Sampling and Inference for Beta Neutral-to-the-Left Models of Sparse Networks

Benjamin Bloem-Reddy, Adam Foster, Emile Mathieu, Yee Whye Teh

Department of Statistics, University of Oxford

Contents

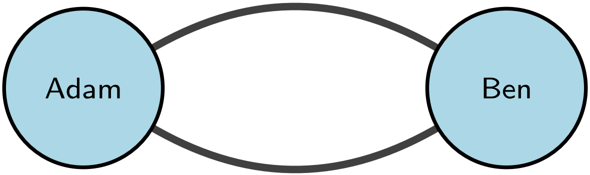
Temporal networks

Example

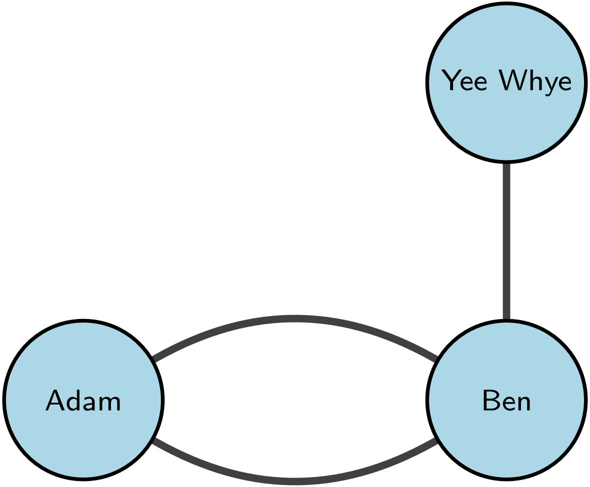
- Messages sent between people over time
- Protein-protein interactions

Temporal networks

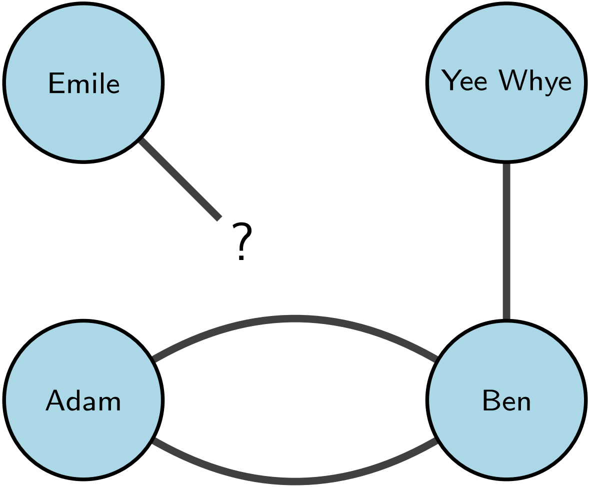
Temporal networks



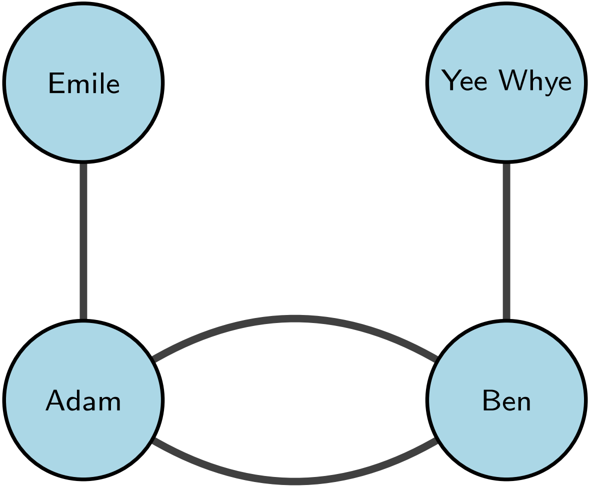
Temporal networks



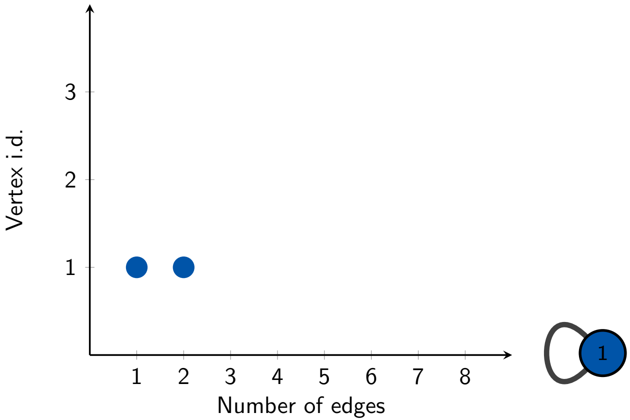
Temporal networks



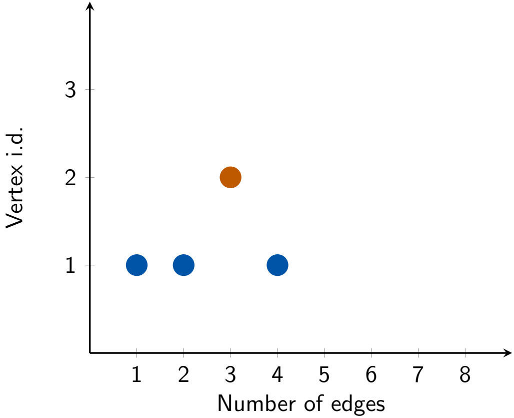
Temporal networks



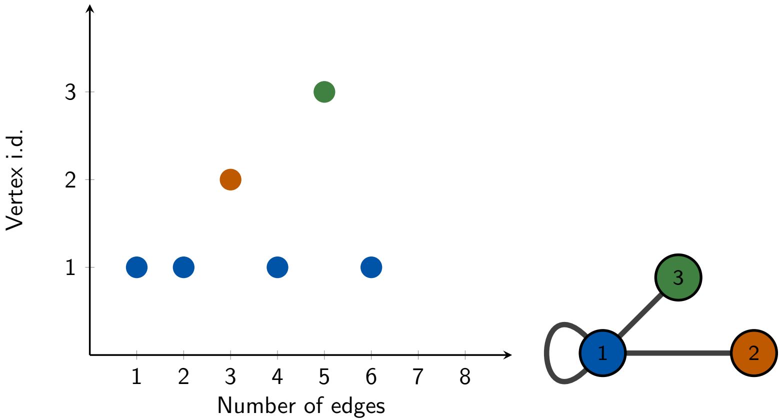
Edges and vertices



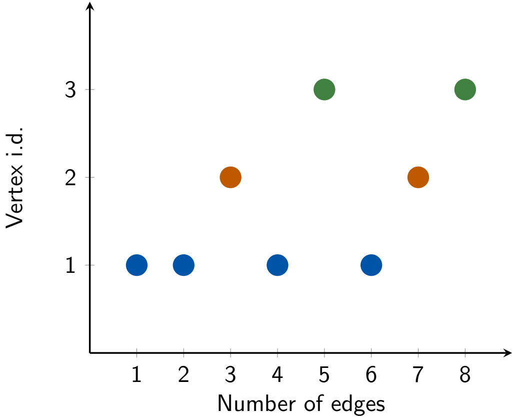
Edges and vertices

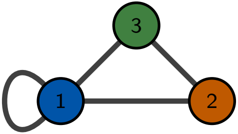


Edges and vertices

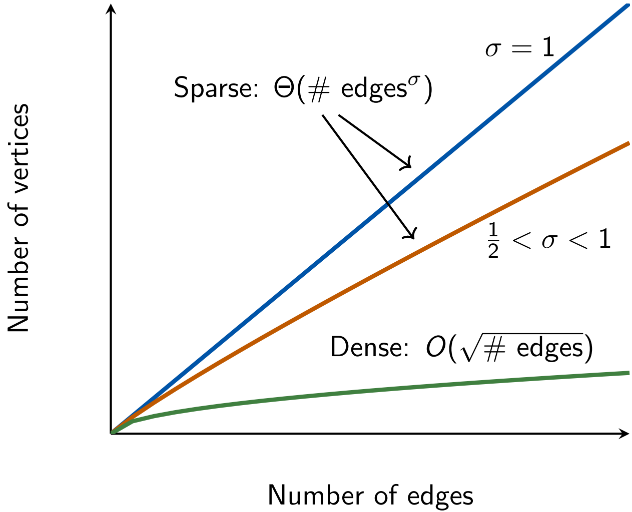


Edges and vertices



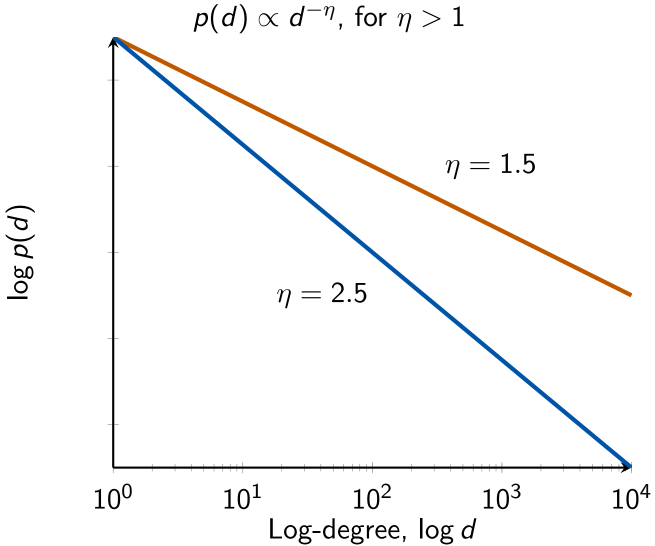


Sparsity



$\sigma = 1$
Sparse: $\Theta(\#\ \text{edges}^{\sigma})$
$\frac{1}{2} < \sigma < 1$
Dense: $O(\sqrt{\#\ \text{edges}})$
Number of vertices
Number of edges

Power law degree distribution



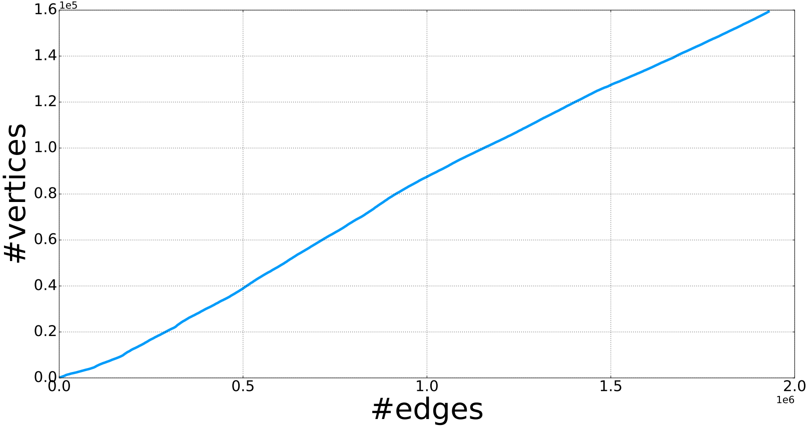
Sparsity and power law

$$\textbf{Sublinear}\text{ sparsity} \quad \Longleftrightarrow \quad \eta \in (1,2)$$
$$\textbf{Linear}\text{ sparsity} \quad \Longleftrightarrow \quad \eta > 2$$

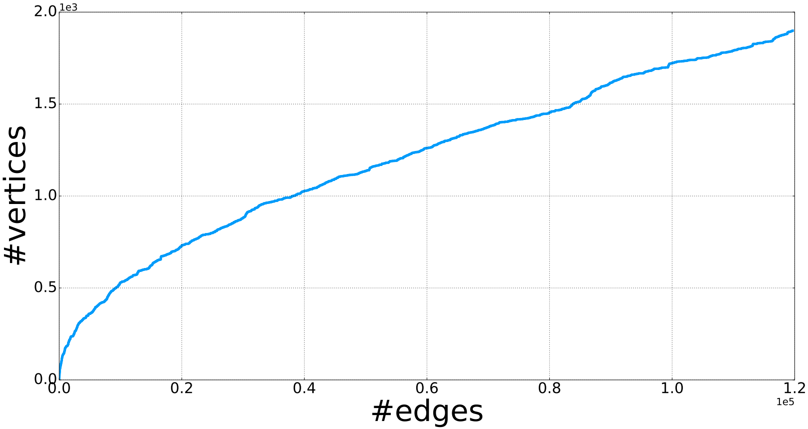
Empirical study

SNAP dataset [1]	# of vertices	# of edges
Ask Ubuntu	159,316	964,437
UCI social network	1,899	20,296
⋮	⋮	⋮

Ask Ubuntu



UCI social network



Models

Vertex exchangeable

Models

dense

sparse

Vertex exchangeable

Models

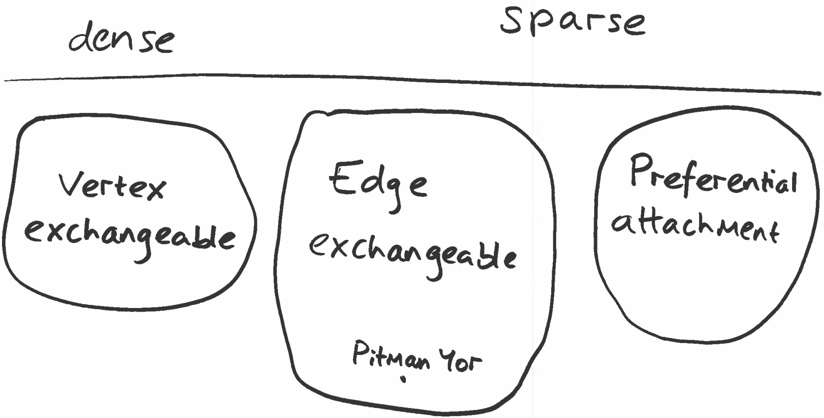
dense

sparse

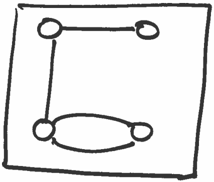
Vertex exchangeable

Preferential attachment

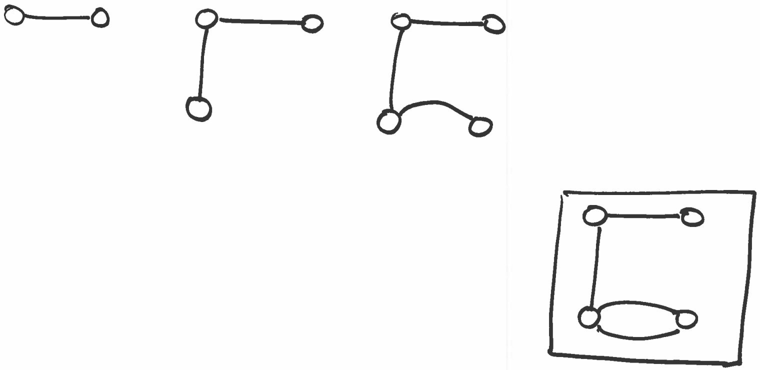
Models


dense
sparse
Vertex exchangeable
Edge exchangeable
Pitman Yor
Preferential attachment

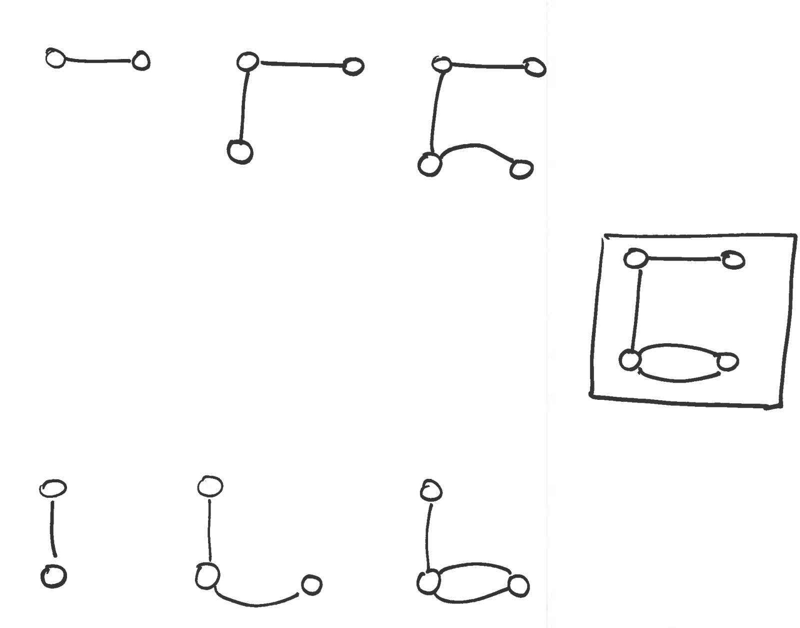
Edge exchangeable models [2], [3]



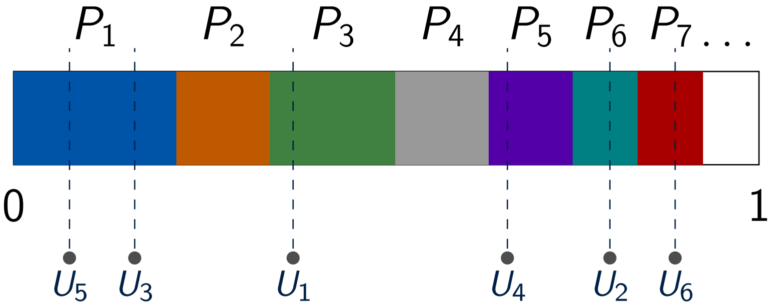
Edge exchangeable models [2], [3]



Edge exchangeable models [2], [3]

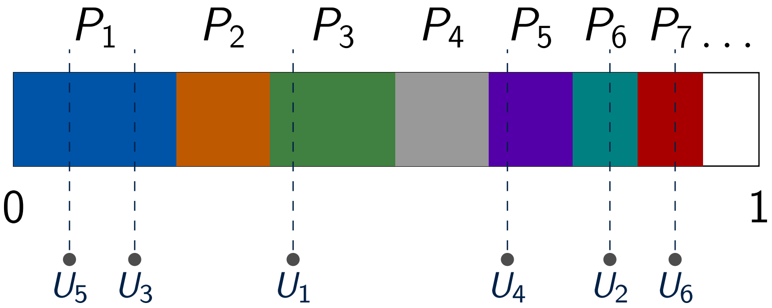


Factorizable ("rank one") paintbox representation

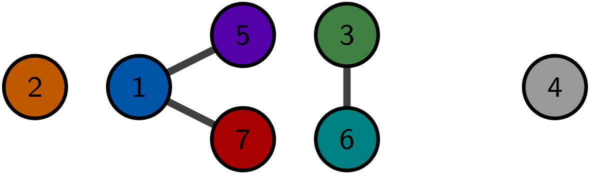


Sequence of edges: $((3, 6), (1, 5), (1, 7))$

Factorizable ("rank one") paintbox representation



Sequence of edges: $((3,6),(1,5),(1,7))$

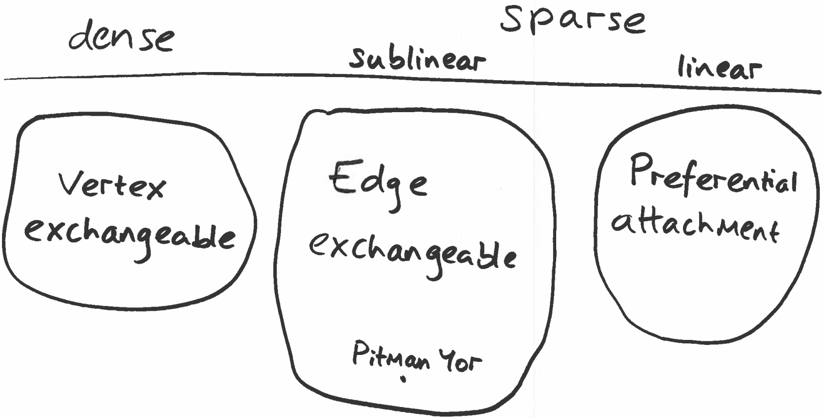


Paintbox representation

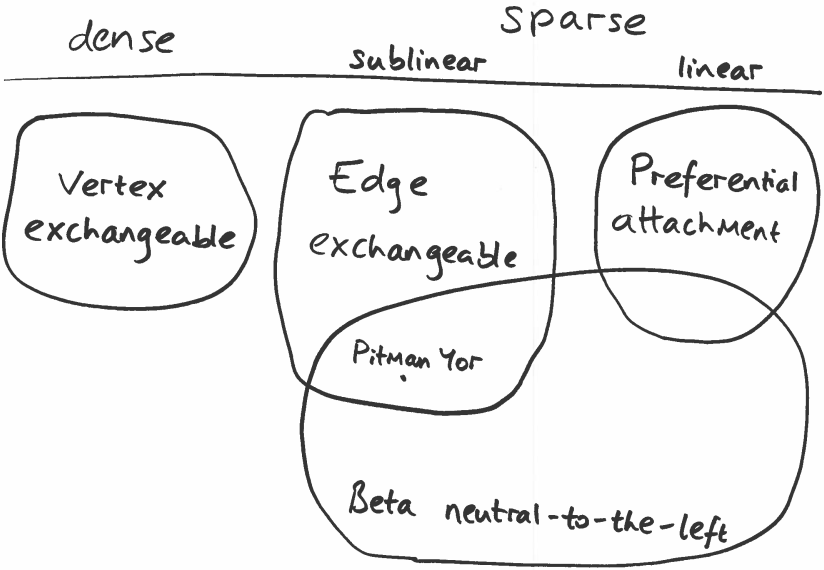
Consequences for edge exchangeable models:

- Rate of vertex arrival gets slower and slower
  $\Rightarrow$ sublinear sparsity: # vertices$= o(n)$
- Edges pile up
  $\Rightarrow$ linear scaling of degrees: $d_{j,n} = \Theta(n)$

Models

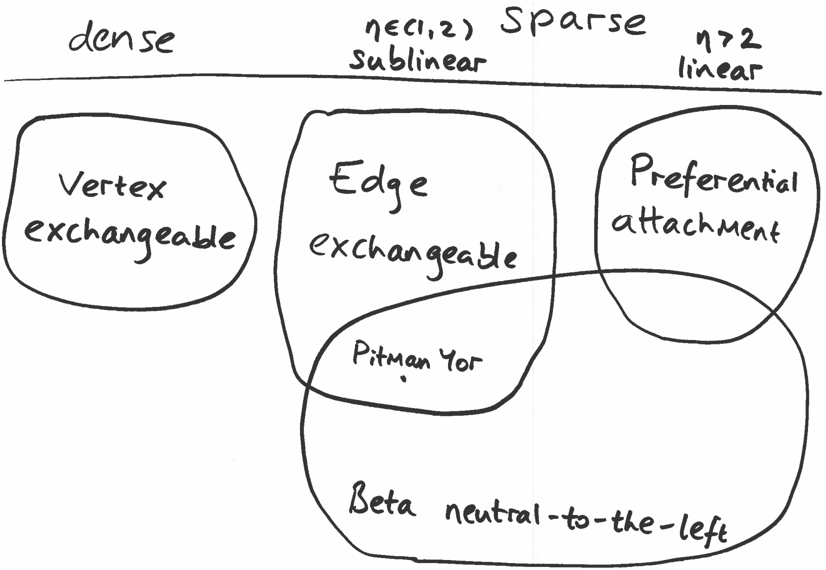

dense
sparse
sublinear
linear
Vertex exchangeable
Edge exchangeable
Pitman Yor
Preferential attachment

Models



dense
sparse
sublinear
linear
Vertex exchangeable
Edge exchangeable
Preferential attachment
Pitman Yor
Beta neutral-to-the-left

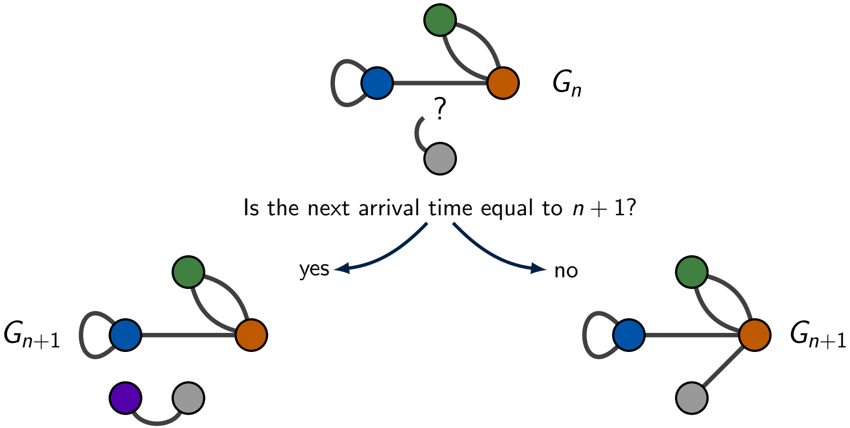
Models



dense
$\eta \in (1,2)$
Sublinear
Sparse
$\eta > 2$
linear
Vertex exchangeable
Edge exchangeable
Preferential attachment
Pitman Yor
Beta neutral-to-the-left

Beta Neutral-to-the-left Model [4]

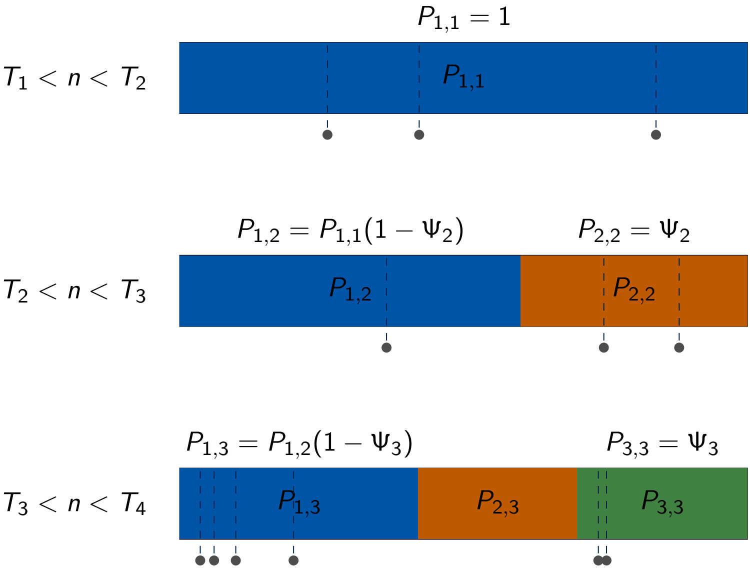
1. Generate arrival times $1 = T_1 < T_2 < T_3 < \ldots$ in any way.
2. Generate ends of edges sequentially:



$$n + 1 = T_{K+1} \Rightarrow Z_{n+1} = K + 1$$

$$P[Z_{n+1} = j] \propto \mathrm{deg}_{j,n} - \alpha$$

Sequence of paintboxes representation



Sampling and inference

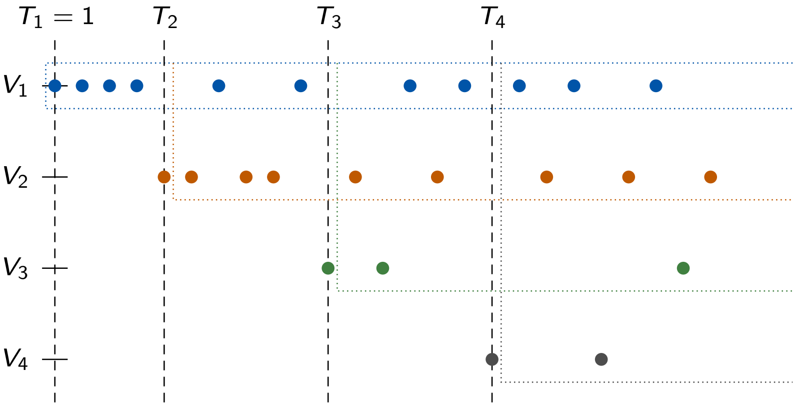
Why a paper on sampling and inference for BNTL models?

Sampling and inference

Why a paper on sampling and inference for BNTL models?

- Inference is notoriously difficult for *non-exchangeable* structures
- Need to identify *exchangeable substructures*

Exchangeable substructure



Gibbs structure

The joint probability has **Gibbs structure** due to left-neutrality

$$P(\text{graph}|\mathbf{T}) = \prod_{j=1}^{K} P(\text{choose } j \; d_j - 1 \text{ times out of } n - T_j \text{ trials })$$

- $K = \#\text{vertices}$
- $n = \#\text{edges}$
- d_j = degree of vertex j
- T_j = arrival time of vertex j

Gibbs structure

Explicitly,

$$p(\text{graph}|\mathbf{T}) = \frac{\Gamma(d_1 - \alpha)}{\Gamma(n - K\alpha)} \prod_{j=2}^{K} \frac{\Gamma(T_j - j\alpha)\Gamma(d_j - \alpha)}{\Gamma(T_j - 1 - (j-1)\alpha)\Gamma(1 - \alpha)}$$

- $K = \#$vertices
- $n = \#$edges
- $d_j =$ degree of vertex j
- $T_j =$ arrival time of vertex j

Available data

Observation	**Unobserved variables**
Entire history	$\alpha, \phi, \boldsymbol{\Psi}$
Degrees in arrival order	$\alpha, \phi, \boldsymbol{\Psi}, \mathbf{T}$
Snapshot	$\alpha, \phi, \boldsymbol{\Psi}, \mathbf{T}, \sigma$

- α = BTNL parameter $\in (-\infty, 1)$
- ϕ = arrival distribution parameters
- $\boldsymbol{\Psi}$ = latent sociabilities
- $\mathbf{T}$ = arrival times
- σ = arrival order

Gibbs sampler

Variable	**Gibbs sampling scheme**
α	MCMC, e.g. slice sampling
ϕ	Depends on arrival dist. family Λ_ϕ
$\boldsymbol{\Psi}$	$\Psi_j \vert \text{graph}, \boldsymbol{\Psi}_{\setminus j} \sim \text{Beta}(d_j - \alpha, \bar{d}_{j-1} - (j-1)\alpha)$ where $\bar{d}_j = \sum_{i=1}^{j} d_j$ can marginalise out $\boldsymbol{\Psi}$
$\mathbf{T}$	Simple update for T_j, can't move past neighbours
σ	Initialise in descending degree order use M-H with adjacent swap proposal $\sigma_j \leftrightarrow \sigma_{j+1}$ fast to compute due to Gibbs structure

Point estimation

If entire history observed,**maximum a posterior** (or **maximum likelihood**) estimates for α, ϕ computable

Experiments

- Gibbs: parameter recovery
- Gibbs: scalability
- Point estimation with massive graphs

Parameter recovery

- Simulate 500 edges with fixed α
- Arrivals either $\mathcal{PYP}$ or Geom
- Observe final snapshot of the graph

Gen. arrival distn.	Inference model	$\lvert\hat{\alpha} - \alpha^*\rvert$	Pred. log-lik.
$\mathcal{PYP}(1.0, 0.75)$	$(\tau, \mathcal{PYP}(\theta, \tau))$	**0.046** $\pm$ **0.002**	**-2637.0** $\pm$ **0.1**
$\mathcal{PYP}(1.0, 0.75)$	$(\alpha, \text{Geom}(\beta))$	0.049 $\pm$ 0.004	-2660.5 $\pm$ 0.7
Geom(0.25)	$(\tau, \mathcal{PYP}(\theta, \tau))$	0.086 $\pm$ 0.002	-2386.8 $\pm$ 0.1
Geom(0.25)	$(\alpha, \text{Geom}(\beta))$	**0.043** $\pm$ **0.003**	**-2382.6** $\pm$ **0.2**

Scalability

- Simulate with fixed α and Geom(β) arrivals

	100 edges	10000 edges
$\lvert\hat{\alpha} - \alpha^*\rvert$	0.12 ± 0.01	0.01 ± 0.00
$\lvert\hat{\beta} - \beta^*\rvert$	0.02 ± 0.00	0.00 ± 0.00
Effective Sample Size	0.90 ± 0.04	0.75 ± 0.08
Runtime (s)	21 ± 0	2267 ± 2

- Runtime linear in #edges
- Most expensive Gibbs update is for **T**

MLEs for SNAP datasets

- SNAP datasets
- Fit point estimates for α, ϕ
- Fit: coupled $\mathcal{PYP}$, uncoupled $\mathcal{PYP}$ and Geom(β) arrivals

MLEs for SNAP datasets

Ask Ubuntu

- Estimates of $\mathcal{PYP}$ parameters vary significantly between coupled and uncoupled
  - $\hat{\theta}, \hat{\alpha} = 18080, 0.25$
  - $\hat{\theta}, \hat{\alpha}, \hat{\tau} = -0.99, -2.54, 0.99$
- Edge exchangeable models misspecified ($\eta > 2$)
- Using Geom estimates η well

Conclusion

- BNTL models are *flexible*
- BNTL models are *tractable*

Future work

- Scalability of inference
  - Metropolis-Hastings to update $\mathbf{T}$ altogether
  - Variational inference for σ

References

[1] Jure Leskovec and Andrej Krevl. SNAP Datasets: Stanford large network dataset collection. `http://snap.stanford.edu/data`, June 2014.

[2] Diana Cai, Trevor Campbell, and Tamara Broderick. Edge-exchangeable graphs and sparsity. In *Advances in Neural Information Processing Systems*, pages 4249–4257, 2016.

[3] Harry Crane and Walter Dempsey. Edge exchangeable models for interaction networks. *Journal of the American Statistical Association*, 2017. In press.

[4] Benjamin Bloem-Reddy and Peter Orbanz. Preferential attachment and vertex arrival times. *arXiv preprint arXiv:1710.02159*, 2017.