Affinity

April 10, 2025

[104]:

```python
# import libraries

import pandas as pd
import numpy as np
from sklearn.cluster import KMeans
import matplotlib.pyplot as plt
import seaborn as sns
```

[106]:

```python
# loading orders dataset

orders = pd.read_csv('orders.csv', low_memory=False)
```

[107]:

```python
# previewing basic info

print(orders.columns)
orders.info()
```

```
Index(['Unnamed: 0', 'orderId', 'clientId', 'lastModified', 'status',
       'truckKey', 'location', 'fee', 'volume(kg)', 'deliveryItem',
       'dateGenerated'],
      dtype='object')
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 526945 entries, 0 to 526944
Data columns (total 11 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   Unnamed: 0    526945 non-null  int64
 1   orderId       526945 non-null  object
 2   clientId      525702 non-null  object
 3   lastModified  526945 non-null  object
 4   status        526945 non-null  object
 5   truckKey      216170 non-null  object
 6   location      526945 non-null  object
 7   fee           526945 non-null  float64
 8   volume(kg)    526945 non-null  float64
 9   deliveryItem  526945 non-null  object
 10  dateGenerated 526945 non-null  object
dtypes: float64(2), int64(1), object(8)
```

```
memory usage: 44.2+ MB
```

[110]:
```
# Previewing first few lines

orders.head()
```

[110]:
```
   Unnamed: 0                     orderId    clientId         lastModified  \
0           0  01GGAPGS3BZKF3AKKRRHPBCMMG   320232240  2025-03-02 18:20:59
1           1  01GGAPK4BCXD9S96QXFYD65E20  2420001880  2025-03-02 18:20:59
2           2  01GGAPRNS3ET0JC79QTAX8PG46   043730707  2025-03-02 18:20:59
3           3  01GGAPX6EG5S77Y5B39SHNXEAM   028737136  2025-03-02 18:20:59
4           4  01GGAPYCWBVZQEFFM0DA1J9VT3   038457808  2025-03-02 18:20:59

    status truckKey  location     fee  volume(kg) deliveryItem  \
0  PENDING      NaN  KINTAMPO  6400.0       340.0          BED
1  PENDING      NaN  KINTAMPO   320.0        17.0          BED
2  PENDING      NaN  KINTAMPO  3200.0       170.0          BED
3  PENDING      NaN  KINTAMPO   640.0        34.0          BED
4  PENDING      NaN  KINTAMPO  1600.0        85.0          BED

         dateGenerated
0  2025-03-02 18:20:59
1  2025-03-02 18:20:59
2  2025-03-02 18:20:59
3  2025-03-02 18:20:59
4  2025-03-02 18:20:59
```

[112]:
```
#dropping Unnamed: 0 column

orders.drop(columns=['Unnamed: 0'], inplace=True)
```

[114]:
```
orders.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 526945 entries, 0 to 526944
Data columns (total 10 columns):
 #   Column        Non-Null Count   Dtype
---  ------        --------------   -----
 0   orderId       526945 non-null  object
 1   clientId      525702 non-null  object
 2   lastModified  526945 non-null  object
 3   status        526945 non-null  object
 4   truckKey      216170 non-null  object
 5   location      526945 non-null  object
 6   fee           526945 non-null  float64
 7   volume(kg)    526945 non-null  float64
 8   deliveryItem  526945 non-null  object
 9   dateGenerated 526945 non-null  object
```

```
dtypes: float64(2), object(8)
memory usage: 40.2+ MB
```

[116]:
```
# Checking for null values

orders.isnull().sum()
```

[116]:
```
orderId              0
clientId          1243
lastModified         0
status               0
truckKey        310775
location             0
fee                  0
volume(kg)           0
deliveryItem         0
dateGenerated        0
dtype: int64
```

[142]:
```
# checking Unique Order Statuses

orders['status'].value_counts()
```

[142]:
```
status
PENDING       310775
COMPLETED     186931
IN_TRANSIT     29239
Name: count, dtype: int64
```

[144]:
```
# Converting dateGenerated and lastModified to datetime

orders['dateGenerated'] = pd.to_datetime(orders['dateGenerated'],␣
  ↪errors='coerce')
orders['lastModified'] = pd.to_datetime(orders['lastModified'], errors='coerce')
```

[146]:
```
orders.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 526945 entries, 0 to 526944
Data columns (total 10 columns):
 #   Column         Non-Null Count   Dtype
---  ------         --------------   -----
 0   orderId        526945 non-null  object
 1   clientId       525702 non-null  object
 2   lastModified   526945 non-null  datetime64[ns]
 3   status         526945 non-null  object
 4   truckKey       216170 non-null  object
 5   location       526945 non-null  object
 6   fee            526945 non-null  float64
 7   volume(kg)     526945 non-null  float64
 8   deliveryItem   526945 non-null  object
 9   dateGenerated  526945 non-null  datetime64[ns]
dtypes: datetime64[ns](2), float64(2), object(6)
memory usage: 40.2+ MB
```

[128]:
```
# summary of volumes

orders['volume(kg)'].describe()
```

[128]:
```
count    526945.000000
mean        263.470034
std        1078.814205
min           0.000000
25%          34.000000
50%          85.000000
75%         170.000000
max       17000.000000
Name: volume(kg), dtype: float64
```

[130]:
```
# summary of fees

orders['fee'].describe()
```

[130]:
```
count    526945.000000
mean       4974.850554
std       20334.832398
min          32.000000
25%         640.000000
50%        1600.000000
75%        3200.000000
max      320000.000000
Name: fee, dtype: float64
```

[132]:
```
# are there bad data points in volumes and fees

orders[orders['volume(kg)'] <= 0]
```

[132]:
```
                           orderId     clientId        lastModified  \
337     01GGD5MT8R9001HFZQ200DH0G3    823671253 2025-03-02 18:20:59
1426    01GGQG5J40MS7RAZSGT66B3PJC    786891840 2025-03-02 18:20:59
2152    01GGWDRGZC0RTSN1CHFM23ST27    746975907 2025-03-02 18:20:59
2526    01GGYVKQQG2ST1FKGTFJQQEDMX   2420002333 2025-03-02 18:20:59
2861    01GGZ6R0G1GJAP169477ZNRJM8    270795860 2025-03-02 18:20:59
…                                …          …                  …
338666  01J15D6CTFQJDJFAB760H53ZNA  479080322.0 2025-03-02 18:20:59
339103  01J17HZPRQT07JVHWW3B0BSR4D  967402367.0 2025-03-02 18:20:59
```

```
339359  01J17T5K6SPJRA89VXXXAPZSQ9  943581484.0 2025-03-02 18:20:59
339810  01J182B4J7WE5JE8D7GYETH4ZT  949910635.0 2025-03-02 18:20:59
470753       AG-SFZW-794146-833911    5723969.0 2024-07-16 14:42:20

           status                          truckKey    location    fee  \
337       PENDING                               NaN  BOLGATANGA  1280.0
1426      PENDING                               NaN        TEMA  1600.0
2152      PENDING                               NaN  BOLGATANGA  1600.0
2526      PENDING                               NaN        TEMA   640.0
2861      PENDING                               NaN   KOFORIDUA   960.0
…           …                                …         …     …
338666    PENDING                               NaN       ACCRA  1600.0
339103    PENDING                               NaN       ACCRA   640.0
339359    PENDING                               NaN        TEMA   960.0
339810    PENDING                               NaN        TEMA   960.0
470753  COMPLETED  8a858ebb8e15ce2c018e1d50f7944865       ACCRA   320.0

        volume(kg) deliveryItem       dateGenerated
337            0.0          BED 2025-03-02 18:20:59
1426           0.0          BED 2025-03-02 18:20:59
2152           0.0        TABLE 2025-03-02 18:20:59
2526           0.0          BED 2025-03-02 18:20:59
2861           0.0        TABLE 2025-03-02 18:20:59
…            …          …                 …
338666         0.0        TABLE 2025-03-02 18:20:59
339103         0.0        TABLE 2025-03-02 18:20:59
339359         0.0        TABLE 2025-03-02 18:20:59
339810         0.0          BED 2025-03-02 18:20:59
470753         0.0        TABLE 2024-07-17 05:31:01

[1322 rows x 10 columns]
```

[134]:

```
orders[orders['fee'] < 0]
```

[134]:

```
Empty DataFrame
Columns: [orderId, clientId, lastModified, status, truckKey, location, fee,
volume(kg), deliveryItem, dateGenerated]
Index: []
```

[136]:

```
# Checking for orders with no trucks but marked as delivered/in transit

orders[
    ((orders['status'].isin(['IN_TRANSIT', 'COMPLETED'])) &
     (orders['truckKey'].isnull()))
]
```

[136]:

```
Empty DataFrame
Columns: [orderId, clientId, lastModified, status, truckKey, location, fee,
volume(kg), deliveryItem, dateGenerated]
Index: []
```

[138]:

```python
# checking how the trucks are utilized

assigned_orders = orders[orders['truckKey'].notnull()]
unassigned_orders = orders[orders['truckKey'].isnull()]

print(f"Assigned Orders: {len(assigned_orders)}")
print(f"Unassigned Orders: {len(unassigned_orders)}")
```

```
Assigned Orders: 216170
Unassigned Orders: 310775
```

[ ]:

[ ]:

[ ]:

[ ]: