum(tld))

ld.append(sum(fld))

ld.append(sum(sald))

ld.append(sum(suld))

dd.append(sum(tdd))

dd.append(sum(fdd))

dd.append(sum(sadd))

dd.append(sum(sudd))

hd.append(ld)

hd.append(dd)

fig = px.imshow(hd,

labels=dict(x="day", y="time", color="size"),

x=['Thur', 'Fri','Sat','Sun'],

y=['Lunch','Dinner']

)

fig.update\_xaxes(side="top")

fig.show()

D) Histogram

fig = px.histogram(df, x="total\_bill")

fig.show()

E) Boxplot

df["total\_bill"].describe()]

fig = px.box(df, y="total\_bill")

fig.show()

F) Density Plot

import matplotlib.pyplot as plt

# density plot for 'tip'

df['tip'].plot.density(color='green')

plt.title('Density Plot for Tip')

plt.show()

G) Ogives less than

import numpy as np

data = df["tip"]

print(max(data))

print(min(data))

# creating class interval

classInterval = [0,1,2,3,4,5,6,7,8,9,10]

# calculating frequency and class interval

values, base = np.histogram(data, bins=classInterval)

# calculating cumulative sum

cumsum = np.cumsum(values)

# plotting the ogive graph

plt.plot(base[1:], cumsum, color='red', marker='o', linestyle='-')

# formatting

plt.title('Ogive Graph')

plt.xlabel('Tip')

plt.ylabel('Cumulative Frequency')

H) Ogives more than

# reversing cumulative frequency

res = np.flipud(cumsum)

# plotting ogive

plt.plot(base[1:], res, color='brown', marker='o', linestyle='-')

# formatting the graph

plt.title('Ogive Graph')

plt.xlabel('Tip')

plt.ylabel('Cumulative Frequency')

I) Pie Chart

cd = [sum(ld),sum(dd)]

time = ['Lunch','Dinner']

plt.pie(cd, labels = time)

plt.title("Number of customers during Lunch and Dinner")

plt.show()

J) Line Chart

fig = px.line(df, x=range(244), y="tip", title='Tips per customer')

fig.show()