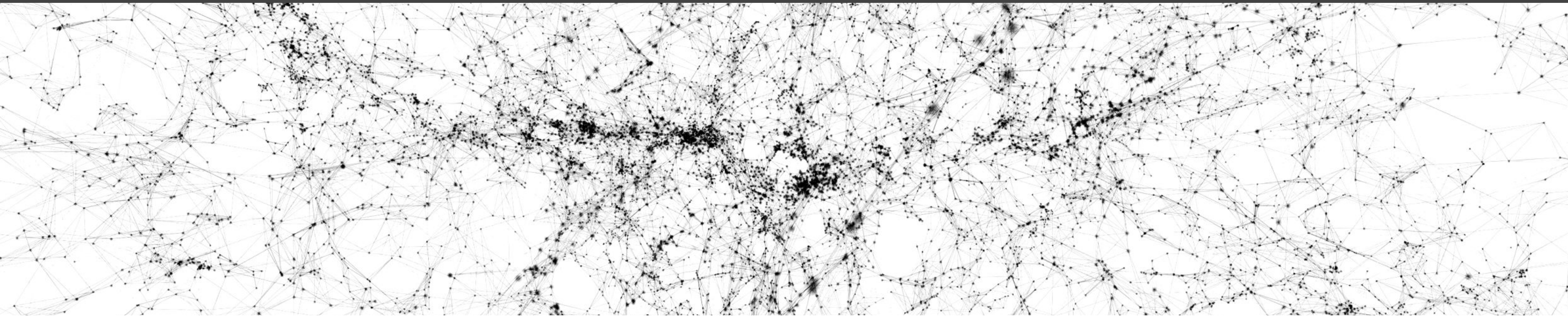
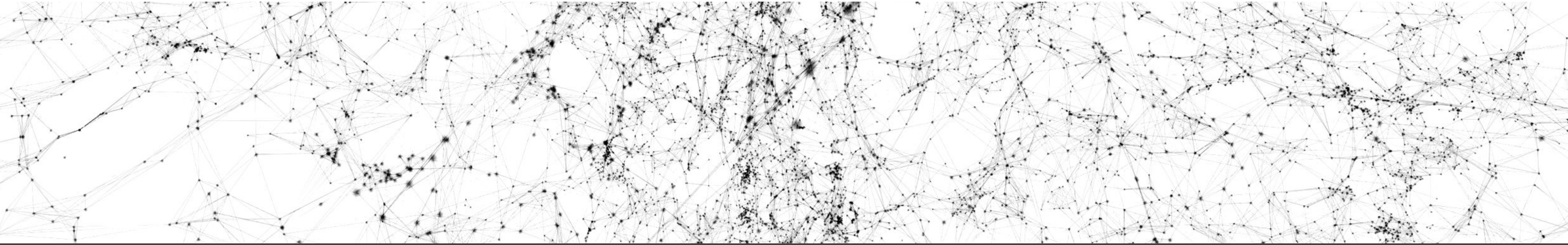


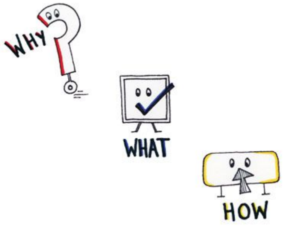
Prediction

Analysis of complex interconnected data



Quick Notes

- Second assignment is due on Oct 4th
  - Submit 2 files (report.pdf, code.zip) as a Group (pairs or two or individual) in Mycourses
- **Tue., Oct. 19, 2021:** Project Proposal Presentations
  - Why & What: Introduction and Motivation, Related Work, Problem Definition, Dataset Description
  - Writeup: 2 pages, due Oct 20th [8pt]
  - Presentation: 2 mins (2-3 slides), slides due Oct 18th [2pt]
  - Email the slides to the course email, use **Google Slides**
  - We will merge them all together, and you will go over it in cla
- Any questions?



Deadlines

- assignment 1 due on Sep. 20th
- assignment 2 due on Oct. 4th
- assignment 3 due on Oct. 18th
- project proposal slides due on Oct. 18th
- project proposal due on Oct. 20th
- Reviews (first round) due on Oct. 27th
- project proposal slides due on Nov. 3rd
- project progress report due on Nov. 5th
- Reviews (second round) due on Nov. 12th
- project final report slides due on Nov. 29th
- project final report due on Dec. 7th
- Reviews (third round) due on Dec. 14th
- project revised report and rebuttal due on Dec. 20th
- note: dates are tentative, subject to change

Common prediction tasks

- Link Prediction
- Node Classification
- Graph Classification

What is unsupervised node classification?

Examples:

https://paperswithcode.com/task/link-prediction

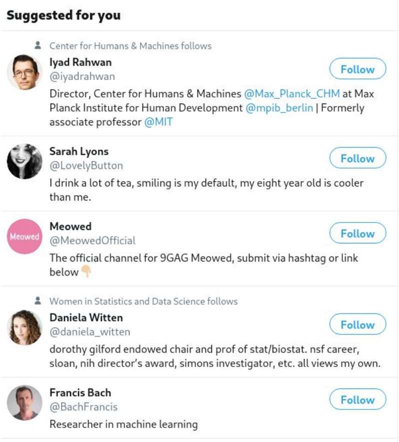
https://paperswithcode.com/task/node-classification

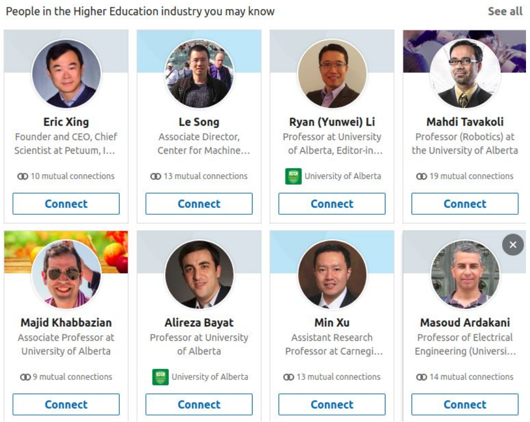
https://paperswithcode.com/task/graph-classification

Link Prediction

Given a graph G(V,E) predict future/missing links between nodes

- Modeling of network evolution
- Predict likely interactions, not explicitly observed (e.g. terrorist network monitoring)
- Link recommendation: “friend” suggestion in social networks

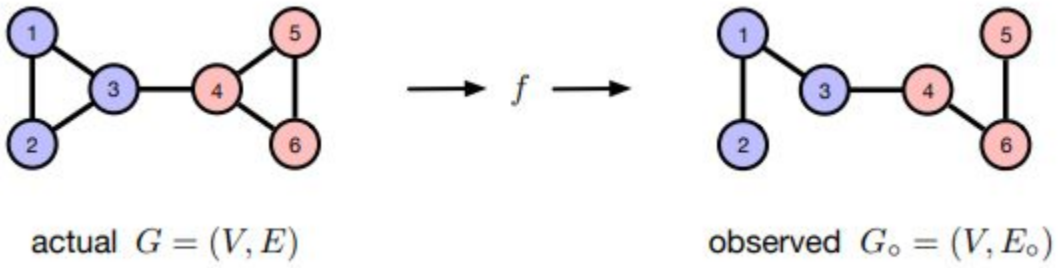




Predicting missing links

Only a subset of edges are observed

Sparse ⇒ searching for O(n) needles in a Θ(n^2) haystack



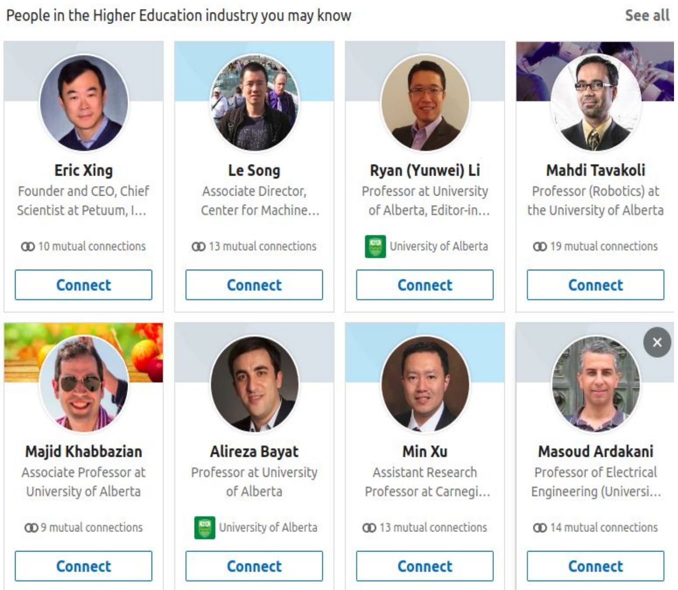
From Clauset's Slides

Link prediction

Score(i , j)

local topological predictors

- Number of common neighbors
- Number of shortest paths
- Product of degree
- Same cluster
- etc.



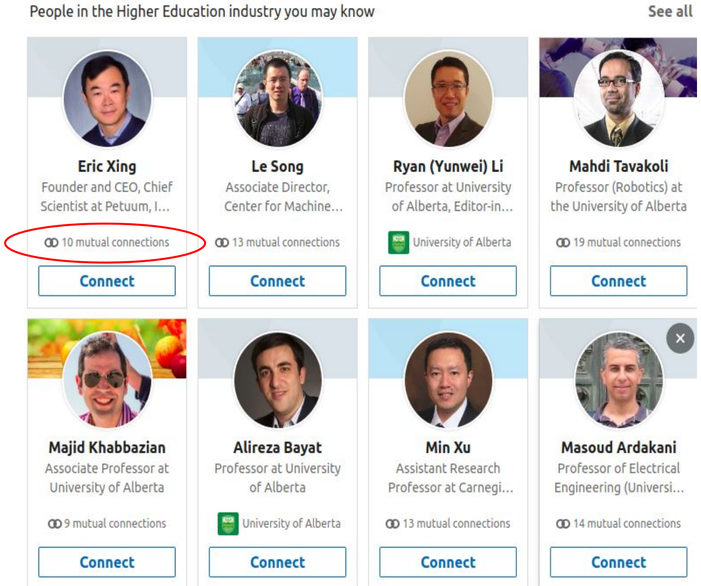
From Clauset's Slides

Link prediction

Score(i , j)

local topological predictors

- **Number of common neighbors**
- Number of shortest paths
- Product of degree
- Same cluster
- etc.



From Clauset's Slides

Link prediction

Jaccard coefficient

$$\text{Jaccard}(i, j) = \frac{|\nu(i) \cap \nu(j)|}{|\nu(i) \cup \nu(j)|}$$

score(i, j) = Jaccard(i, j) + Uniform(0, **ε**)

What happens to the network if we add edges based on this?

[From Clauset's Slides](#)

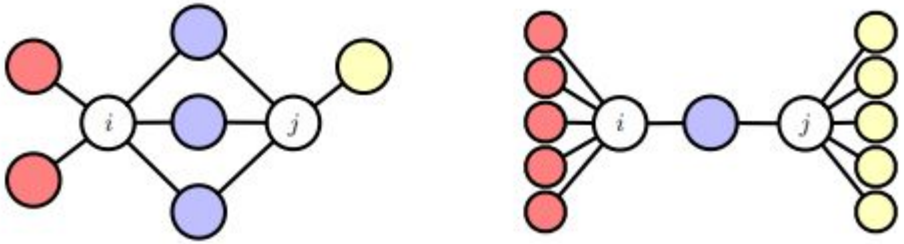
Link prediction

Jaccard coefficient

$$\text{Jaccard}(i, j) = \frac{|\nu(i) \cap \nu(j)|}{|\nu(i) \cup \nu(j)|}$$

score(i, j) = Jaccard(i, j) + Uniform(0, **ε**)

What happens to the network if we add edges based on this? Closes triangles



Example: Jaccard(i, j), what is it for this example ?

[From Clauset's Slides]

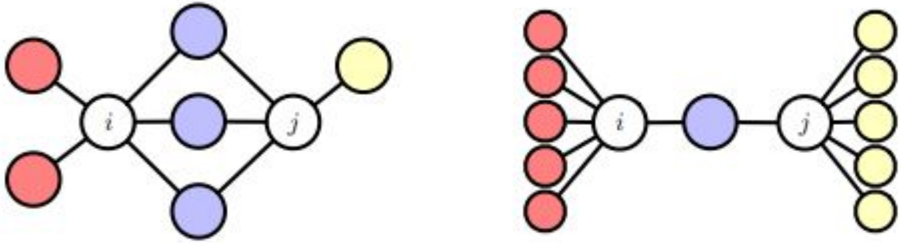
Link prediction

Jaccard coefficient

$$\text{Jaccard}(i, j) = \frac{|\nu(i) \cap \nu(j)|}{|\nu(i) \cup \nu(j)|}$$

score(i, j) = Jaccard(i, j) + Uniform(0, **ε**)

What happens to the network if we add edges based on this? Closes triangles



Example: Jaccard(i, j) of **0.50** (3/6) vs **0.091** (1/11)

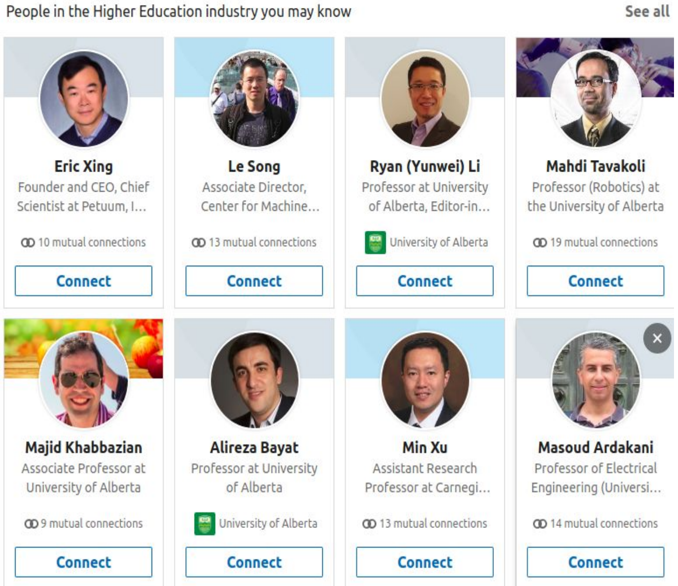
From Clauset's Slides

Link prediction

Score(i , j)

local topological predictors

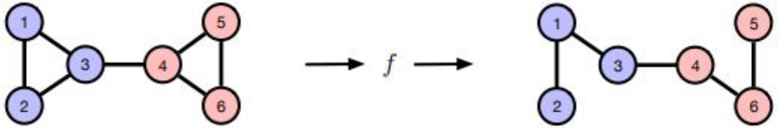
- Number of common neighbors
- Number of shortest paths
- **Product of degree**
    - nodes with high degrees are likely themselves to be connected
- Same cluster
- etc.



[From Clauset's Slides]

Link prediction

degree product: score(i, j) = d_i d_j + Uniform(0, **ε**)



actual $G = (V, E)$ observed $G_\circ = (V, E_\circ)$

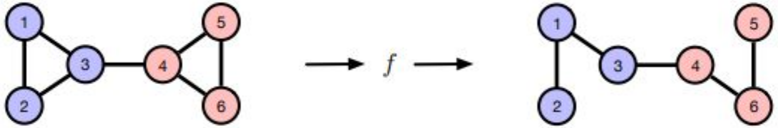
i	j	Jaccard score(i, j)
4	**5**	$1/2 + r$
2	**3**	$1/2 + r$
3	6	$1/3 + r$
1	4	$1/3 + r$
1	5	r
1	6	r
2	6	r
2	4	r
2	5	r
3	5	r

i	j	degree product score(i, j)
1	4	$4 + r$
1	6	$4 + r$
3	6	$4 + r$
1	5	$2 + r$
2	**3**	$2 + r$
2	6	$2 + r$
2	4	$2 + r$
3	5	$2 + r$
4	**5**	$2 + r$
2	5	$1 + r$

From Clauset's Slides

Link prediction

degree product: score(i, j) = $d_i\ d_j$ + Uniform(0, **ε**)



actual $G = (V, E)$ observed $G_\circ = (V, E_\circ)$

i	j	Jaccard score(i, j)
4	**5**	$1/2 + r$
2	**3**	$1/2 + r$
3	6	$1/3 + r$
1	4	$1/3 + r$
1	5	r
1	6	r
2	6	r
2	4	r
2	5	r
3	5	r

i	j	degree product score(i, j)
1	4	$4 + r$
1	6	$4 + r$
3	6	$4 + r$
1	5	$2 + r$
2	**3**	$2 + r$
2	6	$2 + r$
2	4	$2 + r$
3	5	$2 + r$
4	**5**	$2 + r$
2	5	$1 + r$

Which one is more accurate?

From Clauset's Slides

Measuring performance for ranked output

We can consider a threshold and convert it to a **binary classification**

{every i, j is either a missing link or not}

Accuracy = (TP+TN) / (P+N)
Precision = TP / RP
Recall = TP / P {also called sensitivity}
F1 score = 2. Precision x Recall / (Precision + Recall)
{Harmonic mean}

Miss rate = FN / P
Fallout = FP / N {also called false positive rate}
False discovery rate = FP / RP
Selectivity = TN / N {also called specificity}
False omission rate = FN / RN
Negative predictive value = TN / RN

	Truth		Σ
Results	**TP**	**FP**	**RP**
	FN	**TN**	**RN**
Σ	**P**	**N**	

i	j	Jaccard score(i,j)
4	**5**	$1/2+r$
2	**3**	$1/2+r$
3	6	$1/3+r$
1	4	$1/3+r$
1	5	r
1	6	r
2	6	r
2	4	r
2	5	r
3	5	r

Predict edge

Measuring performance for ranked output

We can consider a threshold and convert it to a **binary classification**

{every i, j is either a missing link or not}

	Truth		Σ
Results	TP	FP	RP
	FN	TN	RN
Σ	P	N	

Accuracy = (TP+TN) / (P+N)
Precision = TP / RP
Recall = TP / P {also called sensitivity}
F1 score = 2. Precision x Recall / (Precision + Recall) {Harmonic mean}

Miss rate = FN / P
Fallout = FP / N {also called false positive rate}
False discovery rate = FP / RP
Selectivity = TN / N {also called specificity}
False omission rate = FN / RN
Negative predictive value = TN / RN

Measures depend on the threshold

The tradeoff between precision and recall

no false positive
also no true positive

i	j	Jaccard score(i,j)
4	**5**	$1/2+r$
2	**3**	$1/2+r$
3	6	$1/3+r$
1	4	$1/3+r$
1	5	r
1	6	r
2	6	r
2	4	r
2	5	r
3	5	r

Predict edge

no false negative
also no true negative

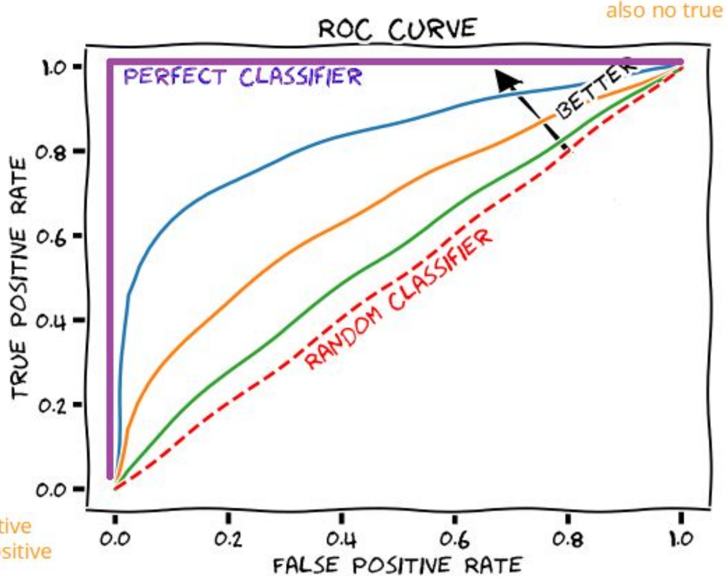
Threshold invariant: ROC & AUC

Receiver Operating Characteristic (ROC) as a function of prediction threshold

TPR(t) = TP(t)/P (recall, sensitivity at t)
FPR(t) = FP(t)/N (fallout, false alarm at t)

Area Under the Curve (AUC) of ROC
gives the probability of ranking a random positive edge higher than a random negative edge

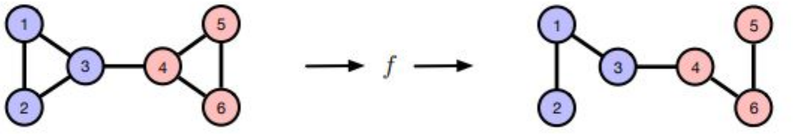
no false negative
also no true negative



no false positive
also no true positive

Threshold invariant: ROC & AUC, example

binary classification ⇒ every candidate i, j is either a missing link or not



actual $G = (V, E)$ → f → observed $G_\circ = (V, E_\circ)$

	i	j	Jaccard score(i,j)	i	j	degree product score(i,j)
link	**4**	**5**	$1/2+r$	1	4	$4+r$
	2	**3**	$1/2+r$	1	6	$4+r$
	3	6	$1/3+r$	3	6	$4+r$
	1	4	$1/3+r$	1	5	$2+r$
threshold						
nonlink	1	5	r	**2**	**3**	$2+r$
	1	6	r	2	6	$2+r$
	2	6	r	2	4	$2+r$
	2	4	r	3	5	$2+r$
	2	5	r	**4**	**5**	$2+r$
	3	5	r	2	5	$1+r$

TPR = TP/P (**recall**, sensitivity)
FPR = FP/N (**fallout**, false alarm)

Measuring performance

binary classification ⇒ every candidate i, j is either a missing link or not

TPR = TP/P (**recall**, sensitivity)
Links above threshold / Total Positives

FPR = FP/N (**fallout**, false alarm)
Non-links above threshold / Total Negatives

TPR = ?, FPR = ?

	Truth		Σ
results	TP = 2	FP = 2	RP = 4
	FN = 0	TN = 6	RN = 6
Σ	P = 2	N = 8	

Jaccard

TPR = ?, FPR = ?

	Truth		Σ
results	TP = 0	FP = 4	RP = 4
	FN = 2	TN= 4	RN = 6
Σ	P = 2	N = 8	

degree product

	i	j	Jaccard score(i,j)	i	j	degree product score(i,j)
link	**4**	**5**	$1/2+r$	1	4	$4+r$
	2	**3**	$1/2+r$	1	6	$4+r$
	3	6	$1/3+r$	3	6	$4+r$
	1	4	$1/3+r$	1	5	$2+r$
threshold						
nonlink	1	5	r	**2**	**3**	$2+r$
	1	6	r	2	6	$2+r$
	2	6	r	2	4	$2+r$
	2	4	r	3	5	$2+r$
	2	5	r	**4**	**5**	$2+r$
	3	5	r	2	5	$1+r$

Measuring performance

binary classification ⇒ every candidate i, j is either a missing link or not

TPR = TP/P (**recall**, sensitivity)
Links above threshold / Total Positives

FPR = FP/N (**fallout**, false alarm)
Non-links above threshold / Total Negatives

	i	j	Jaccard score(i,j)	i	j	degree product score(i,j)
link	**4**	**5**	$1/2+r$	1	4	$4+r$
	2	**3**	$1/2+r$	1	6	$4+r$
	3	6	$1/3+r$	3	6	$4+r$
	1	4	$1/3+r$	1	5	$2+r$
threshold						
nonlink	1	5	r	**2**	**3**	$2+r$
	1	6	r	2	6	$2+r$
	2	6	r	2	4	$2+r$
	2	4	r	3	5	$2+r$
	2	5	r	**4**	**5**	$2+r$
	3	5	r	2	5	$1+r$

TPR = 2/2 = 1, FPR = 2/8=0.25

	Truth		Σ
results	TP = 2	FP = 2	RP = 4
	FN = 0	TN = 6	RN = 6
Σ	P = 2	N = 8	

Jaccard

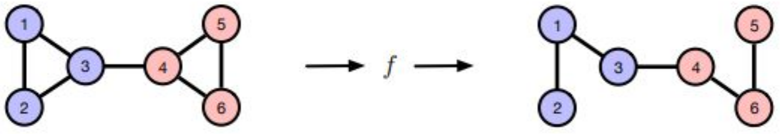
TPR = 0/2 = 0, FPR = 4/8=0.5

	Truth		Σ
results	TP = 0	FP = 4	RP = 4
	FN = 2	TN= 4	RN = 6
Σ	P = 2	N = 8	

degree product

Threshold invariant: ROC & AUC, example

binary classification ⇒ every candidate i, j is either a missing link or not



actual $G = (V, E)$ observed $G_\circ = (V, E_\circ)$

i	j	Jaccard score(i,j)	i	j	degree product score(i,j)
4	**5**	$1/2 + r$	1	4	$4 + r$
2	**3**	$1/2 + r$	1	6	$4 + r$
3	6	$1/3 + r$	3	6	$4 + r$
1	4	$1/3 + r$	1	5	$2 + r$
1	5	r	**2**	**3**	$2 + r$
1	6	r	2	6	$2 + r$
2	6	r	2	4	$2 + r$
2	4	r	3	5	$2 + r$
2	5	r	**4**	**5**	$2 + r$
3	5	r	2	5	$1 + r$

threshold

TPR = fraction of links above threshold

FPR = fraction of non-links above threshold

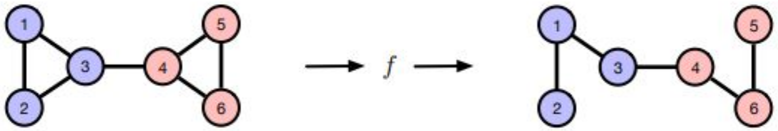
AUC = Σ_t TPR(t) (FPR (t)- FPR (t-1))

box-rule approximation

From Clauset's Slides

Threshold invariant: ROC & AUC, example

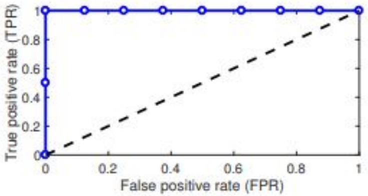
binary classification ⇒ every candidate i, j is either a missing link or not



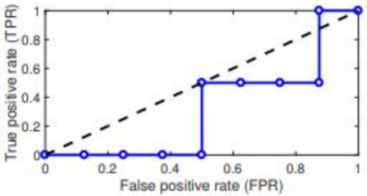
actual $G = (V, E)$ observed $G_\circ = (V, E_\circ)$

i	j	Jaccard score(i,j)
4	**5**	$1/2 + r$
2	**3**	$1/2 + r$
3	6	$1/3 + r$
1	4	$1/3 + r$
1	5	r
1	6	r
2	6	r

i	j	degree product score(i,j)
1	4	$4 + r$
1	6	$4 + r$
3	6	$4 + r$
1	5	$2 + r$
2	**3**	$2 + r$
2	6	$2 + r$
2	4	$2 + r$



Jaccard coefficient, AUC $= 1.00$



degree product, AUC $= 0.31$

From Clauset's Slides

Topological Link Predictors

CN	common neighbors of i, j
SP	shortest path between i, j
LHN	Leicht-Holme-Newman index of neighbor sets of i, j
PPR	j-th entry of the personalized page rank of node i
PA	preferential attachment (degree product) of i, j
JC	Jaccard's coefficient of neighbor sets of i, j
AA	Adamic/Adar index of i, j
RA	resource allocation index of i, j
LRA	entry i, j in low rank approximation (LRA) via singular value decomposition (SVD)
dLRA	dot product of columns i and j in LRA via SVD for each pair of nodes i, j
mLRA	average of entries i and j's neighbors in low rank approximation
LRA-approx	an approximation of LRA
dLRA-approx	an approximation of dLRA
mLRA-approx	an approximation of mLRA
LCC_i, LCC_j	local clustering coefficients for i and j
AND_i, AND_j	average neighbor degrees for i and j
SPBC_i, SPBC_j	shortest-path betweenness centralities for i and j
CC_i, CC_j	closeness centralities for i and j
DC_i, DC_j	degree centralities for i and j
EC_i, EC_j	eigenvector centralities for i and j
KC_i, KC_j	Katz centralities for i and j
LNT_i, LNT_j	local number of triangles for i and j
PR_i, PR_j	Page rank values for i and j
LC_i, LC_j	load centralities for i and j

There are many **alternatives** to give a score to a given pair of nodes (i,j)

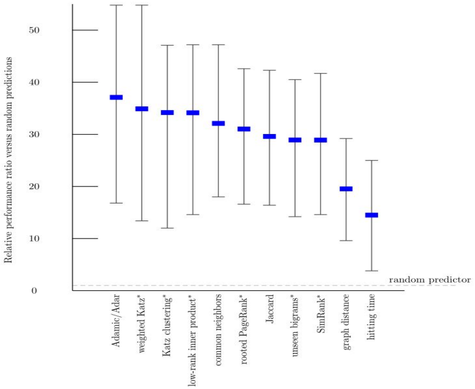
Many also used as similarity measures since homophily is a strong force in link creation

Ghasemian A, Hosseinmardi H, Galstyan A, Airoldi EM, Clauset A. Stacking Models for Nearly Optimal Link Prediction in Complex Networks. PNAS 2020.

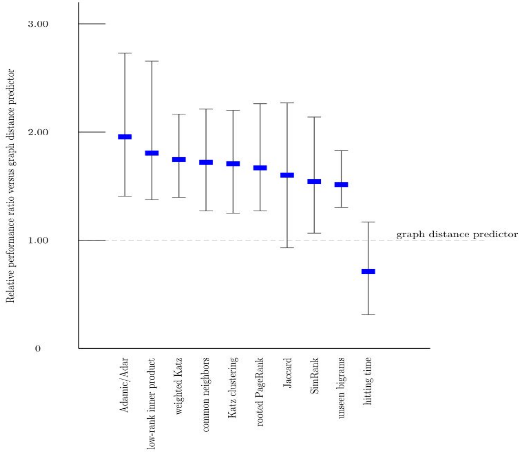
See also A Survey of Link Prediction in Complex Networks, 2015 & Link prediction in complex networks: A survey, 2011

Common neighbour is a decent predictor

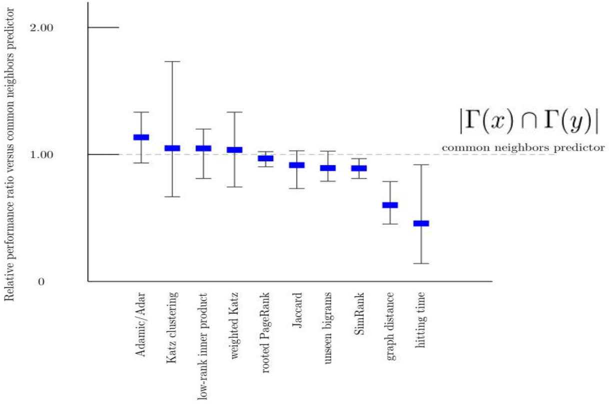
Random



Graph distance



Common neighbor



Adamic/Adar: a notable measure

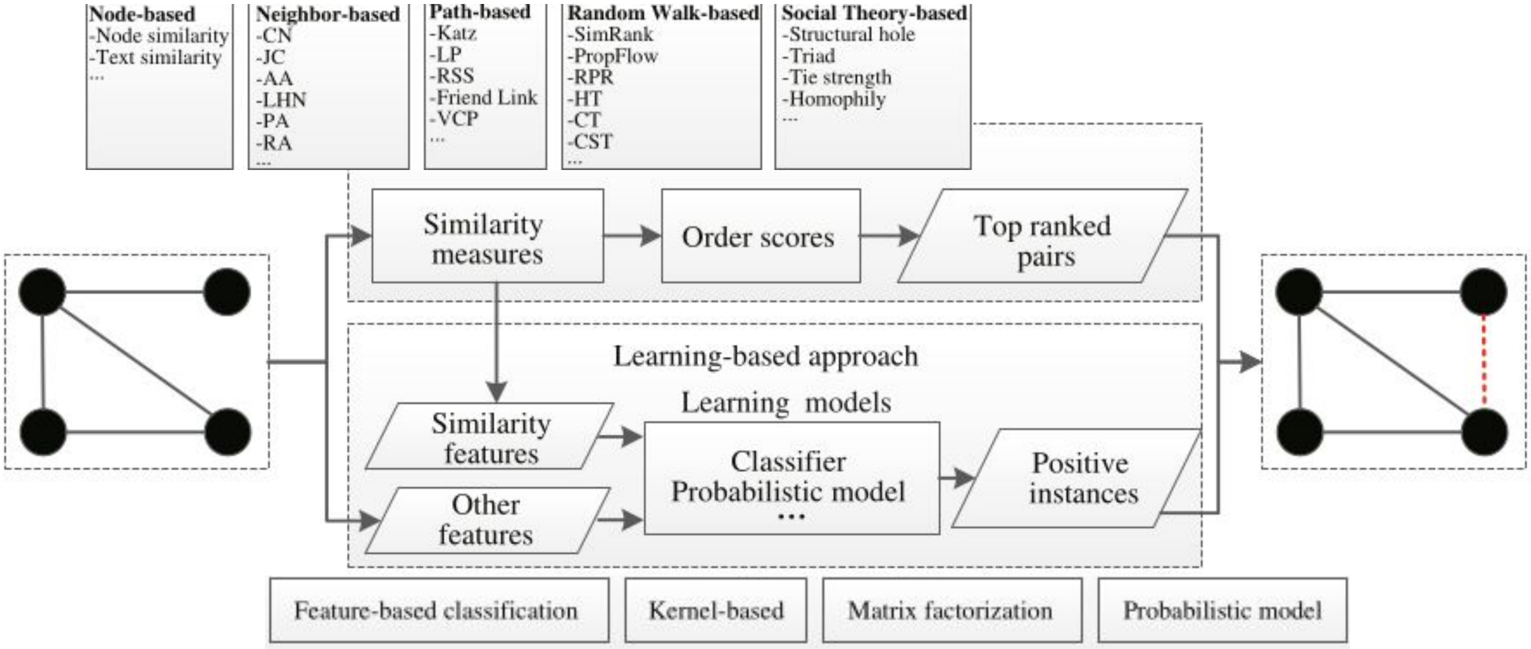
Sums over size inverse of log of degree of common neighbours

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

Liben-Nowell D, Kleinberg J. The link-prediction problem for social networks. Journal of the American society for information science and technology. 2007 May;58(7):1019-31.

We can learn to link

Wang P, Xu B, Wu Y, Zhou X. Link prediction in social networks: the state-of-the-art. Science China Information Sciences. 2015 Jan 1;58(1):1-38.



Learn the features as well

Instead of explicit features, automatically learn a "heuristic" that suits the current network e.g. by extracting a local enclosing subgraph around it, and uses a GNN to learn general graph structure features for link prediction

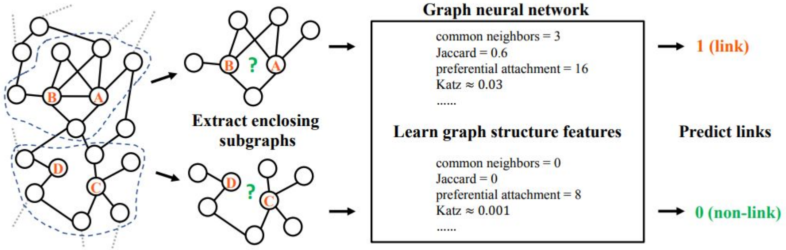


Table 1: Comparison with heuristic methods (AUC).

Data	CN	Jaccard	PA	AA	RA	Katz	PR	SR	ENS	WLK	WLNM	SEAL
USAir	93.80±1.22	89.79±1.61	88.84±1.45	95.06±1.03	95.77±0.92	92.88±1.42	94.67±1.08	78.89±2.31	88.96±1.44	**96.63**±0.73	95.95±1.10	**96.62**±0.72
NS	94.42±0.95	94.43±0.93	68.65±2.03	94.45±0.93	94.45±0.93	94.85±1.10	94.89±1.08	94.79±1.08	97.64±0.25	98.57±0.51	98.61±0.49	**98.85**±0.47
PB	92.04±0.35	87.41±0.39	90.14±0.45	92.36±0.34	92.46±0.37	92.92±0.35	93.54±0.41	77.08±0.80	90.15±0.45	93.83±0.59	93.49±0.47	**94.72**±0.46
Yeast	89.37±0.61	89.32±0.60	82.20±1.02	89.43±0.62	89.45±0.62	92.24±0.61	92.76±0.55	91.49±0.57	82.36±1.02	95.86±0.54	95.62±0.52	**97.91**±0.52
C.ele	85.13±1.61	80.19±1.64	74.79±2.04	86.95±1.40	87.49±1.41	86.34±1.89	**90.32**±1.49	77.07±2.00	74.94±2.04	89.72±1.67	86.18±1.72	**90.30**±1.35
Power	58.80±0.88	58.79±0.88	44.33±1.02	58.79±0.88	58.79±0.88	65.39±1.59	66.00±1.59	76.15±1.06	79.52±1.78	82.41±3.43	84.76±0.98	**87.61**±1.57
Router	56.43±0.52	56.40±0.52	47.58±1.47	56.43±0.51	56.43±0.51	38.62±1.35	38.76±1.39	37.40±1.27	47.58±1.48	87.42±2.08	94.41±0.88	**96.38**±1.45
E.coli	93.71±0.39	81.31±0.61	91.82±0.58	95.36±0.34	95.95±0.35	93.50±0.44	95.57±0.44	62.49±1.43	91.89±0.58	96.94±0.29	97.21±0.27	**97.64**±0.22

Table 2: Comparison with latent feature methods (AUC).

Data	MF	SBM	N2V	LINE	SPC	VGAE	SEAL
USAir	94.08±0.80	94.85±1.14	91.44±1.78	81.47±10.71	74.22±3.11	89.28±1.99	**97.09**±0.70
NS	74.55±4.34	92.30±2.26	91.52±1.28	80.63±1.90	89.94±2.39	94.04±1.64	**97.71**±0.93
PB	94.30±0.53	93.90±0.42	85.79±0.78	76.95±2.76	83.96±0.86	90.70±0.53	**95.01**±0.34
Yeast	90.28±0.69	91.41±0.60	93.67±0.46	87.45±3.33	93.25±0.40	93.88±0.21	**97.20**±0.64
C.ele	85.90±1.74	86.48±2.60	84.11±1.27	69.21±3.14	51.90±2.57	81.80±2.18	**89.54**±2.04
Power	50.63±1.10	66.57±2.05	76.22±0.92	55.63±1.47	**91.78**±0.61	71.20±1.65	84.18±1.82
Router	78.03±1.63	85.65±1.93	65.46±0.86	67.15±2.10	68.79±2.42	61.51±1.22	**95.68**±1.22
E.coli	93.76±0.56	93.82±0.41	90.82±1.49	82.38±2.19	94.92±0.32	90.81±0.63	**97.22**±0.28

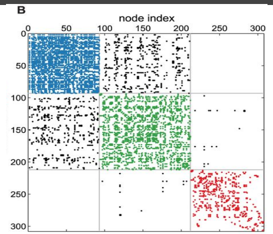
Zhang M, Chen Y. Link prediction based on graph neural networks. NeurIPS 2018

Link prediction approaches

- local topological predictors
- Global predictors, learning
  - **Model-based**
    - Fit a model to data by maximizing likelihood which is defined in terms of edge probabilities ⇒ $\Pr(i \to j \mid \theta)$
      - E.g. stochastic block model
  - **Optimization-based**
    - Adding an edge increases a measure, e.g. Q modularity
  - **Embedding-based**
    - Proximity in the embedded space {put connected nodes close together}

From Clauset's Slides

Model based link prediction

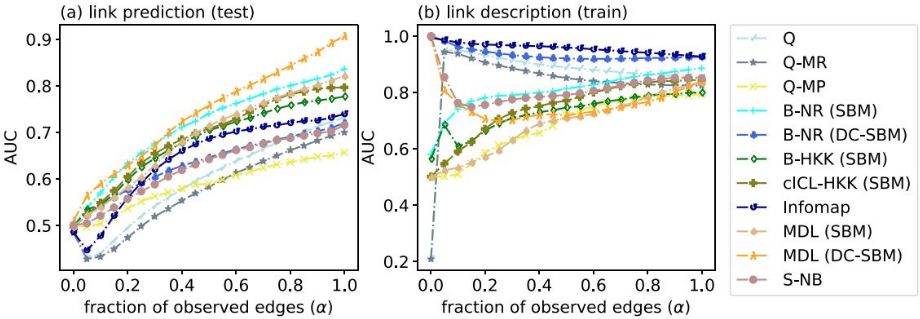


Infer Pr (i → j | θ) for each candidate pair based on maximizing likelihood of the observed graph or other optimizations

SBM variants, Q-modularity and Infomap, e.g. Number of edges between module node i and j belong to divided by maximum possible edges between those modules

Assume links are within modules

Abbreviation	Description
Q	modularity, Newman-Girvan
Q-MR	modularity, Newman's multiresolution
Q-MP	modularity, message passing
B-NR (SBM)	Bayesian stochastic block model, Newman and Reinert
B-NR (DC-SBM)	Bayesian degree-corrected stochastic block model, Newman and Reinert
B-HKK (SBM)	Bayesian stochastic block model, Hayashi, Konishi and Kawamoto
cICL-HKK (SBM)	Corrected integrated classification likelihood, stochastic block model
Infomap	Map equation
MDL (SBM)	Minimum description length, stochastic block model
MDL (DC-SBM)	Minimum description length, degree-corrected stochastic block model
S-NB	Spectral with non-backtracking matrix

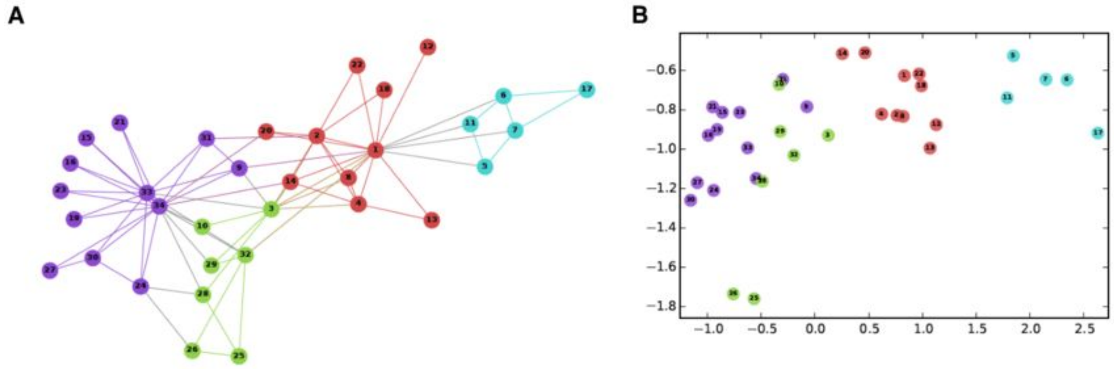


Evaluating Overfit and Underfit in Models of Network Community Structure, TKDE 2020

Embedding based link prediction

G ⟶ embed in vector space ⟶ distance in the embedded space

Node embedding methods derive a vector representation per each node in the graph so that connected nodes have similar vectors ⇒ close in the embedded space means more likely to be linked



Matrix factorization based
e.g. svd
⇒
Deep learning methods
e.g. deepwalk

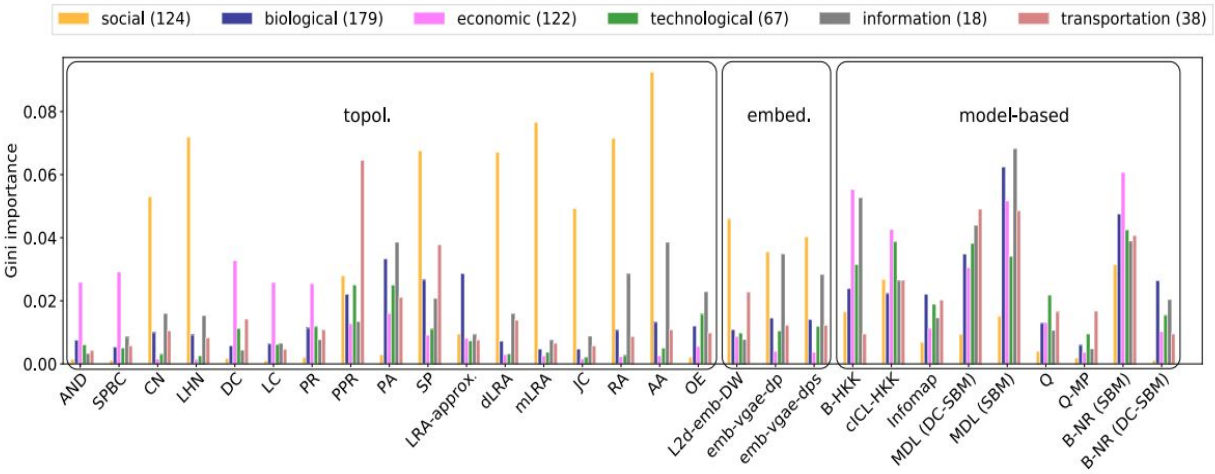
What embedding to use? [Graph Representation Learning]

See [A Tutorial on Network Embeddings], 2018

Which one is the best?

550 structurally diverse networks from six scientific domains

no one predictor or family is best, or worst, across all realistic inputs



Ghasemian A, Hosseinmardi H, Galstyan A, Airoldi EM, Clauset A. Stacking Models for Nearly Optimal Link Prediction in Complex Networks. PNAS 2020.

Which one to choose?

Stack the models

learns to apply the best individual predictor according to the input's characteristics

algorithm	AUC	precision	recall
Q	0.7 ± 0.14	0.14 ± 0.17	0.67 ± 0.15
Q-MR	0.67 ± 0.15	0.12 ± 0.17	0.63 ± 0.13
Q-MP	0.64 ± 0.15	0.09 ± 0.11	0.59 ± 0.17
B-NR (SBM)	0.81 ± 0.13	0.13 ± 0.12	0.65 ± 0.22
B-NR (DC-SBM)	0.7 ± 0.2	0.12 ± 0.12	0.61 ± 0.24
cICL-HKK	0.79 ± 0.13	0.14 ± 0.14	0.58 ± 0.25
B-HKK	0.77 ± 0.13	0.11 ± 0.1	0.51 ± 0.26
Infomap	0.73 ± 0.14	0.12 ± 0.12	0.68 ± 0.13
MDL (SBM)	0.79 ± 0.15	0.14 ± 0.13	0.57 ± 0.3
MDL (DC-SBM)	0.84 ± 0.1	0.13 ± 0.11	0.78 ± 0.12
S-NB	0.71 ± 0.19	0.12 ± 0.13	0.66 ± 0.17
mean model-based	0.74 ± 0.16	0.12 ± 0.13	0.63 ± 0.21
mean indiv. topol.	0.6 ± 0.13	0.09 ± 0.16	0.53 ± 0.35
mean indiv. topol. & model	0.63 ± 0.15	0.09 ± 0.16	0.55 ± 0.33
emb-DW	0.63 ± 0.23	0.17 ± 0.19	0.42 ± 0.35
emb-vgae	0.69 ± 0.19	0.05 ± 0.05	0.69 ± 0.21
all topol.	0.86 ± 0.11	0.42 ± 0.33	0.44 ± 0.32
all model-based	0.83 ± 0.12	0.39 ± 0.34	0.3 ± 0.29
all embed.	0.77 ± 0.16	0.32 ± 0.32	0.32 ± 0.31
all topol. & model	0.87 ± 0.1	0.48 ± 0.36	0.35 ± 0.35
all topol. & embed.	0.84 ± 0.13	0.4 ± 0.34	0.39 ± 0.33
all model & embed.	0.84 ± 0.13	0.36 ± 0.32	0.36 ± 0.31
all topol., model & embed.	0.85 ± 0.14	0.42 ± 0.34	0.39 ± 0.33

550 structurally diverse networks from six scientific domains

Ghasemian A, Hosseinmardi H, Galstyan A, Airoldi EM, Clauset A. Stacking Models for Nearly Optimal Link Prediction in Complex Networks. arXiv preprint arXiv:1909.07578. 2019 Sep 17.

Which one to choose?

Stack the models

learns to apply the best individual predictor according to the input's characteristics

Near perfect in social networks

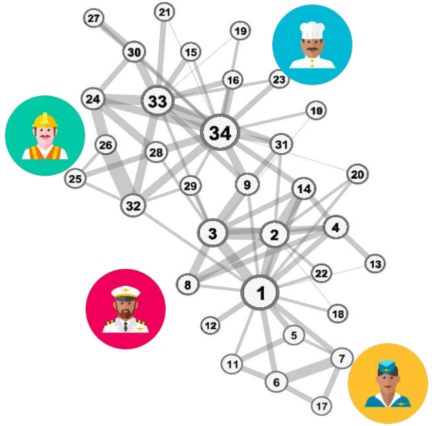
Algorithm	AUC	Precision	Recall
Q	0.89 ± 0.07	0.42 ± 0.13	0.85 ± 0.08
Q-MR	0.87 ± 0.07	0.38 ± 0.16	0.78 ± 0.07
Q-MP	0.86 ± 0.08	0.25 ± 0.07	0.83 ± 0.09
B-NR (SBM)	0.93 ± 0.06	0.3 ± 0.08	0.85 ± 0.12
B-NR (DC-SBM)	0.93 ± 0.07	0.28 ± 0.08	0.88 ± 0.08
cICL-HKK	0.93 ± 0.08	0.34 ± 0.1	0.85 ± 0.14
B-HKK	0.88 ± 0.07	0.17 ± 0.05	0.79 ± 0.17
Infomap	0.91 ± 0.04	0.29 ± 0.08	0.83 ± 0.05
MDL (SBM)	0.94 ± 0.07	0.31 ± 0.09	0.87 ± 0.16
MDL (DC-SBM)	0.93 ± 0.09	0.26 ± 0.09	0.89 ± 0.11
S-NB	0.94 ± 0.07	0.3 ± 0.1	0.87 ± 0.08
mean model-based	0.91 ± 0.08	0.3 ± 0.12	0.84 ± 0.12
mean indiv. topol.	0.64 ± 0.19	0.2 ± 0.27	0.56 ± 0.33
mean indiv. topol. & model	0.7 ± 0.21	0.22 ± 0.25	0.62 ± 0.32
emd-DW	0.95 ± 0.1	0.45 ± 0.16	0.92 ± 0.13
emb-vgae	0.95 ± 0.08	0.09 ± 0.02	0.96 ± 0.09
all topol.	0.97 ± 0.08	0.89 ± 0.21	0.88 ± 0.2
all model-based	0.95 ± 0.07	0.76 ± 0.2	0.68 ± 0.17
all embed.	0.95 ± 0.11	0.75 ± 0.23	0.74 ± 0.23
all topol. & model	0.98 ± 0.06	0.89 ± 0.22	0.88 ± 0.19
all topol. & embed.	0.96 ± 0.1	0.86 ± 0.22	0.83 ± 0.25
all model & embed.	0.96 ± 0.09	0.78 ± 0.21	0.74 ± 0.22
all topol., model & embed.	0.97 ± 0.09	0.86 ± 0.23	0.84 ± 0.23

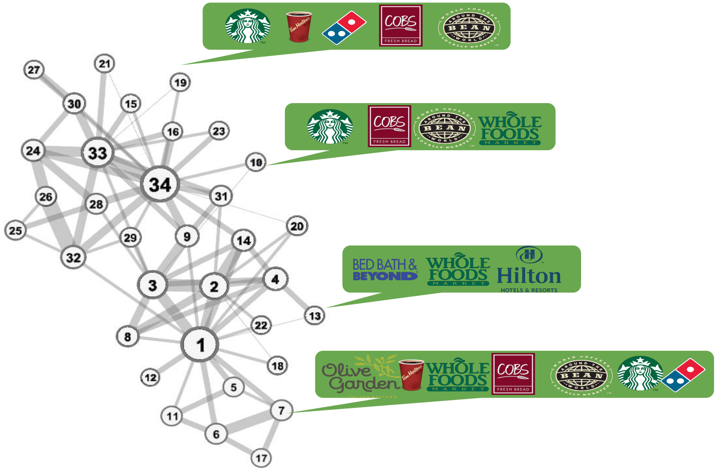
Ghasemian A, Hosseinmardi H, Galstyan A, Airoldi EM, Clauset A. Stacking Models for Nearly Optimal Link Prediction in Complex Networks. arXiv preprint arXiv:1909.07578. 2019 Sep 17.

Link Prediction in Attributed Graphs

Individual characteristics or activity (attributes) & relations (graph)

Annotated networks, **metadata** on nodes, **side** information





characteristics

age, occupation, salary, sex, etc.

activity & interest

check-ins, page-likes, group memberships, movies

Attributed Graphs

Interplay between attributes and relations, a positive feedback loop derived by two social theories:

- Social Selection
  - Similarity of individuals' characteristics motivates them to form relations

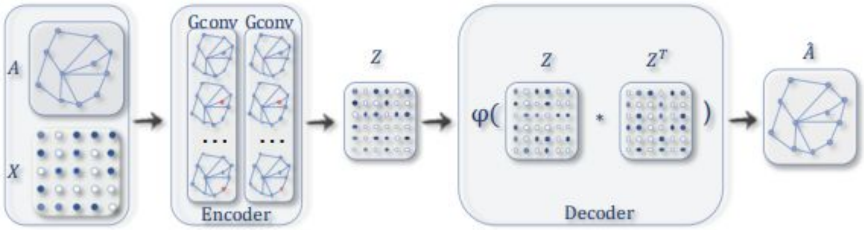
⇒ Similarity of node's attributes is a link predictors in addition to structure proximity

- Social Influence
  - Characteristics of individuals may be affected by the characteristics of their relations

⇒ Your neighbours' attributes can reveal yours

Link Prediction in Attributed Graphs

Graph Neural Networks get attributed graphs as the input and can be used for many tasks including link prediction



Approach	Category	Inputs	Pooling	Readout	Time Complexity
GNN* (2009) [15]	RecGNN	A, X, X^e	-	a dummy super node	-
GraphESN (2010) [16]	RecGNN	A, X	-	mean	-
GGNN (2015) [17]	RecGNN	A, X	-	attention sum	-
SSE (2018) [18]	RecGNN	A, X	-	-	-
Spectral CNN (2014) [19]	Spectral-based ConvGNN	A, X	spectral clustering+max pooling	max	$O(n^3)$
Henaff et al. (2015) [20]	Spectral-based ConvGNN	A, X	spectral clustering+max pooling		$O(n^3)$
ChebNet (2016) [21]	Spectral-based ConvGNN	A, X	efficient pooling	sum	$O(m)$
GCN (2017) [22]	Spectral-based ConvGNN	A, X	-	-	$O(m)$
CayleyNet (2017) [23]	Spectral-based ConvGNN	A, X	mean/graclus pooling	-	$O(m)$
AGCN (2018) [40]	Spectral-based ConvGNN	A, X	max pooling	sum	$O(n^2)$
DualGCN (2018) [41]	Spectral-based ConvGNN	A, X	-	-	$O(m)$
NN4G (2009) [24]	Spatial-based ConvGNN	A, X	-	sum/mean	$O(m)$
DCNN (2016) [25]	Spatial-based ConvGNN	A, X	-	mean	$O(n^2)$
PATCHY-SAN (2016) [26]	Spatial-based ConvGNN	A, X, X^e	-	concat	-
MPNN (2017) [27]	Spatial-based ConvGNN	A, X, X^e	-	attention sum/ set2set	$O(m)$
GraphSage (2017) [42]	Spatial-based ConvGNN	A, X	-	-	-
GAT (2017) [43]	Spatial-based ConvGNN	A, X	-	-	$O(m)$
MoNet (2017) [44]	Spatial-based ConvGNN	A, X	-	-	$O(m)$
PGC-DGCNN (2018) [46]	Spatial-based ConvGNN	A, X	sort pooling	attention sum	$O(n^3)$
CGMM (2018) [47]	Spatial-based ConvGNN	A, X	-	concat	-
LGCN (2018) [45]	Spatial-based ConvGNN	A, X	-	-	-
GAAN (2018) [48]	Spatial-based ConvGNN	A, X	-	-	$O(m)$
FastGCN (2018) [49]	Spatial-based ConvGNN	A, X	-	-	-
StoGCN (2018) [50]	Spatial-based ConvGNN	A, X	-	-	-
Huang et al. (2018) [51]	Spatial-based ConvGNN	A, X	-	-	-
DGCNN (2018) [52]	Spatial-based ConvGNN	A, X	sort pooling	-	$O(m)$
DiffPool (2018) [54]	Spatial-based ConvGNN	A, X	differential pooling	mean	$O(n^2)$
GeniePath (2019) [55]	Spatial-based ConvGNN	A, X	-	-	$O(m)$
DGI (2019) [56]	Spatial-based ConvGNN	A, X	-	-	$O(m)$
GIN (2019) [57]	Spatial-based ConvGNN	A, X	-	concat+sum	$O(m)$
ClusterGCN (2019) [58]	Spatial-based ConvGNN	A, X	-	-	-

numerous methods, multiple surveys, e.g. this one (2019),
two of the notable works: GCN (2016), GAT (2018),
an excellent course last term on this & a new book!

More on this later

Common prediction tasks

- Link Prediction
- **Node Classification**
- Graph Classification

Examples:

https://paperswithcode.com/task/link-prediction

https://paperswithcode.com/task/node-classification

https://paperswithcode.com/task/graph-classification

What is unsupervised node classification?

Attributed Graphs

Interplay between attributes and relations, a positive feedback loop derived by two social theories:

- Social Selection
  - Similarity of individuals' characteristics motivates them to form relations

  ⇒ Similarity of node's attributes is a link predictors in addition to structure proximity

- Social Influence
  - Characteristics of individuals may be affected by the characteristics of their relations

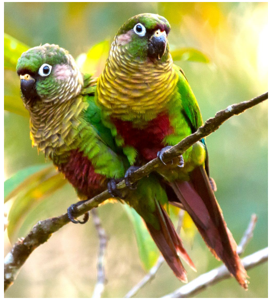
  ⇒ Your neighbours' attributes can reveal yours

Assortativity & Homophily

Similar nodes tend to link to each other

How to measure age homophily in a given friendship graph when you know age of every node?

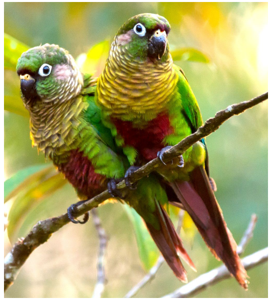
birds of the same feather flock together



Assortativity & Homophily

Similar nodes tend to link to each other

birds of the same feather flock together



How to measure age homophily in a given friendship graph when you know age of every node? Similar to degree assortativity, measure the correlation of age across all edges

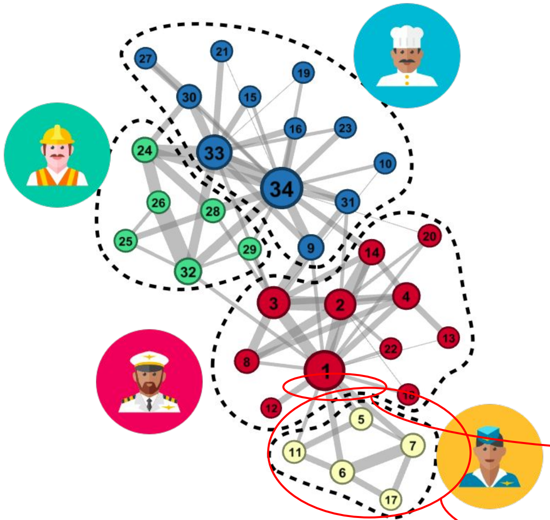
How to measure occupation homophily?

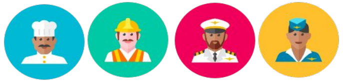
categorical attribute instead of a numeric one

Assortativity & Homophily

Look at the mixing patterns

Mixing matrix shows the number of edges connecting each pair of attribute values





	B	G	R	Y
B	22	7	2	0
G	7	6	7	0
R	2	7	19	4
Y	0	0	4	6

What indicates homophily in this matrix?
How does homophily look like?

Assortativity & Homophily

Look at the mixing patterns

Mixing matrix shows the number of edges connecting each pair of attribute values

e$_{ij}$: ratio of edges between each pair of values

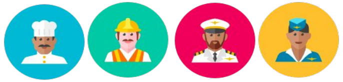
Assortativity index

$$r = \frac{\sum_i e_{ii} - \sum_i e_{i.}e_{.i}}{1 - \sum_i e_{i.}e_{.i}} = \frac{Tr[e] - ||e^2||}{1 - ||e^2||}$$

What indicates homophily in this matrix?
How does homophily look like?
dominant diagonal

Is normalized Q-modularity assuming attributes partition the graph

$$Q = \sum_i e_{ii} - e_{i.}^2 = \mathrm{Tr}[e] - ||e^2||$$



	B	G	R	Y
B	22	7	2	0
G	7	6	7	0
R	2	7	19	4
Y	0	0	4	6

/sum (E)

Assortativity & Homophily

Look at the mixing patterns

Mixing matrix shows the number of edges connecting each pair of attribute values

e_{ij}: ratio of edges between each pair of values

	B	G	R	Y
B	**22**	7	2	0
G	7	**6**	7	0
R	2	7	**19**	4
Y	0	0	4	**6**

/sum (E)

Assortativity index

$$r = \frac{\sum_i e_{ii} - \sum_i e_{i.} e_{.i}}{1 - \sum_i e_{i.} e_{.i}} = \frac{Tr[e] - ||e^2||}{1 - ||e^2||}$$

What indicates homophily in this matrix?
How does homophily look like?
dominant diagonal

Is there other mixing patterns?

Assortativity & Homophily

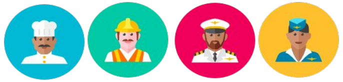
Look at the mixing patterns

Mixing matrix shows the number of edges connecting each pair of attribute values

$\mathbf{e_{ij}}$: ratio of edges between each pair of values

Assortativity index

$$r = \frac{\sum_i e_{ii} - \sum_i e_{i.}e_{.i}}{1 - \sum_i e_{i.}e_{.i}} = \frac{Tr[e] - ||e^2||}{1 - ||e^2||}$$



	B	G	R	Y
B	22	7	2	0
G	7	6	7	0
R	2	7	4	19
Y	0	0	19	6

/sum (E)

What indicates homophily in this matrix?
How does homophily look like?
dominant diagonal

Is there other mixing patterns?

e.g. opposites attract

Structural Correlation

Look at the mixing patterns

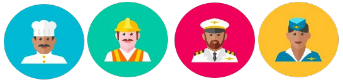
Mixing matrix shows the number of edges connecting each pair of attribute values

Other mixing patterns, such as **heterophily** can also be reflected in the mixing matrix

e_{ij}: ratio of edges between each pair of values

Assortativity index

$$r = \frac{\sum_i e_{ii} - \sum_i e_{i.}e_{.i}}{1 - \sum_i e_{i.}e_{.i}} = \frac{Tr[e] - ||e^2||}{1 - ||e^2||}$$



	B	G	R	Y
B	22	7	2	0
G	7	6	7	0
R	2	7	4	19
Y	0	0	19	6

/sum (E)

How does heterophily look like in the mixing matrix?
How does correlation look like in the mixing matrix?

Structural Correlation

Look at the mixing patterns

Mixing matrix shows the number of edges connecting each pair of attribute values

Other mixing patterns, such as heterophily can also be reflected in the mixing matrix

$\mathbf{e_{ij}}$: ratio of edges between each pair of values

Assortativity index

$$r = \frac{\sum_i e_{ii} - \sum_i e_{i.}e_{.i}}{1 - \sum_i e_{i.}e_{.i}} = \frac{Tr[e] - ||e^2||}{1 - ||e^2||}$$

	B	G	R	Y
B	**22**	7	2	0
G	7	**6**	7	0
R	2	7	4	**19**
Y	0	0	**19**	6

/sum (E)

How does heterophily look like in the mixing matrix?
How does **correlation** look like in the mixing matrix?
A **dominant cell** in each row and column

Structural Correlation

Look at the mixing patterns

Mixing matrix shows the number of edges connecting each pair of attribute values

Other mixing patterns, such as heterophily can also be reflected in the mixing matrix

e_{ij}: ratio of edges between each pair of values

Assortativity index

$$r = \frac{\sum_i e_{ii} - \sum_i e_{i.} e_{.i}}{1 - \sum_i e_{i.} e_{.i}} = \frac{Tr[e] - ||e^2||}{1 - ||e^2||}$$

	B	G	R	Y
B	22	7	2	0
G	7	6	7	0
R	2	7	4	19
Y	0	0	19	6

/sum (E)

How to quantify the overall correlation?

Does it resemble anything?

Structural Correlation

Look at the mixing patterns

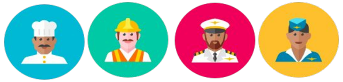
Mixing matrix shows the number of edges connecting each pair of attribute values

Other mixing patterns, such as heterophily can also be reflected in the mixing matrix

e_{ij}: ratio of edges between each pair of values

Assortativity index

$$r = \frac{\sum_i e_{ii} - \sum_i e_{i.} e_{.i}}{1 - \sum_i e_{i.} e_{.i}} = \frac{Tr[e] - ||e^2||}{1 - ||e^2||}$$



	B	G	R	Y
B	22	7	2	0
G	7	6	7	0
R	2	7	4	19
Y	0	0	19	6

/sum (E)

How to quantify the overall correlation?

Does it resemble anything? confusion matrix where pairwise cluster overlaps are changed to edges between pair of values

Structural Correlation

Look at the mixing patterns

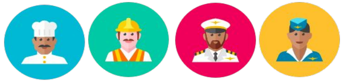
Mixing matrix shows the number of edges connecting each pair of attribute values

Other mixing patterns, such as heterophily can also be reflected in the mixing matrix

$\mathbf{e_{ij}}$: ratio of edges between each pair of values

Assortativity index

$$r = \frac{\sum_i e_{ii} - \sum_i e_{i.}e_{.i}}{1 - \sum_i e_{i.}e_{.i}} = \frac{Tr[e] - ||e^2||}{1 - ||e^2||}$$



.2	.07	.02	0
.07	.06	.07	0
.02	.07	.04	.17
0	0	.17	.06

How to quantify the overall correlation?
measure the total dispersion similar to clustering agreement indexes

Beyond Assortativity: Proclivity Index for Attributed Networks, PAKDD (2017)

Structural Correlation of Attributes

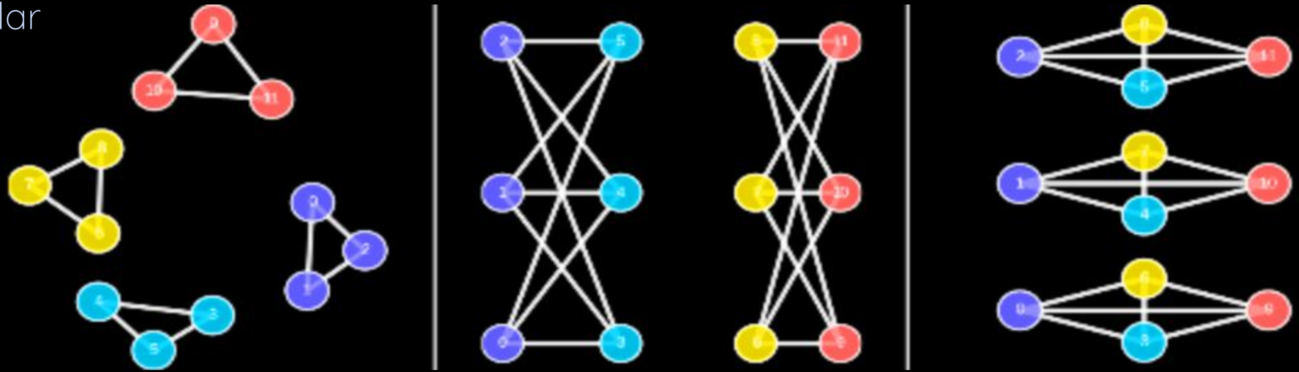
Proclivity

“inclination or predisposition toward a particular thing”

similar
homophily

different
heterophily

random



Assortativity	**1.0**	**-0.33**	**-0.33**
Prone	**1.0**	**1.0**	**0.11**

[Beyond Assortativity: Proclivity Index for Attributed Networks,](#) PAKDD (2017)

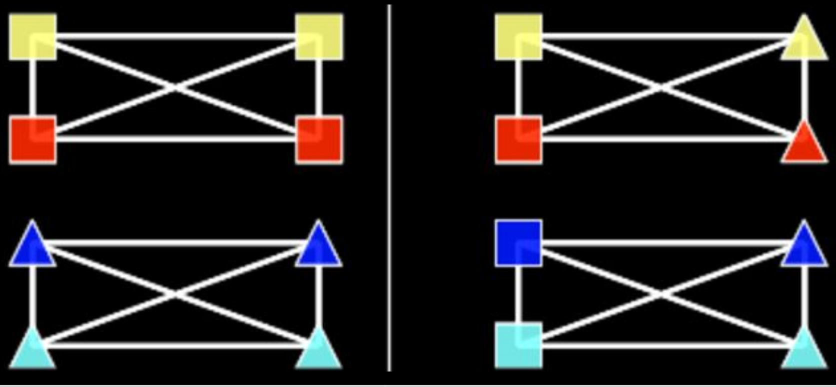
Structural Correlation of Attributes

Proclivity

Cross-proclivity
in addition to
self-proclivity

Shape and Color

correlation | no correlation

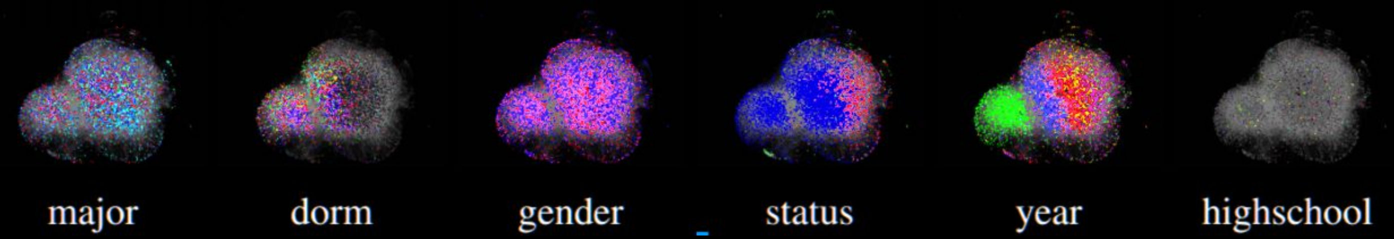


Assortativity	x	x
Prone	**0.5**	**0.0**

Correlation between your income and occupations of your friends

Structural Correlation of Attributes

Facebook friendship network of a US college
Color: distinct attribute values
Placement : connections



correlation between the entrance year a student based on the high school his friends are from, the dormitory they are in and most of all their entrance year

	major	gender	year	status	dorm	highschool
major	0.01	0.00	0.01	0.00	0.01	**0.05**
gender	0.00	0.00	0.00	0.00	0.00	0.00
year	0.01	0.00	**0.25**	**0.07**	**0.07**	**0.07**
status	0.00	0.00	**0.07**	**0.09**	0.02	0.02
dorm	0.01	0.00	**0.07**	0.02	**0.16**	**0.10**
highschool	**0.05**	0.00	**0.07**	0.02	**0.10**	**0.31**

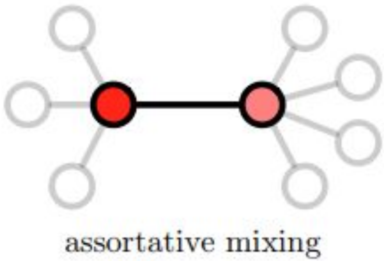
Even if you don't put your information online, that information can be inferred/predicted based on what your friends reveal about themselves

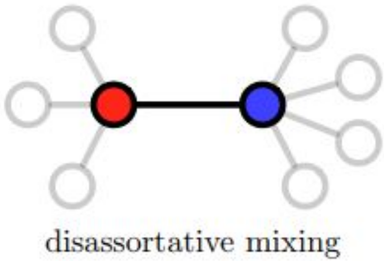
Predicting missing node attributes

Most graphs are incomplete, and often attributes of some nodes are missing

We can use structural correlations to predict a missing attribute to be e.g. the average (scalar) or most common (categorical) value of its neighbors' non-missing attributes

What does this local smoothing assume about the mixing patterns?





[From Clauset's Slides]

Predicting missing node attributes

Most graphs are incomplete, and often attributes of some nodes are missing

We can use structural correlations to predict a missing attribute to be e.g. the average (scalar) or most common (categorical) value of its neighbors' non-missing attributes

What does this local smoothing assume about the mixing patterns?

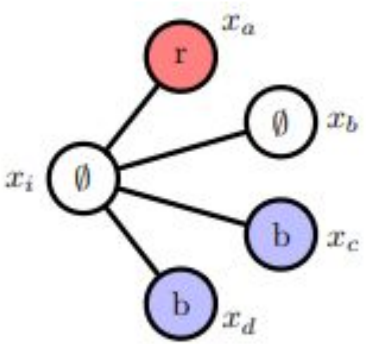
mean (scalar) & mode (categorical) ⇒ assortative

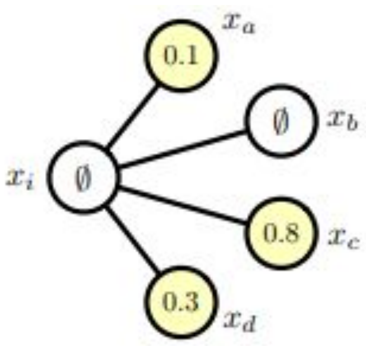
From Clauset's Slides

Predicting missing node attributes, example

missing = mean (scalar) & mode (categorical)

what is the prediction in these two cases for node i?



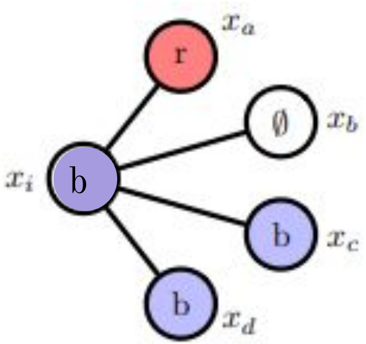


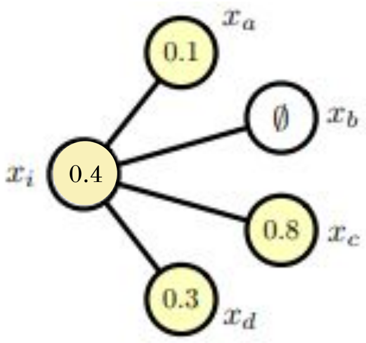
[From Clauset's Slides]

Predicting missing node attributes, example

missing = mean (scalar) & mode (categorical)

what is the prediction in these two cases for node i?



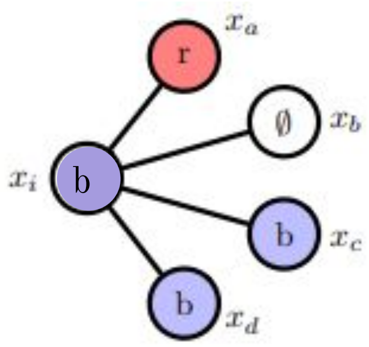


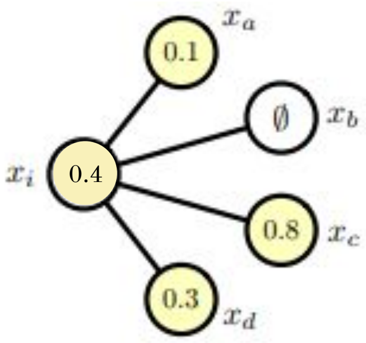
what is the predictions for node x_b ?

Predicting missing node attributes, example

missing = mean (scalar) & mode (categorical)

what is the prediction in these two cases for node i?



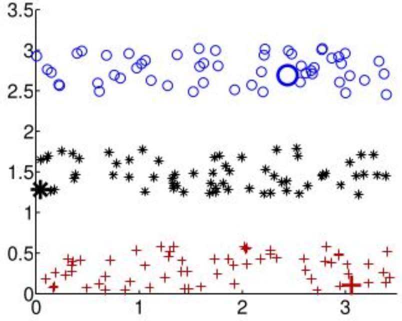


what is the predictions for node x_b ? repeat given current predictions

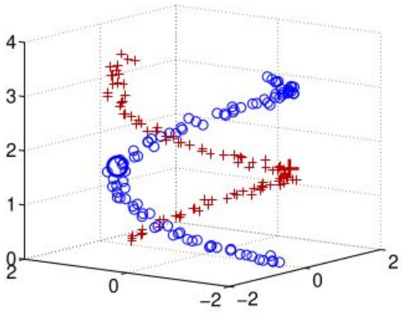
Label Propagation Algorithm

Was proposed for semi-supervised classification of iid data by defining a fully connected distance graph

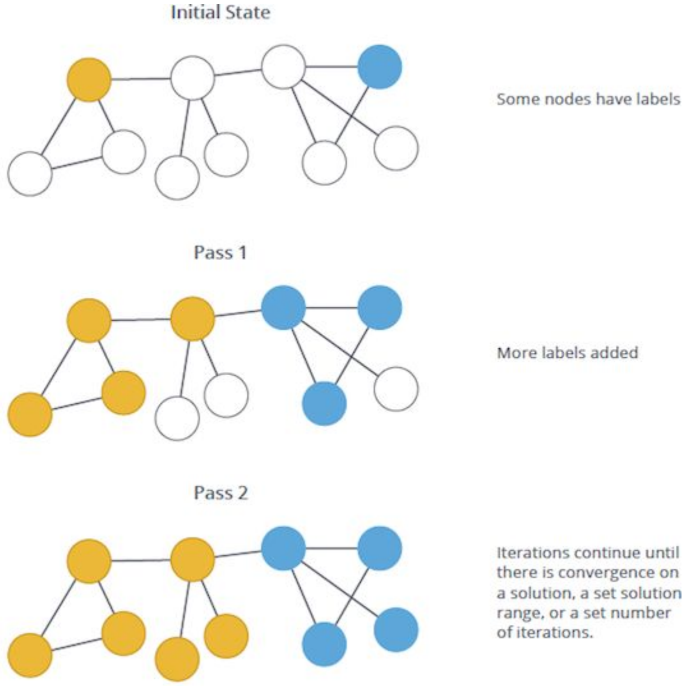
Zhu X, Ghahramani Z. Learning from Labeled and Unlabeled Data with Label Propagation.



(a) 3-Bands



(b) Springs

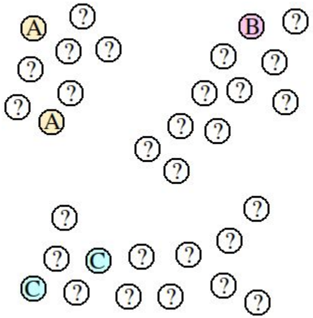


Label Propagation Algorithm

Label Smoothing

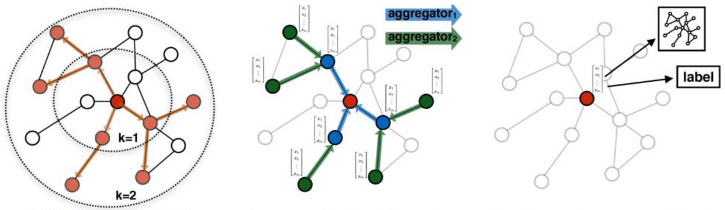
Node classification

- Unsupervised learning
  - clustering, only graph is given, classes/clusters are not predefined
- Supervised learning
  - classifying, input is graph and labels on all nodes
    - You mask some nodes (labels and their connections) for training [inductive]
    - You mask some nodes (only labels) for training [transductive]
- Semi-supervised learning
  - input is graph and labels on some nodes
  - You mask some node labels for training (seeing the whole graph: transductive)
- Active learning
  - Input is graph and a budget that determines how many nodes you can query for labels
  - labels come in sequence and can be queried based on the current set



Semi-Supervised Node classification

- Traditional
  - label propagation & belief propagation
- Recent end-to-end methods (Feature Smoothing)
  - GCN and variants, which use a classification loss
- Embedding based
  - Unsupervised embedding extraction (e.g. node2vec) then apply a classifier



1. Sample neighborhood
2. Aggregate feature information from neighbors
3. Predict graph context and label using aggregated information

More on this later

Unifying Graph Convolutional Neural Networks and Label Propagation, 2020

Measuring performance of attribute prediction

Scalar values ⇒ correlation of predicted & actual values (r^2 correlation)

Categorical values ⇒ confusion matrix & average accuray

C_{ij} = the number of nodes with predicted label i and actual label j

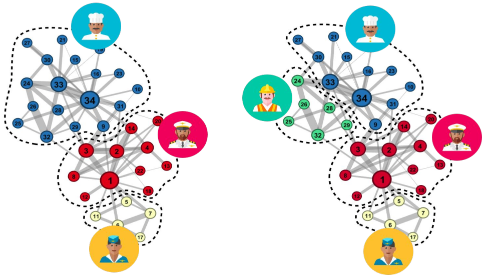
Accuracy = 1/34 Tr(C)

	G	B	R	Y	Σ
G	0	0	0	0	**0**
B	**12**	**6**	0	0	18
R	0	0	**11**	0	**11**
Y	0	0	0	**5**	5
Σ	12	6	11	5	34

results

truth		
	90	0
	10	0

What is the accuracy?



Measuring performance of attribute prediction

Scalar values ⇒ correlation of predicted & actual values ([r^2 correlation]())

Categorical values ⇒ confusion matrix & average accuray

C_{ij} = the number of nodes with predicted label i and actual label j

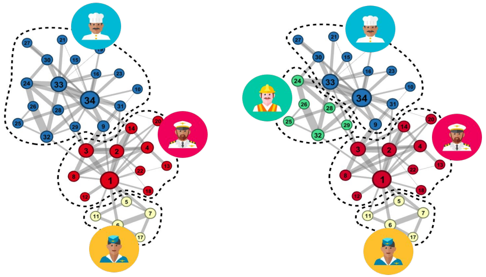
	G	B	R	Y	Σ
G	0	0	0	0	**0**
B	**12**	**6**	0	0	18
R	0	0	**11**	0	11
Y	0	0	0	**5**	5
Σ	12	6	11	5	34

Accuracy = Tr(C)/Sum(C)

	results	
truth	90	0
	10	0

What is the accuracy? 90% but is always guessing the majority class and never getting the minority class correct

Class imbalance problem

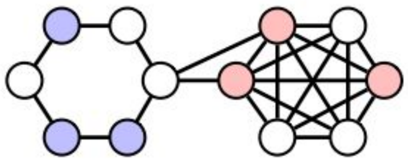


Measuring performance of attribute prediction

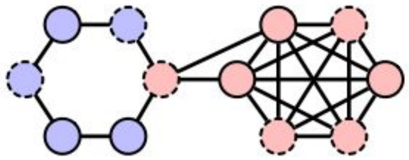
Example

Truth: left circle all blue, right circle all red

Observed: 6 missing values



observed



predicted

What is the accuracy?

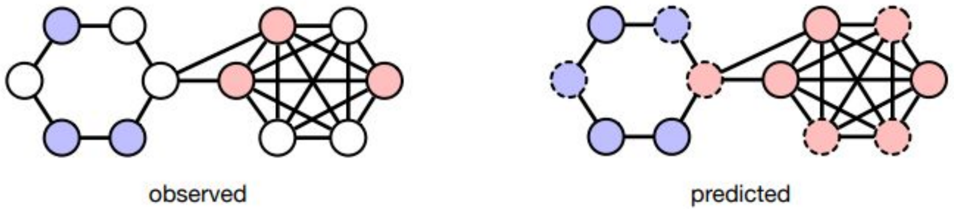
	$\mathcal{C}$	actual	
		r	b
pred.	r	3	1
	b	0	2

Measuring performance of attribute prediction

Example

Truth: left circle all blue, right circle all red

Observed: 6 missing values



	$\mathcal{C}$	actual r	actual b
pred.	r	3	1
pred.	b	0	2

What is the accuracy?

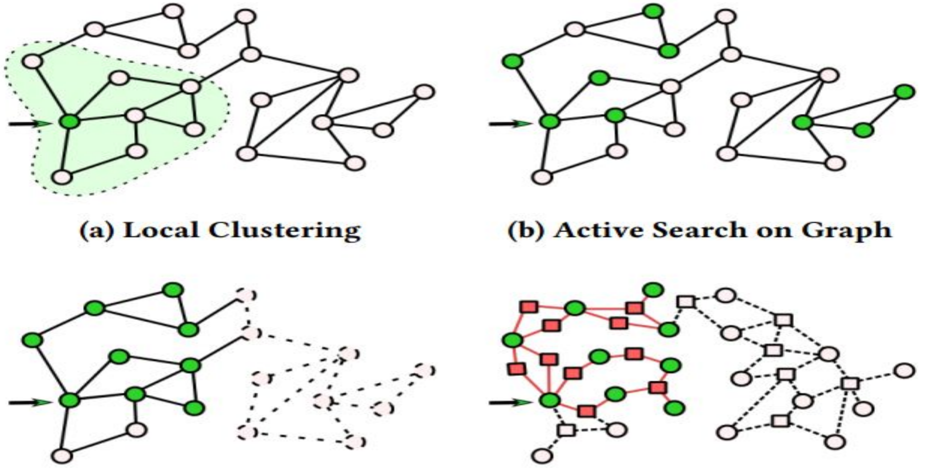
Accuracy = ⅚ = 0.83

Active Search of Connections

What if you can query with some cost or given a budget?

Semi-supervised ⇒ Active learning

labels are local and depend on the given seed



(a) Local Clustering

(b) Active Search on Graph

(c) Active Exploration

(d) Active Search of Connections

[Active Search of Connections for Case Building and Combating Human Trafficking](), 2018