How Does the Apriori Algorithm Work?

The Apriori algorithm operates on a straightforward premise. When the support value of an item set exceeds a certain threshold, it is considered a frequent item set. Take into account the following steps. To begin, set the support criterion, meaning that only those things that have more than the support criterion are considered relevant.

- Step 1: Create a list of all the elements that appear in every transaction and create a frequency table.
- Step 2: Set the minimum level of support. Only those elements whose support exceeds or equals the threshold support are significant.
- Step 3: All potential pairings of important elements must be made, bearing in mind that AB and BA are interchangeable.
- Step 4: Tally the number of times each pair appears in a transaction.
- Step 5: Only those sets of data that meet the criterion of support are significant.
- Step 6: Now, suppose you want to find a set of three things that may be bought together. A rule, known as self-join, is needed to build a three-item set. The item pairings OP, OB, PB, and PM state that two combinations with the same initial letter are sought from these sets.

The rapid rise of e-commerce apps has increased the accumulation of data. To forecast outcomes, data mining, also known as KDD (Knowledge Discovery in Databases), is used to detect irregularities, linkages, trends and patterns in data.

An algorithm known as Apriori is a common one in data mining. It's used to identify the most frequently occurring elements and meaningful associations in a dataset. As an example, products brought in by consumers to a shop may all be used as inputs in this system.

An effective Market Basket Analysis is critical since it allows consumers to purchase their products with more convenience, resulting in a rise in market sales. Furthermore, it has been applied in healthcare to aid in the identification of harmful medication responses. A clustering algorithm is generated that identifies which combinations of drugs and patient factors are associated with adverse drug reactions.

Become the Highest Paid AI Engineer!

With Our Trending AI Engineer Master ProgramKnow More

Apriori Property

In 1994, R. Agrawal and R. Srikant developed the Apriori method for identifying the most frequently occurring itemsets in a dataset using the boolean association rule. Since it makes use of previous knowledge about common itemset features, the method is referred to as Apriori. This is achieved by the use of an iterative technique or level-wise approach, in which k-frequent itemsets are utilized to locate k+1 itemsets.

An essential feature known as the Apriori property is utilized to boost the effectiveness of level-wise production of frequent itemsets. This property helps by minimizing the search area, which in turn serves to maximize the productivity of level-wise creation of frequent patterns.

How Does the Apriori Algorithm Work?

The Apriori algorithm operates on a straightforward premise. When the support value of an item set exceeds a certain threshold, it is considered a frequent item set. Take into account the following steps. To begin, set the support criterion, meaning that only those things that have more than the support criterion are considered relevant.

- Step 1: Create a list of all the elements that appear in every transaction and create a frequency table.
- Step 2: Set the minimum level of support. Only those elements whose support exceeds or equals the threshold support are significant.
- Step 3: All potential pairings of important elements must be made, bearing in mind that AB and BA are interchangeable.
- Step 4: Tally the number of times each pair appears in a transaction.
- Step 5: Only those sets of data that meet the criterion of support are significant.
- Step 6: Now, suppose you want to find a set of three things that may be bought together. A rule, known as self-join, is needed to build a three-item set. The item pairings OP, OB, PB, and PM state that two combinations with the same initial letter are sought from these sets.

1. OPB is the result of OP and OB.
2. PBM is the result of PB and PM.

- Step 7: When the threshold criterion is applied again, you'll get the significant itemset.

Become the Highest Paid AI Engineer!
With Our Trending AI Engineer Master ProgramKnow More

Steps for Apriori Algorithm

The Apriori algorithm has the following steps:

- Step 1: Determine the level of transactional database support and establish the minimal degree of assistance and dependability.

- **Step 2: Take all of the transaction's supports that are greater than the standard or chosen support value.**
- **Step 3: Look for all rules with greater precision than the cutoff or baseline standard, in these subgroups.**
- **Step 4: It is best to arrange the rules in ascending order of strength.**

Methods to Improve Apriori Efficiency

The algorithm's efficiency may be improved in a variety of ways.

- **Hash-Based Technique**

Using a hash-based structure known as a hash table, the k-itemsets and their related counts are generated. The table is generated using a hash function.

- **Transaction Reduction**

There are fewer transactions to scan throughout each loop when using this strategy. Items that are not often used in a process are either tagged or deleted.

- **Partitioning**

Two database searches are all that is needed to find the frequently occurring itemsets using this approach. For any item set to be considered "possibly frequent" in the database, it must be prevalent in at least a few of the database subdivisions.

- **Sampling**

A random sample S is selected from database D, and then a search is conducted for frequent itemsets within that sample S. Global frequent itemsets may be misplaced. By reducing the min sup, this may be decreased.

- **Dynamic Itemset Counting**

During the screening of the dataset, this approach may add new iterations at any indicated starting position of the directory.

Advantages of Apriori

- **An algorithm that is simple to grasp.**
- **The Merge and Squash processes are simple to apply on big itemsets in huge databases.**

Disadvantages of Apriori

- **It requires a significant amount of calculations if the itemsets are extremely big and the minimal support is maintained to a bare minimum.**
- **A full scan of the whole database is required.**

Table of content

- [What Is Association Rule Mining?]
- [What Is an Apriori Algorithm?]
- [How Does the Apriori Algorithm Work?]
- [Support]
- [Confidence]

Show More

Introduction to Apriori Algorithm in Python

The Apriori algorithm is a classical algorithm in Data Mining that is used for mining frequent itemsets and association rule mining.

What is Association Rule Mining?

As mentioned before, the Apriori algorithm is used for association rule mining. Now, what is association rule mining? Association rule mining is a technique to identify frequent patterns and associations among a set of items.

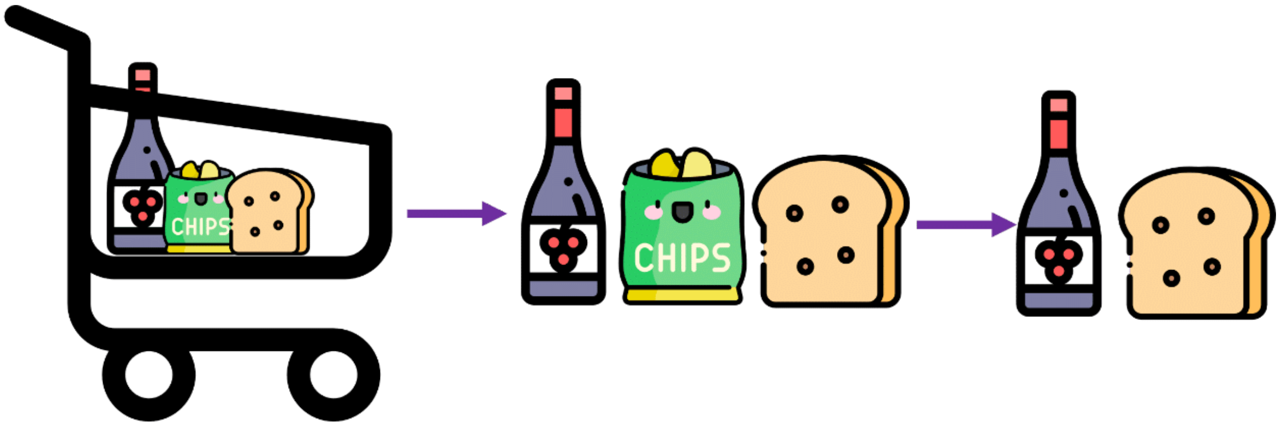
For example, understanding customer buying habits. By finding correlations and associations between different items that customers place in their ‘shopping basket,’ recurring patterns can be derived.

Say, Joshua goes to buy a bottle of wine from the supermarket. He also grabs a couple of chips as well. The manager there analyses that, not only Joshua, people often tend to buy wine and chips together. After finding out the pattern, the manager starts to arrange these items together and notices an increase in sales.

This process of identifying an association between products/items is called association rule mining. To implement association rule mining, many algorithms have been developed. The Apriori algorithm is one of the most popular and arguably the most efficient algorithms among them. Let us discuss what an Apriori algorithm is.

What Is an Apriori Algorithm?

Apriori algorithm assumes that any subset of a frequent itemset must be frequent.



Say, a transaction containing {wine, chips, bread} also contains {wine, bread}. So, according to the principle of Apriori, if {wine, chips, bread} is frequent, then {wine, bread} must also be frequent.

How Does the Apriori Algorithm Work?

The key concept in the Apriori algorithm is that it assumes all subsets of a frequent itemset to be frequent. Similarly, for any infrequent itemset, all its supersets must also be infrequent.

Let us try and understand the workings of an Apriori algorithm with the help of a very famous business scenario, market basket analysis.

Here is a dataset consisting of six transactions in an hour. Each transaction is a combination of 0s and 1s, where 0 represents the absence of an item and 1 represents the presence of it.

Transaction ID	Wine	Chips	Bread	Milk
1	1	1	1	1
2	1	0	1	1
3	0	0	1	1
4	0	1	0	0
5	1	1	1	1
6	1	1	0	1

We can find multiple rules in this scenario. For example, in a transaction of wine, chips, and bread, if wine and chips are bought, then customers also buy bread.

{wine, chips} =>; {bread}

To select the interesting rules out of multiple possible rules from this small business scenario, we will be using the following measures:

- Support
- Confidence
- List
- Conviction

Support

Support of item x is nothing but the ratio of the number of transactions in which item x appears to the total number of transactions.

i.e.,

$$\text{Support(wine)} = \frac{\text{Number of transactions in which the item wine appears}}{\text{Total number of transactions}}$$

$$\text{Support(wine)} = \frac{4}{6} = 0.66667$$

Confidence

Confidence (x => y) signifies the likelihood of the item y being purchased when item x is purchased. This method takes into account the popularity of item x.

i.e.,

$$\text{Conf}(\{\text{wine, chips}\} \Rightarrow \{\text{bread}\}) = \frac{support(wine,chips,bread)}{support(wine,chips)}$$

$$\text{Conf}(\{\text{wine, chips}\} \Rightarrow \{\text{bread}\}) = \frac{\frac{2}{6}}{\frac{3}{6}} = 0.667$$

Lift

Lift (x => y) is nothing but the 'interestingness' or the likelihood of the item y being purchased when item x is sold. Unlike confidence (x => y), this method takes into account the popularity of the item y.

i.e.,

$$\text{lift}(\{\text{wine, chips}\} \Rightarrow \{\text{bread}\}) = \frac{support(wine,chips,bread)}{support(wine,chips)}$$

$$\text{lift}(\{\text{wine, chips}\} \Rightarrow \{\text{bread}\}) = \frac{\frac{2}{6}}{\frac{3}{6} * \frac{4}{6}} = 1$$

- Lift (x => y) = 1 means that there is no correlation within the itemset.
- Lift (x => y) > 1 means that there is a positive correlation within the itemset, i.e., products in the itemset, x and y, are more likely to be bought together.
- Lift (x => y) < 1 means that there is a negative correlation within the itemset, i.e., products in itemset, x and y, are unlikely to be bought together.

Conviction

Conviction of a rule can be defined as follows:

conv($x \Rightarrow y$) = $\frac{1 - supp(y)}{1 - conf(x => y)}$

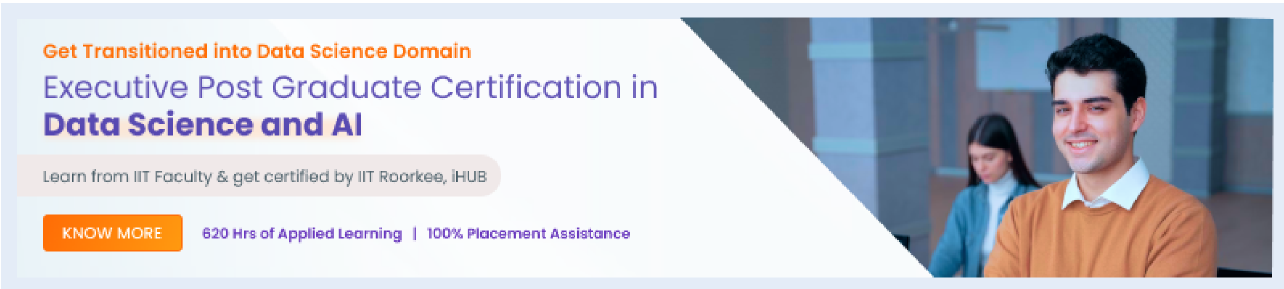
i.e.,

conv({wine, chips} => {bread}) = $\frac{1 - supp(bread)}{1 - conf(\{wine, chips\} => \{bread\})}$ = $\frac{1 - \frac{4}{6}}{1 - \frac{2}{3}}$ =1

Its value range is [0, +∞].

- Conv(x => y) = 1 means that x has no relation with y.
- Greater the conviction higher the interest in the rule.

Now that we know the methods to find out the interesting rules, let us go back to the example. Before we get started, let us fix the support threshold to 50 percent.



Step 1: Create a frequency table of all the items that occur in all transactions

Item	Frequency
Wine	4
Chips	4
Bread	4
Milk	5

Step 2: Find the significant items based on the support threshold

Support threshold = 3

Item	Frequency
Wine	4
Chips	4
Bread	4
Milk	5

Step 3: From the significant items, make possible pairs irrespective of the order

Item	Frequency
Wine, Chips	3
Wine, Bread	3
Wine, Milk	4
Chips, Bread	2
Chips, Milk	3
Bread, Milk	4

Step 4: Again, find the significant items based on the support threshold

Item	Frequency
Wine, Milk	4
Bread, Milk	4

Step 5: Now, make a set of three items that are bought together based on the significant items from Step 4

Item	Frequency
Wine, Bread, Milk	3

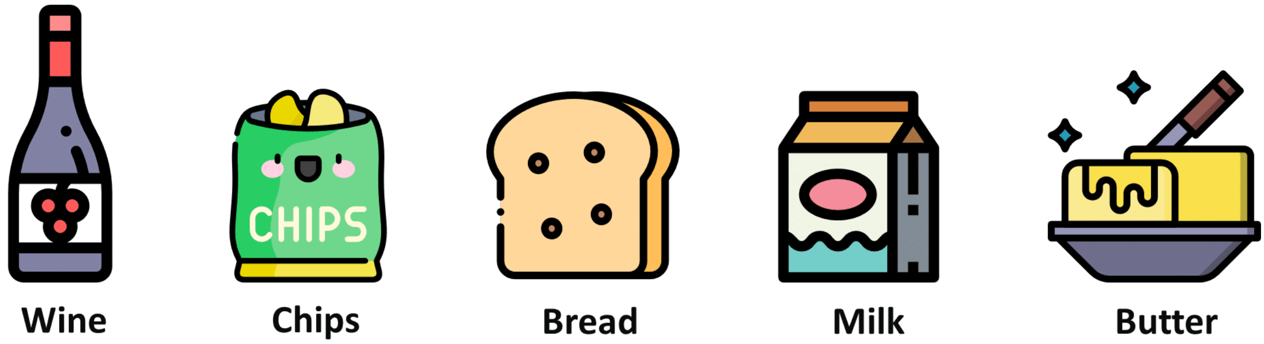
{Wine, Bread, Milk} is the only significant item set we have got from the given data. But in real-world scenarios, we would have dozens of items to build rules from. Then, we might have to make four/five-pair itemsets.

Watch Apriori Algorithm Tutorial

Hands-on: Apriori Algorithm in Python- Market Basket Analysis

Problem Statement

The manager of a retail store is trying to find out an association rule between six items, to figure out which items are more often bought together so that he can keep the items together in order to increase sales.



Dataset

Below is the transaction data from Day 1. This dataset contains 6 items and 22 transaction records.

	A	B	C	D	E	F
1	Wine	Chips	Bread	Butter	Milk	Apple
2	Wine		Bread	Butter	Milk	
3			Bread	Butter	Milk	
4		Chips				Apple
5	Wine	Chips	Bread	Butter	Milk	Apple
6	Wine	Chips			Milk	
7	Wine	Chips	Bread	Butter		Apple
8	Wine	Chips			Milk	
9	Wine		Bread			Apple
10	Wine		Bread	Butter	Milk	
11		Chips	Bread	Butter		Apple
12	Wine			Butter	Milk	Apple
13	Wine	Chips	Bread	Butter	Milk	
14	Wine		Bread		Milk	Apple
15	Wine		Bread	Butter	Milk	Apple
16	Wine	Chips	Bread	Butter	Milk	Apple
17		Chips	Bread	Butter	Milk	Apple
18		Chips		Butter	Milk	Apple
19	Wine	Chips	Bread	Butter	Milk	Apple
20	Wine		Bread	Butter	Milk	Apple
21	Wine	Chips	Bread		Milk	Apple
22		Chips				

Environment Setup:

Before we move forward, we need to install the ‘apyori’ package first.

pip install apyori

Anaconda Prompt

```
(base) C:\Users\intellipaat>pip install apyori
Collecting apyori
  Downloading https://files.pythonhosted.org/packages/25/fd/0561e2dd29aeed544bad2d1991636e38700cdaef9530490b863741f35295/apyori-1.1.1.tar.gz
Building wheels for collected packages: apyori
  Running setup.py bdist_wheel for apyori ... done
  Stored in directory: C:\Users\intellipaat\AppData\Local\pip\Cache\wheels\7b\2a\35\c0c3749c1a36d4f454ea22d8396e1b854b86340d63cbbb7949
Successfully built apyori
Installing collected packages: apyori
Successfully installed apyori-1.1.1
```

Get 100% Hike!

Master Most in Demand Skills Now!

+91 IN INDIA+44 UK UNITED KINGDOM+1 US UNITED STATES+1 CA CANADA------ -- ------------------------+376 AD ANDORRA+971 AE UNITED ARAB EMIRATES+93 AF AFGHANISTAN+1268 AG ANTIGUA AND BARBUDA+1264 AI ANGUILLA+355 AL ALBANIA+374 AM ARMENIA+599 AN NETHERLANDS ANTILLES+244 AO ANGOLA+672 AQ ANTARCTICA+54 AR ARGENTINA+1684 AS AMERICAN SAMOA+43 AT AUSTRIA+61 AU AUSTRALIA+297 AW ARUBA+994 AZ AZERBAIJAN+387 BA BOSNIA AND HERZEGOVINA+1246 BB BARBADOS+880 BD BANGLADESH+32 BE BELGIUM+226 BF BURKINA FASO+359 BG BULGARIA+973 BH BAHRAIN+257 BI BURUNDI+229 BJ BENIN+590 BL SAINT BARTHELEMY+1441 BM BERMUDA+673 BN BRUNEI DARUSSALAM+591 BO BOLIVIA+55 BR BRAZIL+1242 BS BAHAMAS+975 BT BHUTAN+267 BW BOTSWANA+375 BY BELARUS+501 BZ BELIZE+61 CC COCOS (KEELING ISLANDS+243 CD CONGO, THE DEMOCRATIC REPUBLIC OF THE+236 CF CENTRAL AFRICAN REPUBLIC+242 CG CONGO+41 CH SWITZERLAND+225 CI COTE D IVOIRE+682 CK COOK ISLANDS+56 CL CHILE+237 CM CAMEROON+86 CN CHINA+57 CO COLOMBIA+506 CR COSTA RICA+53 CU CUBA+238 CV CAPE VERDE+61 CX CHRISTMAS ISLAND+357 CY CYPRUS+420 CZ CZECH REPUBLIC+49 DE GERMANY+253 DJ DJIBOUTI+45 DK DENMARK+1767 DM DOMINICA+1809 DO DOMINICAN REPUBLIC+213 DZ ALGERIA+593 EC ECUADOR+372 EE ESTONIA+20 EG EGYPT+291 ER ERITREA+34 ES SPAIN+251 ET ETHIOPIA+358 FI FINLAND+679 FJ FIJI+500 FK FALKLAND ISLANDS (MALVINAS+691 FM MICRONESIA, FEDERATED STATES OF+298 FO FAROE

ISLANDS+33 FR FRANCE+241 GA GABON+1473 GD GRENADA+995 GE GEORGIA+233 GH GHANA+350 GI GIBRALTAR+299 GL GREENLAND+220 GM GAMBIA+224 GN GUINEA+240 GQ EQUATORIAL GUINEA+30 GR GREECE+502 GT GUATEMALA+1671 GU GUAM+245 GW GUINEA-BISSAU+592 GY GUYANA+852 HK HONG KONG+504 HN HONDURAS+385 HR CROATIA+509 HT HAITI+36 HU HUNGARY+62 ID INDONESIA+353 IE IRELAND+972 IL ISRAEL+44 IM ISLE OF MAN+964 IQ IRAQ+98 IR IRAN, ISLAMIC REPUBLIC OF+354 IS ICELAND+39 IT ITALY+1876 JM JAMAICA+962 JO JORDAN+81 JP JAPAN+254 KE KENYA+996 KG KYRGYZSTAN+855 KH CAMBODIA+686 KI KIRIBATI+269 KM COMOROS+1869 KN SAINT KITTS AND NEVIS+850 KP KOREA DEMOCRATIC PEOPLES REPUBLIC OF+82 KR KOREA REPUBLIC OF+965 KW KUWAIT+1345 KY CAYMAN ISLANDS+7 KZ KAZAKSTAN+856 LA LAO PEOPLES DEMOCRATIC REPUBLIC+961 LB LEBANON+1758 LC SAINT LUCIA+423 LI LIECHTENSTEIN+94 LK SRI LANKA+231 LR LIBERIA+266 LS LESOTHO+370 LT LITHUANIA+352 LU LUXEMBOURG+371 LV LATVIA+218 LY LIBYAN ARAB JAMAHIRIYA+212 MA MOROCCO+377 MC MONACO+373 MD MOLDOVA, REPUBLIC OF+382 ME MONTENEGRO+1599 MF SAINT MARTIN+261 MG MADAGASCAR+692 MH MARSHALL ISLANDS+389 MK MACEDONIA, THE FORMER YUGOSLAV REPUBLIC OF+223 ML MALI+95 MM MYANMAR+976 MN MONGOLIA+853 MO MACAU+1670 MP NORTHERN MARIANA ISLANDS+222 MR MAURITANIA+1664 MS MONTSERRAT+356 MT MALTA+230 MU MAURITIUS+960 MV MALDIVES+265 MW MALAWI+52 MX MEXICO+60 MY MALAYSIA+258 MZ MOZAMBIQUE+264 NA NAMIBIA+687 NC NEW CALEDONIA+227 NE NIGER+234 NG NIGERIA+505 NI NICARAGUA+31 NL NETHERLANDS+47 NO NORWAY+977 NP NEPAL+674 NR NAURU+683 NU NIUE+64 NZ NEW ZEALAND+968 OM OMAN+507 PA PANAMA+51 PE PERU+689 PF FRENCH POLYNESIA+675 PG PAPUA NEW GUINEA+63 PH PHILIPPINES+92 PK PAKISTAN+48 PL POLAND+508 PM SAINT PIERRE

AND MIQUELON+870 PN PITCAIRN+1 PR PUERTO RICO+351 PT PORTUGAL+680 PW PALAU+595 PY PARAGUAY+974 QA QATAR+40 RO ROMANIA+381 RS SERBIA+7 RU RUSSIAN FEDERATION+250 RW RWANDA+966 SA SAUDI ARABIA+677 SB SOLOMON ISLANDS+248 SC SEYCHELLES+249 SD SUDAN+46 SE SWEDEN+65 SG SINGAPORE+290 SH SAINT HELENA+386 SI SLOVENIA+421 SK SLOVAKIA+232 SL SIERRA LEONE+378 SM SAN MARINO+221 SN SENEGAL+252 SO SOMALIA+597 SR SURINAME+239 ST SAO TOME AND PRINCIPE+503 SV EL SALVADOR+963 SY SYRIAN ARAB REPUBLIC+268 SZ SWAZILAND+1649 TC TURKS AND CAICOS ISLANDS+235 TD CHAD+228 TG TOGO+66 TH THAILAND+992 TJ TAJIKISTAN+690 TK TOKELAU+670 TL TIMOR-LESTE+993 TM TURKMENISTAN+216 TN TUNISIA+676 TO TONGA+90 TR TURKEY+1868 TT TRINIDAD AND TOBAGO+688 TV TUVALU+886 TW TAIWAN, PROVINCE OF CHINA+255 TZ TANZANIA, UNITED REPUBLIC OF+380 UA UKRAINE+256 UG UGANDA+598 UY URUGUAY+998 UZ UZBEKISTAN+39 VA HOLY SEE (VATICAN CITY STATE+1784 VC SAINT VINCENT AND THE GRENADINES+58 VE VENEZUELA+1284 VG VIRGIN ISLANDS, BRITISH+1340 VI VIRGIN ISLANDS, U.S.+84 VN VIET NAM+678 VU VANUATU+681 WF WALLIS AND FUTUNA+685 WS SAMOA+381 XK KOSOVO+967 YE YEMEN+262 YT MAYOTTE+27 ZA SOUTH AFRICA+260 ZM ZAMBIA+263 ZW ZIMBABWE

By providing your contact details, you agree to our Terms of Use & Privacy Policy

Market Basket Analysis Implementation within Python

With the help of the apyori package, we will be implementing the Apriori algorithm in order to help the manager in market basket analysis.

Step 1: Import the libraries

```
In [1]: #Importing the required datasets
        import numpy as np
        import pandas as pd
        from apyori import apriori
```

Step 2: Load the dataset

```
In [2]: #Loading the dataset
        store_data = pd.read_csv('Day1.csv', header=None)
```

Step 3: Have a glance at the records

```
In [3]: #Having a glance at the records
        store_data
```

Out[3]:

	0	1	2	3	4	5
0	Wine	Chips	Bread	Butter	Milk	Apple
1	Wine	NaN	Bread	Butter	Milk	NaN
2	NaN	NaN	Bread	Butter	Milk	NaN
3	NaN	Chips	NaN	NaN	NaN	Apple
4	Wine	Chips	Bread	Butter	Milk	Apple
5	Wine	Chips	NaN	NaN	Milk	NaN
6	Wine	Chips	Bread	Butter	NaN	Apple
7	Wine	Chips	NaN	NaN	Milk	NaN
8	Wine	NaN	Bread	NaN	NaN	Apple
9	Wine	NaN	Bread	Butter	Milk	NaN
10	NaN	Chips	Bread	Butter	NaN	Apple
11	Wine	NaN	NaN	Butter	Milk	Apple
12	Wine	Chips	Bread	Butter	Milk	NaN
13	Wine	NaN	Bread	NaN	Milk	Apple
14	Wine	NaN	Bread	Butter	Milk	Apple
15	Wine	Chips	Bread	Butter	Milk	Apple
16	NaN	Chips	Bread	Butter	Milk	Apple
17	NaN	Chips	NaN	Butter	Milk	Apple
18	Wine	Chips	Bread	Butter	Milk	Apple
19	Wine	NaN	Bread	Butter	Milk	Apple
20	Wine	Chips	Bread	NaN	Milk	Apple
21	NaN	Chips	NaN	NaN	NaN	NaN

Step 4: Look at the shape

In [4]:
```
store_data.shape
```

Out[4]: (22, 6)

Step 5: Convert Pandas DataFrame into a list of lists

In [5]:
```
#Converting the pandas dataframe into a list of lists
records = []
for i in range(0, 22):
    records.append([str(store_data.values[i,j]) for j in range(0, 6)])
```

Step 6: Build the Apriori model

In [7]:
```
#Building the first apriori model
association_rules = apriori(records, min_support=0.50, min_confidence=0.7, min_lift=1.2, min_length=2)
association_results = list(association_rules)
```

Step 7: Print out the number of rules

In [8]:
```
#Getting the number of rules
print(len(association_results))
```

Step 8: Have a glance at the rule

```
In [10]: #Glancing at the first rule
         print(association_results)
```

```
[RelationRecord(items=frozenset({'Milk', 'Butter', 'Bread'}), support=0.5, ordered_statistics=[OrderedStatistic(items_base=froz
enset({'Milk', 'Bread'}), items_add=frozenset({'Butter'}), confidence=0.8461538461538461, lift=1.241025641025641)])]
```

The support value for the first rule is 0.5. This number is calculated by dividing the number of transactions containing 'Milk,' 'Bread,' and 'Butter' by the total number of transactions.

The confidence level for the rule is 0.846, which shows that out of all the transactions that contain both "Milk" and "Bread", 84.6 % contain 'Butter' too.

The lift of 1.241 tells us that 'Butter' is 1.241 times more likely to be bought by the customers who buy both 'Milk' and 'Butter' compared to the default likelihood of sale of 'Butter.'

Limitations of Apriori Algorithm

Despite being simple one, Apriori algorithms have some limitations including:

- Waste of time when it comes to handling a large number of candidates with frequent itemsets.
- The efficiency of this algorithm goes down when there is a large number of transactions going on through a limited memory capacity.
- Requires high computation power and needs to scan the entire database.

Improvements

The following are the ways to improve the efficiency of the algorithm:

- Use hashing techniques to reduce the number of database scans.
- Do not take the infrequent transaction further into consideration.

- If a purchase is frequent in one partition, it should be frequent in another partition.
- Try to pick up random samples to improve the accuracy of your algorithm.
- Use dynamic itemset counting to introduce new candidate itemsets while the scanning of the database is performed.

Applications of Apriori Algorithm

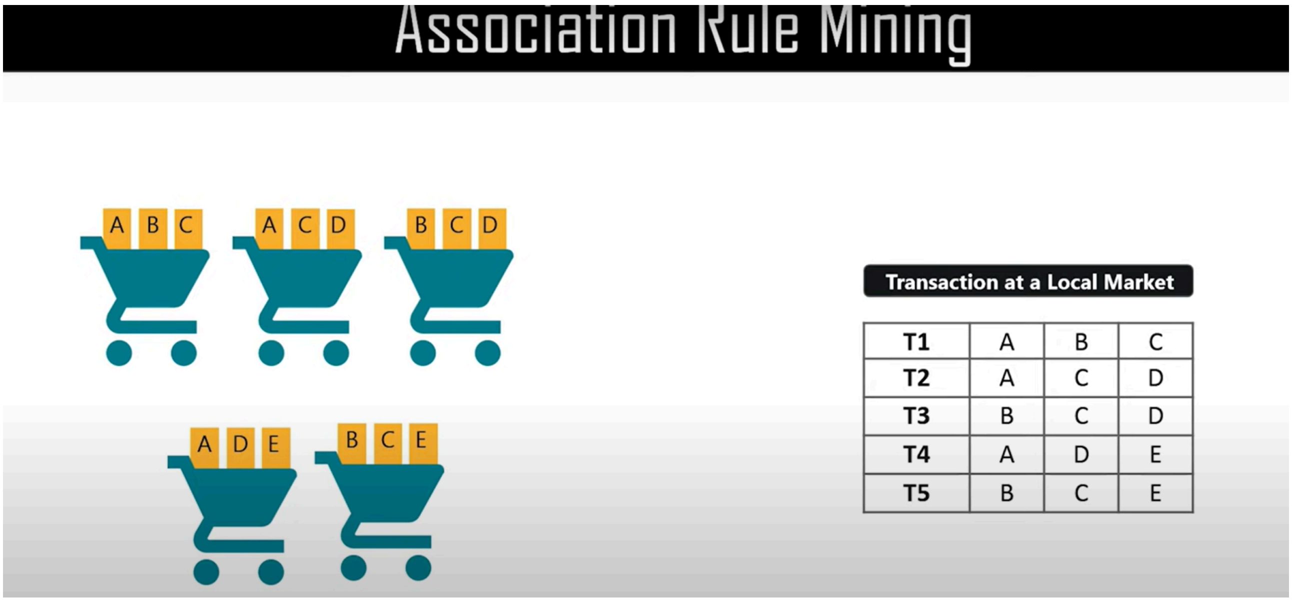
Some of the popular applications of the algorithm are:

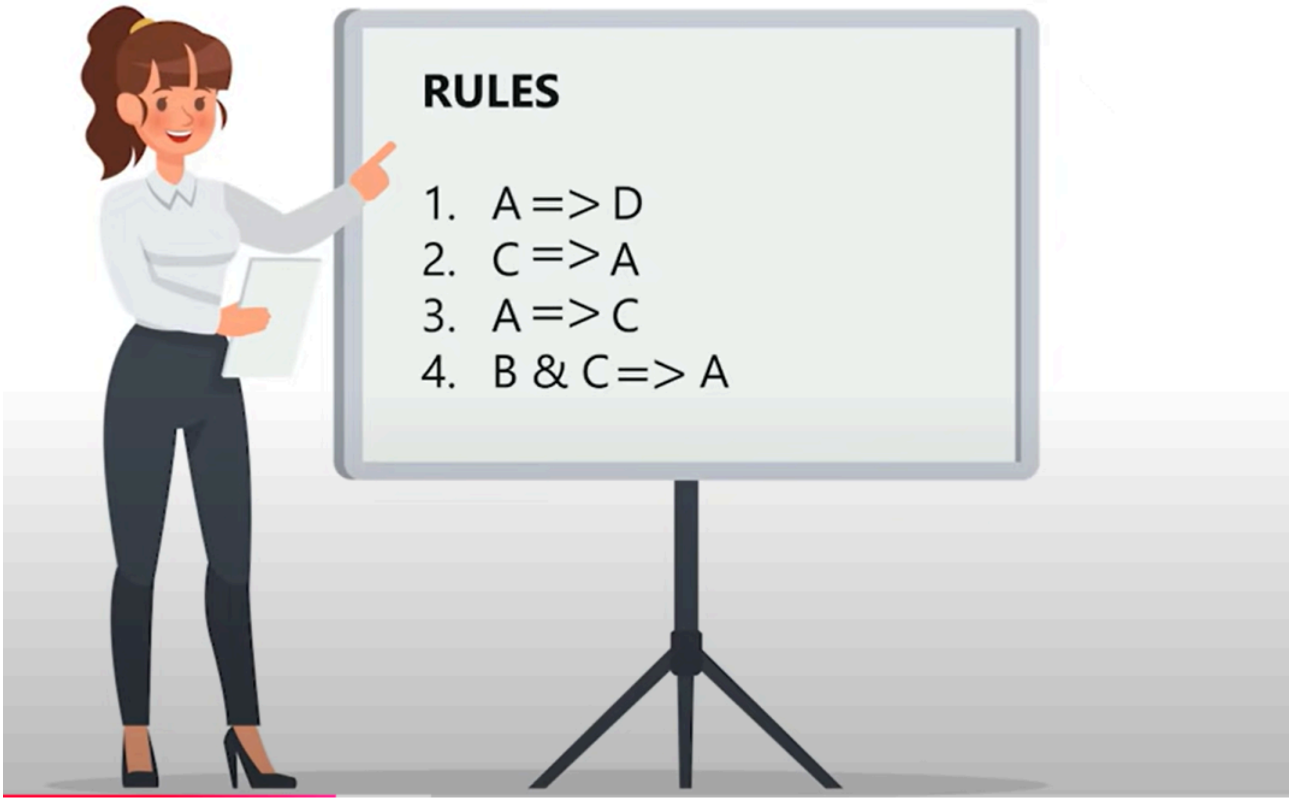
- Used in forest departments to understand the intensity and probability of forest fires.
- Used by Google and other search engines for their auto-complete features.
- The Healthcare department used such algorithms to analyze the patients' database and predict which patients might develop blood pressure, diabetes, or other common disease.
- Used to categorize students based on their specialties and performance to improve their academic performance.
- E-commerce websites use it in their recommendation systems to provide a better user experience.

What Did We Learn?

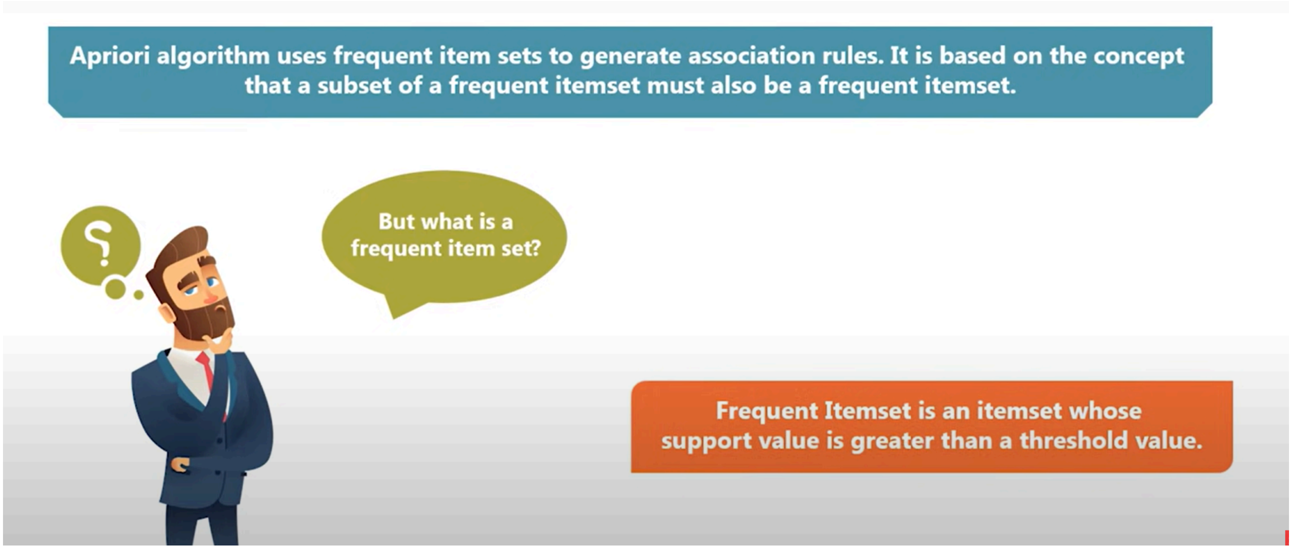
In this tutorial, we have learned what association rule mining is, and what the Apriori algorithm is, and with the help of an Apriori algorithm example, we learned how the Apriori algorithm works. In the end, we have built an Apriori model in Python programming language on market basket analysis. Python Programming Course is one of the most demanding skills right now in the market.

Example





Rule	Support	Confidence	Lift
A=>D	2/5	2/3	10/9
C => A	2/5	2/4	5/6
A => C	2/5	2/3	5/6
B, C=>A	1/5	1/3	5/9



TID	Items
T1	1 3 4
T2	2 3 5
T3	1 2 3 5
T4	2 5
T5	1 3 5

Min. Support count = 2

C1

TID	Items
T1	1 3 4
T2	2 3 5
T3	1 2 3 5
T4	2 5
T5	1 3 5

Itemset	Support
{1}	3
{2}	3
{3}	4
{4}	1
{5}	4

Apriori Algorithm - 1st Iteration

C1

Itemset	Support
{1}	3
{2}	3
{3}	4
{4}	1
{5}	4

F1

Itemset	Support
{1}	3
{2}	3
{3}	4
{5}	4

Item sets with support value less than min. support value (i.e. 2) are eliminated