BigBasketAnalysis

June 18, 2024

1 Bigbasket Support Confidence and Lift

[37]:
```
import pandas as pd
from mlxtend.frequent_patterns import apriori, association_rules
```

[8]:
```
df = pd.read_excel("IMB575-XLS-ENG.xlsx")
df.head()
```

[8]:

	Member	Order	SKU	Created On	Description
0	M09736	6468572	34993740	22-09-2014 22:45	Other Sauces
1	M09736	6468572	15669800	22-09-2014 22:45	Cashews
2	M09736	6468572	34989501	22-09-2014 22:45	Other Dals
3	M09736	6468572	7572303	22-09-2014 22:45	Namkeen
4	M09736	6468572	15669856	22-09-2014 22:45	Sugar

[12]:
```
df_grouped = df.groupby('Order').agg({'Description': lambda x: list(set(x))}).
↪reset_index()
df_grouped = df_grouped.rename(columns={'Description': 'Items'})
df_grouped.head()
```

[12]:

	Order	Items
0	6422558	[Health Drinks, Sugar, Mosquito Repellent, Roo…
1	6422636	[Beans, Toor Dal, Sunflower Oils, Brinjals, Go…
2	6423338	[Sooji & Rava, Beans, Toor Dal, Almonds, Cashe…
3	6423534	[Cookies, Facial Tissues, Cakes, Organic F&V]
4	6423959	[Other Dals, Namkeen, Other Sweets, Sunflower …

[44]:
```
# Remove the transaction ID column
transactions = df_grouped['Items']

# # Convert transactions to a list of lists
transaction_list = transactions.tolist()
itemset = pd.DataFrame(transaction_list)
itemset = itemset.stack().str.get_dummies().sum(level=0)
```

```
/var/folders/2c/db_0lrw50h16d5cqvl89fg3c0000gn/T/ipykernel_46941/638049410.py:7:
FutureWarning: Using the level keyword in DataFrame and Series aggregations is
deprecated and will be removed in a future version. Use groupby instead.
```

```
df.sum(level=1) should use df.groupby(level=1).sum().
  itemset = itemset.stack().str.get_dummies().sum(level=0)
/Library/Frameworks/Python.framework/Versions/3.11/lib/python3.11/site-
packages/mlxtend/frequent_patterns/fpcommon.py:109: DeprecationWarning:
DataFrames with non-bool types result in worse computationalperformance and
their support might be discontinued in the future.Please use a DataFrame with
bool type
  warnings.warn(
```

[47]:
```
frequent_itemsets = apriori(itemset, min_support=0.01, use_colnames=True,␣
↪max_len=2)
print(frequent_itemsets.head())
```

```
    support                      itemsets
0  0.062001                     (Almonds)
1  0.016096  (Aluminium Foil & Cling Wrap)
2  0.077024             (Avalakki / Poha)
3  0.260284                      (Banana)
4  0.028973                 (Basmati Rice)
```

```
/Library/Frameworks/Python.framework/Versions/3.11/lib/python3.11/site-
packages/mlxtend/frequent_patterns/fpcommon.py:109: DeprecationWarning:
DataFrames with non-bool types result in worse computationalperformance and
their support might be discontinued in the future.Please use a DataFrame with
bool type
  warnings.warn(
```

[48]:
```
rules = association_rules(frequent_itemsets, metric="lift", min_threshold=3)
rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])
print(rules.head())
```

```
       antecedents     consequents  antecedent support  consequent support  \
25  (Other Flours)  (Sooji & Rava)            0.017885            0.132228
19   (Snacky Nuts)       (Namkeen)            0.039704            0.146298
29    (Rice Flour)  (Sooji & Rava)            0.023131            0.132228
30    (Rice Flour)      (Toor Dal)            0.023131            0.152259
33      (Urad Dal)      (Toor Dal)            0.102659            0.152259

     support  confidence      lift  leverage  conviction  zhangs_metric
25  0.011089    0.620000  4.688855  0.008724    2.283609       0.801055
19  0.021343    0.537538  3.674268  0.015534    1.845992       0.757930
29  0.012042    0.520619  3.937266  0.008984    1.810190       0.763681
30  0.011327    0.489691  3.216160  0.007805    1.661229       0.705386
33  0.048766    0.475029  3.119866  0.033135    1.614833       0.757207
```