Comparing Dataset Characteristics that Favor the Apriori, Eclat or FP-Growth Frequent Itemset Mining Algorithms

Jeff Heaton

Nova Southeastern University - Ft. Lauderdale, FL USA
Reinsurance Group of America (RGA) - St. Louis, MO USA

April 1, 2016

http://www.jeffheaton.com/

What this Talk About?

- **Frequent Itemset Mining** - Determine what items frequently go together. Used by companies such as Amazon to suggest sales.
- **Items** - The individual discrete components of a basket/transaction.
- **Basket/Transaction** - A variable length collection of items than become the rows in the dataset. Databases (lists of baskets) can become very large ("Big Data").
- **Performance** - For the purposes of this research, performance is measured by the amount of time the algorithm takes to process a dataset (wall clock).

Outline

1. Research Objective
2. Related Work
3. Algorithm Survey
4. Algorithm Survey
5. Research Methodology
   - Summary of Results
6. Future Work

Motivation and Problem Statement

What dictates the choice of frequent itemset algorithm? Are different algorithms better suited for different dataset characteristics? Create reusable script to evaluate two dataset characteristics:

- **Basket Size** - What is the average basket size?
- **Dataset Density** - How often do frequent itemsets occur?

Measure the execution time of Apriori, Eclat, and FP-Growth as these two dataset characteristics are varied.

Related Work

Other similar research in frequent itemset research.

- **Seminal Research**
  - Apriori Algorithm (Agrawal & Srikant, 1994)
  - Eclat Algorithm (Zaki, Parthasarathy, Ogihara, & Li 1997)
  - FP-Growth Algorithm (Han, Pei, & Yin 2000)
- **Dataset Generation**
  - IBM Quest Synthetic Data Generator (Pitman, 2011)
- **Frequent Itemset Software** (used in this paper)
  - Efficient Implementations of Apriori and Eclat (Borgelt, 2003)
  - An Implementation of the FP-growth Algorithm (Borgelt, 2005)
- *Others given in the conference paper...*

Algorithm Survey

Toy dataset.

Frequent itemset mining looks for groups of items that frequently occur together in datasets.

- The following toy database:

```
[mp3player usb-charger book-dct book-ths]
[mp3player usb-charger]
[usb-charger mp3player book-dct book-ths]
[usb-charger]
[book-dct book-ths]
```

- Might yield the following frequent itemsets:

```
[mp3player usb-charger]
[book-dct book-ths]
...
```

Algorithm Survey

Toy dataset.

Calculating support of an itemset:

$$\text{supp}(X) = \frac{Xcount}{N} \tag{1}$$

This equation can be applied to calculate the support for {mp3-player usb-charger} from the previously presented set of baskets.

$$\text{supp}(\{\text{mp3-player usb-charger}\}) = \frac{3}{5} = 0.6 \tag{2}$$

The support statistic of 0.6 indicates that 60% of the five baskets contain the candidate itemset {mp3-player usb-charger}. Most frequent itemset algorithms accept a minimum support parameter to filter out less common itemsets.

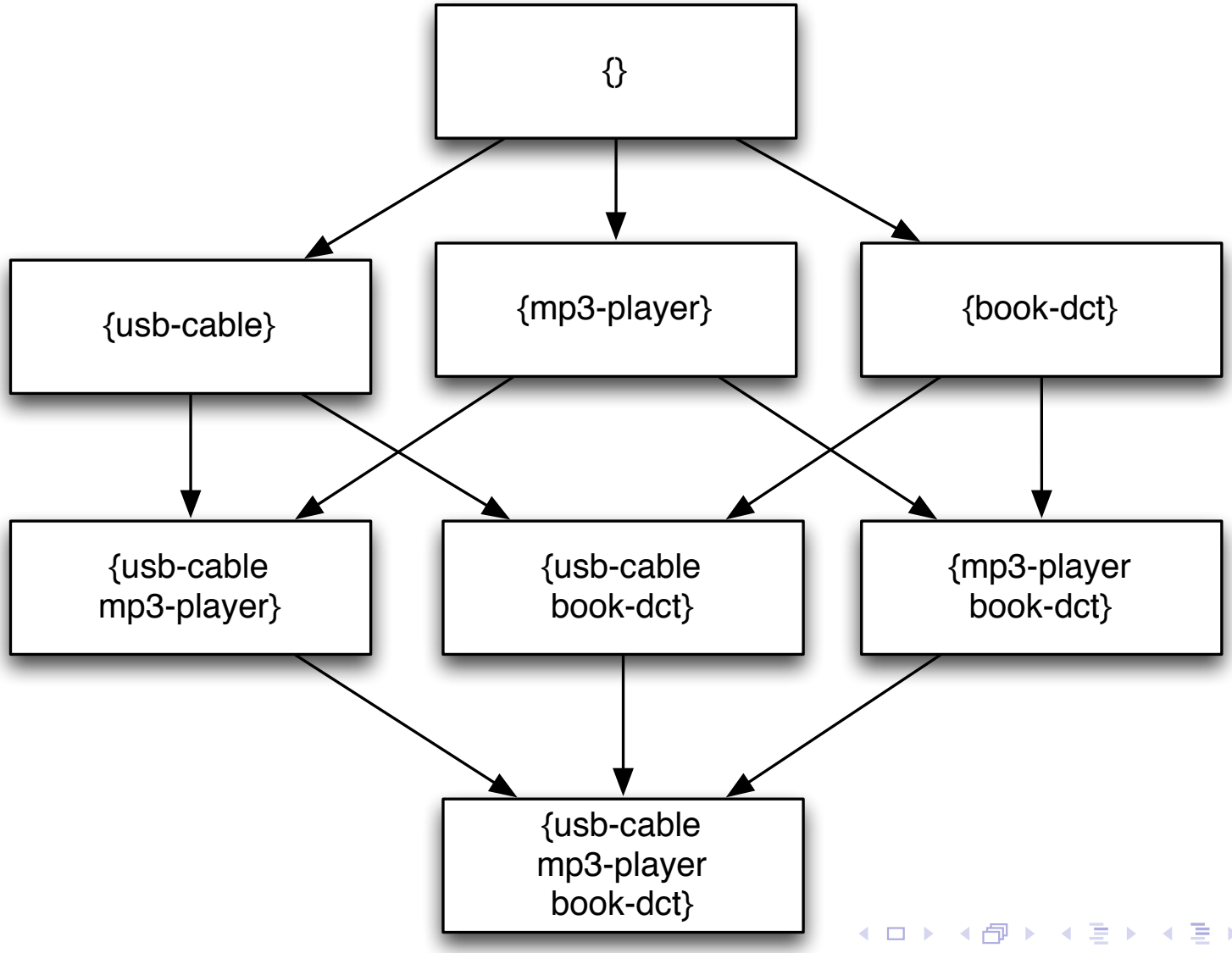
Apriori Algorithm

Overview

- Apriori is the most popular frequent itemset mining algorithm.
- Based on the hierarchical monotonicity of frequent itemsets between their supersets and subsets.
- Aprioiri first builds a list of all singleton itemsets with sufficient support.
- Building on the monotonicity principle, the next set of candidate frequent itemsets is built of combinations of the singleton itemsets.
- This process continues until the maximum length specified for frequent itemsets is reached.
- Aprori is a horizontal, breadth-first, algorithm.

Apriori Algorithm

Item Lattice



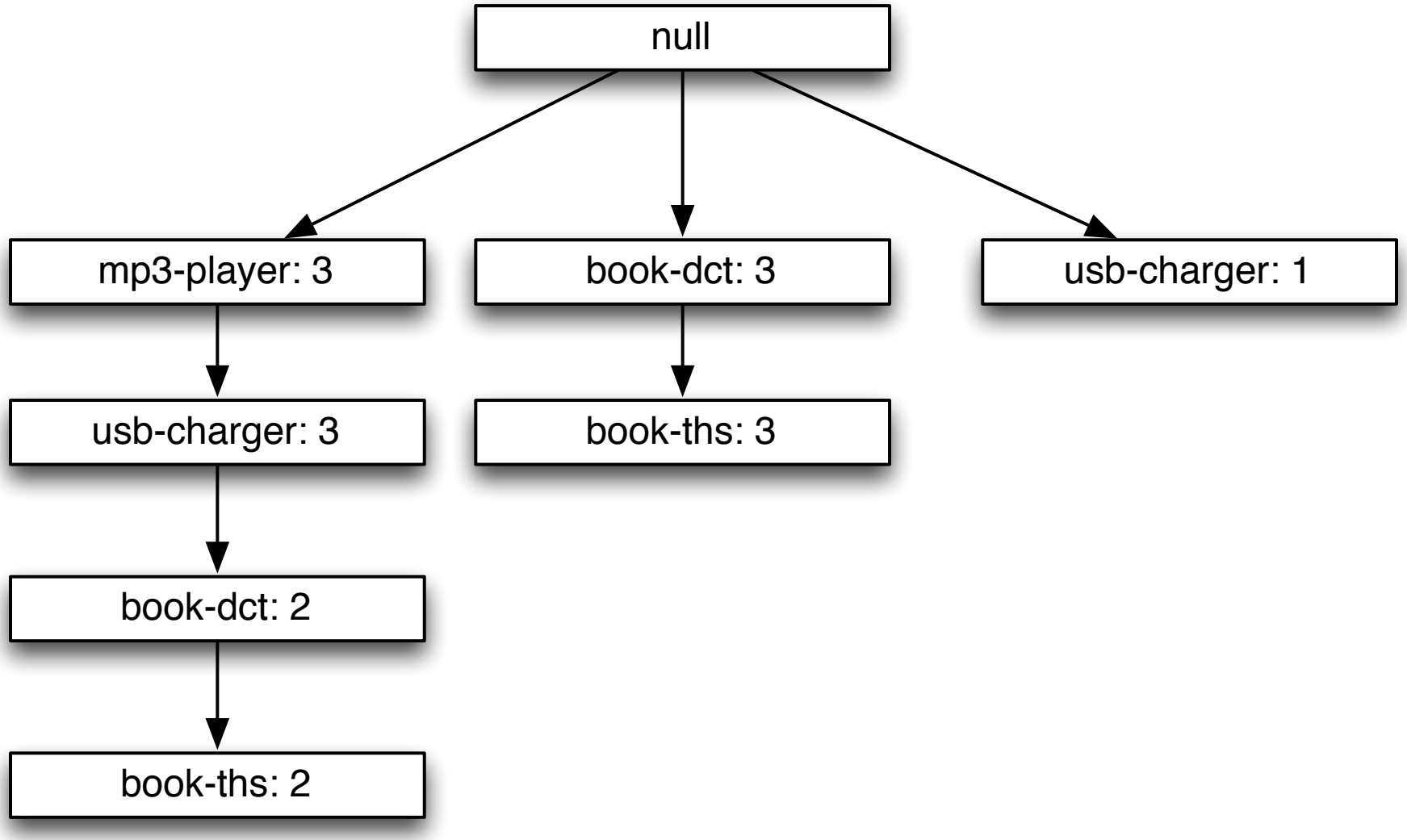
Eclat Algorithm

Overview

- The primary difference between Eclat and Apriori is that Eclat abandons Apriori's breadth-first search for a recursive depth-first search.
- Support values are stored in a structure called a trie. Start with empty root node. Add node for each item in the set, starting at the left. No downward traversal should encounter the same item.
- The trie allows quick lookup of the support values.
- Eclat is a vertical, depth-first, algorithm.

Eclat Algorithm

Eclat Trie



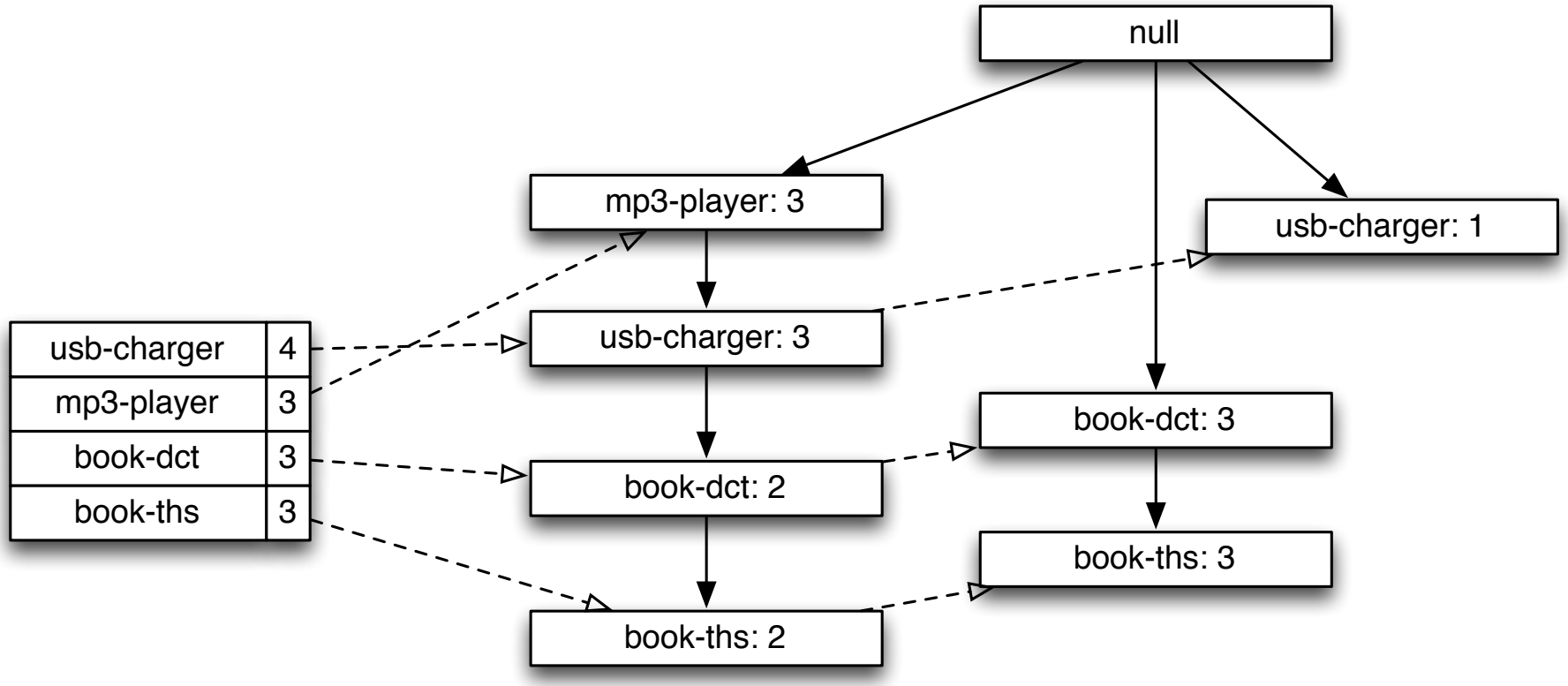
FP-Growth Algorithm

Overview

- FP-Growth was introduced to forego candidate generation altogether.
- This is done by using a trie to store the actual baskets, rather than storing candidates like Apriori and Eclat do.
- FP-Growth provides both vertical and horizontal access to the data.

FP-Growth Algorithm

Horizontal and Vertical Access



Generated dataset files.

Toy dataset.

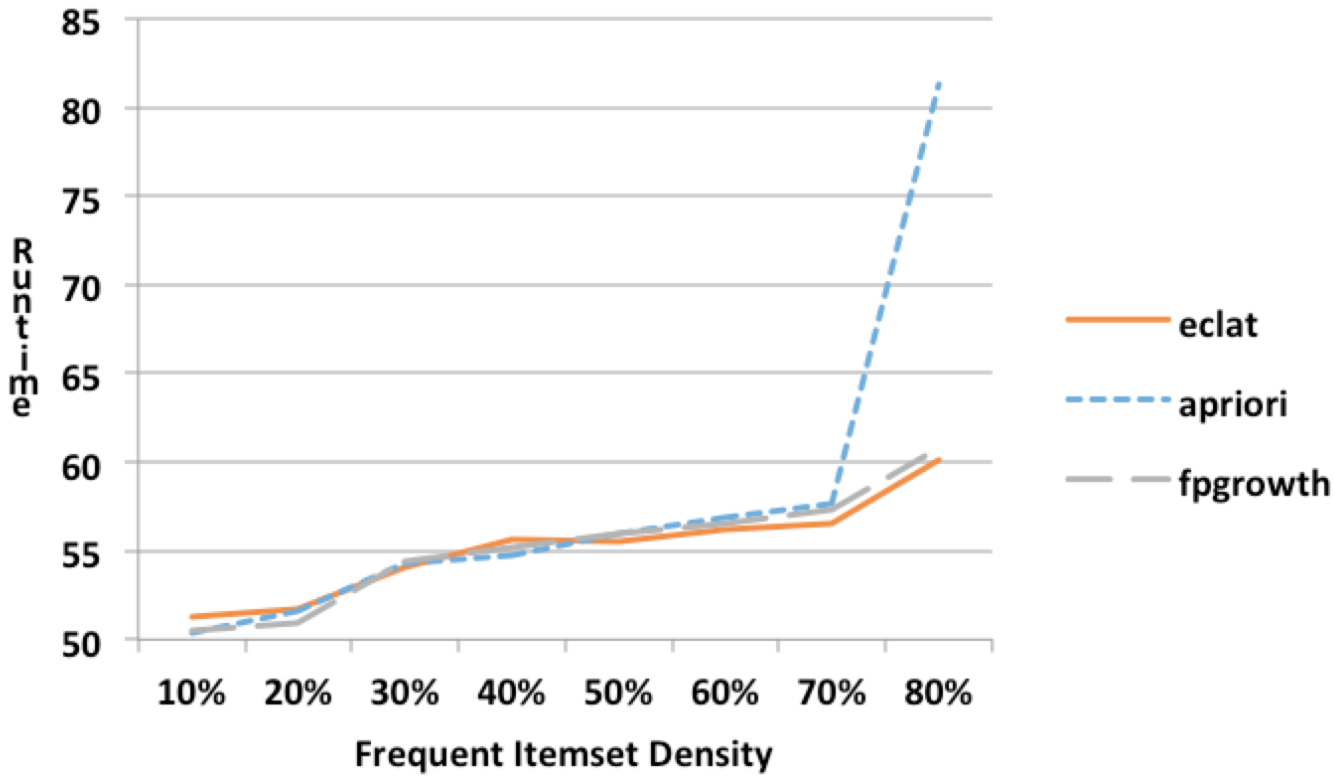
- Transaction/Basket count: 10 million default
- Number of items: 50,000 default
- Number of frequent sets: 100 default
- Max transaction/basket size: independent variable, 5-100 range
- Frequent set density: independent variable, 0.1 to 0.8 range

---

```
I36 I94
I71 I13 I91 I89 I34
F6 F5 F3 F4
I39 I16 I49 I62 I31 I54 I91
I22 I31
I70 I85 I78 I63
F4 F3 F1 F6 F0 I69 I44
...
```

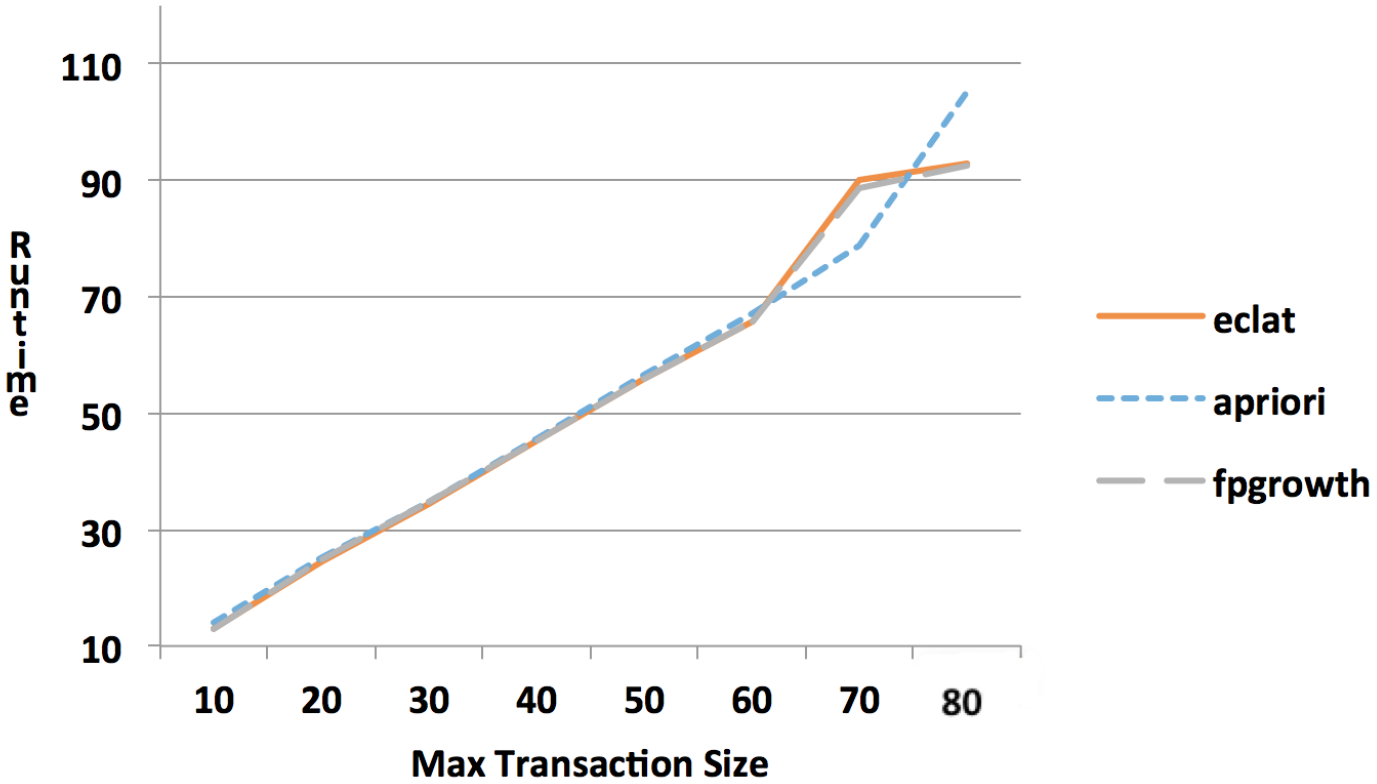
Experiment Results

Frequent Itemset Density



Experiment Results

Basket Size



Summary of Results

- Apriori is an easily understandable frequent itemset mining algorithm.
- However, Apriori has serious scalability issues.
- Most frequent itemset applications should consider using either FP-Growth or Eclat.
- Similar Eclat and FP-Growth performance, though FP-Growth did show slightly better performance than Eclat.

Future Work

- Continue to extend the dataset generator
- Try frequent itemset algorithms other than Apriori, Eclat and FP-Growth
- Experiment with other dataset characteristics that might influence algorithm performance.