

```
import pandas as pd
from apyori import apriori
import [URL_14]plot as plt
import seaborn as sns
```

```
df = [URL_14]ad_csv("Market_Basket_Optimisation.csv", header=None)
df.head()
```

0

1

2

3

4

5

6

7

8

0

shrimp

almonds avocado

vegetables  
mix

green  
grapes

whole  
weat  
■our

yams

cottage  
cheese

energy  
drink

[PERSON\_3] burgers meatballs

eggs

NaN

NaN

NaN NaN

2 chutney

NaN

turkey

avocado

NaN

NaN

NaN

NaN

NaN NaN

NaN

NaN

NaN NaN

3

4

mineral  
water

milk

energy  
bar

whole  
wheat rice

green  
tea

NaN NaN

NaN

NaN

NaN

NaN

NaN

NaN

NaN

NaN

N

N

N

N

```
transactions = []
for i in range(0, [URL_14]ape[0]):
    transaction = [str([URL_14]lues[i, j]) for j in range(0, 20) if str(df.v
    [URL_14]end(transaction)
print("Sample transaction: ", transaction[0])
print("Total transactions: ", len(transactions))
```

Sample transaction: eggs  
Total transactions: 7501

```
rules = apriori(transactions, min_support=0.003, min_confidence=0.2, min_
results = list(rules)
print("Total rules generated: ", len(results))
```

Total rules generated: 80

```
def inspect(results):
    lhs = [tuple(result[2][0][0])[0] for result in results]
    rhs = [tuple(result[2][0][1])[0] for result in results]
    support = [result[1] for result in results]
    confidence = [result[2][0][2] for result in results]
    lifts = [result[2][0][3] for result in results]
    return [LOCATION_1, rhs, support, confidence, lifts])
```

```
rules_df = pd.DataFrame(inspect(results), columns=['Left Hand Side', 'Rig
```

```
rules_df.head()
```

1/3

```
In [9]:In [10]:Out[10]:In [15]:In [16]:In [19]:  
■Left Hand Side Right Hand Side
```

Support Con■dence

Lift

0

light cream

chicken 0.[US\_DRIVER\_LICENSE\_52]

[PHONE\_NUMBER\_22]

1 mushroom cream sauce

escalope 0.[US\_DRIVER\_LICENSE\_52]

[PHONE\_NUMBER\_22]

2

3

4

pasta

escalope 0.[US\_DRIVER\_LICENSE\_52]

[PHONE\_NUMBER\_22]

fromage blanc

honey 0.[US\_DRIVER\_LICENSE\_52]

[PHONE\_NUMBER\_22]

herb & pepper

ground beef 0.[US\_DRIVER\_LICENSE\_52]

[PHONE\_NUMBER\_22]

[URL\_14]gure(figsize=(10,6))

[URL\_14]atterplot(x='Support', y='Confidence', size='Lift', data=rules\_df)

```
plt.title("Association Rules: Support vs Confidence")
plt.xlabel("Support")
plt.ylabel("Confidence")
[URL_14]ow()

rules_high_conf = apriori(transactions,
                           min_support=0.003,
                           min_confidence=0.5, # Increased from 0.2 → 0.5
                           min_lift=3,
                           min_length=2)

results_high_conf = [PERSON_3])
rules_high_conf_df = pd.DataFrame(inspect(results_high_conf),
                                   columns=['Left Hand Side', 'Right Hand

print("Rules generated after increasing confidence:", [PERSON_3]
rules_high_conf_df.head())
```

Rules generated after increasing confidence: 14

2/3

Out[19]:In [20]:In [22]:■Left Hand Side Right Hand Side

Support Con■dence

Lift

0

1

2

3

4

cereals

chicken

spaghetti 0.[US\_DRIVER\_LICENSE\_52]

[PHONE\_NUMBER\_22]

milk 0.[US\_DRIVER\_LICENSE\_52]

0.[US\_DRIVER\_LICENSE\_52] 3.[US\_DRIVER\_LICENSE\_52]

cooking oil

spaghetti 0.[US\_DRIVER\_LICENSE\_52]

[PHONE\_NUMBER\_22]

red wine

spaghetti 0.[US\_DRIVER\_LICENSE\_52]

[PHONE\_NUMBER\_22]

soup

milk 0.[US\_DRIVER\_LICENSE\_52]

0.[US\_DRIVER\_LICENSE\_52] 3.[US\_DRIVER\_LICENSE\_52]

[URL\_14]gure(figsize=(10,6))

[URL\_14]atterplot(x='Support', y='Confidence', size='Lift', data=rules\_high

plt.title("Association Rules (Higher Confidence): Support vs Confidence")

plt.xlabel("Support")

plt.ylabel("Confidence")

[URL\_14]ow()

3/3

Out[22]:In [23]:In [ ]:In [ ]:In [ ]:

