



AMAZON CO-PURCHASE NETWORK

ANALYSING PRODUCT PURCHASE PATTERN IN AMAZON COPURCHASE
NETWORK

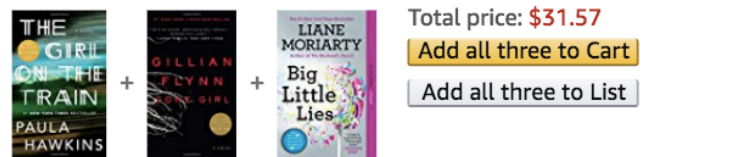


SUBMITTED BY
SWAPNIL PARKHE, ASHIMA HORRA AND VINAYAK KUDVA

Introduction

In recent times, online shopping has become popular among customers. In this period of time, along with sustaining a business model, online retailers have also generated a colossal amount of profit, converting it into a flourishing business. For this project, we have analyzed a temporal network dataset from Amazon; perhaps the most popular and successful online retailers to study patterns in which customers purchase products. The most important feature of such a network is the next product the customer purchases, which is also known as the co-purchase.

Frequently bought together

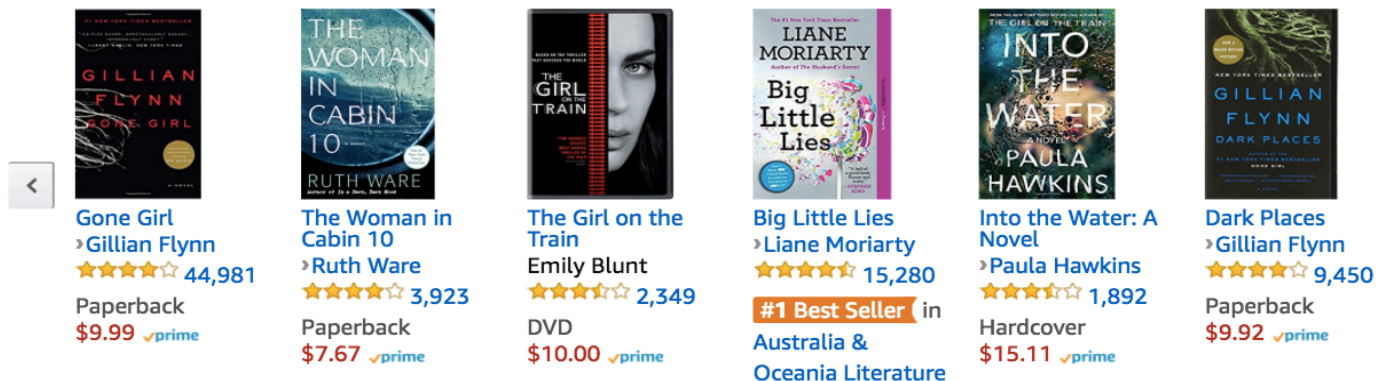


Total price: **\$31.57**
Add all three to Cart
Add all three to List

- ☑ **This item:** The Girl on the Train by Paula Hawkins Paperback **\$8.80**
- ☑ **Gone Girl** by Gillian Flynn Paperback **\$9.99**
- ☑ **Big Little Lies** by Liane Moriarty Paperback **\$12.78**

Fig. 1: Example of Amazon suggesting products representing plausible co-purchases

Customers who bought this item also bought



Book Title	Author	Rating	Price	Format
Gone Girl	Gillian Flynn	★★★★☆ 44,981	\$9.99	Paperback
The Woman in Cabin 10	Ruth Ware	★★★★☆ 3,923	\$7.67	Paperback
The Girl on the Train	Emily Blunt	★★★★☆ 2,349	\$10.00	DVD
Big Little Lies	Liane Moriarty	★★★★☆ 15,280	#1 Best Seller in Australia & Oceania Literature	
Into the Water: A Novel	Paula Hawkins	★★★★☆ 1,892	\$15.11	Hardcover
Dark Places	Gillian Flynn	★★★★☆ 9,450	\$9.92	Paperback

Fig. 2: Another example of Amazon co-purchase suggestions

Motivation



- Amazon uses recommendation-based marketing to suggest frequently co-purchased items.
- Customers tend to purchase these recommended items if put on display, & especially on discount.
- Using network analysis, we can find out popular, consistent & valuable item co-purchases.

Scope



- Recommendation systems could be empowered by leveraging power of co-purchase networks.
- Network statistics & properties could be analyzed to determine valuable items that consistently generate demand or are purchased after others.

Key Questions



How to determine valuable item co-purchase pairs? It is related to consistency germane to:

- Which items generate demand for others?
- Which items go with others?
- Which pairs stay popular?

Data Source



The data was picked up from the Stanford Network Analysis Platform (SNAP) which can be found at the following link: <https://snap.stanford.edu/data/amazon-meta.html> It contains information snapshots for 4 time snaps namely “amazon0302”, “amazon0312”, “amazon0505” and amazon0601” along with a Meta Data file containing the details: Title of the product and Group the product belongs to. Following are the details of each of the data sets:

Table 1: Basic statistics for each timestamp

	TimeStamp1	TimeStamp2	TimeStamp3	TimeStamp4
Dated	42796	42806	42860	42887
Edge Count	1234877	3200440	3356824	3387388
Vertex Count	262111	400727	410236	403394

Exploratory Data Analysis

Table 2: Network Statistics

Network Statistics	March-02	March-12	May-05	June-01
Node Count	262111	400727	410236	403394
Edge Count	1234877	3200440	3356824	3387388
Simple Graph	TRUE	TRUE	TRUE	TRUE
Density	0.000018	0.00002	0.00002	0.000021
Connectivity - Strong	FALSE	FALSE	FALSE	FALSE
Connectivity - Weak	TRUE	TRUE	TRUE	FALSE
Reciprocity	0.54	0.53	0.55	0.56
Transitivity - Global	0.24	0.16	0.16	0.17
Transitivity - Local	0.2	0.2	0.2	0.19
Transitivity - Average	0.42	0.4	0.41	0.42
Assortativity - Degree	0	-0.04	-0.04	-0.04
Fast Greedy Clustering - Group	1636	1636	1749	1477
Fast Greedy Clustering - Mod	0.82	0.74	0.75	0.74

- From the above table, we see that the network statistics almost stay consistent at an overall level (except the fact that the node and edge counts in the Mar-02 data is way less than other timestamps)

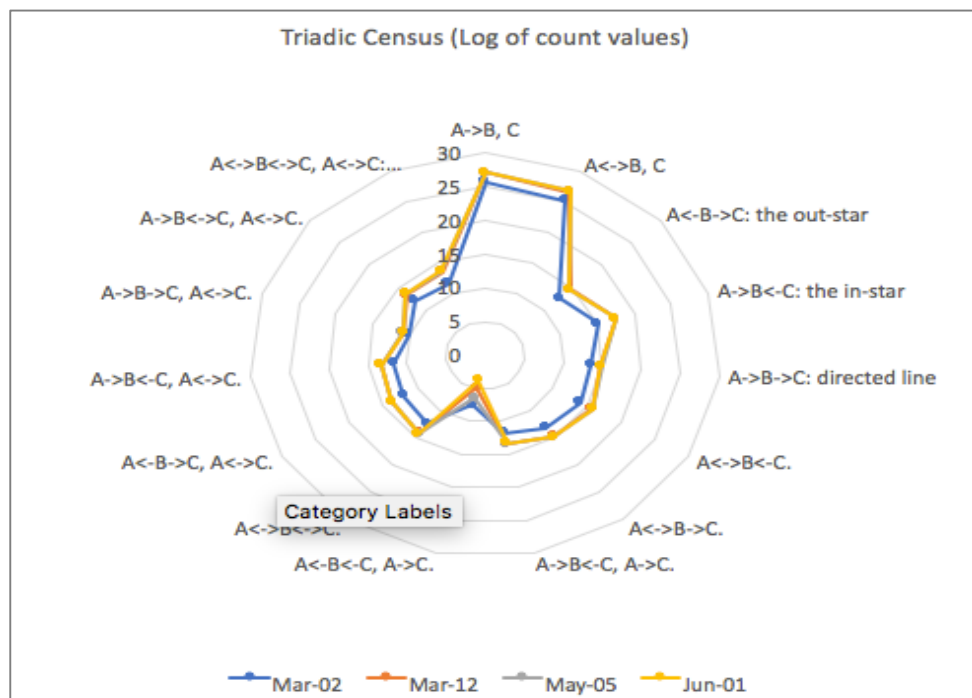


Fig. 3: Radial plot showing the triadic census of different category labels.

- Observing the above figure for Triadic Census – Most of the triads are formed where we have (A->B, C) or (A<->B, C) kind of relationship between nodes.

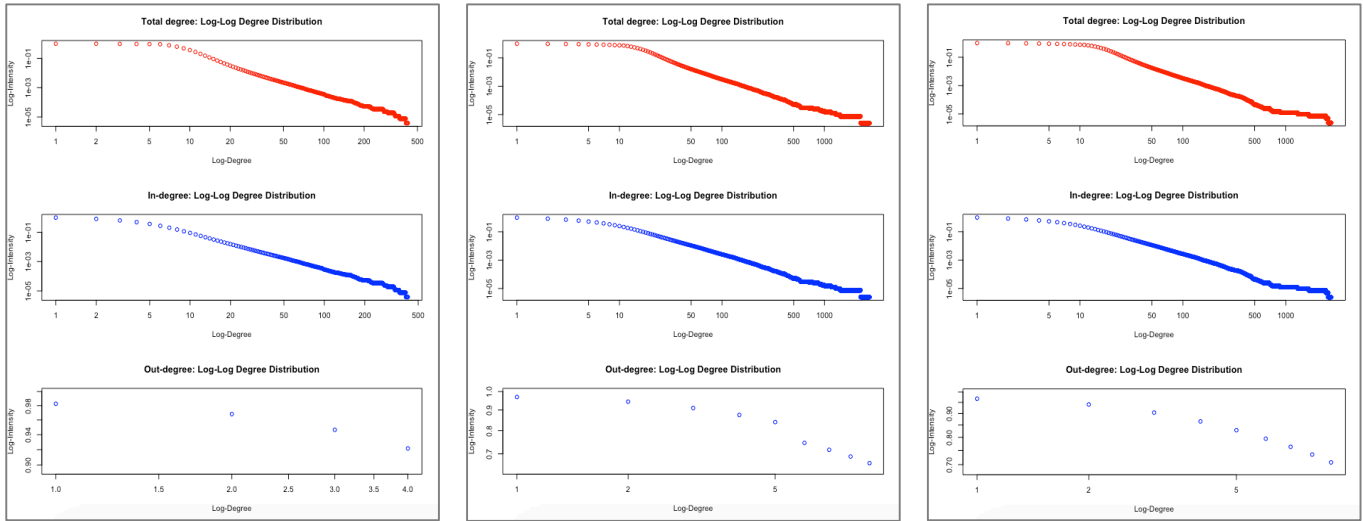


Fig. 4: Degree Distributions (Power Law) for various timestamps; left to right in order of Mar – 02, Mar – 12, May – 05

- In-degree range is larger than out-degree (across different time stamps), indicating presence of many authorities, but less possible hubs. In-degree distribution follows a power law with a heavy tail
- Out-degree follows a similar distribution (typical of small-world scale free network)

Data Mining - Community detection and segregation (and Why Louvain Algorithm?)

The method is a greedy optimization method that attempts to optimize the "modularity" of a partition of the network). The optimization is performed in two steps. First, the method looks for "small" communities by optimizing modularity locally. Second, it aggregates nodes belonging to the same community and builds a new network whose nodes are the communities. These steps are repeated iteratively until a maximum of modularity is attained.

Table 3: Comparison of different community detection models

Algorithm	Description	Modularity
Fast Greedy Algorithm	At each step two groups merge. The merging is decided by optimizing modularity.	0.653
Walk Trap Algorithm	This algorithm finds densely connected sub-graphs by performing random walks. The idea is that random walks will tend to stay inside communities instead of jumping to other communities	0.726
Louvain Algorithm	The method consists of repeated application of two steps. The first step is a "greedy" assignment of nodes to communities, favoring local optimizations of modularity. The second step is the definition of a new coarse-grained network in terms of the communities found in the first step.	0.8968

Process Flow, Analysis, and Results

1. Find the giant component of the entire graph. It was observed that the giant component comprised of ~90% of the edges and nodes, while the most the nodes remaining very singletons. For these reasons, throughout the analysis, the giant component has been used.
2. Community detection on the giant component for each of the four time stamps. This is the first step of clustering the products together into logical co-purchasing groups. Each product (vertex) now belongs to a specific community.

Table 4: Edge and Vertex counts for the full graph and giant components of the network for each timestamp

	TimeStamp1		TimeStamp2		TimeStamp3		TimeStamp4	
	Full Graph	Giant Comp.	Full Graph	Giant Comp.	Full Graph	Giant Comp.	Full Graph	Giant Comp.
Edge Count	1234877	1131217	3200440	3069889	3356824	3255816	3387388	3301092
Vertex Count	262111	241761	400727	380167	410236	390304	403394	395234

Table 5: #Communities w.r.t time stamps

	TimeStamp1	TimeStamp2	TimeStamp3	TimeStamp4
# of Communities	126	120	170	182

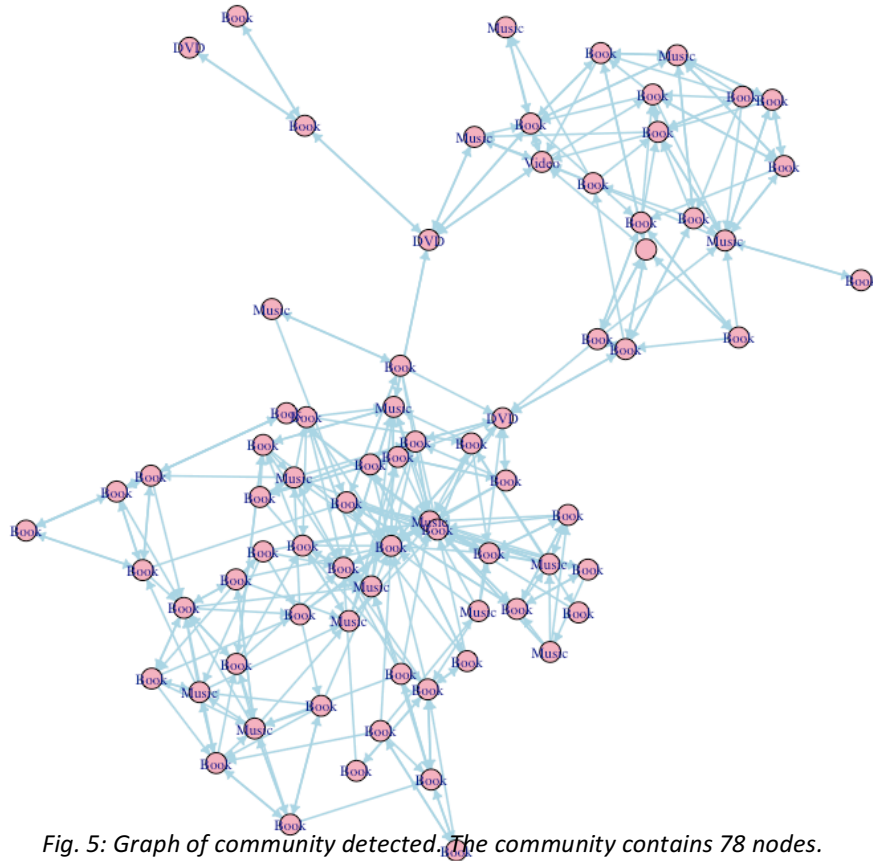


Fig. 5: Graph of community detected. The community contains 78 nodes.

- Community detection on each community that was detected in the step above, for all time stamps. This is the second level of clustering and divides each community into sub-communities. Each product now belongs to a specific community and a sub-community.

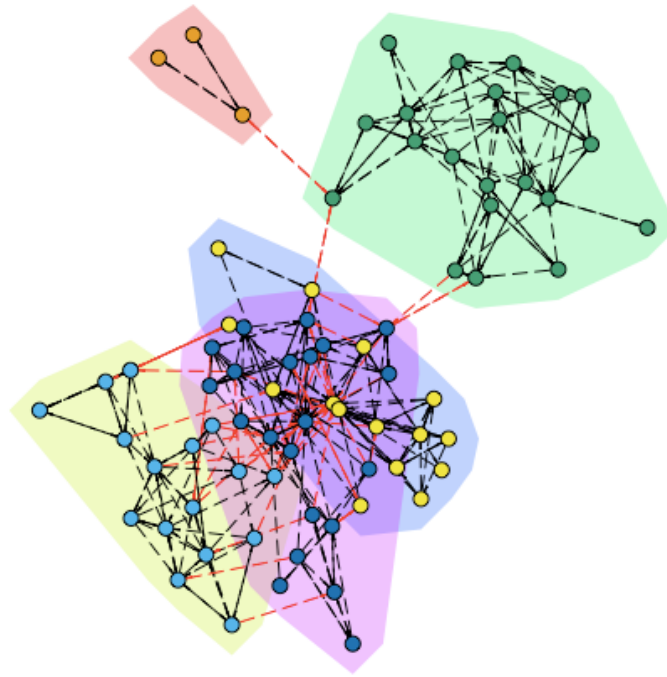


Fig.6: Community detection on the graph in Fig. 5. The community is divided into 5 sub-communities.

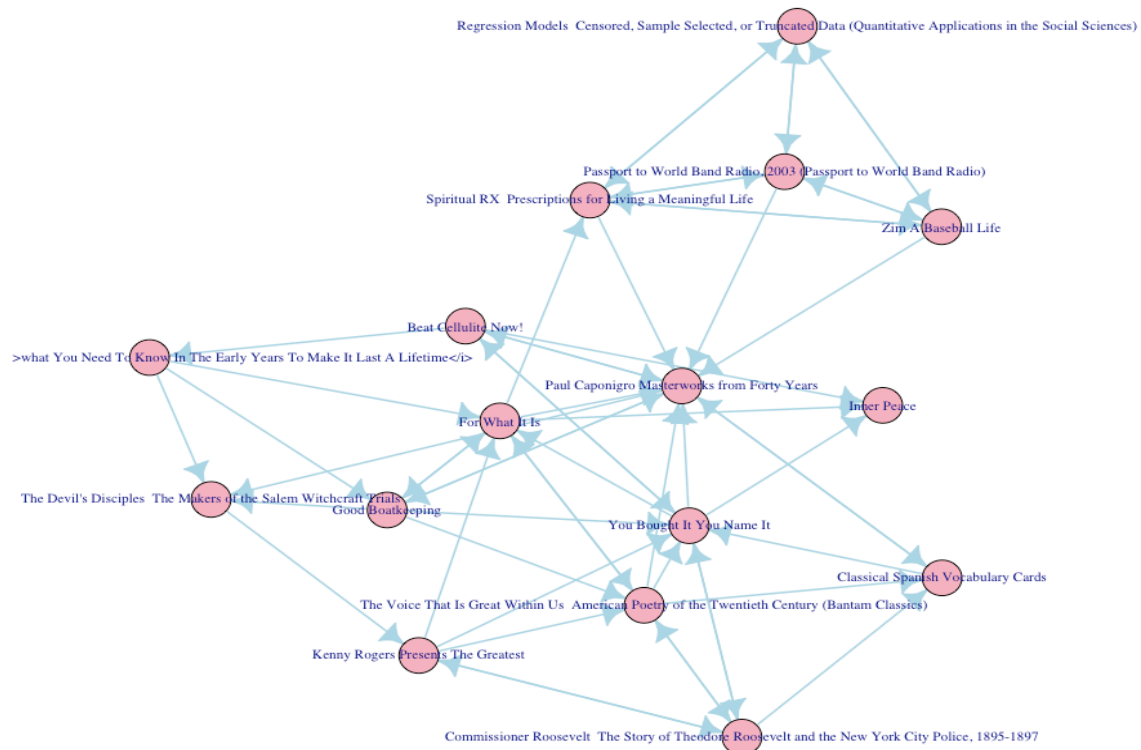


Fig.7. Example of sub-community created after community detection on the graph in Fig.1.

- Community detection clusters products that are co-purchased together. Another level of community detection creates sub-communities that ensures a second level of clustering. As a result of the two-phase community detection, only nodes that are densely connected with be clustered together.

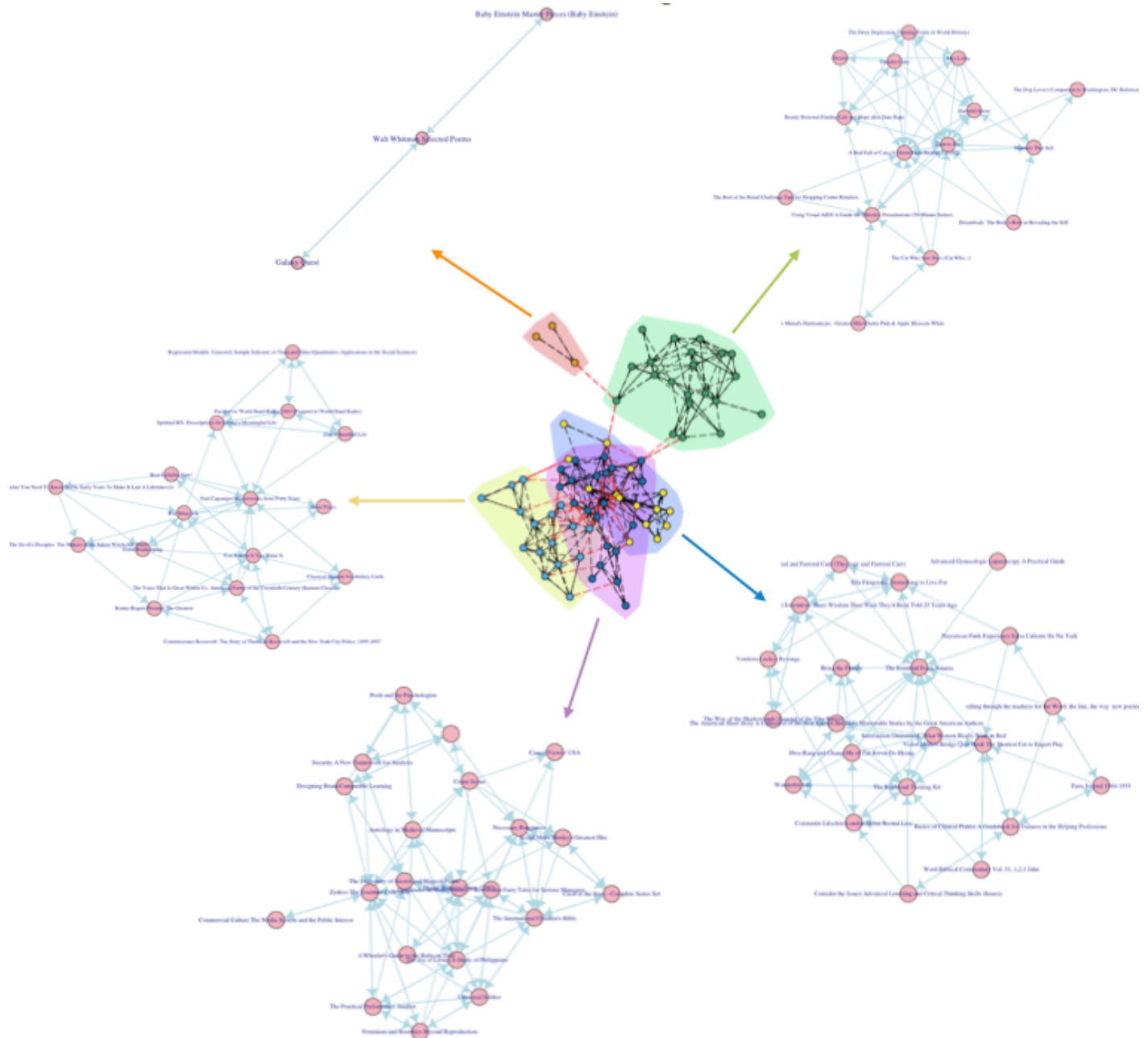


Fig.8. Division of community in Fig.1. into five different sub-communities which are clustered more strongly

- Track nodes that stick together in a community over the period of all four timestamps. Communities evolve over time. What would be a community at time t , could disseminate into two different communities or merge with a community at time $t+1$. So, count the number of times (in a weighted manner – more weight to occurrence in $t-1$, then $t-2$, etc. so as to accommodate recency factor)

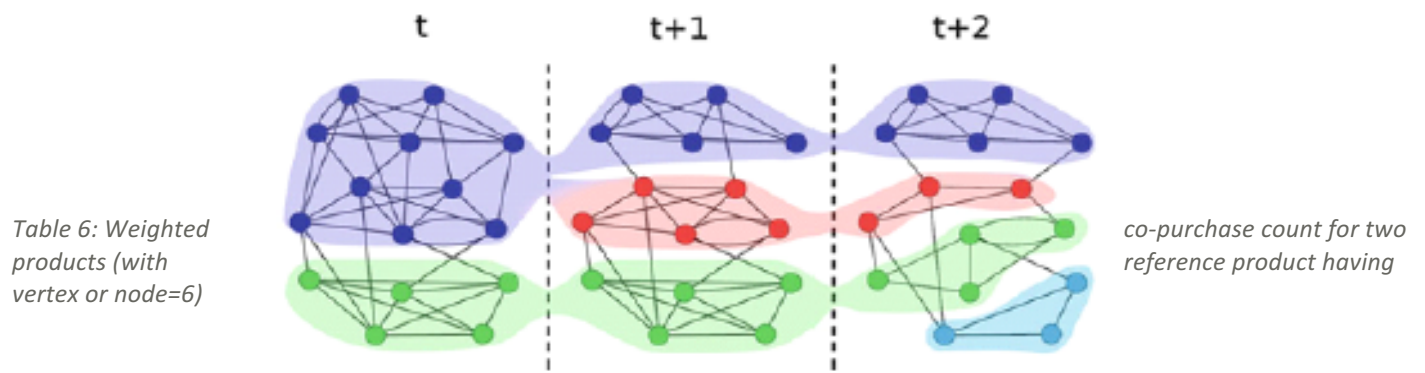


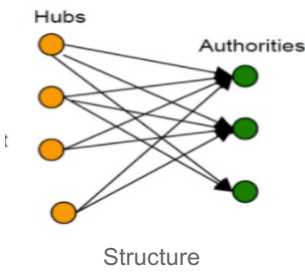
Fig.9. Tracking the evolution communities leads us to node pairs that are consistently together in a community, representing products that were bought together for consecutive time stamps.

Ref_vertex	Title	Group	Vertex	Title	Group	wt_copurchase_cnt
6	How the Other Half Lives Studies Among the Tenements of New York	Book	7	Batik	Music	9
6	How the Other Half Lives Studies Among the Tenements of New York	Book	10	The Edward Said Reader	Book	7
6	How the Other Half Lives Studies Among the Tenements of New York	Book	111	The Maine Coon Cat (Learning About Cats)	Book	7
6	How the Other Half Lives Studies Among the Tenements of New York	Book	113	Official Microsoft Image Composer Book	Book	7
6	How the Other Half Lives Studies Among the Tenements of New York	Book	114	Jesus A Life	Book	7

Note or Key:

- Reference vertex is the node that occurred in top100 nodes when they were ranked by degree across 4 timestamps. This was done to study nodes that are part of the top popular co-purchase across 4 timestamps
- 'wt_copurchase_cnt' is evaluated such that more weight is given to $t-1$ timestamp ($wt=4$), then for $t-2$ ($wt=3$), and so on. The metric is sum of all the corresponding counts ($wt=9$ means the products were co-purchased in $t-1$, $t-2$, $t-3$, and not in $t-4$), and hence higher the value, higher is their rank for co-purchase (this metric includes recency)
- This whole analysis has been replicated for all other product pairs, and could be regulated with different cut-offs. Please see R-code for reference

6. Evaluate the hubs and authorities score of the nodes within same community so as to understand certain patterns in product co-purchases by determining two class of products:
 - a. One having high hub scores: Popular products which could trigger demand for others (like smartphone, camera) – Antecedents or Sources
 - b. One having high authority score: Popular products whose purchase gets triggered due to purchase of other products (like accessories, SD cards) – Consequents or Sinks



Hub-type (for instance)



Authority-type (for instance)

NodeID	timestamp1		timestamp4	
	Hub Score	Auth Score	Hub Score	Auth Score
6	0.008176	0.002157	0.078176	0.004157
7	0.007980	0.000321	0.077980	0.007321
10	0.000559	0.000079	0.090559	0.002079
111	0.000330	0.000131	0.050330	0.005131
113	0.000136	0.000016	0.070136	0.003016
114	0.000023	0.000046	0.070023	0.002046

7. Recommendations to empower the product recommendation system could be now based upon the incorporation of:
 - a. Conventional association based rules driven by market basket analysis
 - b. Network properties related to filtering based on logically related products (regularly found together within same cluster), along with hubs and authorities' property to figure out potential sources and sinks of demand opportunities

Insights and Conclusions

We have found the statistically and logically recommendable pairs of products, along with their source (Antecedents) and sink (Consequent) property based on:

- Popularity of the product co-purchase (referencing above using node degree)
- Co-occurrence in corresponding clusters (logical pairing) across timestamps
- Hubs and authorities scoring across timestamps