Causal Inference and Machine Learning
Session 7: Bayesian causal discovery and causal effect estimation

Alessandro Mascaro

February 18th, 2025

Barcelona School of Economics,
Master's degree in Data Science Methodology

Bayesian inference and model selection in a few slides

Bayesian inference in one slide

- Suppose we observe a (n, p) data matrix of observations $\boldsymbol{X}$, containing i.i.d. samples of a random vector $X := (X_1, \ldots, X_p)$, and that X follows parametric distribution with unknown parameter Θ;
- In the Bayesian setting, Θ is a random variable and inference on it is done using Bayes theorem to derive the **posterior distribution**

$$p(\Theta \mid \boldsymbol{X}) \propto p(\boldsymbol{X} \mid \Theta)\, p(\Theta)$$

  where $p(\Theta)$ is the **prior distribution**, specified by the statistician;
- Sometimes the posterior has a known form, sometimes it has to approximated, mainly through **sampling methods**;

Bayesian Model Selection in one slide

- Bayesian inference is very flexible and it can be seamlessly used to do **model selection**: one just specifies a prior on the model M_k and its parameters Θ_k and derives the joint posterior distribution:

$$p(M_k, \Theta_k \mid \boldsymbol{X}) \propto p(\boldsymbol{X} \mid \Theta_k, M_k)\, p(\Theta_k \mid M_k)\, p(M_k)$$

- Usually, Bayesian Model Selection (BMS) procedures directly target the marginal posterior distribution of the model:

$$p(M_k \mid \boldsymbol{X}) \propto p(\boldsymbol{X} \mid M_k)\, p(M_k);$$

  where

$$p(\boldsymbol{X} \mid \mathcal{M}_k) = \int p(\boldsymbol{X} \mid \Theta_k, \mathcal{M}_k)\, p(\Theta_k \mid \mathcal{M}_k)\, d\Theta_k$$

  is the **marginal** (i.e. integrated w.r.t. Θ_k) **likelihood** of $\mathcal{M}_k$

Marginal likelihood

- The **marginal likelihood** is a fundamental quantity in BMS;
- You can informally think of it as a score assigned to model M_k;
- The marginal likelihood heavily depends on the specification of the prior $p(\Theta_k \mid \mathcal{M}_k)$, which is chosen by the analyst. As it influences the "score", specific care must be paid in the choice of the parameter prior distributions!
- Typically, $p(\Theta_k \mid \mathcal{M}_k)$ should satisfy some *compatibility* requirements. For instance, we would like two unidentifiable model to be assigned the same marginal likelihood (**score equivalence**)

Bayesian Causal Discovery

- In the Bayesian setting, causal discovery can be tackled as a Bayesian model selection problem, where the models considered are DAG models
- Suppose we have a collection of possible DAGs $(\mathcal{D}_1, \ldots, \mathcal{D}_q)$. If $\mathcal{S}_q$ is the space of "all" DAGs on q nodes/variables, our target is

$$\boxed{p(\mathcal{D} \,|\, \boldsymbol{X}) = \frac{m(\boldsymbol{X} \,|\, \mathcal{D})\, p(\mathcal{D})}{\sum_{\mathcal{D} \in \mathcal{S}_q} m(\boldsymbol{X} \,|\, \mathcal{D})\, p(\mathcal{D})} \propto m(\boldsymbol{X} \,|\, \mathcal{D})\, p(\mathcal{D})}$$

  i.e. the posterior distribution over DAG models

Bayesian Causal Discovery: How to

- The first step is to specify a Bayesian model that will define the posterior distribution. This consists of:
  - $p(X \mid \Theta, \mathcal{D})$: the statistical model;
  - $p(\Theta \mid \mathcal{D})$: the parameter prior;
  - $p(\mathcal{D})$: the model prior;
- Once the model is specified, we can derive the posterior distribution. Usually, this is not calculated exactly, but approximated via **sampling methods**.
  - For example, in our setting as p grows the number of DAGs grows exponentially, and evaluating the posterior probability of each of them becomes infeasible
- We will now tackle both steps, focusing on the Gaussian setting;

Bayesian causal discovery: Model specification in the Gaussian setting

Statistical model - 1

- Suppose X is generated by a linear Gaussian Structural Equation Model (SEM) with independent error components and causal structure represented by the DAG $\mathcal{D}$.
- As we saw in **L1**, this implies that X is distributed as a Gaussian DAG model, i.e.

$$
\begin{aligned}
X_1, \ldots, X_q \,|\, \boldsymbol{\Sigma}_{\mathcal{D}} &\sim \mathcal{N}_p(\mathbf{0}, \boldsymbol{\Sigma}_{\mathcal{D}}) \\
\boldsymbol{\Sigma}_{\mathcal{D}} &= \boldsymbol{L}^{-T}\boldsymbol{D}\boldsymbol{L}^{-1}
\end{aligned}
$$

where $\boldsymbol{L} = (\boldsymbol{I}_p - \boldsymbol{B})$ and $\boldsymbol{B}$ is the matrix of coefficients of the SEM;

coefficients

Statistical model - 2

- The joint pdf of X factorizes as

$$p(x \,|\, (\boldsymbol{L}, \boldsymbol{D}), \mathcal{D}) = \prod_{j=1}^{p} d\,\mathcal{N}(x_j\,; -\boldsymbol{L}^{\top}_{\mathrm{pa}_{\mathcal{D}}(j),j} x_{\mathrm{pa}_j(\mathcal{D})}, \boldsymbol{D}_{jj})$$

- Consequently, we can write the likelihood of the (n, p) data matrix $\boldsymbol{X}$, containing i.i.d. samples from X as

$$p(\boldsymbol{X} \,|\, (\boldsymbol{L}, \boldsymbol{D}), \mathcal{D}) = \prod_{j=1}^{p} d\mathcal{N}_n(\boldsymbol{X}_{.j}\,; -\boldsymbol{X}_{.\mathrm{pa}_j(\mathcal{D})} \boldsymbol{L}_{\mathrm{pa}_j(\mathcal{D}),j}, \boldsymbol{D}_{jj} \boldsymbol{I}_n)$$

Parameter prior - 1

- Ben-David et al. (2011) developed the **DAG-Wishart** distribution, which is defined on the space of matrices $(\boldsymbol{L}, \boldsymbol{D})$ of a DAG.

  $\Longrightarrow$ It is a natural candidate for our parameter prior specification problem!

- The DAG-Wishart distribution is parameterized by a shape parameter $\boldsymbol{\alpha}(\mathcal{D}) := (\alpha_1(\mathcal{D}), \dots, \alpha_p(\mathcal{D}))$ and a rate hyperparameter $\boldsymbol{U}$, a (p, p) s.p.d. matrix;

- In general terms, we thus specify the parameter prior as:

$$(\boldsymbol{L}, \boldsymbol{D}) \mid \mathcal{D} \sim \text{DAG-Wishart}(\boldsymbol{\alpha}(\mathcal{D}), \boldsymbol{U})$$

The DAG-Wishart prior

- The pdf of a DAG-Wishart distribution has the following form:

$$p(\boldsymbol{L}, \boldsymbol{D} \mid \mathcal{D}) = \frac{1}{\mathcal{Z}_{\mathcal{D}}(\boldsymbol{U}, \boldsymbol{\alpha}(\mathcal{D}))} \exp\left\{-\frac{1}{2}\operatorname{tr}\left(\left(\boldsymbol{L}\boldsymbol{D}^{-1}\boldsymbol{L}^T\right)\boldsymbol{U}\right)\right\} \cdot \prod_{j=1}^{p} \boldsymbol{D}_{jj}^{-\frac{\alpha_j(\mathcal{D})}{2}}$$

- The normalizing constant $\mathcal{Z}_{\mathcal{D}}(\boldsymbol{U}, \boldsymbol{\alpha}(\mathcal{D}))$ is

$$\mathcal{Z}_{\mathcal{D}}(\boldsymbol{U}, \boldsymbol{\alpha}) = \prod_{j=1}^{p} 2^{\left(\frac{\alpha_j(\mathcal{D})-2}{2}\right)} \pi^{\frac{|pa_j(\mathcal{D})|}{2}} \Gamma\left(\frac{\alpha_j(\mathcal{D}) - |\mathrm{pa}_j(\mathcal{D})| - 2}{2}\right) \cdot \frac{\left|\boldsymbol{U}_{\mathrm{pa}_j(\mathcal{D})}\right|^{\frac{\alpha_j(\mathcal{D})-\mathrm{pa}_j(\mathcal{D})-3}{2}}}{\left|\boldsymbol{U}_{\mathrm{fa}_j(\mathcal{D})}\right|^{\frac{\alpha_j(\mathcal{D})-\mathrm{pa}_j(\mathcal{D})-2}{2}}}$$

The DAG-Wishart prior

- The DAG-Wishart distribution presents many useful features
- First, it is **conjugate** to the likelihood of a Gaussian DAG model, which means that, if $\boldsymbol{X}$ contains i.i.d. samples from a Gaussian DAG Model $\mathcal{D}$, we have

$$(\boldsymbol{L}, \boldsymbol{D}) \mid \boldsymbol{X}, \mathcal{D} \sim \text{DAG-Wishart}\Big(\boldsymbol{\alpha}(\mathcal{D}) + n, \boldsymbol{U} + \boldsymbol{X}^T\boldsymbol{X}\Big)$$

  i.e., the posterior distribution has the same form as the prior distribution;
- Moreover, its normalizing constant, although involved, is available in closed form, which allows the calculation of the marginal likelihood;

The DAG-Wishart prior

- Finally, the DAG-Wishart distribution implies a set of local distributions on the non-null elements of $(\boldsymbol{L}, \boldsymbol{D})$ that are node-wise independent. In other words, it holds that

$$p(\boldsymbol{L}, \boldsymbol{D} \,|\, \mathcal{D}) = \prod_{j=1}^{p} p(\boldsymbol{L}_{\mathrm{pa}_j(\mathcal{D}),j} \,|\, \boldsymbol{D}_{jj}, \mathcal{D}) p(\boldsymbol{D}_{jj} \,|\, \mathcal{D})$$

- In particular, we have that:

$$\boldsymbol{D}_{jj} \mid \mathcal{D} \sim \text{Inv-Ga}\left(\frac{a_j(\mathcal{D}) - |pa_j(\mathcal{D})|}{2} - 1, \frac{1}{2}\boldsymbol{U}_{j|\mathrm{pa}_j(\mathcal{D})}\right),$$

$$\boldsymbol{L}_{\mathrm{pa}_j(\mathcal{D}),j} \mid \boldsymbol{D}_{jj}, \mathcal{D} \sim \mathcal{N}_{|\mathrm{pa}_j(\mathcal{D})|}\left(-\boldsymbol{U}^{-1}_{\mathrm{pa}_j(\mathcal{D})}\boldsymbol{U}_{\mathrm{pa}_j(\mathcal{D}),j}, \boldsymbol{D}_{jj}\boldsymbol{U}^{-1}_{\mathrm{pa}_j(\mathcal{D})}\right),$$

where $\boldsymbol{U}_{j|\mathrm{pa}_j(\mathcal{D})} := \boldsymbol{U}_{jj} - \boldsymbol{U}_{j,\mathrm{pa}_j(\mathcal{D}}(\boldsymbol{U}_{\mathrm{pa}_j(\mathcal{D}),\mathrm{pa}_j(\mathcal{D})})^{-1}\boldsymbol{U}_{\mathrm{pa}_j(\mathcal{D}),j}$

- Nice consequence: sampling from the posterior of $(\boldsymbol{L}, \boldsymbol{D})$ is easy!

The DAG-Wishart prior

- The most important effect of this last property of the DAG-Wishart distribution emerges when computing the marginal likelihood of a DAG $\mathcal{D}$:

$$
\begin{aligned}
m(\boldsymbol{X} \mid \mathcal{D}) &= \int p(\boldsymbol{X} | (\boldsymbol{L}, \boldsymbol{D}), \mathcal{D}) p(\boldsymbol{L}, \boldsymbol{D} \mid \mathcal{D}) \, d(\boldsymbol{L}, \boldsymbol{D}) \\
&= \int \prod_{j=1}^{p} p(\boldsymbol{X}_{.j} \mid \boldsymbol{X}_{.\mathrm{pa}_j(\mathcal{D})}, \boldsymbol{L}_{\mathrm{pa}_j(\mathcal{D}),j}, \boldsymbol{D}_{jj}, \mathcal{D}) \\
&\qquad p(\boldsymbol{L}_{\mathrm{pa}_j(\mathcal{D}),j} \mid \boldsymbol{D}_{jj}, \mathcal{D}) p(\boldsymbol{D}_{jj} \mid \mathcal{D}) \, d(\boldsymbol{L}, \boldsymbol{D}) \\
&= \prod_{j=1}^{p} \int p(\boldsymbol{X}_{.j} \mid \boldsymbol{X}_{.\mathrm{pa}_j(\mathcal{D})}, \boldsymbol{L}_{\mathrm{pa}_j(\mathcal{D}),j}, \boldsymbol{D}_{jj}, \mathcal{D}) \\
&\qquad p(\boldsymbol{L}_{\mathrm{pa}_j(\mathcal{D}),j} \mid \boldsymbol{D}_{jj}, \mathcal{D}) p(\boldsymbol{D}_{jj} \mid \mathcal{D}) \, d(\boldsymbol{L}_{\mathrm{pa}_j(\mathcal{D})}, \boldsymbol{D}_{jj}) \\
&= \prod_{j=1}^{p} m(\boldsymbol{X}_{.j} \mid \boldsymbol{X}_{.\mathrm{pa}_j(\mathcal{D})}, \mathcal{D})
\end{aligned}
$$

The DAG-Wishart prior

- Using a DAG-Wishart prior, the marginal likelihood follows the same factorization that the Gaussian DAG model implies on the likelihood;
- In this case, we say that the marginal likelihood is decomposable, a property that will turn out to be very useful when designing sampling algorithms to sample from the posterior over the DAG space!
- Each element of that factorisation can be easily computed from the normalising constant of the DAG-Wishart distribution. Denoting with $\tilde{\boldsymbol{U}} = \boldsymbol{U} + \boldsymbol{X}^T \boldsymbol{X}$, and with $\tilde{a}_j = a_j + n$ we have

$$m\left(\boldsymbol{X}_j \mid \boldsymbol{X}_{\mathrm{pa}(j)}, \mathcal{D}\right) = (2\pi)^{-\frac{n}{2}} \cdot \frac{\left|\boldsymbol{U}_{\mathrm{pa}_j, \mathrm{pa}_j}\right|^{\frac{1}{2}}}{\left|\widetilde{\boldsymbol{U}}_{\mathrm{pa}_j, \mathrm{pa}_j}\right|^{\frac{1}{2}}} \cdot \frac{\Gamma\left(\frac{1}{2}\widetilde{a}_j\right)}{\Gamma\left(\frac{1}{2}a_j\right)} \cdot \frac{\left(\frac{1}{2}\boldsymbol{U}_{j \mid \mathrm{pa}_j}\right)^{\frac{1}{2}a_j}}{\left(\frac{1}{2}\widetilde{\boldsymbol{U}}_{j \mid \mathrm{pa}_j}\right)^{\frac{1}{2}\widetilde{a}_j}}$$

Score equivalence

- As we discussed at the beginning, the parameter prior we specify influences the marginal likelihood, i.e. the "score" we assign to each model;
- In the Gaussian setting, two Markov equivalent DAGs are unidentifiable given data alone;
- However, their marginal likelihood can be different because of the prior information we included via our prior distribution on the parameters, which is undesirable;
- Peluso & Consonni (2020) showed that we can ensure score equivalence with DAG-Wishart prior by setting, for each $j \in [q]$:

$$a_j(\mathcal{D}) = a - p + 2|\mathrm{pa}_j(\mathcal{D})| + 3$$

where $a > p - 1$ ensures that the prior is proper;

DAG prior

- $p(\mathcal{D})$ can be assigned through a collection of Bernoulli distributions on 0-1 elements indicating absence/presence of edges in DAG $\mathcal{D}$
- Let $\boldsymbol{S}^{\mathcal{D}}$ be the 0-1 *adjacency matrix* of the skeleton of $\mathcal{D}$:

$$\boldsymbol{S}^{\mathcal{D}}_{u,v} = \begin{cases} 1 & \text{if } u \rightarrow v \in \mathcal{D} \text{ or } u \leftarrow v \in \mathcal{D} \\ 0 & \text{otherwise} \end{cases}$$

- We assign a prior on the DAGs based on their skeleton. In particular: $\boxed{\boldsymbol{S}^{\mathcal{D}}_{u,v} \,|\, \pi \overset{\text{iid}}{\sim} \text{Ber}(\pi)} \quad u < v, \pi \in (0,1)$
- The prior probability assigned to each DAG is thus:

$$p(\mathcal{D} \,|\, \pi) = \pi^{|\boldsymbol{S}^{\mathcal{D}}|} (1-\pi)^{\frac{q(q-1)}{2} - |\boldsymbol{S}^{\mathcal{D}}|}$$

Model specified

- We have now specified the whole Bayesian model for our Bayesian causal discovery problem!
- We just need to compute the posterior distribution;

Bayesian causal discovery: Sampling from the posterior distribution

Sampling from the posterior

- The denominator of $p(\mathcal{D} \mid \boldsymbol{X})$ involves a sum over a finite, but *very large*, number of DAGs $\implies$ we cannot compute the posterior in exact form:
- The only thing we need to can do is to approximate it using, for example, Markov Chain Monte Carlo (MCMC) sampling schemes;
- In particular, we will use a **Metropolis-Hastings** algorithm, which is based on the following steps
  - Start from an (arbitrary) initial DAG $\mathcal{D}^{(0)}$
  - Given a current DAG $\mathcal{D}$ propose a new candidate DAG $\widetilde{\mathcal{D}}$
  - Accept/reject $\widetilde{\mathcal{D}}$ with "some" probability
  - Iterate the previous steps for a number of times S

Proposing a new DAG

- Suppose $\mathcal{D}$ is the current DAG. We **propose** a new candidate DAG by **inserting**, **deleting** or **reversing** at random an edge in $\mathcal{D}$ and **checking that the resulting graph is a DAG**!
- In practice, we build the set $\mathcal{O}_{\mathcal{D}}$ of all possible DAGs that can be reached from $\mathcal{D}$ in one of the three moves above and propose uniformly one DAG among them
- The probability of transition from $\mathcal{D}$ to $\widetilde{\mathcal{D}}$ is then

$$q(\widetilde{\mathcal{D}} \,|\, \mathcal{D}) = \frac{1}{|\mathcal{O}_{\mathcal{D}}|}$$

with$|\mathcal{O}_{\mathcal{D}}|$ number of DAGs obtained from $\mathcal{D}$ by one of the local moves above, i.e. the number of *direct successors* DAGs of $\mathcal{D}$;

Acceptance/Rejection step

- Given a current DAG $\mathcal{D}$, a new DAG $\widetilde{\mathcal{D}}$ drawn from the proposal $q(\widetilde{\mathcal{D}} \,|\, \mathcal{D})$ is accepted with probability

$$\alpha_{\widetilde{\mathcal{D}},\mathcal{D}} = \min\left\{1; \frac{m(\boldsymbol{X} \,|\, \widetilde{\mathcal{D}})}{m(\boldsymbol{X} \,|\, \mathcal{D})} \cdot \frac{p(\widetilde{\mathcal{D}})}{p(\mathcal{D})} \cdot \frac{q(\mathcal{D} \,|\, \widetilde{\mathcal{D}})}{q(\widetilde{\mathcal{D}} \,|\, \mathcal{D})}\right\}$$

  which depends on:

  - The marginal likelihood ratio;
  - The prior ratio;
  - The proposal ratio;

Acceptance/Rejection step

- The **proposal ratio**
$$\frac{q(\mathcal{D}\,|\,\widetilde{\mathcal{D}})}{q(\widetilde{\mathcal{D}}\,|\,\mathcal{D})} = \frac{|\mathcal{O}_{\mathcal{D}}|}{|\mathcal{O}_{\widetilde{\mathcal{D}}}|}$$
requires the enumeration of all operators that can be applied to $\mathcal{D}$ and lead to a valid graph (i.e. a DAG).
- It is usually computationally expensive, but for p large it can be approximated to 1;

Acceptance/Rejection step

- As we are using local moves and thanks to the decomposability of the marginal likelihood, the **marginal likelihood** ratio simplifies to the components which are affected by the local move.
- If, for instance, $\tilde{\mathcal{D}}$ differs from $\mathcal{D}$ for the addition of an edge pointing towards node t, we have:

$$\frac{m(\boldsymbol{X} \mid \widetilde{\mathcal{D}})}{m(\boldsymbol{X} \mid \mathcal{D})} = \frac{m(\boldsymbol{X}_{.t} \mid \boldsymbol{X}_{.\mathrm{pa}_t(\widetilde{\mathcal{D}})}\widetilde{\mathcal{D}})}{m(\boldsymbol{X}_{.t} \mid \boldsymbol{X}_{.\mathrm{pa}_t(\mathcal{D})}\mathcal{D})},$$

which significantly speeds up computations!

Algorithm

Algorithm 1: Collapsed MCMC to sample from $p(\mathcal{D} \mid \boldsymbol{X})$

Input: $\boldsymbol{X}$ (n, q) dataset; S number of MCMC iterations; prior hyperparameters

Output: S samples from the posterior $p(\mathcal{D} \mid \boldsymbol{X})$

```
Initialize D^(0), e.g. the empty DAG;
for s = 1, ..., S do
    Sample D~ from q(D~ | D^(s-1));
    Compute the acceptance probability α_{D~,D};
    Set D^(s) = D~ with probability α_{D~,D}, otherwise D^(s) = D^(s-1);
end
return {D^(1), ..., D^(S)}
```

- The resulting algorithm is a *collapsed* sampler over the space of DAGs, since we integrated out the parameter;
- Its output is a collection of DAGs approximately drawn from the posterior $p(\mathcal{D} \mid \boldsymbol{X})$

Posterior inference

- Output of the MH algorithm is a collection of DAGs $\left\{\mathcal{D}^{(1)}, \ldots, \mathcal{D}^{(S)}\right\}$
- We can provide an estimate of the posterior probability of $\mathcal{D} \in \mathcal{S}_q$ as

$$\boxed{\widehat{p}(\mathcal{D} \,|\, \boldsymbol{X}) = \frac{1}{S} \sum_{s=1}^{S} \mathbf{1}\left\{\mathcal{D}^{(s)} = \mathcal{D}\right\}}$$

  i.e. the proportion of DAGs, visited by the MCMC, equal to $\mathcal{D}$
- Other summaries:
    - Estimate of the (marginal) posterior probability of edge inclusion for each $u \to v$

$$\boxed{\widehat{p}(u \to v \,|\, \boldsymbol{X}) = \frac{1}{S} \sum_{s=1}^{S} \mathbf{1}\left\{u \to v \in \mathcal{D}^{(s)}\right\}}$$

    computed as the proportion of DAGs, visited by the MCMC, containing $u \to v$;

Posterior inference

- DAG point estimates from the posterior over DAGs can be obtained by:
  - including those edges whose posterior probability is higher than some threshold, e.g. 0.5 (Median Probability DAG Model, MPM)

$$\widehat{\boldsymbol{S}}_{u,v} = \begin{cases} 1 & \text{if } \ \widehat{p}(u \to v \mid \boldsymbol{X}) \geq 0.5 \\ 0 & \text{otherwise} \end{cases}$$

  - selecting the DAG having the highest posterior probability (Maximum A Posteriori DAG, MAP)

$$\widehat{\mathcal{D}}_{MAP} = \underset{\mathcal{D}}{\text{argmax}}\, \widehat{p}(\mathcal{D} \mid \boldsymbol{X})$$

Software

- Algorithm 1, specialized to Gaussian DAGs, is implemented in the `R` package `BCDAG` (Bayesian structure and Causal learning of Gaussian DAGs) within the function `learnDAG` and under the setting `collapse = TRUE`
- Inputs of the function are:
  - `data` : the (n, q) data matrix $\boldsymbol{X}$
  - `S` : the number of MCMC iterations
  - `burn` : a burn-in period
  - `a,U` : hyperparameters of the DAG-Wishart prior
  - `w` : prior probability of edge inclusion for $p(\mathcal{D})$
- Function for posterior summaries and MCMC diagnostics are also provided within the package. See Castelletti & Mascaro (2022) for full details

Bayesian estimation of causal effects

Estimating causal effects for fixed DAG

- As we saw yesterday discussing Maximum Likelihood Estimation, a causal effect can be identified and estimated directly from $\mathbf{\Sigma}^{\mathcal{D}}$ as

$$\gamma_{ty} = [(\mathbf{\Sigma}^{\mathcal{D}}_{\tilde{Z},\tilde{Z}})^{-1}(\mathbf{\Sigma}^{\mathcal{D}}_{\tilde{Z},y})]_1$$

  where $\tilde{\boldsymbol{Z}} := (X_t, \boldsymbol{Z})$ and $\boldsymbol{Z}$ is an adjustment set;
- In the Bayesian setting, the "estimate" of $\mathbf{\Sigma}^{\mathcal{D}}$ is its posterior distribution $p(\mathbf{\Sigma}^{\mathcal{D}} \mid \boldsymbol{X}, \mathcal{D})$;
- In the same way, the "estimate" of the causal effect will be a posterior distribution $p(\gamma_{ty} \mid \boldsymbol{X})$
- We cannot compute these posteriors exactly, but again we can approximate them via sampling!

Estimating causal effects for unknown DAG

- As both γ_{ty} and $\boldsymbol{\Sigma}_{\mathcal{D}}$ are functions of the DAG parameters $(\boldsymbol{L}, \boldsymbol{D})$, we can first try to approximate the joint posterior distribution

$$p(\boldsymbol{L}, \boldsymbol{D}, \mathcal{D} \,|\, \boldsymbol{X}) = p(\boldsymbol{L}, \boldsymbol{D} \,|\, \mathcal{D}, \boldsymbol{X}) \cdot p(\mathcal{D} \,|\, \boldsymbol{X});$$

- As we saw before, $p(\boldsymbol{L}, \boldsymbol{D} \,|\, \mathcal{D}, \boldsymbol{X})$ is just a DAG-Wishart and it can be easily sampled from once the DAG is known, and we can sample from $p(\mathcal{D} \,|\, \boldsymbol{X})$ using the same MH as before! Hence:

Algorithm 2: MCMC to sample from $p(\boldsymbol{\theta}, \mathcal{D} \,|\, \boldsymbol{X})$

Input: $\boldsymbol{X}$ (n, q) dataset; S number of MCMC iterations; prior hyperparameters

Output: S samples from the posterior $p(\boldsymbol{\theta}, \mathcal{D} \,|\, \boldsymbol{X})$

```
Initialize D^(0), e.g. the empty DAG;
for s = 1, ..., S do
    Sample D~ from q(D~ | D^(s-1));
    Compute the acceptance probability α_{D~,D};
    Set D^(s) = D~ with probability α_{D~,D}, otherwise D^(s) = D^(s-1);
    Sample θ^(s) from its full conditional;
end
return {(θ^(1), D^(1)), ..., (θ^(S), D^(S))}
```

Posterior sampler for DAGs and parameters

- Output of Algorithm 2 is a collection of draws from the posterior $p(\boldsymbol{L}, \boldsymbol{D}, \mathcal{D} \mid \boldsymbol{X})$ of the form

$$\left\{ \left(\boldsymbol{L}^{(1)}, \boldsymbol{D}^{(1)}, \mathcal{D}^{(1)} \right), \ldots, \left(\boldsymbol{L}^{(S)}, \boldsymbol{D}^{(S)}, \mathcal{D}^{(S)} \right) \right\}$$

- We are interested in the causal effect γ_{ty}
- We can obtain a sample from it by, for each $s \in [S]$:
  - Calculating $\boldsymbol{\Sigma}_{\mathcal{D}}^{(s)} = (\boldsymbol{L}^{(S)})^{-T} \boldsymbol{D}^{(S)} (\boldsymbol{L}^{(S)})^{-1}$;
  - From $\boldsymbol{\Sigma}_{\mathcal{D}}^{(s)}$ deriving $\gamma_{ty}^{(s)}$ using the adjustment formula above;
- The collection $\left\{ \gamma_{ty}^{(1)}, \ldots, \gamma_{ty}^{(S)} \right\}$ provides an approximation of $p(\gamma_{ty} \mid \boldsymbol{X})$ which naturally accounts for DAG-model uncertainty, since each draw potentially depends on a different underlying DAG;

Posterior inference on causal effects

- A point estimate of γ_{ty} is then

$$\boxed{\widehat{\gamma}_{ty}^{BMA} = \frac{1}{S}\sum_{s=1}^{S}\gamma_{ty}^{(s)}},$$

  which implicitly performs **Bayesian Model Averaging** (BMA) through the MCMC frequencies of the visited DAGs;

- Other summaries/queries of interest are:
  - $\widehat{p}\big(\gamma_{ty} < 0 \,|\, \boldsymbol{X}\big) = \frac{1}{S}\sum_{s=1}^{S}\mathbf{1}\Big\{\gamma_{ty}^{(s)} < 0\Big\}$;
  - $\widehat{p}\big(\gamma_{ty} = 0 \,|\, \boldsymbol{X}\big) = \frac{1}{S}\sum_{s=1}^{S}\mathbf{1}\Big\{\gamma_{ty}^{(s)} = 0\Big\}$;
  - $\widehat{p}\big(\gamma_{ty} > 0 \,|\, \boldsymbol{X}\big) = \frac{1}{S}\sum_{s=1}^{S}\mathbf{1}\Big\{\gamma_{ty}^{(s)} > 0\Big\}$

Software

- Algorithm 2 is also implemented in `BCDAG` within the function `learn_DAG` and under the setting `collapse = FALSE`;
- Inputs of the function are the same as of Algorithm 1:
    - `data` : the (n, q) data matrix $\boldsymbol{X}$;
    - `S` : the number of MCMC iterations;
    - `burn` : a burn-in period;
    - `a,U` : hyperparameters of the DAG-Wishart prior;
    - `w` : prior probability of edge inclusion for $p(\mathcal{D})$;

Software

- Function `get_causaleffect` recovers the posterior of causal effect parameters of interest from the output of `learnDAG`;
- Inputs of `get_causaleffect` are:
  - `learnDAG_output` : output of `learnDAG`, an object of class `bcdag`;
  - `targets` : numerical label of variable X_v (target);
  - `response` : numerical label of variable Y (response)
- Functions for MCMC diagnostics and summaries are also available (see `get_diagnostics` function);

Thank you!

References:

- Ben-David, E., Li, T., Massam, H., Rajaratnam, B. (2011). *High dimensional Bayesian inference for Gaussian directed acyclic graph models*. arXiv preprint arXiv:1109.4371.
- Castelletti, F., & Consonni, G. (2021). *Bayesian inference of causal effects from observational data in Gaussian graphical models*. Biometrics, 77(1), 136-149.
- Castelletti, F., & Mascaro, A. (2022). *BCDAG: An R package for Bayesian structure and causal learning of Gaussian DAGs*. arXiv preprint arXiv:2201.12003.
- Peluso, S., & Consonni, G. (2020). *Compatible priors for model selection of high-dimensional Gaussian DAGs*. Electronic Journal of Statistics, 14(2), 4110-4132.