Association Rule Mining via Apriori Algorithm in Python and Knime

Noreen Naz
Riphah International University
Lahore, Pakistan
noreennazmrd123@gmail.com

Umair Ashraf
Riphah International University
Lahore, Pakistan
umairashraf7566@gmail.com

Hadiqa Ilyas
Riphah International University
Lahore, Pakistan
hadiqailyas15@gmail.com

***Abstract*— Association Rule Mining has many useful applications in business environment like Market Basket Analysis. In Market Basket Analysis, the relationship between multiple purchased items is derived so that the products most often bought together should be kept together in racks and their marketing should be done together and so on.**
This paper implements Apriori Algorithm on a store sale data using Python and Knime tools, in order to extract and then compare useful association rules between store items.

Keywords— Association Rule Mining, Apriori, Python, Knime, Market Basket Analysis, Confidence, Support, Lift

I. INTRODUCTION

Association Rule Mining is a technique used to uncover hidden relationships between different items in transactions data. It highlights that which items come together in a particular transaction. For example mothers buying diapers in a super market typically buy baby wipes too. A customer who buys bread might also buy eggs and so on.
Association Rule discovers some interesting relationships in such transactions, which can further be utilized to gain more profit for the company.
Apriori is one of the most commonly used algorithm for Association Rule Mining. Apriori Algorithm was developed by Agrawal and Srikant [1]. It has three main parts:

- Support- Support is calculated by dividing the number of records/transactions containing any product by total number of over all records.
- Confidence- Confidence for two items A and B is the ratio of number of transactions having both A and B by the total number of transactions having A.
- Lift- Lift is used to measure the accuracy of the confidence over how often item B is purchased.
  (Confidence A+B) / (Support B)
  If the lift value is more than 1, the particular items are said be associated with each other in buying relationship. If the lift value is less than 1, the particular items have no together buying history. If this value is 1, the particular items have no association.

In this paper, we have implemented Apriori algorithm on a super store transactional data in order to extract the association rules between different store items. Python and Knime tools have been used for this purpose.

II. RELATED WORK

[1] has explored Educational Data Mining and suggested an improved form of Apriori algorithm in order to evaluate the frequent trends in educational data.
[2] has described Apriori algorithm and Market Basket Analysis in a very effective way.
[3] have proposed a set of association rules extracted from different research studies data using Apriori method. [4] have explained Association Rule Mining and Apriori algorithm in a simple way.
[5] has introduced a new weighted association rule algorithm in order to enhance the overall functionality and work speed of data mining process [6] have presented a very compact and brief survey of Association Rule Mining research work done by many different researchers in this field.
[7] have a done a comparison between different Association Rule Mining Algorithms in a simple way.
[8] have explored Apriori algorithm and FP-growth algorithm for Association Rule Mining.

III. RESEARCH METHODOLOGY

A. *Dataset used*

A French Retail store dataset taken have been selected. This dataset has almost 7500 transactions of a week.
This dataset can be downloaded from this link:
https://drive.google.com/file/d/1y5DYn0dGoSbC22xowBq2d4po6h1JxcTQ/view?usp=sharing

B. *Software Used*

Kaggle is a platform to perform analytics and predictive modeling, arrange competitions related to Data Science around the world. Moreover it has powerful tools and resources to test our projects. Python language is used for project coding [9].
Knime (Konstanz Information Miner) tool is one of the best data analytics tool developed by Silicon Valley Software Company at the University of Konstanz in early 2004.
Knime has better Graphical User Interface (GUI), where the connections of data nodes are easily built.

Knime has useful features to support the implementation of Association Rule Mining for Market Basket Analysis [10].

C. Pyhton Project Steps explained:

All Steps of the project explained here:

Step 1: Importing required libraries and Loading the dataset

Step 2: Data Preprocessing (Converting dataset to list to fulfill the requirements of Apriori Algorithm)

Step 3: Applying Apriori Algorithm with minimum values for support, confidence and lift parameters so that proper Association Rues can be extracted. I set minimum_support=0.0045, minimum_confidence=0.2, minimum_lift=3, minimum_length=2. I got 48 Association rules as a result, shown in the results section.

D. Knime Workflow explained

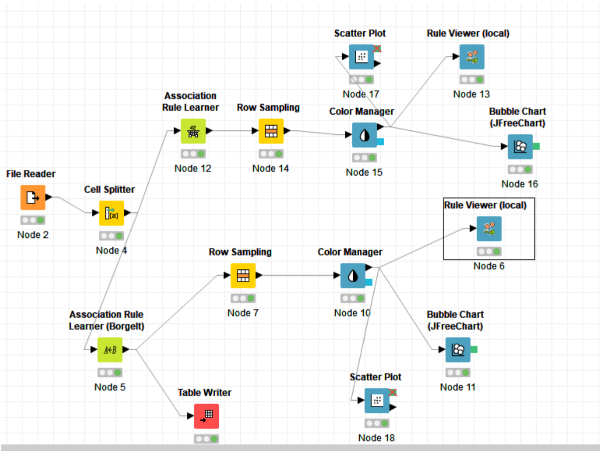

Fig. 1. Knime workflow

Step 1: File reader node is used to read the store data.
Step 2: Cell splitter node is used to split the purchased products list. Comma(,) is used as delimiter.
Step 3: Association Rule Learner node is applied in order to get frequent itemsets.
Step 4: Row Sampling node is applied to get a sample of rules and then scatter plot, Rule Viewer and Bubble chart are used in order to visualize the extracted rules graphically.
Step 5: Association Rule Learner(Borgelt) node is also used on splitted dataset. This node uses Apriori algorithm being implemented by Christian Borgelt[2].
Step 6: Output Association Rules from Association Rule Learner(Borgelt) node are written to a file on hard disk through Table writer node.
Step 7: Row Sampling node is applied to get a sample of rules and then scatter plot, Rule Viewer and Bubble chart are used in order to visualize the extracted rules graphically.

IV. RESULTS AND COMPARATIVE ANALYSIS

Python Results

Fig. 2 shows some of the rules extracted through Python code.

```
Rule: chicken -> light cream
Support: 0.004532728969470737
Confidence: 0.29059829059829057
Lift: 4.84395061728395
=====================================
Rule: escalope -> mushroom cream sauce
Support: 0.005732568990801226
Confidence: 0.3006993006993007
Lift: 3.790832696715049
=====================================
Rule: escalope -> pasta
Support: 0.005865884548726837
Confidence: 0.3728813559322034
Lift: 4.700811850163794
=====================================
Rule: herb & pepper -> ground beef
Support: 0.015997866951073192
Confidence: 0.3234501347708895
Lift: 3.2919938411349285
=====================================
Rule: ground beef -> tomato sauce
Support: 0.005332622317024397
Confidence: 0.3773584905660377
Lift: 3.840659481324083
=====================================
Rule: olive oil -> whole wheat pasta
Support: 0.007998933475536596
Confidence: 0.2714932126696833
Lift: 4.122410097642296
-------------------------------------
=====================================
Rule: ground beef -> spaghetti
Support: 0.008665511265164644
Confidence: 0.3110047846889952
Lift: 3.165328208890303
=====================================
Rule: milk -> olive oil
Support: 0.004799360085321957
Confidence: 0.20338983050847456
Lift: 3.088314005352364
=====================================
Rule: shrimp -> mineral water
Support: 0.007199040127982935
Confidence: 0.3050847457627118
Lift: 3.200616332819722
=====================================
=====================================
Rule: cooking oil -> ground beef
Support: 0.004799360085321957
Confidence: 0.5714285714285714
Lift: 3.2819951870487856
=====================================
Rule: escalope -> mushroom cream sauce
Support: 0.005732568990801226
Confidence: 0.3006993006993007
Lift: 3.790832696715049
=====================================
Rule: escalope -> pasta
Support: 0.005865884548726837
Confidence: 0.3728813559322034
Lift: 4.700811850163794
=====================================
```

Fig. 2. Association Rules (Python)

Knime Results:

Fig. 3 shows the output of the Association Rule Learner node in Knime.

Row ID	Support	Confide...	Lift	Consequent	imp...	Items
rule0	0.005	0.333	1.398	mineral water	<---	[fromage blanc]
rule1	0.005	0.228	1.27	eggs	<---	[parmesan cheese]
rule2	0.005	0.291	4.844	chicken	<---	[light cream]
rule3	0.005	0.318	1.825	spaghetti	<---	[black tea]
rule4	0.005	0.274	1.575	spaghetti	<---	[white wine]
rule5	0.005	0.235	0.985	mineral water	<---	[parmesan cheese]
rule6	0.005	0.245	1.432	french fries	<---	[mushroom cream sauce]
rule7	0.005	0.304	1.748	spaghetti	<---	[carrots]
rule8	0.005	0.216	1.664	milk	<---	[fresh tuna]
rule9	0.005	0.255	1.494	french fries	<---	[rice]
rule10	0.005	0.216	1.316	chocolate	<---	[fresh tuna]
rule11	0.005	0.366	1.537	mineral water	<---	[gums]
rule12	0.005	0.346	1.924	eggs	<---	[black tea]
rule13	0.005	0.219	1.69	milk	<---	[french wine]
rule14	0.005	0.322	4.507	shrimp	<---	[pasta]
rule15	0.005	0.248	1.88	green tea	<---	[almonds]
rule16	0.005	0.237	0.996	mineral water	<---	[strawberries]
rule17	0.005	0.487	2.044	mineral water	<---	[nonfat milk]
rule18	0.005	0.273	1.57	spaghetti	<---	[protein bar]
rule19	0.005	0.27	1.548	spaghetti	<---	[rice]
rule20	0.005	0.358	2.188	chocolate	<---	[tomato sauce]
rule21	0.005	0.228	2.394	pancakes	<---	[fresh tuna]
rule22	0.005	0.255	2.924	burgers	<---	[almonds]
rule23	0.005	0.255	1.967	milk	<---	[almonds]
rule24	0.005	0.231	1.408	chocolate	<---	[french wine]
rule25	0.005	0.374	1.568	mineral water	<---	[black tea]
rule26	0.005	0.348	1.459	mineral water	<---	[carrots]
rule27	0.005	0.221	1.293	french fries	<---	[muffins]
rule28	0.005	0.342	1.964	spaghetti	<---	[light cream]
rule29	0.005	0.284	1.731	chocolate	<---	[rice]
rule30	0.005	0.377	3.841	ground beef	<---	[tomato sauce]
rule31	0.005	0.261	2.015	milk	<---	[meatballs]
rule32	0.005	0.201	1.118	eggs	<---	[light mayo]
rule33	0.005	0.202	1.559	milk	<---	[energy bar]
rule34	0.005	0.275	2.887	frozen vegetables	<---	[parmesan cheese]
rule35	0.005	0.201	2.114	pancakes	<---	[light mayo]
rule36	0.006	0.218	1.273	french fries	<---	[vegetables mix]
rule37	0.006	0.218	1.25	spaghetti	<---	[vegetables mix]
rule38	0.006	0.21	0.881	mineral water	<---	[energy drink]
rule39	0.006	0.205	0.86	mineral water	<---	[yogurt cake]

Fig. 3. Association Rules (Knime)

Fig. 4 shows output of the Association Rule Learner(Borgelt) node.

Consequent	Antecedent	ItemSe...	Relativ...	RuleCo...	Absolut...	Relativ...	RuleLift	RuleLift%	Absolut...	Relativ...
water spray	[pasta,shrimp]	1	0.013	2.63	38	0.507	65.798	6,579.8	3	0.04
water spray	[pasta]	1	0.013	0.847	118	1.57	21.189	2,118.9	3	0.04
water spray	[shrimp,frozen vegetables]	1	0.013	0.8	125	1.67	20.003	2,000.3	3	0.04
water spray	[shrimp,chocolate]	1	0.013	0.741	135	1.8	18.521	1,852.1	3	0.04
water spray	[shrimp]	2	0.027	0.373	536	7.15	9.33	932.96	3	0.04
napkins	[soda]	1	0.013	2.13	47	0.627	31.919	3,191.9	5	0.067
napkins	[shampoo]	1	0.013	2.7	37	0.493	40.546	4,054.6	5	0.067
napkins	[gluten free bar]	1	0.013	1.92	52	0.693	28.85	2,885	5	0.067
napkins	[cider]	1	0.013	1.27	79	1.05	18.99	1,899	5	0.067
napkins	[bacon]	1	0.013	1.54	65	0.867	23.08	2,308	5	0.067
napkins	[tomato sauce,mineral water]	1	0.013	2.33	43	0.573	34.888	3,488.8	5	0.067
napkins	[tomato sauce]	1	0.013	0.943	106	1.41	14.153	1,415.3	5	0.067
napkins	[carrots]	1	0.013	0.87	115	1.53	13.045	1,304.5	5	0.067
napkins	[muffins]	1	0.013	0.552	181	2.41	8.288	828.84	5	0.067
napkins	[parmesan cheese,spaghetti]	1	0.013	1.96	51	0.68	29.416	2,941.6	5	0.067
napkins	[parmesan cheese,mineral water]	1	0.013	2.86	35	0.467	42.863	4,286.3	5	0.067
napkins	[parmesan cheese]	1	0.013	0.671	149	1.99	10.068	1,006.8	5	0.067
napkins	[almonds]	1	0.013	0.654	153	2.04	9.805	980.52	5	0.067
napkins	[light mayo,mineral water]	1	0.013	2.13	47	0.627	31.919	3,191.9	5	0.067
napkins	[light mayo]	1	0.013	0.49	204	2.72	7.354	735.39	5	0.067
napkins	[red wine]	1	0.013	0.474	211	2.81	7.11	711	5	0.067
napkins	[butter,green tea]	1	0.013	1.82	55	0.733	27.276	2,727.6	5	0.067
napkins	[butter,spaghetti]	1	0.013	2.13	47	0.627	31.919	3,191.9	5	0.067
napkins	[butter]	1	0.013	0.442	226	3.01	6.638	663.81	5	0.067
napkins	[avocado]	1	0.013	0.4	250	3.33	6.001	600.08	5	0.067
napkins	[herb &pepper,grated cheese]	1	0.013	2.04	49	0.653	30.616	3,061.6	5	0.067
napkins	[herb &pepper,low fat yogurt]	1	0.013	2.27	44	0.587	34.095	3,409.5	5	0.067
napkins	[herb &pepper,pancakes]	1	0.013	1.82	55	0.733	27.276	2,727.6	5	0.067
napkins	[herb &pepper,ground beef,spaghetti]	1	0.013	2.08	48	0.64	31.254	3,125.4	5	0.067
napkins	[herb &pepper,ground beef]	2	0.027	1.67	120	1.6	25.003	2,500.3	5	0.067
napkins	[herb &pepper,green tea]	1	0.013	2.08	48	0.64	31.254	3,125.4	5	0.067
napkins	[herb &pepper,spaghetti]	1	0.013	0.82	122	1.63	12.297	1,229.7	5	0.067
napkins	[herb &pepper]	2	0.027	0.539	371	4.95	8.087	808.73	5	0.067
napkins	[grated cheese,low fat yogurt]	1	0.013	2.38	42	0.56	35.719	3,571.9	5	0.067
napkins	[grated cheese,pancakes]	1	0.013	1.92	52	0.693	28.85	2,885	5	0.067
napkins	[grated cheese,frozen vegetables]	1	0.013	1.96	51	0.68	29.416	2,941.6	5	0.067
napkins	[grated cheese,ground beef,spaghetti]	1	0.013	2.5	40	0.533	37.505	3,750.5	5	0.067
napkins	[grated cheese,ground beef]	1	0.013	1.18	85	1.13	17.649	1,764.9	5	0.067
napkins	[grated cheese,green tea]	1	0.013	1.69	59	0.787	25.427	2,542.7	5	0.067
napkins	[grated cheese,spaghetti]	1	0.013	0.806	124	1.65	12.098	1,209.8	5	0.067

Fig. 4 Association Rule Learner(Borgelt) node

Figure-5 shows output of the Rule viewer for Association Rule Learner node(Knime).

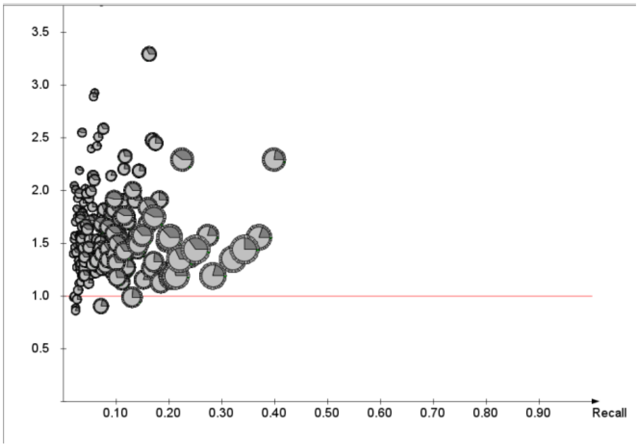

Fig. 5. Rule Viewer node output (Knime)

Fig. 6 shows output of the Rule viewer for Association Rule Learner node(Borgelt) .

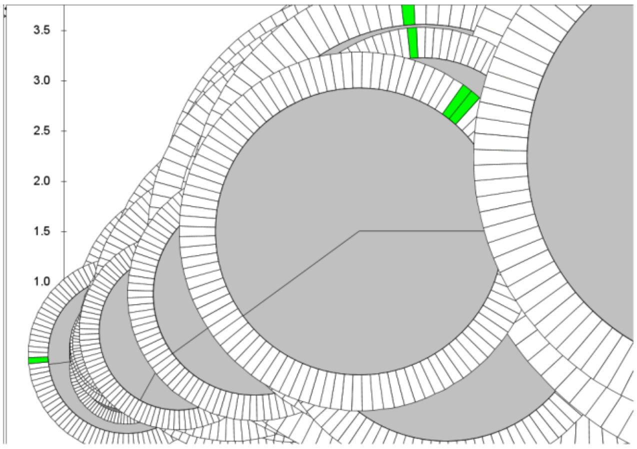

Fig. 6. Rule Viewer for Association Rule Learner node(Borgelt)

Fig. 7 shows output of the Scatter Plot for Association Rule Learner node.

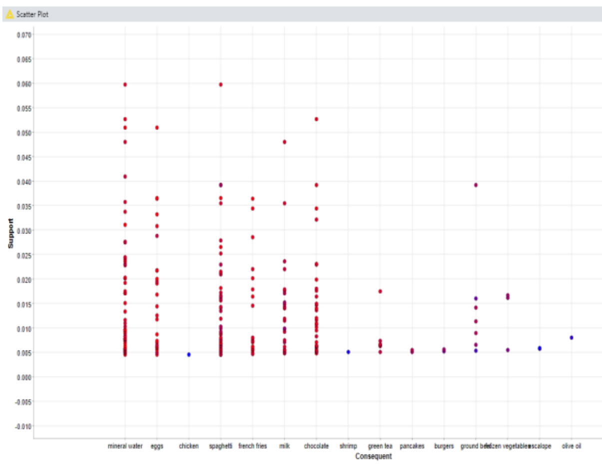

Fig. 7. Scatter plot for Association Rule Learner node

Fig. 8 shows output of the Bubble chart for Association Rule Learner node.

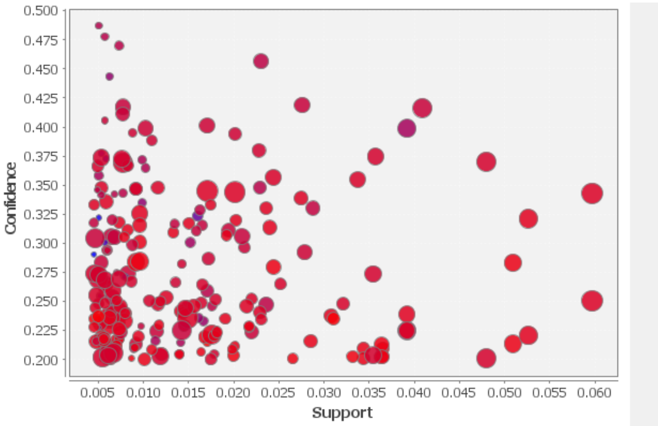

Fig. 8 Bubble chart for Association Rule Learner node

Figure-9 shows output of the Bubble chart for Association Rule Learner node(Borgelt).

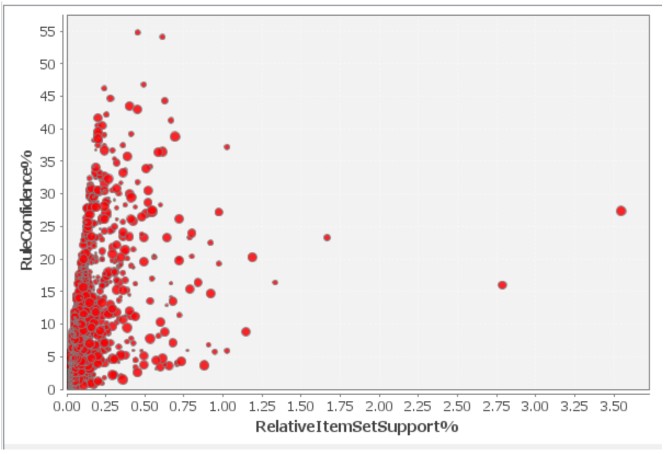

Fig. 9. Bubble chart for Association Rule Learner node(Borgelt)

From the above results, we can conclude that Knime tool results are more interactive and Association rules can be plotted in different ways.

V. CONCLUSION & FUTURE WORK

Association Rule Mining is a powerful technique used to find out some interesting hidden relationships between different products in transactional data. Apriori algorithm is most frequently used to derive association rules from given data.

In this paper Apriori algorithm is applied both in Python and Knime tools. Extracted association rules from both tools have been displayed. Knime tool being a GUI produces more healthy and graphical representations of extracted rules. Moreover with the help of different charts in Knime, support, confidence and lift factors of association rules are visualized very well.

In the future, we will try to use Association rule technique in order to build a Recommender System.

REFERENCES

[1] Educational Data Mining using Improved Apriori Algorithm Jayshree Jha and Leena Ragha, India.

[2] Frequent Item Set Mining Christian Borgelt Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery 2(6):437-456.
J. Wiley & Sons, Chichester, United Kingdom 2012
doi:10.1002/widm.1074 wiley.com

[3] Extraction of Association Rules used for Assessing Web Sites' Quality from a Set of Criteria
Rim Rekik, Ilhem Kallel, Adel M. Alimi REGIM-Lab: Research Groups in Intelligent Machines

[4] APRIORI ALGORITHM AND FILTERED ASSOCIATOR IN ASSOCIATION RULE MINING Himani Bathla, Ms. Kavita Kathuria

[5] The Research of Data Mining Algorithm Based on Association Rules Lei Chen Basic Courses Teaching Department, The Chinese People's Armed Police Force Academy

[6] Research on Association Rule Mining Ziauddin, Shahid Kammal, Khaiuz Zaman Khan, Muhammad Ijaz Khan Gomal University, D.I.Khan (Pakistan)

[7] A Review Paper: A Comparative Analysis on Association Rule Mining Algorithms Gurpreet Singh, Sonia Jassi

[8] Research and Improvement on Association Rule Algorithm Based on FP-Growth Jingbo Yuan Shunli Ding

[9] https://stackabuse.com/association-rule-mining-via-apriori-algorithm-in-python/

[10] https://www.knime.com/blog/market-basket-analysis-and-recommendation-engines