

Bayesian modeling of biological networks

Frédéric Bois[‡], **Ghislaine Gayraud**^{*}, Sagnik Datta[†]

[‡] I.N.E.R.I.S

^{*} Laboratoire de Math. Appliquées de Compiègne, UTC

[†] BMBI, UTC

Colloque CARTABLE, INRA-TOULOUSE

- 1 Motivation
- 2 Graph Modeling
- 3 Bayesian inference
- 4 Numerical experiments
- 5 Conclusion

Biological Networks' (NWs) examples

- Protein-protein interaction NWs
- Signaling pathways
- Gene regulation NWs
- Metabolic NWs
-

AIM

	Graph	Nodes	Edges
Notation	\mathcal{G}	$\mathbf{V} = (v_i)$ $1 \leq i \leq N$	$\mathbf{E} = (e_{i,j})$ $1 \leq i, j \leq N$
Observed	NO	$\mathbf{X} = (X_{i,k})_{(i,k)}$ $1 \leq i \leq N$ $1 \leq k \leq n$	NO

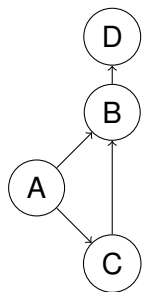
⇒ GOAL: Infer on NW's structure

How do we tackle the structure learning problem?

- \leftrightarrow Probabilistic modeling of the graph $\mathcal{G} = \{(\mathbf{V}, \mathbf{E})\}$ from \mathbf{X} ;
- \leftrightarrow Statistical procedure to recover the structure of \mathcal{G} ;
- \leftrightarrow Implementation and simulation study.

- 1 Motivation
- 2 Graph Modeling
 - Directed Acyclic Graph (D.A.G.)
 - Directed graphs involving loop(s)
 - Marginal likelihood
- 3 Bayesian inference
- 4 Numerical experiments
- 5 Conclusion

Bayesian Network (BN): probabilistic model of D.A.G.



- $N = 4$
- $\mathbf{X} = (X_A, X_B, X_C, X_D)$
- $Pa(\cdot)$: Parents of ' \cdot ';
 $Pa(X_A) = \emptyset$;
 $Pa(X_B) = \{X_A, X_C\}$;
 $Pa(X_C) = \{X_A\}$;
 $Pa(X_D) = \{X_B\}$;
- $\mathcal{L}(\mathbf{X}|\mathcal{G})$: distribution of \mathbf{X} given \mathcal{G}

\leftrightarrow Factorization definition of BN:

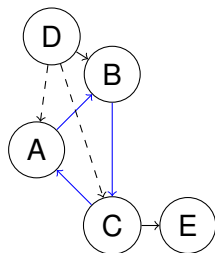
$$\mathcal{L}(\mathbf{X}|\mathcal{G}) = \mathcal{L}(X_A|Pa(X_A))\mathcal{L}(X_B|Pa(X_B))\mathcal{L}(X_C|Pa(X_C))\mathcal{L}(X_D|Pa(X_D))$$

References:

- Heckerman, Geiger and Chickering (1995) in *Machine Learning*
- Pearl (1988) Book, Morgan Kaufmann Publishers

- 1 Motivation
- 2 Graph Modeling
 - Directed Acyclic Graph (D.A.G.)
 - Directed graphs involving loop(s)
 - Marginal likelihood
- 3 Bayesian inference
- 4 Numerical experiments
- 5 Conclusion

Loops considered as super nodes



- $N = 5$
- $\mathbf{X} = (X_A, X_B, X_C, X_D, X_E)$
- $L = \text{Loop} = \text{super node}$
 $Y_L = (X_A, X_B, X_C)$
- $Pa(\cdot)$: Parents of ' \cdot ';
 $Pa(X_D) = \emptyset$;
 $Pa(Y_L) = \{X_D\}$;
 $Pa(X_E) = \{X_C\}$;

$$\Leftrightarrow \mathcal{L}(\mathbf{X}|\mathcal{G}) = \mathcal{L}(Y_L|Pa(Y_L)) \times \mathcal{L}(X_D|Pa(X_D)) \times \mathcal{L}(X_E|Pa(X_E))$$

References:

- Bois, Datta, Gayraud (2016) work in preparation

Models for acyclic parts

$$\mathcal{L}(X_{ac}|Pa(X_{ac})) = \int \mathcal{L}(X_{ac}|Pa(X_{ac}), \theta_{ac}) d\Pi(\theta_{ac})$$

- θ_{ac} nuisance parameter
- n : number of data per nodes;
- $X_{ac} = (X_{ac,i})_{1 \leq i \leq n}$; $Pa(X_{ac})$ dim = $n \times k$.

Discrete:

$$\begin{cases} \theta_{ac,j,k} = \mathbb{P}(X_{ac,i} = k | Pa(X_{ac}) = j, \theta_{ac}), \\ (\theta_{ac,j,k})_k \sim \Pi_j : \text{Dirichlet prior} \end{cases}$$

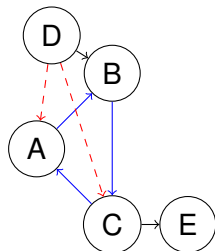
Continuous:

$$\begin{cases} X_{ac}|Pa(X_{ac}), \theta_{ac} \sim \mathcal{N}_n \left(\begin{pmatrix} 1 & Pa(X_{1,ac}) \\ 1 & Pa(X_{2,ac}) \\ \dots & \dots \\ 1 & Pa(X_{n,ac}) \end{pmatrix} \begin{pmatrix} \beta_1 \\ \beta_2 \\ \dots \\ \beta_{k+1} \end{pmatrix}, \lambda^{-1} I_n \right) \\ \theta_{ac} = (\beta, \lambda) \sim \Pi \text{ Normal Gamma prior OR } g\text{-Zellner prior} \end{cases}$$

Models for loops

$$\mathcal{L}(Y_L | Pa(Y_L)) = \int \mathcal{L}(Y_L | Pa(Y_L), \theta_L) d\Pi(\theta_L)$$

- θ_L nuisance parameter
- n : number of data per nodes;
- $Y_L = (X_1, \dots, X_l)$ $\dim = n \times l$;
- $Pa(X_L)$ $\dim = n \times k$.



Continuous:

$$\left\{ \begin{array}{l} Y_L = \begin{pmatrix} 1 & Pa(X_{1,L}) \\ 1 & Pa(X_{2,L}) \\ \dots & \dots \\ 1 & Pa(X_{n,L}) \end{pmatrix} \begin{pmatrix} \beta_{1,1} & \dots & \beta_{1,l} \\ \beta_{2,1} & \dots & \beta_{2,l} \\ \dots & \dots & \dots \\ \beta_{k+1,1} & \dots & \beta_{k+1,l} \end{pmatrix} + u, \\ u = (u_{i,j})_{1 \leq i \leq n; 1 \leq j \leq l} \\ u_i \stackrel{i.i.d.}{\sim} \mathcal{N}_l(0, \Sigma) \\ \theta_L = (\beta, \Sigma) \sim \Pi \propto \beta \times \text{Inverse-Wishart} \end{array} \right.$$

- 1 Motivation
- 2 Graph Modeling
- 3 Bayesian inference**
- 4 Numerical experiments
- 5 Conclusion

Bayesian: random \mathcal{G}

↪ updated knowledge about \mathcal{G} through the posterior distribution of \mathcal{G}

- Combine $\mathcal{L}(\mathbf{X}|\mathcal{G})$ with the prior π to update the distribution of $\mathcal{G}|\mathbf{X}$, i.e. the posterior distribution $\pi^{\mathbf{X}}$ given the data \mathbf{X} :

$$\pi^{\mathbf{X}}(\mathcal{G}) \propto \mathcal{L}(\mathbf{X}|\mathcal{G}) \times \pi(\mathcal{G})$$

- Nice features:
 - It provides not only a single \mathcal{G} but an updated distribution of \mathcal{G}
 - $\pi^{\mathbf{X}}(\mathcal{G})$ can be summarized through Bayesian estimates
- How π may be chosen? ↪ (a) belief/knowledge; (b) practical choice: posterior tractable; (c) ~~theoretical point of view~~

Priors on \mathcal{G}

- Independent Bernoulli $e_{i,j} \sim B(p_{i,j})$ with $p_{i,j} \in (0, 1)$

$$\pi_B(\mathcal{G}) = \prod_{1 \leq i,j \leq N} p_{i,j}^{e_{i,j}} (1 - p_{i,j})^{1 - e_{i,j}}$$

- Degree prior,

$$\pi_D(\mathcal{G}) \propto \prod_{1 \leq i \leq N} \sum_{j=1}^N e_{i,j}^{-\gamma} \text{ with } \sum_{j=1}^N e_{i,j} > 0 \text{ and } \gamma > 0$$

- Concordance prior,

$$\pi_C(\mathcal{G}) \propto \exp(-\rho \left(\sum_{(i,j) \in I_{\tilde{E}}} |a_{i,j} - e_{i,j}| \right)) \text{ with the prior matrix}$$

$$A = (a_{i,j})_{(i,j) \in I_{\tilde{E}}} \text{ where } a_{i,j} \in \{-1, 1\}, \tilde{E} \subset \mathbf{E} \text{ and } \rho > 0$$

- ...

↪ Prior total: $\pi(\mathcal{G}) \propto \pi_B(\mathcal{G}) \times \pi_D(\mathcal{G}) \times \pi_C(\mathcal{G}) \times \dots$

MCMC algorithm (DAG)

Notation. \mathcal{G}^t : current graph at the t -th iteration; \mathcal{G}^p : proposal graph

- 1 Select (deterministic) $e_{i,j}^t$ in \mathbf{E} with $i \neq j$
- 2 $e_{i,j}^p | e_{i,j}^t \sim B(p_{i,j})$ $e_{i,j}^p = 1$ provided \mathcal{G}^p is still a DAG
- 3 Acceptance ratio :

$$\delta = \min\left(1, \left(\frac{\mathcal{L}(\mathbf{X}|\mathcal{G}^p)\pi(\mathcal{G}^p)P(\mathcal{G}^t|\mathcal{G}^p)}{\mathcal{L}(\mathbf{X}|\mathcal{G}^t)\pi(\mathcal{G}^t)P(\mathcal{G}^p|\mathcal{G}^t)}\right)\right)$$

- 4 Choose $\mathcal{G}^{t+1} = \begin{cases} \mathcal{G}^p & \text{with probability } \delta \\ \mathcal{G}^t & \text{with probability } 1 - \delta \end{cases}$

- 1 Motivation
- 2 Graph Modeling
- 3 Bayesian inference
- 4 Numerical experiments**
- 5 Conclusion

- DAG

- simulated networks: 5 to 120 nodes; 100 data points per node
- life biological network "EGFR" : 14 nodes; 200 data points per node
- continuous and discrete data (only discrete for the "EGRF")

- LOOP

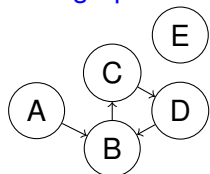
- simulated networks with 5 and 14 nodes with 50 data points per node
- continuous data only

- Performance analyses

- Convergence : Gelman' \hat{R} convergence diagnostic
- Comparison with Structmcmc (R Software): Mukherjee and Speed, 2008
- Edge posterior distributions through their mean
- Accuracy curve = (true positive edges + true negative edges)/ number of possible edges

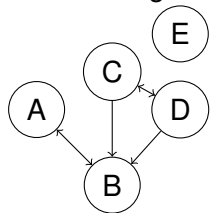
Structure learning; 10 million iterations with 1 million burning runs

True graph:



↔ `Graph_sampler_Loop` software written in C

Present edges with posterior greater than 0.5



- 1 Motivation
- 2 Graph Modeling
- 3 Bayesian inference
- 4 Numerical experiments
- 5 Conclusion**

Concluding remarks

- Directed graphs Modeling with different sets of structure prior (concordance, Bernoulli, degree,...);
- Software/algorithms : time-efficient & convergent-efficient;

	N	Number of iterations	Time
D.A.G.	30	$2 \cdot 10^7$	$\simeq 4.5$ mn
D.A.G.	100	$5 \cdot 10^7$	$\simeq 15$ mn
Loop	14	10^8	$\simeq 25$ mn

- For DAG: works efficiently with 120 nodes and 100 data whatever the nature of data (discrete or continuous)
- For directed graphs involving loops: first time of such stochastic modeling
- Only on simulated data
- Limitations : ultra-high dimension when N is huge compared to n
Verzelen, 2012 "no statistical procedure can provide satisfying results" when $d_{max} \log(N/d_{max}) \asymp n$, d_{max} maximal degree