In [1]:
```
import numpy as np
import pandas as pd
```

In [10]:
```
import warnings
warnings.filterwarnings('ignore')
```

In [104…
```
order=pd.read_csv('sample-orders.csv',encoding = 'ISO-8859-1')
```

In [105…
```
order.head()
```

Out[105]:

| | order_date | order_id | customer | grand_total |
|---|---|---|---|---|
| **0** | 9/7/11 | CA-2011-100006 | Dennis Kane | 378 |
| **1** | 7/8/11 | CA-2011-100090 | Ed Braxton | 699 |
| **2** | 3/14/11 | CA-2011-100293 | Neil Franz⸗sisch | 91 |
| **3** | 1/29/11 | CA-2011-100328 | Jasper Cacioppo | 4 |
| **4** | 4/8/11 | CA-2011-100363 | Jim Mitchum | 21 |

In [106…
```
order.info()   # no null values
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5009 entries, 0 to 5008
Data columns (total 4 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   order_date    5009 non-null   object
 1   order_id      5009 non-null   object
 2   customer      5009 non-null   object
 3   grand_total   5009 non-null   int64
dtypes: int64(1), object(3)
memory usage: 97.9+ KB
```

In [107…
```
order['customer'].nunique()  # 793 customer
```

Out[107]:
```
793
```

In [108…
```
order['grand_total'].sum()   # total sales
```

Out[108]:
```
2297261
```

In [109…
```
from datetime import date
import datetime as dt
```

In [110…
```
NOW = dt.datetime(2014,12,31)
```

In [111…
```
order['order_date']=pd.to_datetime(order['order_date'])
```

In [112…
```
# RFM
rfm_table=order.groupby('customer').agg({'order_date' : lambda x: (NOW-x.max()).days,
                                         'order_id' : lambda x : len(x),
                                         'grand_total' : lambda x : x.sum()})
rfm_table['order_date']=rfm_table['order_date'].astype(int)
rfm_table.rename(columns={'order_date' : 'recency',
                          'order_id' : 'freq',
                          'grand_total' : 'money'},inplace=True)
```

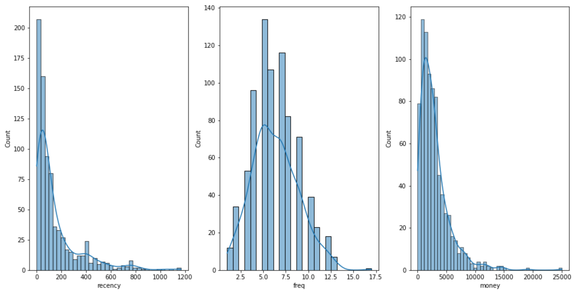
In [113…
```
rfm_table.head() # Aaron Bergman R-415 F-3 M-887
```

Out[113]:

| customer | recency | freq | money |
|---|---|---|---|
| **Aaron Bergman** | 415 | 3 | 887 |
| **Aaron Hawkins** | 12 | 7 | 1744 |
| **Aaron Smayling** | 88 | 7 | 3050 |
| **Adam Bellavance** | 54 | 8 | 7756 |
| **Adam Hart** | 34 | 10 | 3249 |

In [114…
```
import seaborn as sns
import matplotlib.pyplot as plt
fig, ax = plt.subplots(1, 3, figsize=(16, 8))
sns.histplot(rfm_table['recency'], kde=True, ax=ax[0])
sns.histplot(rfm_table['freq'], kde=True, ax=ax[1])
sns.histplot(rfm_table['money'], kde=True, ax=ax[2])
```

Out[114]:
```
<AxesSubplot:xlabel='money', ylabel='Count'>
```




In [115…
```
aaron=order[order['customer']=='Aaron Bergman']
aaron
```

Out[115]:

| | order_date | order_id | customer | grand_total |
|---|---|---|---|---|
| **624** | 2011-02-19 | CA-2011-152905 | Aaron Bergman | 13 |
| **665** | 2011-03-07 | CA-2011-156587 | Aaron Bergman | 310 |
| **2336** | 2013-11-11 | CA-2013-140935 | Aaron Bergman | 564 |

In [116…
```
(NOW-dt.datetime(2013,11,11)).days==415
```

Out[116]:
```
True
```

In [117…
```
# rfm quantile for rfm score
quantile=rfm_table.quantile(q=[0.25,0.5,0.75])
quantile
```

Out[117]:

| | recency | freq | money |
|---|---|---|---|
| **0.25** | 30.0 | 5.0 | 1145.0 |
| **0.50** | 75.0 | 6.0 | 2257.0 |
| **0.75** | 183.0 | 8.0 | 3784.0 |

In [118…
```
quantile=quantile.to_dict()
quantile
```

Out[118]:
```
{'recency': {0.25: 30.0, 0.5: 75.0, 0.75: 183.0},
 'freq': {0.25: 5.0, 0.5: 6.0, 0.75: 8.0},
 'money': {0.25: 1145.0, 0.5: 2257.0, 0.75: 3784.0}}
```

In [119…
```
# RFM Segmentation
def rclass(x,p,d):
    if x<=d[p][0.25]:
        return 1
    elif x<=d[p][0.5]:
        return 2
    elif x<=d[p][0.75]:
        return 3
    else:
        return 4
```

In [120…
```
def fclass(x,p,d):
    if x<=d[p][0.25]:
        return 4
    elif x<=d[p][0.5]:
        return 3
    elif x<=d[p][0.75]:
        return 2
    else:
        return 1
```

In [121…
```
rfm_table['r_quartile']=rfm_table['recency'].apply(rclass,args=('recency',quantile,))
rfm_table['f_quartile']=rfm_table['freq'].apply(fclass,args=('freq',quantile,))
rfm_table['m_quartile']=rfm_table['money'].apply(fclass,args=('money',quantile,))
```

In [122…
```
rfm_table.head(3)
```

Out[122]:

| customer | recency | freq | money | r_quartile | f_quartile | m_quartile |
|---|---|---|---|---|---|---|
| **Aaron Bergman** | 415 | 3 | 887 | 4 | 4 | 4 |
| **Aaron Hawkins** | 12 | 7 | 1744 | 1 | 2 | 3 |
| **Aaron Smayling** | 88 | 7 | 3050 | 3 | 2 | 2 |

In [123…
```
rfm_table['rfm_score']=rfm_table['r_quartile']+rfm_table['f_quartile']+rfm_table['m_quartile']
```

In [124…
```
rfm_table.head(1)
```

Out[124]:

| customer | recency | freq | money | r_quartile | f_quartile | m_quartile | rfm_score |
|---|---|---|---|---|---|---|---|
| **Aaron Bergman** | 415 | 3 | 887 | 4 | 4 | 4 | 12 |

In [125…
```
rfm_table['rfm_class']=rfm_table.r_quartile.map(str) \
                      + rfm_table.f_quartile.map(str)\
                      + rfm_table.m_quartile.map(str)
```

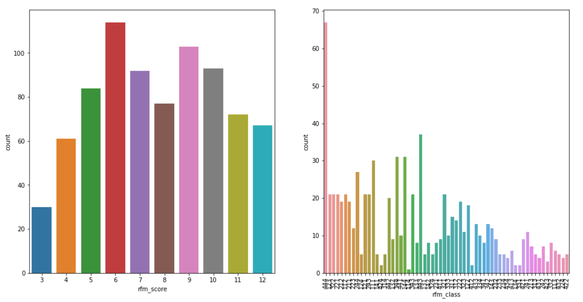
In [126…
```
rfm_table.head(5)
```

Out[126]:

| customer | recency | freq | money | r_quartile | f_quartile | m_quartile | rfm_score | rfm_class |
|---|---|---|---|---|---|---|---|---|
| **Aaron Bergman** | 415 | 3 | 887 | 4 | 4 | 4 | 12 | 444 |
| **Aaron Hawkins** | 12 | 7 | 1744 | 1 | 2 | 3 | 6 | 123 |
| **Aaron Smayling** | 88 | 7 | 3050 | 3 | 2 | 2 | 7 | 322 |
| **Adam Bellavance** | 54 | 8 | 7756 | 2 | 2 | 1 | 5 | 221 |
| **Adam Hart** | 34 | 10 | 3249 | 2 | 1 | 2 | 5 | 212 |

In [127…
```
fig, ax = plt.subplots(1, 2, figsize=(16, 8))
plt.xticks(rotation=90)
sns.countplot(x='rfm_score',data=rfm_table,ax=ax[0])
sns.countplot(x='rfm_class',data=rfm_table,ax=ax[1])
```

Out[127]:
```
<AxesSubplot:xlabel='rfm_class', ylabel='count'>
```




In [ ]: