Complex networks

Network growth, node similarity

János Török

Department of Theoretical Physics

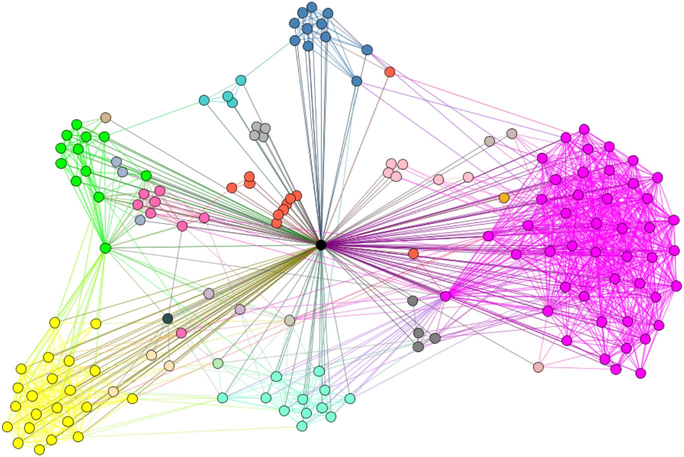
April 5, 2023

SBM: Summary

- Very flexible, generative method to model
- Communities, but also arbitrary mixing patterns, including, for example, bipartite, and core-periphery structures;
- Able to separate noise from structure;
- No resolution limit
- Generalization to directed, weighted networks possible.
- Structure detection is converted to parameter inference
- Increasingly efficient algorithms
- Can be used to detect communities

Growing networks

- Simulate real life
- Use minimal elements
- Do not incorporate effect what one wants to recover
- Example: simulate social network (modular)



Growth models

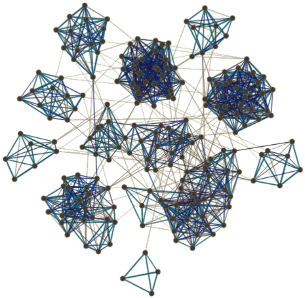
- Barabási-Albert model: Simple growth mechanism, preferential attachment, model for Internet
- More complicated systems?
- Two version of a simple model for social networks

Social networks

- Human relation
- Very complicated dynamics
- Not really a growth model, more a dynamics steady state
- Observations:
  - Weighted network
  - Large clustering coefficient (friend of friends usually know each other)
  - Not scale free
  - Small world
  - Granovetter: Strength of the weak ties

Granovetter: Strength of the weak ties

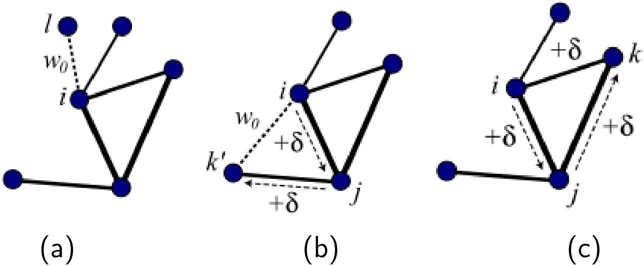
- Human groups are strongly connected
- There are weak connections connecting the groups
- These weak connections mean sproadic meeting
- Important for information flow
- Example: Find a job



Granovetter, Mark S. "The strength of weak ties." American journal of sociology 78.6 (1973): 1360-1380.

Kumpula model

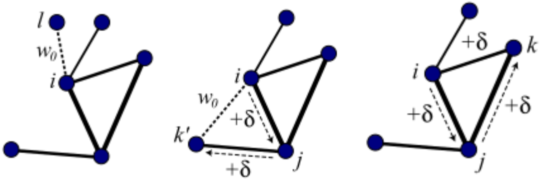
- N nodes (originally unconnected)
- (a) Randomly meet someone (low probability) global attachment
- (b) Two friends of someone get to know each other, cyclic closure
- (c) An already present triangle gets strengthened



JM. Kumpula, Emergence of communities in weighted networks. PRL, 99(22), 228701 (2007).

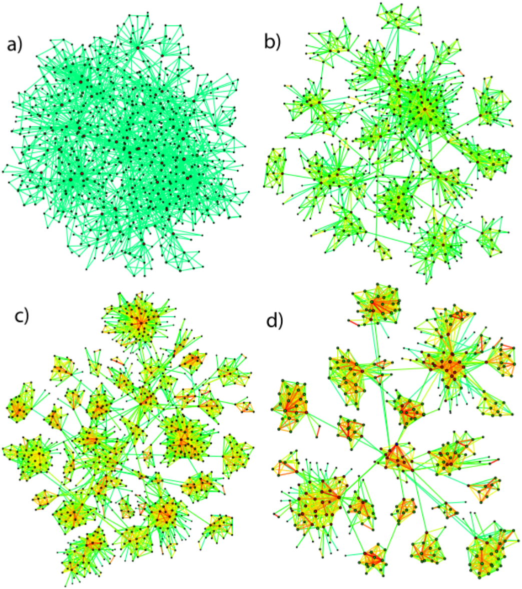
Kumpula model

- N nodes (originally unconnected)
- (a) (with prob. p_r) random link to an unconnected node. Link weight w_0
- (with prob. p_d) i selects friend j with prob. proportional to the link weight. j selects friend k similarly. Both links are strengthened by δ. Two cases:
  - (b) There is no link between i and k: create a link with p_Δ with weight w_0
  - (c) There is a link between i and k: strengthen by δ
- (d) (with prob. p_d) clear the links of a node (enforce steady state, there are more realistic versions)



(a) (b) (c)

Kumpula model: results ($\delta = 0,\ 0.1,\ 0.5,\ 1.0$)



Kumpula model: results

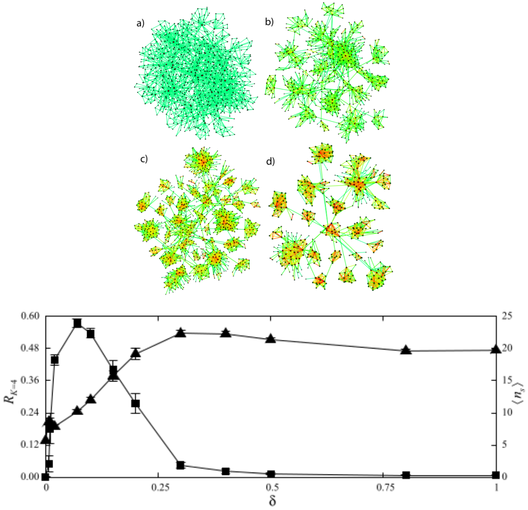


FIG. 3: $R_{k=4}$ (□) and $\langle n_s \rangle$ (△) as a function of δ. Results are averaged over 10 realizations of $N = 5 \times 10^4$ networks. Error bars are measured standard deviations.

Kumpula model: results

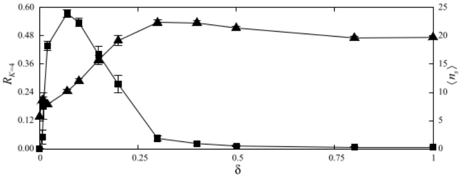
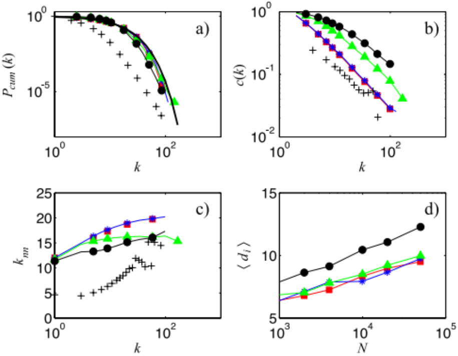


FIG. 3: $R_{k=4}$ (□) and $\langle n_s \rangle$ (△) as a function of δ. Results are averaged over 10 realizations of $N = 5 \times 10^4$ networks. Error bars are measured standard deviations.



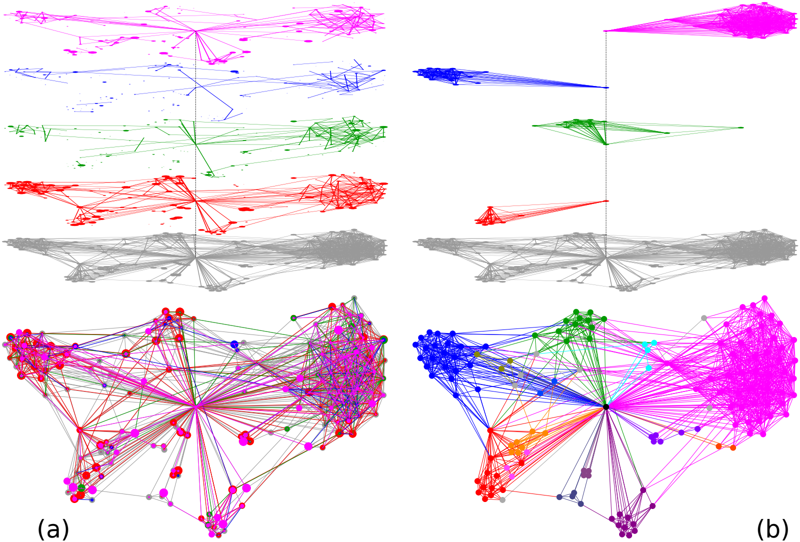
Kumpula model: results

- Very simple assumptions
- Emergence of community structure (depending on parameters)
- Good to test effects of elementary processes on global structure
- Not apt for recovering well defined structures

Multiplex networks: Social networks

Communication channel

Social context



Axelrod model of dissemination of culture

- Each individual is endowed with a certain culture
- They have cultural needs and preferencies therein
- An individual's culture is characterised by a list of F features
- Each feature has q different traits
- Assumptions
  - people are more likely to interact with others who share many of their cultural attributes
  - these interactions tend to increase the number of cultural attributes they share (thus making them more likely to interact again).

Axelrod 1997

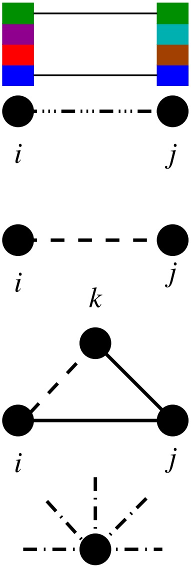
Axelrod model of dissemination of culture

- Model
  - One agent k (active) is selected at random.
  - One of agent k's neighbours, denoted agent r (passive), is selected at random.
  - n_{kr} number of features in which agents k and r matches
  - Agents k and r interact with probability equal to their cultural similarity n_{kr}/f
  - The interaction consist of k copying one of the unmatched features of agent r
  - In this way, agent k approaches agent r's cultural interests

MOVIE

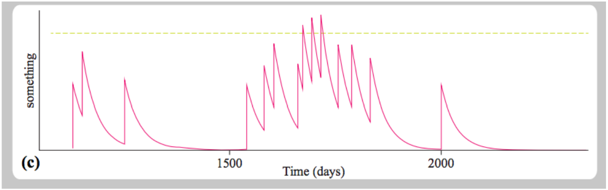
Multi layer model of social networks

- Peaple have F social features with q values each
- Ego first selects feature (s)he wants to do some social action
- (S)he can do it only with people with matching the specific feature
- Random connection, rare
- Triangles: common
  - Link selection proportional to weight
  - Link establishment with some probability and strengthening participating links
  - Link aging

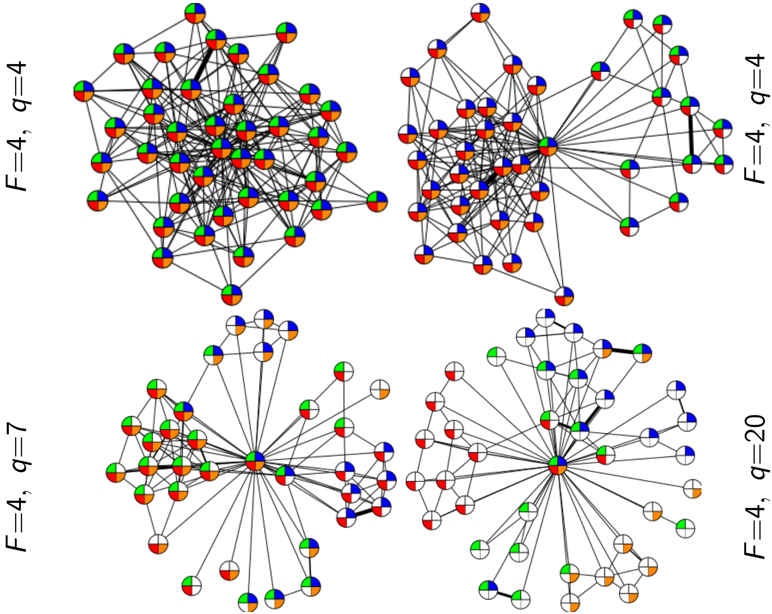


Link aging

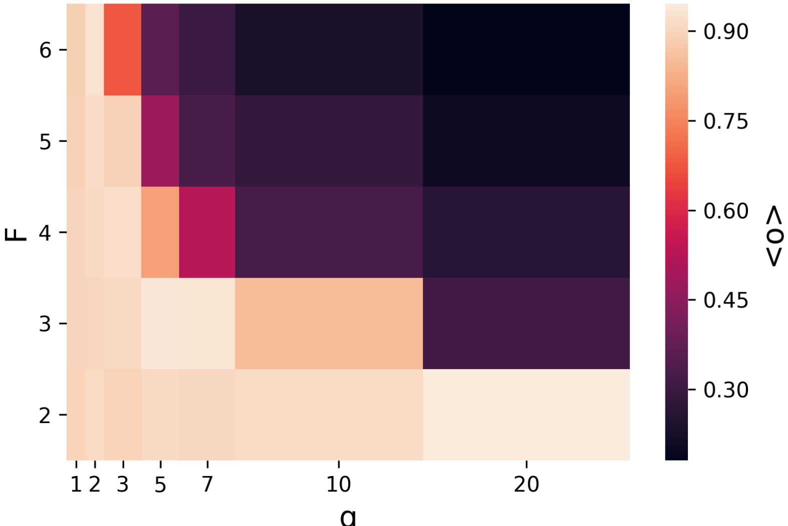
- Steady state
- Relationships fade with time
- Communication is an instantaneous strengthening



Multilayer social model: egocentric networks



Multilayer social model: Phase diagram

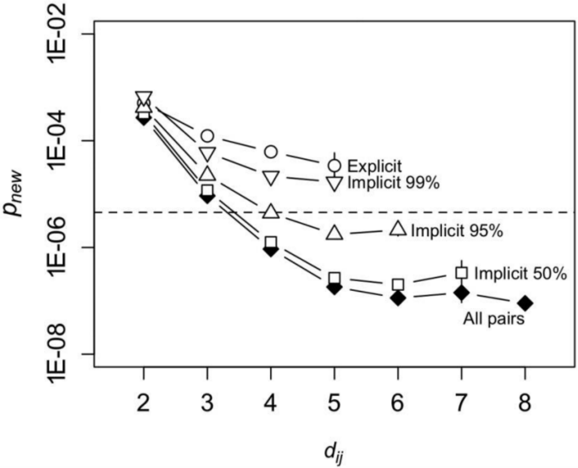


Link prediction

- If next link can be predicted, we can guess dynamics
- If process is known, we can rebuild the network (e.g. preferential attachment)
- Correct missing links in ICT data
- Important for companies

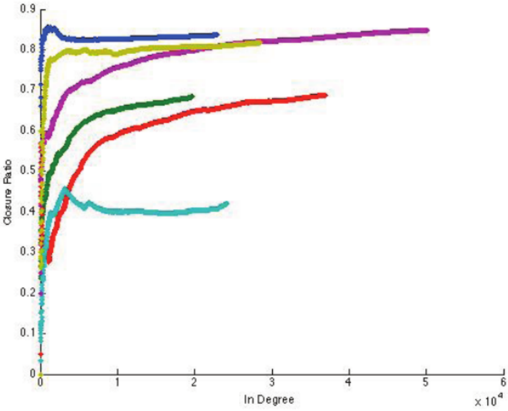
Triadic closure

- Triadic closure: friends of friends get friends.
- Cycloc closure: firends at distance *d* get friends
- Focal closure: tie formation is related to social focus (interest, work, etc.)



Triadic closure in twitter

- Twitter data
- Middle size celebrity ($10^4 - 5 \cdot 10^4$ followers)
- Closure: New follower had link to an existing follower



Comedian, TV Presenter, Actor, Musician, Filmmaker, Actor

Link prediction

- Given a social network structure can we predict, which links will be formed in the future?
- Recommendation systems: If costumer A has chosen items x,y,z what shall we recommend?
- How to uncover a criminal network from sparse data?
- How to reconstruct the network if only partial information is available?

Supervised learning

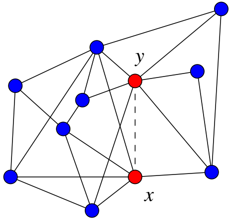
- Artificial neural network
- Use data to teach and test
- Useful for companies, can always be updated with new data
- Black box, does not help to recover important features

Measures

- Given two nodes
- Define a measure
- The links with the highest measure will have the largest probability to appear
- Let us visit the zoo of measures!

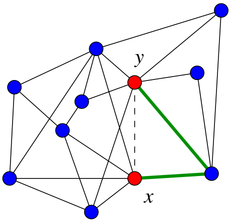
Common neighbors

- Local
  - Graph distance
  - Common neighbors (CN)
  - Jaccard (JC)
  - Adamic-Adar (AA)
  - Preferential attachment (PA)
- Global
  - Katz score
  - Hitting time
  - PageRank



Graph distance

- Length of the shortest path
- Negated, (or inverse) to give higher score for better guesses
- Generally not very reliable, as it starts with value of 2 and the value of 3 is already around average value
- Cannot distinguish between the second neighbors



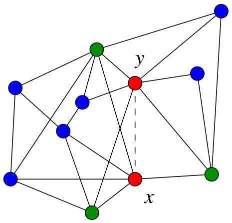
$GD = 2$

Common neighbors

- Number of common neighbors

$$CN = |\Gamma(x) \cap \Gamma(y)|$$

- $\Gamma(x)$ neighbors of x
- $|S|$ size of set S
- In spite of its simplicity surprisingly accurate
- Use this if you have no better idea



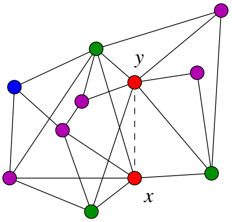
$CN = 3$

Jaccard's coefficient

- Number of common neighbors normalized by the number of total neighbors

$$CN = \frac{|\Gamma(x) \cap \Gamma(y)|}{|\Gamma(x) \cup \Gamma(y)|}$$

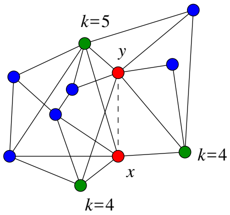
- Normalization does not necessarily improve results especially if k is large
- In most cases it is worse than common neighbors



$CN = 3/8$

Adamic/Adar

- Consider all common neighbors
- Weight common neighbors with low degree higher
- The idea behind this is that a low degree node connects both they are more likely to get connected



$$AA = \sum_{z \in \Gamma(x) \cap \Gamma(y)} \frac{1}{\log |\Gamma(z)|}$$

$$AA = \frac{1}{\log(4)} + \frac{1}{\log(4)} + \frac{1}{\log(5)}$$

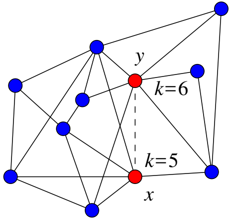
- Generally the best performance

Preferential attachment

- Neighborhood size as feature value
- Rich gets richer

$$PA = |\Gamma(x)| \cdot |\Gamma(y)|$$

- Far the worst



$$PA = 5 \cdot 6 = 30$$

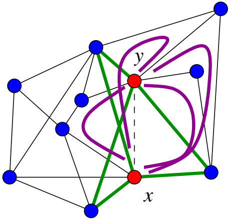
Katz$_\beta$

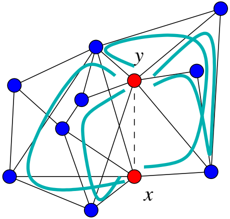
- Consider all possible path between x and y
- Sum them with penalty for longer path

$$KS = \sum_{p \in \text{path}(x,y)} \beta^{|p|}$$

where $|p|$ is the length of the path

- $\beta < 1$, but generally $\beta \simeq \mathcal{O}(10^{-2} - 10^{-4})$
- Very small β is similar to common neighbors because then only paths of length contribute





$\beta = 0.1$

$$KS = 3 \cdot 0.1^3 + 4 \cdot 0.1^4 + + 5 \cdot 0.1^5 + \mathcal{O}(0.1^6) \quad (1)$$

Katz$_\beta$

- Consider all possible path between x and y
- Sum them with penalty for longer path

$$KS = \sum_{p \in \mathrm{path}(x,y)} \beta^{|p|}$$

  where $|p|$ is the length of the path
- Generally excellent performance
- An $\mathcal{O}(N^3)$ method
- It is equivalent to calculating

$$(I - \beta A)^{-1} - I$$

  where A is the adjacency matrix, I the identity matrix

Hitting time

- Start a random walker at x
- Measure the expected time it needs to reach y
- It is the hitting time

$$HT = -H_{x,y}$$

- From mediocre to worst performance

Commute time

- Symmetrized hitting time

$$HT = -H_{x,y} - H_{y,x}$$

- Much better, acceptable performance

Normalized commute time

- Problem with hitting time that high degree nodes with high stationary probability (π) get the walker fast irrespective of the starting point
- Normalize with it

$$HT = -H_{x,y}\pi_y - H_{y,x}\pi_x$$

- Worse than unnormalized

Rooted page rank

- Random walker starting from x
- With probability $1 - \alpha$ to goes on randomly
- With probability α it is reset to x
- Depending on α may achieve very good performance
- Equivalent to

$$RPR = (1 - \alpha)(I - \alpha A \cdot D^{-1})^{-1}$$

where $D_{ii} = k_i$ a diagonal matrix with the degrees

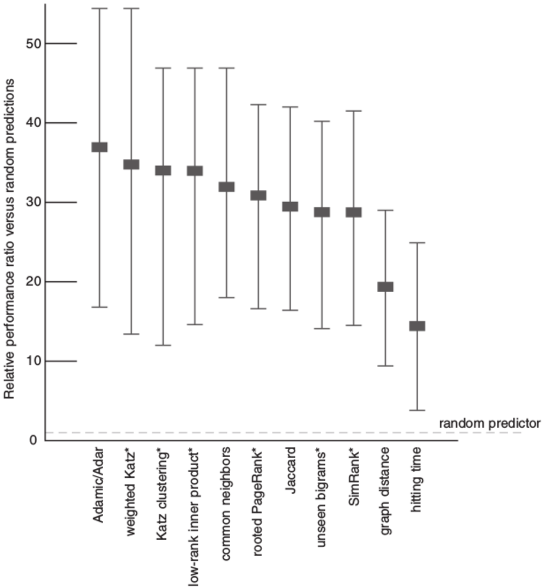
SimRank

- Two objects are similar if they are similar to two similar objects
- Check all neighboring pairs and average similarity
- Similarity is defined in a recursive way

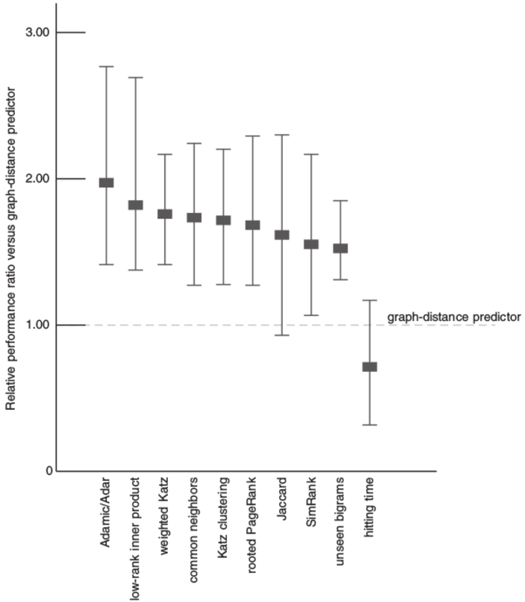
$$\text{simRank}(x, y) = \begin{cases} 1 & \text{if } x = y \\ \gamma \frac{\sum_{a \in \Gamma(x)} \sum_{b \in \Gamma(y)} \text{simRank}(a,b)}{|\Gamma(x)| \cdot |\Gamma(y)|} & \text{otherwise} \end{cases}$$

- Acceptable performance

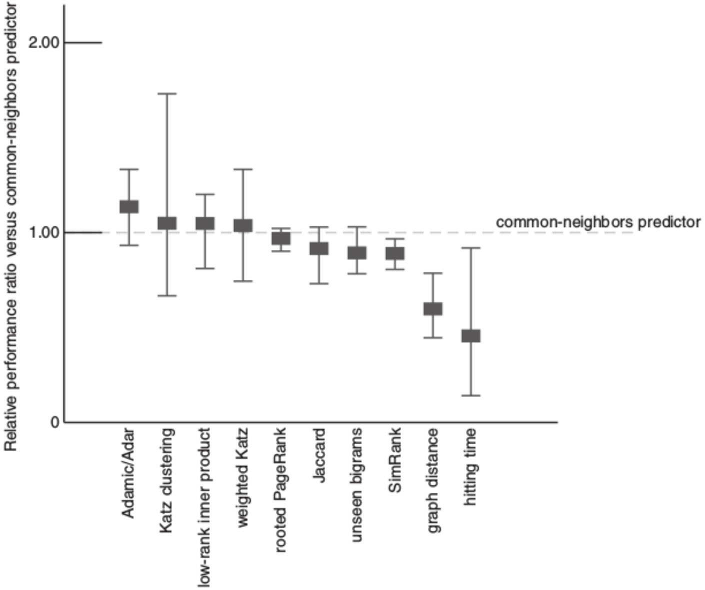
Link predictor comparison: Random prediction



Link predictor comparison: Graph distance



Link predictor comparison: Common neighbors



Link predictor comparison: Table

Predictor		astro-ph	cond-mat	gr-qc	hep-ph	hep-th
probability that a random prediction is correct		0.475%	0.147%	0.341%	0.207%	0.153%
graph distance (all distance-2 pairs)		*9.4*	*25.1*	*21.3*	*12.0*	*29.0*
common neighbors		**18.0**	**40.8**	**27.1**	**26.9**	**46.9**
preferential attachment		4.7	6.0	7.5	*15.2*	7.4
Adamic/Adar		*16.8*	**54.4**	**30.1**	**33.2**	**50.2**
Jaccard		*16.4*	**42.0**	19.8	**27.6**	*41.5*
SimRank	$\gamma = 0.8$	*14.5*	*39.0*	*22.7*	*26.0*	*41.5*
hitting time		6.4	23.7	*24.9*	3.8	13.3
hitting time—normed by stationary distribution		5.3	23.7	11.0	11.3	21.2
commute time		5.2	15.4	**33.0**	*17.0*	23.2
commute time—normed by stationary distribution		5.3	16.0	11.0	11.3	16.2
rooted PageRank	$\alpha = 0.01$	*10.8*	*27.8*	**33.0**	*18.7*	*29.1*
	$\alpha = 0.05$	*13.8*	*39.6*	**35.2**	*24.5*	*41.1*
	$\alpha = 0.15$	*16.6*	**40.8**	**27.1**	**27.5**	*42.3*
	$\alpha = 0.30$	*17.1*	**42.0**	*24.9*	**29.8**	*46.5*
	$\alpha = 0.50$	*16.8*	**40.8**	*24.2*	**30.6**	*46.5*
Katz (weighted)	$\beta = 0.05$	3.0	21.3	19.8	2.4	12.9
	$\beta = 0.005$	*13.4*	**54.4**	**30.1**	*24.0*	**51.9**
	$\beta = 0.0005$	*14.5*	**53.8**	**30.1**	**32.5**	**51.5**
Katz (unweighted)	$\beta = 0.05$	*10.9*	**41.4**	**37.4**	*18.7*	**47.7**
	$\beta = 0.005$	*16.8*	**41.4**	**37.4**	*24.1*	**49.4**
	$\beta = 0.0005$	*16.7*	**41.4**	**37.4**	*24.8*	**49.4**

Diffusion on networks

- Random walk
- On lattices we know how it works.
- In what sense will it be different?
- What are the relevant measure for the probability distribution of the walker?
- Why is it important?

Diffusion on one dimensional lattice

- Master equation, lattice and arbitrary coordinates:

$$P(i,t+1) = P(i,t) + \underbrace{\frac{1}{2}P(i-1,t) + \frac{1}{2}P(i+1,t)}_{\text{gain}} - \underbrace{P(i,t)}_{\text{loss}}$$

$$P(x,t+\Delta t) = P(x,t) + \\ + D\frac{\Delta t}{\Delta x^2}[P(x-\Delta x,t) - 2P(x,t) + P(x+\Delta x,t)]$$

- Continuum limit: diffusion equation

$$\frac{\partial P(x,t)}{\partial t} = D\frac{\partial^2 P(x,t)}{\partial x^2}$$

- Solution

$$P(x,t) = \frac{1}{\sqrt{4\pi Dt}}e^{-\frac{x^2}{4Dt}}$$

Diffusion on one dimensional lattice

- Continuum limit: diffusion equation

$$\frac{\partial P(x,t)}{\partial t} = D\frac{\partial^2 P(x,t)}{\partial x^2}$$

- Solution

$$P(x,t) = \frac{1}{\sqrt{4\pi Dt}} e^{-\frac{x^2}{4Dt}}$$

- Moments of the coordinate

$$\langle x \rangle = \int_{-\infty}^{\infty} xP(x,t)dx = 0$$

$$\langle x^2 \rangle = \int_{-\infty}^{\infty} x^2 P(x,t)dx = 2Dt$$

Random walk on lattice

- Moments of the coordinate

$$\langle x \rangle = \int_{-\infty}^{\infty} x P(x,t) dx = 0$$

$$\langle x^2 \rangle = \int_{-\infty}^{\infty} x^2 P(x,t) dx = 2Dt$$

- Probability to return to origin (Pólya theorem):

d	p_{ret}
1	1
2	1
3	0.34
4	0.19
5	0.145

Random walk on lattice

- Expected number of distinct sites visited by the random walk

d	D_t
1	$\sim \sqrt{t}$
2	$\sim t/\log t$
$3 \leq d$	$\sim t$

- The trail of the random walk is a fractal with fractal dimension $d = 2$
- In $d = 1$ the trail is self-overlapping
- In $d = 2$ it gradually fills the space
- In $d > 4$ the walk does not cross itself

Random walk on graphs

- Distance is not as important of a quantity as in lattices
- Important quantities:
  - Number of visited distinct sites
  - Probability of return
  - Probability of finding the walker on a given node
  - Probability from going one node to the other

Random walk on Watts-Strogatz graph

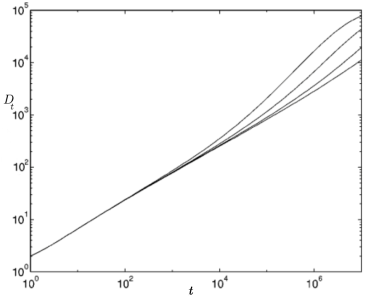
- $p = 0$: We have a one dimensional lattice
- $p = 1$: Random network is similar to trees upon trees, always new regions are explored, or infinite dimension
- Interesting regime $0 < p \ll 1$:
  - Characteristic distance between two crosslink ending: $\xi \sim 1/p$
  - One dimensional system up to $t_\xi \sim \xi^2$
  - Infinite dimension afterwards

Random walk on Watts-Strogatz graph

- Interesting regime $0 < p \ll 1$:
- Characteristic distance between two crosslink ending: $\xi \sim 1/p$
- One dimensional system up to $t_\xi \sim \xi^2$
- Infinite dimension afterwards
- Number of visited distinct sites:

$$D_t = \sqrt{t} f(t/t\xi) = \sqrt{t} f(tp^2)$$

$$f(x) = \begin{cases} \text{const} & \text{if } x \ll 1 \\ \sqrt{x} & \text{if } x \gg 1 \end{cases}$$



Random walk on graphs

- Let r be the rate of leaving a site
- The walker at node i
- Moves randomly to any neighbour, with the same probability
- Nodes are characterized by their degree k_i
- In order to land on a node with degree k from a node with degree k' the latter must have a neighbour with degree k
- The probability of going from a node with degree k' to a node with degree k is $P(k'|k)/k'$, where the former is the probability of a node with degree k' have a neighbour with degree k (assortativity)
- Master equation ($n_k(t)$ number of walkers on nodes with degree k)

$$\frac{\partial n_k(t)}{\partial t} = -rn_k(t) + rk \sum_{k'} P(k'|k) n_{k'}(t)/k'$$

Random walk on graphs

- Master equation ($n_k(t)$ number of walkers on nodes with degree k)

$$\frac{\partial n_k(t)}{\partial t} = -rn_k(t) + rk \sum_{k'} P(k'|k) n_{k'}(t)/k'$$

- The first term is the loss term: walkers leave with rate r
- The gain term is proportional to
  - Walking rate
  - The degree of the node k (walkers may come in through k links)
  - The probability that it comes from a node with degree k'

Random walk on graphs

- Master equation ($n_k(t)$ number of walkers on nodes with degree k)

$$\frac{\partial n_k(t)}{\partial t} = -r n_k(t) + rk \sum_{k'} P(k'|k) n_{k'}(t)/k'$$

- For uncorrelated networks we have

$$P(k'|k) = \frac{k' P(k')}{\langle k \rangle}$$

- Which leads to

$$\frac{\partial n_k(t)}{\partial t} = -r n_k(t) + r \frac{k}{\langle k \rangle} \sum_{k'} n_{k'}(t)$$

Random walk on graphs

- Master equation on uncorrelated graphs

$$\frac{\partial n_k(t)}{\partial t} = -rn_k(t) + r\frac{k}{\langle k \rangle}\sum_{k'} n_{k'}(t)$$

- The stationary solution (left hand side vanishes):

$$n_k = \frac{k}{\langle k \rangle}\frac{n}{N},$$

where n is the number of walkers. Or with probability

$$p_k = \frac{k}{\langle k \rangle}\frac{1}{N},$$

where p_k is the probability of finding the walker at a node with degree k

Random walk on graphs

- The probability of finding the walker at a node with degree k

$$p_k = \frac{k}{\langle k \rangle} \frac{1}{N},$$

- It is more likely to find the walkers at hubs than in a dead end
- *There are more drunk people at Deák tér and at Nyugati than e.g. at Gárdonyi tér.*