CED19I028

PS - VI Apriori Algorithm

In [1]:
```
import matplotlib.pyplot as plt
import pandas as pd
import numpy as np
import random
from mlxtend.frequent_patterns import apriori, association_rules
import time
```

In [ ]:

In [ ]:

Running Apriori on test data..

In [2]:
```
data = [['Apple', 'Beer', 'Rice', 'Chicken'],
           ['Apple', 'Beer', 'Rice'],
           ['Apple', 'Beer'],
           ['Apple', 'Bananas'],
           ['Milk', 'Beer', 'Rice', 'Chicken'],
           ['Milk', 'Beer', 'Rice'],
           ['Milk', 'Beer'],
           ['Apple', 'Bananas']]

df = pd.DataFrame(data)
df
```

Out[2]:

	0	1	2	3
0	Apple	Beer	Rice	Chicken
1	Apple	Beer	Rice	None
2	Apple	Beer	None	None
3	Apple	Bananas	None	None
4	Milk	Beer	Rice	Chicken
5	Milk	Beer	Rice	None
6	Milk	Beer	None	None
7	Apple	Bananas	None	None

In [3]:
```
# convert data into one-hot encoding..

from mlxtend.preprocessing import TransactionEncoder

TE = TransactionEncoder()
TE_arr = TE.fit(data).transform(data)
TE_arr
```

Out[3]:
```
array([[ True, False,  True,  True, False,  True],
       [ True, False,  True, False, False,  True],
       [ True, False,  True, False, False, False],
       [ True,  True, False, False, False, False],
       [False, False,  True,  True,  True,  True],
       [False, False,  True, False,  True,  True],
       [False, False,  True, False,  True, False],
       [ True,  True, False, False, False, False]])
```

In [4]:
```
TE_arr.astype("int")
```

Out[4]:
```
array([[1, 0, 1, 1, 0, 1],
       [1, 0, 1, 0, 0, 1],
       [1, 0, 1, 0, 0, 0],
       [1, 1, 0, 0, 0, 0],
       [0, 0, 1, 1, 1, 1],
       [0, 0, 1, 0, 1, 1],
       [0, 0, 1, 0, 1, 0],
       [1, 1, 0, 0, 0, 0]])
```

In [5]:
```
columns = ['Apple', 'Bananas', 'Beer', 'Chicken', 'Milk', 'Rice']

df = pd.DataFrame(TE_arr, columns=columns)
df
```

Out[5]:

	Apple	Bananas	Beer	Chicken	Milk	Rice
0	True	False	True	True	False	True
1	True	False	True	False	False	True
2	True	False	True	False	False	False
3	True	True	False	False	False	False
4	False	False	True	True	True	True
5	False	False	True	False	True	True
6	False	False	True	False	True	False
7	True	True	False	False	False	False

In [6]:

```
# Building the model
frquent_items = apriori(df, min_support = 0.2, use_colnames = True)

frquent_items
```

Out[6]:

	support	itemsets
0	0.625	(Apple)
1	0.250	(Bananas)
2	0.750	(Beer)
3	0.250	(Chicken)
4	0.375	(Milk)
5	0.500	(Rice)
6	0.250	(Bananas, Apple)
7	0.375	(Beer, Apple)
8	0.250	(Apple, Rice)
9	0.250	(Chicken, Beer)
10	0.375	(Milk, Beer)
11	0.500	(Beer, Rice)
12	0.250	(Chicken, Rice)
13	0.250	(Milk, Rice)
14	0.250	(Beer, Apple, Rice)
15	0.250	(Chicken, Beer, Rice)
16	0.250	(Milk, Beer, Rice)

In [7]:

```
# Collecting the inferred rules in a dataframe
# min_threshold = confidence of metric selected
rules = association_rules(frquent_items, metric ="confidence", min_threshold = 0.8)
rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])

rules.iloc[:,:6]
```

Out[7]:

	antecedents	consequents	antecedent support	consequent support	support	confidence
4	(Chicken)	(Rice)	0.250	0.500	0.250	1.0
6	(Chicken, Beer)	(Rice)	0.250	0.500	0.250	1.0
8	(Chicken)	(Beer, Rice)	0.250	0.500	0.250	1.0
0	(Bananas)	(Apple)	0.250	0.625	0.250	1.0
1	(Chicken)	(Beer)	0.250	0.750	0.250	1.0
2	(Milk)	(Beer)	0.375	0.750	0.375	1.0
3	(Rice)	(Beer)	0.500	0.750	0.500	1.0
5	(Apple, Rice)	(Beer)	0.250	0.750	0.250	1.0
7	(Chicken, Rice)	(Beer)	0.250	0.750	0.250	1.0
9	(Milk, Rice)	(Beer)	0.250	0.750	0.250	1.0

In [ ]:

In [ ]:

Running appriori on mushroom FIMI dataset

In [8]:
```
df = pd.read_csv("mushroom.dat", delimiter=" ")
df.drop(['Unnamed: 23'], axis=1, inplace=True)
df
```

Out[8]:

	1	3	9	13	23	25	34	36	38	40	...	63	67	76	85	86	90	93	98	107	113
0	2	3	9	14	23	26	34	36	39	40	...	63	67	76	85	86	90	93	99	108	114
1	2	4	9	15	23	27	34	36	39	41	...	63	67	76	85	86	90	93	99	108	115
2	1	3	10	15	23	25	34	36	38	41	...	63	67	76	85	86	90	93	98	107	113
3	2	3	9	16	24	28	34	37	39	40	...	63	67	76	85	86	90	94	99	109	114
4	2	3	10	14	23	26	34	36	39	41	...	63	67	76	85	86	90	93	98	108	114
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
8118	2	7	9	13	24	28	35	36	39	50	...	63	73	83	85	88	90	93	106	112	119
8119	2	3	9	13	24	28	35	36	39	50	...	63	73	83	85	87	90	93	106	110	119
8120	2	6	9	13	24	28	35	36	39	41	...	63	73	83	85	88	90	93	106	112	119
8121	1	7	10	13	24	31	34	36	38	48	...	66	67	76	85	86	90	94	102	110	119
8122	2	3	9	13	24	28	35	36	39	50	...	63	73	83	85	88	90	93	104	112	119

8123 rows × 23 columns

In [9]:
```
# convert data into one-hot encoding..

from mlxtend.preprocessing import TransactionEncoder

data = df.values.tolist()
TE = TransactionEncoder()
TE_arr = TE.fit(data).transform(data)
TE_arr = TE_arr.astype("int")
TE_arr
```

Out[9]:
```
array([[0, 1, 1, ..., 0, 0, 0],
       [0, 1, 0, ..., 0, 0, 0],
       [1, 0, 1, ..., 0, 0, 0],
       ...,
       [0, 1, 0, ..., 0, 0, 1],
       [1, 0, 0, ..., 0, 0, 1],
       [0, 1, 1, ..., 0, 0, 1]])
```

In [10]:
```
df = pd.DataFrame(TE_arr)
df
```

Out[10]:

	0	1	2	3	4	5	6	7	8	9	...	109	110	111	112	113	114	115	116	117	118
0	0	1	1	0	0	0	0	0	1	0	...	0	0	0	0	1	0	0	0	0	0
1	0	1	0	1	0	0	0	0	1	0	...	0	0	0	0	0	1	0	0	0	0
2	1	0	1	0	0	0	0	0	0	1	...	0	0	0	1	0	0	0	0	0	0
3	0	1	1	0	0	0	0	0	1	0	...	0	0	0	0	1	0	0	0	0	0
4	0	1	1	0	0	0	0	0	0	1	...	0	0	0	0	1	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
8118	0	1	0	0	0	0	1	0	1	0	...	0	0	1	0	0	0	0	0	0	1
8119	0	1	1	0	0	0	0	0	1	0	...	1	0	0	0	0	0	0	0	0	1
8120	0	1	0	0	0	1	0	0	1	0	...	0	0	1	0	0	0	0	0	0	1
8121	1	0	0	0	0	0	1	0	0	1	...	1	0	0	0	0	0	0	0	0	1
8122	0	1	1	0	0	0	0	0	1	0	...	0	0	1	0	0	0	0	0	0	1

8123 rows × 119 columns

In [11]:
```
# Building the model
frquent_items = apriori(df, min_support = 0.6, use_colnames = True, verbose=1)
frquent_items
```

```
Processing 15 combinations | Sampling itemset size 5
```

```
C:\Python310\lib\site-packages\mlxtend\frequent_patterns\fpcommon.py:111: 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(
```

Out[11]:

	support	itemsets
0	0.974147	(33)
1	0.838483	(35)
2	0.690878	(38)
3	0.637080	(58)
4	0.607534	(62)
5	1.000000	(84)
6	0.975379	(85)
7	0.921704	(89)
8	0.812631	(33, 35)
9	0.665025	(33, 38)
10	0.613443	(33, 58)
11	0.974147	(33, 84)
12	0.973163	(33, 85)
13	0.898067	(89, 33)
14	0.838483	(35, 84)
15	0.814847	(35, 85)
16	0.795642	(89, 35)
17	0.690878	(84, 38)
18	0.667241	(85, 38)
19	0.612582	(89, 38)
20	0.637080	(58, 84)
21	0.613443	(58, 85)
22	0.607534	(84, 62)
23	0.975379	(84, 85)
24	0.921704	(89, 84)
25	0.897082	(89, 85)
26	0.812631	(33, 35, 84)
27	0.812631	(33, 35, 85)

	support	itemsets
28	0.772005	(89, 35, 33)
29	0.665025	(33, 84, 38)
30	0.665025	(33, 85, 38)
31	0.613443	(33, 58, 84)
32	0.613443	(33, 58, 85)
33	0.973163	(33, 84, 85)
34	0.898067	(89, 84, 33)
35	0.897082	(89, 85, 33)
36	0.814847	(35, 84, 85)
37	0.795642	(89, 35, 84)
38	0.772005	(89, 35, 85)
39	0.667241	(84, 85, 38)
40	0.612582	(89, 84, 38)
41	0.613443	(58, 84, 85)
42	0.897082	(89, 84, 85)
43	0.812631	(33, 35, 84, 85)
44	0.772005	(89, 35, 84, 33)
45	0.772005	(89, 35, 85, 33)
46	0.665025	(33, 84, 85, 38)
47	0.613443	(33, 58, 84, 85)
48	0.897082	(89, 84, 85, 33)
49	0.772005	(89, 35, 84, 85)
50	0.772005	(33, 35, 84, 85, 89)

In [ ]:

In [12]:
```
# Collecting the inferred rules in a dataframe
# min_threshold = confidence of metric selected
rules = association_rules(frquent_items, metric ="confidence", min_threshold = 0.4)
rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])
rules.iloc[:,:6]
```

Out[12]:

	antecedents	consequents	antecedent support	consequent support	support	confidence
74	(58, 85)	(33)	0.613443	0.974147	0.613443	1.000000
90	(89, 85)	(33)	0.897082	0.974147	0.897082	1.000000
166	(89, 35, 85)	(33)	0.772005	0.974147	0.772005	1.000000
197	(58, 84, 85)	(33)	0.613443	0.974147	0.613443	1.000000
202	(58, 85)	(33, 84)	0.613443	0.974147	0.613443	1.000000
...	...	...	...	...	...	...
71	(84)	(33, 58)	1.000000	0.613443	0.613443	0.613443
130	(84)	(58, 85)	1.000000	0.613443	0.613443	0.613443
206	(84)	(33, 58, 85)	1.000000	0.613443	0.613443	0.613443
124	(84)	(89, 38)	1.000000	0.612582	0.612582	0.612582
28	(84)	(62)	1.000000	0.607534	0.607534	0.607534

266 rows × 6 columns

In [13]:
```
# min suppport = 30, 40, 60
# confidence = 70,80,90
```

In [14]:

```
import warnings
warnings.filterwarnings('ignore')

min_supports = (30, 40, 50)
confidences = (70, 80, 90)

print("========================================================================================\n\n")

for each in min_supports:
    for percentage in confidences:
        start_time = time.process_time()
        # Building the model
        support = each/100
        frquent_items = apriori(df, min_support = support, use_colnames = True, verbose=1)
        print(frquent_items)

        # Collecting the inferred rules in a dataframe
        # min_threshold = confidence of metric selected
        minThreshold = percentage/100
        rules = association_rules(frquent_items, metric ="confidence", min_threshold = minThreshold)
        rules = rules.sort_values(['confidence', 'lift'], ascending =[False, False])
        display(rules.iloc[:,:6])

        end_time = time.process_time()
        time_taken = end_time - start_time
        print(f"Time Taken for min_support = {each}% and min_threshold = {minThreshold} is {time_taken}sec \n")
        print("========================================================================================\n\n\n")
```

```
========================================================================================


Processing 20 combinations | Sampling itemset size 1087
       support                              itemsets
0     0.481965                                   (0)
1     0.518035                                   (1)
2     0.449957                                   (2)
3     0.388034                                   (5)
4     0.314539                                   (8)
...        ...                                   ...
2730  0.339653      (33, 35, 84, 85, 89, 58, 92, 62)
2731  0.303336   (1, 33, 35, 38, 84, 85, 22, 58, 62)
2732  0.303336   (1, 33, 35, 38, 84, 85, 22, 58, 92)
2733  0.317124  (33, 35, 38, 84, 85, 22, 89, 58, 92)
2734  0.316016  (33, 35, 84, 85, 22, 89, 58, 92, 62)

[2735 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
14119	(52, 23)	(89, 93, 85)	0.307276	0.317124	0.307276	1.0
30741	(33, 52, 23)	(89, 93, 85)	0.307276	0.317124	0.307276	1.0
30755	(52, 23)	(89, 33, 93, 85)	0.307276	0.317124	0.307276	1.0
30851	(52, 23, 84)	(89, 93, 85)	0.307276	0.317124	0.307276	1.0
30862	(52, 23)	(89, 93, 84, 85)	0.307276	0.317124	0.307276	1.0
...	...	...	...	...	...	...
23451	(89, 1, 84)	(33, 35, 85)	0.453035	0.812631	0.317124	0.7
23454	(89, 1)	(33, 35, 84, 85)	0.453035	0.812631	0.317124	0.7
2673	(89, 1)	(35, 85)	0.453035	0.814847	0.317124	0.7
10504	(89, 1, 84)	(35, 85)	0.453035	0.814847	0.317124	0.7
10507	(89, 1)	(35, 84, 85)	0.453035	0.814847	0.317124	0.7

58016 rows × 6 columns

```
Time Taken for min_support = 30% and min_threshold = 0.7 is 2.171875sec

================================================================================================
```

```
Processing 20 combinations | Sampling itemset size 1087
       support                              itemsets
0     0.481965                                   (0)
1     0.518035                                   (1)
2     0.449957                                   (2)
3     0.388034                                   (5)
4     0.314539                                   (8)
...        ...                                   ...
2730  0.339653      (33, 35, 84, 85, 89, 58, 92, 62)
2731  0.303336   (1, 33, 35, 38, 84, 85, 22, 58, 62)
2732  0.303336   (1, 33, 35, 38, 84, 85, 22, 58, 92)
2733  0.317124  (33, 35, 38, 84, 85, 22, 89, 58, 92)
2734  0.316016  (33, 35, 84, 85, 22, 89, 58, 92, 62)

[2735 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
10819	(52, 23)	(89, 93, 85)	0.307276	0.317124	0.307276	1.000000
23107	(33, 52, 23)	(89, 93, 85)	0.307276	0.317124	0.307276	1.000000
23121	(52, 23)	(89, 33, 93, 85)	0.307276	0.317124	0.307276	1.000000
23209	(52, 23, 84)	(89, 93, 85)	0.307276	0.317124	0.307276	1.000000
23220	(52, 23)	(89, 93, 84, 85)	0.307276	0.317124	0.307276	1.000000
...	...	...	...	...	...	...
27283	(33, 35, 84, 85, 22, 92)	(1)	0.379047	0.518035	0.303336	0.800260
27300	(33, 35, 85, 22, 92)	(1, 84)	0.379047	0.518035	0.303336	0.800260
7600	(58, 35, 92)	(1, 38)	0.408593	0.482580	0.326973	0.800241
18315	(58, 35, 84, 92)	(1, 38)	0.408593	0.482580	0.326973	0.800241
18326	(58, 35, 92)	(1, 84, 38)	0.408593	0.482580	0.326973	0.800241

43555 rows × 6 columns

```
Time Taken for min_support = 30% and min_threshold = 0.8 is 2.09375sec

================================================================================================
```

```
Processing 20 combinations | Sampling itemset size 1087
       support                                itemsets
0     0.481965                                     (0)
1     0.518035                                     (1)
2     0.449957                                     (2)
3     0.388034                                     (5)
4     0.314539                                     (8)
...        ...                                     ...
2730  0.339653      (33, 35, 84, 85, 89, 58, 92, 62)
2731  0.303336   (1, 33, 35, 38, 84, 85, 22, 58, 62)
2732  0.303336   (1, 33, 35, 38, 84, 85, 22, 58, 92)
2733  0.317124  (33, 35, 38, 84, 85, 22, 89, 58, 92)
2734  0.316016  (33, 35, 84, 85, 22, 89, 58, 92, 62)

[2735 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
6821	(52, 23)	(89, 93, 85)	0.307276	0.317124	0.307276	1.000000
13726	(33, 52, 23)	(89, 93, 85)	0.307276	0.317124	0.307276	1.000000
13739	(52, 23)	(89, 33, 93, 85)	0.307276	0.317124	0.307276	1.000000
13826	(52, 23, 84)	(89, 93, 85)	0.307276	0.317124	0.307276	1.000000
13837	(52, 23)	(89, 93, 84, 85)	0.307276	0.317124	0.307276	1.000000
...	...	...	...	...	...	...
8074	(89, 33, 92, 85)	(62)	0.414502	0.607534	0.373138	0.900208
8682	(89, 84, 85, 92)	(62)	0.414502	0.607534	0.373138	0.900208
8686	(89, 92, 85)	(84, 62)	0.414502	0.607534	0.373138	0.900208
15021	(33, 84, 85, 89, 92)	(62)	0.414502	0.607534	0.373138	0.900208
15028	(89, 33, 92, 85)	(84, 62)	0.414502	0.607534	0.373138	0.900208

24396 rows × 6 columns

```
Time Taken for min_support = 30% and min_threshold = 0.9 is 1.9375sec

================================================================================================
```

```
Processing 112 combinations | Sampling itemset size 75
      support                      itemsets
0    0.481965                           (0)
1    0.518035                           (1)
2    0.449957                           (2)
3    0.415487                          (22)
4    0.584513                          (23)
..        ...                           ...
560  0.451065     (33, 35, 109, 84, 85, 89)
561  0.407731      (33, 38, 84, 85, 55, 89)
562  0.442570      (33, 84, 85, 89, 58, 62)
563  0.407731      (35, 38, 84, 85, 55, 89)
564  0.407731  (33, 35, 38, 84, 85, 55, 89)

[565 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
1068	(38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
2277	(84, 38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
2281	(38, 55)	(33, 35, 84)	0.422504	0.812631	0.422504	1.000000
2296	(85, 38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
2300	(38, 55)	(33, 35, 85)	0.422504	0.812631	0.422504	1.000000
...	...	...	...	...	...	...
991	(84, 23)	(33, 35)	0.584513	0.812631	0.409947	0.701348
992	(23)	(33, 35, 84)	0.584513	0.812631	0.409947	0.701348
999	(23)	(33, 35, 85)	0.584513	0.812631	0.409947	0.701348
2207	(84, 23)	(33, 35, 85)	0.584513	0.812631	0.409947	0.701348
2209	(23)	(33, 35, 84, 85)	0.584513	0.812631	0.409947	0.701348

3828 rows × 6 columns

```
Time Taken for min_support = 40% and min_threshold = 0.7 is 0.3125sec

================================================================================================
```

```
Processing 112 combinations | Sampling itemset size 75
      support                      itemsets
0    0.481965                           (0)
1    0.518035                           (1)
2    0.449957                           (2)
3    0.415487                          (22)
4    0.584513                          (23)
..        ...                           ...
560  0.451065     (33, 35, 109, 84, 85, 89)
561  0.407731      (33, 38, 84, 85, 55, 89)
562  0.442570      (33, 84, 85, 89, 58, 62)
563  0.407731      (35, 38, 84, 85, 55, 89)
564  0.407731  (33, 35, 38, 84, 85, 55, 89)

[565 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
967	(38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
2013	(84, 38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
2016	(38, 55)	(33, 35, 84)	0.422504	0.812631	0.422504	1.000000
2028	(85, 38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
2031	(38, 55)	(33, 35, 85)	0.422504	0.812631	0.422504	1.000000
...	...	...	...	...	...	...
2059	(84, 85, 38)	(33, 35)	0.667241	0.812631	0.535024	0.801845
2062	(85, 38)	(33, 35, 84)	0.667241	0.812631	0.535024	0.801845
569	(58, 62)	(92)	0.511511	0.488366	0.409578	0.800722
1692	(58, 84, 62)	(92)	0.511511	0.488366	0.409578	0.800722
1696	(58, 62)	(84, 92)	0.511511	0.488366	0.409578	0.800722

3302 rows × 6 columns

```
Time Taken for min_support = 40% and min_threshold = 0.8 is 0.234375sec

================================================================================
```

```
Processing 112 combinations | Sampling itemset size 75
      support                      itemsets
0    0.481965                           (0)
1    0.518035                           (1)
2    0.449957                           (2)
3    0.415487                          (22)
4    0.584513                          (23)
..        ...                           ...
560  0.451065     (33, 35, 109, 84, 85, 89)
561  0.407731      (33, 38, 84, 85, 55, 89)
562  0.442570      (33, 84, 85, 89, 58, 62)
563  0.407731      (35, 38, 84, 85, 55, 89)
564  0.407731  (33, 35, 38, 84, 85, 55, 89)

[565 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
732	(38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
1488	(84, 38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
1491	(38, 55)	(33, 35, 84)	0.422504	0.812631	0.422504	1.000000
1503	(85, 38, 55)	(33, 35)	0.422504	0.812631	0.422504	1.000000
1506	(38, 55)	(33, 35, 85)	0.422504	0.812631	0.422504	1.000000
...	...	...	...	...	...	...
1596	(33, 85, 55)	(89, 35)	0.464853	0.795642	0.419549	0.902542
1977	(84, 85, 55)	(89, 35)	0.464853	0.795642	0.419549	0.902542
1982	(85, 55)	(89, 35, 84)	0.464853	0.795642	0.419549	0.902542
2199	(33, 84, 85, 55)	(89, 35)	0.464853	0.795642	0.419549	0.902542
2208	(33, 85, 55)	(89, 35, 84)	0.464853	0.795642	0.419549	0.902542

2404 rows × 6 columns

```
Time Taken for min_support = 40% and min_threshold = 0.9 is 0.21875sec

================================================================================================
```

```
Processing 12 combinations | Sampling itemset size 6
      support              itemsets
0    0.518035                   (1)
1    0.584513                  (23)
2    0.974147                  (33)
3    0.838483                  (35)
4    0.690878                  (38)
..        ...                   ...
148  0.772005  (33, 35, 84, 85, 89)
149  0.588945  (33, 38, 84, 85, 89)
150  0.567278  (33, 52, 85, 84, 89)
151  0.559276  (33, 84, 85, 89, 58)
152  0.529730  (33, 84, 85, 89, 62)

[153 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
216	(52)	(89, 85)	0.567278	0.897082	0.567278	1.000000
390	(33, 52)	(89, 85)	0.567278	0.897082	0.567278	1.000000
391	(52)	(89, 85, 33)	0.567278	0.897082	0.567278	1.000000
516	(52, 84)	(89, 85)	0.567278	0.897082	0.567278	1.000000
517	(52)	(89, 84, 85)	0.567278	0.897082	0.567278	1.000000
...	...	...	...	...	...	...
502	(84)	(89, 35, 85)	1.000000	0.772005	0.772005	0.772005
604	(84)	(89, 33, 35, 85)	1.000000	0.772005	0.772005	0.772005
168	(38)	(89, 35)	0.690878	0.795642	0.518035	0.749822
474	(84, 38)	(89, 35)	0.690878	0.795642	0.518035	0.749822
475	(38)	(89, 35, 84)	0.690878	0.795642	0.518035	0.749822

667 rows × 6 columns

```
Time Taken for min_support = 50% and min_threshold = 0.7 is 0.046875sec

================================================================================================
```

```
Processing 12 combinations | Sampling itemset size 64
      support              itemsets
0    0.518035                   (1)
1    0.584513                  (23)
2    0.974147                  (33)
3    0.838483                  (35)
4    0.690878                  (38)
..        ...                   ...
148  0.772005  (33, 35, 84, 85, 89)
149  0.588945  (33, 38, 84, 85, 89)
150  0.567278  (33, 52, 85, 84, 89)
151  0.559276  (33, 84, 85, 89, 58)
152  0.529730  (33, 84, 85, 89, 62)

[153 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
209	(52)	(89, 85)	0.567278	0.897082	0.567278	1.000000
374	(33, 52)	(89, 85)	0.567278	0.897082	0.567278	1.000000
375	(52)	(89, 85, 33)	0.567278	0.897082	0.567278	1.000000
491	(52, 84)	(89, 85)	0.567278	0.897082	0.567278	1.000000
492	(52)	(89, 84, 85)	0.567278	0.897082	0.567278	1.000000
...	...	...	...	...	...	...
210	(58, 84)	(62)	0.637080	0.607534	0.511511	0.802899
213	(58)	(84, 62)	0.637080	0.607534	0.511511	0.802899
283	(85, 38)	(33, 35)	0.667241	0.812631	0.535024	0.801845
531	(84, 85, 38)	(33, 35)	0.667241	0.812631	0.535024	0.801845
534	(85, 38)	(33, 35, 84)	0.667241	0.812631	0.535024	0.801845

633 rows × 6 columns

```
Time Taken for min_support = 50% and min_threshold = 0.8 is 0.0sec

================================================================================================
```

```
Processing 12 combinations | Sampling itemset size 6
      support              itemsets
0    0.518035                   (1)
1    0.584513                  (23)
2    0.974147                  (33)
3    0.838483                  (35)
4    0.690878                  (38)
..        ...                   ...
148  0.772005  (33, 35, 84, 85, 89)
149  0.588945  (33, 38, 84, 85, 89)
150  0.567278  (33, 52, 85, 84, 89)
151  0.559276  (33, 84, 85, 89, 58)
152  0.529730  (33, 84, 85, 89, 62)

[153 rows x 2 columns]
```

	antecedents	consequents	antecedent support	consequent support	support	confidence
159	(52)	(89, 85)	0.567278	0.897082	0.567278	1.000000
276	(33, 52)	(89, 85)	0.567278	0.897082	0.567278	1.000000
277	(52)	(89, 85, 33)	0.567278	0.897082	0.567278	1.000000
368	(52, 84)	(89, 85)	0.567278	0.897082	0.567278	1.000000
369	(52)	(89, 84, 85)	0.567278	0.897082	0.567278	1.000000
...	...	...	...	...	...	...
310	(33, 85, 62)	(89)	0.583898	0.921704	0.529730	0.907232
377	(84, 85, 62)	(89)	0.583898	0.921704	0.529730	0.907232
379	(85, 62)	(89, 84)	0.583898	0.921704	0.529730	0.907232
457	(33, 84, 85, 62)	(89)	0.583898	0.921704	0.529730	0.907232
462	(33, 85, 62)	(89, 84)	0.583898	0.921704	0.529730	0.907232

470 rows × 6 columns

```
Time Taken for min_support = 50% and min_threshold = 0.9 is 0.046875sec

================================================================================
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: