

```
import csv
from mlxtend.preprocessing import TransactionEncoder
from mlxtend.frequent_patterns import apriori, association_rules
```

```
dataset = []
with open('Market_Basket_Optimisation.csv') as file:
    reader = csv.reader(file, delimiter=',')
    for row in reader:
        dataset += [row]
```

```
dataset
```

```
[ 'french fries', 'cookies'],
['milk', 'butter', 'eggs'],
['eggs', 'mushroom cream sauce', 'low fat yogurt'],
['eggs', 'green tea'],
['mineral water', 'whole wheat rice'],
['shrimp',
 'frozen vegetables',
 'parmesan cheese',
 'mineral water',
 'whole wheat rice',
 'cake'],
['whole wheat rice', 'pancakes'],
['frozen vegetables',
 'spaghetti',
 'olive oil',
 'butter',
 'salmon',
 'oil',
 'cooking oil',
 'frozen smoothie',
 'cauliflower'],
['green tea'],
['fresh tuna', 'eggs', 'escalope', 'strawberries'],
['spaghetti', 'soup', 'milk', 'carrots', 'chocolate'],
['burgers', 'mineral water', 'soup', 'meatballs', 'olive oil'],
['grated cheese',
 'mineral water',
 'cooking oil',
 'french fries',
 'cookies',
 'low fat yogurt'],
['shrimp', 'milk', 'olive oil', 'french fries'],
['frozen vegetables', 'mineral water', 'pancakes', 'cake', 'tomato juice'],
['spaghetti', 'milk', 'chicken'],
['grated cheese',
 'mineral water',
 'chicken',
 'french fries',
 'cottage cheese',
 'pancakes'],
['ground beef', 'mineral water', 'milk', 'eggs', 'mint'],
['shrimp', 'body spray', 'green tea'],
['frozen smoothie'],
['herb & pepper', 'frozen vegetables', 'mineral water', 'muffins', 'cereals'],
['turkey',
 'tomatoes',
 'spaghetti',
 'milk',
 'cider',
 'eggs',
 'honey',
 'cake',
 'green tea',
 'french fries',
 'brownies',
 'tomato juice'],
...]
```

```
len(dataset)
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform`
and should_run_async(code)
7501
```

```
te = TransactionEncoder()
x = te.fit_transform(dataset)
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform`
and should_run_async(code)
```

x

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
and should_run_async(code)
array([[False,  True,  True, ...,  True, False, False],
       [False, False, False, ..., False, False, False],
       [False, False, False, ..., False, False, False],
       ...,
       [False, False, False, ..., False, False, False],
       [False, False, False, ..., False, False, False],
       [False, False, False, ..., False,  True, False]])
```

```
df = pd.DataFrame(x, columns=te.columns_)

/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
and should_run_async(code)
```

```
len(te.columns_)

/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
and should_run_async(code)
120
```

```
df.head()

/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
and should_run_async(code)
```

	asparagus	almonds	antioxydant juice	asparagus	avocado	babies food	bacon	barbecue sauce	black tea	blueberries	...	turkey	vegetables mix	water spray
0	False	True	True	False	True	False	False	False	False	False	...	False	True	False
1	False	False	False	False	False	False	False	False	False	False	...	False	False	False
2	False	False	False	False	False	False	False	False	False	False	...	False	False	False
3	False	False	False	False	True	False	False	False	False	False	...	True	False	False
4	False	False	False	False	False	False	False	False	False	False	...	False	False	False

5 rows × 120 columns

```
df.shape

/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
and should_run_async(code)
(7501, 120)
```

```
# 1. Frequent itemsets
freq_itemset = apriori(df, min_support=0.01, use_colnames=True)

/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
and should_run_async(code)
```

freq_itemset

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
and should_run_async(code)
```

	support	itemsets	
0	0.020397	(almonds)	
1	0.033329	(avocado)	
2	0.010799	(barbecue sauce)	
3	0.014265	(black tea)	

```
# Find the rules
```

```
rules = association_rules(freq_itemset, metric='confidence', min_threshold=0.25,)
```

```
rules = rules[['antecedents', 'consequents', 'support', 'confidence']]
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning: `should_run_async` will not call `transform_c
and should_run_async(code)
```

```
rules.head(10)
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning
and should_run_async(code)
```

	antecedents	consequents	support	confidence	
0	(avocado)	(mineral water)	0.011598	0.348000	
1	(burgers)	(eggs)	0.028796	0.330275	
2	(burgers)	(french fries)	0.021997	0.252294	
3	(burgers)	(mineral water)	0.024397	0.279817	
4	(cake)	(mineral water)	0.027463	0.338816	
5	(cereals)	(mineral water)	0.010265	0.398964	
6	(chicken)	(mineral water)	0.022797	0.380000	
7	(chicken)	(spaghetti)	0.017198	0.286667	
8	(cooking oil)	(chocolate)	0.013598	0.266319	
9	(chocolate)	(mineral water)	0.052660	0.321400	

```
rules[rules['antecedents'] == {'cake'}]
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning
and should_run_async(code)
```

	antecedents	consequents	support	confidence	
4	(cake)	(mineral water)	0.027463	0.338816	

```
rules[rules['antecedents'] == {'milk'}]
```

```
/usr/local/lib/python3.10/dist-packages/ipykernel/ipkernel.py:283: DeprecationWarning
and should_run_async(code)
```

	antecedents	consequents	support	confidence	
31	(milk)	(mineral water)	0.047994	0.370370	
34	(milk)	(spaghetti)	0.035462	0.273663	