FREQUENT DATA MINING REPORT

Nitin K (2017csb1093)

05.10.2020
CSE FINAL YEAR, UG
IIT ROPAR

INTRODUCTION

This is a report for different frequent data Mining Algorithms. Here Three Algorithms (Apriori, FPTree, Eclat) are explained how they behave on the different dataset.

Algorithms Details:

I have Taken All the Algorithms from the books. That is why I am providing the reference only as the algorithm has already been discussed in the class.

Apriori

Reference book for the Algorithm is **Data Mining Concept And Techniques, Jiawei Han and Micheline Kamber.**

FPTree

Reference book for the Algorithm is **Data Mining Concept And Techniques, Jiawei Han and Micheline Kamber.** Here I have modified the algorithm to some instant not much. I have not created the ordered Table and instead of that I have Use hashing for ordering during old transaction traversal.

Eclat

Reference book for the Algorithm is **Data Mining and Analysis: Fundamental Concepts and Algorithms, Mohammed J. Zaki and Wagner Meira Jr.**

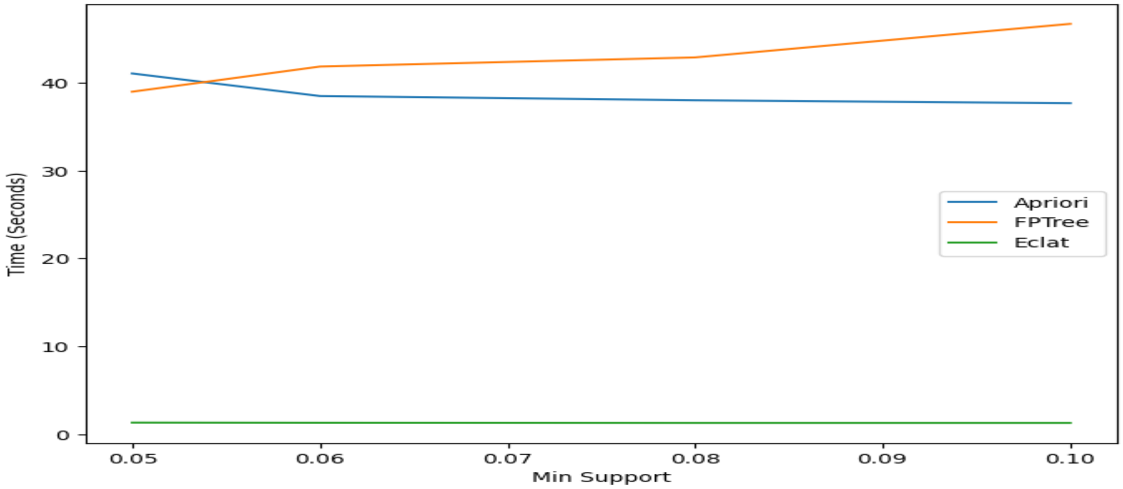
DATA

DataSet Descriptions

DataSet No	**Data1**	**Data2**	**Data3**	**Data4**
Total Transaction	100000	100000	88162	9835
Total Items	870	942	16470	169
Avg. Transaction Width	10.1	39.61	10.31	4.41
FreqItemSize(MinSup=0.05)	1	2	3	2
FreqItemSize(MinSup=0.06)	1	2	3	2
FreqItemSize(MinSup=0.08)	0	1	3	1
FreqItemSize(MinSup=0.1)	0	1	2	1
NoOfFreqItem(MinSup=0.05)	10	15	3	3
NoOfFreqItem(MinSup=0.06)	4	5	3	1
NoOfFreqItem(MinSup=0.08)	0	137	1	13
NoOfFreqItem(MinSup=0.1)	0	82	4	8
FreqItemSize(MinSup=0.005)	5	18	5	4
NoOfFreqItem(MinSup=0.005)	9	2	1	12

RESULTS

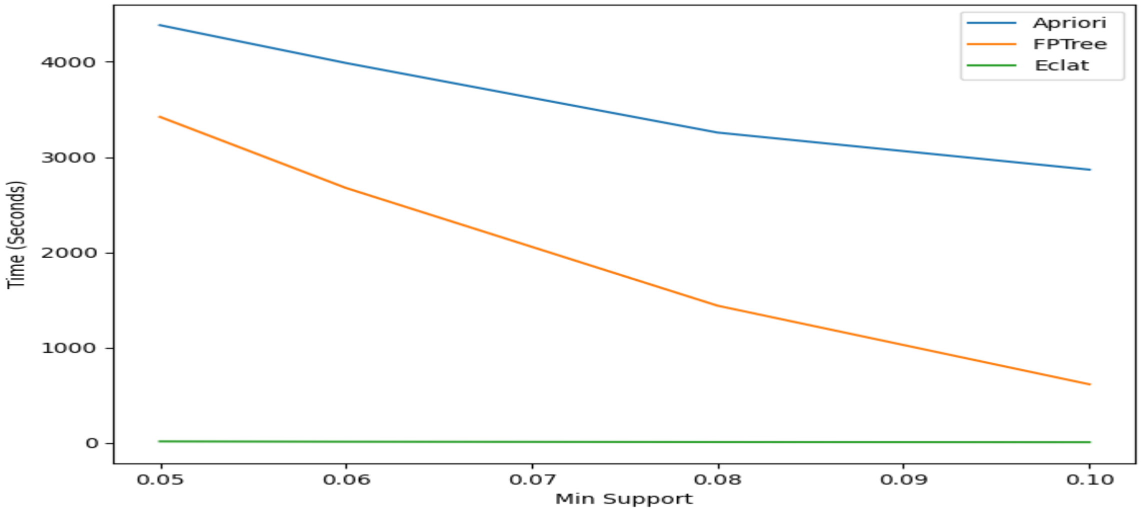
1. DataSet-1



From the Above graph It is clear that if we increase Min Support, Time Decreases Because size of frequent set decreases. Time for FPTree is increasing as per **graph** but actually I have taken such Min support to run fast which does not allow it to create conditional FPTree for **support** > **0.05** and I have Taken support greater than 0.05 that is why we have only frequent item on 1 size and number of 1 size frequent is decreasing with increase in support so we have to traverse more to filter the item which increasing the time.

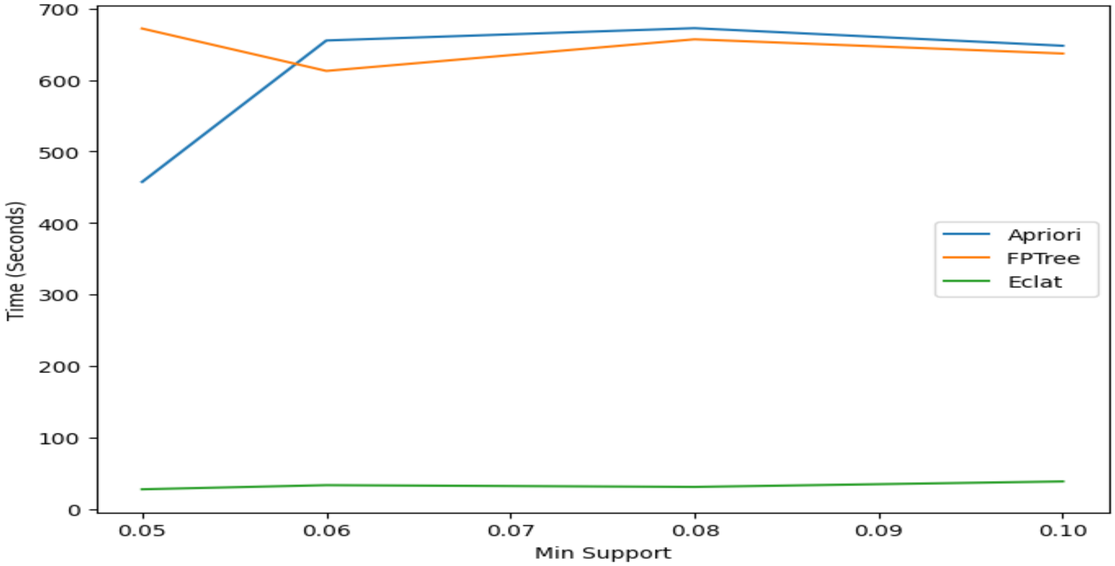
But If We take Support less < 0.05 for DataSet 1 it would so correct the result. The time for Apriori and FPtree is similar but much higher than Eclat.

2. DataSet 2



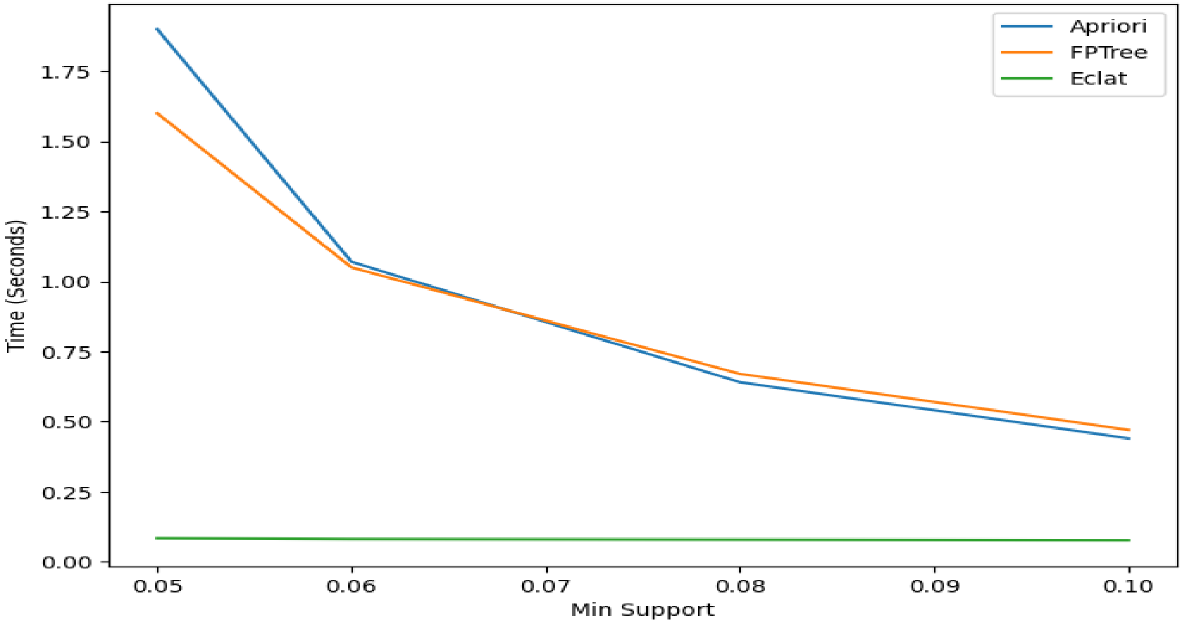
In this graph there is a good result that shows How the time is decreasing with increase in min support. The time for Apriori is much higher than FPTree and Eclat. The time for FPtree is decreasing rapidly with respect to min support increase.

3. DataSet 3



In this graph there is a good result that shows How the time is decreasing with increase in min support. But time for Apriori and Eclat is similar but much higher than Eclat.

4. DataSet 4



In this graph there is a good result that shows How the time is decreasing with increase in min support. The time is decreasing in a similar way for both Apriori and FPTree. But they are much higher than Eclat.

Memory Used

DataSet No	Data1	Data2	Data3	Data4
Apriori(Memory In Bytes)	140148736	-	134725632	64778240
FPtree(Memory In Bytes)	139939840	-	133152768	67051520
Eclat(Memory In Bytes)	141103104	356741120	147656704	65265664

CONCLUSION

Lets First Compare for The DataSet-1 and DataSet2. Both the dataSet Contains Same Number of Transaction but their Avg Transaction width is different.

Avg Data 2 > Data 1

For Apriori:

The time in DataSet 2 is higher than DataSet 1 because comparison(Finding a Frequent Items Set) in a particular transaction gets increased because of Increase in avg width.

For FPTree:

The time in DataSet 2 is higher than DataSet 1 because Height of FPTree gets increased because of the increase in avg width which increases the tree traversal time.

For Eclat:

It has less effect on run time because we are traversing in vertical manner and the difference between avg width is not much hence it is working very fast on both the data Set.

The Same Above comparison can be applied for DataSet2 and DataSet3 as DataSet3 has much total items than dataset 2 but dataSet3 has less transaction than dataSet2 but avg width is much more dominating than transaction because dataSet 2 has much total items than dataSet3

DataSet1 and DataSet3:

DataSet1 has less total item than dataSet 3 that is why dataset 3 is taking more run time.

In Apriori , Increases in items cause an **increase in width of the prefix tree** which results in an increase in time. In FPTree, Increases in items cause The tree more **Scattered** which results in increase in time. It has less effect on Eclat because of its vertical order

DataSet 1 , DataSet2 and DataSet3 has higher transaction than DataSet4 that is why all the algorithm on the dataSet 4 are running faster than other dataSet4

From Above all the observation We found that **Eclat is much faster** than Apriori and FPTree.