ISSN (Online) 2394-2320

International Journal of Engineering Research in Computer Science and Engineering
(IJERCSE)
Vol 3, Issue 12, December 2016

Overview of Various Frequent Item Set Mining Algorithms of Big Data

[1] Ms. Ann Sara Sajee [2] Ms. Sheryl Saji [3] Ms. Akshada Potdar [4] Mrs. Smita Dange
[1][2][3][4] Department of Computer Engineering,
Fr.. Conceicao Rodrigues Institute of Technology,
Vashi, Navi Mumbai.

Abstract: -- **Market basket analysis (MBA) is an important component of analytical system used in retail organizations. It helps in determining the placement of goods, designing sales promotions for different segments of customers so as to improve customer satisfaction .Thus, increasing the profits of the super market. The transactions can be huge for a supermarket and hence, we have used data analysis technique to get the desired results. It works on frequent item sets to mine data .The frequent item sets are mined from the market basket database (sales records) by applying the efficient algorithms which generates the association rules as output. In this paper, we have discussed how A-priori the most popular MBA algorithm for traditional data set ,is insufficient when applied for big data .We have listed and shown the working of other frequent item set mining algorithm such as PCY and SON that can be used for big data . There are various data mining tools which are available. The various tools will also be compared in this paper.**

Keywords:- **Big Data, Data Mining Market basket analysis (MBA),A-priori, PCY(Park-Chen-Yu),SON(Savasere, Omiecinski and Navathe)..**

I. INTRODUCTION

Market Basket Analysis is one of the most common and useful type of data analysis that is used for marketing and retail. The purpose of market basket analysis is to determine which products customers usually purchased together. It takes its name from the idea of customers throwing all their purchases into a shopping cart (a "market basket") during grocery shopping. In a market basket analysis, you look to see if there are combinations of products that frequently co-occur in transactions. For example, maybe people who buy sugar and milk, also tend to buy coffee powder (because a high proportion of them would want to make coffee). A retailer m,can use this information to improve to store layout (put products that co-occur together close to one another, to improve the customer shopping experience) and marketing (e.g. target customers who buy flour with offers on eggs, to encourage them to spend more on their shopping basket) [10]

The data produced from the Mart will be massive. Hence, implementing Market Basket Analysis using Big Data technology is understandable. Big data is a term for data sets that are so large or complex those traditional data processing applications are inadequate. Challenges include analysis, capture, search, sharing, storage, transfer, visualization, querying, updating and information privacy. The term often refers simply to the use of predictive analytics, user behavior analytics, or certain other advanced data analytics methods that extract value from data, and seldom to a particular size of data set. Accuracy in big data may lead to more confident decision making, and better decisions can result in greater operational efficiency, cost reduction and reduced risk. [14]

II. LITERATURE SURVEY

First we will discuss MBA model the different algorithms.

A. Market Basket Analysis: Basic Terminologies

Items*:-* These are objects that we are identifying associations between. For a retailer, each item is a product in the super mart. A group of items is an item set.

$I=\{i_1,i_2,...,i_n\}I=\{i_1,i_2,...,i_n\}$

Transactions:- These are instances of groups of items cooccurring together. For a retailer, a transaction is, generally, a, transaction. For each transaction, then, we have an item set.
$t_n=\{i_i,i_j,\ldots,i_k\}t_n=\{i_i,i_j,\ldots,i_k\}$

Rules :- These are statements of the form $\{i_1,i_2,\ldots\}\Rightarrow\{i_k\}\{i_1,i_2,\ldots\}\Rightarrow\{i_k\}$ i.e. if you have the items in item set (on the left hand side (LHS) of the rule i.e. $\{i_1, i_2,\ldots\}$), then it is likely that a customer will be interested in the item on the right hand side (RHS i.e. $\{i_k\}$). Consider an example:

The output of a market basket analysis is generally a set of rules that we can then exploit to make business decisions (Related to marketing or product placement, for example).

A-Priori Algorithm [7]

A-Priori algorithm is a level-wise, breadth-first algorithm in which transaction. Are been counted. This algorithm uses prior knowledge of frequent item set properties. A-Priori uses an iterative approach known as a level-wise search, in which n-item sets are used to explore (n+1)-item sets. The A-priori algorithm is designed to reduce the number of pairs that must be counted, at the expense of performing two passes over data, rather than one pass.

A-Priori Algorithm--- Pass 1

First, we create two tables in which first table converts the item names into integers from 1 to n. The other table is an array of counts, the i_{th} array element counts the occurrences of the numbered item i. Read baskets, we look at each item in the basket and translate its name into integer next, we use that integer to refer the array of counts and add 1to the integer found there.

A-Priori Algorithm--- Pass 2

Read baskets again and count in main memory only those pairs both of which were found in Pass 1 to be frequent. In a double loop, generate all frequent pairs Requires memory proportional to square of frequent items only (for counts), plus a list of the frequent items (so you know what must be counted.

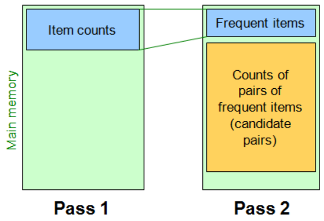


Fig.2. Shows the memory utilization in A-priori algorithm during Pass 1 and Pass 2.

A-Priori for All Frequent Item sets

For each k, we construct two sets of k -sets (sets of size k): C_k = candidate k -sets = those that might be frequent sets (Support > s) based on information from the pass for k –1. L_k = the set of truly frequent k -sets.

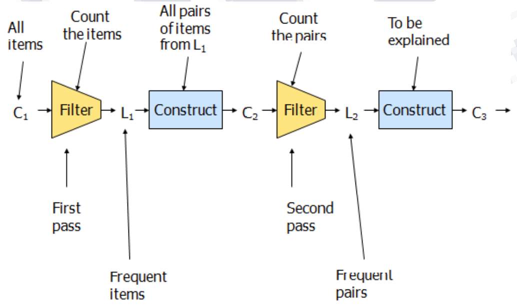


Fig 3. Shows the flow of events during A-priori Algorithm One pass for each k.

Needs room in main memory to count each candidate k -set. For typical market-basket data and reasonable support (e.g., 1%), k = 2 requires the most memory.
C1 = all items
In general, Lk = members of Ck with support ≥ s.
Ck +1 = (k +1) -sets, each k of which is in Lk

PCY Algorithm [2]

It is a Hash-based improvement to A-Priori developed by Park, Chen and Yu. Hence, it is called PCY

Algorithm:

During Pass 1 of A-priori, most memory is idle. Use that memory to keep counts of buckets into which pairs of items are hashed. Just the count, not the pairs themselves. Gives extra condition that candidate pairs must satisfy on Pass 2.

PCY Algorithm --- Before Pass 1 Organize Main Memory: Space to count each item. One (typically) 4-byte integer per item. Use the rest of the space for as many integers, representing buckets, as we can.

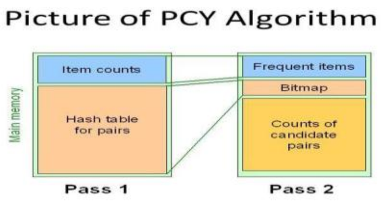


Fig 4. The memory utilization in PCY algorithm in the two passes.

PCY Algorithm --- Pass 1

FOR (each basket) {FOR (each item) Add 1 to item's count; FOR (each pair of items) { Hash the pair to a bucket; Add 1 to the count for that bucket;

PCY Algorithm --- Between Passes Replace the buckets by a bit-vector: 1 means the bucket count ≥ the support s (frequent bucket); 0 means it did not. Integers are replaced by bits, so the bit vector requires little second-pass space. Also, decide which items are frequent and list them for the second pass.

PCY Algorithm --- Pass 2

Count all pairs {i,j } that meet the conditions: Both i and j are frequent items. The pair {i,j}, hashes to a bucket number whose bit in the bit vector is 1. Notice all these conditions are necessary for the pair to have a chance of being frequent.

Observations about Buckets

1. If a bucket contains a frequent pair, then the bucket is surely frequent. We cannot use the hash table to eliminate any member of this bucket.

2. Even without any frequent pair, a bucket can be frequent. Again, nothing in the bucket can be eliminated.

3. But in the best case, the count for a bucket is less than the support s. Now, all pairs that hash to this bucket can be eliminated as candidates, even if the pair consists of two frequent items.

3) SON ALGORITHM

Ashok Savasere, Edward Omiecinski, Shamkant Navathe discovered the SON algorithm .It works on the partition, that is fundamentally different from all the previous algorithms .This algorithm has two map-reduce function .The map-reduce function finds the frequent item sets and mines it in the memory .

Algorithm [16]:

```
The first map-reduce Map(key,value);
Count occurrence of item in the dataset //frequent itemsets are mined //
For itemset in the itemsets
If sup(itemset) //
the support value has to be compared //
Emit(itemset,null)
Reduce(key,value) //
those items are below the threshold s
value are removed//
Emit(key, null)
The second map –reduce step
Map(key,value)
Count occurrence of item in the dataset
For itemset in the itemsets
Emit(itemset,sup(itemset))
Reduce(key,values) // the key value is also considered//
Result=0
For value in values
Result+=values
If Result>=s
Emit(key,values)
```

III. CASE STUDY

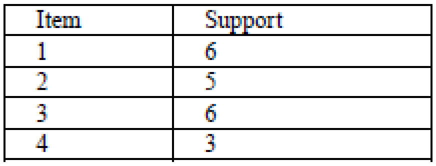
For the given - Items: Milk (1), Coke (2), Bread (3), Pepsi (4) with support =3 .apply A-priori, PCY and SON algorithms.

Transactions are:-

T1= {1, 2, 3} T2= {1, 4, 3} T3= {1, 3}
T4= {2, 4} T5= {1, 3, 4} T6= {1, 2, 3}
T7= {2, 3} T8= {1,2}

1) A-PRIORI ALGORITHM

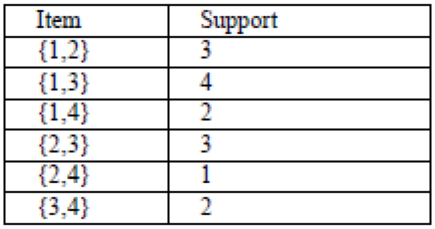
Step 1:



Item	Support
1	6
2	5
3	6
4	3

We can define a minimum support level to qualify as "frequent," which depends on the context. For this case, let min support = 3. Therefore, all are frequent. The next step is to generate a list of all 2-pairs of the frequent items. Had any of the above items not been frequent, they wouldn't have been included as a possible member of possible 2-item pairs. In this way, A-priori prunes the tree of all possible sets. In next step we again select only these items (now 2-pairs are items) which are frequent (the pairs written in bold text):

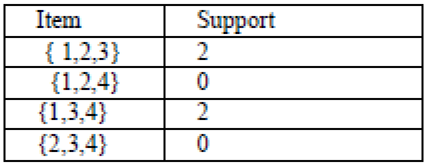
Step 2:



Item	Support
{1,2}	3
{1,3}	4
{1,4}	2
{2,3}	3
{2,4}	1
{3,4}	2

We can define a minimum support level to qualify as "frequent," which depends on the context. For this case, let min support = 1.We generate the list of all 3-triples of the frequent items (by connecting frequent pair with frequent single item).

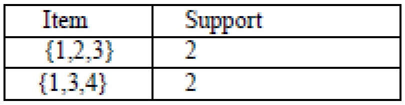
Step3:



Item	Support
{ 1,2,3}	2
{1,2,4}	0
{1,3,4}	2
{2,3,4}	0

The algorithm will end here because the pair {1,2,4} and {2,3,4} generated at the previous step does not have the desired support. So we will now apply the same algorithm on the same set of data considering that the min support is 2. We get the following results.

Step 4:

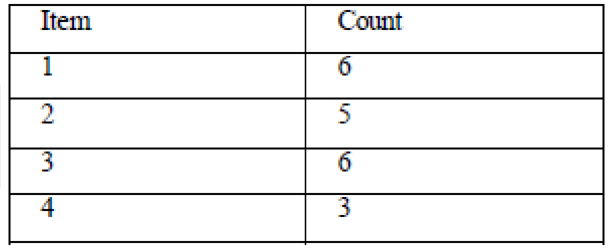


Item	Support
{1,2,3}	2
{1,3,4}	2

Thus these are the frequent items that can be paired together.

2) PCY ALGORITHM

Step 1: Hashing of a pair {i,j} to a bucket k, where
k =hash (i,j)=(i+j)mod 4. This is, for pairs:
(1,3) -> k=0 (1,4) and (2,3) -> k=1
(2,4) -> k=2 (1,2) and (3,4) -> k=3
PASS 1: Item's Count



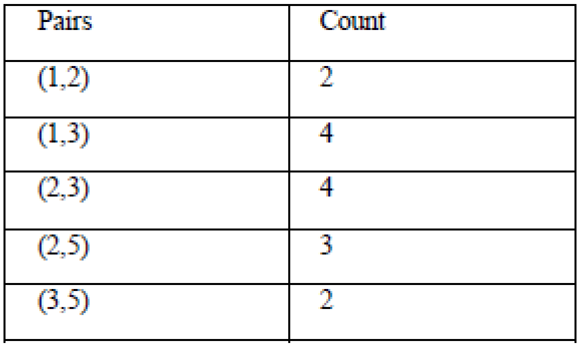
Item	Count
1	6
2	5
3	6
4	3

Step 2 :

Since, Support=4, We observe that the Item 4 does not exceed support. For each pair in each transaction:
T1= (1, 2)3 (2, 3)1 (1, 3)0
T2= (1, 4)1 (1, 3)0
T3= (1, 3)0
T4= (2, 4)2
T5= (1, 3)0 (1, 4)1
T6= (1,2)3 (1,3)0
T7= (2, 3)1
T8= (1, 2)3
Hash Table is:-
Bucket 2 and 3 does not exceed the support, i.e. (2, 4), (1,2) and (3,4) are not frequent.

Step 3:

PASS 2: Frequent items are {1, 2, 3}.From the frequent items the candidate pairs are (1, 3) (1,4) and (2, 3). Candidate (1,2), (2,4) and (3,4)are discarded because Bucket 2 and 3are not frequent -> Discarded by PCY Now let's count the "surviving" pairs-

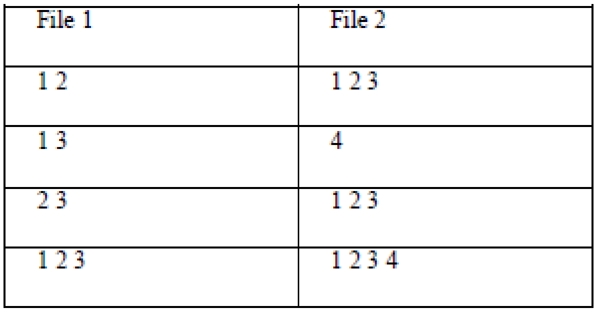


Pairs	Count
(1,2)	2
(1,3)	4
(2,3)	4
(2,5)	3
(3,5)	2

3)SON ALGORITHM

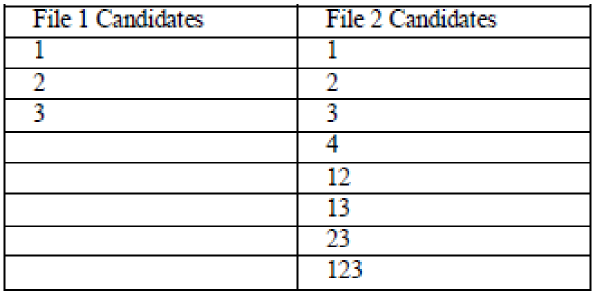
Step 1 – Split the table Here we check if the files are being split up correctly. We set, k = 2 i.e. file into two equal sized chunks with 4 items in each. Thus we expect two files containing items:

There exist two sub files composed of these baskets.



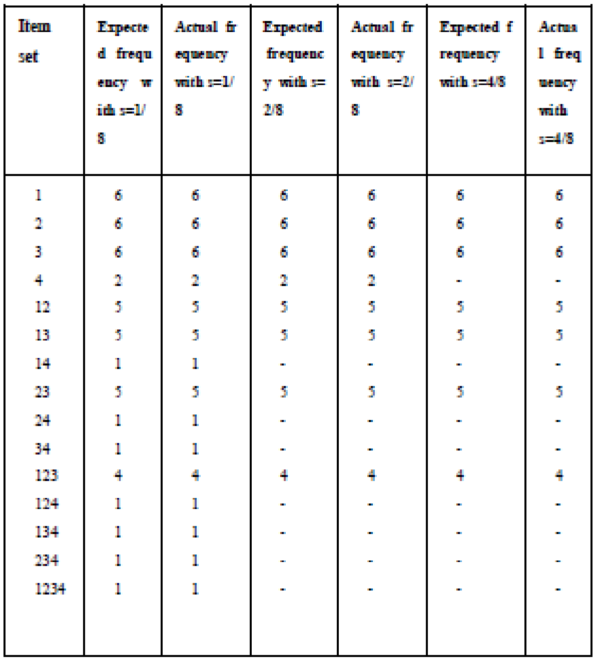
File 1	File 2
1 2	1 2 3
1 3	4
2 3	1 2 3
1 2 3	1 2 3 4

Step 2 - Candidate frequent item set check Here we check that the expected candidate item sets are found in each mapper. We set, k = 2 to split the sub files as described above and set the support threshold . Each mapper s = 5/8 receives baskets, were n is the total number of lines in the file. Thus within a mapper we lower the local threshold to slow=s(n/k) =(5/8)(8/2) =2/5 . Thus for an item set to be considered frequent, it must occur at least 3 times within the sub file. Thus we would expec t the candidate itemsets to be as follows.



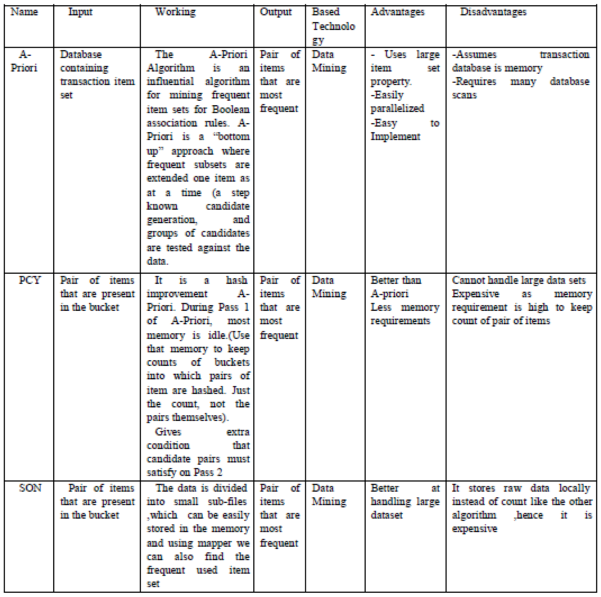
File 1 Candidates	File 2 Candidates
1	1
2	2
3	3
	4
	12
	13
	23
	123

Step3: Check the pipeline Here we try our entire method with different support threshold s. We also specify different k values of k = {1,2,4} and verify that the results do not change.



Item set	Expected frequency with s=1/8	Actual frequency with s=1/8	Expected frequency with s=2/8	Actual frequency with s=2/8	Expected frequency with s=4/8	Actual frequency with s=4/8
1	6	6	6	6	6	6
2	6	6	6	6	6	6
3	6	6	6	6	6	6
4	2	2	2	2	-	-
12	5	5	5	5	5	5
13	5	5	5	5	5	5
14	1	1	-	-	-	-
23	5	5	5	5	5	5
24	1	1	-	-	-	-
34	1	1	-	-	-	-
123	4	4	4	4	4	4
124	1	1	-	-	-	-
134	1	1	-	-	-	-
234	1	1	-	-	-	-
1234	1	1	-	-	-	-

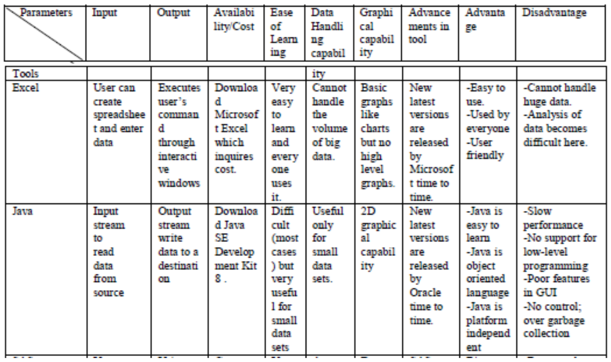
IV. COMPARISION BETWEEN VARIOUS FREQUENT ITEM SET MINING ALGORITHMS

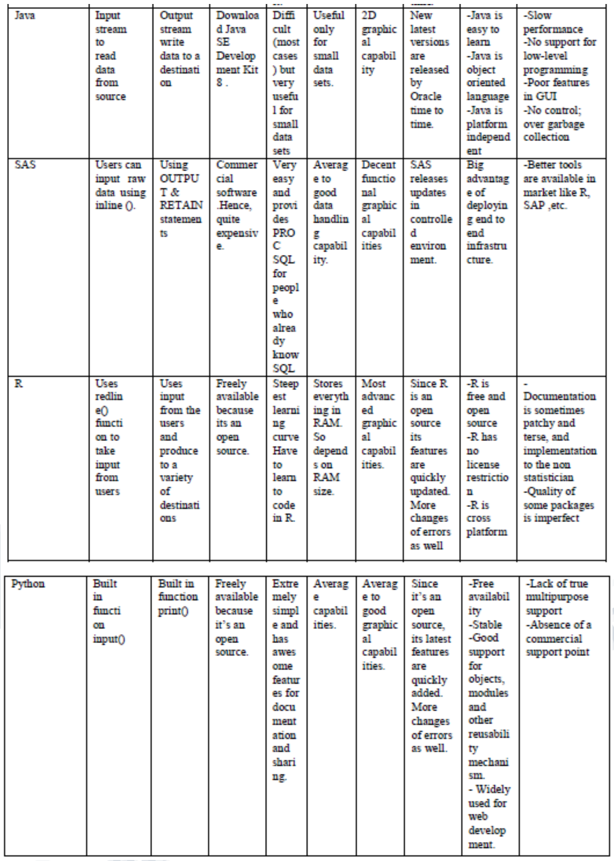


Name	Input	Working	Output	Based Technology	Advantages	Disadvantages
A-Priori	Database containing transaction item set	The A-Priori Algorithm is an influential algorithm for mining frequent item sets for Boolean association rules. A-Priori is a "bottom up" approach where frequent subsets are extended one item as at a time (a step known candidate generation, and groups of candidates are tested against the data.	Pair of items that are most frequent	Data Mining	- Uses large item set property. -Easily parallelized -Easy to Implement	-Assumes transaction database is memory -Requires many database scans
PCY	Pair of items that are present in the bucket	It is a hash improvement A-Priori. During Pass 1 of A-Priori, most memory is idle.(Use that memory to keep counts of buckets into which pairs of item are hashed. Just the count, not the pairs themselves). Gives extra condition that candidate pairs must satisfy on Pass 2	Pair of items that are most frequent	Data Mining	Better than A-priori Less memory requirements	Cannot handle large data sets Expensive as memory requirement is high to keep count of pair of items
SON	Pair of items that are present in the bucket	The data is divided into small sub-files ,which can be easily stored in the memory and using mapper we can also find the frequent used item set	Pair of items that are most frequent	Data Mining	Better at handling large dataset	It stores raw data locally instead of count like the other algorithm ,hence it is expensive

V. IMPORTANT DATA MINING USING TOOLS COMPARISON

There are many tools that are available for data mining, some of which are listed below. These tools are good use in their own way but we have done a comparative study to understand which data mining tool can be best used to analyze big data sets. The comparison of some tools like- R, Python, SAS, Excel etc. are done below. [4]





Parameters Tools	Input	Output	Availability/Cost	Ease of Learning	Data Handling capability	Graphical capability	Advancements in tool	Advantage	Disadvantage
Excel	User can create spreadsheet and enter data	Executes user's command through interactive windows	Download Microsoft Excel which inquires cost.	Very easy to learn and every one uses it.	Cannot handle the volume of big data.	Basic graphs like charts but no high level graphs.	New latest versions are released by Microsoft time to time.	-Easy to use. -Used by everyone -User friendly	-Cannot handle huge data. -Analysis of data becomes difficult here.
Java	Input stream to read data from source	Output stream write data to a destination	Download Java SE Development Kit 8 .	Difficult (most cases) but very useful for small data sets	Useful only for small data sets.	2D graphical capability	New latest versions are released by Oracle time to time.	-Java is easy to learn -Java is object oriented language -Java is platform independent	-Slow performance -No support for low-level programming -Poor features in GUI -No control; over garbage collection
Java	Input stream to read data from source	Output stream write data to a destination	Download Java SE Development Kit 8 .	Difficult (most cases) but very useful for small data sets	Useful only for small data sets.	2D graphical capability	New latest versions are released by Oracle time to time.	-Java is easy to learn -Java is object oriented language -Java is platform independent	-Slow performance -No support for low-level programming -Poor features in GUI -No control; over garbage collection
SAS	Users can input raw data using inline ().	Using OUTPUT & RETAIN statements	Commercial software .Hence, quite expensive.	Very easy and provides PROC SQL for people who already know SQL	Average to good data handling capability.	Decent functional graphical capabilities	SAS releases updates in controlled environment.	Big advantage of deploying end to end infrastructure.	-Better tools are available in market like R, SAP ,etc.
R	Uses redline() function to take input from users	Uses input from the users and produce to a variety of destinations	Freely available because its an open source.	Steepest learning curve Have to learn to code in R.	Stores everything in RAM. So depends on RAM size.	Most advanced graphical capabilities.	Since R is an open source its features are quickly updated. More changes of errors as well	-R is free and open source -R has no license restriction -R is cross platform	-Documentation is sometimes patchy and terse, and implementation to the non statistician -Quality of some packages is imperfect
Python	Built in function input()	Built in function print()	Freely available because it's an open source.	Extremely simple and has awesome features for documentation and sharing.	Average capabilities.	Average to good graphical capabilities.	Since it's an open source, its latest features are quickly added. More changes of errors as well.	-Free availability -Stable -Good support for objects, modules and other reusability mechanism. - Widely used for web development.	-Lack of true multipurpose support -Absence of a commercial support point

VI. CONCLUSION

Data Analysis is a very vast and interesting topic which can be used along with different technologies to analyze the data efficiently and accurately .Data mining brings a lot of benefits to businesses, society, governments as well as the individual.. However, when huge data comes into picture, it is more efficient to use Big Data analysis using a tool that is very effective to analyze. Out of all the method in which data mining can take place, we found R to be very easy, user friendly and effective when large data has to be analyzed. Thus, we will be using R for implementation in our further analysis.

REFERENCES

[1]"http://www.vldb.org/conf/1995/P432.PDF"www.vldb.org/conf/1995/P432 PDF

[2]Hash-Based Improvements to A-Priori – Stanford InfoLab infolab.stanford. edu/~ullman/mining/pdf/assocrules2. Pdf

[3]www.cs.nthu.edu.tw/~dr824349/personal/survey/anti- skew%20ICDE98.pd

[4]http://www.analyticsvidhya.com/blog/2014/03/sas-vs-vs-pythontool-learn/

[5]"http://www.revolutionanalytics.com/what -r"r

[6]https://www.researchgate.net/topic/market_basket_analysis

[7] Book- Mining of Massive Data Sets by- Anand Rajaram and Jeffrey David Ullman

[8] Association rules The goal of mining association rules is to generate all possible rules that exceed some minimum userspecifiedsupport and confidence http://slideplayer.com/slide/4463546/

[9]http://software.ucv.ro/~cmihaescu/ro/teaching/AIR/docs/Lab8 - Apriori.pdf

[10]http://snowplowanalytics.com/guides/recipes/catalog analytics/market-basket-analysis-identifying-products-that-sell-welltogether.html#apriori

[11]www.sfu.ca/~cjbrown/pdfs/cmpt741_proj_hadoop.pdf

[12]https://en.wikipedia.org/wiki/R_(programming_language)

[13]https://en.wikipedia.org/wiki/Big_data

[14]http://www.burns-stat.com/documents/tutorials/why-use- the-rlanguage/

[15] Advantages and Disadvantages of Data Mining – ZenTut www.zentut.com › Data Mining

[16]Big Data Analytics by Radha Shankarmani,M.Vijayalakshmi