Food_delivery_costs

July 30, 2024

[1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
df = pd.read_csv('/content/food delivery costs.csv')
df.head()
```

[2]:

	Order ID	Customer ID	Restaurant ID	Order Date and Time	Delivery Date and Time	Order Value	Delivery Fee	Payment Method	Discounts and Offers	Commission Fee	Payment Processing Fee	Refunds/Chargebacks
0	1	C8270	R2924	2024-02-01 01:11:52	2024-02-01 02:39:52	1914	0	Credit Card	5% on App	150	47	0
1	2	C1860	R2054	2024-02-02 22:11:04	2024-02-02 22:46:04	986	40	Digital Wallet	10%	198	23	0
2	3	C6390	R2870	2024-01-31 05:54:35	2024-01-31 06:52:35	937	30	Cash on Delivery	15% New User	195	45	0
3	4	C6191	R2642	2024-01-16 22:52:49	2024-01-16 23:38:49	1463	50	Cash on Delivery	NaN	146	27	0
4	5	C6734	R2799	2024-01-29 01:19:30	2024-01-29 02:48:30	1992	30	Cash on Delivery	50 off Promo	130	50	0

[48]:
```
df.tail()
```

[48]:
```
     Order ID Customer ID Restaurant ID Order Date and Time  \
995       996       C6232         R2129 2024-01-14 05:57:00
996       997       C6797         R2742 2024-01-28 08:50:43
997       998       C5926         R2837 2024-01-21 09:43:19
998       999       C7016         R2144 2024-01-30 22:23:38
999      1000       C4335         R2890 2024-01-08 14:46:43

    Delivery Date and Time  Order Value  Delivery Fee    Payment Method  \
995    2024-01-14 06:39:00          825             0    Digital Wallet
996    2024-01-28 10:10:43         1627            50  Cash on Delivery
997    2024-01-21 10:44:19          553            20  Cash on Delivery
998    2024-01-31 00:07:38         1414             0  Cash on Delivery
999    2024-01-08 15:39:43         1657            20    Digital Wallet

     Discounts and Offers  Commission Fee  Payment Processing Fee  \
995                 41.25             165                      47
996                  0.00             110                      42
997                  0.00              64                      31
998                212.10             199                      34
999                248.55             180                      27

     Refunds/Chargebacks   Costs  Profit
995                   50   88.25   76.75
996                    0   92.00   18.00
997                    0   51.00   13.00
998                    0  246.10  -47.10
999                  100  295.55 -115.55
```

[50]:
```
df.shape
```

[50]:
```
(1000, 14)
```

[55]:
```
df.duplicated().sum()
```

[55]:
```
0
```

[56]:
```
df.isnull().sum()
```

[56]:
```
Order ID                  0
Customer ID               0
Restaurant ID             0
Order Date and Time       0
Delivery Date and Time    0
Order Value               0
Delivery Fee              0
Payment Method            0
Discounts and Offers      0
```

```
Commission Fee            0
Payment Processing Fee    0
Refunds/Chargebacks       0
Costs                     0
Profit                    0
dtype: int64
```

[5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 12 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Order ID                1000 non-null   int64
 1   Customer ID             1000 non-null   object
 2   Restaurant ID           1000 non-null   object
 3   Order Date and Time     1000 non-null   object
 4   Delivery Date and Time  1000 non-null   object
 5   Order Value             1000 non-null   int64
 6   Delivery Fee            1000 non-null   int64
 7   Payment Method          1000 non-null   object
 8   Discounts and Offers    815 non-null    object
 9   Commission Fee          1000 non-null   int64
 10  Payment Processing Fee  1000 non-null   int64
 11  Refunds/Chargebacks     1000 non-null   int64
dtypes: int64(6), object(6)
memory usage: 93.9+ KB
```

[6]:
```
df.describe()
```

[6]:

	Order ID	Order Value	Delivery Fee	Commission Fee	\
count	1000.000000	1000.000000	1000.000000	1000.00000	
mean	500.500000	1053.969000	28.620000	126.99000	
std	288.819436	530.975339	16.958278	43.06405	
min	1.000000	104.000000	0.000000	50.00000	
25%	250.750000	597.750000	20.000000	90.00000	
50%	500.500000	1038.500000	30.000000	127.00000	
75%	750.250000	1494.000000	40.000000	164.00000	
max	1000.000000	1995.000000	50.000000	200.00000	

	Payment Processing Fee	Refunds/Chargebacks
count	1000.000000	1000.000000
mean	29.832000	28.300000
std	11.627165	49.614228
min	10.000000	0.000000
25%	20.000000	0.000000

```
50%                 30.000000             0.000000
75%                 40.000000            50.000000
max                 50.000000           150.000000
```

####**Cleaning**

[7]:
```
df["Order Date and Time"] = pd.to_datetime(df["Order Date and Time"])
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 12 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Order ID                1000 non-null   int64
 1   Customer ID             1000 non-null   object
 2   Restaurant ID           1000 non-null   object
 3   Order Date and Time     1000 non-null   datetime64[ns]
 4   Delivery Date and Time  1000 non-null   object
 5   Order Value             1000 non-null   int64
 6   Delivery Fee            1000 non-null   int64
 7   Payment Method          1000 non-null   object
 8   Discounts and Offers    815 non-null    object
 9   Commission Fee          1000 non-null   int64
 10  Payment Processing Fee  1000 non-null   int64
 11  Refunds/Chargebacks     1000 non-null   int64
dtypes: datetime64[ns](1), int64(6), object(5)
memory usage: 93.9+ KB
```

[9]:
```
df["Delivery Date and Time"] = pd.to_datetime(df["Delivery Date and Time"])
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 12 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   Order ID                1000 non-null   int64
 1   Customer ID             1000 non-null   object
 2   Restaurant ID           1000 non-null   object
 3   Order Date and Time     1000 non-null   datetime64[ns]
 4   Delivery Date and Time  1000 non-null   datetime64[ns]
 5   Order Value             1000 non-null   int64
 6   Delivery Fee            1000 non-null   int64
 7   Payment Method          1000 non-null   object
 8   Discounts and Offers    815 non-null    object
 9   Commission Fee          1000 non-null   int64
 10  Payment Processing Fee  1000 non-null   int64
```

```
 11  Refunds/Chargebacks     1000 non-null   int64
dtypes: datetime64[ns](2), int64(6), object(4)
memory usage: 93.9+ KB
```

[10]:
```
df.head()
```

[10]:
```
   Order ID Customer ID Restaurant ID Order Date and Time  \
0         1       C8270         R2924 2024-02-01 01:11:52
1         2       C1860         R2054 2024-02-02 22:11:04
2         3       C6390         R2870 2024-01-31 05:54:35
3         4       C6191         R2642 2024-01-16 22:52:49
4         5       C6734         R2799 2024-01-29 01:19:30

  Delivery Date and Time  Order Value  Delivery Fee    Payment Method  \
0    2024-02-01 02:39:52         1914             0       Credit Card
1    2024-02-02 22:46:04          986            40    Digital Wallet
2    2024-01-31 06:52:35          937            30  Cash on Delivery
3    2024-01-16 23:38:49         1463            50  Cash on Delivery
4    2024-01-29 02:48:30         1992            30  Cash on Delivery

  Discounts and Offers  Commission Fee  Payment Processing Fee  \
0            5% on App             150                      47
1                  10%             198                      23
2         15% New User             195                      45
3                  NaN             146                      27
4         50 off Promo             130                      50

   Refunds/Chargebacks
0                    0
1                    0
2                    0
3                    0
4                    0
```

[14]:
```
def extract(value):
    a = str(value).split(" ")
    return a[0]

df["Discounts and Offers"] = df["Discounts and Offers"].apply(extract)
df.head()
```

[14]:
```
   Order ID Customer ID Restaurant ID Order Date and Time  \
0         1       C8270         R2924 2024-02-01 01:11:52
1         2       C1860         R2054 2024-02-02 22:11:04
2         3       C6390         R2870 2024-01-31 05:54:35
3         4       C6191         R2642 2024-01-16 22:52:49
4         5       C6734         R2799 2024-01-29 01:19:30
```

```
  Delivery Date and Time  Order Value  Delivery Fee    Payment Method  \
0    2024-02-01 02:39:52         1914             0       Credit Card
1    2024-02-02 22:46:04          986            40    Digital Wallet
2    2024-01-31 06:52:35          937            30  Cash on Delivery
3    2024-01-16 23:38:49         1463            50  Cash on Delivery
4    2024-01-29 02:48:30         1992            30  Cash on Delivery

  Discounts and Offers  Commission Fee  Payment Processing Fee  \
0                   5%             150                      47
1                  10%             198                      23
2                  15%             195                      45
3                  nan             146                      27
4                   50             130                      50

   Refunds/Chargebacks
0                    0
1                    0
2                    0
3                    0
4                    0
```

[26]:
```
def removep(value):
    if "%" in value:
        a = value.replace("%","")
        return float(a)
    else:
        return float(value)

df["Discounts and Offers"] = df["Discounts and Offers"].apply(removep)
df.head()
```

[26]:
```
   Order ID Customer ID Restaurant ID Order Date and Time  \
0         1       C8270         R2924 2024-02-01 01:11:52
1         2       C1860         R2054 2024-02-02 22:11:04
2         3       C6390         R2870 2024-01-31 05:54:35
3         4       C6191         R2642 2024-01-16 22:52:49
4         5       C6734         R2799 2024-01-29 01:19:30

  Delivery Date and Time  Order Value  Delivery Fee    Payment Method  \
0    2024-02-01 02:39:52         1914             0       Credit Card
1    2024-02-02 22:46:04          986            40    Digital Wallet
2    2024-01-31 06:52:35          937            30  Cash on Delivery
3    2024-01-16 23:38:49         1463            50  Cash on Delivery
4    2024-01-29 02:48:30         1992            30  Cash on Delivery

   Discounts and Offers  Commission Fee  Payment Processing Fee  \
```

```
0                    5.0              150                        47
1                   10.0              198                        23
2                   15.0              195                        45
3                    NaN              146                        27
4                   50.0              130                        50

   Refunds/Chargebacks
0                    0
1                    0
2                    0
3                    0
4                    0
```

[29]:
```
df.loc[(df["Discounts and Offers"] <= 15),"Discounts and Offers" ] =␣
 ↪(df["Discounts and Offers"]/100) * df["Order Value"]
df.head()
```

[29]:
```
   Order ID Customer ID Restaurant ID Order Date and Time  \
0         1       C8270         R2924 2024-02-01 01:11:52
1         2       C1860         R2054 2024-02-02 22:11:04
2         3       C6390         R2870 2024-01-31 05:54:35
3         4       C6191         R2642 2024-01-16 22:52:49
4         5       C6734         R2799 2024-01-29 01:19:30

  Delivery Date and Time  Order Value  Delivery Fee    Payment Method  \
0    2024-02-01 02:39:52         1914             0       Credit Card
1    2024-02-02 22:46:04          986            40    Digital Wallet
2    2024-01-31 06:52:35          937            30  Cash on Delivery
3    2024-01-16 23:38:49         1463            50  Cash on Delivery
4    2024-01-29 02:48:30         1992            30  Cash on Delivery

   Discounts and Offers  Commission Fee  Payment Processing Fee  \
0                 95.70              150                        47
1                 98.60              198                        23
2                140.55              195                        45
3                   NaN              146                        27
4                 50.00              130                        50

   Refunds/Chargebacks
0                    0
1                    0
2                    0
3                    0
4                    0
```

[30]:
```
df["Discounts and Offers"]=df["Discounts and Offers"].fillna(0)
df.head()
```

[30]:
```
   Order ID Customer ID Restaurant ID Order Date and Time  \
0         1       C8270         R2924 2024-02-01 01:11:52
1         2       C1860         R2054 2024-02-02 22:11:04
2         3       C6390         R2870 2024-01-31 05:54:35
3         4       C6191         R2642 2024-01-16 22:52:49
4         5       C6734         R2799 2024-01-29 01:19:30

  Delivery Date and Time  Order Value  Delivery Fee    Payment Method  \
0    2024-02-01 02:39:52         1914             0       Credit Card
1    2024-02-02 22:46:04          986            40    Digital Wallet
2    2024-01-31 06:52:35          937            30  Cash on Delivery
3    2024-01-16 23:38:49         1463            50  Cash on Delivery
4    2024-01-29 02:48:30         1992            30  Cash on Delivery

   Discounts and Offers  Commission Fee  Payment Processing Fee  \
0                 95.70             150                      47
1                 98.60             198                      23
2                140.55             195                      45
3                  0.00             146                      27
4                 50.00             130                      50

   Refunds/Chargebacks
0                    0
1                    0
2                    0
3                    0
4                    0
```

[33]:
```
df["Costs"] = df["Delivery Fee"] + df['Discounts and Offers'] + df["Payment␣
 ↪Processing Fee"]
df.head()
```

[33]:
```
   Order ID Customer ID Restaurant ID Order Date and Time  \
0         1       C8270         R2924 2024-02-01 01:11:52
1         2       C1860         R2054 2024-02-02 22:11:04
2         3       C6390         R2870 2024-01-31 05:54:35
3         4       C6191         R2642 2024-01-16 22:52:49
4         5       C6734         R2799 2024-01-29 01:19:30

  Delivery Date and Time  Order Value  Delivery Fee    Payment Method  \
0    2024-02-01 02:39:52         1914             0       Credit Card
1    2024-02-02 22:46:04          986            40    Digital Wallet
2    2024-01-31 06:52:35          937            30  Cash on Delivery
3    2024-01-16 23:38:49         1463            50  Cash on Delivery
4    2024-01-29 02:48:30         1992            30  Cash on Delivery

   Discounts and Offers  Commission Fee  Payment Processing Fee  \
```

```
0                 95.70            150                      47
1                 98.60            198                      23
2                140.55            195                      45
3                  0.00            146                      27
4                 50.00            130                      50

   Refunds/Chargebacks   Costs
0                    0  142.70
1                    0  161.60
2                    0  215.55
3                    0   77.00
4                    0  130.00
```

[35]:
```
df["Profit"] = df["Commission Fee"] - df['Costs']
df.head()
```

[35]:
```
   Order ID Customer ID Restaurant ID Order Date and Time  \
0         1       C8270         R2924 2024-02-01 01:11:52
1         2       C1860         R2054 2024-02-02 22:11:04
2         3       C6390         R2870 2024-01-31 05:54:35
3         4       C6191         R2642 2024-01-16 22:52:49
4         5       C6734         R2799 2024-01-29 01:19:30

  Delivery Date and Time  Order Value  Delivery Fee    Payment Method  \
0    2024-02-01 02:39:52         1914             0       Credit Card
1    2024-02-02 22:46:04          986            40    Digital Wallet
2    2024-01-31 06:52:35          937            30  Cash on Delivery
3    2024-01-16 23:38:49         1463            50  Cash on Delivery
4    2024-01-29 02:48:30         1992            30  Cash on Delivery

   Discounts and Offers  Commission Fee  Payment Processing Fee  \
0                 95.70            150                      47
1                 98.60            198                      23
2                140.55            195                      45
3                  0.00            146                      27
4                 50.00            130                      50

   Refunds/Chargebacks   Costs  Profit
0                    0  142.70    7.30
1                    0  161.60   36.40
2                    0  215.55  -20.55
3                    0   77.00   69.00
4                    0  130.00    0.00
```

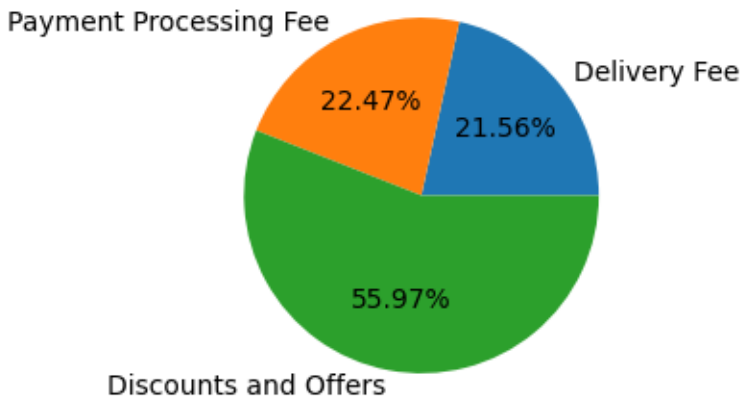
[38]:
```
df["Profit"].sum()
```

[38]:
```
-5751.85
```

[40]:
```
cost_dist = df[["Delivery Fee", "Payment Processing Fee", "Discounts and␣
↪Offers"]].sum()
cost_dist
```

[40]:
```
Delivery Fee              28620.00
Payment Processing Fee    29832.00
Discounts and Offers      74289.85
dtype: float64
```

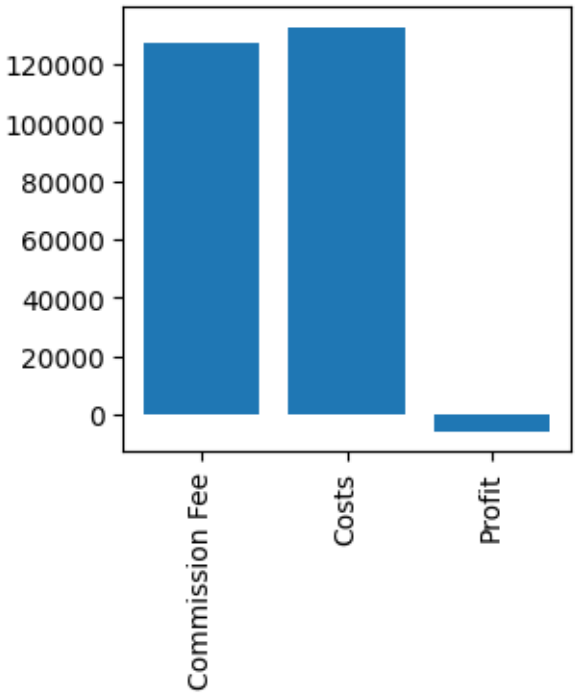
[43]:
```
plt.figure(figsize = (3,3))
plt.pie(cost_dist, labels = cost_dist.index, autopct = "%1.2f%%")
plt.show()
```



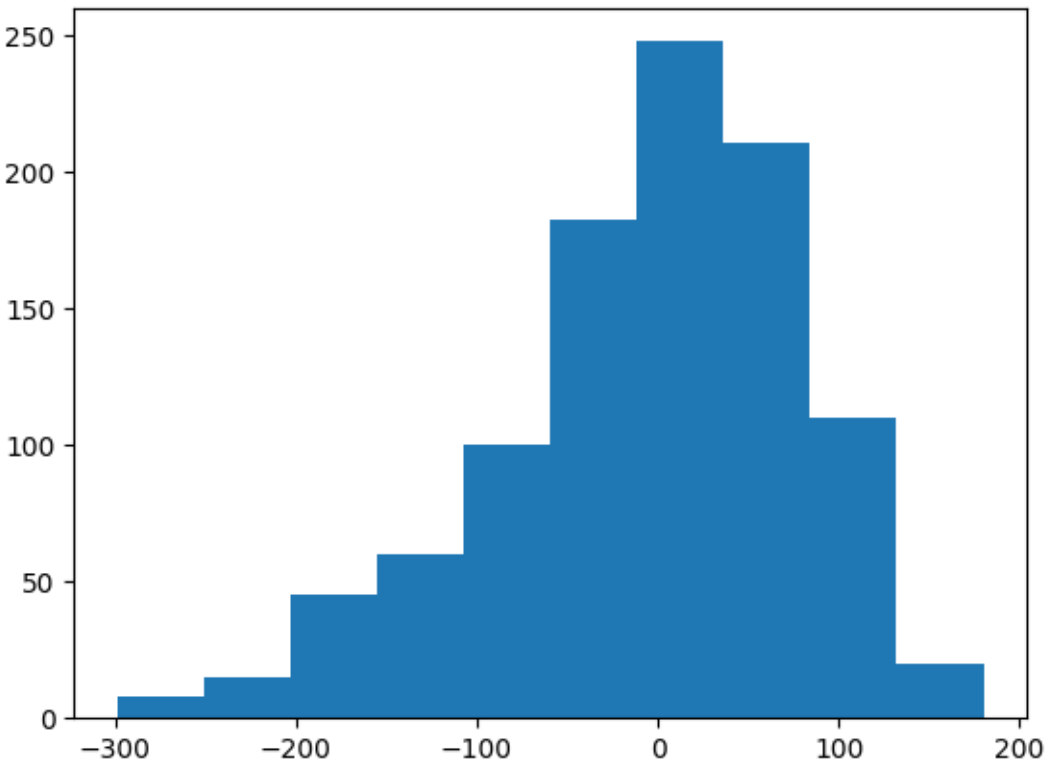
[44]:
```
abc = df[["Commission Fee", "Costs", "Profit"]].sum()
abc
```

[44]:
```
Commission Fee    126990.00
Costs             132741.85
Profit             -5751.85
dtype: float64
```

[46]:
```
plt.figure(figsize = (3,3))
plt.bar(abc.index, abc)
plt.xticks(rotation = 90)
plt.show()
```



[47]:
```
plt.hist(df["Profit"])
plt.show()
```


250
200
150
100
50
0
−300
−200
−100
0
100
200

[ ]: