Importing libraries

In [1]:

```
!pip install apyori
```

```
Requirement already satisfied: apyori in /usr/local/lib/python3.7/dist-packages (1.1.
2)
```

In [2]:

```
import numpy as np # linear algebra
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
import matplotlib.pyplot as plt
from apyori import apriori
%matplotlib inline
import os
```

Loading Dataset

In [3]:

```
lastfm1 = pd.read_csv("lastfm.csv")
lastfm = lastfm1
```

In [4]:

```
lastfm.head(5)
```

Out[4]:

	user	artist	sex	country
0	1	red hot chili peppers	f	Germany
1	1	the black dahlia murder	f	Germany
2	1	goldfrapp	f	Germany
3	1	dropkick murphys	f	Germany
4	1	le tigre	f	Germany

Preprocessing and EDA

In [5]:

```
lastfm.shape
```

Out[5]:

```
(289955, 4)
```

In [6]:

```
lastfm.describe()
```

Out[6]:

	user
count	289955.000000
mean	9852.460447
std	5692.355041
min	1.000000
25%	4935.000000
50%	9838.000000

	user
75%	14769.000000
max	19718.000000

In [7]:

```
#Since we are looking at user artist listening patterns, we only take those attribute
lastfm = lastfm[['user','artist']]
```

In [8]:

```
lastfm.isna().sum()
```

Out[8]:

```
user      0
artist    0
dtype: int64
```

In [9]:

```
lastfm = lastfm.drop_duplicates()
lastfm.shape
```

Out[9]:

```
(289953, 2)
```

Transforming Data in the form of User-Artists transaction list

In [10]:

```
records = []
for i in lastfm['user'].unique():
    records.append(list(lastfm[lastfm['user'] == i]['artist'].values))
print(type(records))
```

```
<class 'list'>
```

Generate rules using Apriori Algorithm

In [11]:

```
association_rules = apriori(records, min_support=0.01, min_confidence=0.4, min_lift=3
association_results = list(association_rules)
```

In [12]:

```
print(f"There are {len(association_results)} relations derived.")
```

```
There are 91 relations derived.
```

Displaying a few Rules with their Support, Confidence and Lift

In [13]:

```
i=1
for item in association_results:
    # first index of the inner list
    # Contains base item and add item
    pair = item[0]
    items = [x for x in pair]
    print("Rule: " + items[0] + " ==> " + items[1])

    # second index of the inner list
    print("Support: " + str(item[1]))
    # third index of the list located at 0th
    # of the third index of the inner list

    print("Confidence: " + str(item[2][0][2]))
    print("Lift: " + str(item[2][0][3]))
    print("==========================================================================
    i+=1
    if i==10:
      break
```

```
Rule: tool ==> a perfect circle
Support: 0.016266666666666665
Confidence: 0.44283121597096187
Lift: 8.717149920688225
===============================================================================
Rule: kaiser chiefs ==> arctic monkeys
Support: 0.012533333333333334
Confidence: 0.4008528784648188
Lift: 5.3116547499755145
===============================================================================
Rule: beyoncé ==> rihanna
Support: 0.013933333333333334
Confidence: 0.46860986547085204
Lift: 10.88103402796096
===============================================================================
Rule: metallica ==> black sabbath
Support: 0.0172
Confidence: 0.45263157894736844
Lift: 4.06555310431768
===============================================================================
Rule: sum 41 ==> blink-182
Support: 0.014133333333333333
Confidence: 0.42741935483870963
Lift: 7.420474910394264
===============================================================================
Rule: breaking benjamin ==> linkin park
Support: 0.0108
Confidence: 0.4426229508196721
Lift: 4.507362024640246
===============================================================================
Rule: death cab for cutie ==> bright eyes
Support: 0.0152
Confidence: 0.4021164021164021
Lift: 4.944054124381993
===============================================================================
Rule: death cab for cutie ==> broken social scene
Support: 0.011466666666666667
Confidence: 0.41646489104116224
Lift: 5.120469971817569
===============================================================================
Rule: broken social scene ==> radiohead
Support: 0.015066666666666667
Confidence: 0.5472154963680388
Lift: 3.0355889221599788
===============================================================================
```