Simplicial closure and higher-order link prediction

Eduart Uzeir

eduart.uzeir@studio.unibo.it

Master degree in Computer Science

Agenda

Introduction

Datasets

Static analysis

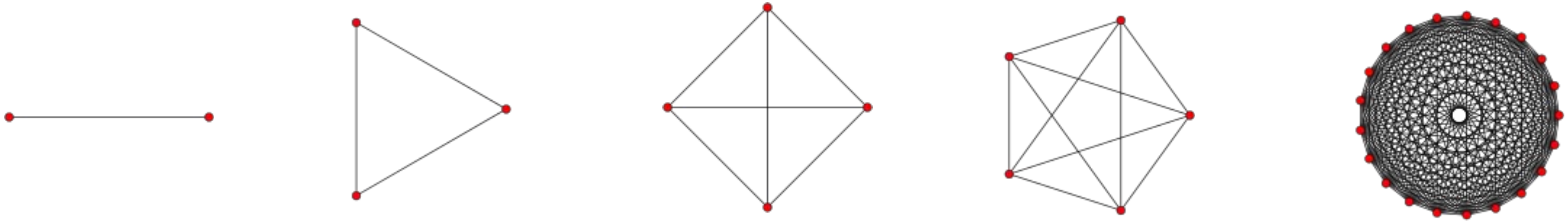
Dynamic analysis - Simplicial closure

Evaluation for higher-oreder link prediction

Conclusion

Preliminaries

➢ Simplex :: generalization of the notion of a triangle to arbitrary dimensions



k=1 k=2 k=3 k=4 k=20

➢ Higher-order interactions :: simultaneous interactions on sets of more than two nodes

Biological interactions between sets of molecules rather than pairs of them

Co-authorship networks

Email networks, emails with multiplet recipients

Preliminaries

- Link prediction :: given a snapshot of a network at time *t*, predict edges added in the interval *(t, t')*

  Idea: Use the properties of the structrure up to some time *t* to predict the appearance of new edges after *t*

- Scoring algorithms based on local similarity (proximity) between nodes – here we measure features like: node degree, node neighbors and common neigbors

  Number of common neighbors

  Jaccard's coefficient

  Adamic – Adar

- Scoring algorithms based on global similarities – consider global features like number of paths and information flow between two nodes

  Shortest path

  Katz

  PageRank

Introduction

- **Graphs, networks, ...?**

  Networks are a powerful abstraction for modeling complex systems and their **pairwise** interactions BUT... , what about the higher-order interactions?!

  Existing formalisms for higher-order structures (set systems, hypergraphs, simplicial complexes etc) are difficult to adapt to the case of graphs and networks due to the lack of a general **framework** for evaluating such models

- **Proposed framework**

  Goal : create a framework analogue to link prediction for the evaluation of models in any dataset where the structure evolves over time through the appearance of new higher-order interactions. For example predicting which set of authors (rather than pair of authors) will write a paper together

  Study of the temporal evolution of 19 network datasets from different domains. Each dataset consists of a collection of time-stamped set of nodes, which is called **simplices**

- **Which features of the structure up to time *t* we can use for higher-order link prediction?**

  In order to find a candidate structural feature we consider the **Projected graph** of the structure

Introduction

➢ Timestamped simplicies, Projected graph, Open triangles and Simplicial closure

$t_1: \{1, 2, 3, 4\}$

$t_2: \{1, 3, 5\}$

$t_3: \{1, 6\}$

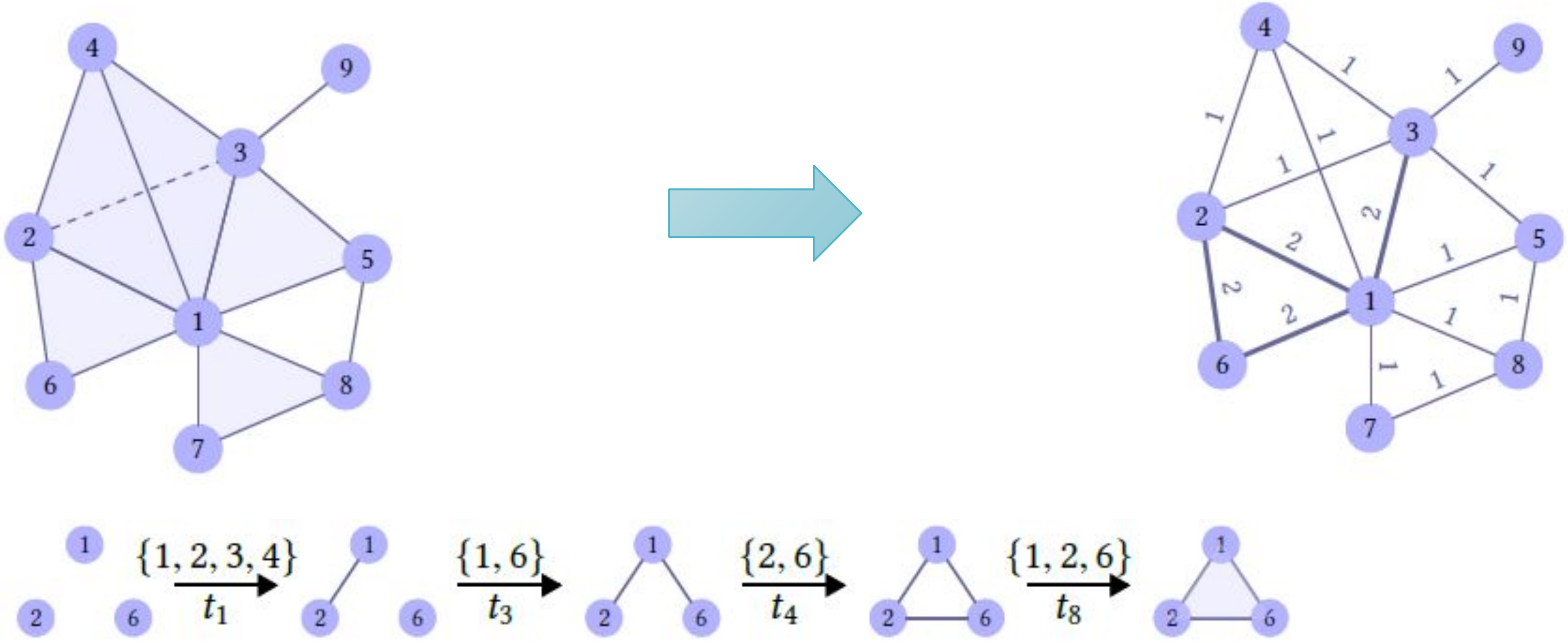
$t_4: \{2, 6\}$

$t_5: \{1, 7, 8\}$

$t_6: \{3, 9\}$

$t_7: \{5, 8\}$

$t_8: \{1, 2, 6\}$



➢ Features for the framework: open triangles and simplicial closure

Introduction

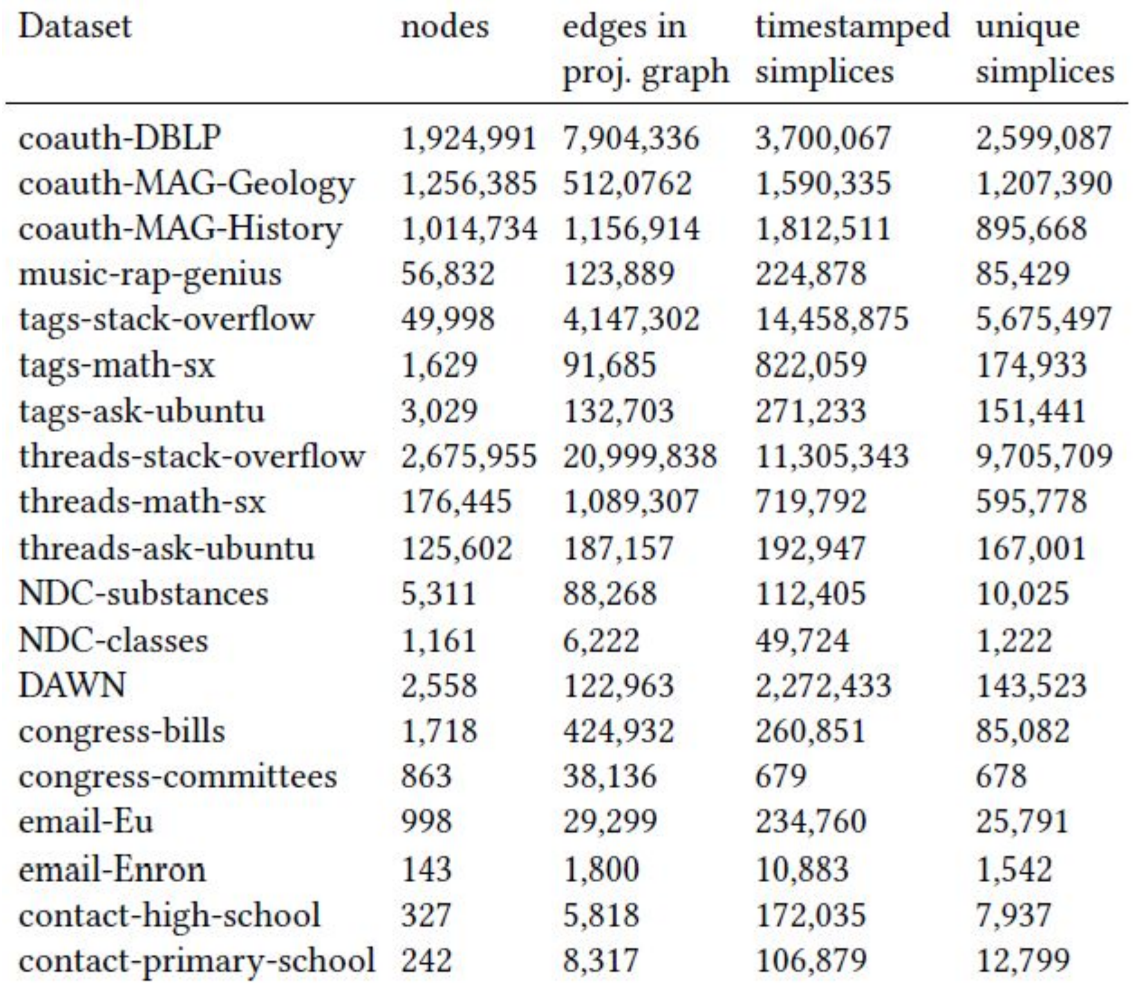
- First results...

The context underlying the network matters, and within a given context higher-order structural parameters such as the fraction of open triangles and edge density are stable. Datasets within the same domain tends to have similar values of this parameters

The relavtive predictive power of edge density and edge weight is different . The study reveals that the datasets differ on which of this two parameters has stronger predictive power, but in datasets within a single domain the efficiency of both parameters is approximately the same

Link prediction for higher-order structures exhibits some fundamental differences from the traditional pairwise link prediction. In order to predict the formation of a simplex between 3 nodes *u*, *v*, *w* we consider the **local** information contained in the egde weights of the projected graph.

Datasets



Dataset	nodes	edges in proj. graph	timestamped simplices	unique simplices
coauth-DBLP	1,924,991	7,904,336	3,700,067	2,599,087
coauth-MAG-Geology	1,256,385	512,0762	1,590,335	1,207,390
coauth-MAG-History	1,014,734	1,156,914	1,812,511	895,668
music-rap-genius	56,832	123,889	224,878	85,429
tags-stack-overflow	49,998	4,147,302	14,458,875	5,675,497
tags-math-sx	1,629	91,685	822,059	174,933
tags-ask-ubuntu	3,029	132,703	271,233	151,441
threads-stack-overflow	2,675,955	20,999,838	11,305,343	9,705,709
threads-math-sx	176,445	1,089,307	719,792	595,778
threads-ask-ubuntu	125,602	187,157	192,947	167,001
NDC-substances	5,311	88,268	112,405	10,025
NDC-classes	1,161	6,222	49,724	1,222
DAWN	2,558	122,963	2,272,433	143,523
congress-bills	1,718	424,932	260,851	85,082
congress-committees	863	38,136	679	678
email-Eu	998	29,299	234,760	25,791
email-Enron	143	1,800	10,883	1,542
contact-high-school	327	5,818	172,035	7,937
contact-primary-school	242	8,317	106,879	12,799

http://www.cs.cornell.edu/~arb/data/

19 datasets from different domains, each of them with a collection of N timestamped simplices

$$\{(S_i, t_i)\}_{i=1}^{N}, t_i \in \Re$$

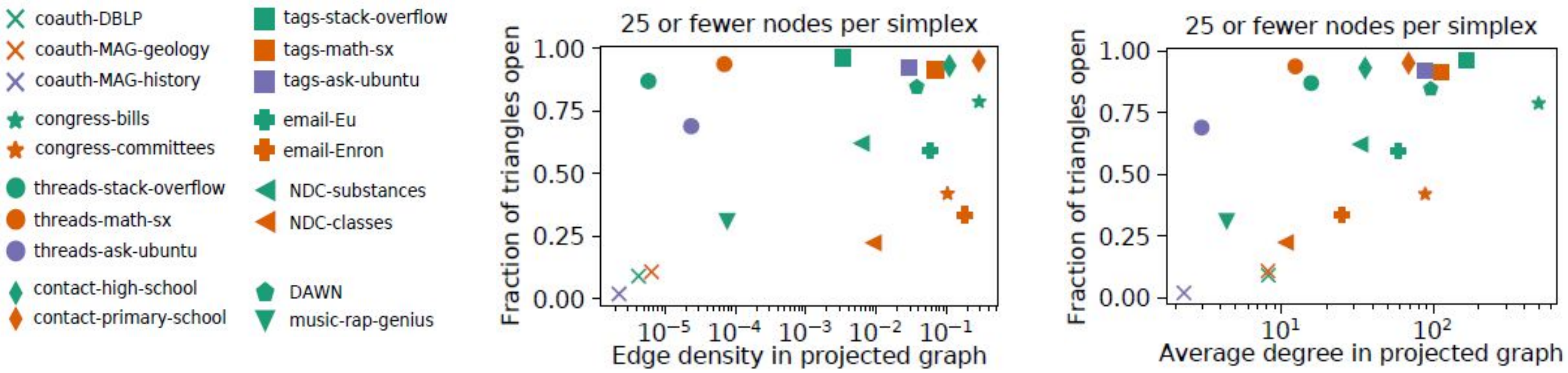
- **Co-authorship data**: nodes are authors and a simplex is a publication of those authors
- **Online tagging data**: nodes are tags, a simplexe is a set of tags for a question in a forum
- **Online thread participation data**: nodes are users, a simplex a set of users answering a particular quesiton
- **Drug networks from the NDCD**: nodes are class labels, a simplex is a set of class labels applied to a drug
- **U.S Congress data**: nodes are members of Congress, a simplex is Congress members in a committee
- **Email networks**: nodes are email addresses, a simplex is the set of addresses sending or receiving an email
- **Contact networks**: nodes are persons, a simplex is a set of persons in a close proximity to each other in a given time
- **Music collaborations**: nodes are rap artists, a simplex is a set of artists collaborating on a song

Static Analysis

- ➢ Open vs. Close triangles

$$\{u,v,w\},\neg\exists S_i \mid \{u,v,w\}\subset S_i \qquad \{u,v,w\},\exists S_i \mid \{u,v,w\}\subset S_i$$

- ➢ Structure of datasets



Static Analysis

- Intuition...

  Every simplex with at least 3 nodes, will directly create a closed triangle, while open triangles are coincidental. Thus closed triangles should be more common than open triangles... but surprisingly whe observe that...

- Open triangles are more common than close traingles. Why?!

  Hypothesis: “3-node simplices form independently with a fixed probability”

  Suppose that the dataset consist of 3-node simplices, and a given set of 3 nodes {u, v, w}, is a simplex with probability $p = 1/n^b$,for b > 0

  Let **Xuvw** be the random variable that indicates that {u, v, w} is an open triangle, then for large n we have :

$$\mathrm{E}[Xuvw] \approx (1-(1-1/n^b)^n)^3$$

$$\begin{cases} (1-1/n^b)^n \le e^{-n^{1-b}} \wedge \mathrm{E}[Xuvw] \to 1, b<1 \\ \mathrm{E}[Xuvw] \approx (1-(1-1/n^b)^n)^3 = \mathrm{O}(1/n^{3b-3}), b>1 \end{cases}$$

We denote with **O** the set of open triangles and with **C** the set of close triangles, according to the previous formulas we will have:

$$\mathrm{E}[|O|] = \begin{cases} \sum_{\{u,v,w\}} \mathrm{E}[Xuvw] = \mathrm{O}(n^3), b<1 \\ \sum_{\{u,v,w\}} \mathrm{E}[Xuvw] = \mathrm{O}(n^{3(2-b)}), b>1 \end{cases} \qquad \mathrm{E}[|C|] = p\binom{n}{3} = \mathrm{O}(n^{3-b})$$

Conclusion, if the probability of 3-node simplex to form is larger $p > 1/n^{3/2}$ we can expect more open triangles than close in this model

Dynamic Analysis

- How does high-order stuctures evolve over time?

One way to explain the prevalence of the open triangles may be the temporal asynchronicity. This can be observed in the Congress committee dataset but in most of the other datasets, temporal asynchronicity does not explain the dominance of open triangles

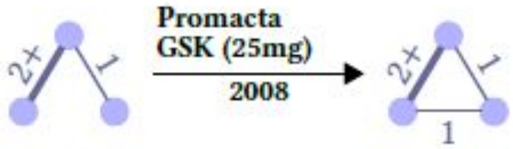
- Simplicial closure

Three nodes in an open triangle may form a simplex in the future as the network evolve. This process is called “Simplicial closure”

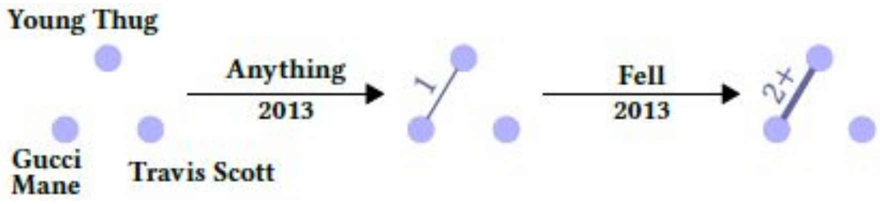
- Lifecycle of nodes

All possible configurations of 3 nodes before they appear in a simplex together (simplicial closure). A triplet of nodes can undergo two kind of changes during its lifecycle:

a) an extra edge can be added between two nodes *u*, *v* increasing the edge density



b) projected graph edge weights can increase, increasing the tie strength



Dynamic Analysis

- ## Simplicial closure of three nodes

The goal is to compute the probability of simplicial closure of a node-triple taking into account the lifecycle of this nodes. In order to achieve this goal, first divide each dataset in 2 distinct sets based on the temporal order of the appearance of the simplicies

Consider a timestamp *t**and put 80% of the simplices created untill *t** in the “training set” and the remaining 20% in the “testing set”

At this point measure the probability that a node-triple from the training set will form a closed triangle in the test set as a function of its previous configuration. This result on 10 closure probabilities for each dataset, equal to the number of states in a node-triple lifecycle

- ## Findings

a). Simplicial closure probability increases as the edge density increases (112/113 cases)

b). Simplicial closure probability increases with tie strength (82/113 cases against 6/113)

c). Neither edge density nor tie strength dominates the influence on simplicial closure

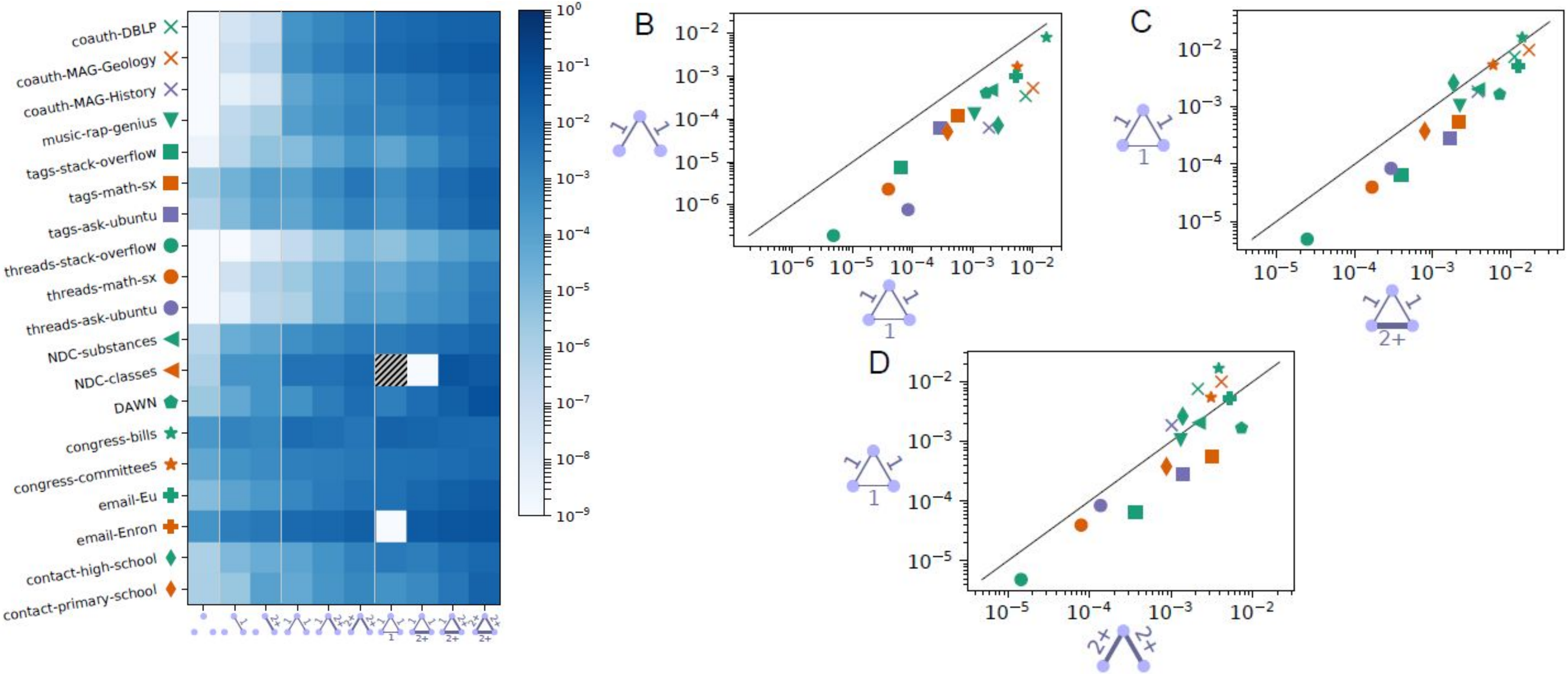
d). These results suggests different closure dynamics for different dataset domains

For example, human social interactions are driven by a topological form of triadic closure (mutual aqcuaintance between nodes).

In the case of online discussion platforms the process of simplicial closure is based on the transitive closure

- ## The previous results apply even to four node simplices

Dynamic Analysis



Higher-order link prediction

- ➢ What we know up to this point?

  We know that **edge density** and **tie strength** are significat positive indicators for simplicial closure

  Now we want to evaluate the behaviour of higher-order link prediction algorithms

- ➢ How can we express the problem of higher-oreder link prediction?

  If we consider for simplicity the node-triplets, the idea is to predict which node-triplets that have not co-appeared in a simplex together will form a simplex in the future, in other words which sets of node-triplets will undergo a process of simplicial closure

  Another way to see this is to predict which open triangles in the training set will close in the testing set

  We can observe that this process is easy to compute because it is just about counting the number of open triangles on the training set

Algorithms

- How does the higher-order link prediction algorithms work?

  The algorithm defines a *score function* ***s*** for each open triangle based on the algorithm's confidence that each given open triangle will close. This ranking algorithm can be formalized as follow

$$s : V \times V \times V \to \Re$$

  For example if ***s(i, j , k) > s(a, b , c)*** then the algorithm consider the node-triple ***{i, j, k}*** more likely to co-appeare in a simplex in the future than the node-triple ***{a, b, c}***

- Algorithms for higher-order link prediction (4 categories)

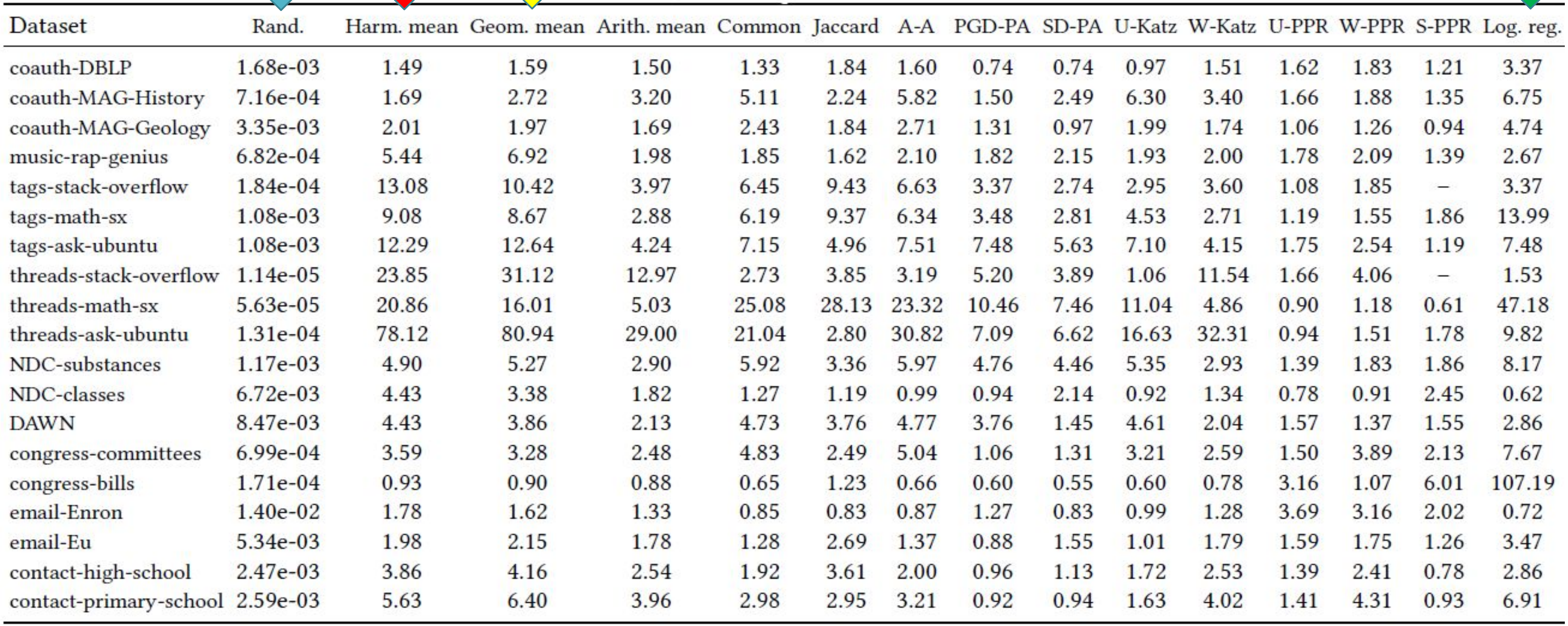
  a). The score function depends only on the weights of edges in the projected graph

  b). The score function depends on the local neighborhood

  c). The score funcition is based on paths and random walks

  d). The score function is based on a supervised learning

Prediction Performance



Dataset	Rand.	Harm. mean	Geom. mean	Arith. mean	Common	Jaccard	A-A	PGD-PA	SD-PA	U-Katz	W-Katz	U-PPR	W-PPR	S-PPR	Log. reg.
coauth-DBLP	1.68e-03	1.49	1.59	1.50	1.33	1.84	1.60	0.74	0.74	0.97	1.51	1.62	1.83	1.21	3.37
coauth-MAG-History	7.16e-04	1.69	2.72	3.20	5.11	2.24	5.82	1.50	2.49	6.30	3.40	1.66	1.88	1.35	6.75
coauth-MAG-Geology	3.35e-03	2.01	1.97	1.69	2.43	1.84	2.71	1.31	0.97	1.99	1.74	1.06	1.26	0.94	4.74
music-rap-genius	6.82e-04	5.44	6.92	1.98	1.85	1.62	2.10	1.82	2.15	1.93	2.00	1.78	2.09	1.39	2.67
tags-stack-overflow	1.84e-04	13.08	10.42	3.97	6.45	9.43	6.63	3.37	2.74	2.95	3.60	1.08	1.85	–	3.37
tags-math-sx	1.08e-03	9.08	8.67	2.88	6.19	9.37	6.34	3.48	2.81	4.53	2.71	1.19	1.55	1.86	13.99
tags-ask-ubuntu	1.08e-03	12.29	12.64	4.24	7.15	4.96	7.51	7.48	5.63	7.10	4.15	1.75	2.54	1.19	7.48
threads-stack-overflow	1.14e-05	23.85	31.12	12.97	2.73	3.85	3.19	5.20	3.89	1.06	11.54	1.66	4.06	–	1.53
threads-math-sx	5.63e-05	20.86	16.01	5.03	25.08	28.13	23.32	10.46	7.46	11.04	4.86	0.90	1.18	0.61	47.18
threads-ask-ubuntu	1.31e-04	78.12	80.94	29.00	21.04	2.80	30.82	7.09	6.62	16.63	32.31	0.94	1.51	1.78	9.82
NDC-substances	1.17e-03	4.90	5.27	2.90	5.92	3.36	5.97	4.76	4.46	5.35	2.93	1.39	1.83	1.86	8.17
NDC-classes	6.72e-03	4.43	3.38	1.82	1.27	1.19	0.99	0.94	2.14	0.92	1.34	0.78	0.91	2.45	0.62
DAWN	8.47e-03	4.43	3.86	2.13	4.73	3.76	4.77	3.76	1.45	4.61	2.04	1.57	1.37	1.55	2.86
congress-committees	6.99e-04	3.59	3.28	2.48	4.83	2.49	5.04	1.06	1.31	3.21	2.59	1.50	3.89	2.13	7.67
congress-bills	1.71e-04	0.93	0.90	0.88	0.65	1.23	0.66	0.60	0.55	0.60	0.78	3.16	1.07	6.01	107.19
email-Enron	1.40e-02	1.78	1.62	1.33	0.85	0.83	0.87	1.27	0.83	0.99	1.28	3.69	3.16	2.02	0.72
email-Eu	5.34e-03	1.98	2.15	1.78	1.28	2.69	1.37	0.88	1.55	1.01	1.79	1.59	1.75	1.26	3.47
contact-high-school	2.47e-03	3.86	4.16	2.54	1.92	3.61	2.00	0.96	1.13	1.72	2.53	1.39	2.41	0.78	2.86
contact-primary-school	2.59e-03	5.63	6.40	3.96	2.98	2.95	3.21	0.92	0.94	1.63	4.02	1.41	4.31	0.93	6.91

Prediction Performance

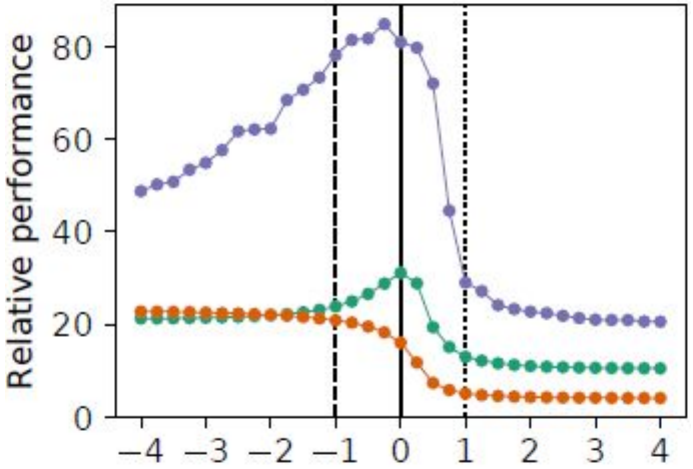
- Findings...

a). There is not one score function that performes best over all datasets

b). Harmonic and geometric means of edge wieghts perform well in many datasets

c). Considering the generalized mean with parameter ***p*** as score functions and ***Wab*** the weight between 2 nodes ***a***, ***b***
:

$$s_p(i,j,k) = [(W_{ij}^p + W_{jk}^p + W_{ik}^p)/3]^{1/p}$$



the peredicting performance is:

(i) unimodal in p,

(ii) maximized for p in [-1, 0],

(iii) better for p < -1 than for p > 1

d). The supervized learning performed very well in the large datasets

h --- g — a ....

Conclusion

“I am turned into a sort of machine for observing facts and grinding out conclusions”.

Ch. Darwin

Conclusions

- ➢ In the quest for predicting which set of nodes will interact simultaneously in the future we found...

1. Great variety on different datasets regarding the fraction of open triangles and other structural properties of the projected graph such as the edge denisty and average degree but inside the same domain the datasets behave similarly.
2. Edge density and tie strength are very important positive indicators for simplicial closure of three and four node siplices.
3. Which of this two features is most influential, depends on the dataset domain.
4. Local measures such as the tie strength between two nodes are fundamental informations to use in the link prediction process.
5. The previous result suggest that higher-order temporal evolution is different from the traditional network evolution, (the first is based on local informations, the second on informations contained on long paths).

References

[1] Simplicial Closure and Higher-order Link Prediction (https://arxiv.org/pdf/1802.06916.pdf)

[2] Link Prediction in Social Networks – Role of Power Law distribution (V. Srinivas, P. Mitra)

[3] Statistical mechanics of complex networks (R. Albert, A. Barabasi, https://goo.gl/sLzHsb)

[4] Network Analysis – Link Prediction (L. Zhukov, https://goo.gl/EnjBv2)

[5] Graph Theory and Complex Networks (M. Van Steen, https://goo.gl/UMsSvW)

[6] Bipartite network projection and personal recommendation (https://goo.gl/4v7Q9k)

Appendix

Harmonic Mean (H)

Given x1, x2, ...xn real numbers, H is defined:

$$H = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \ldots + \frac{1}{x_n}} = \frac{n}{\sum_{i=1}^{n} \frac{1}{x_i}} = \left(\frac{\sum_{i=1}^{n} x_i^{-1}}{n} \right)^{-1}$$

Arithmetic Mean (A)

Given the values x1, x2, ... , xn, A is defined:

$$A = \frac{1}{n} \sum_{i=1}^{n} x_i = \frac{x_1 + x_2 + \ldots + x_n}{n}$$

Geometric Mean (G)

Given a set of data {x1, x2, ..., xn}, G is defined:

$$G = \left(\prod_{i=1}^{n} x_i \right)^{\frac{1}{n}} = \sqrt[n]{x_1 x_2 \ldots x_n}$$

Generalized Mean (M)

Given the non-zero, real number p and x1, x2, ...xn positive real numbers then Mp is defined:

$$M_p(x_1, x_2 \ldots x_n) = \left(\frac{1}{n} \sum_{i=1}^{n} x_i^p \right)^{\frac{1}{p}}$$

Appendix

Algorithms

- ➢ First category of algorithms

  The score function ***s(i, j, k)*** depends only on the weights of the edges ***(i, j)***, ***(i, k)***, ***(j, k)*** in the projected graph

  Harmonic, geometric and arithmetic means of the edges of the projected graph are considered, this imply that stronger ties lead to a higher score

- ➢ Second category of algorithms

  The score function ***s(i, j, k)*** is based on local neighborhood features in the projected graph such as the common neighbors of nodes ***i***, ***j*** and ***k***

  Here we are looking at the number of common fourth neighbors of the three nodes in the projected

  These score functions are generalizations of the score functions used on dyadic link prediction (Jaccard, Adamic-Adar)

- ➢ Third category of algorithms

  The score function here is based on paths and random walks on projected graph

  Algorithms like Katz and PageRank are used to compute pairwise similarities between nodes and the total score is the sum of pairwise scores between the three nodes

Appendix

Algorithms

- Fourth category of algorithms

  In this category of algorithms is used a supervised machine learning approach that looks for the optimal score function based on features of the open triangles

  For each open triangle ***(i, j, k)*** the features are:

  a). Number of simplices containing pairs of nodes i and j, i and k, j and k

  b). The degree of nodes i, j, k in the projected graph (*N|(i)|, N|(j)|, N|(k)|*)

  c). The number of simplices containing nodes i, j, k

  d). The number of common neighbors in the projected graph of nodes i and j, i and k, j and k

  e). The number of common neighbors of all three nodes