Apriori

April 4, 2024

0.0.1 Association Rule Mining

[285]:
```
import numpy as  np
import pandas as pd
import pickle
import itertools
from collections import Counter
```

[286]:
```
class Apriori:
    def __init__(self,minimum_support,confidence_thresh):
        self.data=pd.read_csv(r'E:\Machine Learning Algorithms\Ex9 Association␣
 ↪Rule Mining\Market_Basket_Optimisation.csv')
        self.minimum_support=minimum_support
        self.confidence_thresh=confidence_thresh


    def _lazyread(self):
        dataset=self.data.fillna(0)
        self.transactions=[]
        for i in range(0,7500):
            self.transactions.append([str(dataset.values[i,j]) for j in␣
 ↪range(0, 20)])

    def _pkl_write_(self):
        with open(r'E:\Machine Learning Algorithms\Ex9 Association Rule␣
 ↪Mining\transactions.pkl','wb') as f:
            pickle.dump(self.transactions,f)
    def _pkl_read(self):
        with open(r'E:\Machine Learning Algorithms\Ex9 Association Rule␣
 ↪Mining\transactions.pkl','rb') as f:
            self.transactions=pickle.load(f)
        self.transactions=self.transactions[:50]
    def candidate_generation(self):
        d={}
        for i in range(len(self.transactions)):
            d.update({i:0})
        tmp=[]
        for i in ((self.transactions)):
            for j in ((i)):
                if(j!='0' and j!='0.0'):
                    tmp.append(j)
        d={}
        for i in tmp:
            d.update({i:0})
        for j in range(len(self.transactions)):
            _dict=dict(Counter(self.transactions[j]))
            print(_dict)
            try:
                _dict.pop('0')
                _dict.pop('0.0')
            except KeyError:
                continue
            for k in _dict:
                d[k]+=_dict.get(k)
        self.d=d
        self.data=pd.DataFrame(d.items(),columns=['items','frequency'])
        return self.data
    def cartesian_product(self,number):
        s=0
        d={}
        l=list(itertools.combinations(self.data['items'],number ))
        for i in range(len(l)):
            for j in range(len(self.transactions)):
                #tmp=list(itertools.product(list(z.transactions[j]),list(z.
↪transactions[j])))
                if (set(l[i]).intersection(set(self.
↪transactions[j])))==set(l[i]):
                    s+=1

            if (s>=1):
                d.update({l[i]:s})
            s=0
        self._candidate_combination=d
        return d
```

[ ]:

```
[287]: z=Apriori(3,50)
```

```
[288]: z._pkl_read()
```

```
[289]: z.transactions
```

```
[289]: [['burgers',
         'meatballs',
         'eggs',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['chutney',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['turkey',
         'avocado',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['mineral water',
         'milk',
         'energy bar',
         'whole wheat rice',
         'green tea',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['low fat yogurt',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['whole wheat pasta',
         'french fries',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['soup',
         'light cream',
         'shallot',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['frozen vegetables',
         'spaghetti',
         'green tea',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['french fries',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['eggs',
         'pet food',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['cookies',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['turkey',
         'burgers',
         'mineral water',
         'eggs',
         'cooking oil',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['spaghetti',
         'champagne',
         'cookies',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['mineral water',
         'salmon',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['mineral water',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['shrimp',
         'chocolate',
         'chicken',
         'honey',
         'oil',
         'cooking oil',
         'low fat yogurt',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['turkey',
         'eggs',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['turkey',
         'fresh tuna',
         'tomatoes',
         'spaghetti',
         'mineral water',
         'black tea',
         'salmon',
         'eggs',
         'chicken',
         'extra dark chocolate',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['meatballs',
         'milk',
         'honey',
         'french fries',
         'protein bar',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['red wine',
         'shrimp',
         'pasta',
         'pepper',
         'eggs',
         'chocolate',
         'shampoo',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['rice',
         'sparkling water',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['spaghetti',
         'mineral water',
         'ham',
         'body spray',
         'pancakes',
         'green tea',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['burgers',
         'grated cheese',
         'shrimp',
         'pasta',
         'avocado',
         'honey',
         'white wine',
         'toothpaste',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['eggs',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['parmesan cheese',
         'spaghetti',
         'soup',
         'avocado',
         'milk',
         'fresh bread',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['ground beef',
         'spaghetti',
         'mineral water',
         'milk',
         'energy bar',
         'black tea',
         'salmon',
         'frozen smoothie',
         'escalope',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['sparkling water',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['mineral water',
         'eggs',
         'chicken',
         'chocolate',
         'french fries',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['frozen vegetables',
         'spaghetti',
         'yams',
         'mineral water',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['herb & pepper',
         'tomato sauce',
         'light cream',
         'magazines',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['mineral water',
         'chocolate',
         'avocado',
         'eggs',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['turkey',
         'french fries',
         'strawberries',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['frozen vegetables',
         'strong cheese',
         'chocolate',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['cookies',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['pickles',
         'spaghetti',
         'salmon',
         'escalope',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0',
         '0.0'],
        ['energy bar',
         'french fries',
         '0',
         '0',
         '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0',
        '0.0'],
['red wine',
 'ground beef',
 'mineral water',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['mineral water',
 'cake',
 'cottage cheese',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['pickles',
 'champagne',
 'green tea',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['spaghetti',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['fresh tuna',
 'frozen vegetables',
 'spaghetti',
 'mineral water',
 'honey',
 'whole wheat rice',
 'frozen smoothie',
 'escalope',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['spaghetti',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['soup',
 'meatballs',
 'hot dogs',
 'sparkling water',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['escalope',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['soup',
 'avocado',
 'french fries',
 'hot dogs',
 'brownies',
 'body spray',
 'pancakes',
 'green tea',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['mineral water',
 'chicken',
 'cereals',
 'clothes accessories',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['mineral water',
 'bug spray',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['avocado',
 'muffins',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['burgers',
 'black tea',
 'green tea',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0',
 '0.0'],
['spaghetti',
 'chocolate',
 'brownies',
 'white wine',
 'green tea',
 '0',
 '0',
  '0',
  '0',
  '0',
  '0',
  '0',
  '0',
  '0',
  '0',
  '0',
  '0',
  '0',
  '0',
  '0.0']]
```

[290]:
```
df=z.candidate_generation()
```

```
{'burgers': 1, 'meatballs': 1, 'eggs': 1, '0': 16, '0.0': 1}
{'chutney': 1, '0': 18, '0.0': 1}
{'turkey': 1, 'avocado': 1, '0': 17, '0.0': 1}
{'mineral water': 1, 'milk': 1, 'energy bar': 1, 'whole wheat rice': 1, 'green
tea': 1, '0': 14, '0.0': 1}
{'low fat yogurt': 1, '0': 18, '0.0': 1}
{'whole wheat pasta': 1, 'french fries': 1, '0': 17, '0.0': 1}
{'soup': 1, 'light cream': 1, 'shallot': 1, '0': 16, '0.0': 1}
{'frozen vegetables': 1, 'spaghetti': 1, 'green tea': 1, '0': 16, '0.0': 1}
{'french fries': 1, '0': 18, '0.0': 1}
{'eggs': 1, 'pet food': 1, '0': 17, '0.0': 1}
{'cookies': 1, '0': 18, '0.0': 1}
{'turkey': 1, 'burgers': 1, 'mineral water': 1, 'eggs': 1, 'cooking oil': 1,
'0': 14, '0.0': 1}
{'spaghetti': 1, 'champagne': 1, 'cookies': 1, '0': 16, '0.0': 1}
{'mineral water': 1, 'salmon': 1, '0': 17, '0.0': 1}
{'mineral water': 1, '0': 18, '0.0': 1}
{'shrimp': 1, 'chocolate': 1, 'chicken': 1, 'honey': 1, 'oil': 1, 'cooking oil':
1, 'low fat yogurt': 1, '0': 12, '0.0': 1}
{'turkey': 1, 'eggs': 1, '0': 17, '0.0': 1}
{'turkey': 1, 'fresh tuna': 1, 'tomatoes': 1, 'spaghetti': 1, 'mineral water':
1, 'black tea': 1, 'salmon': 1, 'eggs': 1, 'chicken': 1, 'extra dark chocolate':
1, '0': 9, '0.0': 1}
{'meatballs': 1, 'milk': 1, 'honey': 1, 'french fries': 1, 'protein bar': 1,
'0': 14, '0.0': 1}
{'red wine': 1, 'shrimp': 1, 'pasta': 1, 'pepper': 1, 'eggs': 1, 'chocolate': 1,
'shampoo': 1, '0': 12, '0.0': 1}
{'rice': 1, 'sparkling water': 1, '0': 17, '0.0': 1}
{'spaghetti': 1, 'mineral water': 1, 'ham': 1, 'body spray': 1, 'pancakes': 1,
'green tea': 1, '0': 13, '0.0': 1}
{'burgers': 1, 'grated cheese': 1, 'shrimp': 1, 'pasta': 1, 'avocado': 1,
'honey': 1, 'white wine': 1, 'toothpaste': 1, '0': 11, '0.0': 1}
```

```
{'eggs': 1, '0': 18, '0.0': 1}
{'parmesan cheese': 1, 'spaghetti': 1, 'soup': 1, 'avocado': 1, 'milk': 1,
'fresh bread': 1, '0': 13, '0.0': 1}
{'ground beef': 1, 'spaghetti': 1, 'mineral water': 1, 'milk': 1, 'energy bar':
1, 'black tea': 1, 'salmon': 1, 'frozen smoothie': 1, 'escalope': 1, '0': 10,
'0.0': 1}
{'sparkling water': 1, '0': 18, '0.0': 1}
{'mineral water': 1, 'eggs': 1, 'chicken': 1, 'chocolate': 1, 'french fries': 1,
'0': 14, '0.0': 1}
{'frozen vegetables': 1, 'spaghetti': 1, 'yams': 1, 'mineral water': 1, '0': 15,
'0.0': 1}
{'herb & pepper': 1, 'tomato sauce': 1, 'light cream': 1, 'magazines': 1, '0':
15, '0.0': 1}
{'mineral water': 1, 'chocolate': 1, 'avocado': 1, 'eggs': 1, '0': 15, '0.0': 1}
{'turkey': 1, 'french fries': 1, 'strawberries': 1, '0': 16, '0.0': 1}
{'frozen vegetables': 1, 'strong cheese': 1, 'chocolate': 1, '0': 16, '0.0': 1}
{'cookies': 1, '0': 18, '0.0': 1}
{'pickles': 1, 'spaghetti': 1, 'salmon': 1, 'escalope': 1, '0': 15, '0.0': 1}
{'energy bar': 1, 'french fries': 1, '0': 17, '0.0': 1}
{'red wine': 1, 'ground beef': 1, 'mineral water': 1, '0': 16, '0.0': 1}
{'mineral water': 1, 'cake': 1, 'cottage cheese': 1, '0': 16, '0.0': 1}
{'pickles': 1, 'champagne': 1, 'green tea': 1, '0': 16, '0.0': 1}
{'spaghetti': 1, '0': 18, '0.0': 1}
{'fresh tuna': 1, 'frozen vegetables': 1, 'spaghetti': 1, 'mineral water': 1,
'honey': 1, 'whole wheat rice': 1, 'frozen smoothie': 1, 'escalope': 1, '0': 11,
'0.0': 1}
{'spaghetti': 1, '0': 18, '0.0': 1}
{'soup': 1, 'meatballs': 1, 'hot dogs': 1, 'sparkling water': 1, '0': 15, '0.0':
1}
{'escalope': 1, '0': 18, '0.0': 1}
{'soup': 1, 'avocado': 1, 'french fries': 1, 'hot dogs': 1, 'brownies': 1, 'body
spray': 1, 'pancakes': 1, 'green tea': 1, '0': 11, '0.0': 1}
{'mineral water': 1, 'chicken': 1, 'cereals': 1, 'clothes accessories': 1, '0':
15, '0.0': 1}
{'mineral water': 1, 'bug spray': 1, '0': 17, '0.0': 1}
{'avocado': 1, 'muffins': 1, '0': 17, '0.0': 1}
{'burgers': 1, 'black tea': 1, 'green tea': 1, '0': 16, '0.0': 1}
{'spaghetti': 1, 'chocolate': 1, 'brownies': 1, 'white wine': 1, 'green tea': 1,
'0': 14, '0.0': 1}
```

[241]:
```
df
```

[241]:
```
                      items  frequency
0                   burgers          4
1                 meatballs          3
2                      eggs          9
3                   chutney          1
```

```
4                   turkey          5
..                     …         …
61                brownies          2
62                 cereals          1
63   clothes accessories          1
64               bug spray          1
65                 muffins          1

[66 rows x 2 columns]
```

[291]:
```
df_1=df[df['frequency']>=2] #min support 2
```

[243]:
```
df_1
```

[243]:
```
                 items  frequency
0              burgers          4
1            meatballs          3
2                 eggs          9
4               turkey          5
5              avocado          6
6        mineral water         15
7                 milk          4
8           energy bar          3
9     whole wheat rice          2
10           green tea          7
11       low fat yogurt          2
13        french fries          7
14                soup          4
15         light cream          2
17   frozen vegetables          4
18           spaghetti         12
20             cookies          3
21         cooking oil          2
22           champagne          2
23              salmon          4
24              shrimp          3
25           chocolate          6
26             chicken          4
27               honey          4
29          fresh tuna          2
31           black tea          3
34            red wine          2
35               pasta          2
39     sparkling water          3
41          body spray          2
42            pancakes          2
44          white wine          2
```

```
48        ground beef          2
49    frozen smoothie          2
50           escalope          4
57            pickles          2
60           hot dogs          2
61           brownies          2
```

```
[292]: z.data=df_1
```

```
[293]: k=z.cartesian_product(2)
```

```
[294]: df_2=pd.DataFrame(k.items(),columns=['items','freq'])
```

```
[295]: df_2=df_2[df_2['freq']>=2]
```

```
[299]: z.data=df_2
```

```
[296]: df_1[df_1['items']=='eggs']
```

```
[296]:   items  frequency
       2  eggs          9
```

```
[297]: df_1[df_1['items']=='burgers']['frequency'].values[0] #denominator
```

```
[297]: 4
```

```
[266]: df_1
```

```
[266]:                  items  frequency
       0               burgers          4
       1             meatballs          3
       2                  eggs          9
       4                turkey          5
       5               avocado          6
       6         mineral water         15
       7                  milk          4
       8            energy bar          3
       9      whole wheat rice          2
       10            green tea          7
       11        low fat yogurt          2
       13         french fries          7
       14                 soup          4
       15          light cream          2
       17    frozen vegetables          4
       18            spaghetti         12
       20              cookies          3
       21          cooking oil          2
       22            champagne          2
```

```
23           salmon      4
24           shrimp      3
25        chocolate      6
26          chicken      4
27            honey      4
29       fresh tuna      2
31        black tea      3
34         red wine      2
35            pasta      2
39  sparkling water      3
41       body spray      2
42         pancakes      2
44       white wine      2
48      ground beef      2
49  frozen smoothie      2
50         escalope      4
57          pickles      2
60         hot dogs      2
61         brownies      2
```

[301]:
```
s1=z.data['items'].reset_index(drop=True)
```

[300]:
```
z.data
```

[300]:
```
                                  items  freq
1                       (burgers, eggs)     2
19                       (eggs, turkey)     3
21                (eggs, mineral water)     4
27                    (eggs, chocolate)     3
28                      (eggs, chicken)     2
34              (turkey, mineral water)     2
46                      (avocado, soup)     2
57                (mineral water, milk)     2
58          (mineral water, energy bar)     2
59    (mineral water, whole wheat rice)     2
60           (mineral water, green tea)     2
62  (mineral water, frozen vegetables)     2
63           (mineral water, spaghetti)     5
65              (mineral water, salmon)     3
66           (mineral water, chocolate)     2
67             (mineral water, chicken)     3
69          (mineral water, fresh tuna)     2
70           (mineral water, black tea)     2
74         (mineral water, ground beef)     2
75     (mineral water, frozen smoothie)     2
76            (mineral water, escalope)     2
77                   (milk, energy bar)     2
```

```
82                        (milk, spaghetti)     2
108                  (green tea, spaghetti)     3
112                 (green tea, body spray)     2
113                   (green tea, pancakes)     2
117                   (green tea, brownies)     2
136                        (soup, hot dogs)     2
138       (frozen vegetables, spaghetti)     3
146                    (spaghetti, salmon)     3
150                (spaghetti, fresh tuna)     2
151                 (spaghetti, black tea)     2
156          (spaghetti, frozen smoothie)     2
157                  (spaghetti, escalope)     3
168                    (salmon, black tea)     2
171                     (salmon, escalope)     2
173                   (shrimp, chocolate)     2
175                       (shrimp, honey)     2
177                       (shrimp, pasta)     2
179                  (chocolate, chicken)     2
203                 (body spray, pancakes)     2
211            (frozen smoothie, escalope)     2
```

```
[307]: s1=pd.DataFrame(s1,columns=['items'])
```

```
[346]: df_2=pd.DataFrame(df_2.reset_index(drop=True),columns=['items','freq'])
```

```
[318]:
```

```
---------------------------------------------------------------------------
KeyError                                  Traceback (most recent call last)
~\AppData\Local\Temp\ipykernel_37620\2579744479.py in <cell line: 1>()
----> 1 df_1[df_1['items']]

e:\anaconda\lib\site-packages\pandas\core\frame.py in __getitem__(self, key)
   3811             if is_iterator(key):
   3812                 key = list(key)
-> 3813             indexer = self.columns._get_indexer_strict(key, "columns")[
   3814 
   3815         # take() does not accept boolean indexers

e:\anaconda\lib\site-packages\pandas\core\indexes\base.py in␣
 ↪_get_indexer_strict(self, key, axis_name)
   6068             keyarr, indexer, new_indexer = self.
 ↪_reindex_non_unique(keyarr)
   6069 
-> 6070         self._raise_if_missing(keyarr, indexer, axis_name)
   6071 
   6072         keyarr = self.take(indexer)
```

```
e:\anaconda\lib\site-packages\pandas\core\indexes\base.py in␣
↪_raise_if_missing(self, key, indexer, axis_name)
   6128                 if use_interval_msg:
   6129                     key = list(key)
-> 6130                 raise KeyError(f"None of [{key}] are in the␣
↪[{axis_name}]")
   6131
   6132             not_found = list(ensure_index(key)[missing_mask.
↪nonzero()[0]].unique())

KeyError: "None of [Index(['burgers', 'meatballs', 'eggs', 'turkey', 'avocado',␣
↪'mineral water',\n       'milk', 'energy bar', 'whole wheat rice', 'green␣
↪tea', 'low fat yogurt',\n       'french fries', 'soup', 'light cream', 'froze␣
↪vegetables', 'spaghetti',\n       'cookies', 'cooking oil', 'champagne',␣
↪'salmon', 'shrimp', 'chocolate',\n       'chicken', 'honey', 'fresh tuna',␣
↪'black tea', 'red wine', 'pasta',\n       'sparkling water', 'body spray',␣
↪'pancakes', 'white wine',\n       'ground beef', 'frozen smoothie',␣
↪'escalope', 'pickles', 'hot dogs',\n       'brownies'],\n      ␣
↪dtype='object')] are in the [columns]"
```

[319]:
```
df_1['items'][0]
```

[319]:
```
'burgers'
```

[343]:
```
df_2=df_2.reindex(drop=True)
```

[343]:
```
                                  items  freq
1                       (burgers, eggs)     2
19                       (eggs, turkey)     3
21                (eggs, mineral water)     4
27                    (eggs, chocolate)     3
28                      (eggs, chicken)     2
34              (turkey, mineral water)     2
46                      (avocado, soup)     2
57                (mineral water, milk)     2
58          (mineral water, energy bar)     2
59    (mineral water, whole wheat rice)     2
60           (mineral water, green tea)     2
62   (mineral water, frozen vegetables)     2
63           (mineral water, spaghetti)     5
65              (mineral water, salmon)     3
66           (mineral water, chocolate)     2
67             (mineral water, chicken)     3
69          (mineral water, fresh tuna)     2
70           (mineral water, black tea)     2
74         (mineral water, ground beef)     2
75     (mineral water, frozen smoothie)     2
```

```
76            (mineral water, escalope)     2
77                     (milk, energy bar)     2
82                      (milk, spaghetti)     2
108                 (green tea, spaghetti)     3
112                (green tea, body spray)     2
113                  (green tea, pancakes)     2
117                  (green tea, brownies)     2
136                       (soup, hot dogs)     2
138       (frozen vegetables, spaghetti)     3
146                   (spaghetti, salmon)     3
150              (spaghetti, fresh tuna)     2
151               (spaghetti, black tea)     2
156         (spaghetti, frozen smoothie)     2
157                 (spaghetti, escalope)     3
168                   (salmon, black tea)     2
171                    (salmon, escalope)     2
173                   (shrimp, chocolate)     2
175                       (shrimp, honey)     2
177                       (shrimp, pasta)     2
179                  (chocolate, chicken)     2
203               (body spray, pancakes)     2
211          (frozen smoothie, escalope)     2
```

[349]:
```python
df_1=pd.DataFrame(df_1.reset_index(drop=True),columns=['items','frequency'])
```

[375]:
```python
df=pd.DataFrame()
l=[]
for i in range(1,len(df_1)):
    t=s1['items'][i]
    #print(t[0])
    #for j in t:
    #print(df_1[df_1['items']=='burgers'])
    #print(t[0],df_1['items'][0])
    for j in range(len(df_1)):
        if(df_1['items'][j]==t[0]):
            #df.concat(df_2['freq'][j]/df_1['frequency'][i])
            #print('a')
            l.append(df_2['freq'][j]/df_1['frequency'][i])


    #print(df_1[df_1['items']][0])
    #df.append()
```

[376]:
```python
z=pd.DataFrame(l,columns=['Confidence'])
```

[379]:
```python
df_2=df_2.append(z)
```

```
C:\Users\Admin\AppData\Local\Temp\ipykernel_37620\4253449237.py:1:
```

```
FutureWarning: The frame.append method is deprecated and will be removed from
pandas in a future version. Use pandas.concat instead.
  df_2=df_2.append(z)
```

```
[382]: df_2
```

```
[382]:                    items  freq  Confidence
       0         (burgers, eggs)   2.0         NaN
       1          (eggs, turkey)   3.0         NaN
       2   (eggs, mineral water)   4.0         NaN
       3       (eggs, chocolate)   3.0         NaN
       4         (eggs, chicken)   2.0         NaN
       ..                    ...   ...         ...
       32                    NaN   NaN         1.5
       33                    NaN   NaN         0.5
       34                    NaN   NaN         1.0
       35                    NaN   NaN         1.0
       36                    NaN   NaN         1.0

       [79 rows x 3 columns]
```

```
[373]: z=z[z['Confidence']<1]
```

```
[374]: z
```

```
[374]:     Confidence
       1     0.444444
       2     0.800000
       3     0.666667
       4     0.200000
       5     0.500000
       6     0.666667
       8     0.285714
       10    0.285714
       11    0.500000
       13    0.500000
       14    0.166667
       15    0.666667
       18    0.500000
       19    0.666667
       20    0.333333
       21    0.500000
       22    0.500000
       24    0.666667
       27    0.666667
       33    0.500000
```

```
[355]: s1
```

[355]:
```
                                      items
0                           (burgers, eggs)
1                            (eggs, turkey)
2                     (eggs, mineral water)
3                         (eggs, chocolate)
4                           (eggs, chicken)
5                   (turkey, mineral water)
6                           (avocado, soup)
7                     (mineral water, milk)
8               (mineral water, energy bar)
9        (mineral water, whole wheat rice)
10               (mineral water, green tea)
11      (mineral water, frozen vegetables)
12               (mineral water, spaghetti)
13                  (mineral water, salmon)
14               (mineral water, chocolate)
15                 (mineral water, chicken)
16              (mineral water, fresh tuna)
17               (mineral water, black tea)
18             (mineral water, ground beef)
19         (mineral water, frozen smoothie)
20                (mineral water, escalope)
21                       (milk, energy bar)
22                        (milk, spaghetti)
23                   (green tea, spaghetti)
24                  (green tea, body spray)
25                    (green tea, pancakes)
26                    (green tea, brownies)
27                         (soup, hot dogs)
28          (frozen vegetables, spaghetti)
29                      (spaghetti, salmon)
30                  (spaghetti, fresh tuna)
31                   (spaghetti, black tea)
32             (spaghetti, frozen smoothie)
33                    (spaghetti, escalope)
34                      (salmon, black tea)
35                       (salmon, escalope)
36                      (shrimp, chocolate)
37                          (shrimp, honey)
38                          (shrimp, pasta)
39                     (chocolate, chicken)
40                   (body spray, pancakes)
41              (frozen smoothie, escalope)
```

[352]:
```
df_2
```

[352]:
```
                                  items  freq
0                     (burgers, eggs)     2
1                      (eggs, turkey)     3
2               (eggs, mineral water)     4
3                   (eggs, chocolate)     3
4                     (eggs, chicken)     2
5             (turkey, mineral water)     2
6                     (avocado, soup)     2
7               (mineral water, milk)     2
8         (mineral water, energy bar)     2
9   (mineral water, whole wheat rice)     2
10         (mineral water, green tea)     2
11 (mineral water, frozen vegetables)     2
12         (mineral water, spaghetti)     5
13            (mineral water, salmon)     3
14         (mineral water, chocolate)     2
15           (mineral water, chicken)     3
16        (mineral water, fresh tuna)     2
17         (mineral water, black tea)     2
18       (mineral water, ground beef)     2
19    (mineral water, frozen smoothie)     2
20          (mineral water, escalope)     2
21                 (milk, energy bar)     2
22                  (milk, spaghetti)     2
23             (green tea, spaghetti)     3
24            (green tea, body spray)     2
25              (green tea, pancakes)     2
26              (green tea, brownies)     2
27                   (soup, hot dogs)     2
28      (frozen vegetables, spaghetti)     3
29                (spaghetti, salmon)     3
30            (spaghetti, fresh tuna)     2
31             (spaghetti, black tea)     2
32       (spaghetti, frozen smoothie)     2
33              (spaghetti, escalope)     3
34                (salmon, black tea)     2
35                 (salmon, escalope)     2
36                (shrimp, chocolate)     2
37                    (shrimp, honey)     2
38                    (shrimp, pasta)     2
39               (chocolate, chicken)     2
40              (body spray, pancakes)     2
41        (frozen smoothie, escalope)     2
```

[251]:
```
k2=z.cartesian_product(3)
```

[252]:
```
k2
```

```
[252]: {}
```

```
[253]: g=pd.DataFrame(k.items(),columns=['items','freq'])
```

```
[254]: g[g['freq']>=2]
```

```
[254]:                                    items  freq
       1                      (burgers, eggs)     2
       19                      (eggs, turkey)     3
       21               (eggs, mineral water)     4
       27                   (eggs, chocolate)     3
       28                     (eggs, chicken)     2
       34             (turkey, mineral water)     2
       46                     (avocado, soup)     2
       57               (mineral water, milk)     2
       58         (mineral water, energy bar)     2
       59   (mineral water, whole wheat rice)     2
       60          (mineral water, green tea)     2
       62  (mineral water, frozen vegetables)     2
       63          (mineral water, spaghetti)     5
       65             (mineral water, salmon)     3
       66          (mineral water, chocolate)     2
       67            (mineral water, chicken)     3
       69         (mineral water, fresh tuna)     2
       70          (mineral water, black tea)     2
       74        (mineral water, ground beef)     2
       75    (mineral water, frozen smoothie)     2
       76           (mineral water, escalope)     2
       77                  (milk, energy bar)     2
       82                   (milk, spaghetti)     2
       108             (green tea, spaghetti)     3
       112            (green tea, body spray)     2
       113              (green tea, pancakes)     2
       117              (green tea, brownies)     2
       136                  (soup, hot dogs)      2
       138     (frozen vegetables, spaghetti)     3
       146                (spaghetti, salmon)     3
       150            (spaghetti, fresh tuna)     2
       151             (spaghetti, black tea)     2
       156       (spaghetti, frozen smoothie)     2
       157              (spaghetti, escalope)     3
       168                (salmon, black tea)     2
       171                 (salmon, escalope)     2
       173                (shrimp, chocolate)     2
       175                    (shrimp, honey)     2
       177                    (shrimp, pasta)     2
       179               (chocolate, chicken)     2
```

```
203               (body spray, pancakes)     2
211         (frozen smoothie, escalope)     2
```