Information-theoretic evaluation of stochastic graph models using a large collection of real-world graphs.

Kevin Lang: Yahoo Research

Barcelona, Feb 12, 2009

High-Level Outline

- Part 1: Use likelihood method to study the fit of several well-known stochastic graph models to a collection of real-world graphs.
- Part 2: Use the graphs to study how well some clustering metrics deal with the k=1 vs k>1 question,
  both for the graphs themselves,
  and for random graphs with the same degree sequences.

In part 1

- We evaluate stochastic models of graphs and degree sequences using likelihood methodology as advocated by Bezáková, Kalai, and Santhanam, in ICML-06.
- They were responding to the profusion of models of complex networks, that were often being evaluated in qualitative ways.
- The likelihood methodology is applicable to probabilistic generative models like G_{np}, G_{nm}, G_D, preferential attachment, etc.
- For a given object, the model with the highest probability of generating that object is considered best.
- Scores can be reported as negative log probabilities, measured in bits.
- [It is not a coincidence that scores for MDL and compression-based schemes are also reported in bits.]

In part 1 (cont'd)

- We compare two models of graphs in this section, namely G_{nm} and G_D.
- G_{nm} places a uniform distribution over the set of all graphs with n nodes and m edges.
- G_D places a uniform distribution over the set of all graphs with n nodes, and degrees exactly specified by the vector D.
- We also compare three models of degree sequences, including the commonly assumed power-law model.
- The experimental testbed is a diverse set of more than 100 graphs collected by Jure Leskovec.

Graph Name	N nodes	M edges	model degs	G
monksLk3	18	41	APH	m
karate	34	78	APH	m
dolphins	62	159	PAH	m
football	115	613	HPA	m
netscience	379	914	AHP	m
mani-facesDeg3	551	1981	HAP	m
mani-swissK10	20000	199955	AHP	m
powergrid-watts	4941	6594	HAP	m
road-USA	126146	161950	HAP	m
road-pa	1087562	1541514	HAP	m
random-deg4	100000	200000	AHP	m
gnutella-30	36646	88303	HPA	D
delicious	147567	301921	HPA	D
epinions	75877	405739	PHA	D
flickr	404733	2110078	HPA	D
LinkedIn	6946668	30507070	HPA	D
LiveJournal01	3766521	30629297	HPA	D
answers	488484	1240189	HPA	D
answers-2	25431	65551	HPA	D
as-newman	22963	48436	HPA	D
as-oregon-all	13579	37448	HPA	D

Graph Name	N nodes	M edges	model degs	G
Au-Pa-cond-mat	57552	104179	HAP	D
Au-Pa-gr-qc	14832	22266	HAP	D
Au-Pa-hep-th	39986	64154	HPA	D
amazon_2003_all	473315	3505519	HAP	D
bio-proteinsVespignani	4626	14801	HAP	D
Blog-nat05-6m	29150	182212	HPA	D
Cit-hep-ph	34401	420784	HAP	D
clickstream-UsrToUrl	199308	951649	HPA	D
CoAuth-cond-mat	21363	91286	HAP	D
CoAuth-hep-ph	11204	117619	HPA	D
dblp-larsWcc	317080	1049866	HAP	D
email-all-inOut	37803	114199	HPA	D
email-ijs	72216	91393	HPA	D
imdbDec07-ActToAct	821810	27394903	HAP	D
imdbDec07-ActToMov	1966620	5771671	HPA	D
imdb-a-m-30countries	198430	566756	HPA	D
Patents	3764105	16511682	HAP	D
Post-nat05-6m	238305	297338	HPA	D
protein_dip	4626	14801	HAP	D
web-google	855802	4291352	HAP	D
web-wt10g-trec	1458316	6225033	HPA	D

Three encoding schemes for vectors of non-negative integers (e.g. degree sequences)

- P: encode using Power-law distribution. (This is a common working assumption in the complex networks field).
- A: an “Agnostic” null hypothesis that doesn’t explicitly assume a distribution, but rather a particular total for the integers.
- H: recursive Histogram-based scheme that encodes using the empirical symbol counts, that we must also pay to encode (so this is not cheating).

Comparative likelihood of P, A, H models of degree sequences

- P (power-law scheme) is *not* usually the best of the three.
- H is best for nearly all large graphs, which is not surprising.
- More surprising is that A sometimes beats P; apparently for some graphs the power-law assumption is very bad.

Remarks: 1) other researchers have already studied the power-law hypothesis more rigorously than this. 2) Experts do not believe that the power-law holds across the entire range of degrees, and that is necessary for good encoding by the P scheme.

Compression using 2 standard graph models

Define $\mathrm{tri}(n) = n(n-1)/2$. Let $\mathbf{sb}()$ and $\mathbf{vb}()$ be "good" encoding costs for scalars and vectors. **Prior-alert!**

- Scheme for G_{nm}: Assume a uniform distribution over all possible N-node M-edge graphs. Then pay $\mathbf{sb}(N) + \mathbf{sb}(M) + \log \binom{\mathrm{tri}(N)}{M}$.
- Scheme for G_D (based on BKS-06 scheme for encoding digraphs).
  1. Pay $\mathbf{sb}(N) + \mathbf{sb}(M)$.
  2. Orient the edges, then pay $\mathbf{vb}(\mathrm{id}) + \mathbf{vb}(\mathrm{od})$ to encode the resulting indeg and outdeg sequences.
  3. Pay $\left(\log(M!) - \sum_i^N \log(\mathrm{id}_i!) - \sum_i^N \log(\mathrm{od}_i!)\right)$ bits to choose from a uniform distribution over digraphs having those indegs and outdegs.

Graph Name	N nodes	M edges	model degs	G
monksLk3	18	41	APH	m
karate	34	78	APH	m
dolphins	62	159	PAH	m
football	115	613	HPA	m
netscience	379	914	AHP	m
mani-facesDeg3	551	1981	HAP	m
mani-swissK10	20000	199955	AHP	m
powergrid-watts	4941	6594	HAP	m
road-USA	126146	161950	HAP	m
road-pa	1087562	1541514	HAP	m
random-deg4	100000	200000	AHP	m
gnutella-30	36646	88303	HPA	D
delicious	147567	301921	HPA	D
epinions	75877	405739	PHA	D
flickr	404733	2110078	HPA	D
LinkedIn	6946668	30507070	HPA	D
LiveJournal01	3766521	30629297	HPA	D
answers	488484	1240189	HPA	D
answers-2	25431	65551	HPA	D
as-newman	22963	48436	HPA	D
as-oregon-all	13579	37448	HPA	D

Graph Name	N nodes	M edges	model degs	G
Au-Pa-cond-mat	57552	104179	HAP	D
Au-Pa-gr-qc	14832	22266	HAP	D
Au-Pa-hep-th	39986	64154	HPA	D
amazon_2003_all	473315	3505519	HAP	D
bio-proteinsVespignani	4626	14801	HAP	D
Blog-nat05-6m	29150	182212	HPA	D
Cit-hep-ph	34401	420784	HAP	D
clickstream-UsrToUrl	199308	951649	HPA	D
CoAuth-cond-mat	21363	91286	HAP	D
CoAuth-hep-ph	11204	117619	HPA	D
dblp-larsWcc	317080	1049866	HAP	D
email-all-inOut	37803	114199	HPA	D
email-ijs	72216	91393	HPA	D
imdbDec07-ActToAct	821810	27394903	HAP	D
imdbDec07-ActToMov	1966620	5771671	HPA	D
imdb-a-m-30countries	198430	566756	HPA	D
Patents	3764105	16511682	HAP	D
Post-nat05-6m	238305	297338	HPA	D
protein_dip	4626	14801	HAP	D
web-google	855802	4291352	HAP	D
web-wt10g-trec	1458316	6225033	HPA	D

Comparative likelihood of G_{nm} and G_D models

- G_D was better for most of the large real-world graphs. [Note that we are not necessarily modeling the degree distribution as a power law.]
- G_m was better for some meshes and manifolds, some constant-degree expanders, and the standard **tiny** graphs used in numerous "complex networks" papers.

Outline of Part 2, in which we point out flaws in several k-choosing schemes for clustering

- General reasons for skepticism about k-choosing schemes.
- The k=1 vs k>1 question for real graphs and for random graphs.
- Our experimental procedure, and results for one graph (IMDB).
- Diagnose specific problems of certain published schemes, focusing on k=1 vs k>1 question for random graphs.

General reasons for skepticism about k-choosing schemes

- Real datasets are often very ambiguous, making it easy to change the answer by tweaking the objective function or by switching to a different one.
- The mechanisms for tweaking objective functions derived in Bayesian or MDL frameworks are especially obvious.
- The optimization problem is usually intractable, so $k = 20$ might seem better than $k = 25$ simply because the algorithm got especially lucky for $k = 25$ or unlucky for $k = 25$.

The k=1 vs k>1 question for real graphs and for random graphs.

- One good thing about this more limited question is that if we happen to find a sufficiently good clustering, then we can conclude $k > 1$, with no remaining uncertainty due to intractability.

- Also, for random graphs most people agree that the answer should be $k = 1$. This provides a sanity check that clustering objective functions *can fail* (and the "wrong" answer $k > 1$ happens to be the one that we can believe.)

Preview of Contents of Part 2

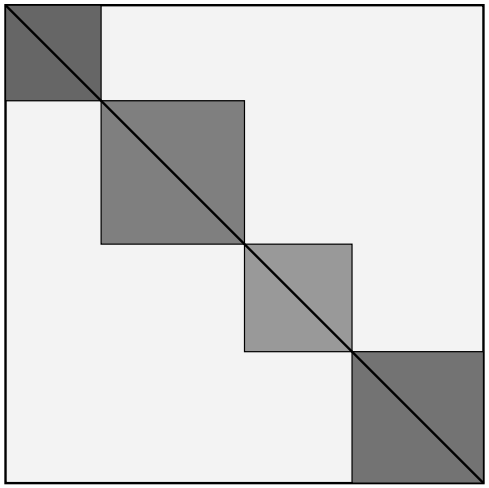
We will discuss some specific flaws in several published schemes, focusing on k=1 question for random graphs.

- Modularity-Q: fails to account for variance in cut values.
- "Buggy" partitioned G_{nm}: serious coding inefficiency in message header dangerously reduces the marginal cost of increasing k.
- Partitioned G_{nm}: overlooks existence of non-clusterlike partitionings of nodes that can also reduce entropy, including partitioning by degree.
- We have tried a partitioned G_D scheme that seems to work better on the k=1 question for real graphs and random graphs with the same degree sequences.
- However, we do not claim that this is the final word on the subject.

Experimental Procedure

```
for each of 100 graphs do:
    create perm-graph by permuting node numbering of original graph.
    create rand-graph, a random graph with the same degree sequence
      for each k-parts in [1,2,4,8...1024] do
        use Graclus to partition perm-graph into k parts
        use Graclus to partition rand-graph into k parts
        for each clustering metric in list do
           evaluate metric (perm-graph, k_parts_of_perm-graph)
           evaluate metric (rand-graph, k_parts_of_rand-graph)
        end of loop over metrics
      end of loop over k-parts
end of loop over 100 graphs.
```

Intuition behind Modularity Q



- Intuitively, good clusterings allow the adjacency matrix to be reordered and divided into blocks so that there is an obvious concentration of edges in the on-diagonal blocks.
- Modularity Q [Newman et al] is a formalization of this idea.
- It measures the *excess* number of edges in on-diagonal blocks, as compared to the **expected** number for a random G_W graph partitioned the same way.

Using Modularity Q

- Modularity Q can be applied after-the-fact to compare the members of any set of clusterings.
- The clustering with the biggest Q score would be considered best.
- That clustering's k value would then also be considered best.

Results on IMDB Actors vs Movies Graph

Source of Partitioning	modularity Q score		MDL w/ bug bits / edge		MDL G_{nm} bits / edge		MDL G_D bits / edge	
	orig	rand	orig	rand	orig	rand	orig	rand
None (Base Model)			18.51	18.51	18.51	18.51	17.11	17.53
None (K=1)	0.000	0.000	26.59	26.59	18.51	**18.51**	17.11	**17.53**
Graclus(K=4)	0.711	0.408	24.88	26.04	17.70	18.87	16.32	18.12
Graclus(K=16)	**0.848**	**0.462**	23.33	25.37	16.94	19.08	15.65	18.22
Graclus(K=64)	0.824	0.412	22.13	24.88	16.49	19.25	**15.50**	18.56
Graclus(K=256)	0.735	0.392	**21.77**	**24.60**	**16.45**	19.27	15.89	18.77
Graclus(K=1024)	0.654	0.373	27.04	29.99	16.66	19.54	16.41	19.01
Segregate Degrees		0.000				17.88		19.00

10-try results	Modu Q		MDL G_{nm}		MDL G_D	
graph	G	R	G	R	G	R
monksLk3	2	4	2	1	1	1
karate	4	4	1	1	1	1
dolphins	4	4	2	1	2	1
football	8	8	16	1	8	1
netscience	16	8	32	1	16	1
mani-facesDeg3	16	8	64	1	16	1
mani-swissK10	32	8	1024	1	128	1
powergrid-watts	32	32	32	1	32	1
road-USA	128	512	512	1	512	1
road-pa	256	32	1024	1	1024	1
random-deg4	16	16	1	1	1	1
gnutella-30	32	16	1	1	1	1
delicious	64	16	1024	1	1024	1
epinions	8	8	1024	256	16	1
flickr	16	8	1024	16	16	1
LinkedIn	8	8	1024	32	1024	1
LiveJournal01	256	8	1024	256	1024	1
answers	16	8	128	1	16	1
answers-2	16	8	8	1	1	1
as-newman	16	256	32	1	8	1
as-oregon-all	8	8	128	1	16	1

graph name	Modu Q		MDL G_{nm}		MDL G_D	
	G	R	G	R	G	R
AuthToPap-cond-mat	128	16	1024	1	512	1
AuthToPap-gr-qc	128	16	128	1	128	1
AuthToPap-hep-th	128	16	512	1	256	1
amazon_2003_all	16	8	1024	1	1024	1
bio-proteinsVespignani	16	16	32	1	1	1
Blog-nat05-6m	4	8	128	32	4	1
Cit-hep-ph	16	8	1024	256	64	1
clickstream-UsrToUrl	32	16	512	64	1	1
CoAuth-cond-mat	16	8	1024	1	512	1
CoAuth-hep-ph	64	8	1024	128	256	1
dblp-larsWcc	64	8	1024	1	1024	1
email-all-inOut	32	128	128	2	1	1
email-ijs	16	16	512	1	16	1
imdbDec07-ActToAct	16	8	1024	1024	1024	1
imdbDec07-ActToMov	16	8	1024	1	128	1
imdb-a-m-30countries	32	8	128	1	32	1
Patents	16	8	1024	1	1024	1
Post-nat05-6m	64	64	512	1	256	1
protein_dip	16	8	32	1	1	1
web-google	128	8	1024	128	1024	1
web-wt10g-trec	256	8	1024	32	1024	1

Modularity Q scores for IMDB graph

- Q says that the $k = 16$ clustering is the best of our set of graclus clusterings of the actual IMDB graph.
- Okay, so far, so good.
- Q also says that the $k = 16$ clustering is the best of our (different) set of graclus clusterings of the random G_D graph (which has the IMDB graph's degree sequence).
- Modularity Q has just failed an important sanity check. What went wrong?

This is a known flaw in Modularity Q

- This was diagnosed in the 2004 paper “Modularity from fluctuations in random graphs and complex networks” by R. Guimera, M. Sales-Pardo, and L. A. N. Amaral.
- The problem is that the number of excess diagonal-block edges is measured relative to the **expected** number for G_W.
- The **variance** of that number is completely ignored.
- Even in the simple case of balanced bisections of a constant-degree random expander, there will be a range of different cut values.
- The minimum and maximum cut values will give a positive and negative Q score respectively.

Avoiding that pitfall

- Guimera et al pointed out that one *should* try to determine whether the excess in diagonal-block edges (relative to the expected number) is statistically significant given the variance.
- They worked out a initial statistical test, but it did not apply to networks with highly skewed degree distributions.
- A rigorous statistical test that *could* handle the skew-degree case would provide a useful alternative to the MDL schemes that we will describe next.

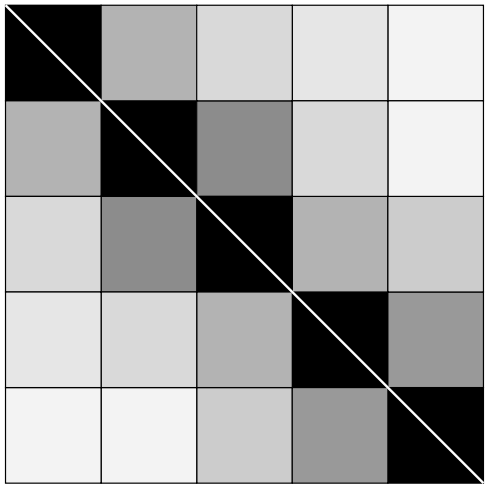
Compression-Based Approach (MDL)

- This approach tends to avoid the particular flaw that we have pointed out in modularity Q.
- It involves a tradeoff between bits in a message's header (which describes some kind of special graph structure) and bits in the message's body (which must eliminate all remaining uncertainty).
- The important thing is that the **entropy reduction** induced by some special structure **is not free**, but must be paid for by a description of that structure.

Additional Remarks on MDL approach

- The earlier comparison between the A and H models of a degree sequence had an MDL flavor. Does sending a symbol count histogram in the msg header cause enough entropy reduction?
- The earlier comparison between the G_{nm} and G_D sequence had an MDL flavor: Does sending a graph's degree sequence in the msg header cause enough entropy reduction?
- For graph clustering, the question will be: Does sending a node partitioning in the msg header cause enough entropy reduction?
- The 2007 PNAS paper by Rosvall and Bergstrom is a good tutorial on the MDL idea as applied to graph clustering.

Compressing with partitioned G_{nm}



- In essence, we apply the earlier G_{nm} encoding scheme separately to each of the $k^2/2$ adjacency matrix blocks induced by the k-way node partitioning.
- Entropy reduction is caused by concentration of edges into sub-blocks.
- Must pay for encoding the node partitioning.

Short Digression about Buggy MDL schemes

- Mistakes like major coding inefficiencies can cause MDL schemes to work as badly as anything else.
- For example, one published paper encoded the node partitioning by first sending a complete permutation of the nodes, then the sizes of the pieces.
- That is more information than is needed, plus it can be viewed as a huge up-front cost for k=1, followed by a tiny marginal cost for increasing k
- The effect of this mistake can be seen in the following table.

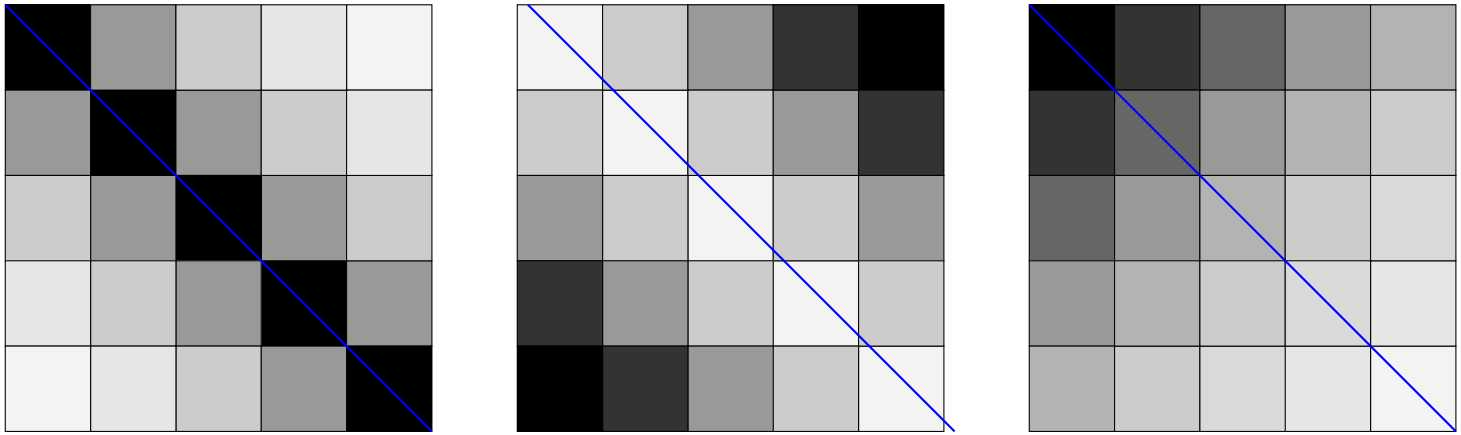
MDL Results for IMDB Actors vs Movies Graph

Source of Partitioning	modularity Q score		MDL w/ bug bits / edge		MDL G_{nm} bits / edge		MDL G_D bits / edge	
	orig	rand	orig	rand	orig	rand	orig	rand
None (Base Model)			18.51	18.51	18.51	18.51	17.11	17.53
None (K=1)	0.000	0.000	26.59	26.59	18.51	**18.51**	17.11	**17.53**
Graclus(K=4)	0.711	0.408	24.88	26.04	17.70	18.87	16.32	18.12
Graclus(K=16)	**0.848**	**0.462**	23.33	25.37	16.94	19.08	15.65	18.22
Graclus(K=64)	0.824	0.412	22.13	24.88	16.49	19.25	**15.50**	18.56
Graclus(K=256)	0.735	0.392	**21.77**	**24.60**	**16.45**	19.27	15.89	18.77
Graclus(K=1024)	0.654	0.373	27.04	29.99	16.66	19.54	16.41	19.01
Segregate Degrees		0.000				17.88		19.00

10-try results	Modu Q		MDL G_{nm}		MDL G_D	
graph	G	R	G	R	G	R
monksLk3	2	4	2	1	1	1
karate	4	4	1	1	1	1
dolphins	4	4	2	1	2	1
football	8	8	16	1	8	1
netscience	16	8	32	1	16	1
mani-facesDeg3	16	8	64	1	16	1
mani-swissK10	32	8	1024	1	128	1
powergrid-watts	32	32	32	1	32	1
road-USA	128	512	512	1	512	1
road-pa	256	32	1024	1	1024	1
random-deg4	16	16	1	1	1	1
gnutella-30	32	16	1	1	1	1
delicious	64	16	1024	1	1024	1
epinions	8	8	1024	256	16	1
flickr	16	8	1024	16	16	1
LinkedIn	8	8	1024	32	1024	1
LiveJournal01	256	8	1024	256	1024	1
answers	16	8	128	1	16	1
answers-2	16	8	8	1	1	1
as-newman	16	256	32	1	8	1
as-oregon-all	8	8	128	1	16	1

graph name	Modu Q		MDL G_{nm}		MDL G_D	
	G	R	G	R	G	R
AuthToPap-cond-mat	128	16	1024	1	512	1
AuthToPap-gr-qc	128	16	128	1	128	1
AuthToPap-hep-th	128	16	512	1	256	1
amazon_2003_all	16	8	1024	1	1024	1
bio-proteinsVespignani	16	16	32	1	1	1
Blog-nat05-6m	4	8	128	32	4	1
Cit-hep-ph	16	8	1024	256	64	1
clickstream-UsrToUrl	32	16	512	64	1	1
CoAuth-cond-mat	16	8	1024	1	512	1
CoAuth-hep-ph	64	8	1024	128	256	1
dblp-larsWcc	64	8	1024	1	1024	1
email-all-inOut	32	128	128	2	1	1
email-ijs	16	16	512	1	16	1
imdbDec07-ActToAct	16	8	1024	1024	1024	1
imdbDec07-ActToMov	16	8	1024	1	128	1
imdb-a-m-30countries	32	8	128	1	32	1
Patents	16	8	1024	1	1024	1
Post-nat05-6m	64	64	512	1	256	1
protein_dip	16	8	32	1	1	1
web-google	128	8	1024	128	1024	1
web-wt10g-trec	256	8	1024	32	1024	1

A flaw in this partitioned G_{nm} scheme, shared by many MDL "clustering metrics"



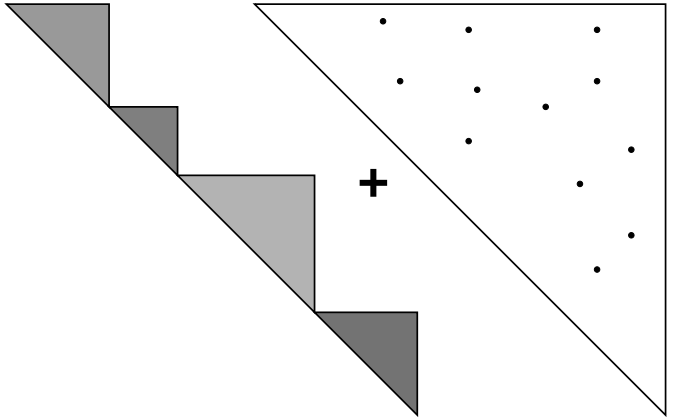
- Any of these kinds of adjacency-matrix block structure allows partitioned G_{nm} to compress better than unpartitioned G_{nm}.
- However, only one of them is the kind of cluster structure that users are probably expecting.

MDL using partitioned G_D model

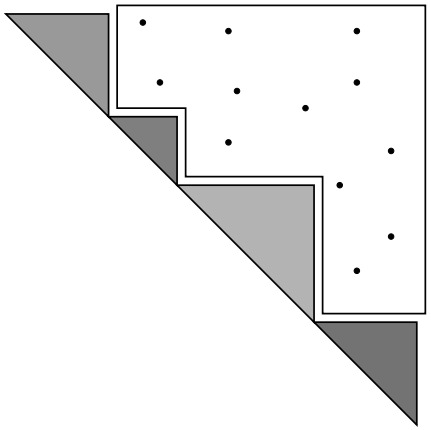
- The idea is to send both the graph's degree sequence, and a partitioning of its nodes in the message header.
- Then it would obviously be wasteful for the partitioning and the degree sequence to have high mutual information.
- The tricky part is devising an encoding scheme for the graph's edges that wrings as much entropy-reduction as possible out of those two sources of information.
- Our scheme isn't perfect, but it allowed us to run the experiment.

Encoding scheme for partitioned G_D

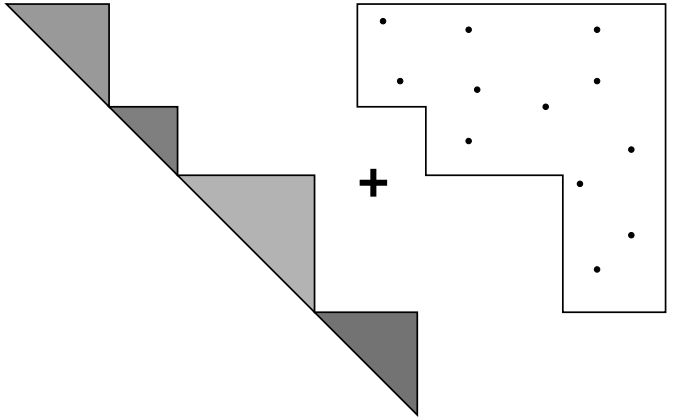
Encoding each triangle with base G_d scheme



Matrix decomposition



(Should really be encoding like this):



10-try results	Modu Q		MDL G_{nm}		MDL G_D	
graph	G	R	G	R	G	R
monksLk3	2	4	2	1	1	1
karate	4	4	1	1	1	1
dolphins	4	4	2	1	2	1
football	8	8	16	1	8	1
netscience	16	8	32	1	16	1
mani-facesDeg3	16	8	64	1	16	1
mani-swissK10	32	8	1024	1	128	1
powergrid-watts	32	32	32	1	32	1
road-USA	128	512	512	1	512	1
road-pa	256	32	1024	1	1024	1
random-deg4	16	16	1	1	1	1
gnutella-30	32	16	1	1	1	1
delicious	64	16	1024	1	1024	1
epinions	8	8	1024	256	16	1
flickr	16	8	1024	16	16	1
LinkedIn	8	8	1024	32	1024	1
LiveJournal01	256	8	1024	256	1024	1
answers	16	8	128	1	16	1
answers-2	16	8	8	1	1	1
as-newman	16	256	32	1	8	1
as-oregon-all	8	8	128	1	16	1

graph name	Modu Q		MDL G_{nm}		MDL G_D	
	G	R	G	R	G	R
AuthToPap-cond-mat	128	16	1024	1	512	1
AuthToPap-gr-qc	128	16	128	1	128	1
AuthToPap-hep-th	128	16	512	1	256	1
amazon_2003_all	16	8	1024	1	1024	1
bio-proteinsVespignani	16	16	32	1	1	1
Blog-nat05-6m	4	8	128	32	4	1
Cit-hep-ph	16	8	1024	256	64	1
clickstream-UsrToUrl	32	16	512	64	1	1
CoAuth-cond-mat	16	8	1024	1	512	1
CoAuth-hep-ph	64	8	1024	128	256	1
dblp-larsWcc	64	8	1024	1	1024	1
email-all-inOut	32	128	128	2	1	1
email-ijs	16	16	512	1	16	1
imdbDec07-ActToAct	16	8	1024	1024	1024	1
imdbDec07-ActToMov	16	8	1024	1	128	1
imdb-a-m-30countries	32	8	128	1	32	1
Patents	16	8	1024	1	1024	1
Post-nat05-6m	64	64	512	1	256	1
protein_dip	16	8	32	1	1	1
web-google	128	8	1024	128	1024	1
web-wt10g-trec	256	8	1024	32	1024	1

1-try results	Modu Q		MDL G_{nm}		MDL G_D	
graph	G	R	G	R	G	R
monksLk3	4	4	2	1	1	1
karate	4	4	1	1	1	1
dolphins	4	8	2	1	2	1
football	8	8	16	1	8	1
netscience	16	8	32	1	16	1
mani-facesDeg3	16	8	64	1	16	1
mani-swissK10	32	8	1024	1	128	1
powergrid-watts	32	32	32	1	32	1
road-USA	128	512	512	1	512	1
road-pa	256	32	1024	1	1024	1
random-deg4	16	16	1	1	1	1
gnutella-30	8	16	1	1	1	1
delicious	128	16	1024	1	1024	1
epinions	8	8	512	32	16	1
flickr	128	8	1024	16	128	1
LinkedIn	8	8	1024	16	1024	1
LiveJournal01	128	8	1024	128	1024	1
answers	16	8	64	1	16	1
answers-2	16	8	8	1	1	1
as-newman	8	512	32	1	8	1
as-oregon-all	8	16	128	1	16	1

graph name	Modu Q		MDL G_{nm}		MDL G_D	
	G	R	G	R	G	R
AuthToPap-cond-mat	128	16	1024	1	512	1
AuthToPap-gr-qc	64	16	256	1	128	1
AuthToPap-hep-th	128	16	512	1	256	1
amazon_2003_all	16	8	1024	1	1024	1
bio-proteinsVespignani	32	8	32	1	1	1
Blog-nat05-6m	4	8	128	64	4	1
Cit-hep-ph	16	8	1024	256	64	1
clickstream-UsrToUrl	32	16	16	1024	1	1
CoAuth-cond-mat	16	8	1024	1	512	1
CoAuth-hep-ph	64	8	1024	128	256	1
dblp-larsWcc	32	8	1024	1	1024	1
email-all-inOut	32	128	64	2	1	1
email-ijs	16	16	512	1	16	1
imdbDec07-ActToAct	16	8	1024	1024	1024	1
imdbDec07-ActToMov	16	8	1024	1	128	1
imdb-a-m-30countries	32	8	128	1	32	1
Patents	16	8	1024	1	1024	1
Post-nat05-6m	64	64	512	1	256	1
protein_dip	16	16	32	1	1	1
web-google	128	8	1024	128	1024	1
web-wt10g-trec	128	8	1024	32	1024	1

Summary

- Three-way comparison of degree sequence models.
  Power-law was not the best.
- Two-way comparison of unpartitioned G_{nm} and G_D graph models.
  Tiny benchmark graphs were atypical.
- Explanations for why certain published objective functions see clusters where they shouldn't.
  - Modularity-Q fails to consider statistical significance.
  - MDL results can be distorted by inefficient coding.
  - Alleged "Clustering Metrics" can be optimized by unanticipated classes of solutions.
- We have done the first MDL experiments using a partitioned G_D model. It seemed to do the right thing, but that is probably not the end of the story.

Some possible future work

- Design a better encoding scheme for partitioned G_D model.
- Suggested by [Guimera et al, 2004]: Derive a rigorous test for the statistical significance of a graph clustering, that works for graphs with highly-skewed degree sequences.