Università degli Studi di Milano-Bicocca

corso di dottorato in statistica

Ph.D. Thesis

Objective Bayesian Analysis for Differential Gaussian Directed Acyclic Graphs

Andrea Artaria

Supervisor:

Prof. Guido Consonni

Tutor:

Prof. Andrea Ongaro

Ciclo XXVII

Contents

Abstract

Often we are confronted with heterogeneous multivariate data, i.e., data coming from several categories, and the interest may center on the differential structure of stochastic dependence among the variables between the groups. We concentrate on the comparison between two Directed Acyclic Graph (DAG) models. For instance, one DAG relates to healthy people, and the other to patients affected by a disease. The objective is to find differences, if any, between the two DAGs, both in terms of conditional independencies and in terms of the strengths of common dependencies. This is achieved through a Bayesian model selection strategy. We assume that the two graph models are jointly distributed according to a multivariate Gaussian family. The advantage of a joint modelling approach is to exploit, whenever they exist, similarities between the graphs. We approach model selection using an objective Bayes framework, so that minimal prior elicitation is needed. We elaborated a modelling framework incorporating a sparsity assumption, which is likely to be satisfied when the number of variables is high or very high. To this end, we make use of non-local priors on the regression coefficients to further enhance simple models having a good fit. We develop an efficient search strategy over the space of pairs of DAGs, and test our procedure by means of simulations. Results are presented in terms of operating characteristic curves and related indexes.

Chapter 1

Conditional Independence

Independence of random variables is a very strong statement on the relationship it exists between variables. In the case of two random variables X and Y, independence is defined as[1]

$$p_{X,Y}(x,y) = p_X(x)p_Y(y) \quad \text{for every } x, y$$

and is denoted by $X \perp\!\!\!\perp Y$. This yields, conditioning on events with positive probability, to

- $p_{X|Y}(x|y) = p(x)$
- $p_{Y|X}(y|x) = p(y)$

which makes clear that independence means that Y doesn't contain any useful information about X and vice versa.

When it comes to complex models, independence may not be sufficient to understand the relationships governing the variables, and a "looser" concept of independence, i.e. *conditional independence*, gives us a lot of information about them. Two random variables X and Y are conditionally independent given the random variable Z if, for all z such that $p_Z(z) > 0$

$$p_{X,Y|Z}(x, y \mid z) = p_{X|Z}(x \mid z)p_{Y|Z}(y \mid z) \quad \text{for every } x, y$$

and we write $X \perp\!\!\!\perp Y \mid Z$. Easily it can be shown that the following properties are equivalent (we omit subscripts to improve readability)

[1] Here like in all this work we let $p(\cdot)$ be the Radon-Nikodym derivative of the probability distribution P with respect to a dominating σ-finite measure (e.g. $p_X(\cdot)$ is the density (pdf) or the probability mass function (pmf) of the random variable X).

- $p(x, y, z) = p(x, z)p(y, z)/p(z)$;
- $p(x, y, z) = p(x \mid z)p(y \mid z)p(z)$;
- $p(x \mid y, z) = p(x \mid z)$.

From the last equation we can interpret conditional independence in the same way we did for marginal independence, namely we can say Y doesn't contain any useful information about X provided that Z is given.

An useful result to easily find conditional independence relationships from probability mass functions or probability density functions is given by the following theorem.

Theorem 1 (factorization criterion) *Let V be a set of random variables and S, U, T a partition of V then:*

$$U \perp\!\!\!\perp T \mid S \Leftrightarrow p(v) = h(u, s)k(t, s) \qquad \square$$

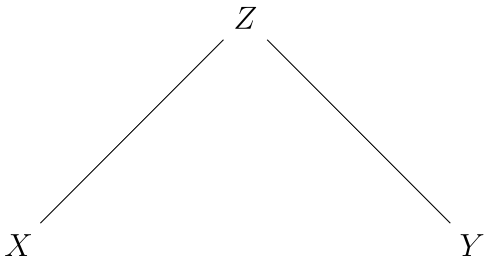
Conditional independence is a very helpful tool, since it helps the researcher to understand complex problems trough simple statements. The problem is that in complex probability systems there can be a huge number of conditional independence statements, making the interpretation of the problem extremely laborious. As always, when properly designed, a graphical representation is a very powerful tool for making problems understandable.

A natural way to represent relationships between variables is through *simple graphs*

Definition 1 (simple graph) A simple[2] graph is an ordered pair $\mathcal{G} = (V, E)$ of sets, with $V = \{1, 2, 3, \dots, q\}$ a finite set of nodes or vertices and $E \subseteq V \times V$ the set of the edges, such that it does not allows for self-loops, $(i, i) \notin E \ \forall i \in V$, and multiple edges. $\square$

We see from this definition that elements of E are couples of elements of V, and thus we may think nodes as random variables and represent dependence between variables through the edges. But in reality we are interested in the lack of dependence, or better conditional independence. The easiest statement we can think about is $X \perp\!\!\!\perp Y \mid Z$, where X, Y, Z are random variables. A natural way to represent this ternary relationship graphically might be

[2] Note that in this work we will always talk about graphs that are simple, and every time we will talk about a graph we will mean a simple graph.



we can see the lack of dependence being represented by the lack of an edge between X and Y, but that they are still connected through Z, so that we can distinguish conditional independence from *marginal independence*, where nodes would not be connected at all. Graphically this can be viewed as a ternary relationship and we can say that Z separates X from Y. This is just a very simple example and systems can become much more complex than that, therefore we need a solid theory that helps us to connect this algebraic objects with probabilistic models. Before we continue we may introduce some basic concepts of graph theory.

Chapter 2

Graphical Models

To simplify our notation we will indicate nodes with integers and small Latin letters in general and set of nodes with capital Latin letters, e.g. $V = \{1, 2, 3, ..., v, ..., q\}$. Finally random variables associated to a node v as X_v and random vectors associated with a set A as X_A.

2.1 Undirected Graphs

Definition 2 (undirected graph (UG)) A graph $\mathcal{G} = (V, E)$ is called *undirected* when edges are seen as unordered pairs

$$(i, j) \in E \quad \Leftrightarrow \quad (j, i) \in E \qquad \square$$

Definition 3 (adjacent) If $(v_i, v_j) \in E$ then v_i and v_j are said to be *adjacent.* $\square$

Definition 4 (adjacency matrix) A matrix A of size $|V|\times|V|$ with entries $a_{ij} = 1$ when v_i is adjacent to v_j and 0 otherwise is called *adjacency matrix.* $\square$

Definition 5 (path) We define a *path* from node a to node b of length n a sequence $a = v_0, \ldots, v_n = b$ of vertices where subsequent elements in the sequence are adjacent:

$$(v_{j-1}, v_j) \in E, j = 1, \ldots, n \qquad \square$$

Definition 6 (cycle) Let $(v_1, \ldots, v_n)$ be a path, if $v_1 = v_n$ then the path $(v_1, \ldots, v_n)$ is called *cycle.* $\square$

Definition 7 (boundary) We define *boundary* of S, $\mathrm{bd}(S)$, the set of the vertices not in S adjacent with at least one vertex in S:

$$\mathrm{bd}(S) = \{a \in V \setminus S : (\exists b \in S : (a, b) \in E)\}$$ □

Definition 8 (vertex induced subgraph) Consider an undirected graph $\mathcal{G} = (V, E)$, let $S \subset V$ and $E_S = E \cap (S \times S)$ then the couple $G_S = (S, E_S)$ is called *subgraph induced by* $S \subset V$. □

From this definition we note that E_S contains only edges in E that connect couples of vertices of S.

Definition 9 (complete graphs and complete subsets) $S \subseteq V$ is said to be *a complete subset* if and only if $\mathcal{G}_S = (S, E_S)$, the graph induced by S, is *complete*, namely if every pair of distinct vertices in S are adjacent, i.e.

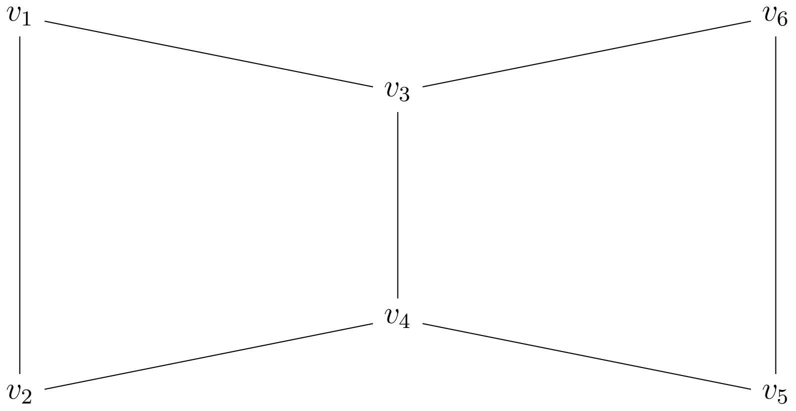
$$E_S = S \times S$$ □

Definition 10 (cliques) A clique is a maximal complete set, i.e. a set that is complete and is not subset of any other complete set. □

The family of the cliques is sometimes called also the graph generator, since an undirected graph is uniquely itentified by the set $\mathcal{C}$ of its cliques.

We present two simple examples to visualize some of the concepts viewed until now

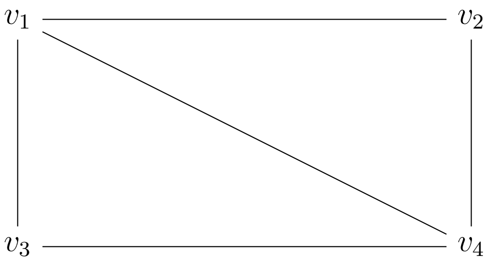
Example 1



□

- (v_1, v_3, v_4, v_5) is a path, $(v_1, v_3, v_4, v_2, v_1)$ is a cycle;
- Let $A = \{v_1, v_2\}$, then $\text{bd}(A) = \{v_3, v_4\}$

Example 2



- the graph induced by $\{v_2, v_3\}$ is not complete;
- the graph induced by $\{v_1, v_2\}$ is complete but $\{v_1, v_2\}$ is not a clique;
- the graph induced by $\{v_1, v_2, v_4\}$ is complete and $\{v_1, v_2, v_4\}$ is a clique; □

Now that we have some terminology we can formalize the concept of *separation*.

Definition 11 (separation in an undirected graph) A subset $S \subseteq V$ is said to separate $a, b \in V$ in $\mathcal{G}$ if every path form a to b intersects S. Let A, B and S be three subsets of V. S separates A and B in $\mathcal{G}$ if S separates every vertex in A from every vertex in B. We write $A \perp_{\mathcal{G}} B \,|\, S$. □

The connection between separation and conditional independence has been formalized from Pearl and Paz (1987) through the concept of *graphoid*.

Definition 12 (semi-graphoid and graphoid) A ternary realationship $\perp_\sigma$ is a *semi-graphoid* if and only if for any disjoint subsets A, B, C and D of V it holds that:

(S1) $A \perp_\sigma B \,|\, C \Rightarrow B \perp_\sigma A \,|\, C$ (simmetry)

(S2) $A \perp_\sigma B \,|\, C$ and $D \subset B \Rightarrow A \perp_\sigma D \,|\, C$ (decomposition)

(S3) $A \perp_\sigma B \,|\, C$ and $D \subset B \Rightarrow A \perp_\sigma B \,|\, C \cup D$ (weak-union)

(S4) $A \perp_\sigma B \,|\, C$ and $A \perp_\sigma D \,|\, B \cup C \Rightarrow A \perp_\sigma B \cup D \,|\, C$ (contraction)

If it is also true that:

(S5) $A \perp_\sigma B \,|\, C \cup D$ and $A \perp_\sigma C \,|\, B \cup D \Rightarrow A \perp_\sigma B \cup C \,|\, D$ (intersection)

then the ternary relationship is called graphoid. □

It is not difficult to see that the ternary relationship $\perp_{\mathcal{G}}$ of separation is a graphoid. Exploiting rules of conditional independence we can also show that the ternary relationship $\perp\!\!\!\perp$ of conditional indipendence is a semi-graphoid and that a sufficient (but not necessary) condition for $\perp\!\!\!\perp$ to be a graphoid is that the joint distribution is strictly positive.

Proof To prove that $\perp\!\!\!\perp$ is a semi-graphoid we have to prove that condition (S1) to (S4) are satisfied by $\perp\!\!\!\perp$. With abuse of notation, we are indicating with A the random vector X_A:

(C1) $A \perp\!\!\!\perp B \,|\, C \Rightarrow B \perp\!\!\!\perp A \,|\, C$

Proof by definition

(C2) $A \perp\!\!\!\perp B \,|\, C$ and $D \subset B \Rightarrow A \perp_\sigma D \,|\, C$

Proof

$$\begin{aligned} p(X_A, X_B, X_C) &= p(X_A, X_D, X_{B\setminus D}, X_C) \overset{(FC)}{=} h(X_A, X_C)k(X_B, X_C) \\ &= h(X_A, X_C)k(X_D, X_{B\setminus D}, X_C) \end{aligned}$$

integrating $X_{B\setminus D}$ we get:

$$p(X_A, X_D, X_C) = h(X_A, X_C)k^*(X_D, X_C)$$ ■

(C3) $A \perp\!\!\!\perp B \,|\, C$ and $D \subset B \Rightarrow A \perp\!\!\!\perp B \,|\, C \cup D$

Proof

$$\begin{aligned} & p(A|B, C) = p(A|C) \wedge p(A|D, C) = p(A|C) \\ & \Leftrightarrow p(A|B, C) = p(A|D, C) \\ & \Leftrightarrow p(A|(B \setminus D), D, C) = p(A|D, C) \\ & \Leftrightarrow A \perp\!\!\!\perp (B \setminus D) \,|\, D \cup C \\ & \Leftrightarrow p(A, B, C, D) = h(A, D, C)k(B\setminus D, D, C) = h(A, D, C)k(B, D, C) \\ & \Leftrightarrow A \perp\!\!\!\perp B \,|\, D \cup C \end{aligned}$$

(C4)

$$A \perp\!\!\!\perp B \,|\, C \tag{2.1}$$

and

$$A \perp\!\!\!\perp D \,|\, B \cup C \tag{2.2}$$

implies

$$A \perp\!\!\!\perp B \cup D \,|\, C$$

Proof

$$\begin{aligned}
&p(A, D, B, C)\\
&\overset{(2.2)}{=} p(A, B, C)p(D, B, C)/p(B, C)\\
&\overset{(2.1)}{=} h(A, C)k(B, C)p(D, B, C)/p(B, C)\\
&= h(A, C)k^*(B, C, D)\\
&\Leftrightarrow A \perp\!\!\!\perp B \cup D \,|\, C
\end{aligned}$$

The first part is now proved. We still need to verify (S5).

(C5) $A \perp\!\!\!\perp C \,|\, B \cup D \quad \wedge \quad A \perp\!\!\!\perp B \,|\, C \cup D \quad \Rightarrow \quad A \perp\!\!\!\perp B \cup C \,|\, D$

Proof If $A \perp\!\!\!\perp B \,|\, C \cup D$ holds, then $p(A, B, C, D) = h_1(A, C, D)k_1(B, C, D)$ and if $A \perp\!\!\!\perp C \,|\, B \cup D$ holds, then $p(A, B, C, D) = h_2(A, B, D)k_2(C, B, D)$. And this implies that:

$$h_1(A, C, D)k_1(B, C, D) = h_2(A, B, D)k_2(C, B, D) \tag{2.3}$$

But this is equal, assuming $k_1(B, C, D) > 0$, to:

$$h_1(A, C, D) = k_2(A, B, D)h_2(C, B, D)/k_1(B, C, D) = h^*(A, D)w(C, D)$$

since $h_1(A, C, D)$ must not depend on B, and this implies, substituing in (2.3), that:

$$p(A, B, C, D) = h^*(A, D)w(C, D)k_1(B, C, D) = h^*(A, D)k^*(B, C, D)$$

with

$$k^*(B, C, D) = w(C, D)k_1(B, C, D)$$ ■

The results now follows from the factorization criterion. ■

2.1.1 Undirected Markov Properties

Given the concept of separation we can define three different ways of connecting conditional independencies with a graph $\mathcal{G} = (V, E)$. We say a probability distribution on X_V satisfies

(P) the *pairwise Markov property* with respect to $\mathcal{G}$ if:

$$(i, j) \notin E \Rightarrow X_i \perp\!\!\!\perp X_j \,|\, X_{V \setminus \{i,j\}}$$

(L) the *local Markov property* with respect to $\mathcal{G}$ if for every $i \in V$:

$$X_i \perp\!\!\!\perp X_{V \setminus (\mathrm{bd}(i) \cup \{i\})} \,|\, X_{\mathrm{bd}(i)}$$

(G) the *global Markov property* with respect to $\mathcal{G}$ if:

$$A \perp_{\mathcal{G}} B \,|\, S \Rightarrow X_A \perp\!\!\!\perp X_B \,|\, X_S$$

It's easy to see that (G) $\Rightarrow$ (L), to see that (L) $\Rightarrow$ (P) we have to note that since i and j are not adjacent $j \in V \setminus (\mathrm{bd}(i) \cup \{i\})$ and therefore $\mathrm{bd}(i) \cup ((V \setminus (\mathrm{bd}(i) \cup \{i\})) \setminus j) = V \setminus \{i, j\}$. So applying (C3) to (L) gives $X_i \perp\!\!\!\perp X_{V \setminus (\mathrm{bd}(i) \cup \{i\})} \,|\, X_{V \setminus \{i,j\}}$, and applying (C2) we get (P). Note that since we used just (C2) and (C3) to prove these implications they hold for any semi-graphoid relationship.

Theorem 1 (Pearl and Paz) *If the probability distribution of X_V satisfies (C5) then it also holds that (P) $\Rightarrow$ (G)* □

To prove this theorem it is possible to use just the graphoid axioms (C1) to (C5) and therefore also here conditional independence could be replaced by any graphoid defined on V.

2.1.2 Factorization

Beside for visual representation graphs are also really useful for computation, since they are objects computers easily understand and through their properties and relationship with probability distributions we can achieve local computation easily. In this context *factorization* and his relationships with Markov properties become essential.

Definition 13 (factorization w.r.t. a graph) Let $G = (V, E)$ be an undirected graph and let $p(x_V)$ be the probability density function (or probability mass function) of X_V. If $p(x_V)$ admits a factorization of the form

$$p(x_V) = \prod_{T \subseteq V : \text{T complete}} g_i(x_T) \tag{F}$$

for some function $g_1(\cdot) \dots g_k(\cdot)$ where $g_j(x_T)$ depends on x_V only through x_T, then we say that the distribution of X_V factorizes with respect to $\mathcal{G}$.

It's really important to note that the functions $g_j(x_T)$ are not uniquely determined and can be multiplied or splitted up in different ways, so without loss of generality we can rewrite (F) with respect to the cliques of $\mathcal{G}$:

$$p(x_V) = \prod_{T \subseteq \mathcal{C}} g_i(x_T)$$

where $\mathcal{C}$ is the set of cliques of $\mathcal{G}$. □

Given the definition of factorization w.r.t. a graph we immediately note that the factorization criterion in conjunction with (C2) gives us the connection between factorization and Markov properties, i.e. $(F) \Rightarrow (G)$, so that:

$$(F) \Rightarrow (G) \Rightarrow (L) \Rightarrow (P)$$

If (P) would imply (F) we could show that all the Markov properties are equivalent, and that the factorization (w.r.t. a graph) would be a necessary and sufficient condition for a distribution to be Markov w.r.t. a graph. This has been shown to be true by Hammersley and Clifford (1971) in the case the pdf (or pmf) of X_V is strictly positive.

Theorem 2 (Hammersley Clifford theorem) *If the pdf (or pmf) of X_V is strictly positive then the pairwise Markov property implies the factorization property.* □

Thanks to the Hammersley Clifford theorem we know that a strictly positive probability distribution factorizes with respect to a graph $\mathcal{G}$ if and only if it is Markovian w.r.t. $\mathcal{G}$ (in any sense). But even if we knew that a distribution is Markov with respect to $\mathcal{G}$ and strictly positive we don't have a tool to determine which are the local components $g_i(x_T)$ in (F), in that sense decomposable graphs have an important role.

2.1.3 Decomposable graphs

Definition 14 (decomposition) The pair (A, B) of subsets of V, with $V = A \cup B$ forms a (weak) *decomposition* of $\mathcal{G}$ if it holds that

- $A \cap B$ separates A from B in $\mathcal{G}$
- $A \cap B$ is a complete subset of V. □

Definition 15 (proper decomposition) We say a decomposition is *proper* if $A \setminus B$ and $B \setminus A$ are both non-empty. □

Definition 16 (components and prime components) A decomposition (A, B) decomposes $\mathcal{G}$ into the *components* $\mathcal{G}_A$ and $\mathcal{G}_B$. Since a component takes the form of a subgraph clearly it may or may not admit further decompositions. If a component does not admit further decompositions it is called prime and the components we get by recursively decomposing $\mathcal{G}$ until no more components can be further decomposed are called *maximal prime components* of $\mathcal{G}$. □

Definition 17 (recursive definition of decomposable graph) An undirected graph $\mathcal{G}$ is said to be decomposable if one of the following statements holds:

1. $\mathcal{G}$ is complete
2. $\mathcal{G}$ admits a proper decomposition (A, B) into decomposable components $\mathcal{G}_A$,$\mathcal{G}_B$ □

Looking at the recursive definition into more detail we see that a graph is decomposable if and only if all its maximal prime components are complete and since we continue decomposing until we find a complete component this has to be induced by a clique of $\mathcal{G}$.

Proposition 1 *Assume that (A,B) decomposes $\mathcal{G} = (V, E)$, then a probability distribution P_V is globally Markov with respect to G if and only if both its marginal distributions P_A and P_B are globally Markov with respect to $\mathcal{G}_A$ and $\mathcal{G}_B$ respectively and the the densities satisfy*

$$p_V(x_V)p(x_{A\cap B}) = p_A(x_A)p_B(x_B)$$ □

Let $\mathcal{G}$ be a decomposable graph with k cliques, $C_1, \dots, C_k$, and $k-1$ separators, $S_1, \dots, S_{k-1}$, resulting from the recursive decomposition process, applying Proposition 1 recursively we obtain the following factorization

$$p_V(x_V) = \frac{\prod_{i=1}^{k} p_{C_i}(x_{C_i})}{\prod_{i=1}^{k-1} p_{S_i}(x_{S_i})}$$

Definition 18 (Chordal/Triangulated graphs) Another characterization of a decomposable graph is given through the concept of chordality, i.e. an undirected graph is decomposable if and only if it is chordal, where a graph is chordal if every cycle of length ≥ 4 has a chord, that is, two non-consecutive vertices that are adjacent. □

Strictly connected with chordality is the concept of perfect numbering of the vertices of an undericted graph

Definition 19 (Perfect numbering) A numbering $V = \{1, 2, \dots, q\}$ is perfect if for every $i = 2, \dots, q$ it holds that:

$$\text{bd}(i) \cap \{1, \dots, i-1\} \quad \text{is complete in } \mathcal{G}$$ □

Both these definitions become important since it can be proved that an undirected graph is decomposable if and only if its vertices admit a perfect numbering, and algorithms, like the maximum cardinality search, can be developed efficiently to perform this check.

2.1.4 Gaussian graphical models

Given a graph $G = (V, E)$, when we talk about a *graphical model* $M(\mathcal{G})$ we're essentially talking about the family of distributions on X_V that satisfy the conditional independence statements encoded in $\mathcal{G}$. Then, a *Gaussian (undirected) graphical model*, is a subset of $M(\mathcal{G})$, which contains just multivariate normal distributions. Furthermore we don't even have to care about specifying which undirected Markov properties they have to satisfy, since in this context they're all equivalent, because $p_V(x) > 0 \ \forall x$.

A nice feature of Gaussian graphical models is that we can exploit the precision matrix of the Gaussian distribution to read conditional independence relationships easily. Without loss of generality, suppose to have a zero mean multivariate Gaussian distribution with density

$$\begin{aligned} p_V(x) &\propto \exp\left\{-\frac{1}{2}x^\top \Sigma^{-1} x\right\} \\ &= \exp\left\{-\frac{1}{2}\operatorname{tr}\left(x^\top \Sigma^{-1} x\right)\right\} \\ &= \exp\left\{-\frac{1}{2}\sum_{i=1}^{q}\sum_{j=1}^{q}\sigma^{ij}x_i x_j\right\} \end{aligned} \tag{2.4}$$

where σ^{ij} are the entries of the concentration matrix Σ^{-1}. We see that setting σ^{ij} to zero x_i and x_j will never compare in the same factor again, and thus from the factorization criterion

$$\sigma^{ij} = 0 \quad \Leftrightarrow \quad X_i \perp\!\!\!\perp X_j \mid X_{V\setminus\{i,j\}}$$

Therefore a multivariate normal distribution will be pairwise Markov, and thus Markov, with respect to a graph which has a missing edge between i and j for each $\sigma^{ij} = 0$, this graph is often called the *concentration graph.*

2.2 Directed acyclic graphical models

In the previous chapter we put a lot of attention on the concept of factorization w.r.t. an undirected graph $\mathcal{G}$. But if we ask about factorization in probability, probably the most of us would immediately think about the chain rule of probability

$$p(x_1, \dots, x_n) = p(x_1)p(x_2 \mid x_1) \cdots p(x_n \mid x_1, \dots, x_{n-1})$$

From this relationship we see that if we knew which conditional independence statements were true, we could highly simply the model, and already work on an efficient framework for computation.

To express our chain rule we impose an ordering to the variables and implicitly we are not allowing to condition on variables with higher index. That reflects in our graph theory, respectively, in adding arrows to the edges, in order to direct the relationships, and not allowing directed cycles, to be coherent with the allowed conditioning sets. Under this framework we call the graph representation a *directed acyclic graph.* Among other things DAGs are useful to represent structural equation models (SEMs), which are nothing else than sets of multivariate regression models where the dependent variable in one regression may appear as independent variable in another equation. This brief note will become very important when considering Gaussian distributions, since we remember that conditioning in a Gaussian framework is essentially a regression.

We may note that adding arrows and disallowing directed cycles is not enough since we want our graph to be coherent with the other conditional independence statements implied by the model, and applying the rules derived for undirected graphs in this framework would not satisfy our needs.

2.2.1 Basic concepts

Before we can go into more detail we may introduce some new definition relative to directed graphs.

Definition 20 (directed graph) A *directed graph* is a graph $\mathcal{D} = (V, E)$ where edges are seen as ordered pairs. In general an edge $(a, b) \in E$ will be represented by an arrow $a \rightarrow b$. □

Definition 21 (directed path, directed cycle) a directed path from v_0 to v_k is path where all subsequent nodes are directed in a way such that

$$v_{i-1} \rightarrow v_i \quad \forall i = 1, \ldots, k$$

A directed path which begins and ends with the same vertex is a *directed cycle.* □

Definition 22 (directed acyclic graph) A *directed acyclic graph (DAG)* is a directed graph $\mathcal{D} = (V, E)$ with no directed cycles. □

Definition 23 (trail) a trail from v_0 to v_k is a path where edges may be directed in any sense. □

Definition 24 (parents, children) If there is an arrow that connects node a with b ($a \rightarrow b$) we say a is a *parent* of b and b is a *child* of a. The *set of parents* of node b is indicated as $\mathrm{pa}(b)$ and the *set of the children* of a as $\mathrm{ch}(a)$. □

Definition 25 (ancestors, descendants, non-descendants) If from a there is a *direct path* that leads to b ($a \rightarrow b$) we say that a is an *ancestor* of b, and b is a *descendant* of a. The *set of the ancestors* of b is indicated as $\mathrm{an}(b)$ and the *set of descendants* of a as $\mathrm{de}(a)$. The *set of the non-descendants* of a is $\mathrm{nd}(a) = V \setminus (\mathrm{de}(a) \cup \{a\})$ □

Definition 26 (ancestral set) If $\mathrm{an}(a) \subseteq A$ for all $a \in A$ we say that A is an *ancestral set*, and the *smallest ancestral set* containing A is indicated as $\mathrm{An}(A)$. □

2.2.2 Recursive Factorization and the directed Markov properties

We say a random vector X_V admits a *recursive factorization* (DF) with respect to a DAG $\mathcal{D} = (V, E)$ if the pdf (pmf) of X_V can be represented as

$$p_V(x_v) = \prod_{v \in V} p(x_v \mid x_{\mathrm{pa}(v)})$$

There is an important connection between DAGs and UGs, namely, given a DAG $\mathcal{D}$, if we join all unmarried, i.e. not connected, parents with a common child and make all the edges of the graph undirected, we get an undirected graph $\mathcal{D}^m$ called the the moral version of $\mathcal{D}$. Since the sets $\{a\} \cup \mathrm{pa}(a)$ are complete in $\mathcal{D}^m$, it is not difficult to see that if a distribution admits a recursive factorization w.r.t. $\mathcal{D}$ then it has the property to factorize (in the undirected sense) with respect to $\mathcal{D}^m$ and it is therefore globally Markov with respect to $\mathcal{D}^m$. Even though we note that if a probability distribution P factorizes according to to $\mathcal{D}^m$, P does not necessary recursively factorizes according to $\mathcal{D}$. We also note that if A is an ancestral set then the marginal distribution of A admits a recursive factorization with respect to $\mathcal{D}_A$, thus we can restrict our attention on this subgraph only, and applying the same reasoning again we can conclude that if a distribution P recursively factorizes with respect to a DAG $\mathcal{D}$ then

$$X_A \perp\!\!\!\perp X_B \,|\, X_S$$

whenever A and B are separated by S in the moral graph of the smallest ancestral set containing $A \cup B \cup S$, and this is called the *directed global Markov property.*

Definition 27 (well-ordering) We say that the vertices $v_1, \ldots, v_q$ of DAG $\mathcal{D}$ are well ordered when the arrows always point from vertices with lower index to vertices with higher index. Note that that a well-ordering is not unique. □

Definition 28 (predecessors) Given a well-ordering the predecessors of a node v, $\mathrm{pr}(v)$, are all nodes that come before v in the well-ordering. □

Another Markov property in the DAG framework is the *ordered Markov Property (O)*: we say that a probability distribution on X_V satisfies the ordered Markov property with respect to a DAG $\mathcal{D}$ if $\forall v \in V$

$$X_v \perp\!\!\!\perp X_{\mathrm{pr}(v) \setminus \mathrm{pa}(v)} \,|\, X_{\mathrm{pa}(v)}$$

As we have in the undirected graph framework also for directed acyclic graphs we have a local Markov property, namely a probability distribution on X_V is said to satisfy the *local Markov property (DL)* w.r.t. a DAG $\mathcal{D}$ if $\forall v \in V$ it holds that

$$X_v \perp\!\!\!\perp X_{\mathrm{nd}(v) \setminus \mathrm{pa}(v)} \,|\, X_{\mathrm{pa}(v)}$$

And finally also a *directed pairwise Markov property* exists namely a probability distribution on X_V satisfies the directed pairwise Markov property (DP) w.r.t. a DAG $\mathcal{D} = (V, E)$ if for any $i < j$

$$(i, j) \notin E \Rightarrow X_i \perp\!\!\!\perp X_j \,|\, X_{\mathrm{an}(i) \cup \mathrm{an}(j) \setminus \{i,j\}}$$

An important result is that here (O),(DL) and (DG) are equivalent, even without assuming positive pdf (pmf), while $(DG) \Rightarrow (DP)$, and are equivalent just in case of positive probability distribution.

2.2.3 d-separation

Another way to express the directed global Markov property is through d-separation: a trail π from a to c in $\mathcal{D}$ is said to be blocked by $S \in V$ if it contains a vertex b such that either

- $b \in S$ and at b the arrows of π do not meet head to head, or
- $b \notin S \wedge \mathrm{de}(b) \cap S = \emptyset$, and at b the arrows of π do meet head to head

A trail that is not blocked is said to be *active.* Two subsets A and B are said to be *d-separated* by S if all trails from A to B are blocked by S.

It can be proved that S d-separates A from B if and only if S separates A from B in the moral Graph induced by the smallest ancestral set containing $A \cup B \cup S$ and therefore that the directed global Markov property can be equivalently expressed in terms of d-separation.

2.2.4 Directed Gaussian Graphical Models

Directed Gaussian Graphical Models were first formalized by Shachter and Kenley (1989), noting that *the influence diagram graphical structure*, this is how they used to call a DAG, represents a particular representation of the joint distribution into conditional distributions. In fact a q-variate normal distribution $X_V = \mathcal{N}_q(\mu, \Sigma)$, given an ordering of the nodes, always admits a factorization of the type:

$$p_V(x_V) = \prod_{j=1}^{q} p_{j|\mathrm{pa}(j)}(x_j \mid x_{\mathrm{pa}(j)}) \tag{2.5}$$

where $X_i \mid \mathrm{pa}(X_i)$ are univariate normal distributions with mean

$$\mu_j + \Sigma_{j,\mathrm{pa}(j)}\Sigma^{-1}_{\mathrm{pa}(j),\mathrm{pa}(j)}(X_{\mathrm{pa}(j)} - \mu_{pa(j)})$$

and variance

$$\sigma_{jj} - \Sigma_{j,\mathrm{pa}(j)}\Sigma^{-1}_{\mathrm{pa}(j),\mathrm{pa}(j)}\Sigma_{\mathrm{pa}(j),j}$$

We note that in this formulation some conditional independence statements determine the conditioning sets for each variable. Furthermore an interesting fact is that the conditional models are independent and therefore we can think our model as a set of separate multivariate regressions. This highly simplifies our problem and puts us in an efficient conditional independence framework for computation.

Let $\mathcal{D}$ be the DAG that connects the children to the parents according to (2.5), then clearly X_V factorizes w.r.t. to $\mathcal{D} = (V, E)$ and it is therefore Markov with respect to $\mathcal{D}$. Anyway it is still interesting to note that how this factorization follows the ordered Markov property (O). Let indicate with $\Sigma_{A,B}$ the submatrix of Σ relative to the vectors X_A and X_B. Then the vector

$$b_j = \Sigma_{j,\mathrm{pa}(j)}\Sigma^{-1}_{\mathrm{pa}(j),\mathrm{pa}(j)}$$

is the vector of the regression coefficients for the jth regression. If we think each variable regressed against all its *predecessors* in a well-ordering, then we can think the vector b_j as a vector which states a conditional independence between i and j (given the parents of j) when the ith entry of b_j is zero, which follows by decomposition (C2) from (O). Therefore if we summarize all our regression coefficients in a matrix B, where per column we put our b_j relative to all the predecessors, and fill the rest of the vector with zeros, we get an upper triangular matrix B which will have the same zero-entries as the *adjacency matrix* induced by $\mathcal{D}$.

Since B is upper triangular $I - B$ can be inverted, so let $U = (I - B)^{-1}$ then the covariance matrix can be found as

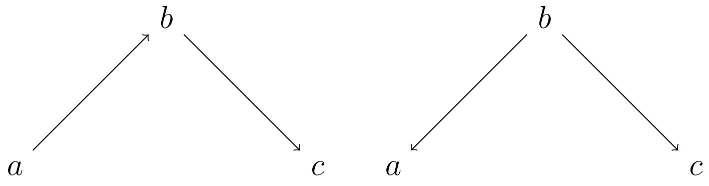
$$\Sigma = U^\top D U \tag{2.6}$$

with D a diagonal matrix with entries the conditional variances.

2.3 Markov equivalence

We say that two graphs are Markov equivalent if they encode the same set of conditional independecies. In the directed case we can have graphs with different structure which are Markov equivalent:

Example 3 Given 3 vertices this two graph are Markov equivalent



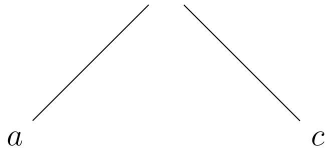
The first graph admits a recursive factorization of the form:

$$p(a, b, c) = p(a)p(b|a)p(c|b) = p(a, b)p(c|b)$$

and the second

$$p(a, b, c) = p(b)p(a|b)p(c|b) = p(a, b)p(c|b)$$

Therefore both include only the conditional independence statement $c \perp\!\!\!\perp a \mid b$ which can be red also from the moral graph



□

If we think about the directed global Markov property then we notice that if two DAGs have the same *immoralities*, i.e. unmarried parents with a common child, and and the same undirected version then their moral graph is exactly the same, and thus they will encode the same set of conditional independencies, in effect it can be proved that two DAGs are Markov equivalent if and only if they have the same undirected version, often called *skeleton*, and the same immoralities.

In contrast, in the undirected case two graphs are Markov equivalent if and only if they are identical, but there still exists undirected graphs which are Markov equivalent to direct graphs. Infact a DAG $\mathcal{D}$ and an UG $\mathcal{G}$ are Markov equivalent if and only if the DAG is *perfect*, i.e. has no immoralities, and the skeleton of the DAG is equal to $\mathcal{G}$. Moreover we can see that the skeleton of a perfect DAG is decomposable and that from every decomposable UG directing the edges from lower to higher positions in a perfect numbering we get a perfect DAG, therefore the family of perfect DAG models coincides with the family of decomposable undirected models.

Chapter 3

Bayes Factor

Bayes factors can be seen as the Bayesian analog of the classical tests of hypotheses. In fact Bayes factors summarizes the evidence provided by the data in favor of a statistical model opposed to another. While their formal definition refers to just two models, thanks to their properties they are often used for model selection problems. In this chapter we will review the basic properties of Bayes factors with particular attention to the case of objective Bayesian analysis.

3.1 Bayes factor and posterior model probabilities

Definition 29 (Posterior probability of a Model M) Let M be a model and D be a set of data then the *posterior probability* of the model M given the data D is as follow:

$$\mathbb{P}(M|D) = \frac{\mathbb{P}(D|M)\mathbb{P}(M)}{\mathbb{P}(D)}$$

□

Definition 30 (Marginal Likelihood) The integral of the *likelihood* over all possible values of θ:

$$m(D) = \mathbb{P}(D|M) = \int \mathbb{P}(D|\theta, M)\mathbb{P}(\theta|M)\,\mathrm{d}\,\theta$$

is defined as the *marginal likelihood.* □

Let M_1 and M_2 be two different generating models for the same set of data D. If we want to evaluate which model is more probable *a posteriori* we might consider the ratio:

$$\frac{\mathbb{P}(M_1|D)}{\mathbb{P}(M_2|D)} = \frac{\mathbb{P}(M_1)\mathbb{P}(D|M_1)}{\mathbb{P}(M_2)\mathbb{P}(D|M_2)} = \frac{\mathbb{P}(M_1)m_1(D)}{\mathbb{P}(M_2)m_2(D)}$$

If two models are equally probable *a priori* (which is a common assumption) the ratio simplifies to the Bayes Factor

Definition 31 (Bayes Factor (BF)) We define Bayes Factor the quantity:

$$B_{ij} = \frac{\mathbb{P}(D|M_1)}{\mathbb{P}(D|M_2)} = \frac{m_1(D)}{m_2(D)} = \frac{\int \mathbb{P}(D|\theta_1, M_1)\mathbb{P}(\theta_1|M_1)\,\mathrm{d}\,\theta_1}{\int \mathbb{P}(D|\theta_2, M_2)\mathbb{P}(\theta_2|M_2)\,\mathrm{d}\,\theta_2}$$

which is a ratio of *marginal likelihoods* □

Let H_i denote a hypothesis and E evidence. To compare hypotheses *post-experimentally*, we often calculate the posterior odds:

$$\begin{aligned}\frac{\Pr(H_i \mid E)}{\Pr(H_j \mid E)} &= \frac{\Pr(E \mid H_i)Pr(H_i)/Pr(E)}{\Pr(E \mid H_j)Pr(H_j)/Pr(E)} \\ &= \frac{\Pr(E \mid H_i)Pr(H_i)}{\Pr(E \mid H_j)Pr(H_j)} \\ &= \frac{\Pr(E \mid H_i)}{\Pr(E \mid H_j)} \times \frac{\Pr(H_i)}{\Pr(H_j)} \\ &= \text{Bayes factor} \times \text{prior odds}\end{aligned}$$

Note that the Bayes factor is the same as

$$\frac{\mathbb{P}(H_i \mid E)}{\mathbb{P}(H_j \mid E)} \Big/ \frac{\mathbb{P}(H_i)}{\mathbb{P}(H_j)}$$

and that it does not depend on the priors on the hypothesis on H_i's. Note that in the Bayes factor definition given above hypothesis were prior probabilities on the models.

An useful result that joins the Bayes factor with the model posterior probabilities is the following: suppose that we are comparing q models $(M_1, \ldots, M_q)$ if the prior probabilities $\mathbb{P}(M_j)$ are available for each model, then one can compute the *model posterior probabilities* from the Bayes factors

$$P(M_i|D) = \frac{P(D|M_i)P(M_i)}{\sum_{j=1}^{q} P(D|M_j)P(M_j)} = \left[\sum \frac{\mathbb{P}(M_j)}{\mathbb{P}(M_i)} B_{ji}\right]^{-1} \tag{3.1}$$

Definition 32 (Renormalized marginal probabilities) We define *renormalized marginal probabilities* as:

$$\bar{m}_i = \frac{m_i}{\sum_{j=1}^{q} m_j} = \left(\sum_j B_{ji} \right)^{-1} \tag{3.2}$$

□

We note from (3.1) that if $\mathbb{P}(M_i) = 1/q$ for all $i = 1, \dots, q$ then the posterior probabilities are the same as the *renormalized marginal probabilities.* Further note that there is a one to one relationship between Bayes factors and renormalized marginal probabilities since beside (3.2) it also holds that

$$\frac{\bar{m}_j}{\bar{m}_i} = \frac{\frac{m_j(D)}{\sum_{j=1}^{q} m_j(D)}}{\frac{m_i(D)}{\sum_{j=1}^{q} m_j(D)}} = \frac{m_j(D)}{m_i(D)} = B_{ji}$$

In effect reporting the renormalized marginal probabilities (or equivalently Bayes factors) beside/rather then the posterior probabilities is good practice since anyone, through (3.1), can use the renormalized marginal probabilities to determine their personal posterior probabilities choosing their favorite prior probabilities.

Proposition 2 (Bayes factor composition)

$$B_{ij} = B_{ik} B_{kj} \tag{3.3}$$

Proof

$$B_{ik} B_{kj} = \frac{m_i}{m_k} \frac{m_k}{m_j} = \frac{m_i}{m_j} = B_{ij}$$

■

Definition 33 (Bayes factor chain rule) Applying the previous proposition iteratively we can see that given $i + 1$ models index by $s = 0, \dots, i$

$$B_{i0} = \prod_{s=1}^{i} B_{s,s-1}$$

and this is called the *Bayes factor chain rule* □

3.1.1 Encompassing

In some situations it may occur that the prior elicitation requires the BF to be calculated between nested models, in those cases we can exploit the *encompassing approach* (Cox, 1961) to get any other Bayes factor. Suppose we have a model M_0 that includes all other models, i.e. the *encompassing model*, we can then compute the Bayes factors for the encompassing model against any other and thanks to the Bayes factor composition (3.3) we can obtain Bayes factors also for non-nested models, e.g. let model M_i and model M_j be two non nested models then

$$B_{ij} = B_{0j}/B_{0i}$$

The encompassing approach can also be used to get any desired model posterior probability, let M_0 be the reference model, we can reformulate (3.1) as

$$\begin{aligned} P(M_i \mid D) &= \frac{m_i(x)\mathbb{P}(M_i)}{\sum_{j=1}^{q} m_j(x)\mathbb{P}(M_j)} \\ &= \frac{\frac{m_i(x)}{m_0(x)}\mathbb{P}(M_i)}{\sum_{j=1}^{q} \frac{m_j(x)}{m_0(x)}\mathbb{P}(M_j)} \\ &= \frac{B_{i0}\mathbb{P}(M_i)}{\sum_{j=1}^{q} B_{j0}\mathbb{P}(M_j)} \end{aligned} \tag{3.4}$$

Clearly this approach can be also pursued when the reference model is a model that is included in every other model, in this case we call the approach *encompassing from below.*

3.2 Objective Bayes in model selection

The motivation for using objective Bayesian methods for variable selection is that, in a subjective framework, the specification of all prior distributions for all models is a huge work, and the usually limited expert time available one would typically use it for model formulation and, possibly, prior elicitation for the model that is ultimately selected. That said, following the objective Bayes approach involves also some problems. For a complete review of such problems refer to Berger *et al.* (2001), the major ones are

1. we have seen that to compute the BFs we need to integrate out the parameters and the computation of the integrals can be hard, moreover

doing it for all models (too many) may be not feasible in a reasonable time span.

2. improper priors yields indeterminate BFs (when the parameter space has different dimension)
3. vague priors give bad answers, i.e. the Bayes factor will always strongly depend on the vague prior even when n is large.
4. in different models the same parameters may have different interpretation and the prior has to change in a corresponding fashion.

To address at least some of these issues we can use methodologies derived from the partial Bayes factor presented in the next section.

3.2.1 Partial Bayes factor

To introduce the partial Bayes factor we need the concept of *proper minimal training sample*. Let $M_1, \ldots, M_q$ be q models and x data. Under model M_i let the data be related to parameters θ_i by a distribution $f(x|\theta_i)$ and let $p_i^N(\theta_i)$, for $i = 1, \ldots, q$, be noninformative priors for θ_i and $m_i^N(x)$ the corresponding marginal likelihoods. Finally Let x be partitioned in to sets $x = (x(l), x(-l))$ with $x(l)$ a training sample.

Definition 34 (proper minimal training sample) A training sample $x(l)$, is called *proper* if $0 < m_i^N(x(l)) < \infty$ for all M_i, and *minimal* if it is proper and no subset is proper. □

Definition 35 (partial Bayes factor) Given a proper minimal training sample $x(l)$ we can define a proper (conditional) prior

$$p_i^N(\theta_i|x(l)) = \frac{f_i(x(l)|\theta_i)p_i^N(\theta_i)}{m_i^N(x(l))} \tag{3.5}$$

□

Then the *partial Bayes factor* is the Bayes factor on the remaining dataset, $x(-l)$, conditioned on having observed $x(l)$

$$B_{ji}(l) = \frac{m_j^N\left(x(-l)|x(l)\right)}{m_i^N\left(x(-l)|x(l)\right)}$$

Focusing on the numerator the *partial marginal likelihood* results:

$$\begin{aligned} m_j^N\left(x(-l) \mid x(l)\right) &= \int f_j(x(-l) \mid x(l), \theta_j) p_j^N(\theta_j \mid x(l)) \,\mathrm{d}\,\theta_j \\ &= \int \frac{f(x(-l)), x(l) \mid \theta_j)}{f_j(x(l) \mid \theta_j)} \frac{f_j(x(l) \mid \theta_j) p_j^N(\theta_j)}{m_j^N(x(l))} \,\mathrm{d}\,\theta_j \\ &= \int \frac{f(x \mid \theta_j) p_j^N(\theta_j)}{m_j^N(x(l))} \,\mathrm{d}\,\theta_j \\ &= \frac{m_j^N(x)}{m_j^N(x(l))} \end{aligned} \tag{3.6}$$

and therefore partial Bayes factor can be also expressed as

$$B_{ji}(l) = \frac{m_j^N(x)}{m_j^N(x(l))} \frac{m_i^N(x(l))}{m_i^N(x)} = B_{ji}^N(x) B_{ij}^N(x(l)) \tag{3.7}$$

Note that if the noninformative prior has an improper form then the indeterminate constant cancels out.

3.2.2 Intrinsic Bayes factor

We have just seen that the partial Bayes factor for comparing model M_j with M_i no longer depends on the scales of the noninformative priors p_j^N and p_i^N, but it still depends on the arbitrary choice of the minimal training sample $x(l)$. To eliminate this dependence and to increase stability, Berger and Pericchi (1996) proposed the Intrinsic Bayes factor.

Definition 36 (Intrinsic Bayes factor (IBF)) Let $B_{ji}(l)$ be the partial Bayes factor in favor of Model M_j versus model M_i relative to a training sample $x(l)$. If we "average" the $B_{ji}(l)$ over all possible training sample $x(l)$, $l = 1, \ldots, L$, we obtain the *arithmetic IBF* (AIBF) or the *median IBF* (MIBF):

$$B_{ji}^{AI} = B_{ji}^N(x) \frac{1}{L} \sum_{l=1}^{L} B_{ij}^N(x(l)) \qquad B_{ji}^{MI} = B_{ji}^N(x) \mathrm{Med}[B_{ij}^N(x(l))]$$

□

Note that IBFs are *resampling summaries* of the evidence of the data for the comparison of models.

3.2.3 Fractional Bayes factor

The fractional Bayes factor follows the same "philosophy" of the partial Bayes factor. So, suppose that the observed dataset x is partitioned in two datasets, let say $x(l)$ of size m and $x(-l)$ of size $n-m$.[1] Let $b = m/n$ be the *fraction* of the data of x contained in $x(l)$, if both m and n are large, the likelihood $f(x(l)|\theta_i)$ based only on the training sample $x(l)$ will approximate to the full likelihood $f(x|\theta_i)$ raised to the power b.[2]

Definition 37 (Fractional prior) As we did in (3.5) we use the trick of creating a proper (conditional) prior, and exploiting the approximation readily introduced we can define $p^F(\theta_i \mid x)$

$$\begin{aligned} p(\theta_i \mid x(l)) &= \frac{f(x(l) \mid \theta_i) p_i^N(\theta_i)}{\int f(x(l) \mid \theta_i) p_i^N(\theta_i) \,\mathrm{d}\,\theta_i} \\ &\approx \frac{f(x \mid \theta_i)^b p_i^N(\theta_i)}{\int f(x \mid \theta_i)^b p_i^N(\theta_i) \,\mathrm{d}\,\theta_i} \\ &= p^F(\theta_i \mid x) \end{aligned}$$

the so called *fractional prior*. □

Definition 38 (Fractional Bayes factor) Remembering that the *partial Bayes factor* is the Bayes factor on the remaining dataset, namely $x(-l)$, conditioned on having observed $x(l)$, we can simply exploit the previously introduced approximation and redo the same steps done for the partial Bayes factor to obtain the *fractional Bayes factor*

$$B_{ji}^F = \frac{m_j^F(x)}{m_i^F(x)} = B_{ji}^N(x) \frac{m_{i,b}^N(x)}{m_{j,b}^N(x)} \tag{3.8}$$

[1] it will become clearer in a minute that we don't even need to choose $x(l)$.

[2] since already with m simple random samples we are considering a lot of information from the generating distribution, considering the likelihood on $x(l)$ the remaining contribution (to reach x) will be similar to the average contribution already considered with $x(l)$

$$f_i(x(-l)|\theta_i) \approx f_i(x(l)|\theta_i)^{\frac{n-m}{m}}$$

then

$$f_i(x|\theta_i) = f_i(x(l)|\theta_i) f_i(x(-l)|\theta_i) \approx f_i(x(l)|\theta_i) f_i(x(l)|\theta_i)^{\frac{n-m}{m}} = f_i(x(l)|\theta_i)^{\frac{1}{b}}$$

which is equal to

$$f_i(x|\theta_i)^b \approx f_i(x(l)|\theta_i)$$

Proof In detail the *fractional marginal likelihood* results:

$$\begin{aligned} m_j^F(x) &= \int f(x \mid \theta_j)^{1-b} p^F(\theta_j \mid x) \,\mathrm{d}\,\theta_j \\ &= \int f(x \mid \theta_j)^{1-b} \frac{f(x \mid \theta_j)^b p_j^N(\theta_j)}{\int f(x \mid \theta_j)^b p_j^N(\theta_j) \,\mathrm{d}\,\theta_j} \,\mathrm{d}\,\theta_j \\ &= \frac{m_j^N(x)}{m_{j,b}^N(x)} \end{aligned}$$

with

$$m_{j,b}^N(x) = \int f(x \mid \theta_j)^b p_j^N(\theta_j) \,\mathrm{d}\,\theta_j$$

and finally, as usual, the ratio of the marginal likelihoods gives us the Bayes factor, here the so called *fractional Bayes factor*. ■

Note that (3.8) does not depend anymore from the dataset $x(l)$, and so there is no arbitrariness on the choice of the sampling dataset as we had with the partial Bayes factor but that it inherits the fact that if the noninformative prior has an improper form then the indeterminate constant cancels out.

The fractional prior $p^F(\theta_i|x) \propto f(x|\theta_i)^b p_i^N(\theta_i)$ clearly depends on the observed dataset x, intuitively to make the dependence weak b should be small, in fact consistency of the FBF is achieved as long as $b \to 0$ for $n \to \infty$. One common choice of b, suggested by O'Hagan (1995), is $b = m/n$ where m is the "minimal training sample size" as defined above.

Example 4 (FBF for the Normal Linear Model) Let M_j be defined by

$$M_j : f(y|X;\beta_j,\gamma_j) = \left(\frac{\gamma_j}{2\pi}\right)^{n/2} \exp\left(-\frac{\gamma_j}{2}\|y - X\beta_j\|^2\right)$$ □

A default noninformative prior for (β_j, γ_j) is proportional to $\frac{1}{\gamma_j}$ (i.e. reference prior), so that the marginal fractional likelihood for model j is given by:

$$\begin{aligned} m_j^F(y) &= \frac{m_j^N(y)}{m_{j,b}^N(y)} \\ &= \frac{\int\int f(y \mid X;\beta,\gamma) p^N(\beta,\gamma) \,\mathrm{d}\,\beta \,\mathrm{d}\,\gamma}{\int\int f(y \mid X;\beta,\gamma)^b p^N(\beta,\gamma) \,\mathrm{d}\,\beta \,\mathrm{d}\,\gamma} \end{aligned}$$

lets concentrate on $m^N_{j,b}(y)$, suppressing the model subscript to simplify the notation it can be rewritten as:

$$\begin{aligned}
m^N_b(y) &= \int\int f(y \mid X; \beta, \gamma)^b p^N(\beta, \gamma)\, \mathrm{d}\,\beta\, \mathrm{d}\,\gamma \\
&= \int\int \left(\frac{\gamma}{2\pi}\right)^{bn/2} \exp\left(-\frac{b\gamma}{2}\|y - X\beta\|^2\right) \frac{1}{\gamma}\, \mathrm{d}\,\beta\, \mathrm{d}\,\gamma \\
&= \int\int \left(\frac{1}{2\pi}\right)^{bn/2} \gamma^{bn/2-1} \exp\left(-\frac{b\gamma}{2}\|y - X\beta\|^2\right) \mathrm{d}\,\beta\, \mathrm{d}\,\gamma \qquad (3.9) \\
&= \int \left(\frac{1}{2\pi}\right)^{bn/2} \gamma^{bn/2-1} \exp\left(-\frac{b\gamma}{2}\|y - \right. \\
&\left. X\hat{\beta}\|^2\right) \int \exp\left(-\frac{b\gamma}{2}\|X(\beta - \hat{\beta})\|\right) \mathrm{d}\,\beta\, \mathrm{d}\,\gamma
\end{aligned}$$

since

$$\begin{aligned}
\|y - X\beta\|^2 &= \|y - X\hat{\beta} + X\hat{\beta} - X\beta\|^2 \\
&= \|y - X\hat{\beta}\|^2 + \|X\hat{\beta} - X\beta\|^2 + 2(y - X\hat{\beta})^\top (X\beta - X\hat{\beta}) \\
&= \|y - X\hat{\beta}\|^2 + \|X\hat{\beta} - X\beta\|^2 + 2(y - Hy)^\top (X\beta - Hy) \\
&= \|y - X\hat{\beta}\|^2 + \|X\hat{\beta} - X\beta\|^2 \\
&\quad + 2(y^\top X\beta - y^\top Hy - y^\top H^\top X\beta + y^\top H^\top Hy \\
&= \|y - X\hat{\beta}\|^2 + \|X\hat{\beta} - X\beta\|^2
\end{aligned}$$

We recognize in the integral in β the kernel of a multivariate normal distribution $\mathcal{N}_p\left(\hat{\beta}, \left[b\gamma(X^\top X)\right]^{-1}\right)$ so that (3.9) becomes:

$$\begin{aligned}
m^N_b(y) &= \int \left(\frac{1}{2\pi}\right)^{bn/2} \gamma^{bn/2-1} \exp\left(-\frac{b\gamma}{2}\|y\right. \\
&\quad \left. - X\hat{\beta}\|^2\right) \left(\frac{2\pi}{b\gamma}\right)^{p/2} \mid (X^\top X) \mid^{-1/2} \mathrm{d}\,\gamma \\
&= K \int \gamma^{\frac{bn-p}{2}-1} \exp\left(-\gamma\frac{b}{2}\|y - X\hat{\beta}\|^2\right) \mathrm{d}\,\gamma \\
&= K \int \gamma^{\frac{bn-p}{2}-1} \exp\left(-\gamma\frac{b}{2}R\right) \mathrm{d}\,\gamma
\end{aligned}$$

with $K = \left(\frac{1}{2\pi}\right)^{bn/2}\left(\frac{2\pi}{b}\right)^{p/2} \mid (X^\top X) \mid^{-1/2}$ and $R = \|y - X\hat{\beta}\|^2$. Thus we recognize a Gamma distribution and $m_b^N(y)$ becomes when $b > \frac{p}{n}$

$$\begin{aligned} m_b^N(y) &= K\Gamma\left(\frac{bn-p}{2}\right)\left(\frac{b}{2}R\right)^{-\frac{bn-p}{2}} \\ &= \left(\frac{1}{2\pi}\right)^{bn/2}\left(\frac{2\pi}{b}\right)^{p/2} \mid (X^\top X) \mid^{-1/2} \Gamma\left(\frac{bn-p}{2}\right)\left(\frac{b}{2}R^2\right)^{-\frac{bn-p}{2}} \\ &= \pi^{\frac{p-bn}{2}} b^{\frac{-bn}{2}} \mid X^\top X \mid^{-\frac{1}{2}} \Gamma\left(\frac{bn-p}{2}\right) R^{-\frac{bn-p}{2}} \end{aligned}$$

and this the fractional marginal likelihood is

$$\begin{aligned} m^F(y) &= \frac{m^N(y)}{m_b^N(y)} \\ &= \frac{\pi^{-\frac{n}{2}} b^{-\frac{n}{2}} \Gamma\left(\frac{n-p}{2}\right) R^{-\frac{n-p}{2}}}{\pi^{-\frac{bn}{2}} b^{-\frac{bn}{2}} \Gamma\left(\frac{bn-p}{2}\right) R^{-\frac{bn-p}{2}}} \\ &= \pi^{-\frac{n(1-b)}{2}} b^{-\frac{n(1-b)}{2}} \frac{\Gamma\left(\frac{n-p}{2}\right)}{\Gamma\left(\frac{bn-p}{2}\right)} R^{-\frac{n(1-b)}{2}} \\ &= \frac{\Gamma\left(\frac{n-p}{2}\right)}{\Gamma\left(\frac{bn-p}{2}\right)} (b\pi)^{-\frac{n(1-b)}{2}} R^{-\frac{n(1-b)}{2}} \end{aligned}$$

Finally the fractional Bayes factor in favor of M_j against M_i results

$$\begin{aligned} B_{ji}^F &= \frac{m_j^F(y)}{m_i^F(y)} \\ &= \frac{\frac{\Gamma\left(\frac{n-p_j}{2}\right)}{\Gamma\left(\frac{bn-p_j}{2}\right)} (b\pi)^{-\frac{n(1-b)}{2}} R_j^{-\frac{n(1-b)}{2}}}{\frac{\Gamma\left(\frac{n-p_i}{2}\right)}{\Gamma\left(\frac{bn-p_i}{2}\right)} (b\pi)^{-\frac{n(1-b)}{2}} R_i^{-\frac{n(1-b)}{2}}} \\ &= \frac{\Gamma\left(\frac{n-p_j}{2}\right)\Gamma\left(\frac{bn-p_i}{2}\right)}{\Gamma\left(\frac{n-p_i}{2}\right)\Gamma\left(\frac{bn-p_j}{2}\right)} \left(\frac{R_i}{R_j}\right)^{\frac{n(1-b)}{2}} \end{aligned}$$

Chapter 4

Non-local Priors

4.1 Non-local priors for hypothesis testing and model selection

In a parametric setting, classical hypothesis tests about a parameter of interests θ are usually posted as

$$H_0 : \theta \in \Theta_0$$
$$H_1 : \theta \in \Theta_1$$

where Θ_0 and Θ_1 are disjoint parameter sets. In the Bayesian paradigm it is required to specify prior distribution on θ under each hypothesis, and in most Bayesian hypothesis tests local alternative prior densities are used, i.e. prior densities that are positive on Θ_0. On a philosophical level this seems to be a contradiction and moreover it can be proved that Bayes factors with local alternative priors have a strong unbalanced learning rate behavior, e.g. for a scalar valued parameter and a point null hypothesis, if H_1 is true, the Bayes factor in favor of the null hypothesis decreases exponentially fast, while if H_0 is true, the Bayes factor in favor of the alternative hypothesis decreases only at rate $O_p(n^{-1/2})$.

Johnson and Rossell (2010) highlighted that priors that separate between the null and alternative hypotheses can improve this convergence rates and mitigate the unbalance.

Definition 39 (non-local prior) We say $p(\theta)$ is a *non-local (alternative) prior* density, if for every $\varepsilon > 0$ there is a $\zeta > 0$ such that

$$p(\theta) < \varepsilon \quad \forall \theta \in \Theta : \inf_{\theta_0 \in \Theta_0} |\theta - \theta_0| < \zeta \tag{4.1}$$

□

An early proposal of non-local priors can be found in Verdinelli and Wasserman (1996) and Rousseau (2007). They propose priors which are defined to be 0 for all θ in a neighborhood of Θ_0. For example, focusing on point null hypothesis, with respect to a scalar parameter θ, these type of priors are defined to be zero in an interval $(\theta_0 - \varepsilon, \theta_0 + \varepsilon)$ for some $\varepsilon > 0$. Johnson and Rossell (2010) state that the problem about these type of priors is that they do not provide "flexibility in the specification of the rate at which 0 is approached at parameter values that are consistent with the null hypothesis" and lack of a "mechnism for rejecting H_0 for values of θ outside but near Θ_0". Therefore they propose *moment priors*.

Definition 40 (moment priors) Let focus on *a point null hypothesis and a scalar parameter*, given a base prior $p_b(\theta)$, with $2k$ finite integer moments, $k > 0$, two bounded derivatives in a neighborhood containing θ_0 and such that $p_b(\theta_0) > 0$, then the *kth moment prior (MOM)* is defined as

$$p_M(\theta) = \frac{(\theta - \theta_0)^{2k}}{\tau_k} p_b(\theta)$$

with τ_k normalizing constant

$$\tau_k = \int (\theta - \theta_0)^{2k} p_b(\theta) \,\mathrm{d}\,\theta = \mathbb{E}_{p_b}[(\theta - \theta_0)^{2k}]$$

ensuring p_M to be a proper density. □

The moment prior is non-local satisfying (4.1), but assigns mass to all θs in Θ_1, and through the choice of the base prior and k it can be controlled the rate at which it approaches zero. It also interesting to note that for $k = 0$ the moment prior reduces to the base prior.

Proposition 3 *The convergence rate of the Bayes factor in favor of the false alternative hypothesis when the alternative model is specified by the kth moment prior is*

$$B_{10} = O_p(n^{-k-1/2})$$

□

The extra power k, this class of priors gives to the Bayes factor with respect to local priors, means that, for instance, if $k = 1$ the rate changes from sublinear to superlinear.

Moment priors can be naturally generalized to the multivariate case as

$$p_M(\theta) = \frac{Q(\theta)^k}{\mathbb{E}_{p_b}[Q(\theta)^k]} p_b(\theta)$$

where

$$Q(\theta) = \frac{(\theta - \theta_0)^\top \Sigma^{-1} (\theta - \theta_0)}{n\tau\sigma^2}$$

with Σ a positive definite matrix and τ and σ^2 scalars. For the choice of τ the authors suggest to choose τ so that the prior probability that a standardized effect size is less then 0.2 is less than 0.05, see section A.2 to learn more about τ.

Following this work in JOHNSON and ROSSELL (2012) they propose the *product moment prior (pMOM)*, suited for model selection in high dimensional normal linear models. This prior arises as the independent products of the univariate MOM prior densities, and is proper. The main difference is that while the multivariate MOM density is 0 only when all components of the parameter vector are 0, pMOM instead is 0 if any of the components is zero. This is important since it introduces a greater penalty for models that have any of the components close to 0, and makes this prior appropriate for variable selection where sparse models are usually preferred.

Finally, for identifying high probability models, they propose a MCMC algorithm to explore the model space based on a Laplace approximation to approximate the marginal likelihood of the data under each model.

4.2 Non-local priors for graphical model choice

Following the work of JOHNSON and ROSSELL (2010) CONSONNI and LA ROCCA (2011) derived a FBF for pairwise comparison of nested Gaussian DAGs in an objective framework. Let (V, D) be a DAG, the joint density of $U_1, \ldots, U_q$ can be written like in (2.5) as

$$p(u_1, \ldots, u_q \mid \beta, \gamma) = \prod_{j=1}^{q} p(u_j \mid u_{\mathrm{pa}(j)}; \beta_j, \gamma_j)$$

Because of the recursive structure of the likelihood it is natural to assume global parameter independence, see GEIGER and HECKERMAN (2002)

$$p(\beta, \gamma) = \prod_j p(\beta_j, \gamma_j)$$

and therefore a natural default prior is

$$p_b(\beta_j, \gamma_j) \propto \gamma_j^{-1} \tag{4.2}$$

Since they focus on the pairwise comparison of two nested models, say $\mathcal{D}_0 \subset \mathcal{D}_1$, they define L_j, for each vertex j, as the set of the edges pointing to j which are present in $\mathcal{D}_1$ but not in $\mathcal{D}_0$. Clearly setting β_{jl} to zero for each j and each $l \in L_j$ brings $\mathcal{D}_1$ to $\mathcal{D}_0$, and exploiting the default prior (4.2) as base prior they derive the (product) moment prior of order h for vertex j.

$$p(\beta_j, \gamma_j) \propto \gamma_j^{-1} \prod_{l \in L_j} \beta_{jl}^{2h} \tag{4.3}$$

Multiplying together these priors one obtains a (product) moment prior for comparing Gaussian DAGs

$$p(\beta, \gamma) \propto \prod_{j=1}^{q} \left\{ \gamma_j^{-1} \prod_{l \in L_j} \beta_{jl}^{2h} \right\} \tag{4.4}$$

To derive the FBF in closed form the main difficulty arises in the fractional marginal likelihood under the alternative hypothesis, that because an expectation of the product of even powers of the parameters in L_j under a multivariate Gaussian distribution. But it can be proved that

Lemma 1 *Let $U = (U_1, \ldots, U_q) \sim \mathcal{N}_q(\mu, \Sigma)$ where $\mu = (\mu_1, \ldots, \mu_q)^\top$ and $\Sigma = \{\sigma_{lj}\}$. Fix $d \leq q$ and a positive integer h, then*

$$\mathbb{E}\left[\prod_{l=1}^{d} U_l^{2h}\right] = \sum_{i=0}^{hd} \frac{1}{2^i} H_i^{(h)}(\mu, \Sigma) \tag{4.5}$$

where

$$H_i^{(h)}(\mu, \Sigma) = \sum_{j \in J_h(i)} \prod_{l=1}^{d} (2h)! \prod_{m=1}^{d} \frac{\sigma_{lm}^{j_{lm}}}{j_{lm}!} \prod_{l=1}^{d} \frac{\mu_l^{j_l^*}}{j_l^*!}$$

having defined

$$j_l^* = 2h - \sum_{m=1}^{d} j_{lm} - \sum_{m=1}^{d} j_{ml}$$

and

$$J_h(i) = \left\{ j : \sum_{l=1}^{d} \sum_{m=1}^{d} j_{lm} = i \wedge \forall l : j_l^* \geq 0 \right\}$$

□

Then focusing on vertex j, we omit index j for notational convenience, let X be a $n \times p$ matrix whose columns contain the observations on the parent variables, then the fractional marginal likelihood based on (4.3) is

$$w_h(y|X,b) = (\pi b S^2)^{-\frac{n(1-b)}{2}} \frac{\sum_{i=0}^{h|L|} 4^{-i} H_i^{(h)}(\hat{\beta}, (X^\top X)^{-1}) \Gamma\left(\frac{n-p-2i}{2}\right) (S^2)^i}{\sum_{i=0}^{h|L|} 4^{-i} H_i^{(h)}(\hat{\beta}, (X^\top X)^{-1}) \Gamma\left(\frac{nb-p-2i}{2}\right) (S^2)^i} \tag{4.6}$$

where $0 < b < 1$ is the sample size dependent fraction satisfying $nb > p + 2h|L|$, $\hat{\beta} = (X^\top X)^{-1} X^\top y$ and $S^2 = (y - X\hat{\beta})^\top (y - X\hat{\beta})$.

For $h = 0$ (4.6) clearly gives the fractional marginal likelihood under the null hypothesis, which is nothing else than the ordinary marginal fractional likelihood under the default prior, see FBF for the Normal Linear Model 4. Therefore the Moment FBF (MFBF)

$$FBF_{10}(y,b) = \frac{\prod_{j=1}^{q} w_0(y_j|X_{1j}, b)}{\prod_{j=1}^{q} w_1(y_j|X_{0j}, b)} \tag{4.7}$$

In a following work Altomare *et al.* (2013), assuming an ordering of the variables, developed an algorithm for model search in the space of DAGs based on the MFBF (4.7). The algorithm exploits the encompassing approach (see subsection 3.1.1) to calculate the model posterior probabilities. Using as reference model the complete independence DAG $\mathcal{D}_0$, which is clearly included in every DAG, and approximating the normalizing constant with an indexed collection $\mathcal{H}$ of high scoring DAGs

$$\hat{p}(\mathcal{D}_k \mid y) = \frac{B_{k0}(y) p(\mathcal{D}_k)}{\sum_{j \in \mathcal{H}} B_{j0}(y) p(\mathcal{D}_j)}$$

These estimated posterior probabilities allow, among other things, also to estimate the *posterior edge inclusion probabilities* as

$$\hat{p}(e|y) = \sum_{j \in \mathcal{H} : e \in D_j} \hat{p}(D_j|y)$$

To complete the model they propose as model prior, adapting the approach of Scott and Berger (2010), a prior that they call *split prior.* The name is due to the fact that also the model prior is thought vertex-wise. Let $\mathcal{D}_k$ be a DAG and denote with $\mathcal{M}_2, \ldots, \mathcal{M}_q$ the corresponding regression models, then

$$p(\mathcal{D}_k) = \prod_{j=2}^{q} p(\mathcal{M}_j) = \prod_{j=2}^{q} \frac{1}{j} \left(\begin{array}{c} j-1 \\ |\text{pa}(j)| \end{array} \right)^{-1}$$

This prior, in conjunction with the factorization of the MFBF, allows the computation the be done locally, i.e. the algorithm can be run separately for each vertex, opening the doors to parallelization to further improve the computation speed.

Although from (4.7) we can see that the MFBF for a model that is far apart from the base DAG involves a lot of computation, therefore to speed up the algorithm the authors propose to use the Bayes factor chain rule considering pairwise comparison between adjacent models, i.e. DAGs differing by exactly one edge. In the context of MFBF this is just an approximation but this highly simplifies the computation.

The search algorithm is similar to the algorithms proposed from Berger and Molina (2005) and Scott and Carvalho (2008). The rationale behind is easy: edge moves which already improved some models are likely to improve other models too. Given a base DAG $\mathcal{D}_0 = (\mathcal{M}_1^{(0)}, \dots, \mathcal{M}_q^{(0)})$, and the number of models we want to explore for each regression (`n_mod`), we can summarize the algorithm like

Algorithm 1 Local stochastic search

```
for j=2;j<=q;j++ do
  based on the collection of the models adjacent to $\mathcal{M}_j^{(0)}$ compute
  (i)   the estimated posterior probability of each model and
  (ii)  the relative edge inclusion probabilities
  and set t equal to the number of adjacent models + 1.
  while t<n_mod do
  (i)    a resampling move, i.e. randomly return to a previously visited
         model according to the estimated posterior probabilities and
  (ii)   and a local move, i.e. randomly choose an edge move (add or
         remove an edge) accordingly to the edge move probabilities
    once a new model is chosen recalculate model and edge posterior probabilities and set t=t+1
  end while
end for
```

Finally, extending the concept of median probability (MP) model, introduced by Barbieri and Berger (2004) in the context of regression models, the authors suggest to evaluate the graph structure through the MP-DAG, i.e. the graph containing those edges whose inclusion probability is at least 0.5.

Chapter 5

Objective Bayesian Analysis for Differential Gaussian Directed Acyclic Graphs

5.1 Introduction

Often we are confronted with heterogeneous multivariate data, i.e., data coming from several categories, and the interest may center on the differential structure of stochastic dependence among the variables between the groups, as an example consider two groups, refractory and relapsed patients affected by a specific cancer with the underlying q variables in each group being expressions from selectively targeted genes.

Suppose we model the dependence among variables through a graph (either undirected or directed). We could do this separately for each category, however it is reasonable to assume that there will be some shared edges across categories, and a joint estimation would be desirable to borrow strength and thus achieve a better inference.

A Bayesian approach to address this issue, in the undirected graph framework, under an informative setting, is presented in Peterson *et al.* (2014), they address the problem of inferring multiple undirected networks in situations where some of the networks may be unrelated. Let have k n_k-dimensional samples $Y^{(k)}$ from $\mathcal{N}_q(0, (\Omega^{(k)})^{-1})$ where $\Omega^{(k)}$ is the inverse of the covariance matrix. Given a graph structure they use a G-Wishart prior (Roverato, 2002) for each precision matrix $\Omega^{(k)}$ and link the graph structures via Markov random field priors which encourage common edges through edge-specific parameters and a supergraph which describes the relatedness between the graphs. On the parameters of the supergraph they put spike and slab priors (George and McCulloch, 1993) with a non-local alternative component and to the edge-specific parameters they assign a prior that encourages higher edge selection probabilities for edges included in a reference graph $\mathcal{G}_0$. Samples from the posterior distribution are obtained with a MCMC algorithm, and graph structures are selected thresholding the posterior marginal probabilities of edge inclusion.

Another Bayesian approach, but for Gaussian DAG models, is presented in Yajima *et al.* (2012) with regard to the two-category case. For category $k = 0, 1$, let the data be $Y^{(k)} = (y_1^{(k)}, \ldots, y_p^{(k)})$ with $y_j^{(k)} = (y_{1j}^{(k)}, \ldots, y_{n_k j}^{(k)})$. They assume the model:

$$y_{ij}^{(k)} \mid y_{i\,\mathrm{pa}_k(j)}^{(k)}, \alpha_j, \beta_j, \delta_j, \sigma_j^2, \mathcal{D}_k \overset{ind}{\sim} N(\alpha_j + \sum_{l \in \mathrm{pa}_k(j)} y_{il}^{(k)}(\beta_{jl} + \delta_{jl} I\{s_i = 1\}), \sigma_j^2),$$

for $i = 1, \ldots, n_k$, $j = 1, \ldots, p$. The scalar α_j is a nuisance parameter, σ_j^2 the error variance, $I\{A\}$ the indicator function of the event A, s_i is a subgroup indicator such that $s_i = I\{\text{differential group}\}$ and β_j the collection of regression coefficients $\{\beta_{sj},\ s \neq j,\ s = 1, \ldots, p\}$, with a similar definition for δ_j representing the vector of *differential* effects. Model determination is performed based on local priors, essentially standard variable selection priors (George and McCulloch, 1993), and using a Reversible Jump Markov Chain Monte Carlo (RJMCMC) algorithm (Green, 1995), along the lines described in Fronk and Giudici (2004).

We concentrate on the two groups problem and we face it modeling the system through a Gaussian DAG couple linked in a fashion similar to Yajima *et al.* (2012), assuming that an ordering of the variable is given. Our aim is thus model selection and we choose to work in a objective Bayesian framework so that no complex prior elicitation is needed. The philosophy we followed when designing this algorithm was the one of exploratory analysis, we tried to provide an output which could be helpful to a researcher who is facing a multidimensional problem, and often this results in having a sparse graph, and in our case a sparse graph couple. Our proposal consists thus in assigning a non-local prior to the regression coefficients with the objective of enforcing stronger sparsity constraints on model selection.

5.2 A new model parametrization

Given q variables of interest and two groups, the *baseline group*, indexed by 0, and the *differential group* indexed by 1, let $\mathcal{D} = \{\mathcal{D}_0, \mathcal{D}_1\}$ be the set of DAGs describing the dependence structure between the q variables in the two groups. It is important to note that, since we are interested in *differential effects*, $\mathcal{D} = \{\mathcal{D}_0, \mathcal{D}_1\}$ alone is not sufficient to describe our model, in fact, *given an ordering of the edges*, there are two different types of differential effect:

(i) we have an edge in $\mathcal{D}_0$ but not in $\mathcal{D}_1$ or viceversa

(ii) both graphs have the edge but the effect is not the same

While the first type of differential effects are caught by $\mathcal{D}$ the second type of differential effects suggests to introduce a third component: Δ, the set of *common edges with differential effect.* Now the triple $\mathcal{M} = (\mathcal{D}_0, \mathcal{D}_1, \Delta)$ defines our model.

As usual in the Gaussian Multivariate framework let see the DAGs, given an ordering of the variables, as sets of regression models, formally $\mathcal{D}_k$, for $k = 0, 1$, can be split into q independent models $\mathcal{M}_{k1}, \ldots, \mathcal{M}_{kq}$ and thus our triple could conveniently be expressed by the triples $\mathcal{M}_j = (\mathcal{M}_{0j}, \mathcal{M}_{1j}, \Delta_j)$ $j = 1, \ldots, q$. Note that for $j = 1$ there is no regression and $\Delta_1 = \emptyset$. To simplify our formulation we will further assume that the variable of interest have zero mean, so that no intercept is needed in our regressions, this is a standard assumption in graphical modeling and is achieved in practice by centering or standardizing the data matrix.

In this new formulation of the model Δ_j does not looses his meaning, since it will be the set of the parents of j which are common in the both regressions but with a differential effect. Let further $\overline{\Delta}_j = \mathrm{pa}_0(j) \cap \mathrm{pa}_1(j) \setminus \Delta_j$ be the set of common parents of j with common effect.

To fix the ideas we represent in a Venn diagram, in Figure 5.1, the parental sets, under the baseline and differential model, relative to vertex j.

5.3 Likelihood

Let us have a sample of n independent observations, n_0 for the baseline group, and n_1 for the differential group, and let $\boldsymbol{Y}$ be the centered or standardized

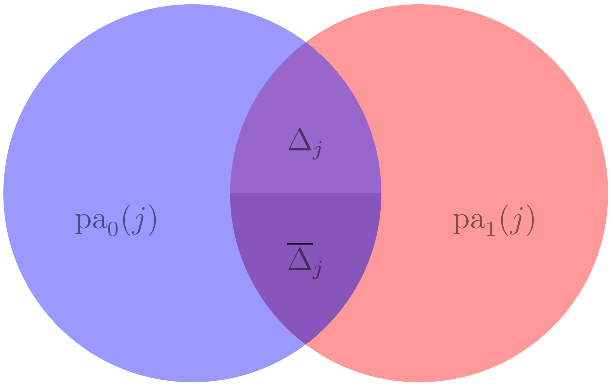


Figure 5.1: Venn diagram representing the sets induced by the triple $\mathcal{M}_j = (\mathcal{M}_{0j}, \mathcal{M}_{1j}, \Delta_j)$.

data matrix, ordered for representational convenience:

$$\underset{n\times q}{\boldsymbol{Y}} = \begin{bmatrix} \underline{y}_1 & \cdots & \underline{y}_q \end{bmatrix} = \begin{bmatrix} \underset{n_0\times q}{\boldsymbol{Y}_0} \\ \underset{n_1\times q}{\boldsymbol{Y}_1} \end{bmatrix} = \begin{bmatrix} \underline{y}_1^{(0)} & \cdots & \underline{y}_q^{(0)} \\ \underline{y}_1^{(1)} & \cdots & \underline{y}_q^{(1)} \end{bmatrix} = \begin{bmatrix} \left(y_{ij}^{(0)}\right)_{n_0\times q} \\ \left(y_{ij}^{(1)}\right)_{n_1\times q} \end{bmatrix}$$

with in mind our model formulation, we can assume, conditionally on knowing the group of each observation, the following likelihood:

$$\begin{aligned} &p(\boldsymbol{Y} \mid \mathcal{D}_0, \mathcal{D}_1, \Delta, \boldsymbol{\beta}, \boldsymbol{\delta}, \underline{\gamma}, \underline{c}) \\ &= \prod_{i=1}^{n_0} \prod_{j=1}^{q} p(y_{ij}^{(0)} \mid y_{i,\mathrm{pa}_0(j)}^{(0)}, \mathcal{D}_0, \underline{\beta}_j, \gamma_j) \prod_{i=1}^{n_1} \prod_{j=1}^{q} p(y_{ij}^{(1)} \mid y_{i,\mathrm{pa}_1(j)}^{(1)}, \mathcal{D}_0, \mathcal{D}_1, \Delta, \underline{\beta}_j, \underline{\delta}_j, \gamma_j, c_j) \end{aligned}$$

$$p(y_{ij}^{(0)} \mid \cdot) = \mathcal{N}\left(y_{ij}^{(0)} \middle| \mu_j^{(0)}, \gamma_j^{-1}\right)$$

$$p(y_{ij}^{(1)} \mid \cdot) = \mathcal{N}\left(y_{ij}^{(1)} \middle| \mu_j^{(1)}, (c_j\gamma_j)^{-1}\right)$$

with

$$\mu_j^{(0)} = \sum_{l\in\mathrm{pa}_0(j)} \beta_{lj} y_{il}^{(0)} \tag{5.1}$$

$$\mu_j^{(1)} = \sum_{l\in\overline{\Delta}_j} \beta_{lj} y_{il}^{(1)} + \sum_{l\in\Delta_j} (\beta_{lj} + \delta_{lj})\, y_{il}^{(1)} + \sum_{l\in\mathrm{pa}_1(j)\setminus\mathrm{pa}_0(j)} \delta_{lj} y_{il}^{(1)} \tag{5.2}$$

We further indicate with $\underline{\gamma} = (\gamma_1, \ldots, \gamma_q)^\top$ a vector of conditional precisions and $\underline{c} = (c_1, \ldots, c_q)^\top$ a multiplicative parameter. In order to make the computation faster, $\underline{c}$ will be treated as constant vector given the model and will be chosen according to the data[1]. Since c_j times the precision γ_j of the jth regression under the baseline model is nothing else than the precision of the jth regression under the differential model it seems reasonable to set c_j equal to the ratio of the estimated conditional variances.

The parameters we are most interested in are instead $\boldsymbol{\beta} = \{\underline{\beta}_1, \ldots, \underline{\beta}_q\}$ and $\boldsymbol{\delta} = \{\underline{\delta}_1, \ldots, \underline{\delta}_q\}$

$$\underline{\beta}_j \overset{\text{def}}{=} \beta_{\text{pa}_0(j)} = \begin{bmatrix} \beta_{\text{pa}_0(j)\setminus\text{pa}_1(j)} \\ \beta_{\text{pa}_0(j)\cap\text{pa}_1(j)} \end{bmatrix} = \begin{bmatrix} \beta_{\text{pa}_0(j)\setminus\text{pa}_1(j)} \\ \beta_{\overline{\Delta}_j} \\ \beta_{\Delta_j} \end{bmatrix}$$

$$\underline{\delta}_j \overset{\text{def}}{=} \delta_{\text{pa}_1(j)\setminus\overline{\Delta}_j} = \begin{bmatrix} \delta_{\Delta_j} \\ \delta_{\text{pa}_1(j)\setminus\text{pa}_0(j)} \end{bmatrix}$$

As we can see from these formulas their structure, i.e. which and how many parameters we have in each regression, will depend on our model triple $\mathcal{M}_j$, and setting a coefficient to 0 will bring us to a smaller model nested in the first one, which we could also express without including this parameter.

We can further see from (5.1) and (5.2) that the differential model depends on the regression coefficients of the baseline model for the edges which are common in both graphs. This happens in two situations, first when an edge is present in both graphs but the effect is different (Δ_j), and second when we suppose that the edge has the same effect in both baseline and differential model ($\overline{\Delta}_j$). Note that in the latter situation the parameter will need to fit both models to make the likelihood reasonably high, but keep in mind that our aim is model selection rather then parameter estimation.

5.4 Priors

5.4.1 Parameter priors

Because of the recursive structure of the likelihood is natural to assume the assumption of global parameter independence, see Geiger and Heckerman (2002):

$$p(\boldsymbol{\beta} \mid \mathcal{D}_0) = \prod_j p(\underline{\beta}_j \mid \mathcal{D}_0)$$

[1] This choice will become clearer when deriving the Fractional Bayes Factor because it will allow us to get it in closed form

$$p(\boldsymbol{\delta} \mid \boldsymbol{\beta}, \mathcal{D}_0, \mathcal{D}_1, \Delta) = \prod_j p(\underline{\delta}_j \mid \underline{\beta}_j, \mathcal{D}_0, \mathcal{D}_1, \Delta)$$

$$p(\underline{\gamma}) = \prod_j p(\gamma_j)$$

Since our aim is to investigate differential effects is natural to prefer possibly sparse models where only strong differential effects are considered, this intuition suggests to work with *product moment priors* (Consonni and La Rocca, 2011; Altomare *et al.*, 2013). We remember from section 4.2 that product moment priors arise from a Bayesian testing procedure based on combining the advantages of the FBF with those of the moment prior, in order to obtain an objective method with enhanced learning behavior. Therefore when assigning parameter priors we have to concentrate on pairwise comparison of two nested models.

Definition 41 (model inclusion) In our framework a model

$$\mathcal{M}^{(A)} = (\mathcal{D}_0^{(A)}, \mathcal{D}_1^{(A)}, \Delta^{(A)})$$

is said to be included in a model

$$\mathcal{M}^{(B)} = (\mathcal{D}_0^{(B)}, \mathcal{D}_1^{(B)}, \Delta^{(B)})$$

if and only if for each $j \in \{1, \ldots, q\}$ and each $l < j$:

1. $l \in \overline{\Delta}_j^{(B)} \Rightarrow l \in \mathrm{pa}_0^{(A)}(j) \Leftrightarrow l \in \mathrm{pa}_1^{(A)}(j)$
2. $l \in \Delta_j^{(B)} \Rightarrow l \notin \mathrm{pa}_0^{(A)}(j) \setminus \mathrm{pa}_1^{(A)}(j)$

or equivalently if and only if

- $\mathrm{pa}_0^{(A)}(j) \setminus \mathrm{pa}_1^{(A)}(j) \subset \mathrm{pa}_0^{(B)}(j) \setminus \mathrm{pa}_1^{(B)}(j)$
- $\mathrm{pa}_1^{(A)}(j) \setminus \mathrm{pa}_0^{(A)}(j) \subset \mathrm{pa}_1^{(B)}(j) \setminus \mathrm{pa}_0^{(B)}(j) \cup \Delta_j^{(B)}$
- $\Delta_j^{(A)} \subset \Delta_j^{(B)}$
- $\overline{\Delta}_j^{(A)} \subset \overline{\Delta}_j^{(B)} \cup \Delta_j^{(B)}$ □

This assures that, when the inclusion is respected, the likelihood under $\mathcal{M}^{(A)}$ can be obtained from the likelihood under $\mathcal{M}^{(B)}$ simply by setting some $\delta_{lj}^{(B)}$,$\beta_{lj}^{(B)}$ to zero. Note that, in contrast with the single graph case where setting a parameter to zero corresponds to removing an edge, here when removing a parameter different scenarios can arise. Since we always remove edges or set parameters to zero from the bigger model $\mathcal{M}^{(B)}$, to enumerate the various scenarios let drop the superscript (B)

1. $\beta_{lj} = 0$
   - $l \in \mathrm{pa}_0 \setminus \mathrm{pa}_1$: we remove edge $l \to j$ in $\mathcal{D}_0$, this is the same as the single graph case.
   - $l \in \overline{\Delta}$: we remove edge $l \to j$ from both $\mathcal{D}_0$ and $\mathcal{D}_1$
   - $l \in \Delta$: we remove edge $l \to j$ from $\mathcal{D}_0$ and Δ
2. $\delta_{lj} = 0$
   - $l \in \mathrm{pa}_1 \setminus \mathrm{pa}_0$: we remove edge $l \to j$ from $\mathcal{D}_1$, this is the same as the single graph case.
   - $l \in \Delta$: we remove edge $l \to j$ from Δ and let $\mathcal{D}_1$ and $\mathcal{D}_0$ untouched
3. $\beta_{lj} = 0$ and $\delta_{lj} = 0$
   - we remove edge $l \to j$ from $\mathcal{D}_0$, $\mathcal{D}_1$ and Δ_j.

This highlights the importance of introducing the third component Δ in our model formulation, without it we could not have an identified model when removing/adding a parameter.

Following the method introduced in section 4.2 we choose as base priors the default priors

$$p^D(\underline{\beta}_j \mid \mathcal{D}_0) \propto 1$$

$$p^D(\underline{\delta}_j \mid \mathcal{D}_1, \Delta) \propto 1$$

$$p^D(\gamma_j) \propto \gamma_j^{-1}$$

and given two models $\mathcal{M}^{(A)} \subset \mathcal{M}^{(B)}$, we define,

$$L_j^{\beta} = \mathrm{pa}_0^{(B)}(j) \setminus \mathrm{pa}_0^{(A)}(j)$$

and

$$L_j^{\delta} = \left[\left(\mathrm{pa}_1^{(B)}(j) \setminus \mathrm{pa}_0^{(B)}(j) \right) \setminus \left(\mathrm{pa}_1^{(A)}(j) \setminus \mathrm{pa}_0^{(A)}(j) \right) \right] \cup \left(\Delta_j^{(B)} \setminus \Delta_j^{(A)} \right)$$

so that the hypothesis that $\mathcal{M}^{(A)}$ holds is equivalent to set $\beta_{lj}^{(B)} = 0$, $l \in L_j^{\beta}$ and $\delta_{lj}^{(B)} = 0$, $l \in L_j^{\delta}$, $j = 1, \dots, q$ in $\mathcal{M}^{(B)}$. The corresponding default product moment priors of order h are then

$$p(\underline{\beta}_j | \mathcal{D}_0) \propto \prod_{l \in L_j^{\beta}} \beta_{lj}^{2h}$$

$$p(\underline{\delta}_j | \underline{\beta}_j, \mathcal{D}_0, \mathcal{D}_1, \Delta) \propto \prod_{l \in L_j^{\delta}} \delta_{lj}^{2h}$$

so that

$$p(\underline{\beta}_j, \underline{\delta}_j, \gamma_j | \mathcal{D}_0, \mathcal{D}_1, \Delta) \propto \gamma_j^{-1} \prod_{l \in L_j^{\beta}} \beta_{lj}^{2h} \prod_{l \in L_j^{\delta}} \delta_{lj}^{2h} \tag{5.3}$$

where h is a positive integer, as usual $h = 0$ returns the initial default prior. Note that when we are comparing against the null model, i.e. a complete independence model in both group, our prior will then be:

$$p(\underline{\beta}_j, \underline{\delta}_j, \gamma_j | \mathcal{D}_0, \mathcal{D}_1, \Delta) \propto \gamma_j^{-1} \prod_{l \in \mathrm{pa}_0(j)} \beta_{lj}^{2h} \prod_{l \in \mathrm{pa}_1(j) \setminus \overline{\Delta}_j} \delta_{lj}^{2h}$$

5.4.2 Model prior

For the model prior after several proposal we choose to go for an uniform prior

$$p(\mathcal{D}_0, \mathcal{D}_1, \Delta) \propto 1$$

since from the simulations we have seen that this prior is more stable. By stable we mean that the uniform prior against sparsity inducing priors or common structure inducing priors, performs slightly worse for pairs of DAGs with high common structure, e.g. when more then 80% of the edges are the same in both graphs, but outperforms these priors when the two graph are not that similar.

5.5 Moment Fractional Bayes Factor

When we are interested in comparing

$$\mathcal{M}^{(A)} = (\mathcal{D}_0^{(A)}, \mathcal{D}_1^{(A)}, \Delta^{(A)})$$

against

$$\mathcal{M}^{(B)} = (\mathcal{D}_0^{(B)}, \mathcal{D}_1^{(B)}, \Delta^{(B)})$$

with $\mathcal{M}^{(A)} \subset \mathcal{M}^{(B)}$ we can note that given the recursive structure of the likelihood and of the priors, then also the fractional marginal likelihood, under each hypothesis, factorizes

$$w_h^{(H)}(\boldsymbol{Y}, g_0, g_1) = \prod_{j=1}^{q} w_{j,h}^{(H)}(\underline{y}_j \mid y^{(0)}_{\mathrm{pa}_0(j)}, y^{(1)}_{\mathrm{pa}_1(j)}, g_0, g_1)$$

with $H = A, B$ and g_0 and g_1 the sample size dependent fractions, see subsection 3.2.3. The subscript h is, as usual, the order of the product moment prior, and we remember that under the null hypothesis our parameter prior is the default local prior used to define the product moment prior. Furthermore this prior can be obtained simply by setting $h = 0$ in the product moment prior of order h, therefore we will simply drop this subscript when talking about the null hypothesis. So let $w_h^{(B)}$ be the fractional marginal likelihood for model $\mathcal{M}^{(B)}$ and $w^{(A)}$ the fractional marginal likelihood for model $\mathcal{M}^{(A)}$, then the moment fractional Bayes factor for comparing $\mathcal{M}^{(A)}$ against $\mathcal{M}^{(B)}$ with $\mathcal{M}^{(A)} \subset \mathcal{M}^{(B)}$ is

$$MFBF_{BA}^{(h)} = \frac{w_h^{(B)}}{w^{(A)}} = \prod_{j=1}^{q} \frac{w_{j,h}^{(B)}}{w_j^{(A)}} = \prod_{j=1}^{q} MFBF_{j,BA}^{(h)}$$

This factorization allows us to work at node j level. So let focus on the fractional marginal likelihood under the alternative model $\mathcal{M}^{(B)}$ and drop the superscript (B) and the subscript j for notational convenience. Let

$$\boldsymbol{X}_0 = \begin{bmatrix} y^{(0)}_{\mathrm{pa}_0 \setminus \mathrm{pa}_1} & y^{(0)}_{\underline{\Delta}} & y^{(0)}_{\Delta} & \mathbf{0} & \mathbf{0} \end{bmatrix}$$

and

$$\boldsymbol{X}_1 = \begin{bmatrix} \mathbf{0} & y^{(1)}_{\underline{\Delta}} & y^{(1)}_{\Delta} & y^{(1)}_{\Delta} & y^{(1)}_{\mathrm{pa}_1 \setminus \mathrm{pa}_0} \end{bmatrix}$$

be the observations of the parents of node j under DAG $\mathcal{D}_0$ and $\mathcal{D}_1$, augmented with 0's to be coherent in size with $\underline{\tilde{\beta}} = \begin{bmatrix} \beta \\ \underline{\delta} \end{bmatrix}$. The fractional marginal likelihood is then

$$w_h = \frac{I(h, \underline{y}^{(0)}, \underline{y}^{(1)}, \boldsymbol{X}_0, \boldsymbol{X}_1, 1, 1)}{I(h, \underline{y}^{(0)}, \underline{y}^{(1)}, \boldsymbol{X}_0, \boldsymbol{X}_1, g_0, g_1)} \tag{5.4}$$

with

$$I(\cdot) = \int p(\underline{y}^{(0)} \mid \underline{\beta}, \gamma, \mathcal{D}_0; \boldsymbol{X}_0)^{g_0} p(\underline{y}^{(1)} \mid \underline{\beta}, \underline{\delta}, \gamma, \mathcal{D}_0, \mathcal{D}_1, \Delta; \boldsymbol{X}_1, \underline{c})^{g_1} \quad (5.5)$$
$$p(\underline{\beta}, \underline{\delta}, \gamma \mid \mathcal{D}_0, \mathcal{D}_1, \Delta)\, \mathrm{d}\,\underline{\beta}\, \mathrm{d}\,\underline{\delta}\, \mathrm{d}\,\gamma$$

The main issue with $I(\cdot)$ is that we have a product of parameters, coming from the product moment prior, that we have to integrate out. In (5.5) we can see the product of two multivariate normal densities one raised to the power of g_0 and the other to the power of g_1. With a single multivariate normal density we could proceed like in Consonni and La Rocca (2011), thus it would be useful to group these two densities in a single object, so focusing on the exponentials of the likelihoods we can note that

$$\begin{aligned}
&\exp\left\{-\frac{g_0\gamma}{2}\|\underline{y}^{(0)} - \boldsymbol{X}_0\underline{\tilde{\beta}}\|^2\right\}\exp\left\{-\frac{cg_1\gamma}{2}\|\underline{y}^{(1)} - \boldsymbol{X}_1\underline{\tilde{\beta}}\|^2\right\} = \\
&= \exp\left\{-\frac{\gamma}{2}\left(\begin{bmatrix} \underline{y}^{(0)} \\ \underline{y}^{(1)} \end{bmatrix} - \begin{bmatrix} \boldsymbol{X}_0 \\ \boldsymbol{X}_1 \end{bmatrix}\underline{\tilde{\beta}}\right)^{\top} \begin{bmatrix} g_0\boldsymbol{I}_{n_0} & \underset{n_0\times n_1}{\mathbf{0}} \\ \underset{n_1\times n_0}{\mathbf{0}} & cg_1\boldsymbol{I}_{n_1} \end{bmatrix} \left(\begin{bmatrix} \underline{y}^{(0)} \\ \underline{y}^{(1)} \end{bmatrix} - \begin{bmatrix} \boldsymbol{X}_0 \\ \boldsymbol{X}_1 \end{bmatrix}\underline{\tilde{\beta}}\right)\right\} \\
&= \exp\left\{-\frac{\gamma}{2}\|\underline{y}_g - \boldsymbol{X}_g\underline{\tilde{\beta}}\|^2\right\}
\end{aligned}$$

where we have indicated

$$\underline{y}_g = \boldsymbol{G}^{\frac{1}{2}}\begin{bmatrix} \underline{y}^{(0)} \\ \underline{y}^{(1)} \end{bmatrix} = \begin{bmatrix} \sqrt{g_0}\underline{y}^{(0)} \\ \sqrt{cg_1}\underline{y}^{(1)} \end{bmatrix} \qquad \boldsymbol{X}_g = \boldsymbol{G}^{\frac{1}{2}}\begin{bmatrix} \boldsymbol{X}_0 \\ \boldsymbol{X}_1 \end{bmatrix} = \begin{bmatrix} \sqrt{g_0}\boldsymbol{X}_0 \\ \sqrt{cg_1}\boldsymbol{X}_1 \end{bmatrix}$$

with

$$\boldsymbol{G} = \begin{bmatrix} g_0\boldsymbol{I}_{n_0} & \underset{n_0\times n_1}{\mathbf{0}} \\ \underset{n_1\times n_0}{\mathbf{0}} & cg_1\boldsymbol{I}_{n_1} \end{bmatrix}$$

if we further define

$$\underline{\hat{\beta}}_g = (\boldsymbol{X}_g^{\top}\boldsymbol{X}_g)^{-1}\boldsymbol{X}_g^{\top}\underline{y}_g$$

and

$$S_g^2 = (\underline{y}_g - \boldsymbol{X}_g\underline{\hat{\beta}}_g)^{\top}(\underline{y}_g - \boldsymbol{X}_g\underline{\hat{\beta}}_g)$$

we can use the well know relation

$$\begin{aligned}
\exp\left\{-\frac{\gamma}{2}\|\underline{y}_g - \boldsymbol{X}_g\underline{\tilde{\beta}}\|^2\right\} &= \exp\left\{-\frac{\gamma}{2}\left(\|\underline{y}_g - \boldsymbol{X}_g\underline{\hat{\beta}}_g\|^2 + \|\boldsymbol{X}_g\underline{\hat{\beta}}_g - \boldsymbol{X}_g\underline{\tilde{\beta}}\|^2\right)\right\} \\
&= \exp\left\{-\frac{\gamma}{2}S_g^2\right\}\exp\left\{-\frac{\gamma}{2}\|\boldsymbol{X}_g(\underline{\tilde{\beta}} - \underline{\hat{\beta}}_g)\|^2\right\}
\end{aligned}$$

from which we can see the kernel of a

$$\mathcal{N}_{|\underline{\tilde{\beta}}|}\left(\underline{\tilde{\beta}};\underline{\hat{\beta}}_g,\frac{1}{\gamma}\left(\boldsymbol{X}_g^\top\boldsymbol{X}_g\right)^{-1}\right)$$

With in in mind (5.3) let $d=|L^\beta|+|L^\delta|$ then

$$\begin{aligned}I(\cdot)&=\int c^{\frac{n_1g_1}{2}}\left(\frac{\gamma}{2\pi}\right)^{\frac{n_0g_0+n_1g_1}{2}}\exp\left\{-\frac{\gamma}{2}S_g^2\right\}\left(\frac{\gamma}{2\pi}\right)^{-\frac{|\underline{\tilde{\beta}}|}{2}}\mid\boldsymbol{X}_g^\top\boldsymbol{X}_g\mid^{-\frac{1}{2}}\mathbb{E}\left[\prod_{l\in L^\beta}\beta_{lj}^{2h}\prod_{l\in L^\delta}\delta_{lj}^{2h}\right]\gamma^{-1}\,\mathrm{d}\,\gamma\\&=c^{\frac{n_1g_1}{2}}(2\pi)^{-\frac{n_0g_0+n_1g_1-|\underline{\tilde{\beta}}|}{2}}\mid\boldsymbol{X}_g^\top\boldsymbol{X}_g\mid^{-\frac{1}{2}}\int\mathbb{E}\left[\prod_{l\in L^\beta}\beta_{lj}^{2h}\prod_{l\in L^\delta}\delta_{lj}^{2h}\right]\gamma^{\frac{n_0g_0+n_1g_1-|\underline{\tilde{\beta}}|}{2}-1}\exp\left\{-\frac{\gamma}{2}S_g^2\right\}\mathrm{d}\,\gamma\end{aligned}$$

applying Lemma 1 and integrating out γ we get

$$\begin{aligned}I(\cdot)&=c^{\frac{n_1g_1}{2}}(2\pi)^{-\frac{n_0g_0+n_1g_1-|\underline{\tilde{\beta}}|}{2}}\mid\boldsymbol{X}_g^\top\boldsymbol{X}_g\mid^{-\frac{1}{2}}\sum_{i=0}^{hd}\frac{1}{2^i}H_i^{(h)}(\underline{\hat{\beta}}_g,\boldsymbol{X}_g^\top\boldsymbol{X}_g^{-1})\int\gamma^{\frac{n_0g_0+n_1g_1-|\underline{\tilde{\beta}}|-2i}{2}-1}\exp\Bigg\{\\&\quad-\frac{\gamma}{2}S_g^2\Bigg\}\mathrm{d}\,\gamma\\&=c^{\frac{n_1g_1}{2}}(2\pi)^{-\frac{n_0g_0+n_1g_1-|\underline{\tilde{\beta}}|}{2}}\mid\boldsymbol{X}_g^\top\boldsymbol{X}_g\mid^{-\frac{1}{2}}\sum_{i=0}^{hd}\frac{1}{2^i}H_i^{(h)}(\underline{\hat{\beta}}_g,\boldsymbol{X}_g^\top\boldsymbol{X}_g^{-1})\frac{\Gamma\left(\frac{n_0g_0+n_1g_1-|\underline{\tilde{\beta}}|-2i}{2}\right)}{\left(\frac{S_g^2}{2}\right)^{\frac{n_0g_0+n_1g_1-|\underline{\tilde{\beta}}|-2i}{2}}}\\&=c^{\frac{n_1g_1}{2}}\mid\boldsymbol{X}_g^\top\boldsymbol{X}_g\mid^{-\frac{1}{2}}\left(\pi S_g^2\right)^{-\frac{n_0g_0+n_1g_1-|\underline{\tilde{\beta}}|}{2}}\sum_{i=0}^{hd}\frac{1}{4^i}\left(S_g^2\right)^i\Gamma\left(\frac{n_0g_0+n_1g_1-\mid\underline{\tilde{\beta}}\mid-2i}{2}\right)H_i^{(h)}(\underline{\hat{\beta}}_g,\boldsymbol{X}_g^\top\boldsymbol{X}_g^{-1})\end{aligned}$$

Focusing on the gamma function we note that the fractions $0<g_0<1$ and $0<g_1<1$ have to be chosen so that

$$n_0g_0+n_1g_1>|\underline{\tilde{\beta}}|+2hd$$

One thing one has to be careful about, is that because of the particular construction we adopted to solve this integral, $\boldsymbol{X}_g$, $\underline{\hat{\beta}}_g$ and S_g^2 they will all depend on the fractions g_0 and g_1, and therefore nothing will cancel out when computing the marginal fractional likelihood $w_h^{(B)}$ (5.4).

Under the null hypothesis $\mathcal{M}^{(A)}$ our parameter prior is the default local prior used to define the product moment prior, thus the fractional marginal likelihood $w^{(A)}$ can be obtained from the same formulas above just setting $h=0$.

5.6 Stochastic search

The recursive structure of the MFBF gives us the opportunity to work locally. Therefore to search in the space of models, following Altomare *et al.* (2013), we exploit the encompassing from below approach and the approximation of the MFBF with the BF chain rule, and we adapt the local search algorithm described previously to this framework.

The main difference between the two algorithms is that in the single graph framework a local move proposes an adjacent model, i.e. a model that differs exactly by one parameter from the starting model. In our framework this could be a limit and to improve the way our algorithm moves through the space of triples we prefer to define a *local move* in slightly different way.

Definition 42 (local move) we define a local move the change of status of an edge couple, where by status of an edge couple we mean

1. the edge is missing in both graphs
2. the edge is present in $\mathcal{D}_0$ but not in $\mathcal{D}_1$
3. the edge is present in $\mathcal{D}_1$ but not in $\mathcal{D}_0$
4. the edge is present in both $\mathcal{D}_0$ and $\mathcal{D}_1$ but not in Δ
5. the edge is present in both $\mathcal{D}_0$ and $\mathcal{D}_1$ and in Δ □

We see from our definition of inclusion, that making a move from status 4 to 2, from status 4 to 3 and from status 5 to 2 it doesn't involves nested models, i.e. we don't have a MFBF with which can compare these 2 models. This can be solved noting that two models, $\mathcal{M}^{(i)}$ and $\mathcal{M}^{(j)}$, differing just by a single edge status e, with $e = 1$ in $\mathcal{M}^{(i)}$ and $e \neq 1$ in $\mathcal{M}^{(j)}$, then $\mathcal{M}^{(i)}$ is always nested in $\mathcal{M}^{(j)}$. Therefore we can calculate the MFBF in two steps, i.e. passing through status 1 and taking the ratio between the two intermediate Bayes Factors.

We have implemented all this logic in MATLAB and at the end of the day, given a starting triple $\mathcal{M}^{(b)} = (\mathcal{D}_0^{(b)}, \mathcal{D}_1^{(b)}, \Delta^{(b)}) = (\mathcal{M}_1^{(b)}, \ldots, \mathcal{M}_q^{(b)})$, with $\mathcal{M}_j^{(b)} = (\mathcal{M}_{0j}^{(b)}, \mathcal{M}_{1j}^{(b)}, \Delta_j^{(b)})$ $j = 1, \ldots, q$ and the number of models we want to explore for each vertex (`n_mod`) our algorithm can be summarized as

Algorithm 2 Local stochastic search for differential Gaussian DAGs

```
for j=2;j<=q;j++ do
    based on the collection of the models adjacent to $\mathcal{M}_j^{(b)}$ compute
    (i)   the estimated posterior probability of each model and
    (ii)  the relative edge inclusion probabilities
    and set t equal to the number of adjacent models + 1.
    while t<n_mod do
    (i)   a resampling move, i.e. randomly return to a previously visited
          model according to the estimated posterior probabilities and
    (ii)  and a local move, i.e. randomly choose an edge move (change
          the status of an edge couple) accordingly to the edge move probabilities
        once a new model is chosen recalculate model and edge posterior probabilities and set t=t+1
    end while
end for
```

5.7 Simulations

5.7.1 Comparison with the disjoint model

First of all we want to evaluate if our joint proposal improves the results obtained by applying on each graph separately the standard method proposed by Altomare *et al.* (2013). In this section we will call this method the disjoint method. In order to complete this task we simulate from three different scenarios. In the first one we suppose that the graphs share 90% of their edges, in the second that 50% of the edges are shared and in the last one just 10%. We apply these scenarios to the 20, 50 and 100 variables problem, see Table 5.1

Simulation strategy

To present how we simulate the data let s be the desired percentage of shared edges between the graphs. We start by simulating a DAG randomly recurring to the `R` package `pcalg` (Kalisch *et al.*, 2012), with sparsity parameter

	q	$n_0 = n_1$	s	z
1	20	20	0.9	0.1
2	20	20	0.5	0.1
3	20	20	0.1	0.1
4	50	50	0.9	0.1
5	50	50	0.5	0.1
6	50	50	0.1	0.1
7	100	50	0.9	0.1
8	100	50	0.5	0.1
9	100	50	0.1	0.1

Table 5.1: Descriptive schema of the simulations considered for the comparison of the joint vs the disjoint method, s is the percentage of shared structure, and z the sparsity level.

$z = 0.1$ and setting edge weights randomly to 0.4 or 0.8. In our simulations edge weights are the weights of the parameters of the conditional regressions with conditional variance equal to 1, this approach gives standardized effects around 0.3 and 0.6 respectively. The DAG obtained this way will be our baseline DAG $\mathcal{D}_0$. To determine $\mathcal{D}_1$ let $\mathcal{D}_1 = \mathcal{D}_0$ and randomly remove from $\mathcal{D}_1$ the desired percentage of edges we want to be different $(1 - s)$, at the same time we add in the same amount new edges to $\mathcal{D}_1$, i.e. edges not present in $\mathcal{D}_0$, with edge weights 0.4 or 0.8 randomly. Finally, we choose randomly $(1 - s) \times 100\%$ of of the edges we have not touched yet and in order to introduce some common edges with differential effect we change their weights to 0.4 or 0.8 respectively.

When we have $\mathcal{D}_0$ and $\mathcal{D}_1$ we generate the data with the function `rmvDAG` of the package `pcalg`. This is equal to simulate from a normal distribution with zero mean and covariance matrix Σ_k $k = 0, 1$ given by (2.6) with $D = I_q$.

Results

In Table 5.1 we summarize with a schema the simulations considered, and in the appendix (subsection B.1.2) we represent the corresponding graphs. We can see that we have chosen situations where the number of observations are relatively small, since we know that the disjoint algorithm, under these conditions, with a moderate sample size reconstructs the graph almost perfectly, thus, with such sample sizes, we would not have had any space to see if there is some improvement in searching the graphs jointly.

Since the sample sizes are small we simulate 3 datasets for each configuration and we asses the performances of the methods with the mean of the AUCs obtained from the graphs selected by thresholding the edge posterior probabilities at various points. In particular we will look at the *overall AUC* (Table 5.2), i.e. the AUC we get considering false positive edge discovery rates and true positive edge discovery rates in both graphs as they would be a single model (Table 5.3), and at the *shared structure AUC* (Table 5.4), i.e. the AUC we get considering as a true positive an edge present in both true graphs (Table 5.5). The corresponding ROC curves can be found in the appendix (subsection B.1.1).

We test the datasets, in both algorithms, with $h = 1$ and $h = 2$.

From Table 5.2 we see that the joint method outperforms the disjoint one when the graphs have many common edges ($s = 0.9$). Instead, when the graphs don't share much structure, with a moderate number of nodes it has performances similar to the disjoint method while it seems to deteriorate when $q = 100$. This problem may be connected to the fact that when the sample size is too small (in relation to the graph size) and there is no significant common structure between the graphs, the algorithm still tries to borrow strength from the two graphs, and since the small sample can not adequately separate between the two models the performances deteriorate. Anyway even in these situations if we look at the AUCs relative to the common structure (Table 5.4) we see that our algorithm at least reconstructs common edges better.

A strategy to exploit the best of the two approaches may be the one of trying to launch the disjoint algorithm first, and if from its output we see that the two graphs are similar (e.g. more then 50% of the edges are the same) then probably we can improve the performances with the joint method.

5.7.2 Common edges with differential effect

To compare the disjoint with the joint method we have not looked at the shared edges with differential effect, since the disjoint method is not able to catch these differences. To understand how our algorithm works in relation to these effects we consider a graph couple with common structure of 0.8, 50 nodes and sparsity 0.1 for various values of the sample size: $50, 100, 250$. Since the method we use to generate the random DAGs doesn't let us control the standardized effects precisely we introduce a threshold on how much these effects have to be different in order to be considered differential. We set these thresholds to $0, 0.05, 0.1, 0.2$ and we test our algorithm with $h = 1$. Results

Model			h=1		h=2	
q	$n_0 = n_1$	s	Joint	Disjoint	Joint	Disjoint
20	20	0.9	0.83	0.77	0.87	0.74
20	20	0.5	0.82	0.81	0.84	0.81
20	20	0.1	0.81	0.81	0.82	0.78
50	50	0.9	0.98	0.94	0.98	0.89
50	50	0.5	0.95	0.95	0.95	0.91
50	50	0.1	0.93	0.93	0.91	0.90
100	50	0.9	0.91	0.89	0.92	0.84
100	50	0.5	0.86	0.88	0.85	0.84
100	50	0.1	0.85	0.90	0.84	0.86

Table 5.2: overall AUCs for the joint and disjoint method for $h = 1$ and $h = 2$

			h=1				h=2			
Model			Joint		Disjoint		Joint		Disjoint	
q	$n0 = n1$	s	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
20	20	0.9	0.65	0.15	0.19	0.02	0.58	0.06	0.13	0.01
20	20	0.5	0.59	0.13	0.31	0.02	0.51	0.06	0.22	0.01
20	20	0.1	0.55	0.14	0.29	0.03	0.45	0.07	0.19	0.01
50	50	0.9	0.88	0.03	0.60	0.01	0.82	0.01	0.51	0.00
50	50	0.5	0.82	0.04	0.67	0.01	0.77	0.02	0.61	0.01
50	50	0.1	0.76	0.05	0.64	0.01	0.68	0.03	0.56	0.01
100	50	0.9	0.80	0.06	0.56	0.01	0.78	0.03	0.46	0.01
100	50	0.5	0.71	0.07	0.56	0.01	0.65	0.05	0.49	0.01
100	50	0.1	0.68	0.07	0.60	0.01	0.60	0.04	0.51	0.01

Table 5.3: mean of the overall TPRs and overall FPRs for the joint and disjoint method for $h = 1$ and $h = 2$

Model			h=1		h=2	
q	$n_0 = n_1$	s	Joint	Disjoint	Joint	Disjoint
20	20	0.9	0.83	0.74	0.88	0.70
20	20	0.5	0.76	0.70	0.79	0.74
20	20	0.1	0.77	0.74	0.89	0.69
50	50	0.9	0.98	0.91	0.98	0.84
50	50	0.5	0.95	0.92	0.97	0.86
50	50	0.1	0.97	0.94	0.98	0.88
100	50	0.9	0.90	0.84	0.92	0.77
100	50	0.5	0.86	0.82	0.86	0.78
100	50	0.1	0.88	0.82	0.86	0.74

Table 5.4: AUCs relative to the shared structure recognition for the joint and disjoint method with $h = 1$ and $h = 2$

			h=1				h=2			
Model			Joint		Disjoint		Joint		Disjoint	
q	$n0 = n1$	s	TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
20	20	0.9	0.59	0.10	0.06	0.00	0.54	0.04	0.02	0.00
20	20	0.5	0.44	0.09	0.11	0.01	0.41	0.04	0.06	0.00
20	20	0.1	0.41	0.11	0.08	0.01	0.40	0.06	0.03	0.00
50	50	0.9	0.86	0.02	0.48	0.00	0.81	0.01	0.40	0.00
50	50	0.5	0.78	0.03	0.48	0.00	0.76	0.02	0.40	0.00
50	50	0.1	0.78	0.04	0.42	0.00	0.70	0.03	0.29	0.00
100	50	0.9	0.74	0.03	0.40	0.00	0.74	0.02	0.31	0.00
100	50	0.5	0.67	0.04	0.37	0.00	0.62	0.03	0.30	0.00
100	50	0.1	0.66	0.05	0.33	0.00	0.61	0.03	0.24	0.00

Table 5.5: mean of the TPRs and FPRs relative to the shared structure recognition for the joint and disjoint method for $h = 1$ and $h = 2$

		0			0.05		
q	$n_0 = n_1$	AUC	TPR	FPR	AUC	TPR	FPR
50	50	0.51	0.04	0.00	0.53	0.06	0.00
50	100	0.60	0.20	0.00	0.63	0.26	0.00
50	250	0.68	0.37	0.00	0.78	0.56	0.00
		0.1			0.2		
q	$n_0 = n_1$	AUC	TPR	FPR	AUC	TPR	FPR
50	50	0.54	0.08	0.00	0.51	0.01	0.01
50	100	0.66	0.31	0.00	0.76	0.50	0.01
50	250	0.83	0.67	0.00	0.93	0.87	0.02

Table 5.6: AUCs for the differential edges recognition for various thresholds: $0, 0.05, 0.1, 0.2$.

are reported in Table 5.6. For these examples the corresponding overall AUCs are 0.97, 0.99 and 1 respectively. We see instead, that catching the common edges with differential effect is an harder task and that even with a threshold of 0.2, which seems a reasonable threshold for us, we need a moderate sample size to get good results. This behavior is to impute to the fact we chose a product moment prior for the δ parameters, and thus just differential effects which are considerably high are highlighted by this method.

5.7.3 Comparison with Peterson *et al.* (2014)

The most advanced Bayesian method we found in literature for making inference on multiple graphs is the one of Peterson *et al.* (2014), we will call it the benchmark method. This algorithm works in the undirected graphs framework, so if we want to compare our method with theirs we have to assume a known ordering and focus on decomposable graphs, see section 2.3.

Simulation strategy

To simulate the data, we proceed in a way similar to the one described earlier, with the only expedient of transforming our randomly chosen DAGs to perfect DAGs and obtaining therefore the UGs just removing the arrows.

Note that sparsity and shared structure will not be fully controllable when generating DAGs and UGs randomly in this way.

Results

Before presenting the results we may note that we put ourselves in a favorite situation, since we're are assuming a known ordering of the variables. The benchmark algorithm cannot exploit the given ordering since it works in an undirected framework, so to rebalance the comparison we decided to check the performances of our algorithm also in the case the ordering is mis-specified.

In order to set a desired mis-specification level note that if we have an ordered sequence $1, \dots, q$, and a permutation of that sequence $\pi = \pi(1), \dots, \pi(q)$, the number of inversions, i.e. the number of pairs $(\pi(i), \pi(j))$ such that $\pi(i) < \pi(j)$ and $i > j$, is a well-established way to asses how far π is from the naturally ordered sequence. Moreover we know that the number of inversions takes values between 0 and $q(q-1)/2$, so dividing the number of inversions by its maximum we get a relative distance d.

We decided to test our algorithm under three different degrees of mis-specification, namely $d = 0, 0.25, 1$. We investigate the performances of the algorithms under 6 situations, in particular we sampled data from 4 different randomly chosen DAG couples with 20 nodes each, and different values of sample size, graph sparsity and common structure. In order to reduce the variability due to the sampling, we decided to run the algorithms on three different samples under each configuration.

	model	q	$n_0 = n_1$	$\approx$ z	$\approx$ s
1	1	20	20	0.1	0.8
2	1	20	50	0.1	0.8
3	1	20	100	0.1	0.8
4	2	20	100	0.2	0.8
5	3	20	100	0.2	0.5
6	4	20	100	0.2	0.3

Table 5.7: Descriptive schema of the simulations considered for the comparison with the benchmark method.

In Table 5.7 we summarize with a schema the situations considered, and in the appendix we represent the corresponding graphs (subsection B.2.2). We set up our algorithm with $h = 1$ and with regard to the benchmark algorithm, following PETERSON *et al.* (2014), we set the hyper-parameters of the G-Wishart prior in the "noninformative setting" $b = 3$ and $D = I_q$. For the edge specific prior we follow the suggestions given to encourage overall sparsity, i.e. we choose as reference graph a $q \times q$ matrix of all ones and set $a = 1$ and $b = 4$ for the parameters of the Beta governing the edge inclusion probability, which leads, for every edge, to a prior probability of edge inclusion of 20% and finally we set the hyperparameter w of the Bernoulli prior on the latent indicator of network relatedness equal to 0.8 for the first 4 examples and to 0.5 in last 2, to set a prior belief that the networks are related in a way similar to the true one. Finally we run a MCMC chain of 30000 samples of which 10000 for burn-in.

We present in Table 5.8 the overall AUCs, in Table 5.10 the common structure AUCs and the corresponding ROC curves in the appendix (subsection B.2.1).

From the overall AUCs we see that our method outperforms the benchmark when the order is not mis-specified. When $d = 0.25$ our algorithm still works better then the benchmark when the sample size is small, while we get similar results when the sample size grows. Finally with $d = 1$, we perform similarly to the benchmark when there is an high sparsity level ($z = 0.1$), which clearly makes the mis-specification less dramatic, otherwise the benchmark outperforms our method. Similar results hold also for the shared structure AUCs in Table 5.10.

We may further note that in ALTOMARE *et al.* (2013) it is stated that the performance of the MFBF search with respect to the mis-specification deteriorates with the number of v-structures, i.e. unmarried parents that meet head to head, present in the true graph. In our examples we have no v-structures since we needed to restrict our attention to decomposable graphs. Anyway these results seem to suggest that when the ordering is known or if we have at least a general idea on how to order the variables, we can get an advantage using our method, while when the ordering is completely unknown we may rather go with the benchmark method, remembering that this implies to carefully set the hyper-parameters in order to get good results.

5.8 Conclusion and Discussion

In this work we concentrated on the comparison between two Directed Acyclic Graph (DAG) models with the objective of finding differences, if any, between the two DAGs, both in terms of conditional independencies and in terms of the strengths of common dependencies.

We approached model selection using an objective Bayes framework, so that minimal prior elicitation is needed and we made use of non-local priors on the regression coefficients to incorporate in our model a sparsity assumption. Finally we tested our method against the approach of Peterson *et al.* (2014) showing that our algorithm is competitive in several settings.

We conclude this work by considering some issues worth of further investigation. Our model selection procedure was based on the Fractional Bayes Factor (FBF), essentially because in this way the expression of the marginal likelihood is available in closed form, and this greatly speeds up computations. Other approaches of model choice, such as the intrinsic priors Berger and Pericchi (1996), may be also theoretically sound, but computationally less efficient, especially in an high dimensional settings, where numerical integrations can slow down the algorithm massively.

A critical point of our procedure is the assumption that there exists a known ordering of the variables. Notice that this requirement is in some way natural for DAGs which are mathematical objects relying on some ordering which provides the orientation of the edges. They help researchers better understand the problem. Moreover we have shown that our algorithm still gives good results when the mis-specification of the ordering is moderate. We would like to point out that our tests on the mis-specification where conducted in a situation where there were no immoralities, and with the growth of the immoralities the performances in relation to the mis-classification should deteriorate.

If we look at DAGs purely as models for conditional independence, a drawback is that our algorithm does not assign equal scores to DAGs contained in the same equivalence classes, since non-local priors do not fall in the (restrictive) class of priors characterized by this feature Geiger and Heckerman (2002). A potential way out of this difficulty, would be to consider only equivalence classes of DAGs, whose representative is an essential graph Andersson *et al.* (1997).

An additional problem that we have noted is that, when the sample size is small in relation to the number of variables, and the graphs do not share enough structure, our method does not perform as well as the disjoint one (analyzing the two graphs separately). Therefore in such situations a recommendation is to run the disjoint method first, asses how much structure the two graphs share, and if the common edges are prevalent, running the joint algorithm should bring improvements on model selection.

Peterson et. Al	d=0	d=0.25	d=1
0.75	0.90	0.87	0.80
0.88	0.98	0.93	0.87
0.90	1.00	0.89	0.89
0.93	0.99	0.92	0.82
0.93	0.98	0.93	0.83
0.93	0.98	0.91	0.83

Table 5.8: overall AUCs of the joint MFBF method (for $d = 0, 0.25, 1$) and of the benchmark method.

		Joint					
Peterson et al.		$d = 0$		$d = 0.25$		$d = 1$	
TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
0.86	0.64	0.68	0.13	0.66	0.12	0.56	0.13
0.68	0.13	0.85	0.04	0.75	0.05	0.62	0.08
0.72	0.08	0.95	0.01	0.66	0.05	0.66	0.05
0.75	0.10	0.94	0.03	0.76	0.04	0.61	0.14
0.75	0.09	0.90	0.06	0.78	0.08	0.61	0.15
0.73	0.08	0.89	0.05	0.74	0.07	0.59	0.11

Table 5.9: mean of the TPRs and FPRs for the joint MFBF method (for $d = 0, 0.25, 1$) and the benchmark method.

Peterson et. Al	d=0	d=0.25	d=1
0.77	0.92	0.88	0.82
0.90	0.99	0.93	0.89
0.91	1.00	0.87	0.87
0.94	0.99	0.90	0.82
0.93	0.98	0.88	0.78
0.91	0.98	0.86	0.79

Table 5.10: Shared Structure AUCs for the joint MFBF method (for $d = 0, 0.25, 1$) and the benchmark method.

		Joint					
Peterson et al.		$d = 0$		$d = 0.25$		$d = 1$	
TPR	FPR	TPR	FPR	TPR	FPR	TPR	FPR
0.84	0.61	0.62	0.09	0.60	0.08	0.51	0.09
0.60	0.07	0.83	0.03	0.73	0.04	0.60	0.06
0.62	0.04	0.94	0.01	0.59	0.04	0.59	0.04
0.66	0.06	0.91	0.02	0.71	0.03	0.54	0.10
0.48	0.06	0.85	0.06	0.62	0.07	0.36	0.09
0.42	0.05	0.87	0.05	0.56	0.07	0.34	0.07

Table 5.11: mean of the TPRs and FPRs for the joint MFBF method vs the benchmark method.

Appendix A

Notes about non-local priors

A.1 Scale Invariance

Given the linear model:

$$\underset{n\times 1}{\underline{y}} = \underset{n\times p}{\boldsymbol{X}}\underset{p\times 1}{\underline{\beta}} + \underset{n\times 1}{\underline{\epsilon}} \tag{A.1}$$

with:

$$H_0 : \beta_i = 0 \quad \forall i$$

To evaluate the properties of the prior proposed by Consonni and La Rocca (2011), since the model selection exploits the FBF, we have to consider the induced fractional prior:

$$\begin{aligned}\pi^{FBF}(\underline{\beta} \mid \sigma^2) &\propto \pi^{CL}(\underline{\beta} \mid \sigma^2) p(\underline{y} \mid \boldsymbol{X}\underline{\beta}, \sigma^2\boldsymbol{I})^b \\ &\propto \prod \beta_i^{2h} \mathcal{N}(\underline{\beta} \mid \hat{\underline{\beta}}, \frac{\sigma^2}{b}(\boldsymbol{X}^\top\boldsymbol{X})^{-1})\end{aligned} \tag{A.2}$$

It is desirable that a prior on the regression parameter will be coherent when defined on a scaled model, in the sense that the prior on the scaled model can be constructed either directly by applying the "construction rule" or by transforming the prior accordingly to the new model. The scaled model can be obtained either scaling $\underline{y}$ or $\boldsymbol{X}$ or even both. In general, letting $c, k > 0$ the scaled model will be:

$$c\underline{y} = k\boldsymbol{X}\frac{c}{k}\underline{\beta} + c\underline{\epsilon} \Leftrightarrow \underline{y}' = \boldsymbol{X}'\underline{\beta}' + \underline{\epsilon}' \tag{A.3}$$

Prior (A.2) has this property since for model (A.3) automatically we would define the prior on $\underline{\beta}'$ like:

$$\pi^{FBF}(\underline{\beta}' \mid \sigma'^2) \propto \prod \beta_i'^2 \mathcal{N}(\underline{\beta}' \mid \hat{\underline{\beta}}', \frac{\sigma'^2}{b}(\boldsymbol{X}'^\top \boldsymbol{X}')^{-1})$$

while defining the prior on (A.1) and then obtaining the prior through transforming the variables would give:

$$\begin{aligned}
\pi_{\underline{\beta}'}(\underline{\beta}' \mid \sigma'^2) &= \pi_{\underline{\beta}}(\frac{k}{c}\underline{\beta}' \mid \sigma^2)\left(\frac{k}{c}\right)^p \\
&\propto \prod \beta_i'^2 \exp\left[-\frac{b}{\sigma^2}\|(\underline{y} - \boldsymbol{X}\frac{k}{c}\underline{\beta}')\|_2^2\right] \\
&= \prod \beta_i'^2 \exp\left[-\frac{b}{\sigma^2 c^2}\|(c\underline{y} - \boldsymbol{X}k\underline{\beta}')\|_2^2\right] \\
&= \prod \beta_i'^2 \exp\left[-\frac{b}{\sigma'^2}\|(\underline{y}' - \boldsymbol{X}'\underline{\beta}')\|_2^2\right] \\
&\propto \prod \beta_i'^2 \mathcal{N}(\underline{\beta}' \mid \hat{\underline{\beta}}', \frac{\sigma'^2}{b}(\boldsymbol{X}'^\top \boldsymbol{X}')^{-1})
\end{aligned}$$

The prior proposed by (Johnson and Rossell, 2010, 2012) instead exploits τ for achieving this result.

A.2 Comparison

If we consider the model:

$$y_i = \mu + \epsilon_i \quad \epsilon_i \overset{iid}{\sim} \mathcal{N}(0, \sigma^2) \quad H_0 : \mu = 0$$

then the first order pMOM proposed by Johnson and Rossell (2010, 2012)) reduces to:

$$\pi^{JR}(\mu) \propto \mu^2 \mathcal{N}(\mu \mid 0, \tau\sigma^2)$$

the fractional prior (A.2) becomes:

$$\pi^{FBF}(\mu) \propto \mu^2 \mathcal{N}(\mu \mid \overline{y}, \frac{\sigma^2}{bn})$$

and interestingly the non-local prior obtained by considering the Zellner g-prior as local component is:

$$\pi^{G}(\mu) \propto \mu^2 \mathcal{N}(\mu, 0, \frac{g\sigma^2}{n})$$

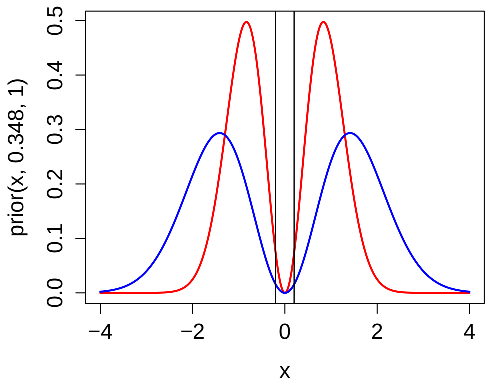


Figure A.1: prior with $\tau = 0.348$ in red and with $\tau = 1$ in blue

If we center the data and consider the default choices suggested for the FBF ($b = \frac{1}{n}$)) and g-prior ($g = \max(n, p^2)$) we see that in several settings these two priors match, and they differ from Johnson and Rossell (2010, 2012) just for the scaling factor τ. It is important to note that even though these priors have the same distribution, when computing the FBF not the whole likelihood is considered but just a fraction, namely $(1 - b) = \frac{n-1}{n}$, but this difference becomes negligible when n is sufficiently large.

In Johnson and Rossell (2010, 2012) it is recommended to choose τ so that: $\mathbb{P}(|\frac{\mu}{\sigma}| \geq 0.2) = 0.99$, giving $\tau = .348$, instead setting $\tau = 1$ we get $\mathbb{P}(|\frac{\mu}{\sigma}| \geq 0.2) = 0.998$, indicating an even stronger selection effect, which can be seen also graphically in fig. A.1.

Appendix B

Supplementary materials

B.1 Comparison with the disjoint model

B.1.1 ROC Curves

ROC Curves for the comparison when $h = 1$

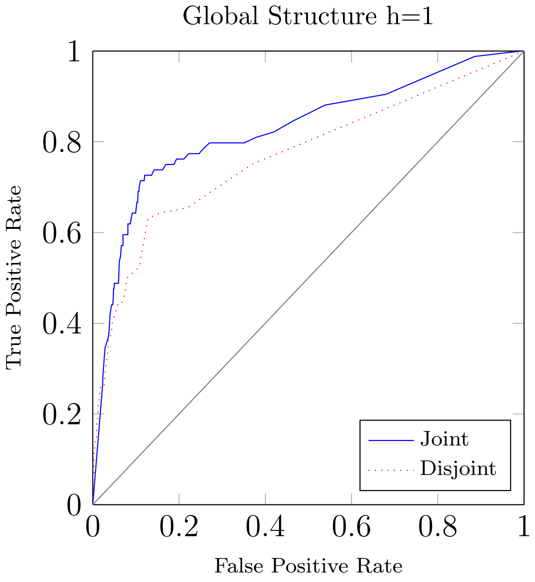


Figure B.1: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 1 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.9$)

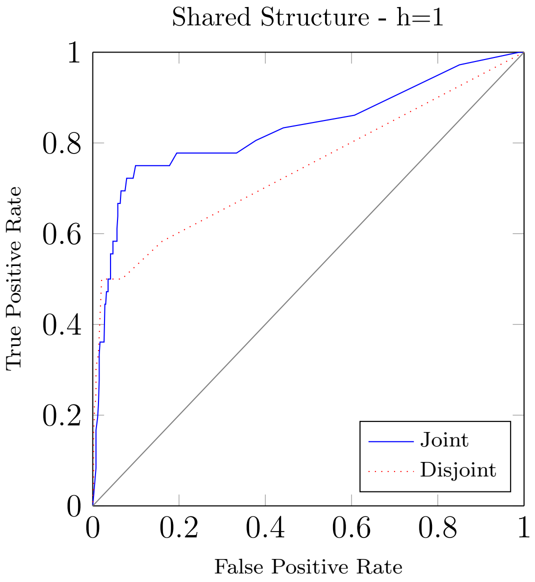


Figure B.2: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 1 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.9$)

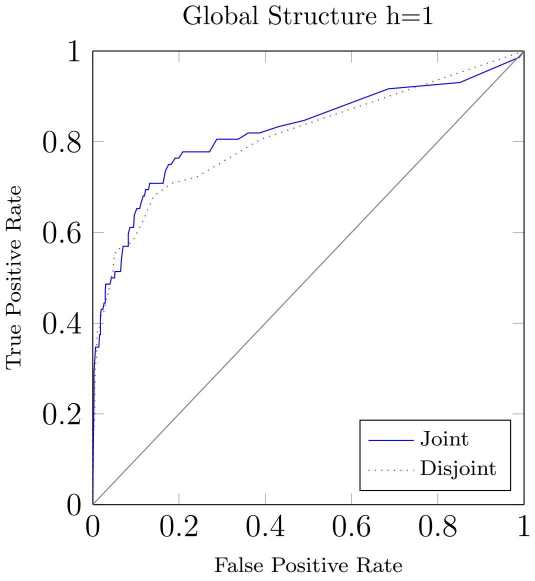


Figure B.3: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 2 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.5$)

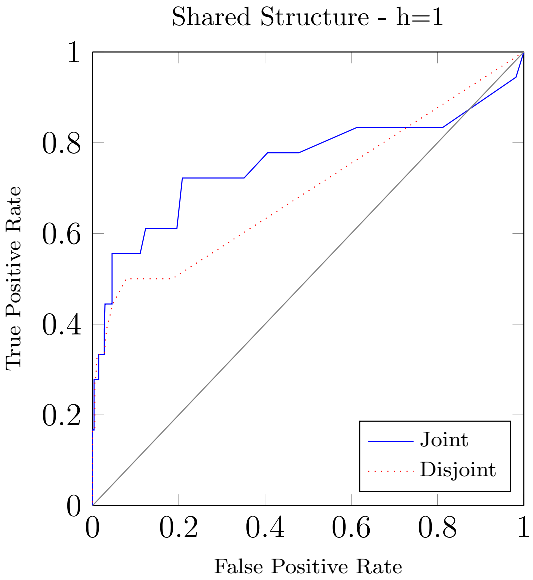


Figure B.4: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 2 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.5$)

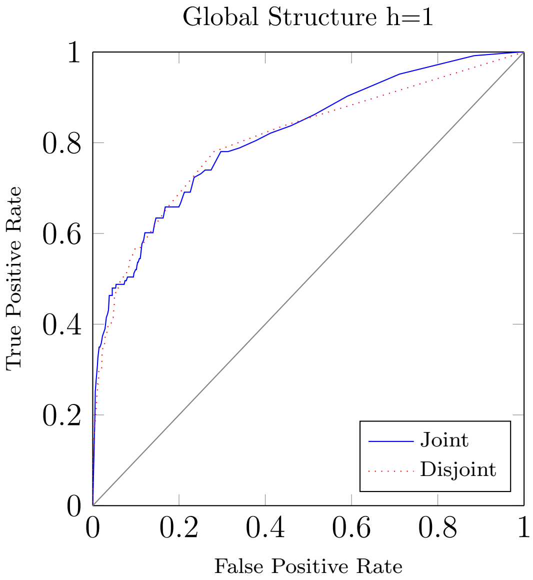


Figure B.5: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 3 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.1$)

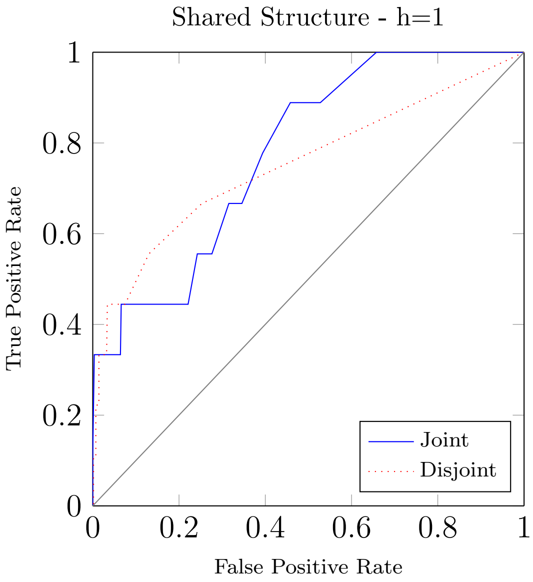


Figure B.6: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 3 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.1$)

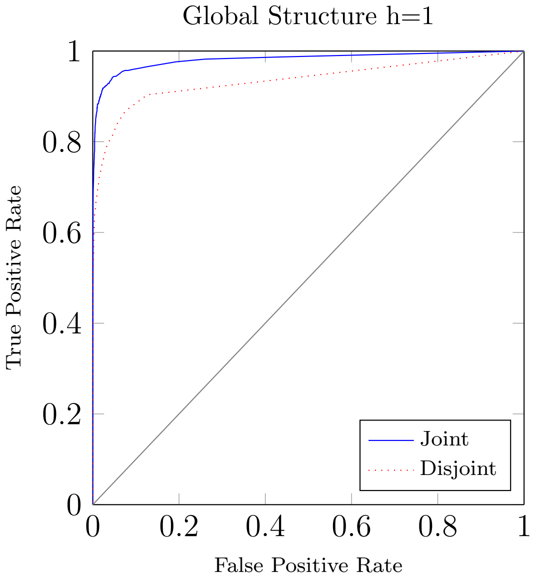


Figure B.7: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 4 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.9$)

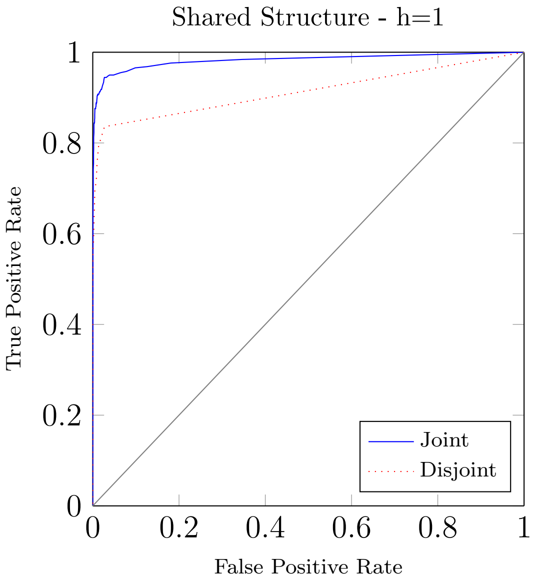


Figure B.8: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 4 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.9$)

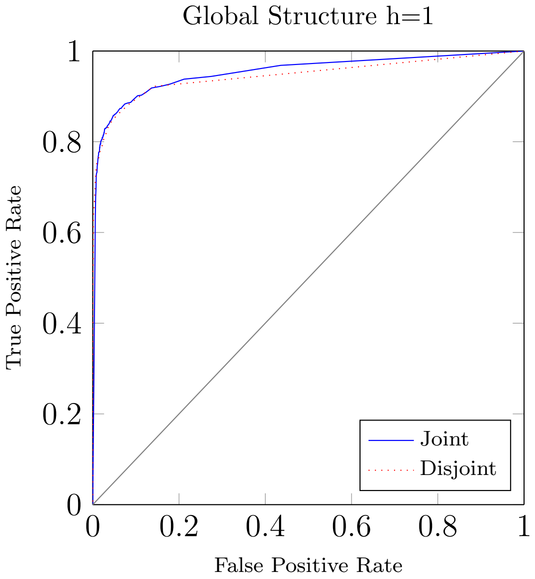


Figure B.9: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 5 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.5$)

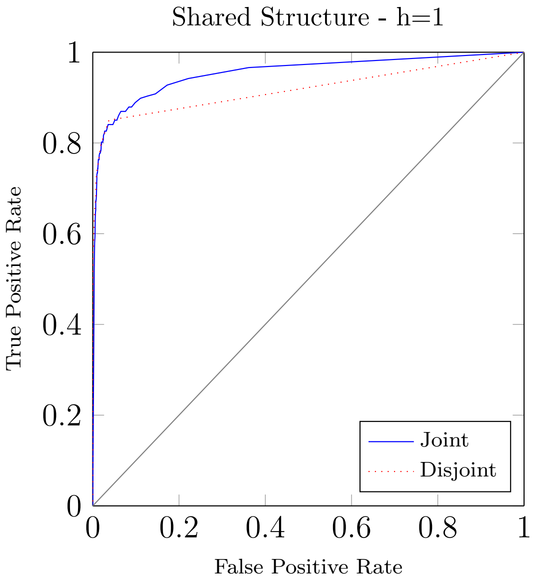


Figure B.10: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 5 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.5$)

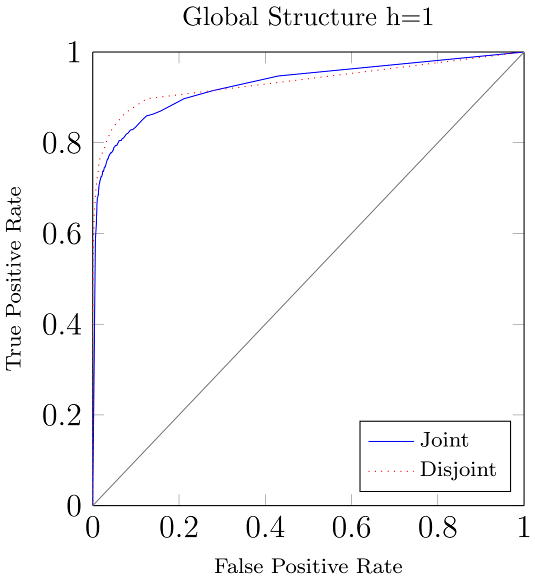


Figure B.11: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 6 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.1$)

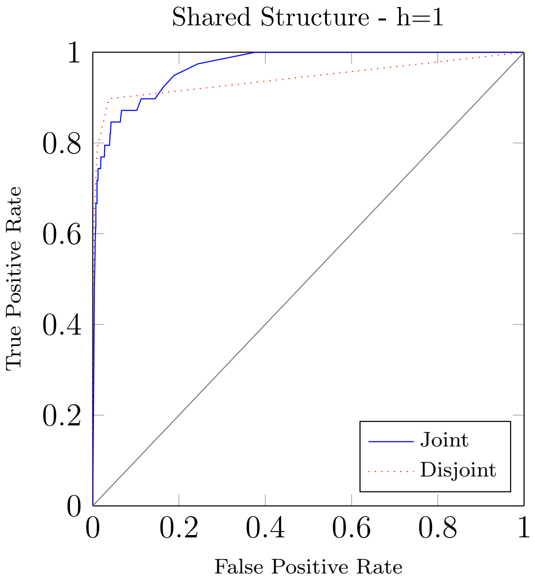


Figure B.12: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 6 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.1$)

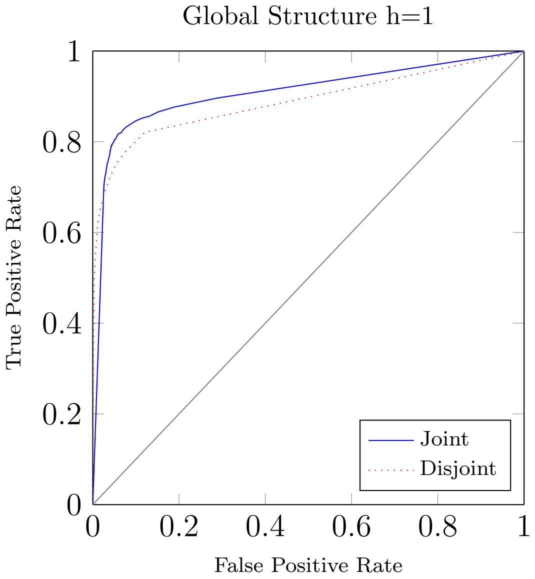


Figure B.13: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 7 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.9$)

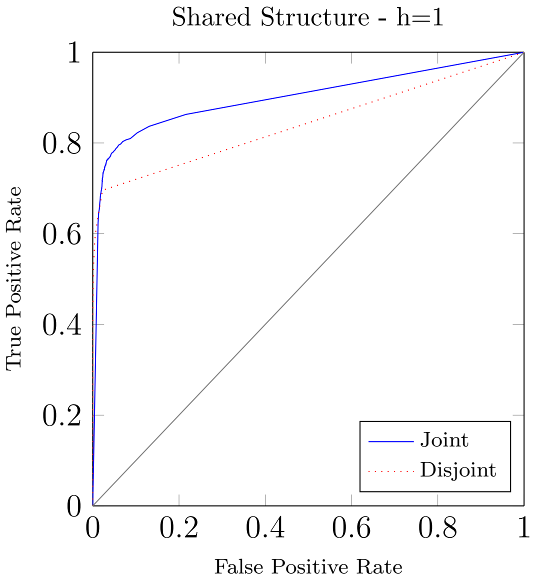


Figure B.14: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 7 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.9$)

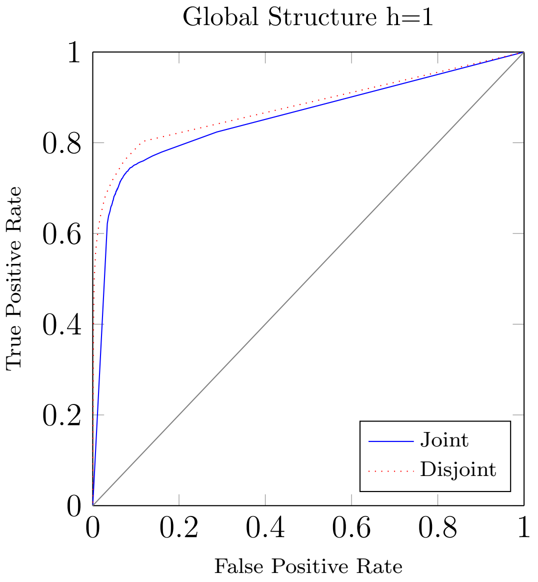


Figure B.15: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 8 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.5$)

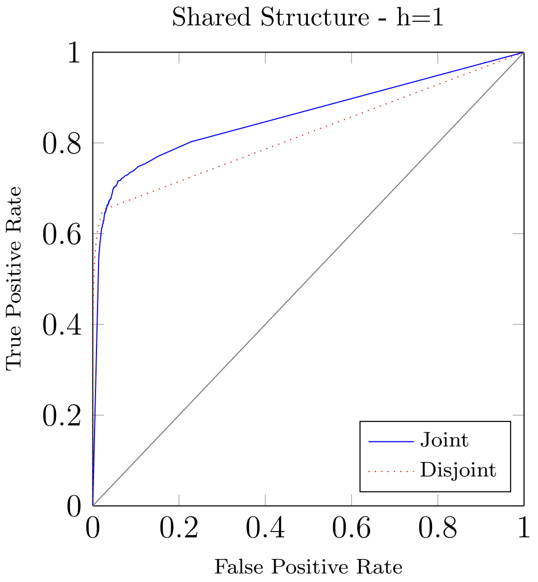


Figure B.16: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 8 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.5$)

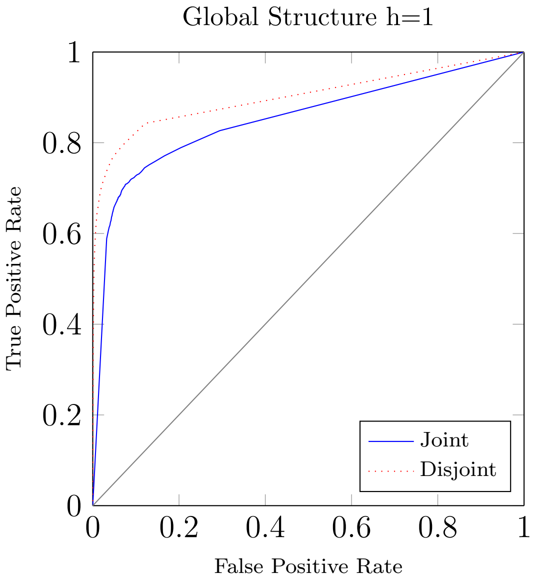


Figure B.17: Joint vs Disjoint with $h = 1$: Overall structure recognition for simulation 9 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.1$)

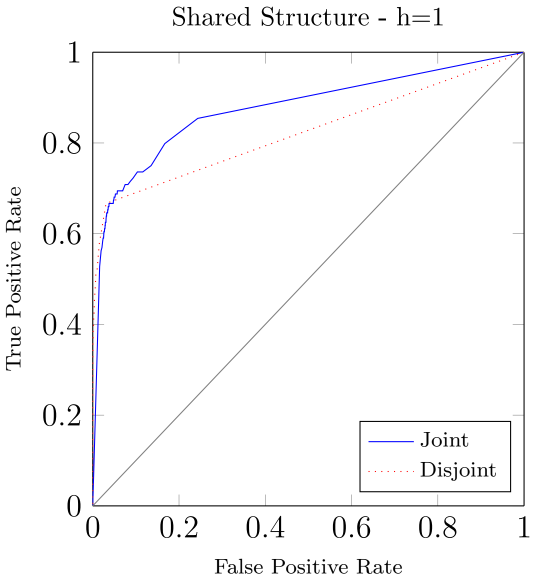


Figure B.18: Joint vs Disjoint with $h = 1$: Shared structure recognition for simulation 9 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.1$)

ROC Curves for the comparison when $h = 2$

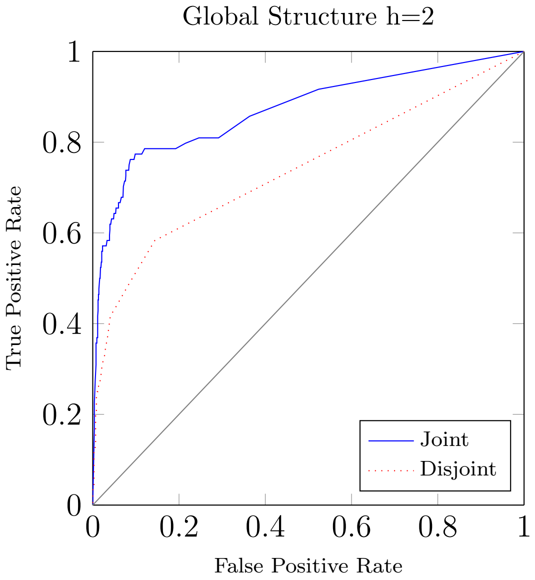


Figure B.19: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 1 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.9$)

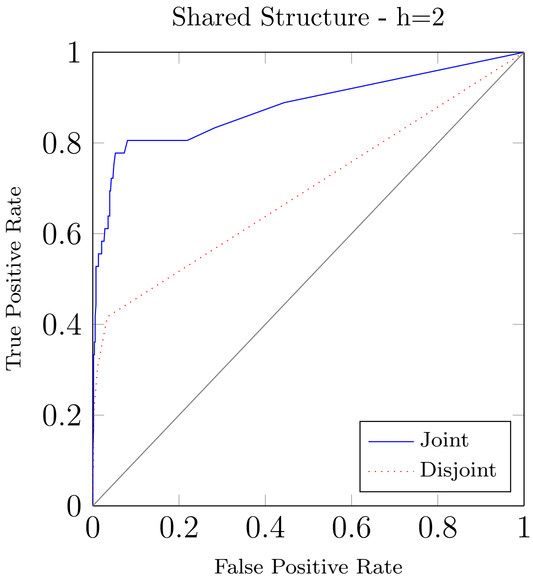


Figure B.20: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 1 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.9$)

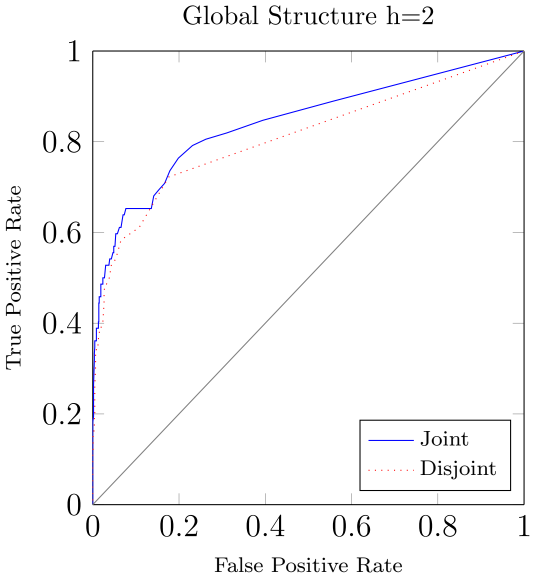


Figure B.21: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 2 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.5$)

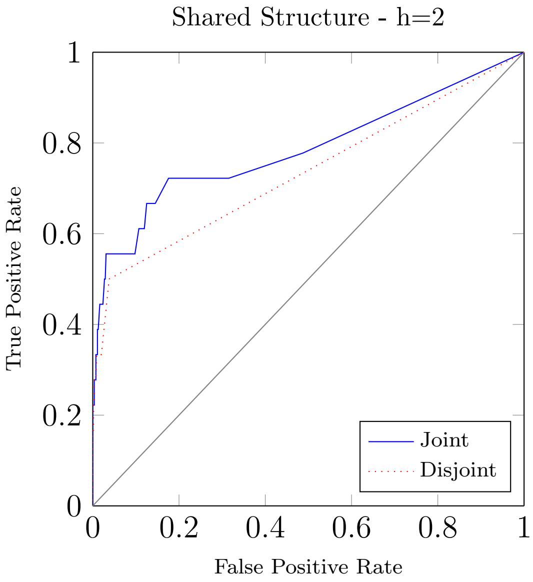


Figure B.22: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 2 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.5$)

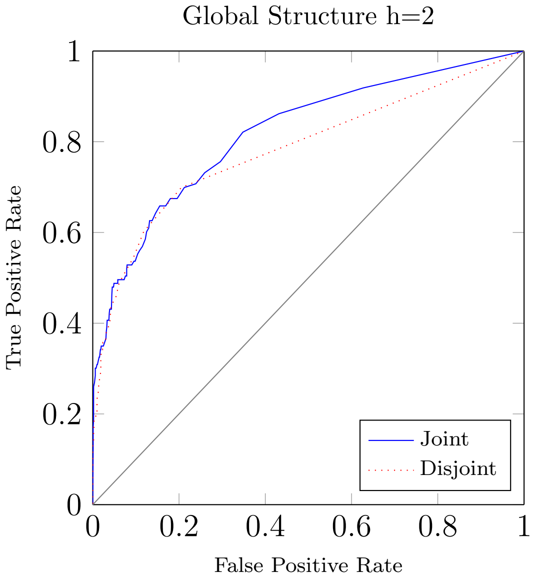


Figure B.23: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 3 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.1$)

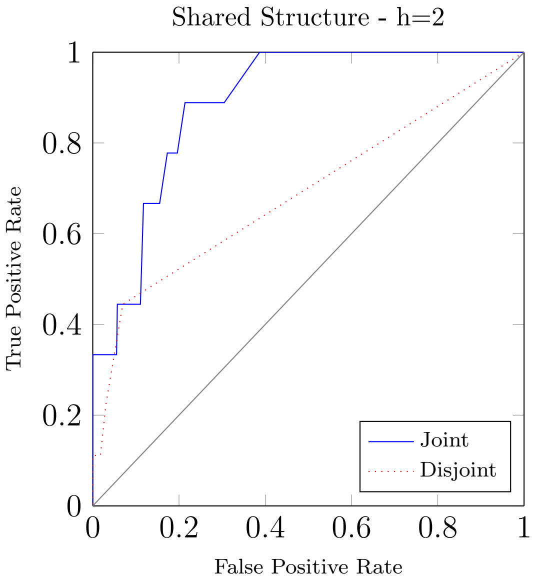


Figure B.24: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 3 ($q = 20$, $n_0 = n_1 = 20$, $s = 0.1$)

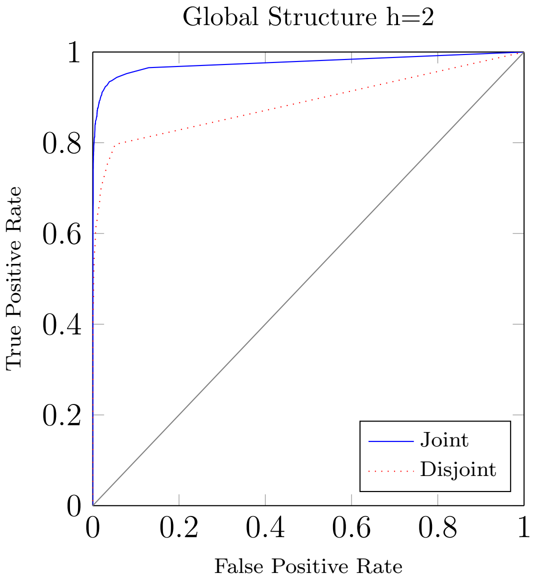


Figure B.25: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 4 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.9$)

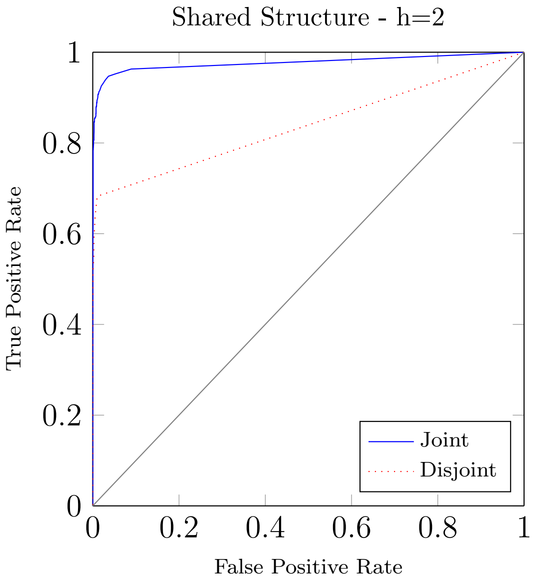


Figure B.26: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 4 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.9$)

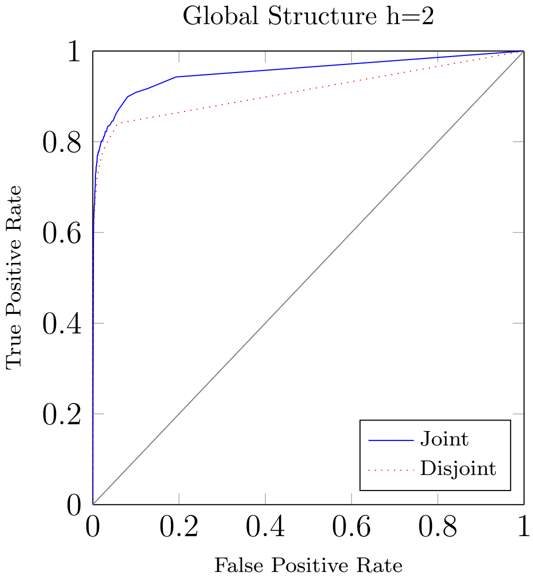


Figure B.27: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 5 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.5$)

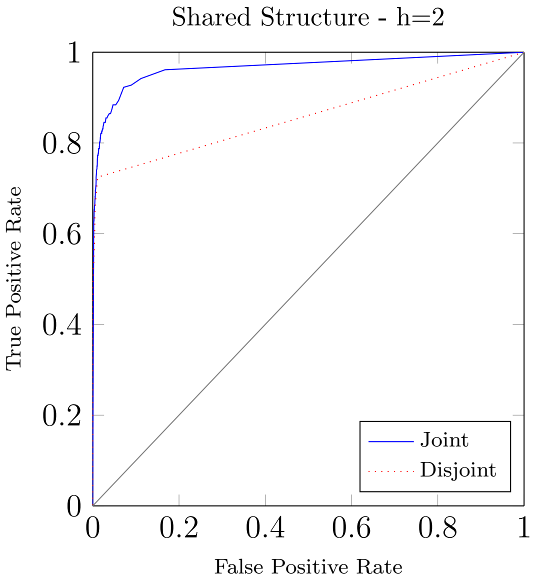


Figure B.28: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 5 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.5$)

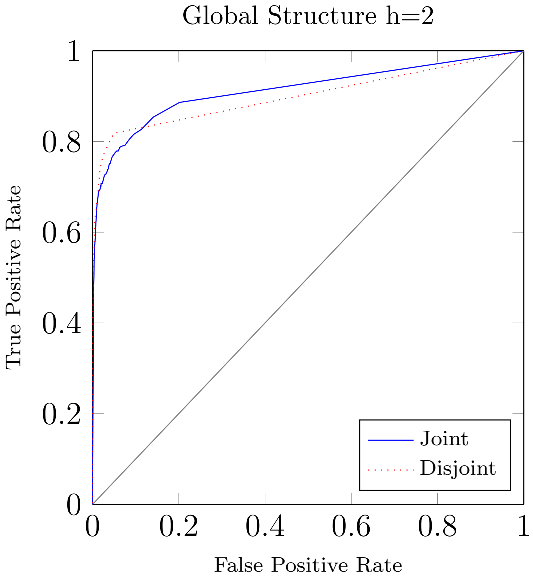


Figure B.29: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 6 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.1$)

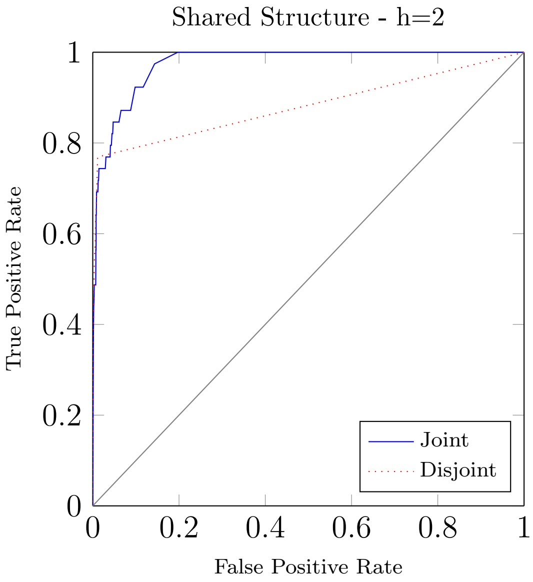


Figure B.30: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 6 ($q = 50$, $n_0 = n_1 = 50$, $s = 0.1$)

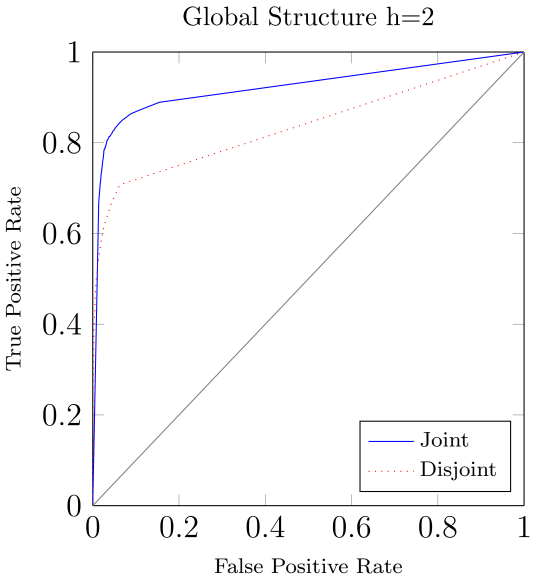


Figure B.31: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 7 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.9$)

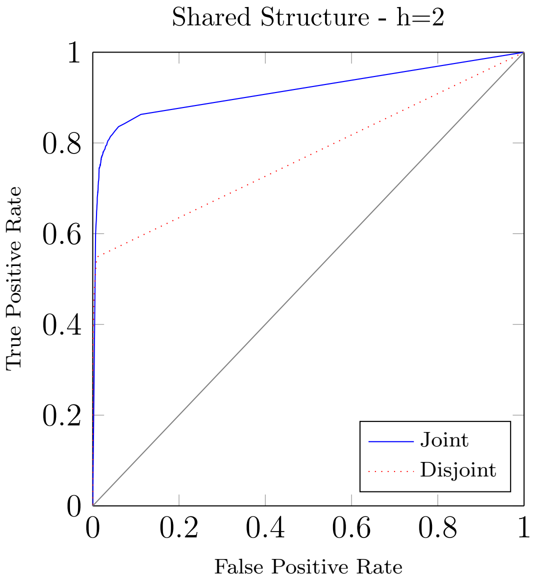


Figure B.32: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 7 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.9$)

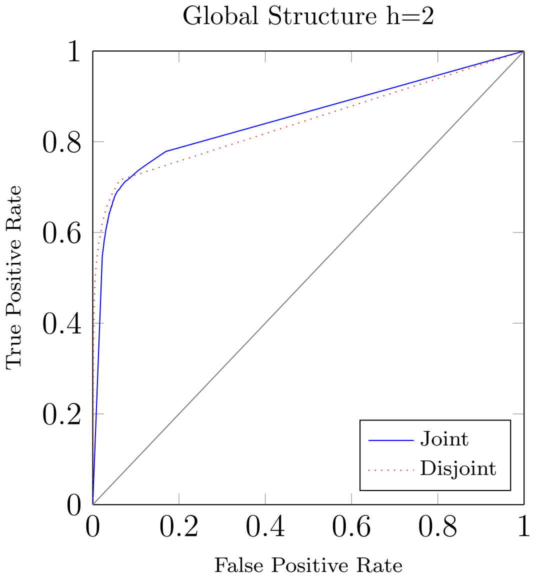


Figure B.33: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 8 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.5$)

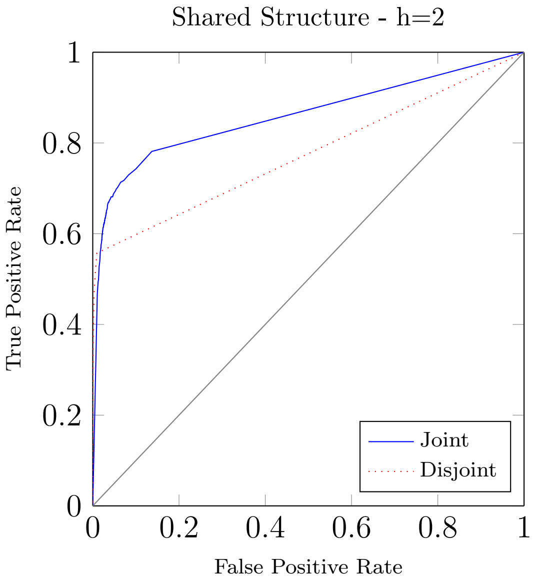


Figure B.34: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 8 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.5$)

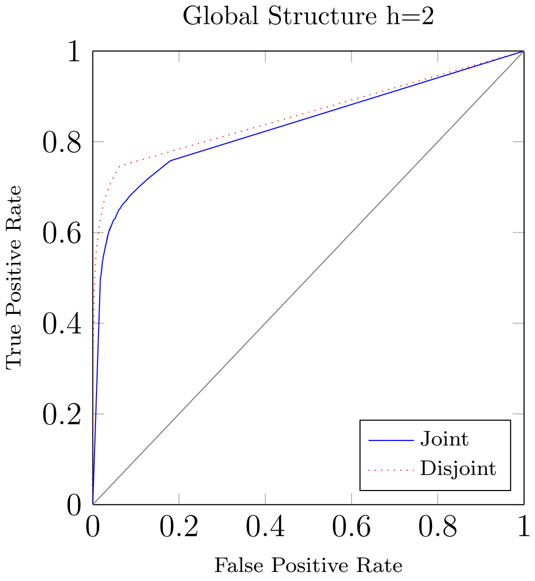


Figure B.35: Joint vs Disjoint with $h = 2$: Overall structure recognition for simulation 9 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.1$)

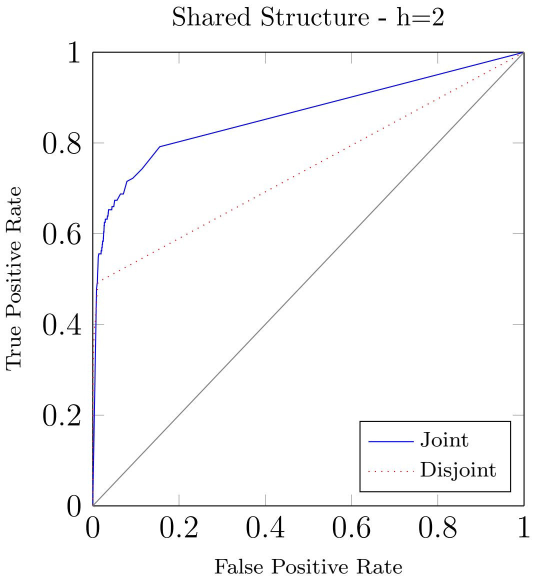


Figure B.36: Joint vs Disjoint with $h = 2$: Shared structure recognition for simulation 9 ($q = 100$, $n_0 = n_1 = 50$, $s = 0.1$)

.

B.1.2 Generating Graphs

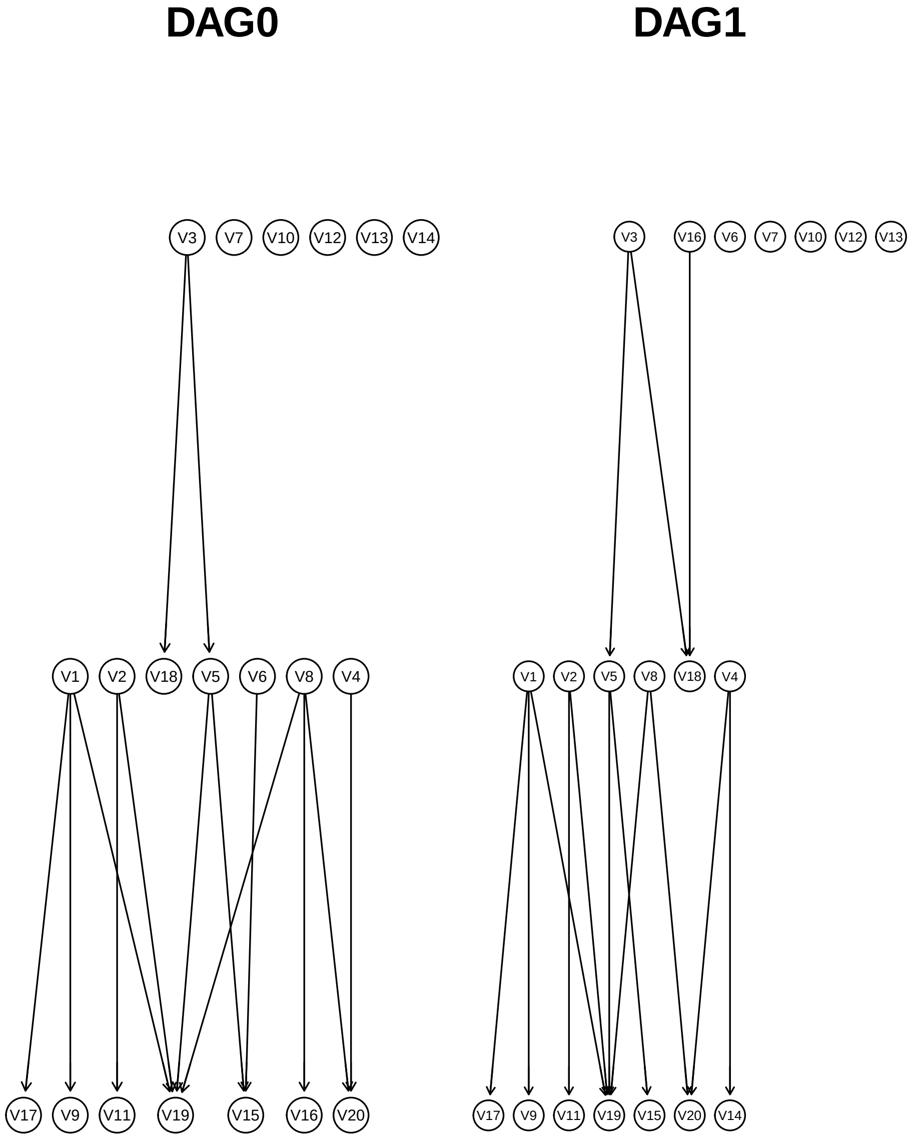


Figure B.37: True graphs for simulation 1

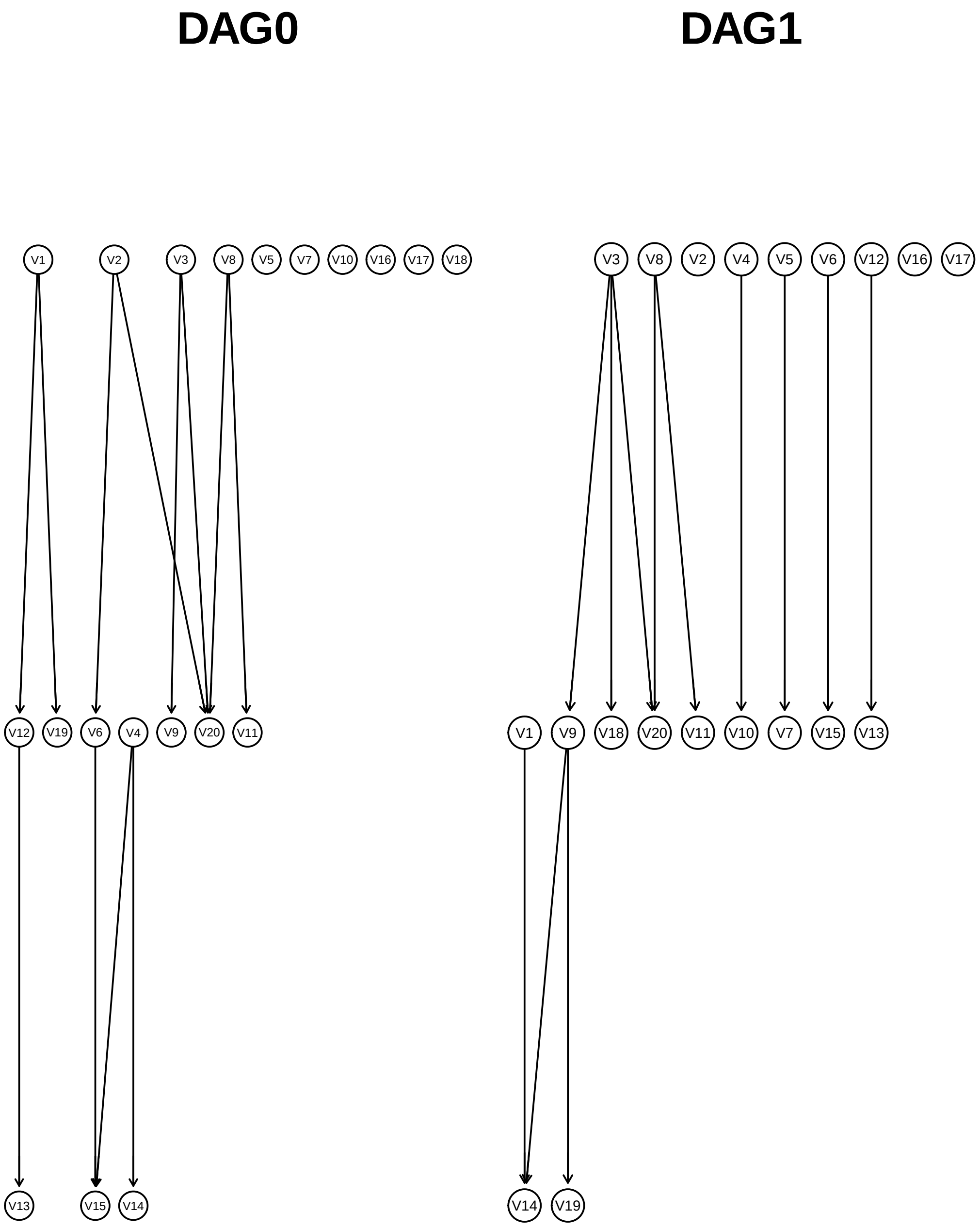
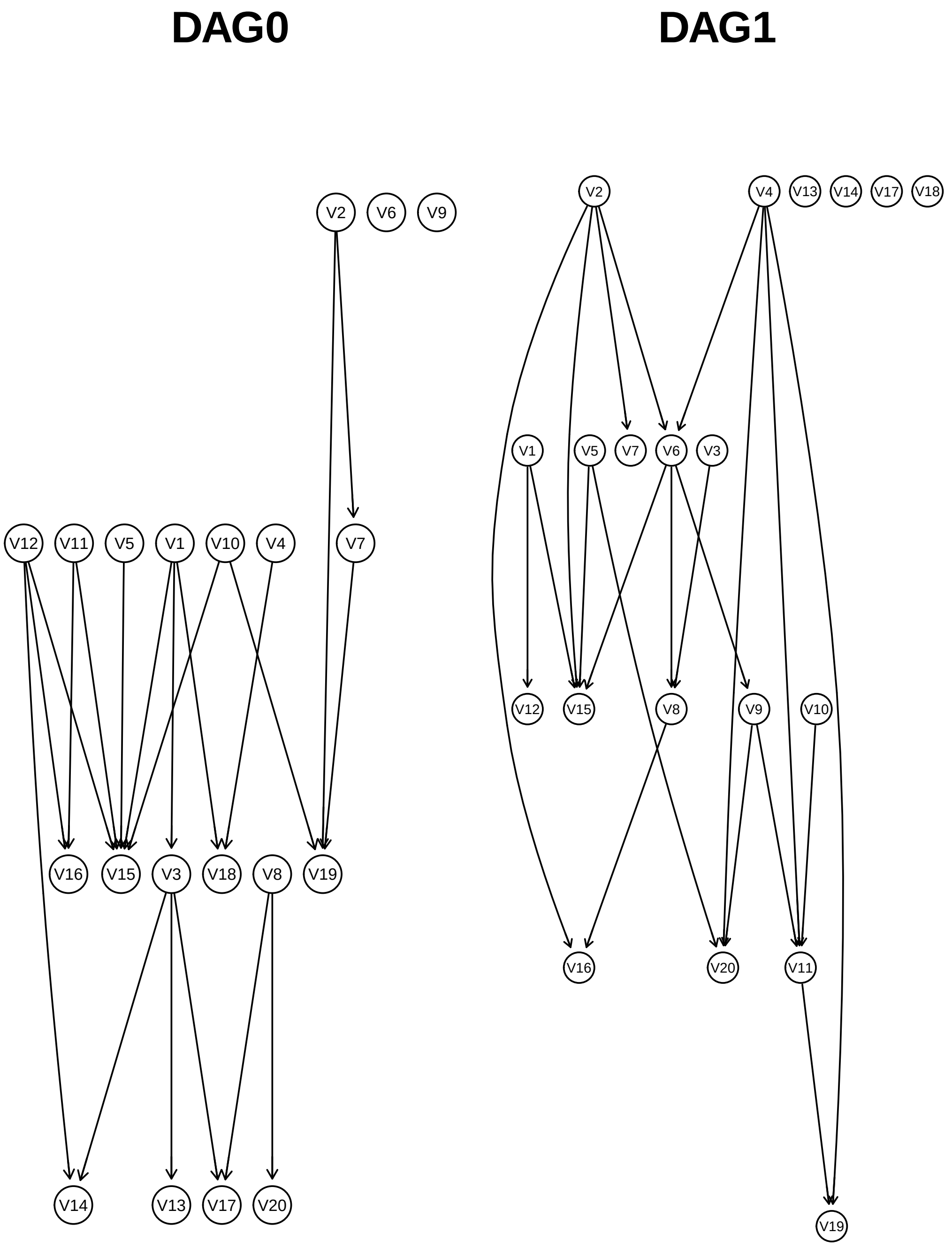


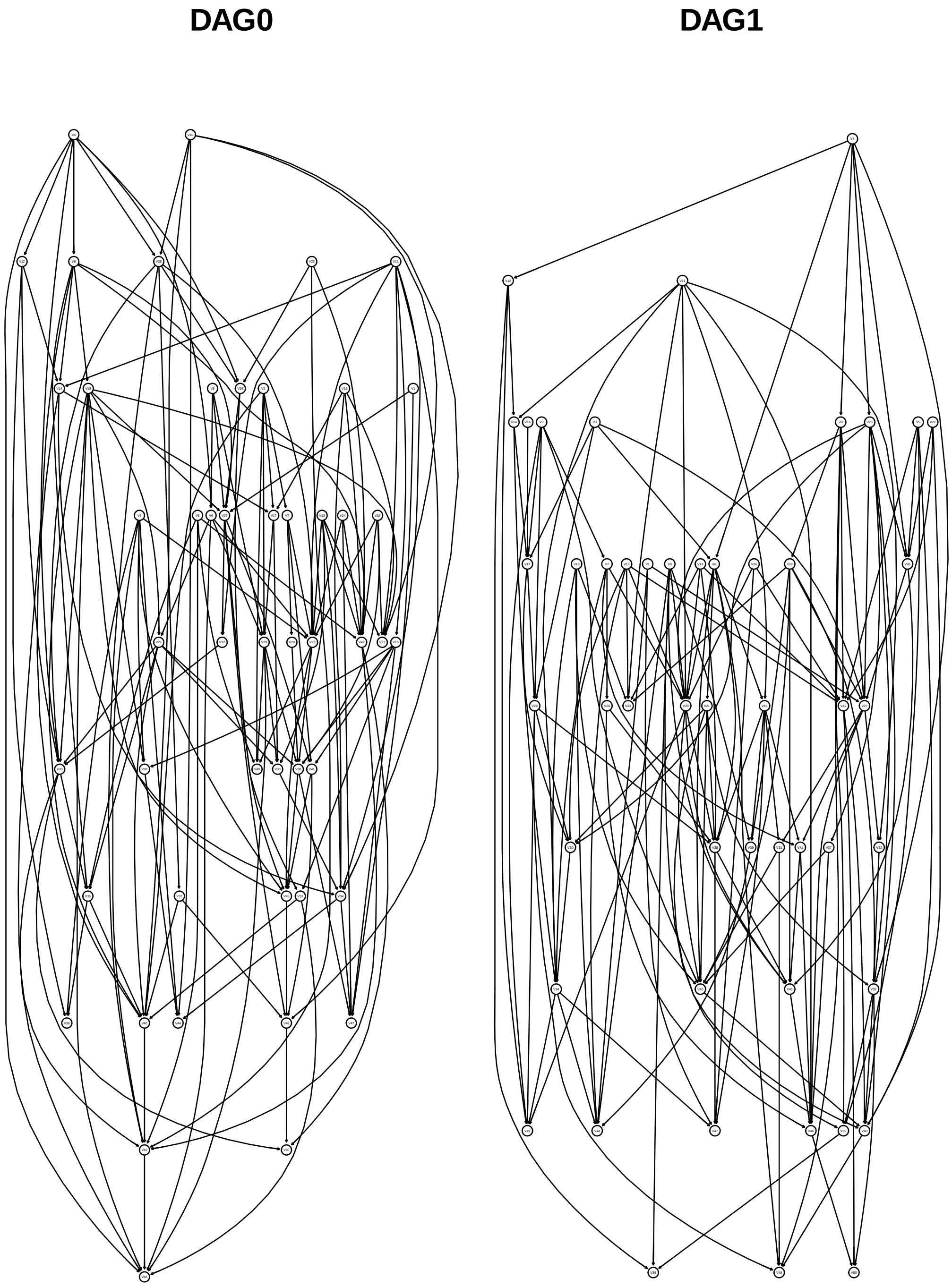
Figure B.38: True graphs for simulation 2



DAG0

DAG1

Figure B.39: True graphs for simulation 3



DAG0

DAG1

Figure B.40: True graphs for simulation 4

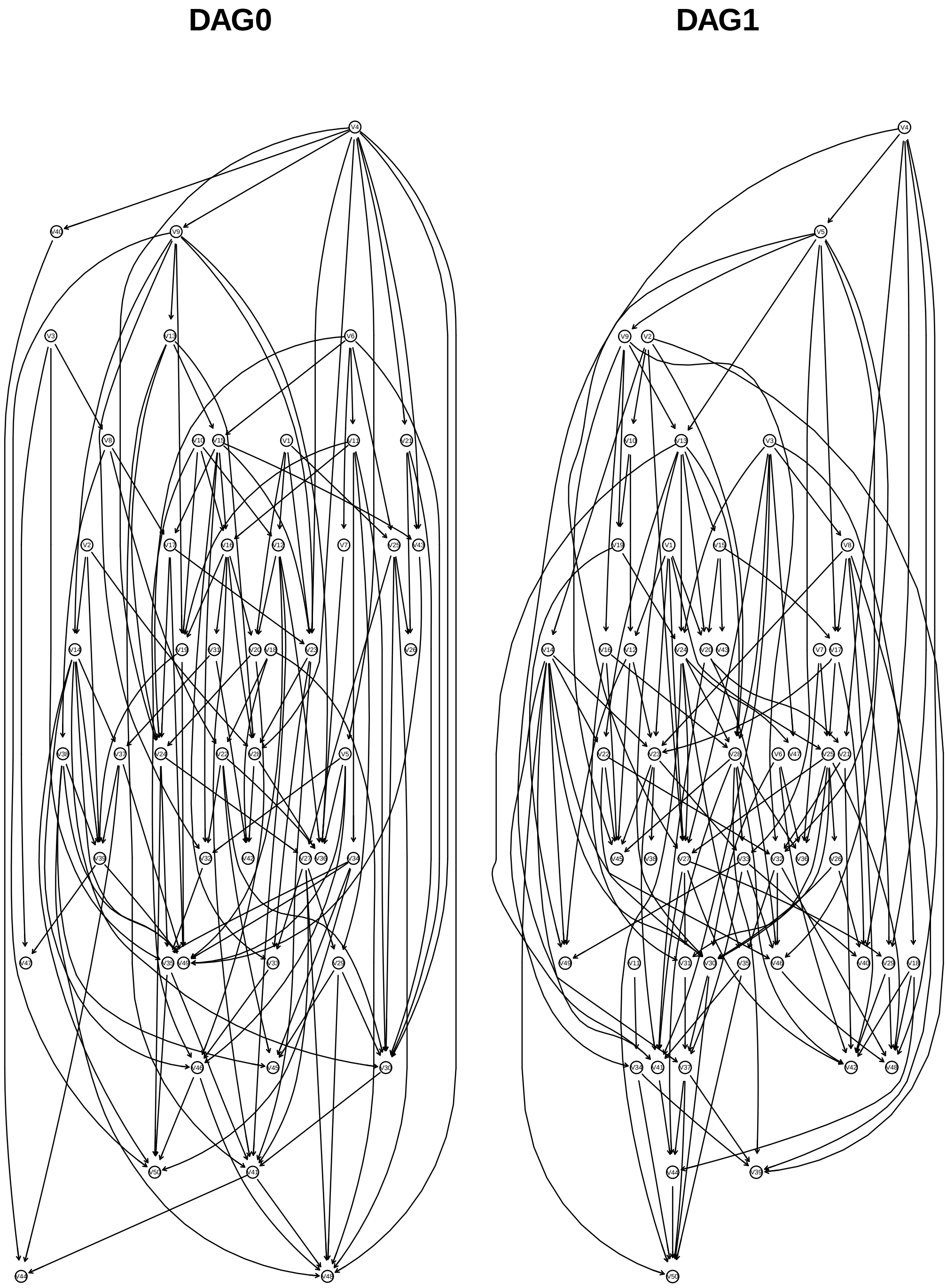


Figure B.41: True graphs for simulation 5

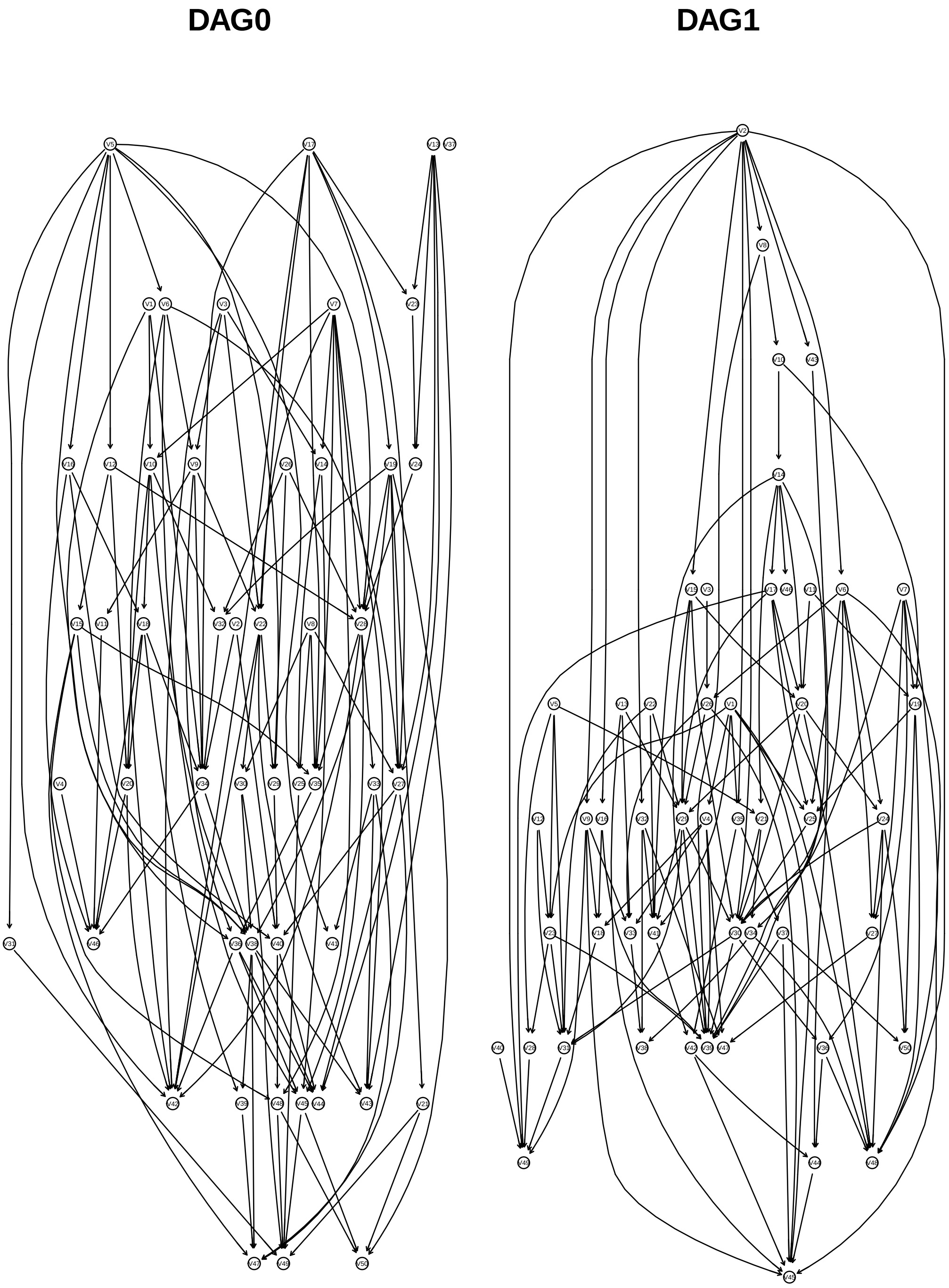
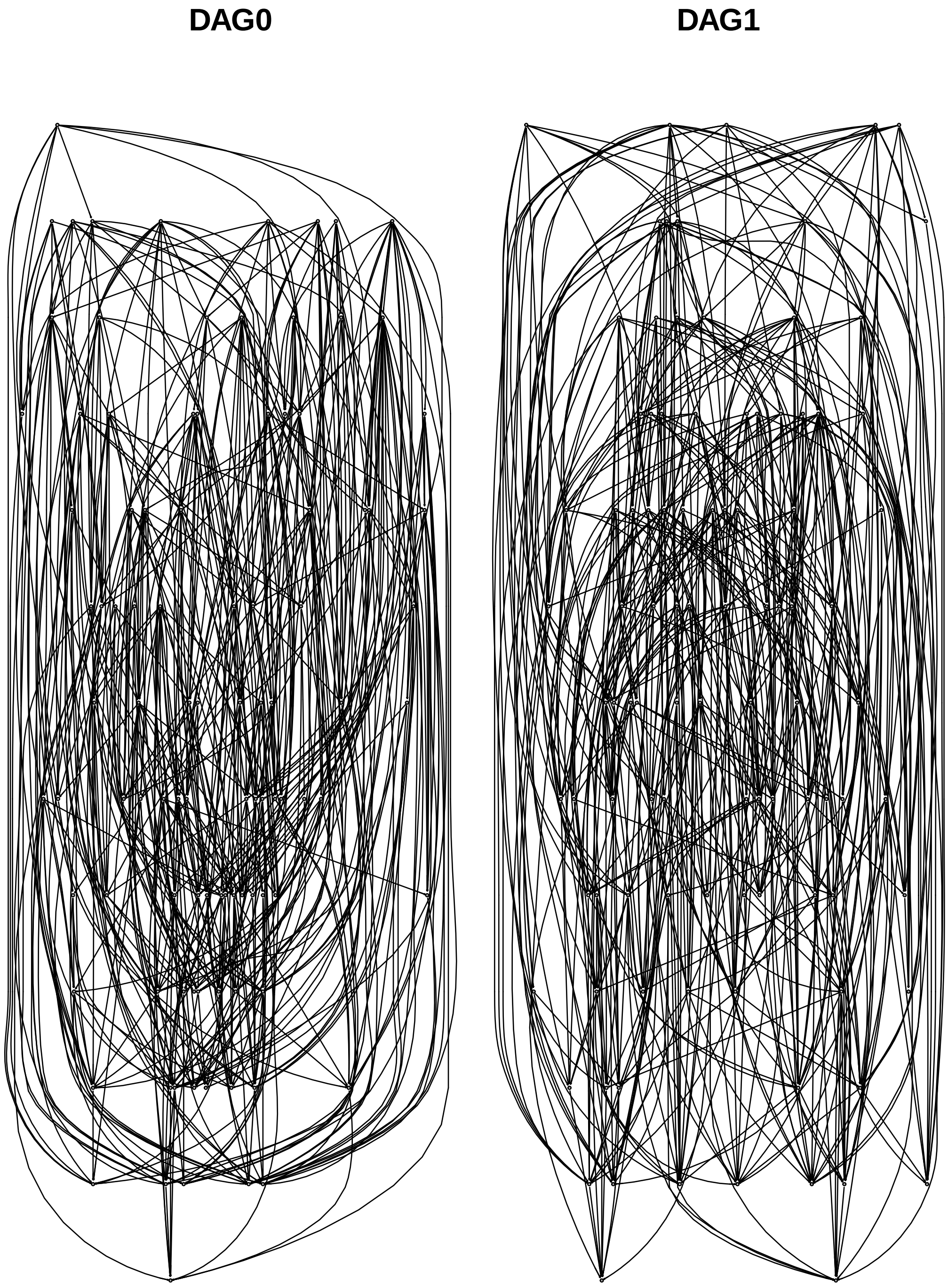


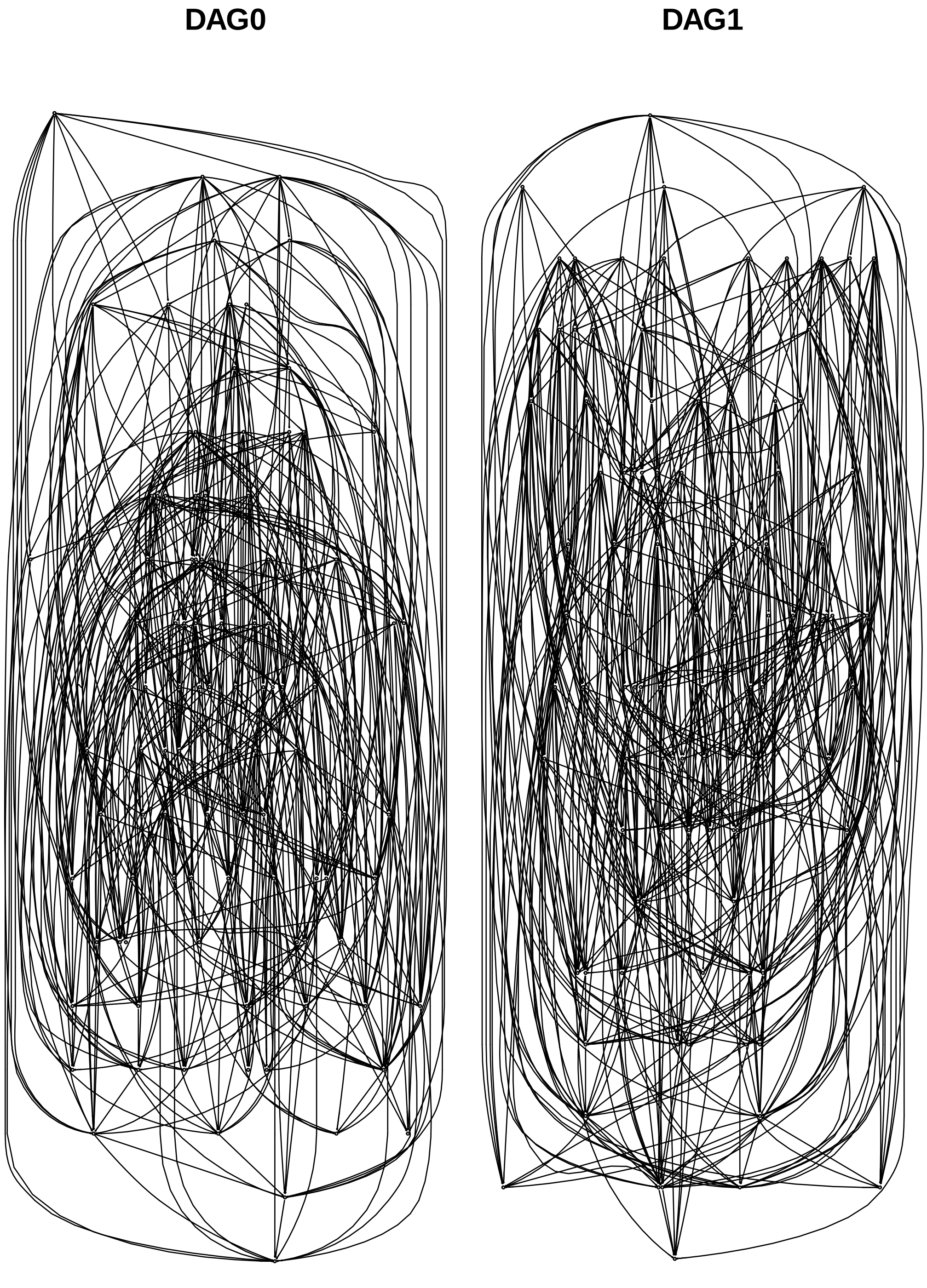
Figure B.42: True graphs for simulation 6



DAG0

DAG1

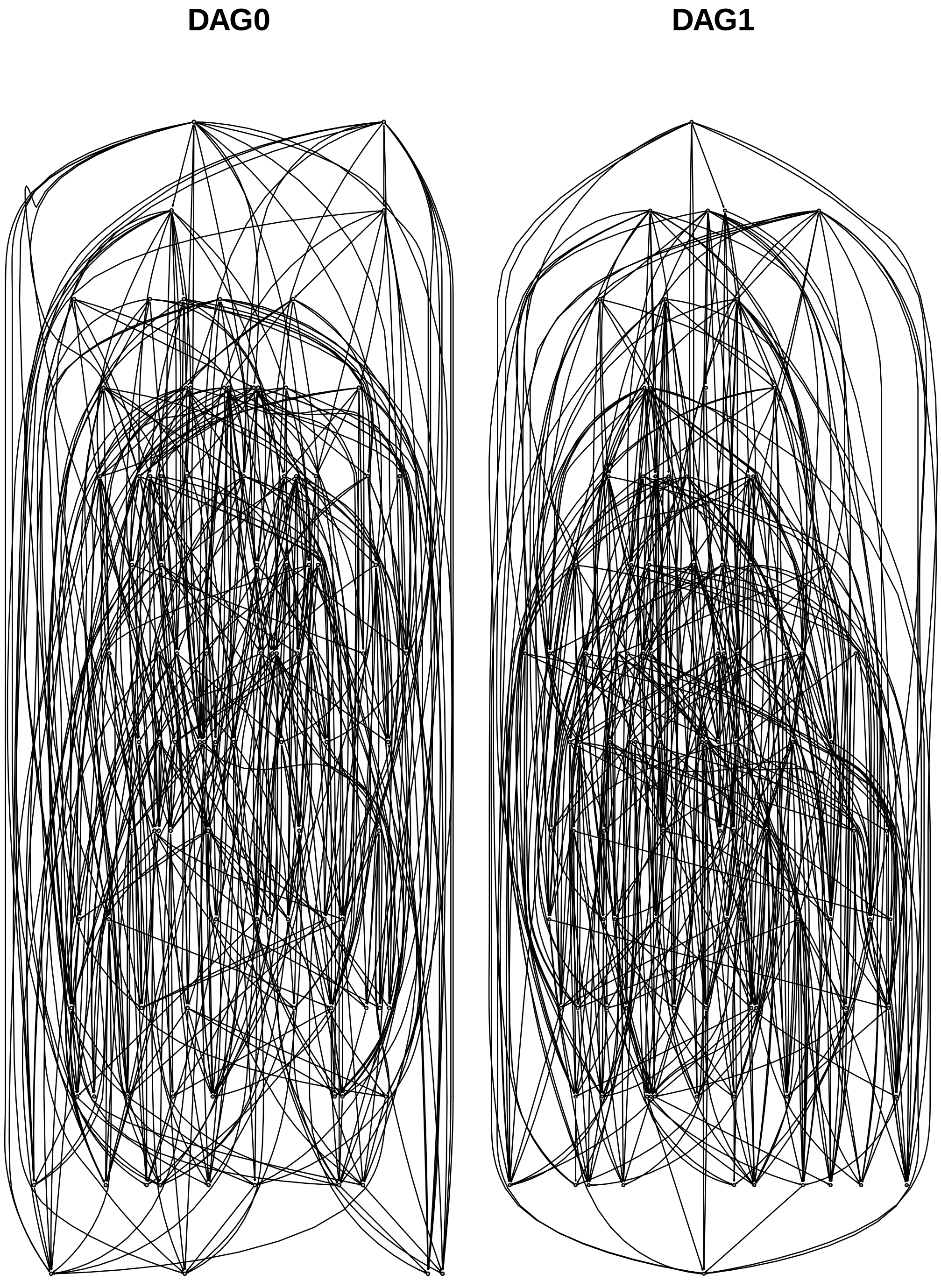
Figure B.43: True graphs for simulation 7



DAG0

DAG1

Figure B.44: True graphs for simulation 8



DAG0

DAG1

Figure B.45: True graphs for simulation 9

B.2 Comparison with Peterson *et al.* (2014)

B.2.1 ROC curves

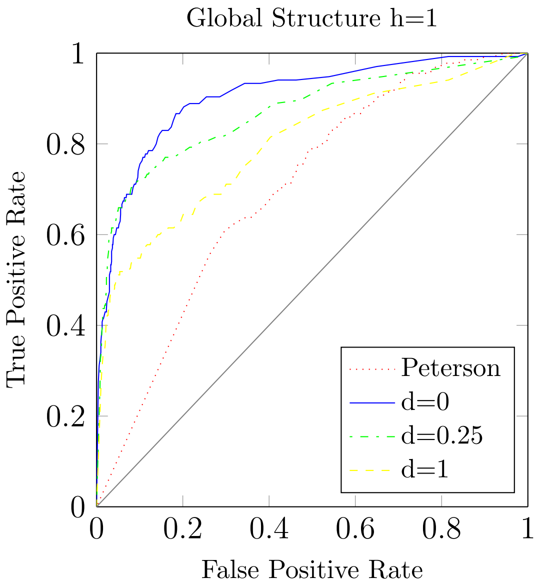


Figure B.46: Joint vs Benchmark: Overall structure recognition for simulation 1 ($q = 20$, $n_0 = n_1 = 20$, $z = 0.1$, $s = 0.8$)

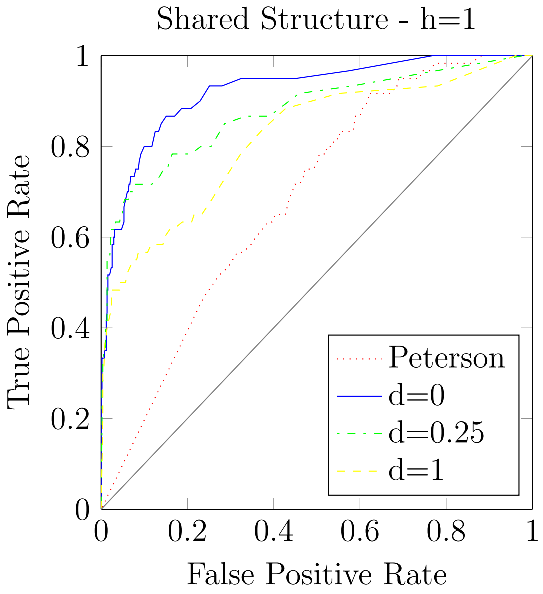


Figure B.47: Joint vs Benchmark: Shared structure recognition for simulation 1 ($q = 20$, $n_0 = n_1 = 20$, $z = 0.1$, $s = 0.8$)

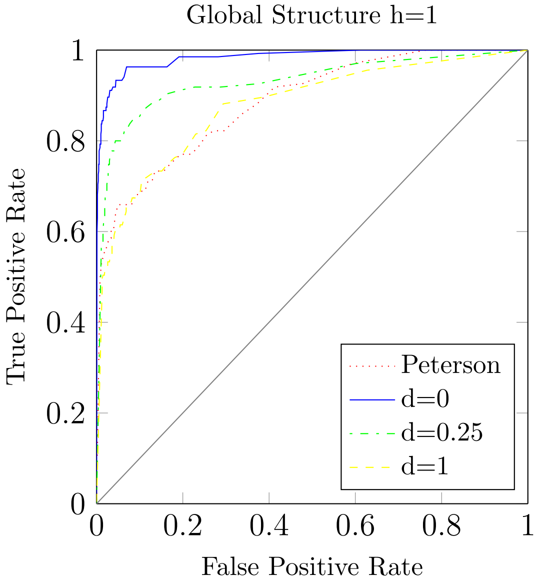


Figure B.48: Joint vs Benchmark: Overall structure recognition for simulation 2 ($q = 20$, $n_0 = n_1 = 50$, $z = 0.1$, $s = 0.8$)

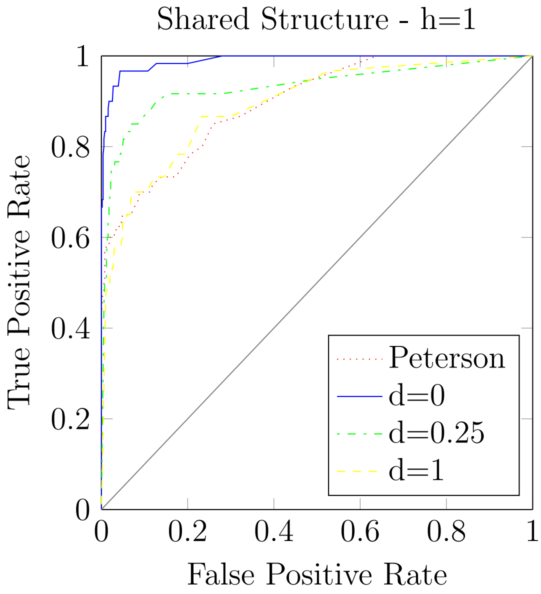


Figure B.49: Joint vs Benchmark: Shared structure recognition for simulation 2 ($q = 20$, $n_0 = n_1 = 50$, $z = 0.1$, $s = 0.8$)

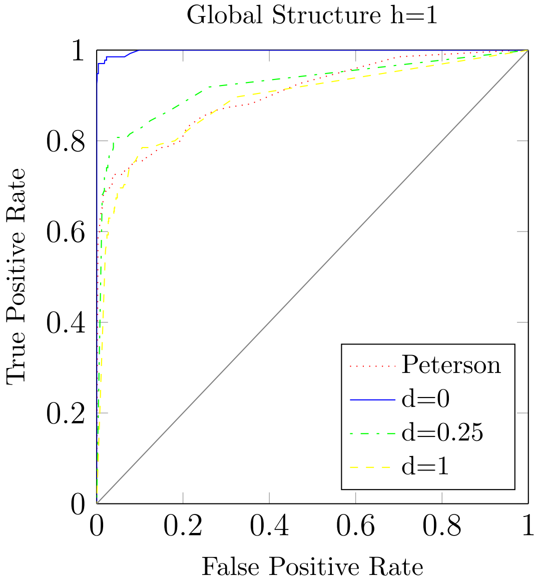


Figure B.50: Joint vs Benchmark: Overall structure recognition for simulation 3 ($q = 20$, $n_0 = n_1 = 100$, $z = 0.1$, $s = 0.8$)

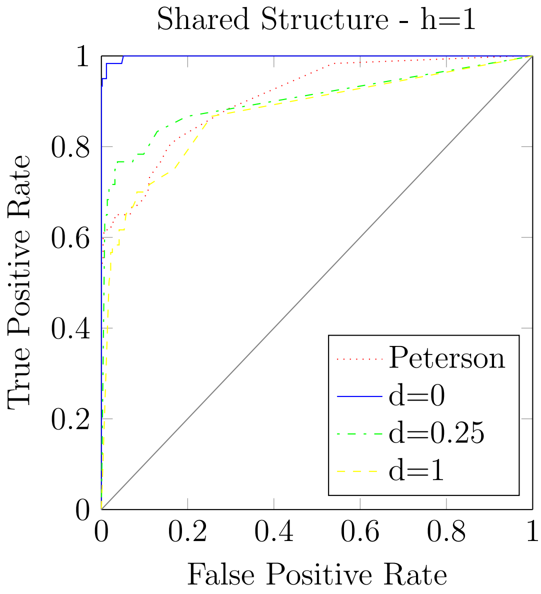


Figure B.51: Joint vs Benchmark: Shared structure recognition for simulation 3 ($q = 20$, $n_0 = n_1 = 100$, $z = 0.1$, $s = 0.8$)

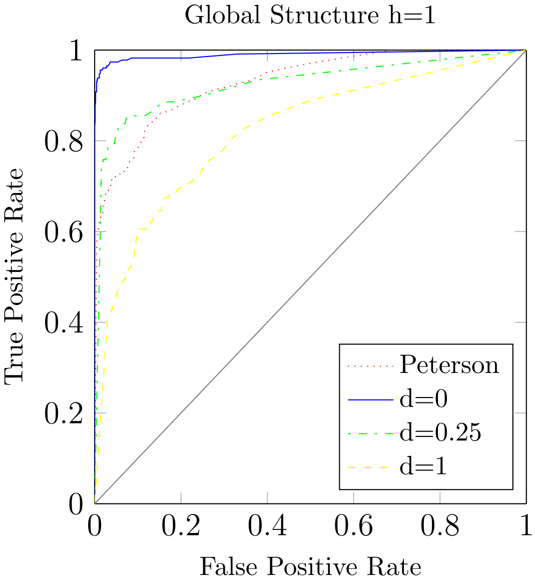


Figure B.52: Joint vs Benchmark: Overall structure recognition for simulation 4 ($q = 20$, $n_0 = n_1 = 100$, $z = 0.2$, $s = 0.8$)

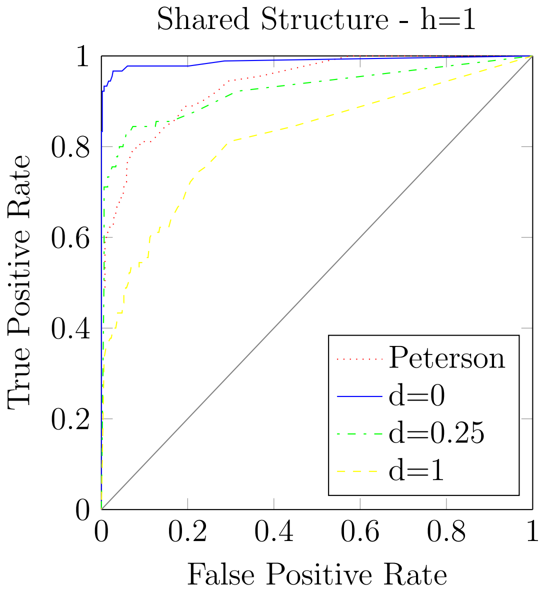


Figure B.53: Joint vs Benchmark: Shared structure recognition for simulation 4 ($q = 20$, $n_0 = n_1 = 100$, $z = 0.2$, $s = 0.8$)

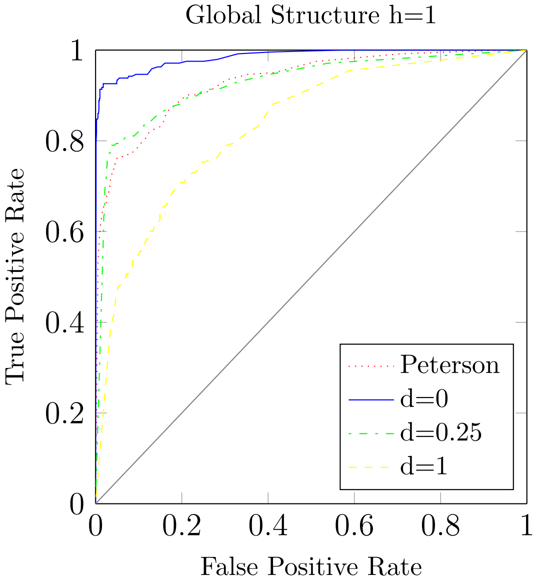


Figure B.54: Joint vs Benchmark: Overall structure recognition for simulation 5 ($q = 20$, $n_0 = n_1 = 100$, $z = 0.2$, $s = 0.5$)

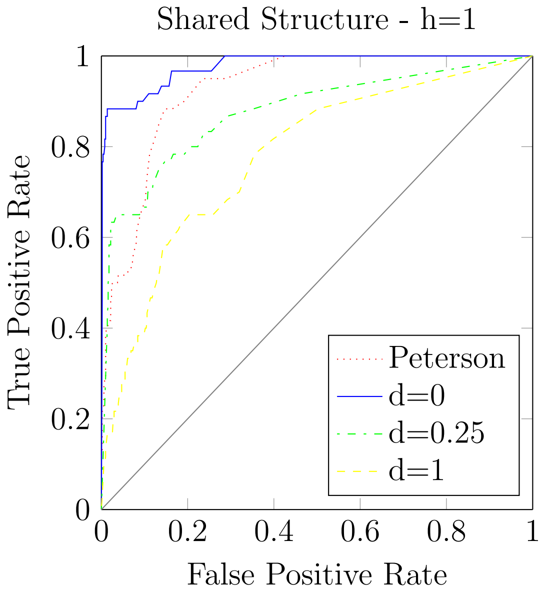


Figure B.55: Joint vs Benchmark: Shared structure recognition for simulation 5 ($q = 20$, $n_0 = n_1 = 100$, $z = 0.2$, $s = 0.5$)

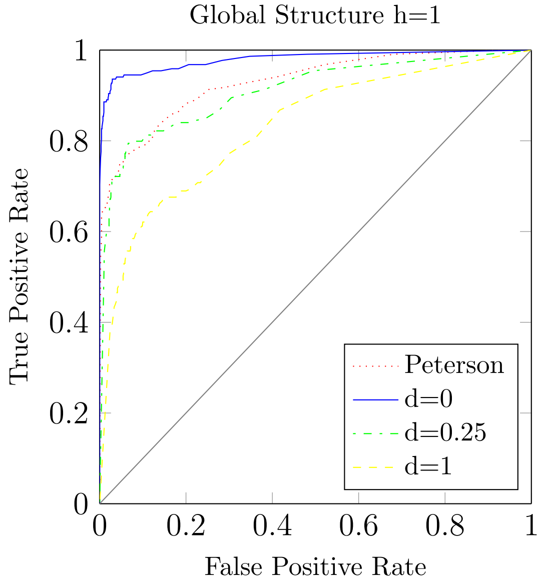


Figure B.56: Joint vs Benchmark: Overall structure recognition for simulation 6 ($q = 20$, $n_0 = n_1 = 100$, $z = 0.2$, $s = 0.3$)

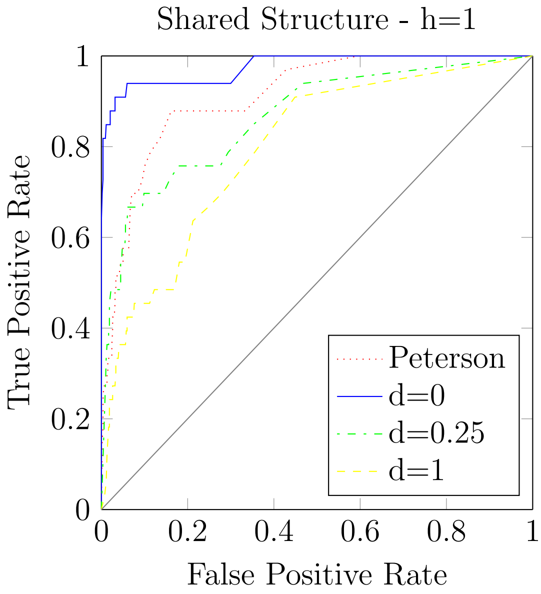


Figure B.57: Joint vs Benchmark: Shared structure recognition for simulation 6 ($q = 20$, $n_0 = n_1 = 100$, $z = 0.2$, $s = 0.3$)

B.2.2 Generating Graphs

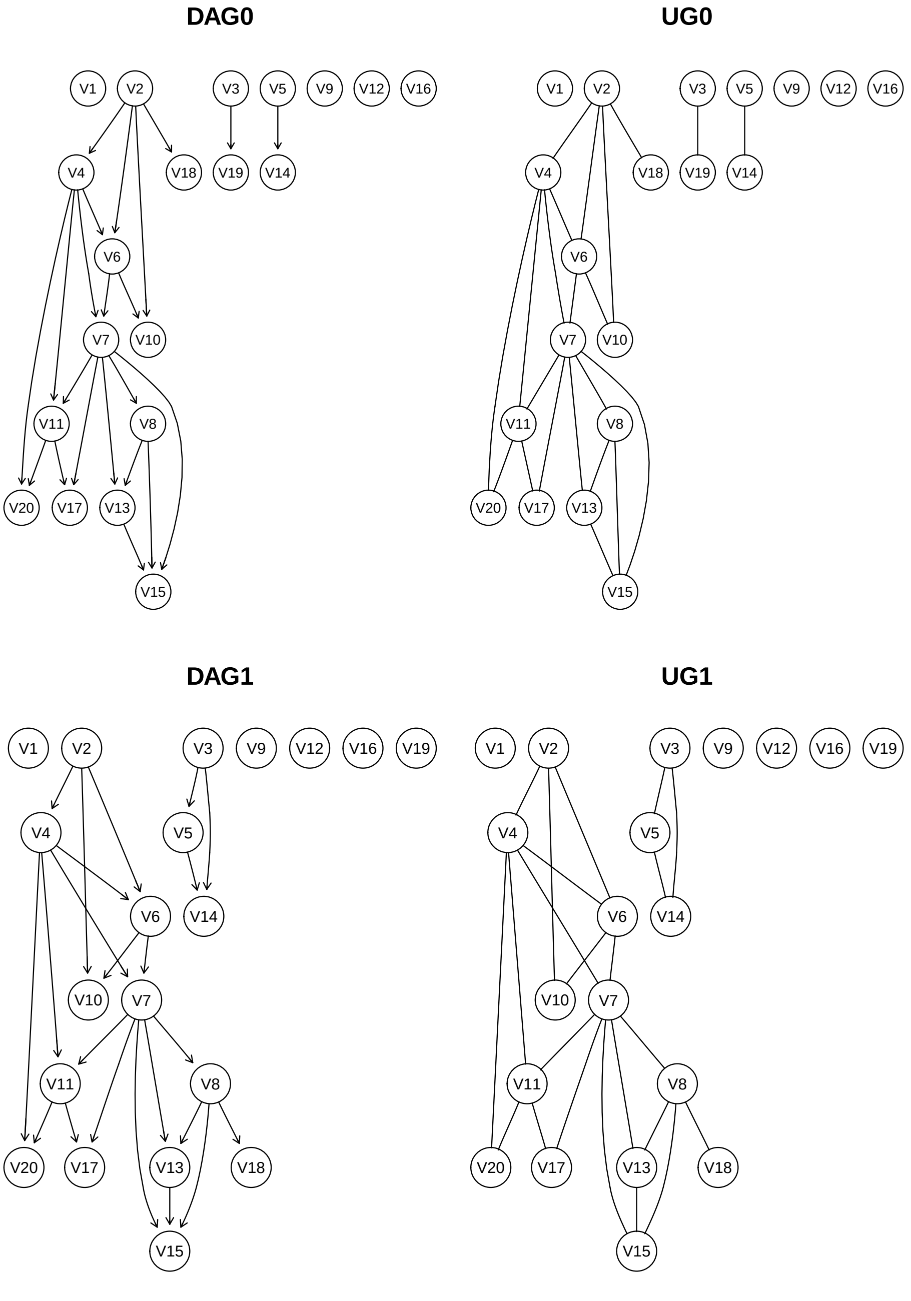
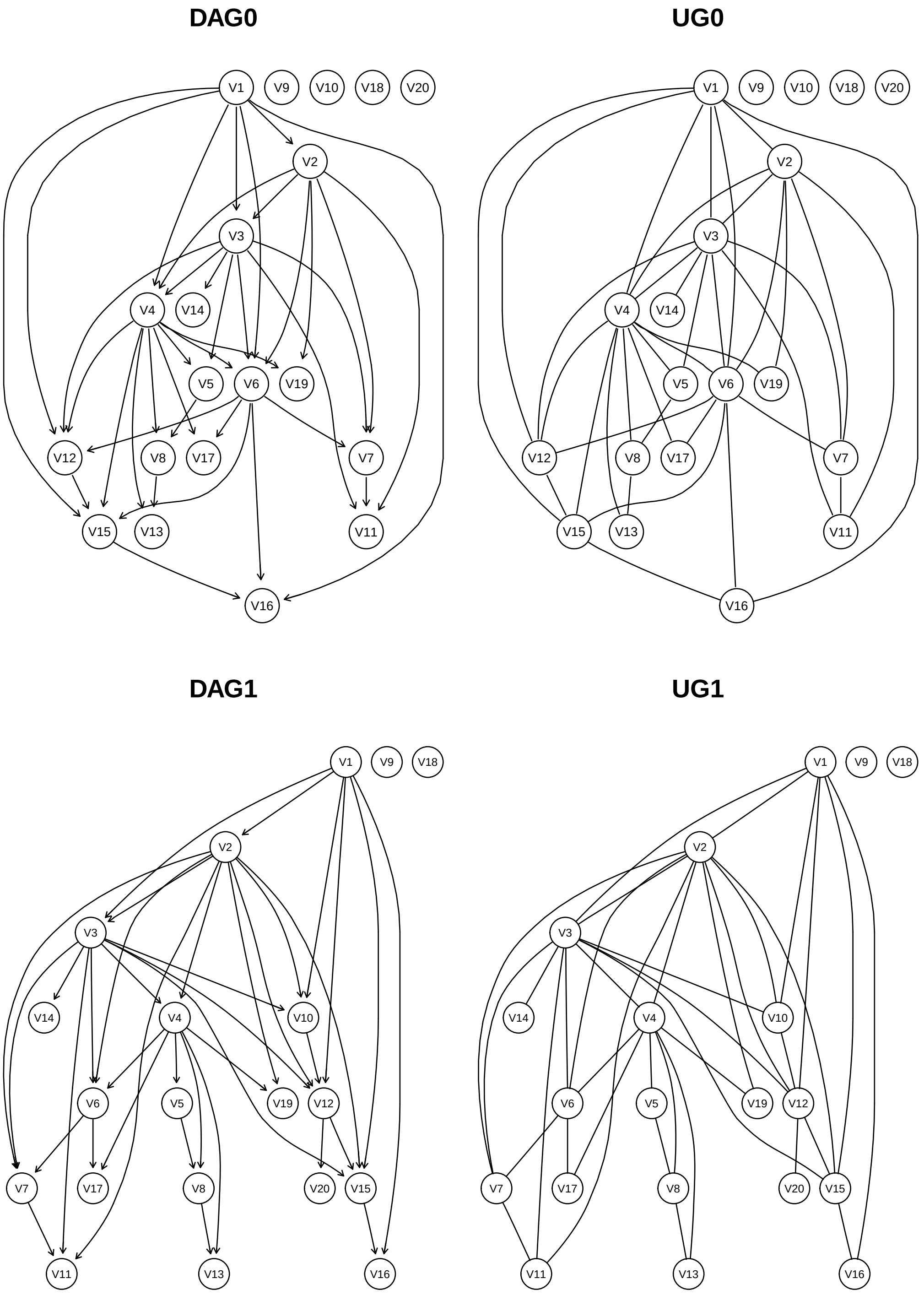


Figure B.58: True graphs for simulation 1,2 and 3.


DAG0

UG0

DAG1

UG1

Figure B.59: True graphs for simulation 4.

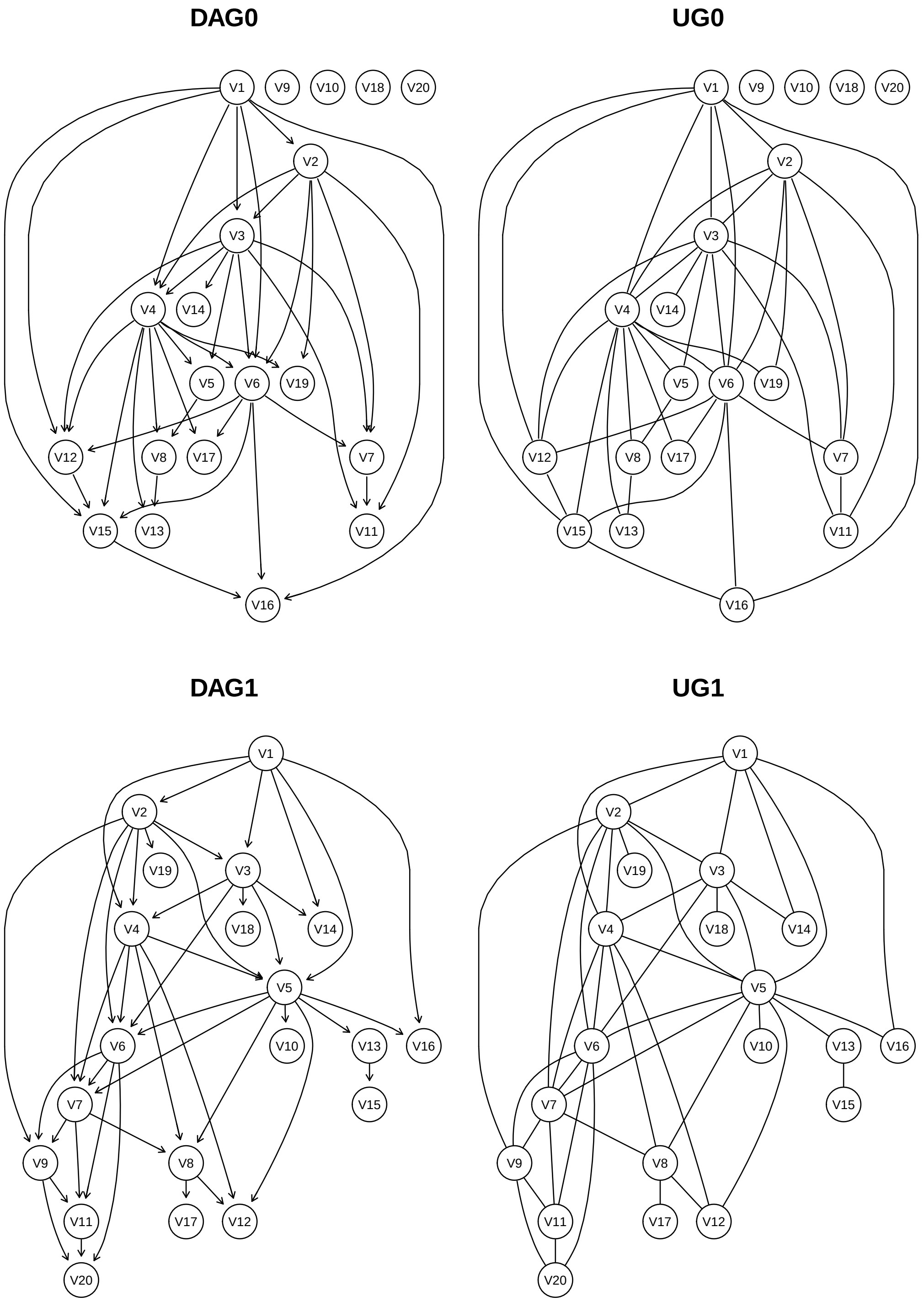
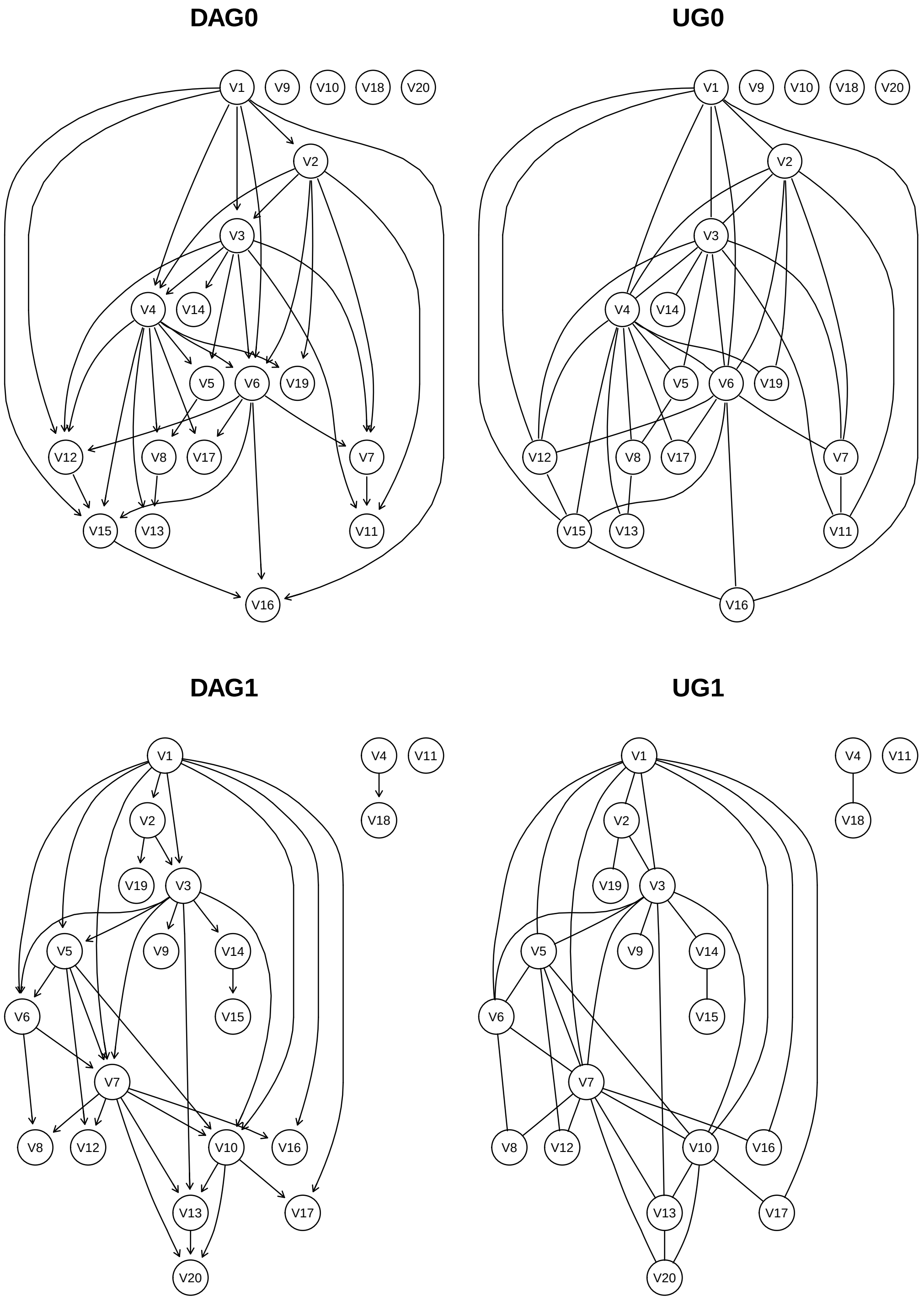


Figure B.60: True graphs for simulation 5.


DAG0

UG0

DAG1

UG1

Figure B.61: True graphs for simulation 6.

Bibliography

Altomare, D., G. Consonni, and L. La Rocca, 2013 Objective Bayesian Search of Gaussian Directed Acyclic Graphical Models for Ordered Variables with Non-Local Priors. Biometrics *69*(2)**:** 478—487.

Andersson, S. A., D. Madigan, M. D. Perlman, and others, 1997 A characterization of Markov equivalence classes for acyclic digraphs. The Annals of Statistics *25*(2)**:** 505–541.

Barber, D., 2012 *Bayesian Reasoning and Machine Learning*. Cambridge University Press.

Barbieri, M. M. and J. O. Berger, 2004 Optimal predictive model selection. The Annals of Statistics *32*(3)**:** 870–897.

Berger, J. O. and G. Molina, 2005 Posterior model probabilities via path-based pairwise priors. Stat. Neerl. **59:** 3–15.

Berger, J. O. and L. R. Pericchi, 1996 The intrinsic Bayes factor for model selection and prediction. Journal of the American Statistical Association *91*(433)**:** 109–122.

Berger, J. O., L. R. Pericchi, J. K. Ghosh, T. Samanta, F. D. Santis, J. O. Berger, and L. R. Pericchi, 2001 Objective Bayesian Methods for Model Selection: Introduction and Comparison. Lecture Notes-Monograph Series **38:** pp. 135–207.

Buntine, W., 1991 Theory refinement on Bayesian networks. In *Proceedings of the Seventh conference on Uncertainty in Artificial Intelligence*, pp. 52–60. Morgan Kaufmann Publishers Inc.

Castillo, I. and A. van der Vaart, 2012, 08)Needles and Straw in a Haystack: Posterior concentration for possibly sparse sequences. The Annals of Statistics *40*(4)**:** 2069–2101.

Consonni, G. and L. La Rocca, 2011 Moment priors for Bayesian model choice with applications to directed acyclic graphs. In J. M. Bernardo, M. J. Bayarri, J. O. Berger, A. P. Dawid, D. Heckerman, A. Smith, and M. West (Eds.), *Bayesian Statistics 9*, pp. 119–144. Oxford University Press.

Cox, D. R., 1961 Tests of Separate Families of Hypotheses. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*, Berkeley, Calif., pp. 105–123. University of California Press.

Friedman, J., T. Hastie, and R. Tibshirani, 2008 Sparse inverse covariance estimation with the graphical lasso. Biostatistics *9*(3)**:** 432–441.

Friedman, N. and D. Koller, 2003 Being Bayesian about Bayesian Network Structure: A Bayesian Approach to Structure Discovery in Bayesian Networks. Machine Learning *50*(1–2)**:** 95–125. Full version of UAI 2000 paper.

Friedman, N., I. Nachman, and D. Peér, 1999 Learning bayesian network structure from massive datasets: the sparse candidate algorithm. In *Proceedings of the Fifteenth conference on Uncertainty in artificial intelligence*, pp. 206–215. Morgan Kaufmann Publishers Inc.

Fronk, E.-M. and P. Giudici, 2004 Markov Chain Monte Carlo Model Selection for DAG Models. Statistical Methods & Applications *13*(3)**:** 259–273.

Geiger, D. and D. Heckerman, 1994 Learning Gaussian networks. In *Proceedings of the Tenth international conference on Uncertainty in artificial intelligence*, UAI'94, San Francisco, CA, USA, pp. 235–243. Morgan Kaufmann Publishers Inc.

Geiger, D. and D. Heckerman, 2002, 10)Parameter priors for directed acyclic graphical models and the characterization of several probability distributions. The Annals of Statistics *30*(5)**:** 1412–1440.

George, E. I. and R. E. McCulloch, 1993 Variable Selection Via Gibbs Sampling. Journal of the American Statistical Association **88:** 881–889.

Green, P., 1995 Reversible Jump Markov Chain Monte Carlo Computation and Bayesian Model Determination. Biometrika **82:** 711–732.

Grzegorczyk, M., 2010 An Introduction to Gaussian Bayesian Networks. In Q. Yan (Ed.), *Systems Biology in Drug Discovery and Development*, Volume 662 of *Methods in Molecular Biology*, pp. 121–147. Humana Press.

Guo, J., E. Levina, G. Michailidis, and J. Zhu, 2011 Joint estimation of multiple graphical models. Biometrika *98*(1): 1–15.

Hammersley, J. M. and P. E. Clifford, 1971 Markov fields on finite graphs and lattices. Unpublished manuscript.

Heckerman, D., D. Geiger, and D. M. Chickering, 1995 Learning Bayesian networks: The combination of knowledge and statistical data. Machine learning *20*(3): 197–243.

Johnson, V. E. and D. Rossell, 2010 On the use of non-local prior densities in Bayesian hypothesis tests. Journal of the Royal Statistical Society: Series B (Statistical Methodology) *72*(2): 143–170.

Johnson, V. E. and D. Rossell, 2012 Bayesian Model Selection in High-Dimensional Settings. J. Amer. Statist. Assoc. *107*(498): 649–660.

Kalisch, M., M. Mächler, D. Colombo, M. H. Maathuis, and P. Bühlmann, 2012 Causal Inference Using Graphical Models with the R Package pcalg. Journal of Statistical Software *47*(11): 1–26.

Kass, R. E. and A. E. Raftery, 1995 Bayes factors. Journal of the american statistical association *90*(430): 773–795.

Lauritzen, S., 1996 *Graphical Models*. Clarendon Press.

O'Hagan, A., 1995 Fractional Bayes Factors for Model Comparison. Journal of the Royal Statistical Society. Series B (Methodological) *57*(1): pp. 99–138.

Pearl, J. and A. Paz, 1987 Graphoids: a graph based logic for reasoning about relevancy relations. In *In Advance in Artificial Intelligence-II*, pp. 357–363. North-Holland.

Peterson, C., F. Stingo, and M. Vannucci, 2014 Bayesian Inference of Multiple Gaussian Graphical Models. Journal of the American Statistical Association *0*(ja): 00–00.

Rousseau, J., 2007 Approximating Interval hypothesis : p-values and Bayes factors. In J. Bernardo, M. Bayarri, J. Berger, A. Dawid, D. Heckerman, A. Smith, and M. West (Eds.), *Bayesian Statistics 8*. Oxford University Press.

Roverato, A., 2002 Hyper Inverse Wishart Distribution for Nondecomposable Graphs and its Application to Bayesian Inference for Gaussian Graphical Models. Scandinavian Journal of Statistics *29*(3): 391–411.

Scott, J. G. and J. O. Berger, 2010 Bayes and empirical-Bayes multiplicity adjustment in the variable-selection problem. The Annals of Statistics *38*(5): 2587–2619.

Scott, J. G. and C. M. Carvalho, 2008 Feature-Inclusion Stochastic Search for Gaussian Graphical Models. Journal of Computational and Graphical Statistics *17*(4): 790–808.

Shachter, R. D. and C. R. Kenley, 1989 Gaussian Influence Diagrams. Management Science *35*(5): 527–550.

Teyssier, M. and D. Koller, 2005 Ordering-Based Search: A Simple and Effective Algorithm for Learning Bayesian Networks. In *Proceedings of the Twenty-First Annual Conference on Uncertainty in Artificial Intelligence (UAI-05)*, pp. 584–590. AUAI Press.

Verdinelli, I. and L. Wasserman, 1996 Bayes Factors, Nuisance Parameters and Imprecise Tests. In J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith (Eds.), *Proceedings of Bayesian Statistics 5*, pp. 765–771. Oxford University Press.

Whittaker, J., 1990 *Graphical models in applied multivariate statistics.* Wiley series in probability and mathematical statistics: Probability and mathematical statistics. Wiley.

Yajima, M., D. Telesca, Y. Ji, and P. Müller, 2012 Differential patterns of interactions and Gaussian graphical models. Technical Report 91, COBRA Preprint Series.