Data Mining (CS634)

Mid Term Project

Kelvinkumar Gandhi

UCID - ksg27

NJIT_ID - 31457955

Apriori Algorithm :

I have implemented apriori algorithm in python language. I’ve read the transaction data from text file.

For my implementation, I want briefly explain steps that I followed to implemented this algorithm.

1. Read transaction from text file.
2. Getting min support and min confidence from user input.
3. Finding frequent 1 itemset.
4. From frequent 1 item, finding frequent k itemsets.
5. From frequent k itemsets, I’ve generated association rules.
6. Display rules which satisfy min support and min confidence.

Program

```
from itertools import combinations

def generateSupportCount(trans):

    item_list=[]
    item_dict={}

    for item in trans:
        i = item.strip().split(',')

        for j in i:
            if j not in item_list:
                item_list.append(j)
                item_dict[j]=1

            else:
                item_dict[j]=item_dict[j] + 1

    #print(item_list)

    #print(item_dict)

    return item_list, item_dict



#Compute support frequent 1 itemset
def satisfySupport(tr_dict, min_sup,trans_dataset, tran_len):

    new_list = []

    for i,c in tr_dict.items():
        if c/tran_len >= min_sup:
            new_list.append([i])
            print(f'[{i}] => {c/tran_len} (support)')

    return new_list
```

```
def associateTrans(candidateItemSet, k):

    newAssociatedList=[]

    for i in range(len(candidateItemSet)):
        iList = candidateItemSet[i];
        for j in range(i+1,len(candidateItemSet)):
            jList = candidateItemSet[j];

            if k == 2:
                join = iList[(len(iList)-1)],jList[len(jList)-1]

                newAssociatedList.append(list(join))

            elif(iList[:len(iList)-1] == jList[:len(jList)-1]):
                join = iList[0:(len(iList)-1)],iList[len(iList)-1],jList[len(jList)-1]
                new = list(iList[0:(len(iList)-1)])
                new.append(iList[len(iList)-1])
                new.append(jList[len(jList)-1])

                newAssociatedList.append(new)


    return newAssociatedList

#Method use to compute candidate itemset and return list that satisfy min sup
def reduceBySupport(candidateItemSet, min_sup, trans_len):

    itemSet_dict = {}

    #first count support
    for t in trans:

        for n,itemSet in enumerate(candidateItemSet):
            index = itemSet

            if n not in itemSet_dict:
                itemSet_dict[n] = 0
            c=0
```

```
            for i in itemSet:
                #print("item value:", i)
                if i in t:
                    c=c+1;

            if c == len(itemSet):

                itemSet_dict[n] = itemSet_dict[n] + 1

    #compare with min support
    reduced_list = []

    for index,count in itemSet_dict.items():
        if count/trans_len >= min_sup:
            reduced_list.append(candidateItemSet[index])

            print(f'{candidateItemSet[index]} => {count/trans_len} (support)')

    return reduced_list


#check support of transactions in association rules
def checkSupportOfList(itemSet):
    #print(itemSet)
    sup_count = 0;
    for t in trans:
        c=0;
        for i in itemSet:
            i = str(i)
            if i in t:
                c=c+1

        if c == len(itemSet):
            sup_count = sup_count + 1

    sup_val = sup_count/len(trans)

    #print(f'{itemSet} -> {sup_count}')
    return sup_val
```

```
#Check confidence of association rules
def checkConf(candi_rule):

    temp_list = []
    combine_list = []

    for i in candi_rule:
        temp_list.append(list(i))

    for j in temp_list:
        for k in j:
            combine_list.append(k)

    rule_confidence = checkSupportOfList(combine_list)/checkSupportOfList(temp_list[0])


    if rule_confidence >= min_conf:
        conf_val=rule_confidence
        sup_val=checkSupportOfList(combine_list)
        return candi_rule, sup_val, conf_val

    return [],'',''


#Generate combinations of association rules from frequent itemsets
def gen_combi(itemlists):

    for ilist in itemlists:
        li_len = len(ilist)

        listset = set(tuple(ilist))
        tempFreqItemset = listset

        while li_len > 1:
            mainset = set()

            for c in combinations(tempFreqItemset, li_len - 1):
                mainset = set(c)

                rule = (mainset, listset - mainset)
```

```
            valid_rules,sup_val,conf_val = checkConf(list(rule))

            #Result the final rules which satisfy min confidence
            if valid_rules != []:
                print(f'{valid_rules[0]} -> {valid_rules[1]} => confidence={conf_val} ,
support={sup_val}')

        li_len=li_len-1;


while True:

    try:
        inp_file = input("Please enter file name: ")

        file=open(inp_file, 'r')

        trans_dict={}

        trans=[]
        for line in file:

            (k,v) = line.strip().split('-')
            trans_dict[k]= v
            trans.append(v)

    except FileNotFoundError:
        print('Either file is missing or filename is wrong')


    print("\nDataset Transactions:\n")

    for tr in trans_dict:
        print(f'{tr} - {trans_dict[tr]}')



    min_sup= float(input("Please enter minimum support="))
    min_conf=float(input("Please enter minimum confidence="))
```

```
tr_list, tr_dict = generateSupportCount(trans)


print("\n")

print("Frequent 1 itemset:")
frequentOne = satisfySupport(tr_dict, min_sup, trans, len(trans))


k=2

freqSetList = []

while(frequentOne != []):

    #make join operation to associate the transactions
    candidate_k_itemSet = associateTrans(frequentOne, k)

    print(f'\nFrequent {k} itemset:')

    #Generate frequent k itemset
    frequentOne = reduceBySupport(candidate_k_itemSet, min_sup, len(trans))

    if frequentOne != []:
        frequent_k_itemSet = frequentOne

        freqSetList.append(frequent_k_itemSet)

    else:
        print("NULL")

    k =  k + 1


#Generate Rules from frequent k itemsets
print('\n\nRules:\n')
if freqSetList != []:
    for iList in freqSetList:
        gen_combi(iList)
```

```
else:
    print("No association rules found")


keepOn = input("Press any key to run again. | Press q to quit = ")

if keepOn in ('q','Q'):
    break;

else:
    continue;
```

Documentation:

- My program is starting with asking user to give file name (with path). Then, program will show input transactions.
- Next user have to give min support and min confidence value at run time.
- Based on min support value, method named **satisfySupport(tr_dict, min_sup, trans, len(trans))** will computer support of all transactions and return frequent 1 item set who satisfy min support.
- Now, we've frequent 1 itemset. So, starts with k=2, in while loop program will repeatedly call **reduceBySupport(candidate_k_itemSet, min_sup, len(trans))** method which will generate frequent k itemset.
- reducedBySupport() method will take each list of transaction from candidate k itemset and compare it with original transaction. Then, it check for min support condition. Only returns those transactions who satisfy min support.
- All frequent k itemsets are stored in **freqSetList** List.
- After generating frequent k itemsets, for each itemset **gen_combi(iList)** will generate association rules.
- Inside **gen_combi(iList)** method, for each transaction it will call **checkSupportOfList(itemSet)** method and **checkConf(candi_rule)** method to check min confidence of each association rule.
- Finally it will return association rules which satisfy both min support and min confidence.
- At the end, my program also ask user to continue with other dataset file or want to quit.

❖ Dataset 1

sample1.txt - Notepad

File Edit Format View Help

```
t1-shirt,pant,hat
t2-glasses,shirt,pen
t3-hat,belt,watch,pant,shoes
t4-bag,shirt
t5-tape,belt,pen,shoes,watch,pant
t6-suit,tie,hat,belt
t7-shoes,watch,pen,glasses,pant
t8-suit,tape,tie,belt
t9-shirt,shoes,bag,watch
t10-suit,belt,shoes,watch,shirt,pant,tie
t11-glasses,hat,tape,tie,suit,belt
t12-pen,pant,watch,suit,shoes
t13-shoes,tape,watch
t14-suit,tie,glasses,belt
t15-hat,pen,tape,belt,shirt
t16-shoes,tie,watch,pant,suit,belt
t17-tape,shirt,bag
t18-suit,pen,watch,tie,belt
t19-shoes,watch,hat,beg,pen,pant
t20-bag,tape,pen
```

IPython console

Console 1/A

```
In [24]: runfile('C:/Users/Kelvin/Documents/Python Scripts/Ap_py.py', wdir='C:/Users/Kelvin/Documents/Python
Scripts')

Please enter file name: C:/Users/Kelvin/Documents/Input files/sample1.txt

Dataset Transactions:

t1 - shirt,pant,hat
t2 - glasses,shirt,pen
t3 - hat,belt,watch,pant,shoes
t4 - bag,shirt
t5 - tape,belt,pen,shoes,watch,pant
t6 - suit,tie,hat,belt
t7 - shoes,watch,pen,glasses,pant
t8 - suit,tape,tie,belt
t9 - shirt,shoes,bag,watch
t10 - suit,belt,shoes,watch,shirt,pant,tie
t11 - glasses,hat,tape,tie,suit,belt
t12 - pen,pant,watch,suit,shoes
t13 - shoes,tape,watch
t14 - suit,tie,glasses,belt
t15 - hat,pen,tape,belt,shirt
t16 - shoes,tie,watch,pant,suit,belt
t17 - tape,shirt,bag
t18 - suit,pen,watch,tie,belt
t19 - shoes,watch,hat,beg,pen,pant
t20 - bag,tape,pen

Please enter minimum support=0.25

Please enter minimum confidence=0.50



Frequent 1-itemset:
```

IPython console

Console 1/A

```
Please enter minimum confidence=0.50



Frequent 1 itemset:
[shirt] => 0.35 (support)
[pant] => 0.4 (support)
[hat] => 0.3 (support)
[pen] => 0.4 (support)
[belt] => 0.5 (support)
[watch] => 0.5 (support)
[shoes] => 0.45 (support)
[tape] => 0.35 (support)
[suit] => 0.4 (support)
[tie] => 0.35 (support)

Frequent 2 itemset:
['pant', 'watch'] => 0.35 (support)
['pant', 'shoes'] => 0.35 (support)
['pen', 'watch'] => 0.25 (support)
['belt', 'watch'] => 0.25 (support)
['belt', 'suit'] => 0.35 (support)
['belt', 'tie'] => 0.35 (support)
['watch', 'shoes'] => 0.45 (support)
['suit', 'tie'] => 0.35 (support)

Frequent 3 itemset:
['pant', 'watch', 'shoes'] => 0.35 (support)
['belt', 'suit', 'tie'] => 0.35 (support)

Frequent 4 itemset:
NULL


Rules:
```

IPython console

Console 1/A

```
Frequent 4 itemset:
NULL


Rules:

{'pant'} -> {'watch'} => confidence=0.8749999999999999 , support=0.35
{'watch'} -> {'pant'} => confidence=0.7 , support=0.35
{'pant'} -> {'shoes'} => confidence=0.8749999999999999 , support=0.35
{'shoes'} -> {'pant'} => confidence=0.7777777777777777 , support=0.35
{'watch'} -> {'pen'} => confidence=0.5 , support=0.25
{'pen'} -> {'watch'} => confidence=0.625 , support=0.25
{'watch'} -> {'belt'} => confidence=0.5 , support=0.25
{'belt'} -> {'watch'} => confidence=0.5 , support=0.25
{'suit'} -> {'belt'} => confidence=0.8749999999999999 , support=0.35
{'belt'} -> {'suit'} => confidence=0.7 , support=0.35
{'tie'} -> {'belt'} => confidence=1.0 , support=0.35
{'belt'} -> {'tie'} => confidence=0.7 , support=0.35
{'shoes'} -> {'watch'} => confidence=1.0 , support=0.45
{'watch'} -> {'shoes'} => confidence=0.9 , support=0.45
{'suit'} -> {'tie'} => confidence=0.8749999999999999 , support=0.35
{'tie'} -> {'suit'} => confidence=1.0 , support=0.35
{'pant', 'shoes'} -> {'watch'} => confidence=1.0 , support=0.35
{'pant', 'watch'} -> {'shoes'} => confidence=1.0 , support=0.35
{'shoes', 'watch'} -> {'pant'} => confidence=0.7777777777777777 , support=0.35
{'pant'} -> {'shoes', 'watch'} => confidence=0.8749999999999999 , support=0.35
{'shoes'} -> {'pant', 'watch'} => confidence=0.7777777777777777 , support=0.35
{'watch'} -> {'pant', 'shoes'} => confidence=0.7 , support=0.35
{'suit', 'tie'} -> {'belt'} => confidence=1.0 , support=0.35
{'suit', 'belt'} -> {'tie'} => confidence=1.0 , support=0.35
{'tie', 'belt'} -> {'suit'} => confidence=1.0 , support=0.35
{'suit'} -> {'tie', 'belt'} => confidence=0.8749999999999999 , support=0.35
{'tie'} -> {'suit', 'belt'} => confidence=1.0 , support=0.35
{'belt'} -> {'suit', 'tie'} => confidence=0.7 , support=0.35
```

- **Dataset 2**

sample2.txt - Notepad

File Edit Format View Help

```
t1-fan,light,bulb,sofa,table
t2-chair,table,fan,bulb,bed,mattress
t3-bag,light,sofa
t4-clock,bulb,fan
t5-bed,sofa,bag,light,mattress,fan,bulb
t6-mattress,bed
t7-sofa,table,fan,bulb
t8-clock,table,light
t9-mattress,bed,fan,clock,bulb,table
t10-light,bag,table
t11-chair,sofa,table
t12-bulb,mattress,table
t13-bag,chair,clock
t14-mattress,bed,fan,clock,chair,bulb,table
t15-bed,clock,light,fan
t16-table,light,mattress
t17-bulb,bag,table,mattress,bed,chair,light,fan
t18-sofa,fan,light,bulb
t19-chair,clock
t20-mattress,bed,fan,table,sofa,bulb
```

IPython console

Console 1/A

```
In [4]: runfile('C:/Users/Kelvin/Documents/Python Scripts/Ap_py.py', wdir='C:/Users/Kelvin/Documents/Python
Scripts')

Please enter file name: C:/Users/Kelvin/Documents/Input files/sample2.txt

Dataset Transactions:

t1 - fan,light,bulb,sofa,table
t2 - chair,table,fan,bulb,bed,mattress
t3 - bag,light,sofa
t4 - clock,bulb,fan
t5 - bed,sofa,bag,light,mattress,fan,bulb
t6 - mattress,bed
t7 - sofa,table,fan,bulb
t8 - clock,table,light
t9 - mattress,bed,fan,clock,bulb,table
t10 - light,bag,table
t11 - chair,sofa,table
t12 - bulb,mattress,table
t13 - bag,chair,clock
t14 - mattress,bed,fan,clock,chair,bulb,table
t15 - bed,clock,light,fan
t16 - table,light,mattress
t17 - bulb,bag,table,mattress,bed,chair,light,fan
t18 - sofa,fan,light,bulb
t19 - chair,clock
t20 - mattress,bed,fan,table,sofa,bulb

Please enter minimum support=0.30

Please enter minimum confidence=0.70


Frequent 1 itemset:
[fan] -> 0.55 (support)
```

IPython console

Console 1/A

```
Frequent 1 itemset:
[fan] => 0.55 (support)
[light] => 0.45 (support)
[bulb] => 0.55 (support)
[sofa] => 0.35 (support)
[table] => 0.6 (support)
[chair] => 0.3 (support)
[bed] => 0.4 (support)
[mattress] => 0.45 (support)
[clock] => 0.35 (support)

Frequent 2 itemset:
['fan', 'bulb'] => 0.5 (support)
['fan', 'table'] => 0.35 (support)
['fan', 'bed'] => 0.35 (support)
['fan', 'mattress'] => 0.3 (support)
['bulb', 'table'] => 0.4 (support)
['bulb', 'bed'] => 0.3 (support)
['bulb', 'mattress'] => 0.35 (support)
['table', 'mattress'] => 0.35 (support)
['bed', 'mattress'] => 0.35 (support)

Frequent 3 itemset:
['fan', 'bulb', 'table'] => 0.35 (support)
['fan', 'bulb', 'bed'] => 0.3 (support)
['fan', 'bulb', 'mattress'] => 0.3 (support)
['fan', 'bed', 'mattress'] => 0.3 (support)
['bulb', 'table', 'mattress'] => 0.3 (support)
['bulb', 'bed', 'mattress'] => 0.3 (support)

Frequent 4 itemset:
['fan', 'bulb', 'bed', 'mattress'] => 0.3 (support)

Frequent 5 itemset:
```

IPython console

Console 1/A

```
Frequent 5 itemset:
NULL


Rules:

{'fan'} -> {'bulb'} => confidence=0.9090909090909091 , support=0.5
{'bulb'} -> {'fan'} => confidence=0.9090909090909091 , support=0.5
{'bed'} -> {'fan'} => confidence=0.8749999999999999 , support=0.35
{'bulb'} -> {'table'} => confidence=0.7272727272727273 , support=0.4
{'bed'} -> {'bulb'} => confidence=0.7499999999999999 , support=0.3
{'mattress'} -> {'bulb'} => confidence=0.7777777777777777 , support=0.35
{'mattress'} -> {'table'} => confidence=0.7777777777777777 , support=0.35
{'bed'} -> {'mattress'} => confidence=0.8749999999999999 , support=0.35
{'mattress'} -> {'bed'} => confidence=0.7777777777777777 , support=0.35
{'table', 'fan'} -> {'bulb'} => confidence=1.0 , support=0.35
{'table', 'bulb'} -> {'fan'} => confidence=0.8749999999999999 , support=0.35
{'fan', 'bulb'} -> {'table'} => confidence=0.7 , support=0.35
{'bed', 'fan'} -> {'bulb'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'bulb'} -> {'fan'} => confidence=1.0 , support=0.3
{'bed'} -> {'fan', 'bulb'} => confidence=0.7499999999999999 , support=0.3
{'fan', 'mattress'} -> {'bulb'} => confidence=1.0 , support=0.3
{'bulb', 'mattress'} -> {'fan'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'fan'} -> {'mattress'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'mattress'} -> {'fan'} => confidence=0.8571428571428572 , support=0.3
{'fan', 'mattress'} -> {'bed'} => confidence=1.0 , support=0.3
{'bed'} -> {'fan', 'mattress'} => confidence=0.7499999999999999 , support=0.3
{'table', 'bulb'} -> {'mattress'} => confidence=0.7499999999999999 , support=0.3
{'table', 'mattress'} -> {'bulb'} => confidence=0.8571428571428572 , support=0.3
{'bulb', 'mattress'} -> {'table'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'bulb'} -> {'mattress'} => confidence=1.0 , support=0.3
{'bed', 'mattress'} -> {'bulb'} => confidence=0.8571428571428572 , support=0.3
{'bulb', 'mattress'} -> {'bed'} => confidence=0.8571428571428572 , support=0.3
{'bed'} -> {'bulb', 'mattress'} => confidence=0.7499999999999999 , support=0.3
```

IPython console

Console 1/A

```
{'bed'} -> {'mattress'} => confidence=0.8749999999999999 , support=0.35
{'mattress'} -> {'bed'} => confidence=0.7777777777777777 , support=0.35
{'table', 'fan'} -> {'bulb'} => confidence=1.0 , support=0.35
{'table', 'bulb'} -> {'fan'} => confidence=0.8749999999999999 , support=0.35
{'fan', 'bulb'} -> {'table'} => confidence=0.7 , support=0.35
{'bed', 'fan'} -> {'bulb'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'bulb'} -> {'fan'} => confidence=1.0 , support=0.3
{'bed'} -> {'fan', 'bulb'} => confidence=0.7499999999999999 , support=0.3
{'fan', 'mattress'} -> {'bulb'} => confidence=1.0 , support=0.3
{'bulb', 'mattress'} -> {'fan'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'fan'} -> {'mattress'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'mattress'} -> {'fan'} => confidence=0.8571428571428572 , support=0.3
{'fan', 'mattress'} -> {'bed'} => confidence=1.0 , support=0.3
{'bed'} -> {'fan', 'mattress'} => confidence=0.7499999999999999 , support=0.3
{'table', 'bulb'} -> {'mattress'} => confidence=0.7499999999999999 , support=0.3
{'table', 'mattress'} -> {'bulb'} => confidence=0.8571428571428572 , support=0.3
{'bulb', 'mattress'} -> {'table'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'bulb'} -> {'mattress'} => confidence=1.0 , support=0.3
{'bed', 'mattress'} -> {'bulb'} => confidence=0.8571428571428572 , support=0.3
{'bulb', 'mattress'} -> {'bed'} => confidence=0.8571428571428572 , support=0.3
{'bed'} -> {'bulb', 'mattress'} => confidence=0.7499999999999999 , support=0.3
{'bed', 'fan', 'bulb'} -> {'mattress'} => confidence=1.0 , support=0.3
{'bed', 'fan', 'mattress'} -> {'bulb'} => confidence=1.0 , support=0.3
{'bed', 'bulb', 'mattress'} -> {'fan'} => confidence=1.0 , support=0.3
{'fan', 'bulb', 'mattress'} -> {'bed'} => confidence=1.0 , support=0.3
{'bed', 'fan'} -> {'bulb', 'mattress'} => confidence=0.8571428571428572 , support=0.3
{'bed', 'bulb'} -> {'fan', 'mattress'} => confidence=1.0 , support=0.3
{'bed', 'mattress'} -> {'fan', 'bulb'} => confidence=0.8571428571428572 , support=0.3
{'fan', 'mattress'} -> {'bed', 'bulb'} => confidence=1.0 , support=0.3
{'bulb', 'mattress'} -> {'bed', 'fan'} => confidence=0.8571428571428572 , support=0.3
{'bed'} -> {'fan', 'bulb', 'mattress'} => confidence=0.7499999999999999 , support=0.3

Press any key to run again. | Press q to quit = 
```

❖ Dataset 3

sample3.txt - Notepad

File Edit Format View Help

```
t1-banana,apple,wholemilk
t2-strawberry,blueberries,orange
t3-almondmilk,soyamilk,wholemilk,paneer
t4-wholemilk,cashew,graps,pistachios,paneer,cheese,banana,apple
t5-apple,banana,wholemilk,skimmilk
t6-graps,apple,orange,carrot
t7-paneer,cheese,cashew,pistachios,wholemilk,yogurt
t8-carrot,eggplant,peanuts,walnut
t9-soyamilk,wholemilk,paneer
t10-almondmilk,soyamilk,wholemilk,eggplant,carrot
t11-tomatoes,spinach,apple,banana,wholemilk,pepper,onion
t12-graps,almondmilk,soyamilk,wholemilk
t13-garlic,onion,pepper,orange,carrot
t14-cheese,cream,wholemilk
t15-butter,cheese,apple,onion,banana,wholemilk,cream
t16-apple,cashew,pistachios,banana,wholemilk,butter,cream,yogurt
t17-cashew,pistachios,almondmilk,strawberry,blueberries,soyamilk,wholemilk
t18-almond,cashew,pistachios,walnut,almond
t19-peanuts,walnut,almond
t20-walnut,almond,cashew,pistachios,peanuts
```

IPython console

Console 1/A

```
In [17]: runfile('C:/Users/Kelvin/Documents/Python Scripts/Ap_py.py', wdir='C:/Users/Kelvin/Documents/Python
Scripts')

Please enter file name: C:/Users/Kelvin/Documents/Input files/sample3.txt

Dataset Transactions:

t1 - banana,apple,wholemilk
t2 - strawberry,blueberries,orange
t3 - almondmilk,soyamilk,wholemilk,paneer
t4 - wholemilk,cashew,graps,pistachios,paneer,cheese,banana,apple
t5 - apple,banana,wholemilk,skimmilk
t6 - graps,apple,orange,carrot
t7 - paneer,cheese,cashew,pistachios,wholemilk,yogurt
t8 - carrot,eggplant,peanuts,walnut
t9 - soyamilk,wholemilk,paneer
t10 - almondmilk,soyamilk,wholemilk,eggplant,carrot
t11 - tomatoes,spinach,apple,banana,wholemilk,pepper,onion
t12 - graps,almondmilk,soyamilk,wholemilk
t13 - garlic,onion,pepper,orange,carrot
t14 - cheese,cream,wholemilk
t15 - butter,cheese,apple,onion,banana,wholemilk,cream
t16 - apple,cashew,pistachios,banana,wholemilk,butter,cream,yogurt
t17 - cashew,pistachios,almondmilk,strawberry,blueberries,soyamilk,wholemilk
t18 - almond,cashew,pistachios,walnut,almond
t19 - peanuts,walnut,almond
t20 - walnut,almond,cashew,pistachios,peanuts

Please enter minimum support=0.25

Please enter minimum confidence=0.80


Frequent 1 itemset:
```

IPython console

Console 1/A

```
Frequent 1 itemset:
[banana] => 0.3 (support)
[apple] => 0.35 (support)
[wholemilk] => 0.65 (support)
[soyamilk] => 0.25 (support)
[cashew] => 0.3 (support)
[pistachios] => 0.3 (support)

Frequent 2 itemset:
['banana', 'apple'] => 0.3 (support)
['banana', 'wholemilk'] => 0.3 (support)
['apple', 'wholemilk'] => 0.3 (support)
['wholemilk', 'soyamilk'] => 0.25 (support)
['cashew', 'pistachios'] => 0.3 (support)

Frequent 3 itemset:
['banana', 'apple', 'wholemilk'] => 0.3 (support)

Frequent 4 itemset:
NULL


Rules:

{'apple'} -> {'banana'} => confidence=0.8571428571428572 , support=0.3
{'banana'} -> {'apple'} => confidence=1.0 , support=0.3
{'banana'} -> {'wholemilk'} => confidence=1.0 , support=0.3
{'apple'} -> {'wholemilk'} => confidence=0.8571428571428572 , support=0.3
{'soyamilk'} -> {'wholemilk'} => confidence=1.0 , support=0.25
{'pistachios'} -> {'cashew'} => confidence=1.0 , support=0.3
{'cashew'} -> {'pistachios'} => confidence=1.0 , support=0.3
{'apple', 'banana'} -> {'wholemilk'} => confidence=1.0 , support=0.3
{'apple', 'wholemilk'} -> {'banana'} => confidence=1.0 , support=0.3
{'banana', 'wholemilk'} -> {'apple'} => confidence=1.0 , support=0.3
```

IPython console

Console 1/A

```
Frequent 2 itemset:
['banana', 'apple'] => 0.3 (support)
['banana', 'wholemilk'] => 0.3 (support)
['apple', 'wholemilk'] => 0.3 (support)
['wholemilk', 'soyamilk'] => 0.25 (support)
['cashew', 'pistachios'] => 0.3 (support)

Frequent 3 itemset:
['banana', 'apple', 'wholemilk'] => 0.3 (support)

Frequent 4 itemset:
NULL


Rules:

{'apple'} -> {'banana'} => confidence=0.8571428571428572 , support=0.3
{'banana'} -> {'apple'} => confidence=1.0 , support=0.3
{'banana'} -> {'wholemilk'} => confidence=1.0 , support=0.3
{'apple'} -> {'wholemilk'} => confidence=0.8571428571428572 , support=0.3
{'soyamilk'} -> {'wholemilk'} => confidence=1.0 , support=0.25
{'pistachios'} -> {'cashew'} => confidence=1.0 , support=0.3
{'cashew'} -> {'pistachios'} => confidence=1.0 , support=0.3
{'apple', 'banana'} -> {'wholemilk'} => confidence=1.0 , support=0.3
{'apple', 'wholemilk'} -> {'banana'} => confidence=1.0 , support=0.3
{'banana', 'wholemilk'} -> {'apple'} => confidence=1.0 , support=0.3
{'apple'} -> {'banana', 'wholemilk'} => confidence=0.8571428571428572 , support=0.3
{'banana'} -> {'apple', 'wholemilk'} => confidence=1.0 , support=0.3

Press any key to run again. | Press q to quit = 
```

- **Dataset 4**

sample4.txt - Notepad

File Edit Format View Help

```
t1-iphone,airpod,ipad,laptop,apple
t2-android,google,tablet
t3-google,android,hp,windows
t4-windows,android,google,tablet
t5-apple,airpod,android,hp
t6-ipad,airpod
t7-airpod,iphone,apple
t8-tablet,android,ipad,hp
t9-laptop,android,google,tablet
t10-hp,windows,tablet,laptop
t11-iphone,airpod,ipad,android,google,tablet
t12-laptop,iphone,ipad
t13-iphone,airpod,ipad,apple
t14-android,google,tablet
t15-iphone,airpod,ipad
t16-android,google,tablet
t17-iphone,airpod,ipad,apple
t18-android,google,tablet
t19-iphone,airpod,ipad,windows,laptop
t20-iphone,airpod,ipad,hp
```

IPython console

Console 1/A

```
In [10]: runfile('C:/Users/Kelvin/Documents/Python Scripts/Ap_py.py', wdir='C:/Users/Kelvin/Documents/Python
Scripts')

Please enter file name: C:/Users/Kelvin/Documents/Input files/sample4.txt

Dataset Transactions:

t1 - iphone,airpod,ipad,laptop,apple
t2 - android,google,tablet
t3 - google,android,hp,windows
t4 - windows,android,google,tablet
t5 - apple,airpod,android,hp
t6 - ipad,airpod
t7 - airpod,iphone,apple
t8 - tablet,android,ipad,hp
t9 - laptop,android,google,tablet
t10 - hp,windows,tablet,laptop
t11 - iphone,airpod,ipad,android,google,tablet
t12 - laptop,iphone,ipad
t13 - iphone,airpod,ipad,apple
t14 - android,google,tablet
t15 - iphone,airpod,ipad
t16 - android,google,tablet
t17 - iphone,airpod,ipad,apple
t18 - android,google,tablet
t19 - iphone,airpod,ipad,windows,laptop
t20 - iphone,airpod,ipad,hp

Please enter minimum support=0.35

Please enter minimum confidence=0.75


Frequent 1 itemset:
```

IPython console

Console 1/A

```
t20 - iphone,airpod,ipad,hp

Please enter minimum support=0.35

Please enter minimum confidence=0.75


Frequent 1 itemset:
[iphone] => 0.45 (support)
[airpod] => 0.5 (support)
[ipad] => 0.5 (support)
[android] => 0.5 (support)
[google] => 0.4 (support)
[tablet] => 0.45 (support)

Frequent 2 itemset:
['iphone', 'airpod'] => 0.4 (support)
['iphone', 'ipad'] => 0.4 (support)
['airpod', 'ipad'] => 0.4 (support)
['android', 'google'] => 0.4 (support)
['android', 'tablet'] => 0.4 (support)
['google', 'tablet'] => 0.35 (support)

Frequent 3 itemset:
['iphone', 'airpod', 'ipad'] => 0.35 (support)
['android', 'google', 'tablet'] => 0.35 (support)

Frequent 4 itemset:
NULL


Rules:

{'airpod'} -> {'iphone'} => confidence=0.8 , support=0.4
```

IPython console

Console 1/A

```
Rules:

{'airpod'} -> {'iphone'} => confidence=0.8 , support=0.4
{'iphone'} -> {'airpod'} => confidence=0.888888888888889 , support=0.4
{'ipad'} -> {'iphone'} => confidence=0.8 , support=0.4
{'iphone'} -> {'ipad'} => confidence=0.888888888888889 , support=0.4
{'ipad'} -> {'airpod'} => confidence=0.8 , support=0.4
{'airpod'} -> {'ipad'} => confidence=0.8 , support=0.4
{'android'} -> {'google'} => confidence=0.8 , support=0.4
{'google'} -> {'android'} => confidence=1.0 , support=0.4
{'android'} -> {'tablet'} => confidence=0.8 , support=0.4
{'tablet'} -> {'android'} => confidence=0.888888888888889 , support=0.4
{'tablet'} -> {'google'} => confidence=0.7777777777777777 , support=0.35
{'google'} -> {'tablet'} => confidence=0.8749999999999999 , support=0.35
{'ipad', 'airpod'} -> {'iphone'} => confidence=0.8749999999999999 , support=0.35
{'ipad', 'iphone'} -> {'airpod'} => confidence=0.8749999999999999 , support=0.35
{'airpod', 'iphone'} -> {'ipad'} => confidence=0.8749999999999999 , support=0.35
{'iphone'} -> {'ipad', 'airpod'} => confidence=0.7777777777777777 , support=0.35
{'android', 'tablet'} -> {'google'} => confidence=0.8749999999999999 , support=0.35
{'android', 'google'} -> {'tablet'} => confidence=0.8749999999999999 , support=0.35
{'google', 'tablet'} -> {'android'} => confidence=1.0 , support=0.35
{'tablet'} -> {'android', 'google'} => confidence=0.7777777777777777 , support=0.35
{'google'} -> {'android', 'tablet'} => confidence=0.8749999999999999 , support=0.35

Press any key to run again. | Press q to quit = |
```

❖ Dataset 5

sample5.txt - Notepad

File Edit Format View Help

```
t1-medicine,sanitizer
t2-shampoo,milk,conditioner
t3-milk,medicine,sanitizer
t4-conditioner,shampoo,soap,basket
t5-medicine,sanitizer,conditioner,shampoo,basket,milk
t6-conditioner,milk,shampoo,soap
t7-medicine,sanitizer
t8-gloves,brace,lipbalm
t9-basket,brace,shampoo,conditioner
t10-medicine,sanitizer,conditioner
t11-milk,basket
t12-basket,medicine,sanitizer,brace,gloves
t13-soap,basket,lipbalm
t14-shampoo,conditioner,lipbalm,soap
t15-conditioner,shampoo,lipbalm,gloves,basket
t16-lipbalm,medicine,sanitizer
t17-brace,medicine,sanitizer,gloves,soap
t18-gloves,medicine,sanitizer,brace,shampoo,conditioner
t19-sanitizer,lipbalm,conditioner,shampoo
t20-medicine,milk,brace,sanitizer,gloves
```

IPython console

Console 1/A

```
In [15]: runfile('C:/Users/Kelvin/Documents/Python Scripts/Ap_py.py', wdir='C:/Users/Kelvin/Documents/Python Scripts')

Please enter file name: C:/Users/Kelvin/Documents/Input files/sample5.txt

Dataset Transactions:

t1 - medicine,sanitizer
t2 - shampoo,milk,conditioner
t3 - milk,medicine,sanitizer
t4 - conditioner,shampoo,soap,basket
t5 - medicine,sanitizer,conditioner,shampoo,basket,milk
t6 - conditioner,milk,shampoo,soap
t7 - medicine,sanitizer
t8 - gloves,brace,lipbalm
t9 - basket,brace,shampoo,conditioner
t10 - medicine,sanitizer,conditioner
t11 - milk,basket
t12 - basket,medicine,sanitizer,brace,gloves
t13 - soap,basket,lipbalm
t14 - shampoo,conditioner,lipbalm,soap
t15 - conditioner,shampoo,lipbalm,gloves,basket
t16 - lipbalm,medicine,sanitizer
t17 - brace,medicine,sanitizer,gloves,soap
t18 - gloves,medicine,sanitizer,brace,shampoo,conditioner
t19 - sanitizer,lipbalm,conditioner,shampoo
t20 - medicine,milk,brace,sanitizer,gloves

Please enter minimum support=0.45

Please enter minimum confidence=0.85


Frequent 1 itemset:
```

IPython console

Console 1/A

```
Please enter minimum support=0.45

Please enter minimum confidence=0.85


Frequent 1 itemset:
[medicine] => 0.5 (support)
[sanitizer] => 0.55 (support)
[shampoo] => 0.45 (support)
[conditioner] => 0.5 (support)

Frequent 2 itemset:
['medicine', 'sanitizer'] => 0.5 (support)
['shampoo', 'conditioner'] => 0.45 (support)

Frequent 3 itemset:
NULL


Rules:

{'medicine'} -> {'sanitizer'} => confidence=1.0 , support=0.5
{'sanitizer'} -> {'medicine'} => confidence=0.9090909090909091 , support=0.5
{'shampoo'} -> {'conditioner'} => confidence=1.0 , support=0.45
{'conditioner'} -> {'shampoo'} => confidence=0.9 , support=0.45

Press any key to run again. | Press q to quit =
```