

DATAMININGMIDTERMPROJECT

First Name: Anvay

Last Name: Awalgaonkar

UCID: aa3272

Email: aa3272@njit.edu

SOURCE CODE FOR BRUTE FORCE

```
import pandas as pd
import time

filePath = input("Please provide the path of the file: ")

start_time = time.time()

transactionss = pd.read_csv(filePath, header=None, delimiter=',', engine='python', names=range(100))
transactionss = transactionss.where((pd.notnull(transactionss)), None)

transactionItemMatrix = pd.get_dummies(transactionss.unstack().dropna()).groupby(level=1).sum()
transactionItemMatrix
```

Out[1]:

	Nike	Nike BackPack	Nike Backpack	Nike Cap	Nike Chair	Nike Door	Nike Jacket	Nike Ring	Nike Shirt	Nike Shoes	Nike Shoes	Nike Slippers	Nike Sweatshirt
0	0	1	0	0	1	0	1	0	0	1	0	0	1
1	0	0	1	1	0	0	0	1	0	1	0	0	1
2	0	1	0	0	0	0	0	0	0	1	0	1	0
3	0	0	0	0	0	0	1	0	1	1	0	0	0
4	0	1	0	0	0	1	0	1	0	0	0	0	1
5	0	1	0	0	1	0	0	0	0	1	0	0	0
6	0	1	0	0	1	0	0	0	0	1	0	0	1
7	0	0	0	0	0	0	0	1	0	1	0	0	1
8	0	0	0	0	0	0	1	0	1	0	0	0	0
9	0	1	0	1	0	0	0	0	0	1	0	0	1
10	1	1	0	0	1	0	0	0	0	1	0	0	0
11	0	0	0	0	0	0	0	1	0	1	0	0	1
12	0	1	0	0	0	1	0	0	0	1	0	0	1
13	0	1	0	0	0	0	1	0	0	0	0	0	0
14	0	0	0	1	0	0	0	0	0	1	0	0	0
15	0	1	0	1	0	0	0	0	0	1	0	0	1
16	0	1	0	0	0	1	0	0	0	0	0	0	1
17	0	0	0	0	0	0	0	1	0	0	1	1	0
18	0	1	0	0	0	0	0	0	0	1	0	1	1
19	0	1	0	0	0	0	1	0	1	1	0	1	1

Nike Sweatshirt	Nike Table	Nike TrackPant	Nike Trackpant	Nike Watch
0	0	1	0	0
0	0	0	0	0
0	0	1	0	1
0	0	1	0	0
0	0	0	0	1
0	1	0	1	0
0	0	0	1	0
0	0	0	0	1
0	0	0	1	1
0	0	0	0	0
0	1	0	0	0
0	0	1	0	0
0	0	0	0	0
1	1	1	0	0
0	0	1	0	1
0	0	1	0	0
0	0	1	0	0
0	1	1	0	0
0	0	1	0	0
0	0	1	0	0

```

transactionCnt, itemCnt = transactionItemMatrix.shape

largeitms = []
for items in range(1, itemCnt+1):
    cnt=0
    from itertools import combinations
    for itemset in combinations(transactionItemMatrix, items):
        cnt+=1
        itemsetSupport = transactionItemMatrix[list(itemset)].all(axis=1).sum()
        s = [str(x) for x in itemset]
        if (len(s) >= 1):
            largeitms.append([".".join(s), itemsetSupport])
    print("No. : ",cnt)
freqItem = pd.DataFrame(largeitms, columns=["Itemset", "Support"])
results = freqItem[freqItem.Support >= 2]
print(results)

mdm = open('output.txt', 'w')

for eachitem in largeitms:
    mdm.write(str(eachitem)+'\n')
mdm.close()

print("- %s seconds -" % (time.time() - start_time))

```

O/P

No. : 19
No. : 171
No. : 969
No. : 3876
No. : 11628
No. : 27132
No. : 50388
No. : 75582
No. : 92378
No. : 92378
No. : 75582
No. : 50388
No. : 27132
No. : 11628
No. : 3876
No. : 969
No. : 171
No. : 19
No. : 1

	Itemset	Support
1	Nike BackPack	13
3	Nike Cap	4
4	Nike Chair	4
5	Nike Door	3
6	Nike Jacket	5
...
4381	Nike Jacket,Nike Shirt,Nike Shoes,Nike TrackPant	2
4438	Nike Jacket,Nike Shoes,Nike Sweatshirt,Nike Tr...	2
4856	Nike Shoes,Nike Slippers,Nike Sweatshirt,Nike ...	2
9878	Nike BackPack,Nike Jacket,Nike Shoes,Nike Swea...	2
10296	Nike BackPack,Nike Shoes,Nike Slippers,Nike Sw...	2

[107 rows x 2 columns]

- 966.2205889225006 seconds -

Apriori Algorithm Source Code

```
import sys
from itertools import combinations
from itertools import permutations
import time
from typing import Dict
#How to run
#python Algo.py datasetname {minimum support(in percentage)} {minimum confidence(in percentage)}
def print_item_set(frequent_set,n):
    print("#####")
    print("Selected itemsets after ",n," iteration")
    print("__Itemset__", "__Support__")
    print("#####")
    print()
    for i in frequent_set:
        print(i,round(frequent_set[i]*100/total_no_tx,2),)
    print()

def find_freq_set(list1, notconsider, item_set_list, n):
    comb = combinations(list1, n)
    item_support_count = {}
    for i in comb:
        set_i = set(i)
        i=tuple(sorted(i))
        for j in item_set_list:
            if set_i.issubset(j):
                if notconsider:
                    count = 0
                    for k in notconsider:
                        if k.issubset(set_i):
                            count = 1
                            break
                    if not count:
                        if i in item_support_count:
                            item_support_count[i] += 1
                        else:
                            item_support_count[i] = 1
                else:
                    if i in item_support_count:
                        item_support_count[i] += 1
                    else:
                        item_support_count[i] = 1
    frequent_set_to_return = {}
    rejected_set_to_return = []

    # print(item_support_count)
    if item_support_count:
        print("#####")
        print("Itemsets for ", n, "Iteration")
        print("#####")
        print()
```

```

        for i in item_support_count:
            print(i,round(item_support_count[i]*100/total_no_tx,2))
            if (item_support_count[i]/total_no_tx)*100 >= min_supp:
                frequent_set_to_return[i] = item_support_count[i]
            else:
                rejected_set_to_return.append(set(list(i)))
    print()
    if frequent_set_to_return:
        print_item_set(frequent_set_to_return,n)
        support_of_all_item_set.update(item_support_count)
        generate_association_rule(frequent_set_to_return)
        return frequent_set_to_return, rejected_set_to_return
    return None,None

def generate_association_rule(frequent_set):
    for items_set_tuple in frequent_set.keys():
        print("Association Rule for itemset - ",items_set_tuple)
        print("__Rules__","__Confidence__")
        size_of_item_set=len(items_set_tuple)
        itemset=set(items_set_tuple)
        while size_of_item_set-1>0:
            comb = combinations(items_set_tuple, size_of_item_set-1)
            for i in comb:
                left_side_items=i
                right_side_items=tuple(itemset-set(i))
                item_conf=round(support_of_all_item_set[items_set_tuple]*100/support_of_all_item_set[left_side_items],2)
                if item_conf>=min_conf:
                    print(left_side_items,"=>",right_side_items,item_conf,"Rule Selected")
                else:
                    print(left_side_items,"=>",right_side_items,item_conf,"Rule Rejected")

            size_of_item_set -=1
        print()

st = time.time()
###Program Start from here###
file_name = r"C:\Users\aa3272\Desktop\Nike Transactions Dataset.csv"
file_object = open(file_name, "r")
lines = file_object.readlines()
all_tx = []
total_no_tx=0
support_of_all_item_set={}
min_supp = 0.5
min_conf = 0.33

```

```

print("*****")
print()

for line in lines:
    line = line.replace("\n", "")
    print(line)
    all_tx.append("".join(line.split(" ")[1].split(",")))
    seen = set()
    for i in "".join(line.split(" ")[1:]).split(","):

        if (i,) in c1:
            c1[(i,)] += 1
        else:
            # print((i,),line)
            c1[(i,)] = 1
            seen.add(i)
    item_set_list.append(seen)
    total_no_tx+=1
    # print(seen)
frequent_set = {}
# print(c1)
rejected_set = []
print()
print("*****")
print("Item Sets", 1, "Iteration")
print("*****")
print()
for i in c1:
    print(i,round(c1[i]*100/total_no_tx))
    if (c1[i]/20)*100 >= min_supp:
        frequent_set[i] = c1[i]
    else:
        rejected_set.append(set(i))
support_of_all_item_set.update(c1)
list1 =[item[0] for item in frequent_set.keys()]

print()
print_item_set(frequent_set,1)
item_set_size = 1
while len(list1) > item_set_size:
    frequent_set1, rejected_set1 = find_freq_set(list1, rejected_set, item_set_list, item_set_size + 1)
    if not frequent_set1:
        break
    item_list = [items for item_tuples in list(frequent_set1.keys()) for items in item_tuples]
    list1 = list(set(item_list))
    rejected_set = rejected_set1
    frequent_set=frequent_set1
    item_set_size += 1

```


min_supp = 0.5

min_conf = 0.33

INPUT

1	Nike Shoes	Nike Sweatshirt	Nike TrackPant	Nike Backpack	Nike Jacket	Nike Chair	
2	Nike Shoes	Nike Sweatshirt	Nike Backpack	Nike Cap	Nike Ring		
3	Nike Shoes	Nike TrackPant	Nike Backpack	Nike Watch	Nike Slippers		
4	Nike TrackPant	Nike Shoes	Nike Jacket	Nike Shirt			
5	Nike Sweatshirt	Nike Backpack	Nike Watch	Nike Ring	Nike Door		
6	Nike Trackpant	Nike Shoes	Nike Backpack	Nike Table	Nike Chair		
7	Nike Backpack	Nike Sweatshirt	Nike Shoes	Nike Trackpant	Nike Chair		
8	Nike Shoes	Nike Sweatshirt	Nike Watch	Nike Ring			
9	Nike Trackpant	Nike Sweatshirt	Nike Watch	Nike Jacket	Nike Shirt		
10	Nike Backpack	Nike Shoes	Nike Sweatshirt	Nike Cap			
11	Nike Shoes	Nike Backpack	Nike	Nike Table	Nike Chair		
12	Nike TrackPant	Nike Shoes	Nike Sweatshirt	Nike Ring			
13	Nike Sweatshirt	Nike Backpack	Nike Shoes	Nike Door			
14	Nike Backpack	Nike Sweatshirt	Nike TrackPant	Nike Jacket	Nike Table		
15	Nike TrackPant	Nike Shoes	Nike Cap	Nike Watch			
16	Nike Shoes	Nike Sweatshirt	Nike Backpack	Nike TrackPant	Nike Cap		
17	Nike Sweatshirt	Nike TrackPant	Nike Backpack	Nike Door			
18	Nike TrackPant	Nike Shoes	Nike Ring	Nike Table	Nike Slippers		
19	Nike Backpack	Nike Sweatshirt	Nike Shoes	Nike TrackPant	Nike Slippers		
20	Nike TrackPant	Nike Sweatshirt	Nike Backpack	Nike Shoes	Nike Jacket	Nike Shirt	Nike Slippers

OUTPUT

Nike TrackPant,Nike Sweatshirt,Nike BackPack,Nike Shoes,Nike Jacket,Nike Shirt,Nike Slippers

Association Rule for itemset - ('NikeBackPack', 'NikeJacket', 'NikeShirt', 'NikeShoes', 'NikeSlippers', 'NikeSweatshirt', 'TrackPant')

Rules	Confidence
('NikeBackPack', 'NikeJacket', 'NikeShirt', 'NikeShoes', 'NikeSlippers', 'NikeSweatshirt') => ('TrackPant',)	100.0 Rule Selected
('NikeBackPack', 'NikeJacket', 'NikeShirt', 'NikeShoes', 'NikeSlippers', 'TrackPant') => ('NikeSweatshirt',)	100.0 Rule Selected
('NikeBackPack', 'NikeJacket', 'NikeShirt', 'NikeShoes', 'NikeSweatshirt', 'TrackPant') => ('NikeSlippers',)	100.0 Rule Selected
('NikeBackPack', 'NikeJacket', 'NikeShirt', 'NikeSlippers', 'NikeSweatshirt', 'TrackPant') => ('NikeShoes',)	100.0 Rule Selected

Association Rule for Itemset = ('', 'NikeBackPack', 'NikeChair', 'NikeJacket', 'NikeSweatshirt', 'NikeTrackPant', 'Shoes')

[illegible]

```

'NikeBackPack', 'NikeChair') => ('', 'NikeTrackPant', 'NikeSweatshirt', 'NikeJacket', 'Shoes') 33.33 Rule Selected
'NikeBackPack', 'NikeJacket') => ('', 'NikeTrackPant', 'NikeSweatshirt', 'Shoes', 'NikeChair') 50.0 Rule Selected
'NikeBackPack', 'NikeSweatshirt') => ('', 'NikeTrackPant', 'NikeJacket', 'Shoes', 'NikeChair') 33.33 Rule Selected
'NikeBackPack', 'NikeTrackPant') => ('', 'NikeSweatshirt', 'NikeJacket', 'Shoes', 'NikeChair') 25.0 Rule Selected
'NikeBackPack', 'Shoes') => ('', 'NikeTrackPant', 'NikeSweatshirt', 'NikeJacket', 'NikeChair') 25.0 Rule Selected
'NikeChair', 'NikeJacket') => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeSweatshirt', 'Shoes') 100.0 Rule Selected
'NikeChair', 'NikeSweatshirt') => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeJacket', 'Shoes') 50.0 Rule Selected
'NikeChair', 'NikeTrackPant') => ('', 'NikeBackPack', 'NikeSweatshirt', 'NikeJacket', 'Shoes') 100.0 Rule Selected
'NikeChair', 'Shoes') => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeSweatshirt', 'NikeJacket') 50.0 Rule Selected
'NikeJacket', 'NikeSweatshirt') => ('', 'NikeBackPack', 'NikeTrackPant', 'Shoes', 'NikeChair') 33.33 Rule Selected
'NikeJacket', 'NikeTrackPant') => ('', 'NikeBackPack', 'NikeSweatshirt', 'Shoes', 'NikeChair') 50.0 Rule Selected
'NikeJacket', 'Shoes') => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeSweatshirt', 'NikeChair') 100.0 Rule Selected
'NikeSweatshirt', 'NikeTrackPant') => ('', 'NikeBackPack', 'NikeJacket', 'Shoes', 'NikeChair') 33.33 Rule Selected
'NikeSweatshirt', 'Shoes') => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeJacket', 'NikeChair') 25.0 Rule Selected
'NikeTrackPant', 'Shoes') => ('', 'NikeBackPack', 'NikeSweatshirt', 'NikeJacket', 'NikeChair') 33.33 Rule Selected
'',) => ('NikeBackPack', 'NikeTrackPant', 'NikeSweatshirt', 'NikeJacket', 'Shoes', 'NikeChair') 2.27 Rule Selected
'NikeBackPack',) => ('', 'NikeTrackPant', 'NikeSweatshirt', 'NikeJacket', 'Shoes', 'NikeChair') 11.11 Rule Selected
'NikeChair',) => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeSweatshirt', 'NikeJacket', 'Shoes') 25.0 Rule Selected
'NikeJacket',) => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeSweatshirt', 'Shoes', 'NikeChair') 20.0 Rule Selected
'NikeSweatshirt',) => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeJacket', 'Shoes', 'NikeChair') 10.0 Rule Selected
'NikeTrackPant',) => ('', 'NikeBackPack', 'NikeSweatshirt', 'NikeJacket', 'Shoes', 'NikeChair') 16.67 Rule Selected
'Shoes',) => ('', 'NikeBackPack', 'NikeTrackPant', 'NikeSweatshirt', 'NikeJacket', 'NikeChair') 16.67 Rule Selected

```

--- 0.6722168922424316 seconds ---

Conclusion for Database 1

Apriori Algorithm is way much faster than Brute Force Method as the time generated in

Apriori Algortihm is --- 0.6722168922424316

Brute Force Method is –

966.2205889225006 seconds –

*****I HAVE DIRECTLY IMPLEMENTED THE SOURCE CODE OF
BRUTE FORCE AND APRIORI ALGORITHM ON DATABASE 1
WHICH CONTAINS
SOURCE CODE******

Apriori Algorithm for Database 2

min_supp = 0.5

min_conf = 0.33

Input

Adidas Sh	Adidas Sw	Adidas Tre	Adidas Ba	Adidas Jac	Adidas Chair		
Adidas Sh	Adidas Sw	Adidas Ba	Adidas Ca	Adidas Ring			
Adidas Sh	Adidas Tre	Adidas Ba	Adidas We	Adidas Slippers			
Adidas Tre	Adidas Sh	Adidas Jac	Adidas Shirt				
Adidas Sw	Adidas Ba	Adidas We	Adidas Rir	Adidas Door			
Adidas Tre	Adidas Sh	Adidas Ba	Adidas Tal	Adidas Chair			
Adidas Ba	Adidas Sw	Adidas Sh	Adidas Tre	Adidas Chair			
Adidas Sh	Adidas Sw	Adidas We	Adidas Ring				
Adidas Tre	Adidas Sw	Adidas We	Adidas Jac	Adidas Shirt			
Adidas Ba	Adidas Sh	Adidas Sw	Adidas Cap				
Adidas Sh	Adidas Ba	Adidas	Adidas Tal	Adidas Chair			
Adidas Tre	Adidas Sh	Adidas Sw	Adidas Ring				
Adidas Sw	Adidas Ba	Adidas Sh	Adidas Door				
Adidas Ba	Adidas Sw	Adidas Tre	Adidas Jac	Adidas Table			
Adidas Tre	Adidas Sh	Adidas Ca	Adidas Watch				
Adidas Sh	Adidas Sw	Adidas Ba	Adidas Tre	Adidas Cap			
Adidas Sw	Adidas Tre	Adidas Ba	Adidas Door				
Adidas Tre	Adidas Sh	Adidas Rir	Adidas Tal	Adidas Slippers			
Adidas Ba	Adidas Sw	Adidas Sh	Adidas Tre	Adidas Slippers			
Adidas Tre	Adidas Sw	Adidas Ba	Adidas Sh	Adidas Jac	Adidas Sh	Adidas Slippers	

Output

Association Rule for itemset - ('AdidasShoes', 'AdidasTrackpant')

__Rules__ __Confidence__

('AdidasShoes',) => ('AdidasTrackpant',) 10.0 Rule Selected

('AdidasTrackpant',) => ('AdidasShoes',) 100.0 Rule Selected

Association Rule for itemset - ('AdidasShirt', 'Trackpant')

__Rules__ __Confidence__

('AdidasShirt',) => ('Trackpant',) 33.33 Rule Selected

('Trackpant',) => ('AdidasShirt',) 50.0 Rule Selected

Association Rule for itemset - ('AdidasDoor', 'Sweatshirt')

__Rules__ __Confidence__

('AdidasDoor',) => ('Sweatshirt',) 100.0 Rule Selected

('Sweatshirt',) => ('AdidasDoor',) 100.0 Rule Selected

```
Rule Selected
('AdidasSweatshirt', 'TrackPant') => ('AdidasShoes', 'AdidasBackPack', 'AdidasSlippers', 'AdidasJacket', 'AdidasShirt') 50.0
Rule Selected
('AdidasBackPack',) => ('TrackPant', 'AdidasShoes', 'AdidasSlippers', 'AdidasJacket', 'AdidasShirt', 'AdidasSweatshirt') 11.1
1 Rule Selected
('AdidasJacket',) => ('TrackPant', 'AdidasShoes', 'AdidasBackPack', 'AdidasSlippers', 'AdidasSweatshirt', 'AdidasShirt') 20.0
Rule Selected
('AdidasShirt',) => ('TrackPant', 'AdidasShoes', 'AdidasBackPack', 'AdidasSlippers', 'AdidasJacket', 'AdidasSweatshirt') 33.3
3 Rule Selected
('AdidasShoes',) => ('TrackPant', 'AdidasBackPack', 'AdidasSlippers', 'AdidasJacket', 'AdidasShirt', 'AdidasSweatshirt') 10.0
Rule Selected
('AdidasSlippers',) => ('TrackPant', 'AdidasShoes', 'AdidasBackPack', 'AdidasJacket', 'AdidasShirt', 'AdidasSweatshirt') 25.0
Rule Selected
('AdidasSweatshirt',) => ('TrackPant', 'AdidasShoes', 'AdidasBackPack', 'AdidasSlippers', 'AdidasJacket', 'AdidasShirt') 10.0
Rule Selected
('TrackPant',) => ('AdidasShoes', 'AdidasBackPack', 'AdidasSlippers', 'AdidasJacket', 'AdidasShirt', 'AdidasSweatshirt') 20.0
Rule Selected
```

--- 0.8886613845825195 seconds ---

BRUTE FORCE OUTPUT FOR DATABASE 2

No.of Iterations : 19
No.of Iterations : 171
No.of Iterations : 969
No.of Iterations : 3876
No.of Iterations : 11628
No.of Iterations : 27132
No.of Iterations : 50388
No.of Iterations : 75582
No.of Iterations : 92378
No.of Iterations : 92378
No.of Iterations : 75582
No.of Iterations : 50388
No.of Iterations : 27132
No.of Iterations : 11628
No.of Iterations : 3876
No.of Iterations : 969
No.of Iterations : 171
No.of Iterations : 19
No.of Iterations : 1

	Itemset	Support
1	Adidas BackPack	13
3	Adidas Cap	4
4	Adidas Chair	4
5	Adidas Door	3
6	Adidas Jacket	5
...
4381	Adidas Jacket,Adidas Shirt,Adidas Shoes,Adidas...	2
4438	Adidas Jacket,Adidas Shoes,Adidas Sweatshirt,A...	2
4856	Adidas Shoes,Adidas Slippers,Adidas Sweatshirt...	2
9878	Adidas BackPack,Adidas Jacket,Adidas Shoes,Adi...	2
10296	Adidas BackPack,Adidas Shoes,Adidas Slippers,A...	2

[107 rows x 2 columns]

- 289.43624210357666 seconds -

Conclusion of Database 2

Apriori Algorithm is way much faster than Brute Force Method as the time generated in

Apriori Algortihm is --- 0.8886613845825195 second

Brute Force Method is –

966.2205889225006 seconds - - 289.43624210357666 second

Apriori Algorithm for Database 3

Input

min_supp = 0.5

min_conf = 0.33

In []:

Zara Shoe	Zara Swea	Zara Track	Zara Back	Zara Jacke	Zara Chair		
Zara Shoe	Zara Swea	Zara Back	Zara Cap	Zara Ring			
Zara Shoe	Zara Track	Zara Back	Zara Watc	Zara Slippers			
Zara Track	Zara Shoe	Zara Jacke	Zara Shirt				
Zara Swea	Zara Back	Zara Watc	Zara Ring	Zara Door			
Zara Track	Zara Shoe	Zara Back	Zara Table	Zara Chair			
Zara Back	Zara Swea	Zara Shoe	Zara Track	Zara Chair			
Zara Shoe	Zara Swea	Zara Watc	Zara Ring				
Zara Track	Zara Swea	Zara Watc	Zara Jacke	Zara Shirt			
Zara Back	Zara Shoe	Zara Swea	Zara Cap				
Zara Shoe	Zara Back	Zara	Zara Table	Zara Chair			
Zara Track	Zara Shoe	Zara Swea	Zara Ring				
Zara Swea	Zara Back	Zara Shoe	Zara Door				
Zara Back	Zara Swea	Zara Track	Zara Jacke	Zara Table			
Zara Track	Zara Shoe	Zara Cap	Zara Watch				
Zara Shoe	Zara Swea	Zara Back	Zara Track	Zara Cap			
Zara Swea	Zara Track	Zara Back	Zara Door				
Zara Track	Zara Shoe	Zara Ring	Zara Table	Zara Slippers			
Zara Back	Zara Swea	Zara Shoe	Zara Track	Zara Slippers			
Zara Track	Zara Swea	Zara Back	Zara Shoe	Zara Jacke	Zara Shirt	Zara Slippers	

Output

```
('ZaraSlippers', 'ZaraSweatshirt') => ('ZaraShoes', 'ZaraShirt', 'ZaraBackPack', 'ZaraJacket', 'TrackPant') 50.0 Rule Selected
('TrackPant',) => ('ZaraShoes', 'ZaraShirt', 'ZaraBackPack', 'ZaraJacket', 'ZaraSlippers', 'ZaraSweatshirt') 20.0 Rule Selected
('ZaraBackPack',) => ('ZaraShoes', 'ZaraShirt', 'ZaraJacket', 'TrackPant', 'ZaraSlippers', 'ZaraSweatshirt') 11.11 Rule Selected
('ZaraJacket',) => ('ZaraShoes', 'ZaraShirt', 'ZaraBackPack', 'TrackPant', 'ZaraSlippers', 'ZaraSweatshirt') 20.0 Rule Selected
('ZaraShirt',) => ('ZaraShoes', 'ZaraBackPack', 'ZaraJacket', 'TrackPant', 'ZaraSlippers', 'ZaraSweatshirt') 33.33 Rule Selected
('ZaraShoes',) => ('ZaraShirt', 'ZaraBackPack', 'ZaraJacket', 'TrackPant', 'ZaraSlippers', 'ZaraSweatshirt') 10.0 Rule Selected
('ZaraSlippers',) => ('ZaraShoes', 'ZaraShirt', 'ZaraBackPack', 'ZaraJacket', 'TrackPant', 'ZaraSweatshirt') 25.0 Rule Selected
('ZaraSweatshirt',) => ('ZaraShoes', 'ZaraShirt', 'ZaraBackPack', 'ZaraJacket', 'TrackPant', 'ZaraSlippers') 10.0 Rule Selected

--- 0.7879278659828557 seconds ---
```

BRUTE FORCE OUTPUT FOR DATABASE 3

No.of Iterations : 19
No.of Iterations : 171
No.of Iterations : 969
No.of Iterations : 3876
No.of Iterations : 11628
No.of Iterations : 27132
No.of Iterations : 50388
No.of Iterations : 75582
No.of Iterations : 92378
No.of Iterations : 92378
No.of Iterations : 75582
No.of Iterations : 50388
No.of Iterations : 27132
No.of Iterations : 11628
No.of Iterations : 3876
No.of Iterations : 969
No.of Iterations : 171
No.of Iterations : 19
No.of Iterations : 1

	Itemset	Support
1	Zara BackPack	13
3	Zara Cap	4
4	Zara Chair	4
5	Zara Door	3
6	Zara Jacket	5
...
4381	Zara Jacket,Zara Shirt,Zara Shoes,Zara TrackPant	2
4438	Zara Jacket,Zara Shoes,Zara Sweatshirt,Zara Tr...	2
4856	Zara Shoes,Zara Slippers,Zara Sweatshirt,Zara ...	2
9878	Zara BackPack,Zara Jacket,Zara Shoes,Zara Swea...	2
10296	Zara BackPack,Zara Shoes,Zara Slippers,Zara Sw...	2

[107 rows x 2 columns]

- 281.35256338119507 seconds -

Conclusion of Database 3

Apriori Algorithm is way much faster than Brute Force Method as the time generated in

Apriori Algortihm is --- 0.7879278659820557 second second

Brute Force Method is –

966.2205889225006 seconds - - 281.35256338119507 second

Apriori Algorithm for Database 4

Input

min_supp = 0.5

min_conf = 0.33

Polo Shoe	Polo Swee	Polo Track	Polo Back	Polo Jacke	Polo Chair			
Polo Shoe	Polo Swee	Polo Back	Polo Cap	Polo Ring				
Polo Shoe	Polo Track	Polo Back	Polo Watc	Polo Slippers				
Polo Track	Polo Shoe	Polo Jacke	Polo Shirt					
Polo Swee	Polo Back	Polo Watc	Polo Ring	Polo Door				
Polo Track	Polo Shoe	Polo Back	Polo Table	Polo Chair				
Polo Back	Polo Swee	Polo Shoe	Polo Track	Polo Chair				
Polo Shoe	Polo Swee	Polo Watc	Polo Ring					
Polo Track	Polo Swee	Polo Watc	Polo Jacke	Polo Shirt				
Polo Back	Polo Shoe	Polo Swee	Polo Cap					
Polo Shoe	Polo Back	Polo	Polo Table	Polo Chair				
Polo Track	Polo Shoe	Polo Swee	Polo Ring					
Polo Swee	Polo Back	Polo Shoe	Polo Door					
Polo Back	Polo Swee	Polo Track	Polo Jacke	Polo Table				
Polo Track	Polo Shoe	Polo Cap	Polo Watch					
Polo Shoe	Polo Swee	Polo Back	Polo Track	Polo Cap				
Polo Swee	Polo Track	Polo Back	Polo Door					
Polo Track	Polo Shoe	Polo Ring	Polo Table	Polo Slippers				
Polo Back	Polo Swee	Polo Shoe	Polo Track	Polo Slippers				
Polo Track	Polo Swee	Polo Back	Polo Shoe	Polo Jacke	Polo Shirt	Polo Slippers		

Output

```
('PoloSweatshirt', 'TrackPant') => ('PoloSlippers', 'PoloShirt', 'PoloBackPack', 'PoloJacket', 'PoloShoes') 50.0 Rule Selected
('PoloBackPack',) => ('PoloSlippers', 'PoloShirt', 'TrackPant', 'PoloJacket', 'PoloSweatshirt', 'PoloShoes') 11.11 Rule Selected
('PoloJacket',) => ('PoloSlippers', 'PoloShirt', 'TrackPant', 'PoloBackPack', 'PoloSweatshirt', 'PoloShoes') 20.0 Rule Selected
('PoloShirt',) => ('PoloSlippers', 'TrackPant', 'PoloBackPack', 'PoloJacket', 'PoloSweatshirt', 'PoloShoes') 33.33 Rule Selected
('PoloShoes',) => ('PoloSlippers', 'PoloShirt', 'TrackPant', 'PoloBackPack', 'PoloJacket', 'PoloSweatshirt') 10.0 Rule Selected
('PoloSlippers',) => ('PoloShirt', 'TrackPant', 'PoloBackPack', 'PoloJacket', 'PoloSweatshirt', 'PoloShoes') 25.0 Rule Selected
('PoloSweatshirt',) => ('PoloSlippers', 'PoloShirt', 'TrackPant', 'PoloBackPack', 'PoloJacket', 'PoloShoes') 10.0 Rule Selected
('TrackPant',) => ('PoloSlippers', 'PoloShirt', 'PoloBackPack', 'PoloJacket', 'PoloSweatshirt', 'PoloShoes') 20.0 Rule Selected

--- 0.8387963771820068 seconds ---
```

Brute Force Output for Database 4

No.of Iterations : 19
No.of Iterations : 171
No.of Iterations : 969
No.of Iterations : 3876
No.of Iterations : 11628
No.of Iterations : 27132
No.of Iterations : 50388
No.of Iterations : 75582
No.of Iterations : 92378
No.of Iterations : 92378
No.of Iterations : 75582
No.of Iterations : 50388
No.of Iterations : 27132
No.of Iterations : 11628
No.of Iterations : 3876
No.of Iterations : 969
No.of Iterations : 171
No.of Iterations : 19
No.of Iterations : 1

	Itemset	Support
1	Polo Backpack	13
3	Polo Cap	4
4	Polo Chair	4
5	Polo Door	3
6	Polo Jacket	5
...
4381	Polo Jacket,Polo Shirt,Polo Shoes,Polo TrackPant	2
4438	Polo Jacket,Polo Shoes,Polo Sweatshirt,Polo Tr...	2
4856	Polo Shoes,Polo Slippers,Polo Sweatshirt,Polo ...	2
9878	Polo Backpack,Polo Jacket,Polo Shoes,Polo Swea...	2
10296	Polo Backpack,Polo Shoes,Polo Slippers,Polo Sw...	2

[107 rows x 2 columns]

- 295.49355816841125 seconds -

Conclusion of Database 4

Apriori Algorithm is way much faster than Brute Force Method as the time generated in

Apriori Algortihm is --- --- 0.8387963771820068 sec

Brute Force Method is - - 295.49355816841125 sec

Apriori Algorithm for Database 5

Input

min_supp = 0.5

min_conf = 0.33

MRF Shoe	MRF Swea	MRF Track	MRF Backi	MRF Jacke	MRF Chair			
MRF Shoe	MRF Swea	MRF Backi	MRF Cap	MRF Ring				
MRF Shoe	MRF Track	MRF Backi	MRF Watc	MRF Slippers				
MRF Track	MRF Shoe	MRF Jacke	MRF Shirt					
MRF Swea	MRF Backi	MRF Watc	MRF Ring	MRF Door				
MRF Track	MRF Shoe	MRF Backi	MRF Table	MRF Chair				
MRF Backi	MRF Swea	MRF Shoe	MRF Track	MRF Chair				
MRF Shoe	MRF Swea	MRF Watc	MRF Ring					
MRF Track	MRF Swea	MRF Watc	MRF Jacke	MRF Shirt				
MRF Backi	MRF Shoe	MRF Swea	MRF Cap					
MRF Shoe	MRF Backi	MRF	MRF Table	MRF Chair				
MRF Track	MRF Shoe	MRF Swea	MRF Ring					
MRF Swea	MRF Backi	MRF Shoe	MRF Door					
MRF Backi	MRF Swea	MRF Track	MRF Jacke	MRF Table				
MRF Track	MRF Shoe	MRF Cap	MRF Watch					
MRF Shoe	MRF Swea	MRF Backi	MRF Track	MRF Cap				
MRF Swea	MRF Track	MRF Backi	MRF Door					
MRF Track	MRF Shoe	MRF Ring	MRF Table	MRF Slippers				
MRF Backi	MRF Swea	MRF Shoe	MRF Track	MRF Slippers				
MRF Track	MRF Swea	MRF Backi	MRF Shoe	MRF Jacke	MRF Shirt	MRF Slippers		

Output

```
( 'MRFSshirt', 'MRFSshoes' ) => ( 'MRFSslippers', 'MRFSsweatshirt', 'TrackPant', 'MRFBBackPack', 'MRFJacket' ) 50.0 Rule Selected
('MRFSshirt', 'MRFSslippers') => ('MRFSsweatshirt', 'TrackPant', 'MRFBBackPack', 'MRFSshoes', 'MRFJacket') 100.0 Rule Selected
('MRFSshirt', 'MRFSsweatshirt') => ('MRFSslippers', 'TrackPant', 'MRFSshoes', 'MRFBBackPack', 'MRFJacket') 50.0 Rule Selected
('MRFSshirt', 'TrackPant') => ('MRFSslippers', 'MRFSsweatshirt', 'MRFSshoes', 'MRFBBackPack', 'MRFJacket') 50.0 Rule Selected
('MRFSshoes', 'MRFSslippers') => ('MRFSshirt', 'MRFSsweatshirt', 'TrackPant', 'MRFBBackPack', 'MRFJacket') 33.33 Rule Selected
('MRFSshoes', 'MRFSsweatshirt') => ('MRFSslippers', 'MRFSshirt', 'TrackPant', 'MRFBBackPack', 'MRFJacket') 20.0 Rule Selected
('MRFSshoes', 'TrackPant') => ('MRFSslippers', 'MRFSshirt', 'MRFSsweatshirt', 'MRFBBackPack', 'MRFJacket') 20.0 Rule Selected
('MRFSslippers', 'MRFSsweatshirt') => ('MRFSshirt', 'TrackPant', 'MRFBBackPack', 'MRFSshoes', 'MRFJacket') 50.0 Rule Selected
('MRFSslippers', 'TrackPant') => ('MRFSshirt', 'MRFSsweatshirt', 'MRFBBackPack', 'MRFSshoes', 'MRFJacket') 50.0 Rule Selected
('MRFSsweatshirt', 'TrackPant') => ('MRFSslippers', 'MRFSshirt', 'MRFSshoes', 'MRFBBackPack', 'MRFJacket') 50.0 Rule Selected
('MRFBBackPack',) => ('MRFSslippers', 'MRFSshirt', 'MRFSsweatshirt', 'TrackPant', 'MRFSshoes', 'MRFJacket') 11.11 Rule Selected
('MRFJacket',) => ('MRFSslippers', 'MRFSshirt', 'MRFSsweatshirt', 'TrackPant', 'MRFBBackPack', 'MRFSshoes') 20.0 Rule Selected
('MRFSshirt',) => ('MRFSslippers', 'MRFSsweatshirt', 'TrackPant', 'MRFSshoes', 'MRFBBackPack', 'MRFJacket') 33.33 Rule Selected
('MRFSshoes',) => ('MRFSslippers', 'MRFSshirt', 'MRFSsweatshirt', 'TrackPant', 'MRFBBackPack', 'MRFJacket') 10.0 Rule Selected
('MRFSslippers',) => ('MRFSshirt', 'MRFSsweatshirt', 'TrackPant', 'MRFSshoes', 'MRFBBackPack', 'MRFJacket') 25.0 Rule Selected
('MRFSsweatshirt',) => ('MRFSslippers', 'MRFSshirt', 'TrackPant', 'MRFSshoes', 'MRFBBackPack', 'MRFJacket') 10.0 Rule Selected
('TrackPant',) => ('MRFSslippers', 'MRFSshirt', 'MRFSsweatshirt', 'MRFSshoes', 'MRFBBackPack', 'MRFJacket') 20.0 Rule Selected

--- 0.8437366485595703 seconds ---
```

Brute Force Output:

```
No.of Iterations : 19
No.of Iterations : 171
No.of Iterations : 969
No.of Iterations : 3876
No.of Iterations : 11628
No.of Iterations : 27132
No.of Iterations : 50388
No.of Iterations : 75582
No.of Iterations : 92378
No.of Iterations : 92378
No.of Iterations : 75582
No.of Iterations : 50388
No.of Iterations : 27132
No.of Iterations : 11628
No.of Iterations : 3876
No.of Iterations : 969
No.of Iterations : 171
No.of Iterations : 19
No.of Iterations : 1
```

	Itemset	Support
1	MRF BackPack	13
3	MRF Cap	4
4	MRF Chair	4
5	MRF Door	3
6	MRF Jacket	5
...
4381	MRF Jacket,MRF Shirt,MRF Shoes,MRF TrackPant	2
4438	MRF Jacket,MRF Shoes,MRF Sweatshirt,MRF TrackPant	2
4856	MRF Shoes,MRF Slippers,MRF Sweatshirt,MRF Trac...	2
9878	MRF BackPack,MRF Jacket,MRF Shoes,MRF Sweatshi...	2
10296	MRF BackPack,MRF Shoes,MRF Slippers,MRF Sweats...	2

[107 rows x 2 columns]

- 310.15418457984924 seconds -

Conclusion of Database 5

**Apriori Algorithm is way much faster than Brute Force Method
as the time generated in**

Apriori Algortihm is 0.8437366485595703 sec

Brute Force Method is 310.15418457984924 sec