Market Basket Analysis Introduction

Reference: https://www.kaggle.com/code, http://pbpython.com/market-basket-analysis.html


```
# Required library
!pip install mlxtend
!pip install xlrd
!pip install --upgrade scikit-learn==1.0.2
!pip install --upgrade numpy==1.21.5
```

```
import pandas as pd
from mlxtend.frequent_patterns import apriori
from mlxtend.frequent_patterns import association_rules
```

```
df = pd.read_excel('https://github.com/davidjohnnn/all_datasets/raw/master/bay/
```

```
df.head()
```

```
# Clean up spaces in description and remove any rows that don't have a valid ir
df['Description'] = df['Description'].str.strip()
df.dropna(axis=0, subset=['InvoiceNo'], inplace=True)
```

```
# Remove the credit transactions (those with invoice numbers containing C)
df['InvoiceNo'] = df['InvoiceNo'].astype('str')
df = df[~df['InvoiceNo'].str.contains('C')]
```

```
# Only looking at sales for France. However, in additional code below, I will c
basket = (df[df['Country'] =="France"]
          .groupby(['InvoiceNo', 'Description'])['Quantity']
          .sum().unstack().reset_index().fillna(0)
          .set_index('InvoiceNo'))
```

```
basket.head()
```

```
# Show a subset of columns
basket.iloc[:,[0,1,2,3,4,5,6, 7]].head()
```

```
# There are a lot of zeros in the data but we also need to make sure any positi

# Convert the units to 1 hot encoded values
def encode_units(x):
    if x <= 0:
        return 0
    if x >= 1:
        return 1

# Convert to one hot vector
basket_sets = basket.applymap(encode_units) # lambda ?
```

```
# No need to track postage
# Remove column "Postage" (1 column)
basket_sets.drop('POSTAGE', inplace=True, axis=1)
```

```
basket_sets.head()
```

```
# Build up the frequent items
frequent_itemsets = apriori(basket_sets, min_support=0.07, use_colnames=True)
```

```
frequent_itemsets.head()
```

```
# Create the rules
rules = association_rules(frequent_itemsets, metric="lift", min_threshold=1)
rules.head()
```

```
rules[ (rules['lift'] >= 6) &
       (rules['confidence'] >= 0.8) ]
```

```
basket['ALARM CLOCK BAKELIKE GREEN'].sum()
```

```
basket['ALARM CLOCK BAKELIKE RED'].sum()
```

```
# What is also interesting is to see how the combinations vary by country of pur
# Let’s check out what some popular combinations might be in Germany
```

```
basket2 = (df[df['Country'] =="Germany"]
          .groupby(['InvoiceNo', 'Description'])['Quantity']
          .sum().unstack().reset_index().fillna(0)
          .set_index('InvoiceNo'))
```

```
# Convert to one hot vector
basket_sets2 = basket2.applymap(encode_units)
```

```
basket_sets2.drop('POSTAGE', inplace=True, axis=1)
```

```
frequent_itemsets2 = apriori(basket_sets2, min_support=0.05, use_colnames=True)
```

+ Code + Text

```
rules2 = association_rules(frequent_itemsets2, metric="lift", min_threshold=1)
rules2.head()
```

```
rules2[ (rules2['lift'] >= 4) &
        (rules2['confidence'] >= 0.5) ]
```

+ Code + Text