Code I have used

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

data=pd.read\_csv("banking\_data.csv")

data

list1=data.age.unique()

list1

no=[]

for i in list1:

  x=data["age"].loc[data["age"]==i].count()

  no.append(x)

no

plt.scatter(list1,no,color="r")

plt.ylabel("Number of people")

plt.xlabel("Age")

plt.show()

jobs=data["job"].unique()

job\_no=[]

for i in jobs:

  x=data["job"].loc[data["job"]==i].count()

  job\_no.append(x)

print(job\_no)

print(jobs)

data["job"].value\_counts()

plt.bar(jobs,job\_no)

plt.xticks(fontsize=10,rotation=90)

for index,height in enumerate(job\_no):

    plt.text(index,height,int(height),ha='center',va='bottom',size=8)

plt.ylabel("Number of people")

plt.xlabel("Job types")

plt.show()

mate=data["marital"].unique()

mate\_no=[]

for i in mate:

  x=data["marital"].loc[(data["marital"]==i)&(data["marital"]!=np.empty)].count()

  mate\_no.append(x)

print(mate\_no)

print(mate)

mate\_no.pop()

mate\_no

mate1=np.delete(mate,-1)

mate1

plt.pie(mate\_no,labels=mate1,autopct='%1.1f%%')

plt.show()

data['education'].isnull().sum()

data['education']=data['education'].fillna(data['education'].mode()[0])

data['education'].isnull().sum()

data['education'].value\_counts()

ax=sns.countplot(x=data['education'],color="r")

ax=ax.bar\_label(ax.containers[0])

plt.show()

data['default'].value\_counts()

data['default'].isnull().sum()

plt.pie(data["default"].value\_counts(),labels=["No","yes"],autopct='%1.1f%%')

plt.show()

balance\_unique=data['balance'].unique()

balance\_unique

Count\_balance=[]

for i in balance\_unique:

  x=data['balance'].loc[data['balance']==i].count()

  Count\_balance.append(x)

Count\_balance

plt.scatter(balance\_unique,Count\_balance)

plt.ylabel("Number of people")

plt.xlabel("Balance")

plt.show()

Q1=data['balance'].quantile(0.25)

Q3=data['balance'].quantile(0.75)

IQR=Q3-Q1

lower\_bound=Q1-(1.5\*IQR)

upper\_bound=Q3+(1.5\*IQR)

data1=data[(data["balance"]>=lower\_bound)&(data["balance"]<=upper\_bound)]

data1

sns.boxplot(data1['balance'],showmeans=True,notch=True)

plt.show()

bx=sns.countplot(x=data['housing'],color="r")

bx=bx.bar\_label(bx.containers[0])

plt.show()

bx=sns.countplot(x=data['loan'],color="r")

bx=bx.bar\_label(bx.containers[0])

plt.show()

bx=sns.countplot(x=data['contact'],color="r")

bx=bx.bar\_label(bx.containers[0])

plt.show()

plt.pie(data['contact'].value\_counts(),labels=data['contact'].unique(),autopct='%1.1f%%')

plt.show()

sns.countplot(x="day",data=data,color="g")

plt.xticks([1,5,10,15,20,25,30])

plt.show()

sns.countplot(x="month",data=data,color="g")

plt.show()

plt.scatter(data["duration"].unique(),data["duration"].value\_counts())

plt.xlabel("Duration of last contact(seconds)")

plt.ylabel("Number of people")

plt.show()

y=data.loc[:,["previous"]]

y

y=y.loc[y.previous!=0]

y

plt.scatter(y.previous.unique(),y.previous.value\_counts())

plt.xlabel("Number of contacts")

plt.ylabel("Number of people")

plt.show()

x=data.loc[:,["pdays"]]

x

x=x.loc[x.pdays!=-1]

x

plt.bar(x.pdays.unique(),x.pdays.value\_counts())

plt.xlabel("Number of days passed after last contacted")

plt.ylabel("Number of people contacted")

plt.show()

sns.countplot(x="poutcome",data=data,color="g")

plt.show()

ax=sns.countplot(x="y",data=data,color="g")

ax=ax.bar\_label(ax.containers[0])

plt.xlabel("Subscribed")

plt.ylabel("Number of people")

plt.show()

from sklearn.preprocessing import LabelEncoder

le=LabelEncoder()

data["y"]=le.fit\_transform(data["y"])

data

a=data.select\_dtypes(include=[np.number])

b=a.corr()

b

import seaborn as sns

sns.heatmap(b, annot=True, cmap="coolwarm")

plt.title("correlation matrix")

plt.tight\_layout()

plt.show()