Importance sampling techniques for sequentially choosing interventions in network structure learning

James Henderson and George Michailidis

Annual Report: December 11, 2014

Background

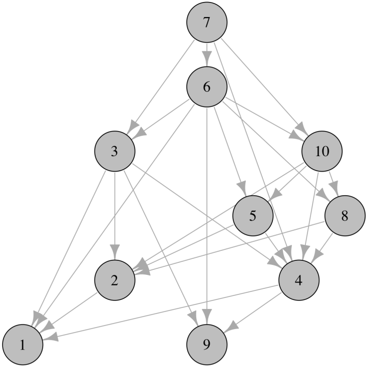
Framework for Sequential Design

Importance Sampling

Estimating the Entropy for Unseen Interventions

Conclusion

Background



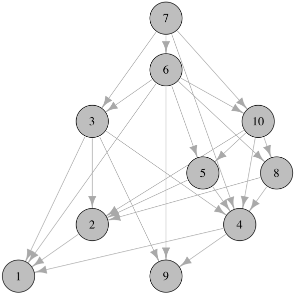
- Network representations play an important role in our understanding of complex biological systems.
- Reconstructing unknown networks from data is a key problem in systems biology.

- Intervention data such as gene knockouts are often used to learn directed networks.
- Directed networks formally represented as Directed Acyclic Graphs (DAGs).

Related literature

- Given observational data and a linear SEM, (Hauser 2012) presents algorithms for selecting interventions.
- Working with multi-target interventions and a likelihood based on steady states of an ODE model, (Molinelli 2013) use belief propogation to jointly estimate structural and model parameters.
- Our work is close in spirit to that of (Murphy 2001) and (Tong 2001) which use MCMC to explore the posterior and importance sampling to choose interventions in networks with discrete data.

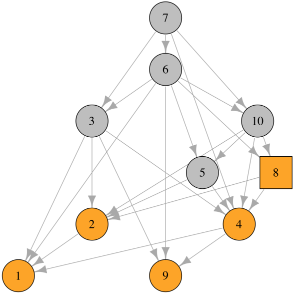
Overview



- The underlying system has a representation as a DAG $G = (V, E)$ with V known but E unkown.
- Observe data Y_i^α on nodes $i \in V$ from interventions $\alpha \in \mathcal{A}$
- For simplicity assume the action space is $\mathcal{A} = \{1, ..., d\}$

- After collecting some initial data $(Y^{\alpha_t}, \alpha_t)_{t=1}^T$ we want to choose $\alpha_{T+1} \in \mathcal{A}/\{\alpha_1, ..., \alpha_T\}$.
- Need to measure anticipated improvement for potential actions α_{T+1}.

Overview



- The underlying system has a representation as a DAG $G = (V, E)$ with V known but E unkown.
- Observe data Y_i^{α} on nodes $i \in V$ from interventions $\alpha \in \mathcal{A}$
- For simplicity assume the action space is $\mathcal{A} = \{1, ..., d\}$

- After collecting some initial data $(Y^{\alpha_t}, \alpha_t)_{t=1}^{T}$ we want to choose $\alpha_{T+1} \in \mathcal{A}/\{\alpha_1, ..., \alpha_T\}$.
- Need to measure anticipated improvement for potential actions α_{T+1}.

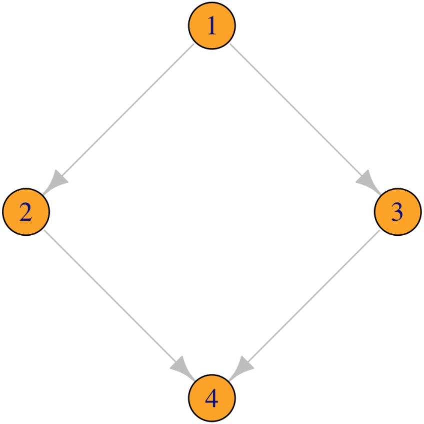
Order Hierarchy

- The paths of a DAG $G = (V, E)$ induce a partial order

$$\Pi := \{(i,j) \in V^2 : \text{ there is a path from } \; i \; to \; j.\}.$$

- A (strict) partial order on V is a set $\Pi \subset V^2$ that is:
  1. Anti-reflexive $(i,i) \notin \Pi$
  2. Anti-symmetric $(i,j) \in \Pi \Rightarrow (j,i) \notin \Pi$
  3. Transitive $(i,j) \in \Pi, (j,k) \in \Pi \Rightarrow (i,j) \in \Pi$.
- A (strict) linear order on V is a partial order $\Lambda \subset V^2$ with the additional property that all pairs are comparable; i.e. $(i,j) \notin \Lambda \Rightarrow (j,i) \in \Lambda$
- A linear order $\Lambda \subset V^2$ is a linear extension of Π if $\Pi \subset \Lambda$

Order Hierarchy



$$\Pi = \{(1,2),(1,3),(1,4),(2,4),(3,4)\}$$

$$\Lambda_1 = \Pi \cup \{(2,3)\}, \Lambda_2 = \Pi \cup \{(3,2)\}$$

Pairwise representations

- Consider a set $V = \{1, .., d\}$.
- A linear order Λ on V can be represented as a vector L ordering the elements in V,

$$L \sim \Lambda \iff \forall i, j \in \{1, ..., d\}, i < j, (L_i, L_j) \in \Lambda.$$

- Let $s = \binom{d}{2}$ and $\xi : V^2 \to \{1, ..., s\}$ impose a canonical ordering on the pairs in V (say lexicographical).
- A partial order Π on V can be represented as a vector $\pi \in \{-1, 0, 1\}^s$ where

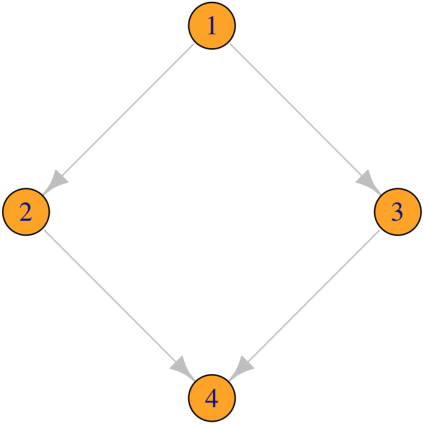
$$\pi_{\xi(i,j)} = \begin{cases} 1, & i < j, (i,j) \in \Pi \\ -1, & i > j, (j,i) \in \Pi \\ 0, & \text{otherwise.} \end{cases}$$

Pairwise representations continued

- For a DAG G and nodes i, j let g_{ij} be the length of the shortest path from i to j or zero if no such path exists.
- Then G can be represented as a vector $\gamma \in \mathbb{Z}^s$ where

$$\gamma_{\xi(i,j)} = \begin{cases} g_{ij}, & i < j, (i,j) \in \Pi \\ -g_{ij}, & i > j, (j,i) \in \Pi \\ 0, & \text{otherwise.} \end{cases}$$

Pairwise representation example



$$\pi = (1, 1, 1, 0, 1, 1)$$
$$\gamma = (1, 1, 2, 0, 1, 1)$$
$$L_1 = (1, 2, 3, 4)$$
$$L_2 = (1, 3, 2, 4)$$

Model

- We use the following model for the observed effect levels Y^α under intervention α:

$$\gamma \sim \text{Uniform}(\mathcal{G})$$

$$\{\beta_i^\alpha | \gamma_i^\alpha\} \overset{ind}{\sim} \text{Beta}(2, 2 + \gamma_i^\alpha)$$

$$\{\psi_i^\alpha\} \overset{iid}{\sim} \text{Binomial}(.5)$$

$$\mu_i^\alpha := \mu_i(\beta_i^\alpha, \psi_i^\alpha) = \begin{cases} \mu_i^0 - \mu_i^0 \beta_i^\alpha, & \psi_i^\alpha = 1 \\ \mu_i^0 + (1 - \mu_i^0)\beta_i^\alpha, & \psi_i^\alpha = 1 \end{cases}$$

$$Y_{ij}^\alpha | \mu \overset{ind}{\sim} \text{TN}(\mu_i^\alpha, \sigma^2); i = 1, ..., d; j = 1, ..., n.$$

- Here $\gamma_i^\alpha = |\gamma_{\xi(\alpha,i)}|$ if $(\alpha, i) \in \Pi$ and is zero otherwise.
- ψ_i^α is the direction of the effect of α on i.
- The observed and true expression levels are assumed to lie in the unit interval $[0, 1]$ as are the baseline expression levels μ_i^0.

Likelihood features

- The magnitudes and signs of effects are independent given the DAG γ, allowing us to work with the marginal likelihoods

$$p(y_i^\alpha|\gamma) = \int p(y_i^\alpha|\mu(\beta_i^\alpha, \psi_i^\alpha))p(\beta_i^\alpha|\gamma)p(\psi_i^\alpha)d(\beta, \psi).$$

- The low-dimensional integrals needed for marginal likelihoods can be precomputed so that computing $p(y_i^\alpha|\gamma)$ reduces to summing an appropriate set of terms.
- For fixed α the likelihood is level-modular, meaning it depends only on the length of the shortest path from α to i and not on the path itself.

Posterior Marginals

- The structural information contained in data $y^{\alpha_{1:T}}$ from an initial set of interventions $\alpha^{1:T}$ is often summarized using the posterior probability of edges,

$$p(\gamma_{\xi(i,j)} = k | y^{\alpha_{1:T}}) \propto \sum_{\gamma} 1[\gamma_{\xi(i,j)} = k] p(y^{\alpha_{1:T}} | \gamma) p(\gamma).$$

- Another useful summary is the posterior probablity that a particular relation is in the induced partial order,

$$p(\pi_{\xi(i,j)} = k | y^{\alpha_{1:T}}) \propto \sum_{\gamma} 1[\pi(\gamma)_{\xi(i,j)} = k] p(y^{\alpha_{1:T}} | \gamma) p(\gamma).$$

- Call the former γ-marginals and the latter π-marginals.

Improvement Function

- Structural information contained in y^α can be further summarized using the entropy of the π-marginals

$$H(y^\alpha) := -\sum_{\xi=1}^{s}\sum_{k=-1}^{1} p(\pi_\xi = k|y^\alpha)\log p(\pi_\xi = k|y^\alpha).$$

- We use H to measure the utility of a new intevention ν

$$H(Y^\nu, y^\alpha) := -\sum_{\xi=1}^{s}\sum_{k=-1}^{1} p(\pi_\xi = k|Y^\nu, y^\alpha)\log p(\pi_\xi = k|Y^\nu, y^\alpha).$$

- Y^ν is unseen, so choose ν based on the improvement function

$$h(\nu) := \mathbb{E}[H(Y^\nu, y^\alpha)].$$

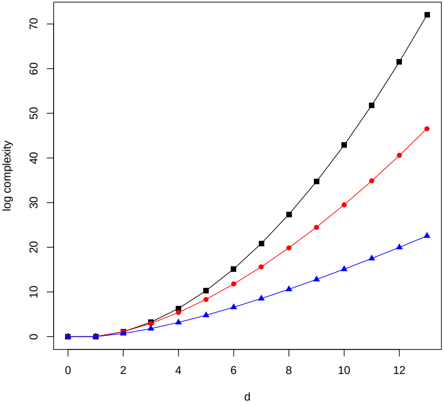
Order Sampling

- MCMC works well in some problems, but in others suffers from slow mixing due to the complexity of DAG space.
- Order-based sampling leads to improve mixing times (Friedman 2003) .
- A related idea generates linear orders using annealed importance sampling (Niinimäki 2012b) .
- An MCMC method for sampling partial orders was proposed in (Niinimäki 2012a).

Bias-Correction

- Sampling from linear orders introduces a bias because the π's and γ's can be consistent with multiple linear orderings.
- Other sampling methods based on linear orders correct this bias by reweighting each sampled γ by $|\mathcal{L}_\gamma|$ (Eaton 2007, Niinimäki 2012b) .
- A subsampling approach is given in (Ellis, 2008) .
- Computing $|\mathcal{L}_\gamma|$ is #P-complete though (Niinimäki 2012a) provide a recursive algorithm generally able to handle $d \approx 40$.

Comparing Complexities



Sampling over the space of linear orders leads to improved mixing because the space of linear orders is small relative to the space of DAGS.

Constructing an Importance Sample

1. Use Metropolis-Hastings to construct a Markov Chain on the space of linear orders with approximate stationary distribution

$$p(L|Y) := \sum_{\gamma \in \Gamma_L} p(\gamma|Y).$$

2. For each linear order L, sample M partial orders from $q_2(\pi|L)$ and then a DAG for each partial order from $q_3(\gamma|\pi)$.
3. Use the sampled γ's to form a Monte Carlo estimate $p(L|Y)$ needed for computing the acceptance probability in the Metropolis algorithm.
4. Collect the γ's from all linear orders and reweight to form a single importance sample for estimating posterior expecations.

Details of the Importance Sample

Input: A random linear order L_0, constants N, M, and data y

0. Estimate $\hat{p}(L_0|y)$ as below.
1. For $n = 1, ..., N$
   1. Sample $L' \sim q_p(L'|L_{n-1})$ by swapping two random positions
   2. For $m = 1, ..., M$ draw $\pi'_m \sim q_2(\pi|L')$ and $\gamma'_m \sim q_1(\gamma|\pi_m)$
   3. Set $w_m = (q_1(\gamma'_m|\pi'_m)q_2(\pi'_m|L'))^{-1}$ and compute $W = \sum_m w_m$
   4. Compute $\hat{p}(L'|Y) \propto \sum_{m=1}^{M} p(Y|\gamma'_m)p(\gamma'_m)w_m/W$
   5. Set $\alpha = 1 \wedge \hat{p}(L'|Y)/\hat{p}(L|Y)$
   6. With probability α set $L_n = L'$ and $\gamma_{n,m} = \gamma'_m$. Otherwise let $L_n = L_{n-1}$ and $(\gamma_{nm})_m = (\gamma_{n-1,m})_m$.
2. Collect $\{\gamma_{nm}\}_{n,m}$ and reweight appropriately.

Details of the Importance Sample

- To use $\{\gamma_{nm}\}_{n,m}$ for estimating posterior expecations we need to compute new importance weights given by the inverse of

$$q(\gamma_{nm}) = \sum_{L} q_1(\gamma_{nm}|\pi_{nm}) q_2(\pi_{nm}|L) p(L).$$

- Here $p(L)$ is the stationary distribution of Markov Chain constructed on the previous slide.
- Having $p(L) \approx p(L|Y)$ improves the efficiency of the sampler, but we don't need $p(L) = p(L|Y)$ for valid estimates.
- For large d the summation above over $d!$ orders is intractable

Estimating the Importance Weights

- The importance weights are found by inverting,

$$\begin{aligned} q(\gamma_{nm}) &= \sum_L q_1(\gamma_{nm}|\pi_{nm}) q_2(\pi_{nm}|L) p(L). \\ &= q_1(\gamma_{nm}|\pi_{nm}) \tilde{q}_2(\pi_{nm}) \sum_L 1[L \in \mathcal{L}_{\pi_{nm}}] p(L). \end{aligned}$$

- The term $\sum_L 1[L \in \mathcal{L}_{\pi_{nm}}] p(L)$ is similar to the bias correction in other methods but in general $\sum_L 1[L \in \mathcal{L}_{\pi_{nm}}] p(L) \neq |\mathcal{L}_{\pi_{nm}}|$
- Crucially we can estimate this term for each π_{nm} using

$$N^{-1} \sum_{i=1}^{N} 1[L_n \in \mathcal{L}_{\pi_{nm}}] \approx \sum_L 1[L \in \mathcal{L}_{\pi_{nm}}] p(L).$$

- Checking whether a linear order is a linear extension of a given partial order is relatively comparatively simple.

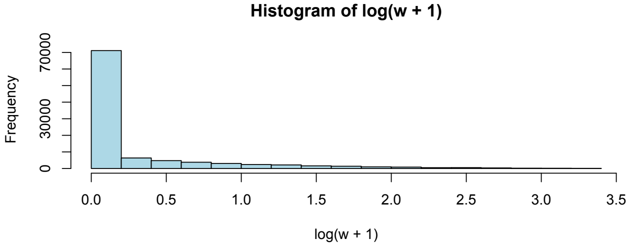
Expected Entropy of Unseen Interventions

- At step $T+1$ we want to use $(y^{\alpha_t}, \alpha_t)_{t=1}^{T}$ to estimate the expected improvement for each potential action $\nu \in \mathcal{A}_T = \{1, ..., d\} \setminus \{\alpha_1, ..., \alpha_T\}$.
- The improvment function $h(\alpha)$ is the expected entropy of the π-marginals,

$$\begin{aligned} h(\alpha) &= \mathbb{E}[H(Y^{\nu}, y^{\alpha_{1:T}})| Y^{\alpha_{1:T}}] \\ &= \mathbb{E}_{\gamma}[\mathbb{E}_{Y^{\nu}|\gamma}[H(Y^{\nu}, y^{\alpha_{1:T}})|\gamma]|y^{\alpha_{1:T}}]. \end{aligned}$$

- The outer expectation over the graphical structure is estimated using a reduced importance sample.
- For each γ in the reduced importance sample, we use a Monte Carlo estimate of the inner expecation.

Reducing the Importance Sample



- Entropy computations are easily parallelized over ν and γ.
- The computational burden may still need to be reduced.
- Importance sample contains many γ's with very small weights.
- Reduce computation time by stratifying the importance sample using weights or likelihoods.
- Devote more resources to top strata and within strata sample with probability proportional to the weights.

Conclusion

- Present a tractable framework for sequentially choosing interventions for structure learning in directed networks.
- Build on order-based sampling to construct an importance sample utilizing the hierarchy among linear order, partial orders, and DAGs.
- ‘Bias-correction’ estimated from sample via linear extension checking does not require counting linear extensions.
- Leverage the importance weights to further reduce the inherent computational burden.

D. Eaton and K. Murphy.
Bayesian structure learning using dynamic programming and MCMC.
UAI, pages 101–108, 2007.

B. Ellis and E. Wong.
Learning causal bayesian network structures from experimental data.
JASA, 103:778–789, 2008.

N. Friedman and D. Koller.
Being bayesian about network structure: A bayesian approach to structure discovery in bayesian networks.
Machine Learning, 50:95–126, 2003.

A. Hauser and P. Bühlmann.
Two optimal strategies for active learning of causal models from interventions.
Proc. of the 6th European Workshop on Probabilistic Graphical Models, pages 123–130, 2012.

EJ Molinelli, A Korkut, W Wang, ML Miller, NP Gauthier, and et al.
Perturbation biology: Inferring signaling networks in cellular systems.
PLoS Comput Biol, 9, 2013.

K. Murphy.
Active learning of causal bayes net structure.
Technical Report, 2001.

- T. Niinimäki and M. Koivisto.
  Annealed Importance Sampling for Structure Learning in Bayesian Networks.
  Proceedings of the Twenty-Third International Joint Conference on Artificial Intelligence, pages 1579–1585, 2012.

- T. Niinimäki, P. Parviainen, and M. Koivisto.
  Partial order mcmc for structure discovery in bayesian networks.
  arXiv:1202.3753, 2012.

- S. Tong and D. Koller.
  Active learning for structure in bayesian networks.
  ICJAI, 2001.