Assignment 6

@author : @ruhend (Mudigonda Himansh)

AP19110010169

Apriori Algorithm

In [1]:
```
import numpy as np
import pandas as pd
from mlxtend.frequent_patterns import apriori, association_rules
```

In [2]:
```
low_memory = False
```

In [3]:
```
data = pd.read_excel('./Online Retail.xlsx')
data
```

Out[3]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	2011-12-09 12:50:00	4.95	12680.0	France

541909 rows × 8 columns

In [4]:
```
data.columns
```

Out[4]:
```
Index(['InvoiceNo', 'StockCode', 'Description', 'Quantity', 'InvoiceDate',
       'UnitPrice', 'CustomerID', 'Country'],
      dtype='object')
```

In [5]:
```
data.Country.unique()
```

Out[5]:
```
array(['United Kingdom', 'France', 'Australia', 'Netherlands', 'Germany',
       'Norway', 'EIRE', 'Switzerland', 'Spain', 'Poland', 'Portugal',
       'Italy', 'Belgium', 'Lithuania', 'Japan', 'Iceland',
       'Channel Islands', 'Denmark', 'Cyprus', 'Sweden', 'Austria',
       'Israel', 'Finland', 'Bahrain', 'Greece', 'Hong Kong', 'Singapore',
       'Lebanon', 'United Arab Emirates', 'Saudi Arabia',
       'Czech Republic', 'Canada', 'Unspecified', 'Brazil', 'USA',
       'European Community', 'Malta', 'RSA'], dtype=object)
```

Clean The Dataset

In [6]:
```
data['Description'] = data['Description'].str.strip()
```

In [7]:
```
data.dropna(axis = 0, subset = ['InvoiceNo'], inplace = True)
data['InvoiceNo'] = data['InvoiceNo'].astype('str')
```

In [8]:
```
data
```

Out[8]:

	InvoiceNo	StockCode	Description	Quantity	InvoiceDate	UnitPrice	CustomerID	Country
0	536365	85123A	WHITE HANGING HEART T-LIGHT HOLDER	6	2010-12-01 08:26:00	2.55	17850.0	United Kingdom
1	536365	71053	WHITE METAL LANTERN	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
2	536365	84406B	CREAM CUPID HEARTS COAT HANGER	8	2010-12-01 08:26:00	2.75	17850.0	United Kingdom
3	536365	84029G	KNITTED UNION FLAG HOT WATER BOTTLE	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
4	536365	84029E	RED WOOLLY HOTTIE WHITE HEART.	6	2010-12-01 08:26:00	3.39	17850.0	United Kingdom
...	...	...	...	...	...	...	...	...
541904	581587	22613	PACK OF 20 SPACEBOY NAPKINS	12	2011-12-09 12:50:00	0.85	12680.0	France
541905	581587	22899	CHILDREN'S APRON DOLLY GIRL	6	2011-12-09 12:50:00	2.10	12680.0	France
541906	581587	23254	CHILDRENS CUTLERY DOLLY GIRL	4	2011-12-09 12:50:00	4.15	12680.0	France
541907	581587	23255	CHILDRENS CUTLERY CIRCUS PARADE	4	2011-12-09 12:50:00	4.15	12680.0	France
541908	581587	22138	BAKING SET 9 PIECE RETROSPOT	3	2011-12-09 12:50:00	4.95	12680.0	France

541909 rows × 8 columns

In [9]:
```
data = data[~data['InvoiceNo'].str.contains('C')]
```

Split the data according to the region of transaction

In [10]:
```
basket_france = (data[data['Country'] == "France"].groupby(['InvoiceNo',"Description"])['Q
```

In [11]:
```
basket_israel = (data[data['Country'] == "Israel"].groupby(['InvoiceNo', 'Description'])['
```

In [12]:
```
basket_portugal = (data[data['Country'] == "Portugal"].groupby(['InvoiceNo',"Description"]
```

In [13]:
```
basket_sweden = (data[data['Country'] == "Sweden"].groupby(['InvoiceNo',"Description"])['Q
```

Hot Encoding the data set to provide to the apriori algorithm to have better results

In [14]:
```
def hot_encode(target):
    if int(target) <= 0:
        return 0
    if int(target) > 0:
        return 1
```

In [15]:
```
basket_france = basket_france.applymap(hot_encode)
```

In [16]:
```
basket_israel = basket_israel.applymap(hot_encode)
```

In [17]:
```
basket_portugal = basket_portugal.applymap(hot_encode)
```

In [18]:
```
basket_sweden = basket_sweden.applymap(hot_encode)
```

Building the models and analysing the results

In [19]:
```
fre_items = apriori(basket_france, min_support = 0.05, use_colnames = True)
rules = association_rules(fre_items, metric = "lift", min_threshold = 1)
rules = rules.sort_values(['confidence','lift'], ascending = [False,False])
print(rules)
```

```
                                          antecedents  \
199                       (JUMBO BAG WOODLAND ANIMALS)
468             (InvoiceNo, JUMBO BAG WOODLAND ANIMALS)
471                       (JUMBO BAG WOODLAND ANIMALS)
952   (RED TOADSTOOL LED NIGHT LIGHT, PLASTERS IN TI...
962   (PLASTERS IN TIN WOODLAND ANIMALS, RED TOADSTO...
...                                                 ...
709                                       (InvoiceNo)
842                                       (InvoiceNo)
1090                                      (InvoiceNo)
1119                                      (InvoiceNo)
1203                                      (InvoiceNo)

                                          consequents  antecedent support  \
199                                         (POSTAGE)            0.076531
468                                         (POSTAGE)            0.076531
471                                (POSTAGE, InvoiceNo)            0.076531
952                                         (POSTAGE)            0.051020
962                                         (POSTAGE)            0.053571
...                                               ...                 ...
709                   (RED  HARMONICA IN BOX, POSTAGE)            1.000000
842   (ROUND SNACK BOXES SET OF4 WOODLAND, RED RETRO...            1.000000
1090  (POSTAGE, LUNCH BAG SPACEBOY DESIGN, LUNCH BAG...            1.000000
1119  (LUNCH BAG RED RETROSPOT, POSTAGE, LUNCH BAG W...            1.000000
1203  (POSTAGE, RED TOADSTOOL LED NIGHT LIGHT, PLAST...            1.000000

      consequent support   support  confidence      lift  leverage  conviction
199             0.765306  0.076531     1.00000  1.306667  0.017961         inf
468             0.765306  0.076531     1.00000  1.306667  0.017961         inf
471             0.765306  0.076531     1.00000  1.306667  0.017961         inf
952             0.765306  0.051020     1.00000  1.306667  0.011974         inf
962             0.765306  0.053571     1.00000  1.306667  0.012573         inf
...                  ...       ...         ...       ...       ...         ...
709             0.051020  0.051020     0.05102  1.000000  0.000000         1.0
842             0.051020  0.051020     0.05102  1.000000  0.000000         1.0
1090            0.051020  0.051020     0.05102  1.000000  0.000000         1.0
1119            0.051020  0.051020     0.05102  1.000000  0.000000         1.0
1203            0.051020  0.051020     0.05102  1.000000  0.000000         1.0

[1434 rows x 9 columns]
```

In [*]:
```
fre_items = apriori(basket_israel, min_support = 0.05, use_colnames = True)
rules = association_rules(fre_items, metric = "lift", min_threshold = 1)
rules = rules.sort_values(['confidence','lift'], ascending = [False,False])
print(rules)
```

In [*]:
```
fre_items = apriori(basket_portugal , min_support = 0.05, use_colnames = True)
rules = association_rules(fre_items, metric = "lift", min_threshold = 1)
rules = rules.sort_values(['confidence','lift'], ascending = [False,False])
print(rules)
```

In [*]:
```
fre_items = apriori(basket_sweden , min_support = 0.05, use_colnames = True)
rules = association_rules(fre_items, metric = "lift", min_threshold = 1)
rules = rules.sort_values(['confidence','lift'], ascending = [False,False])
print(rules)
```

In [*]:
```
### As you can see, there seems to be some hardware limitation of my system...
### But this is how the output would look like for the rest of the countries
```