

Q5

Import Libraries

```
from pprint import pprint
import itertools
import math
```

```
# Transactions
transactions = [(1, frozenset({'A', 'C', 'D', 'E'})),
                (2, frozenset({'A', 'B', 'D', 'E'})),
                (3, frozenset({'C', 'E'})),
                (4, frozenset({'C', 'D'})),
                (5, frozenset({'A', 'B', 'D'})),
                (6, frozenset({'B', 'D', 'E'})),
                (7, frozenset({'A', 'C', 'D'})),
                (8, frozenset({'B', 'C', 'D', 'E'})),
                ]
```

```
# unique items
items = set()
for i, x in transactions:
    items.update(x)
# print('Total items in the data set - {}'.format(len(items)))
print('The unique items are - {}'.format(items))
```

The unique items are - {'E', 'A', 'D', 'C', 'B'}

```
# Apriori Algorithm
# L1
itemsets = {}
for x in itertools.combinations(items, 1):
    for _, y in transactions:
        if set(x).issubset(y):
            if frozenset(x) not in itemsets:
                itemsets[frozenset(x)] = 0
            itemsets[frozenset(x)] += 1
```

```

# Candidate Set
print('Candidate Set C{0}'.format(1))
print('\t Itemset \t Support Count')
for x in itemsets.keys():
    print('\t[{}]'.format(', '.join(x)), '\t {}'.format(itemsets[x]))

itemsets_support3 = dict((itemset,count)for itemset, count in itemsets.items()
if count >= 3)
if len(itemsets_support3) < len(itemsets):
    print('Pruning')
else:
    print('Pruning Not Required.')
itemsets_support3 = dict((itemset,count)for itemset, count in itemsets.items()
if count >= 3)
s = set(itemsets_support3.keys())
print('Step 1 L1')
print('1 - Itemsets with support = 3 are as following: ')
for x in s:
    print('[{}]'.format(', '.join(x)), ' Support Count:
{}'.format(itemsets_support3[x]))

#####

for i in range(2, len(items)+1):
    if len(itemsets_support3) == 0:
        print('No new frequent itemsets identified')
        break
    print('-----')
    print('-----')
    print('Step: {0} L({0})'.format(i))
    itemsets = {}
    # Take only frequent itemsets for next step, prune step of Apriori
    items = set(itemsets_support3.keys())
    for x1,x2 in itertools.combinations(items, 2):
        # Union Step
        next_frequent_itemset = set(x1.union(x2))
        if frozenset(next_frequent_itemset) not in itemsets:
            for _, y in transactions:
                # print('next_frequent_itemset:
{}'.format(next_frequent_itemset))
                if len(next_frequent_itemset) == i and
next_frequent_itemset.issubset(y):
                    # print('next_frequent_itemset:
{}'.format(next_frequent_itemset))
                    if frozenset(next_frequent_itemset) not in itemsets:

```

```

        itemsets[frozenset(next_frequent_itemset)]=0
        itemsets[frozenset(next_frequent_itemset)]+=1

# Candidate Set
print('Candidate Set C{0}'.format(i))
print('\t Itemset \t Support Count')
for x in itemsets.keys():
    print('\t[{0}]'.format(', '.join(x)), '\t {0}'.format(itemsets[x]))

    itemsets_support3 = dict((itemset,count)for itemset, count in
itemsets.items() if count >= 3)
    if len(itemsets_support3) < len(itemsets):
        print('Pruning')
    else:
        print('Pruning Not Required.')

    print('L{0} Itemsets with support = 3 are as following: '.format(i))
    print('\tItemset \tSupport Count')
    for x in itemsets_support3.keys():
        print('\t[{0}]'.format(', '.join(x)), '\t\t{0}'.format(itemsets_support3[x]))

```

Candidate Set C1

Itemset	Support Count
[E]	5
[A]	4
[D]	7
[C]	5
[B]	4

Pruning Not Required.

Step 1 L1

1 - Itemsets with support = 3 are as following:

[B]	Support Count: 4
[A]	Support Count: 4
[D]	Support Count: 7
[C]	Support Count: 5
[E]	Support Count: 5

Step: 2 L(2)

Candidate Set C2

Itemset	Support Count
[B, A]	2
[D, B]	4
[C, B]	1

[E, B]	3
[D, A]	4
[C, A]	2
[E, A]	2
[C, D]	4
[E, D]	4
[C, E]	3

Pruning

L2 Itemsets with support = 3 are as following:

Itemset	Support Count
[D, B]	4
[E, B]	3
[D, A]	4
[C, D]	4
[E, D]	4
[C, E]	3

Step: 3 L(3)

Candidate Set C3

Itemset	Support Count
[E, D, A]	2
[D, B, A]	2
[C, D, A]	2
[C, E, B]	1
[E, D, B]	3
[C, E, D]	2
[C, D, B]	1

Pruning

L3 Itemsets with support = 3 are as following:

Itemset	Support Count
[E, D, B]	3

Step: 4 L(4)

Candidate Set C4

Itemset	Support Count
---------	---------------

Pruning Not Required.

L4 Itemsets with support = 3 are as following:

Itemset	Support Count
---------	---------------

No new frequent itemsets identified