Mapping Complex Data with Bayesian Networks

Marco Scutari
scutari@idsia.ch

Dalle Molle Institute for
Artificial Intelligence (IDSIA)

May 21, 2021

→ Bayesian Networks

Incomplete Data

Dynamic Networks

Related Data Sets

A Graph and a Probability Distribution

A Bayesian network (BN) [10] is defined by:

- a network structure, a directed acyclic graph $\mathcal{G}$ in which each node corresponds to a random variable X_i;
- a global probability distribution $\mathbf{X}$ with parameters Θ, which can be factorised into smaller local probability distributions according to the arcs present in $\mathcal{G}$.

The main role of the network structure is to express the conditional independence relationships among the variables in the model through graphical separation, thus specifying the factorisation of the global distribution:

$$\mathrm{P}(\mathbf{X}) = \prod_{i=1}^{N} \mathrm{P}(X_i \mid \Pi_{X_i}; \Theta_{X_i}) \quad \text{where} \quad \Pi_{X_i} = \{\text{parents of } X_i \text{ in } \mathcal{G}\}\,.$$

Learning a BN $\mathcal{B} = (\mathcal{G}, \Theta)$ from a data set $\mathcal{D}$ involves two steps:

$$\underbrace{\mathrm{P}(\mathcal{B} \mid \mathcal{D}) = \mathrm{P}(\mathcal{G}, \Theta \mid \mathcal{D})}_{\text{learning}} = \underbrace{\mathrm{P}(\mathcal{G} \mid \mathcal{D})}_{\text{structure learning}} \cdot \underbrace{\mathrm{P}(\Theta \mid \mathcal{G}, \mathcal{D})}_{\text{parameter learning}}.$$

Structure learning consists in finding the DAG with the best

$$\mathrm{P}(\mathcal{G} \mid \mathcal{D}) \propto \underbrace{\mathrm{P}(\mathcal{G})}_{\text{graph prior}} \cdot \underbrace{\mathrm{P}(\mathcal{D} \mid \mathcal{G})}_{\text{marginal likelihood}} = \mathrm{P}(\mathcal{G}) \int \mathrm{P}(\mathcal{D} \mid \mathcal{G}, \Theta)\, \mathrm{P}(\Theta \mid \mathcal{G})\, d\Theta$$

which is known as score-based learning [9]. As an alternative, constraint-based learning uses tests to assess conditional independence relationships following Pearl's work on causal networks [15]:

$$\underbrace{X_i \perp\!\!\!\perp_P X_j \mid \mathbf{S}_{X_i, X_j}}_{\text{conditional independence}} \Longrightarrow \underbrace{X_i \perp\!\!\!\perp_G X_j \mid \mathbf{S}_{X_i, X_j}}_{\text{graphical separation}}.$$

Parameter learning then consists in estimating the parameters of the local distributions $X_i \mid \Pi_{X_i}$.

The Classic Definition and Modern Extensions

What are we assuming when trying to learn a BN? Typically that:

- observations are independent and there are no missing values;
- all variables are observed, that is, there are no latent variables introducing confounding in the model;
- we measure probabilistic associations (or rather, independencies) and we cannot necessarily interpret them as causal.

What happens if we relax these assumptions? Many extensions suddenly become possible, see [14] for a recent review. In this talk we will discuss:

- Learning BNs from incomplete data with Structural EM [13] and the node-averaged likelihood [3].
- Learning BNs from continuous-time dynamic data [5].
- Learning BNs from heterogeneous data that are the collation of multiple, related data sets [1].

- ✔ Bayesian Networks
- → Incomplete Data
- Dynamic Networks
- Related Data Sets

Learning the structure of a BN from incomplete data is computationally unfeasible because we need to perform a joint optimisation over the missing values and the parameters to score each candidate network. The maximum a posteriori DAG maximises

$$\begin{aligned} \mathrm{P}\left(\mathcal{D} \mid \mathcal{G}\right) &= \int \mathrm{P}\left(\mathcal{D}^{O}, \mathcal{D}^{M} \mid \mathcal{G}, \Theta\right) \mathrm{P}\left(\Theta \mid \mathcal{G}\right)\, d\Theta \\ &= \int \underbrace{\mathrm{P}\left(\mathcal{D}^{M} \mid \mathcal{D}^{O}, \mathcal{G}, \Theta\right)}_{\text{missing data}} \underbrace{\mathrm{P}\left(\mathcal{D}^{O} \mid \mathcal{G}, \Theta\right)}_{\text{observed data}} \underbrace{\mathrm{P}\left(\Theta \mid \mathcal{G}\right)\, d\Theta}_{\text{averaging over parameters}} . \end{aligned}$$

A full Bayesian approach would require averaging over all the possible configurations of the missing data, leading to

$$\mathrm{P}\left(\mathcal{D} \mid \mathcal{G}\right) = \iint \mathrm{P}\left(\mathcal{D}^{M} \mid \mathcal{D}^{O}, \mathcal{G}, \Theta\right) \mathrm{P}\left(\mathcal{D}^{O} \mid \mathcal{G}, \Theta\right) \mathrm{P}\left(\Theta \mid \mathcal{G}\right)\, d\Theta\, d\mathcal{D}^{M}.$$

which has one one extra dimension for each missing value. An additional problem is that $\mathrm{P}(\mathcal{D}^{M} \mid \mathcal{D}^{O}, \mathcal{G}, \Theta)$ does not factorise in the general case.

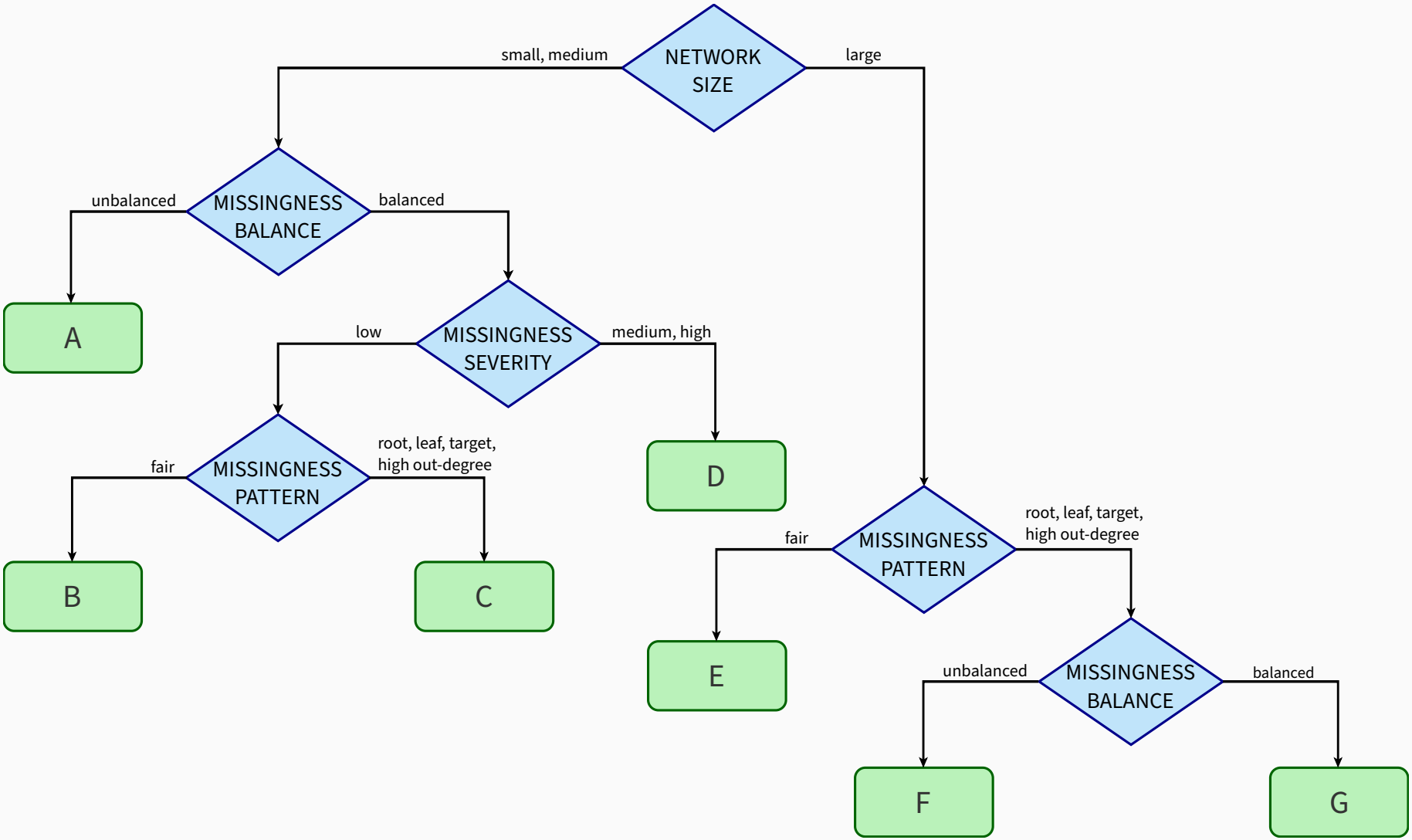
Structure Learning from Incomplete Data: Structural EM

The Structural Expectation-Maximisation (EM) algorithm [7] makes structure learning computationally feasible by searching for the best structure inside of EM instead of embedding EM inside a structure learning algorithm. It consists of two steps like the classic EM:

- in the E-step, we complete the data by computing the expected sufficient statistics using the current network structure;
- in the M-step, we find the structure that maximises the expected score function for the completed data.

Since the scoring in the M-step uses the completed data , structure learning can be implemented using standard algorithms. The original proposal by Friedman [7] used BIC and greedy search; and he [8] later extended SEM to a fully Bayesian approach based posterior scores, and proved the convergence of the resulting algorithm.

The Structural EM Is Hard To Tune



Even just choosing which EM (hard EM or soft EM [11]) approach to use makes for a complicated decision tree [13].

The Node-Averaged Likelihood

Balov [2] proposed a more scalable approach for discrete BNs called Node-Average Likelihood (NAL). NAL computes each term using the locally-complete data $\mathcal{D}_{(i)} \subseteq \mathcal{D}$ for which X_i, Π_{X_i} are observed:

$$\bar{\ell}(X_i \mid \Pi_{X_i}, \widehat{\Theta}_{X_i}) = \frac{1}{|\mathcal{D}_{(i)}|} \sum_{\mathcal{D}_{(i)}} \log \mathrm{P}(X_i \mid \Pi_{X_i}, \widehat{\Theta}_{X_i}) \to \mathrm{E}\left[\ell(X_i \mid \Pi_{X_i})\right],$$

which he used to define

$$S_{\mathrm{PL}}(\mathcal{G} \mid \mathcal{D}) = \bar{\ell}(\mathcal{G}, \Theta \mid \mathcal{D}) - \lambda_n h(\mathcal{G}), \qquad \lambda_n \in \mathbb{R}^+, h : \mathbb{G} \to \mathbb{R}^+$$

and structure learning as $\widehat{\mathcal{G}} = \mathrm{argmax}_{\mathcal{G} \in \mathbb{G}} \, S_{\mathrm{PL}}(\mathcal{G} \mid \mathcal{D})$.

We [3] proved both identifiability and consistency of structure learning when using $S_{\mathrm{PL}}(\mathcal{G} \mid \mathcal{D})$ for conditional Gaussian BNs, which include discrete and Gaussian BNs as special cases.

AIC, BIC, NAL: Which of Them Are Consistent?

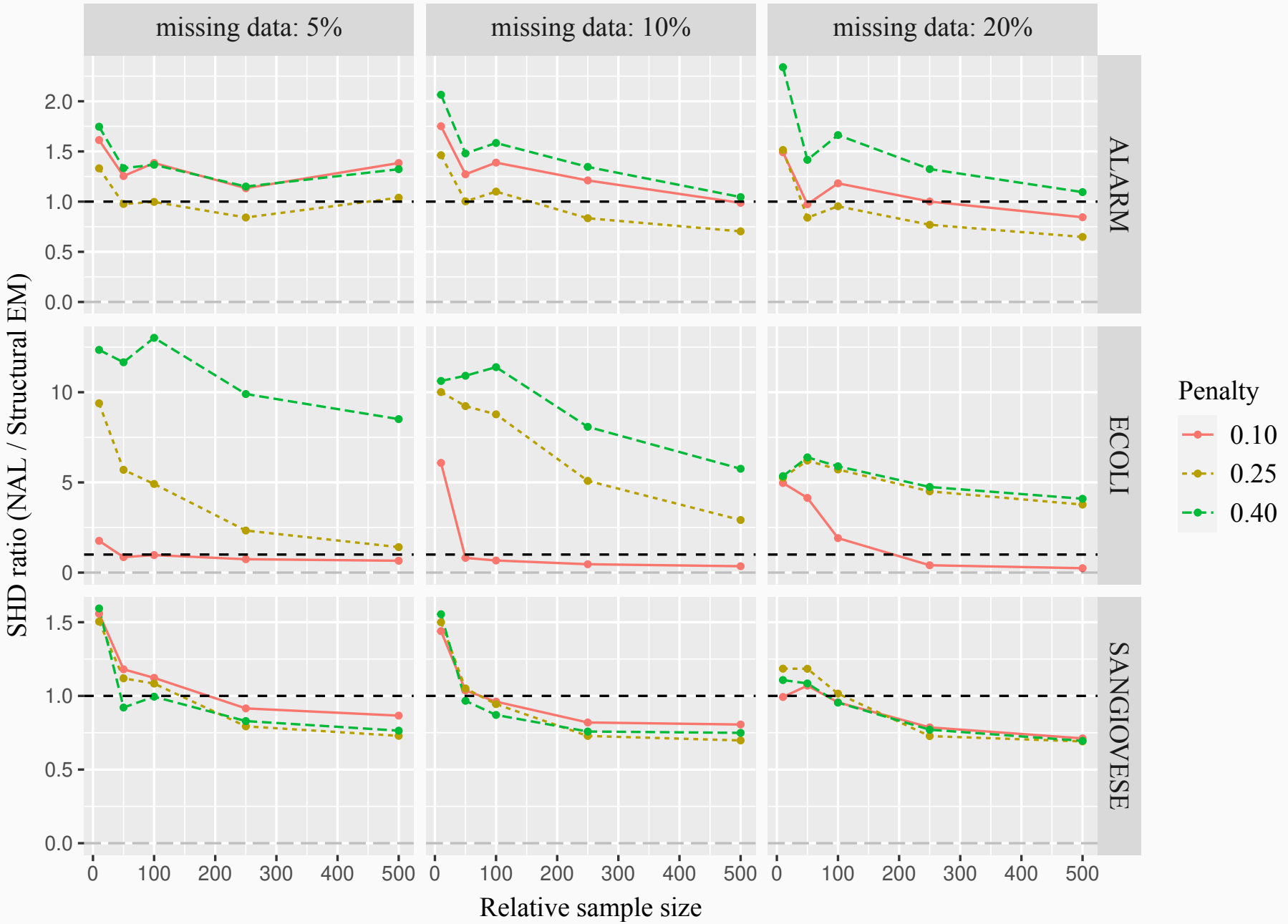
Let $\mathcal{G}_0$ be identifiable, $\lambda_n \to 0$ as $n \to \infty$, and assume MLEs and NAL's Hessian exist finite. Then as $n \to \infty$:

1. If $n\lambda_n \to \infty$, $\widehat{\mathcal{G}}$ is consistent.
2. Under MCAR and $\mathrm{VAR}(\mathrm{NAL}) < \infty$, if $\sqrt{n}\lambda_n \to \infty$, $\widehat{\mathcal{G}}$ is consistent.
3. Under the above, if $\liminf_{n\to\infty} \sqrt{n}\lambda_n < \infty$, then $\widehat{\mathcal{G}}$ is not consistent.

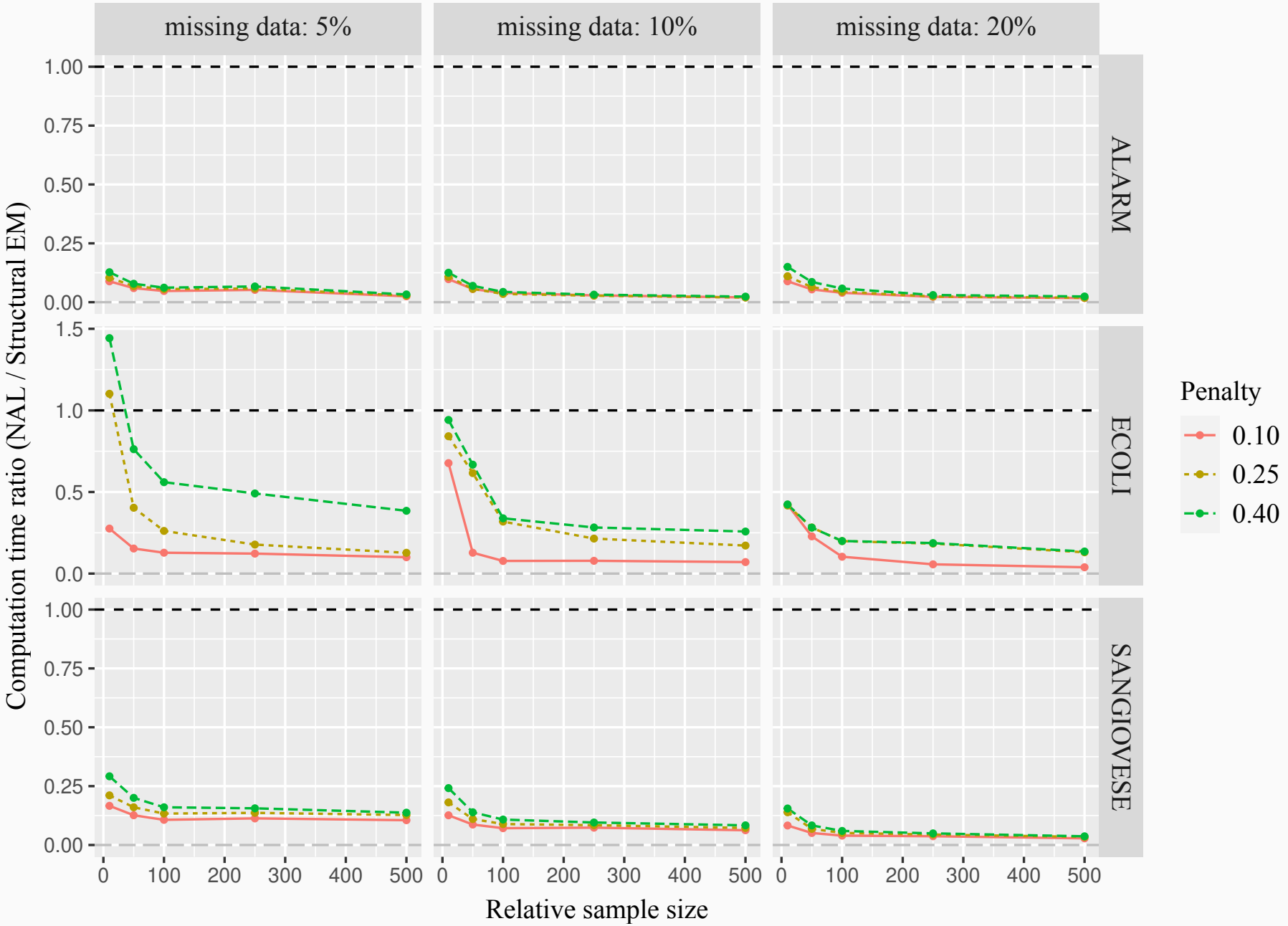
We concluded that:

- In BIC, $n\lambda_n = \log(n)/2 \to \infty$ and $\sqrt{n}\lambda_n = \log(n)/(2\sqrt{n}) \to 0$, so BIC is consistent for complete data but not for incomplete data.
- AIC is not consistent for either complete or incomplete data, confirming [4].
- How to choose λ_n is an open problem.

Structural EM vs Node-Averaged Likelihood: Accuracy



Structural EM vs Node-Averaged Likelihood: Speed



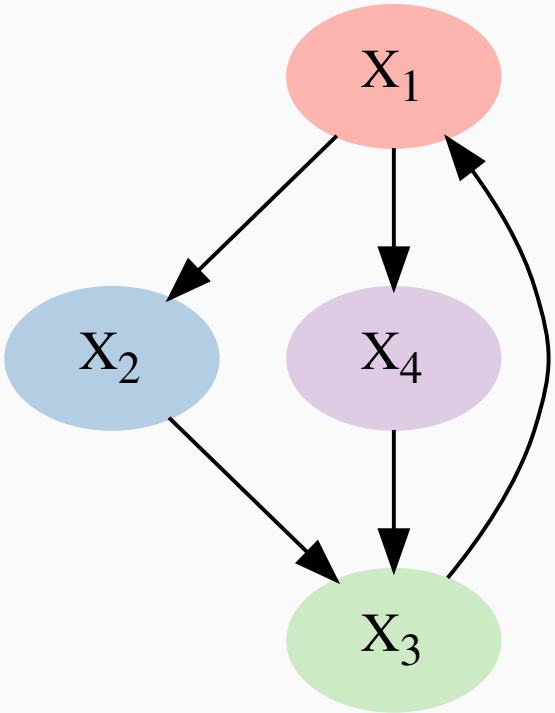
- ✓ Bayesian Networks
- ✓ Incomplete Data
- → Dynamic Networks
- Related Data Sets

Continuous-Time Bayesian Networks

Continuous-Time BNs (CTBNs) are a framework for modelling finite-state, continuous-time processes. Their graphical representation allows for natural, cyclic dependency graphs without having to specify a temporal granularity [12].

A CTBN consists of two components:

- A directed graph encoding conditional independencies.
- A conditional intensity matrix (CIM) $\mathbf{Q}_{X_i \mid \mathbf{u}}$ describing the evolution process of a variable with the parameters
  - $\mathbf{q}_{X_i}$: a set of intensities parameterising the exponential distributions over when the next transition occurs.
  - $\boldsymbol{\theta}_{X_i}$: a set of probabilities parameterising the distribution over where the state transitions.



Constraint-Based Structure Learning?

Score-based learning was covered by Nodelman [12] in his original work on CTBNs. For constraint-based structure learning we need a new definition of conditional independence [5]:

> Let $\mathcal{N}$ be a CTBN with a graph $\mathcal{G}$ over $\mathbf{X}$. We say that $X_i \perp\!\!\!\perp X_j \mid \mathbf{S}_{X_i,X_j}$ if $\mathbf{Q}_{X_i \mid x,\mathbf{s}} = \mathbf{Q}_{X_i \mid \mathbf{s}}$ for all values x, s of X_j and $\mathbf{S}_{X_i,X_j}$.

Note that conditional independence is not symmetric in CTBNs! To test it we need to test two separate hypotheses:

- Time To Transition: independence of the waiting times ($\mathbf{q}_{X_i}$), tested with an F test to compare their exponential distributions.
- State-to-State Transition: independence of the transitions ($\boldsymbol{\theta}_{X_i}$), tested with a two-sample χ^2 test or a Kolmogorov-Smirnov test.

We test time-to-transition hypothesis first and then, if the null is rejected, the state-to-state hypotheses. If both nulls are rejected, X_i and X_j are conditionally independent.

A PC Algorithm for Continuous-Time Bayesian Networks

Given how different is the definition of conditional independence, we need to adapt the PC algorithm [6] to match.

1. Form a complete directed graph $\mathcal{G}$ over $\mathbf{X}$.
2. For each variable X_i:
   - 2.1 Set $\mathbf{U} = \{X_j \in \mathbf{X} : X_j \to X_i\}$, the current parent set.
   - 2.2 For increasing values $b = 0, \ldots, |\mathbf{U}|$:
     - (a) For each $X_j \in \mathbf{U}$, test $X_i \perp\!\!\!\perp X_j \mid \mathbf{S}_{X_i, X_j}$ for all possible subsets of size b of $\mathbf{U} \setminus X_j$.
     - (b) As soon as $X_i \perp\!\!\!\perp X_j \mid \mathbf{S}_{X_i, X_j}$ for some $\mathbf{S}_{X_i, X_j}$, remove $X_j \to X_i$ from $\mathcal{G}$ and X_j from $\mathbf{U}$.
3. Return $\mathcal{G}$.

We call this the Continuous-Time PC (CTPC) algorithm [5]. It has better structural reconstruction accuracy than the score-based approach in [12], both both approaches are slow: they are only practical for less than 20 variables.

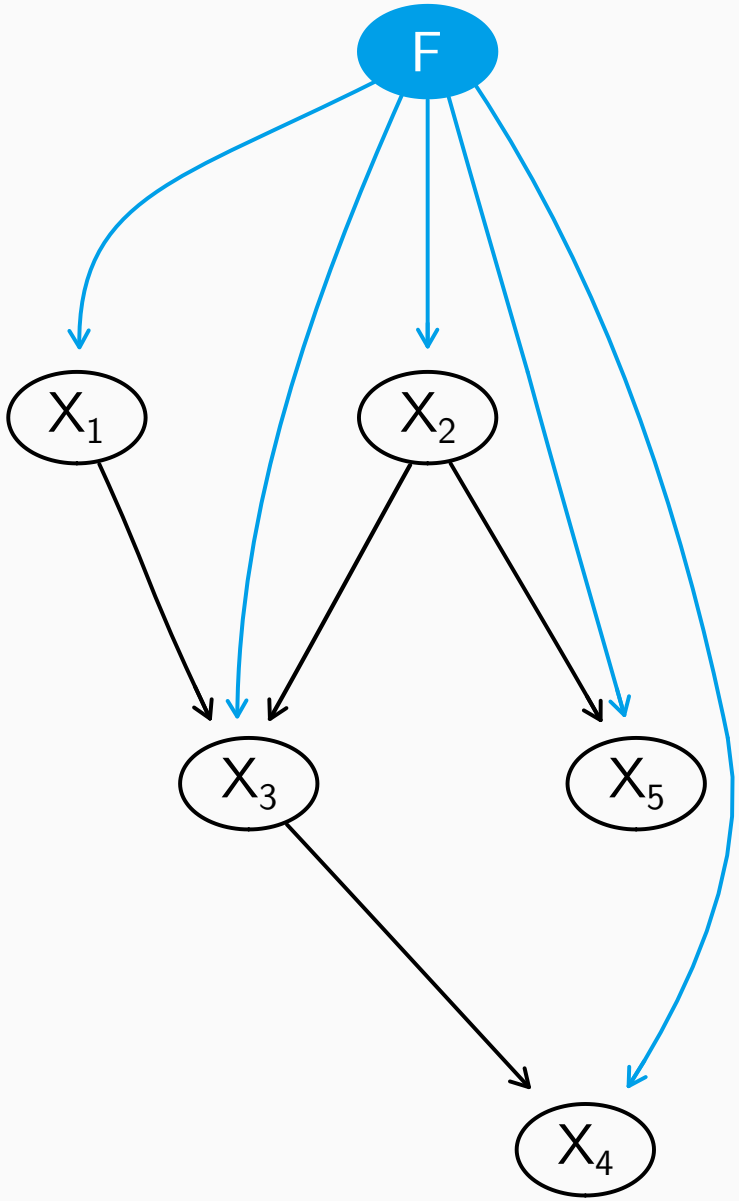
- ✔ Bayesian Networks
- ✔ Incomplete Data
- ✔ Dynamic Networks
- → Related Data Sets

Related Data Sets

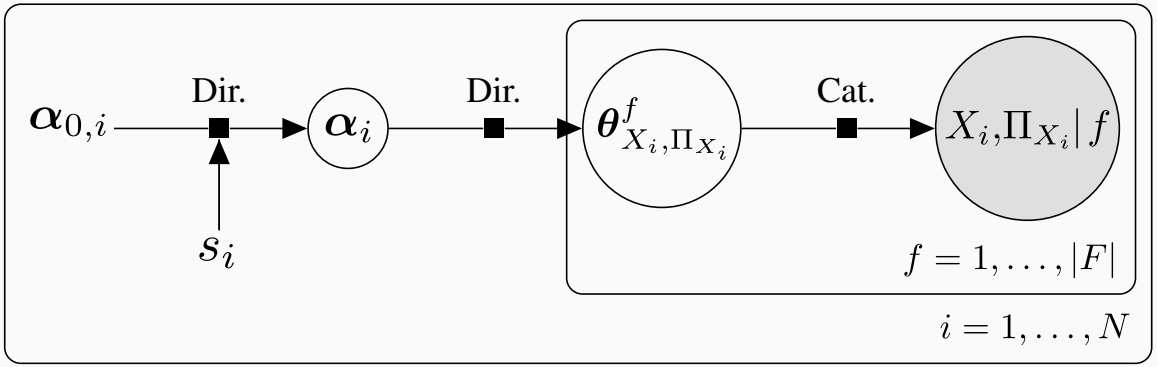
The aim: learning the structure of a BN from a set of related data sets identified by F, which is assumed known.

The approach: we would like to do that by pooling information across different data sets to distil structural features that are common to all of them.

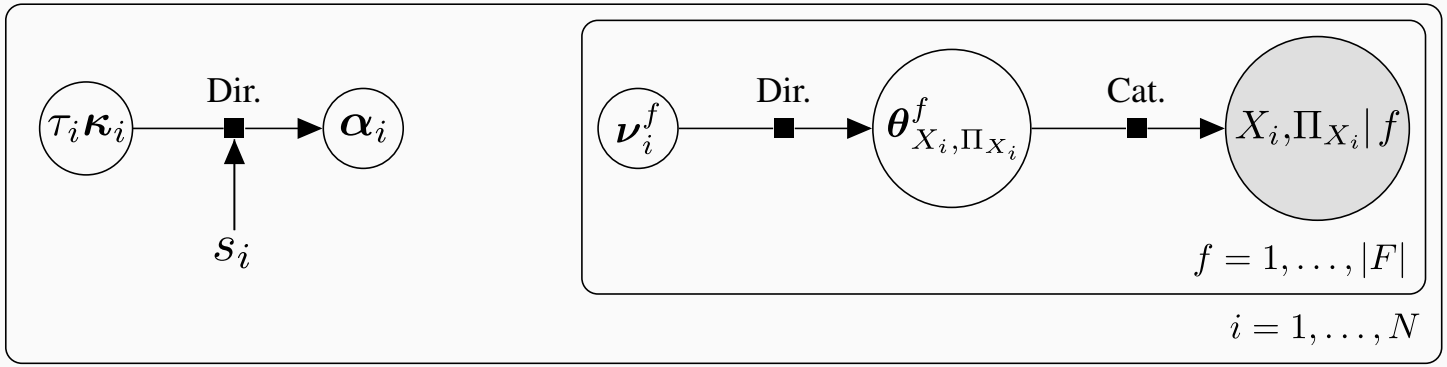
The mathematical formulation: a Bayesian Dirichlet score with a hierarchical prior (BHD), with some variational Bayes sprinkled on top to make it closed form [1].



The Hierarchical Model



Hierarchical Model



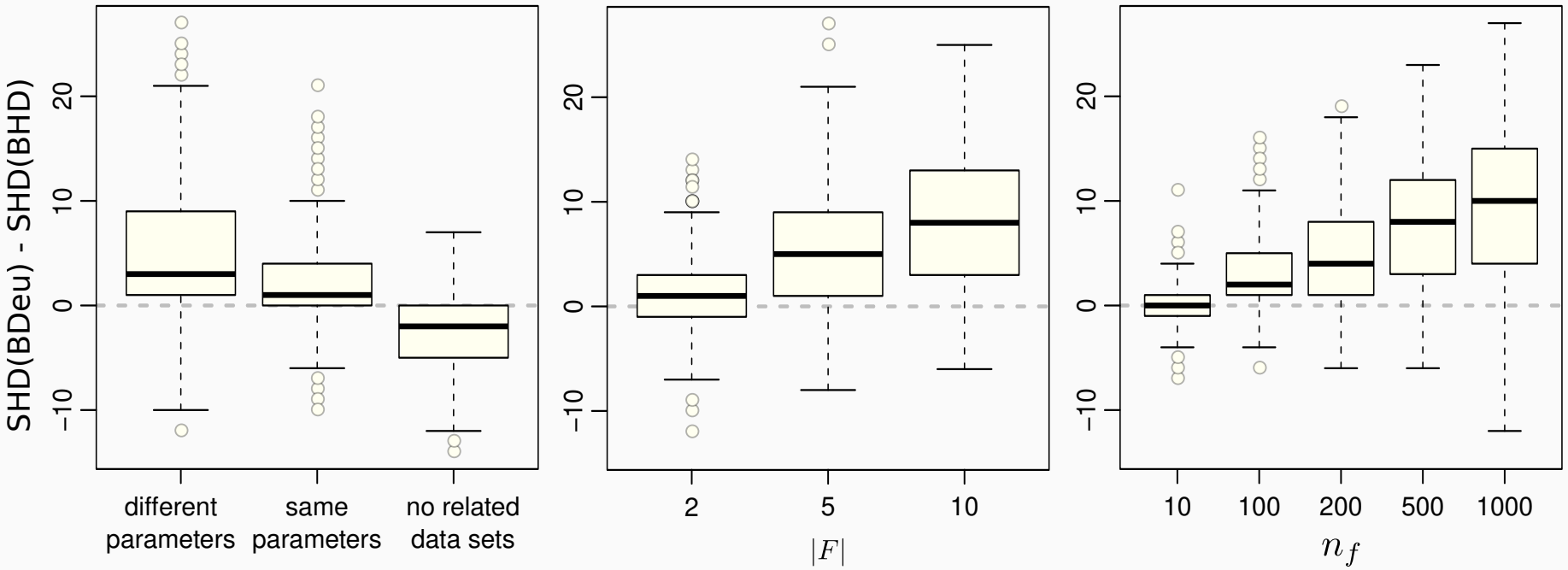
Variational Approximation

Thus we get BHD:

$$\mathrm{P}(\mathcal{D} \mid F, \mathcal{G}) \approx \prod_{i=1}^{N} \prod_{f=1}^{|F|} \prod_{j=1}^{|\Pi_{X_i}|} \left[\frac{\Gamma(s_i \hat{\kappa}_{ij})}{\Gamma(s_i \hat{\kappa}_{ij} + n^f_{ij})} \prod_{k=1}^{|X_i|} \frac{\Gamma(s_i \hat{\kappa}_{ijk} + n^f_{ijk})}{\Gamma(s_i \hat{\kappa}_{ijk})} \right]$$

where $s_i \hat{\kappa}_{ijk}$ = the posterior mean of α_{ijk} under the variational model.

BHD versus BDeu



The BHD score:

- has better structural accuracy than BDeu when we are modelling related data sets;
- it gets increasingly better as the number of related grows;
- it gets increasingly better as the size of (at least some of) the individual related data sets grows.

Conclusions

Bayesian networks are a fundamental tool in machine learning: their definition can be extended to unify and subsume models ranging from missingness patterns to stochastic processes with latent variables [14].

We can further extend and learn them from:

- from incomplete data: moving beyond Structural EM, which is slow and complicated to tune [13], to a simpler score-based approach used with the node-average likelihood [3].
- from continuous-time time series, with the CTPC algorithm and a suitable characterisation of conditional independence [5];
- from collections of related data sets, pooling information with the BHD score [1].

Acknowledgements

Tjebbe Bodewes
University of Oxford
(now at *Zivver* in The Netherlands)

Andrea Ruggieri
Francesco Stranieri
Alessandro Bregoli
Fabio Stella
Università degli Studi di Milano-Bicocca

Laura Azzimonti
Istituto Dalle Molle di Studi sull'Intelligenza Artificiale (IDSIA)

Thanks!

Any questions?

References I

- L. Azzimonti, G. Corani, and M. Scutari.
  Structure Learning for Related Data Sets with a Hierarchical Bayesian Score.
  Proceedings of Machine Learning Research (PGM 2020), 138:5–16, 2020.
- N. Balov.
  Consistent Model Selection of Discrete Bayesian Networks from Incomplete Data.
  Electronic Journal of Statistics, 7:1047–1077, 2013.
- T. Bodewes and M. Scutari.
  Identifiability and Consistency of Bayesian Network Structure Learning from Incomplete Data.
  Proceedings of Machine Learning Research (PGM 2020), 138:29–40, 2020.
- H. Bozdogan.
  Model Selection and Akaike's Information Criterion (AIC): The General Theory and its Analytical Extensions.
  Psychometrika, 52(3):345–370, 1987.
- A. Bregoli, M. Scutari, and F. Stella.
  Constraint-Based Learning for Continuous-Time Bayesian Networks.
  Proceedings of Machine Learning Research (PGM 2020), 138:41–52, 2020.
- D. Colombo and M. H. Maathuis.
  Order-Independent Constraint-Based Causal Structure Learning.
  Journal of Machine Learning Research, 15:3921–3962, 2014.

References II

- N. Friedman.
  Learning Belief Networks in the Presence of Missing Values and Hidden Variables.
  In *ICML*, pages 125–133, 1997.
- N. Friedman.
  The Bayesian Structural EM Algorithm.
  In *UAI*, pages 129–138, 1998.
- D. Heckerman and D. Geiger.
  Learning Bayesian Networks: a Unification for Discrete and Gaussian Domains.
  In *UAI*, pages 274–284, 1995.
- D. Koller and N. Friedman.
  Probabilistic Graphical Models: Principles and Techniques.
  MIT Press, 2009.
- G. J. McLachlan and T. Krishnan.
  The EM Algorithm and Extensions.
  Wiley, 2008.
- U. D. Nodelman.
  Continuous Time Bayesian Networks.
  PhD thesis, Stanford University, 2007.

References III

- A. Ruggieri, F. Stranieri, F. Stella, and M Scutari.
  Hard and Soft EM in Bayesian Network Learning from Incomplete Data.
  Algorithms, 13(12):329, 2020.
- M. Scutari.
  Bayesian Network Models for Incomplete and Dynamic Data.
  Statistica Neerlandica, 74(3):397–419, 2020.
- T. S. Verma and J. Pearl.
  Equivalence and Synthesis of Causal Models.
  Uncertainty in Artificial Intelligence, 6:255–268, 1991.