Apriori Algorithm

Associations Rules는 간단하게 말해서 장바구니에 a,b,c라는 아이템이 있다면 d도 있을것이다 라는 가설을 의미합니다. 이때 사용하는 값에는 confidence와 support값이 있습니다.

Conficence값은 I={i1,i2,….,ik}가 있을 때, 어떠한 아이템 j도 존재할 확률 값을 뜻합니다. support값은 I={i1,i2,….,ik}가 등장한 장바구니 수를 뜻합니다. 따라서 conf(I->j) = support(I U {j}) / support(I) 라고 할 수 있습니다.

문제

저희는 support와 confidence가 각 각, 특정 값 s,c 이상인 association rule을 찾고싶습니다. 그렇게 하기 위해서는 자주 등장하는 아이템셋을 찾고 Rule을 생성하여야 합니다. 그렇기 때문에 자주 등장하는 아이템셋은 Apriori 알고리즘을 이용해서 찾고, 그 후 결과 값을 바탕으로 Association rule을 생성하려고 합니다.

Association_rule을 구하는 알고리즘

```
#Lis : 실제 구매 item 셋   fi : Apriori 알고리즘을 바탕으로 구한 아이템 set  confidence : 확률값
def association_rule(Lis, fi, confidence):

    #Apriori 알고리즘을 바탕으로 구한 아이템들의 부분집합 구하기
    fi=list(fi)
    result=[]
    for i in range(0, len(fi) + 1):
        c = combinations(fi, i)
        result.extend(c)
    fi=set(fi)

    #실제 구매 item 셋에 I={i1,i2,….,ik}가 있을 때, 어떠한 아이템 j 도 존재하는지 확인
    for i in range(1,len(result)-1):
        A = 0
        I = 0
        diff = fi.difference(result[i])
        for L in Lis:
            if L.issuperset(result[i]) == True:
                A+=1
                if L.issuperset(diff):
                    I+=1

        #confidence 값과 비교
        if A==0:
            A=1
        if I/A>confidence:
            print(result[i], "->", diff,": c=",I/A)
```

실행 결과.

1. *S=100 / confidence = 0.4로 하고 실행해본 결과,*

('butter',) -> {'whole milk'} : c= 0.4972477064220184

('hamburger meat',) -> {'whole milk'} : c= 0.4434250764525994

('hamburger meat',) -> {'other vegetables'} : c= 0.41590214067278286

('chicken',) -> {'whole milk'} : c= 0.4099526066350711

('onions',) -> {'other vegetables'} : c= 0.45901639344262296

('curd',) -> {'whole milk'} : c= 0.4904580152671756

('white bread',) -> {'whole milk'} : c= 0.4057971014492754

('tropical fruit',) -> {'whole milk'} : c= 0.40310077519379844

('root vegetables',) -> {'whole milk'} : c= 0.44869402985074625

('whipped/sour cream',) -> {'whole milk'} : c= 0.44964539007092197

('butter milk',) -> {'whole milk'} : c= 0.41454545454545455

('domestic eggs',) -> {'whole milk'} : c= 0.47275641025641024

('beef',) -> {'whole milk'} : c= 0.4050387596899225

('whipped/sour cream',) -> {'other vegetables'} : c= 0.40283687943262414

('chicken',) -> {'other vegetables'} : c= 0.41706161137440756

('frozen vegetables',) -> {'whole milk'} : c= 0.4249471458773784

('yogurt',) -> {'whole milk'} : c= 0.40160349854227406

('sugar',) -> {'whole milk'} : c= 0.4444444444444444

('margarine',) -> {'whole milk'} : c= 0.4131944444444444

('sliced cheese',) -> {'whole milk'} : c= 0.43983402489626555

('ham',) -> {'whole milk'} : c= 0.44140625

('root vegetables',) -> {'other vegetables'} : c= 0.43470149253731344

('cream cheese',) -> {'whole milk'} : c= 0.4153846153846154

('oil',) -> {'whole milk'} : c= 0.40217391304347827

('rolls/buns', 'tropical fruit') -> {'whole milk'} : c= 0.4462809917355372

('yogurt', 'tropical fruit') -> {'other vegetables'} : c= 0.4201388888888889

('whipped/sour cream', 'yogurt') -> {'other vegetables'} : c= 0.49019607843137253

('root vegetables', 'tropical fruit') -> {'whole milk'} : c= 0.5700483091787439

('other vegetables', 'pip fruit') -> {'whole milk'} : c= 0.5175097276264592

('pip fruit', 'whole milk') -> {'other vegetables'} : c= 0.44932432432432434

('yogurt', 'rolls/buns') -> {'whole milk'} : c= 0.4526627218934911

('other vegetables', 'pork') -> {'whole milk'} : c= 0.4694835680751174

('pork', 'whole milk') -> {'other vegetables'} : c= 0.45871559633027525

('other vegetables', 'whipped/sour cream') -> {'whole milk'} : c= 0.5070422535211268

('whipped/sour cream', 'whole milk') -> {'other vegetables'} : c= 0.45425867507886436

('other vegetables', 'fruit/vegetable juice') -> {'whole milk'} : c= 0.4975845410628019

('other vegetables', 'citrus fruit') -> {'whole milk'} : c= 0.4507042253521127

('citrus fruit', 'whole milk') -> {'other vegetables'} : c= 0.4266666666666667

('citrus fruit', 'root vegetables') -> {'other vegetables'} : c= 0.5862068965517241

('whipped/sour cream', 'yogurt') -> {'whole milk'} : c= 0.5245098039215687

('other vegetables', 'butter') -> {'whole milk'} : c= 0.5736040609137056

('butter', 'whole milk') -> {'other vegetables'} : c= 0.41697416974169743

('other vegetables', 'yogurt') -> {'whole milk'} : c= 0.5128805620608899

('other vegetables', 'root vegetables') -> {'whole milk'} : c= 0.4892703862660944

('root vegetables', 'whole milk') -> {'other vegetables'} : c= 0.47401247401247404

('other vegetables', 'domestic eggs') -> {'whole milk'} : c= 0.5525114155251142

('domestic eggs', 'whole milk') -> {'other vegetables'} : c= 0.4101694915254237

('other vegetables', 'pastry') -> {'whole milk'} : c= 0.46846846846846846

('yogurt', 'root vegetables') -> {'other vegetables'} : c= 0.5

('root vegetables', 'tropical fruit') -> {'other vegetables'} : c= 0.5845410628019324

('yogurt', 'root vegetables') -> {'whole milk'} : c= 0.562992125984252

('root vegetables', 'rolls/buns') -> {'whole milk'} : c= 0.5230125523012552

('citrus fruit', 'yogurt') -> {'whole milk'} : c= 0.47417840375586856

('other vegetables', 'tropical fruit') -> {'whole milk'} : c= 0.47592067988668557

('whole milk', 'tropical fruit') -> {'other vegetables'} : c= 0.40384615384615385

('other vegetables', 'bottled water') -> {'whole milk'} : c= 0.4344262295081967

('yogurt', 'tropical fruit') -> {'whole milk'} : c= 0.5173611111111112

('root vegetables', 'rolls/buns') -> {'other vegetables'} : c= 0.502092050209205

('other vegetables', 'rolls/buns') -> {'whole milk'} : c= 0.4200477326969974

('other vegetables', 'soda') -> {'whole milk'} : c= 0.4254658385093168

2. *S=100 / confidence = 0.5로 하고 실행해본 결과,*

('other vegetables', 'butter') -> {'whole milk'} : c= 0.5736040609137056

('whipped/sour cream', 'yogurt') -> {'whole milk'} : c= 0.5245098039215687

('root vegetables', 'tropical fruit') -> {'whole milk'} : c= 0.5700483091787439

('pip fruit', 'other vegetables') -> {'whole milk'} : c= 0.5175097276264592

('other vegetables', 'domestic eggs') -> {'whole milk'} : c= 0.5525114155251142

('tropical fruit', 'yogurt') -> {'whole milk'} : c= 0.5173611111111112

('citrus fruit', 'root vegetables') -> {'other vegetables'} : c= 0.5862068965517241

('root vegetables', 'tropical fruit') -> {'other vegetables'} : c= 0.5845410628019324

('root vegetables', 'yogurt') -> {'whole milk'} : c= 0.562992125984252

('other vegetables', 'whipped/sour cream') -> {'whole milk'} : c= 0.5070422535211268

('rolls/buns', 'root vegetables') -> {'whole milk'} : c= 0.5230125523012552

('other vegetables', 'yogurt') -> {'whole milk'} : c= 0.5128805620608899

('rolls/buns', 'root vegetables') -> {'other vegetables'} : c= 0.502092050209205

느낀점.

실행결과를 보면 알 수 있듯이, confidence값이 0.6보다 커지게 되면 결과값이 나오지 않게되는 것을 볼 수 있습니다. 따라서 아이템셋에 따라서 s값과 confidence값을 적당히 정해야 한다는 것을 알 수 있습니다. 또한 이를 바탕으로 얼마나 서로 어떤 확률로 아이템 셋들이 묶여있는지 알 수 있어서 굉장히 신기한 알고리즘이였던 것 같습니다.