Learning the piece-wise constant graph structure of a varying Ising model

Batiste Le Bars [1] **Pierre Humbert** [1] **Argyris Kalogeratos** [1] **Nicolas Vayatis** [1]

Abstract

This work focuses on the estimation of multiple change-points in a time-varying Ising model that evolves piece-wise constantly. The aim is to identify both the moments at which significant changes occur in the Ising model, as well as the underlying graph structures. For this purpose, we propose to estimate the neighborhood of each node by maximizing a penalized version of its conditional log-likelihood. The objective of the penalization is twofold: it imposes sparsity in the learned graphs and, thanks to a fused-type penalty, it also enforces them to evolve piece-wise constantly. Using few assumptions, we provide two change-points consistency theorems. Those are the first in the context of unknown number of change-points detection in time-varying Ising model. Finally, experimental results on several synthetic datasets and a real-world dataset demonstrate the performance of our method.

1. Introduction

Graphs are fundamental tools to model and study static or varying relationships between variables of potentially high-dimensional vector data. They have many applications in physics, computer vision and statistics (Choi et al., 2010; Marbach et al., 2012). In the static scenario, learning relationships between variables is referred to as *graph inference* and emerges in many fields such as in graph signal processing (Dong et al., 2016; Le Bars et al., 2019), in probabilistic modeling, or in physics and biology (Rodriguez et al., 2011; Du et al., 2012). In this work, we consider a probabilistic framework where the observed data are drawn from an *Ising model*, a discrete *Markov Random Field* (MRF) with $\{-1, 1\}$-outputs. MRF are undirected probabilistic graphical models (Koller et al., 2009) where a set of random variables is represented as different nodes of a graph. An edge between two nodes in this graph indicates the conditional dependency between the two corresponding random variables, given the other variables.

Learning the structure of an MRF using a set of observations has been widely investigated (Banerjee et al., 2008; Meinshausen et al., 2006). In particular for Gaussian graphical models (Yuan & Lin, 2007; Ren et al., 2015) with the well-known graphical lasso (Friedman et al., 2008). The Ising model inference task has also been addressed in the past (Höfling & Tibshirani, 2009; Ravikumar et al., 2010; Xue et al., 2012; Vuffray et al., 2016; Goel et al., 2019). However, previous methods do not consider the case where the underlying structure is evolving through time.

Over the past years, there has been a burst of interest in learning the structure of time-varying MRF (Hallac et al., 2017; Yang & Peng, 2019). This task combined with the *change-point detection*, which is the detection of the moments in time at which significant changes in the graph structure occur, is of particular interest. Those have been widely investigated for piece-wise constant Gaussian graphical models (Gibberd & Nelson, 2017; Wang et al., 2018; Londschien et al., 2019), in all types of the change-point detection objectives: single change-point (Bybee & Atchadé, 2018), multiple change-points (Gibberd & Roy, 2017), offline detection (Kolar & Xing, 2012; Gibberd & Nelson, 2017), online detection (Keshavarz et al., 2018), etc.

The advancements related to the time-varying Ising model are though limited. Especially, the combination of multiple change-points detection and structure inference has not been studied properly in the past. In (Ahmed & Xing, 2009; Kolar et al., 2010), the authors learn the parameters of a time-varying Ising model without looking for change-points since the network is allowed to change at each timestamp. In (Fazayeli & Banerjee, 2016), the authors assume that the change-point location is known and only focus on the inference of the structural changes between Ising models. More recently, the problem of detecting a single change-point has been studied in (Roy et al., 2017).

Contribution. This work focuses on the estimation of multiple change-points in a time-varying Ising model that evolves piece-wise constantly. The aim is to identify both the moments at which significant changes occur in the Ising

[1]Université Paris-Saclay, ENS Paris-Saclay, CNRS, Centre Borelli, F-91190 Gif-sur-Yvette, France. Correspondence to: Batiste Le Bars <batiste.lebars@cmla.ens-cachan.fr>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020.

model, as well as the underlying graph structure of the model among consecutive change-points. Our work extends the work in (Kolar & Xing, 2012; Gibberd & Nelson, 2017) on Gaussian graphical models, to the case of an Ising model. We also derive two change-points consistency theorems that, to our knowledge, we are the first to demonstrate. More specifically, our method follows a "node-wise regression" approach (Ravikumar et al., 2010) and estimates the neighborhood of each node by maximizing a penalized version of its conditional log-likelihood. The penalization allows us to efficiently recover sparse graphs and, thanks to the use of a group-fused penalty (Harchaoui & Lévy-Leduc, 2010; Bleakley & Vert, 2011; Kolar & Xing, 2012), as well to recover the change-points. The proposed method is referred as TVI-FL, which stands for Time-Varying Ising model identified with Fused and Lasso penalties.

Organization. The paper is organized as follows. First, we briefly recall important properties of the static Ising model and describe its piece-wise constant version. Second, we present our methodology for the inference of a piece-wise constant graph structure over time and the moments in time at which significant changes occur. Next, we present our main theoretical results that consist in two change-point consistency theorems. Finally, we demonstrate empirically, on multiple synthetic datasets and a real-world problem, that our method is the best suited to recover both the structure and the change-points.

2. The time-varying Ising model

The *static Ising model* is a discrete MRF with $\{-1, 1\}$-outputs. This model is defined by a graph $G = (V, E)$ where an edge between two nodes indicates that the two corresponding random variables are dependent given the other ones. We associate this graph to a symmetric weight matrix $\Omega \in \mathbb{R}^{p\times p}$ whose non-zero elements correspond to the set of edges E. Formally, we have $\omega_{ab} \neq 0$ iff $(a, b) \in E$ where ω_{ab} stands for the (a, b)-th element of Ω. An Ising model is thus entirely described by its associated weight matrix Ω. Let $X \sim \mathcal{I}(\Omega)$ be a random vector following an Ising model with weight matrix Ω. Let $x \in \{-1, 1\}^p$ be a realization and x_a, x_b respectively its a-th and b-th coordinates. Then, its probability function is given by:

$$\mathbb{P}_\Omega(X = x) = \frac{1}{Z(\Omega)} \exp\left\{\sum_{a<b} x_a x_b \omega_{ab}\right\}, \qquad (1)$$

where $Z(\Omega) = \sum_{x\in\{-1,1\}^p} \exp\left\{\sum_{a<b} x_a x_b \omega_{ab}\right\}$ is the normalizing constant. For clarity in the following we denote $\mathbb{P}_\Omega(X = x) = \mathbb{P}_\Omega(x)$.

A *time-varying Ising model* is defined by a set of n graphs $G^{(i)} = (V, E^{(i)})$, $i \in \{1, ..., n\}$ over a fixed set of nodes V through a time-varying set of edges $\{E^{(i)}\}_{i=1}^n$. Similarly to the static case, each $G^{(i)}$ is associated to a symmetric weight matrix $\Omega^{(i)} \in \mathbb{R}^{p\times p}$ and a distribution $\mathbb{P}_{\Omega^{(i)}}$ given by Eq. (1). A random variable associated to this model is a set of n independent random vectors $X^{(i)} \sim \mathcal{I}(\Omega^{(i)})$. A single realization is therefore a set of n vectors, each denoted by $x^{(i)} \in \{-1, 1\}^p$.

In the sequel, we assume in addition that the model is *piece-wise constant*, i.e. there exist a collection of D timestamps $\mathcal{D} \triangleq \{T_1, ..., T_D\} \subset \{2, ..., n\}$, sorted in ascending order, and a set of symmetric matrices $\{\Theta^{(j)}\}_{j=1}^{D+1}$ such that $\forall i \in \{1, ..., n\}$:

$$\Omega^{(i)} = \sum_{j=0}^{D} \Theta^{(j+1)} \mathbb{1}\{T_j \leq i < T_{j+1}\}, \qquad (2)$$

where $T_0 = 1$, $T_{D+1} = n + 1$. $\mathcal{D}$ thus corresponds to the set of change-points. According to Eq. (2), for a fixed $j \in \{0, ..., D\}$, the set $\left\{x^{(i)} : T_j \leq i < T_{j+1}\right\}$ contains i.i.d. vectors drown from $\mathbb{P}_{\Theta^{(j+1)}}$.

3. Learning Methodology

Assuming the observation of a single realization $\{x^{(i)}\}_{i=1}^n$ of the described time-varying model at each timestamp, our objective is twofold. We want to recover the set of change-points $\mathcal{D}$, as well as the graph structure underlying the observed data vectors, i.e. which edges are activated at each timestamp. In practice, we may observe multiple data vectors at each timestamp. However, since this does not change our analysis, we leave the related discussion for the experimental section. Next, we now describe our methodology to perform the aforementioned tasks.

Neighborhood selection strategy. Due to the intractability of the normalizing constant $Z(\cdot)$, classical maximum likelihood approaches are difficult to apply in practice. Hence, an intuitive approach is to extend the neighborhood selection strategy introduced for the static setting in (Ravikumar et al., 2010) to our time-varying setting. Instead of maximizing the global likelihood of Eq. (1), this approach maximizes, for each node $a \in V$, the conditional likelihood of the node knowing the other nodes in $V \setminus a$. The conditional probability of observing a node's value, knowing the others, when $X \sim \mathcal{I}(\Omega)$, is:

$$\mathbb{P}_{\omega_a}(x_a|x_{\setminus a}) = \frac{\exp\left\{2x_a \sum_{b\in V\setminus a} x_b \omega_{ab}\right\}}{\exp\left\{2x_a \sum_{b\in V\setminus a} x_b \omega_{ab}\right\} + 1}, \qquad (3)$$

where ω_a denotes the a-th column (or row) of Ω that is used to parametrize the probability function of Eq. (3). Here, $x_{\setminus a}$ denotes the vector x without the coordinate a.

For each node, we thus propose to maximize a penalized version of the conditional likelihood of Eq. 3. The detailed procedure is explained below.

3.1. Optimization program

The neighborhood selection strategy works as follows. For each node $a = 1, ..., p$, we solve the regularized optimization program:

$$\widehat{\beta}_a = \underset{\beta \in \mathbb{R}^{p-1 \times n}}{\operatorname{argmin}} \ \mathcal{L}_a(\beta) + pen_{\lambda_1,\lambda_2}(\beta). \tag{4}$$

In this equation, $\mathcal{L}_a(\beta)$ stands for the node-wise negative conditional log-likelihood of node a, knowing $x_{\setminus a}^{(i)}$:

$$\mathcal{L}_a(\beta) \triangleq -\sum_{i=1}^{n} \log\left(\mathbb{P}_{\beta^{(i)}}(x_a^{(i)}|x_{\setminus a}^{(i)})\right) \tag{5}$$

$$= \sum_{i=1}^{n} \log\left\{\exp\left(\beta^{(i)\mathsf{T}} x_{\setminus a}^{(i)}\right) + \exp\left(-\beta^{(i)\mathsf{T}} x_{\setminus a}^{(i)}\right)\right\} - \sum_{i=1}^{n} x_a^{(i)} \beta^{(i)\mathsf{T}} x_{\setminus a}^{(i)}, \tag{6}$$

where $\beta^{(i)}$ is the i-th column of β. The last line is obtained by plugging Eq. (3) in Eq. (5) with $\beta^{(i)}$ instead of ω_a.

With such objective function, we learn at each timestamp i the neighborhood $\beta^{(i)}$ of node a via a penalized logistic regression method.

Penalty term. Provided two hyperparameters, $\lambda_1, \lambda_2 > 0$, we propose the following penalty term for Eq. 4:

$$pen_{\lambda_1,\lambda_2}(\beta) = \lambda_1 \sum_{i=2}^{n} \|\beta^{(i)} - \beta^{(i-1)}\|_2 + \lambda_2 \sum_{i=1}^{n} \|\beta^{(i)}\|_1.$$

The overall two-term penalty is necessary for recovering efficiently the piece-wise constant graph structure. The second term is quite standard: it allows the estimated parameter vectors to be sparse and thus imposes structure in the learned graphs. On the other hand, without the first term, we would fit for each timestamp $i \in \{1, ..., n\}$ a parameter vector $\beta^{(i)}$ that perfectly matches the observation $x^{(i)}$ (in terms of likelihood). In such a situation, we would obtain as many different parameter vectors β as there is different samples, making the piece-wise constant assumption of Eq. 2 impossible to recover. This is why we propose a group-fused penalty, consisting in the ℓ_2-norm of the difference between two consecutive parameter vectors. The sum of the ℓ_2-norms acts as a group-lasso penalty on temporal difference between consecutive parameter vectors, which encourages the two vectors to be equal. This allows us to learn efficiently an evolving piece-wise constant structure and also to detect the change-points.

In conclusion, the hyperparameter λ_1 controls the number of estimated change-points – the larger λ_1 is, the fewer the estimated number of change-points will be. Similarly, when λ_2 increases, the sparsity of each parameter vector increases as well. A priori, choosing the hyperparameters is not straightforward. However, since our objective function corresponds to a penalized logistic regression problem, we can use existing model selection criteria. We discuss further about this aspect in the experimental section.

3.2. Change-point detection and structure estimation

Assume that the optimization program (4) is solved. The set of estimated change-points $\widehat{\mathcal{D}}$ is:

$$\widehat{\mathcal{D}} = \left\{\widehat{T}_j \in \{2, ..., n\} : \|\widehat{\beta}_a^{(\widehat{T}_j)} - \widehat{\beta}_a^{(\widehat{T}_j - 1)}\|_2 \neq 0\right\}.$$

Namely, this corresponds to the set of timestamps at which the estimated parameter vectors have changed. For each submodel $j = 1, ..., |\widehat{\mathcal{D}}| + 1$, the a-th column of $\Theta^{(j)}$ is estimated by $\widehat{\theta}_a^{(j)} \triangleq \widehat{\beta}_a^{(\widehat{T}_{j-1})} = ... = \widehat{\beta}_a^{(\widehat{T}_j - 1)}$. The non-zero elements of $\widehat{\theta}_a^{(j)}$ indicate the *neighborhood* of a.

One should notice that this estimation leads to a non-symmetric weight matrix. To overcome this problem, is was proposed in (Ravikumar et al., 2010; Kolar & Xing, 2012) to either use the $\min$ or $\max$ operator. In the present work, to estimate the structure of the j-th graph, we take:

$$\widehat{E}^{(j)} = \{(a, b) : \max(|\widehat{\theta}_{ab}^{(j)}|, |\widehat{\theta}_{ba}^{(j)}|) > 0\},$$

where $\widehat{\theta}_{ab}^{(j)}$ is the b-th element of $\widehat{\theta}_a^{(j)}$, and conversely for $\widehat{\theta}_{ba}^{(j)}$. In this case, there is an edge between two nodes if at least one of them contains the other node in its neighborhood.

4. Theoretical analysis

In this section, we present two change-point consistency theorems for TVI-FL. The theorems state that, as the number of samples n tends to infinity, the change-points are estimated more and more precisely.

4.1. Technical assumptions

We denote by $\widehat{D} = |\widehat{\mathcal{D}}|$ (the set's cardinality) the total number of detected change-points, respectively for the real changes $D = |\mathcal{D}|$, and by $[D]$ the set of indices $\{1, ..., D\}$. Let us now define two important quantities. The first is the minimal time difference between two change-points:

$$\Delta_{\min} \triangleq \min_{j \in [D]} |T_j - T_{j-1}|.$$

The second quantity is the minimal variation in the model parameters between two change-points, which is given by:

$$\xi_{\min} \triangleq \min_{a \in V, j \in [D]} \|\theta_a^{(j+1)} - \theta_a^{(j)}\|_2.$$

We now introduce three standard assumptions on the Ising model inference and change-points detection. They are assumed to be true for each node $a \in V$.

(A1) *There exist two constants* $\phi_{\min} > 0$ *and* $\phi_{\max} < \infty$ *such that* $\forall j \in [D+1]$, $\phi_{\min} \leq \Lambda_{\min}\left(\mathbb{E}_{\Theta^{(j)}}[X_{\setminus a}X_{\setminus a}^{\mathsf{T}}]\right)$ *and* $\phi_{\max} \geq \Lambda_{\max}\left(\mathbb{E}_{\Theta^{(j)}}[X_{\setminus a}X_{\setminus a}^{\mathsf{T}}]\right)$. *Here* $\Lambda_{\min}(\cdot)$ *and* $\Lambda_{\max}(\cdot)$ *denote, respectively, the smallest and largest eigenvalues of the input matrix.*

This is a standard assumption for such problems: it ensures that the covariates are not too dependent, and makes the model identifiable (Ravikumar et al., 2010; Kolar & Xing, 2012). In fact, this assumption is always verified if the support of the model is sufficiently large. Indeed, if at least p linearly independent vectors have a non-zero probability to be observed, then the matrix $\mathbb{E}_{\Theta^{(j)}}[X_{\setminus a}X_{\setminus a}^{\mathsf{T}}]$ will have full rank.

(A2) *There exists a constant* $C > 0$ *such that* $\max_{j,l\in[D+1]}\|\theta_a^{(j)} - \theta_a^{(l)}\|_2 \leq C$, *and a constant* M *such that* $\max_{j\in[D+1]}\|\theta_a^{(j)}\|_2 \leq M$.

(A3) *The sequence* $\{T_j\}_{j=1}^D$ *satisfies, for each* j, $T_j = \lfloor n\tau_j \rfloor$ *, where* $\lfloor x \rfloor$ *is the largest integer smaller than or equal to* x *and* $\{\tau_j\}_{j=1}^D$ *is a fixed, unknown sequence of the change-point fractions belonging to* $[0, 1]$.

This last assumption says that as n grows, the new observations are sampled uniformly across all the $D+1$ sub-models.

4.2. Main results

Next, we present our theoretical results on *change-point consistency*. The proofs are made for one node, a, but generalize to all the other nodes. We first provide the optimality conditions necessary to demonstrate the main results.

Lemma 1. *(Optimality Conditions) A matrix* $\hat{\beta}$ *is optimal for problem (4) iff there exists a collection of subgradient vectors* $\{\hat{z}^{(i)}\}_{i=2}^n$ *and* $\{\hat{y}^{(i)}\}_{i=1}^n$, *with* $\hat{z}^{(i)} \in \partial\|\widehat{\beta}^{(i)} - \widehat{\beta}^{(i-1)}\|_2$ *and* $\hat{y}^{(i)} \in \partial\|\widehat{\beta}^{(i)}\|_1$, *such that* $\forall k = 1, ..., n$:

$$\sum_{i=k}^{n} x_{\setminus a}^{(i)} \left\{\tanh\left(\widehat{\beta}^{(i)\mathsf{T}} x_{\setminus a}^{(i)}\right) - \tanh\left(\omega_a^{(i)\mathsf{T}} x_{\setminus a}^{(i)}\right)\right\} - \sum_{i=k}^{n} x_{\setminus a}^{(i)} \left\{x_a^{(i)} - \mathbb{E}_{\Omega^{(i)}}\left[X_a | X_{\setminus a} = x_{\setminus a}^{(i)}\right]\right\} + \lambda_1 \hat{z}^{(k)} + \lambda_2 \sum_{i=k}^{n} \hat{y}^{(i)} = \mathbf{0}_{p-1}, \quad (7)$$

where tanh *is the hyperbolic tangent function,* $\mathbf{0}_{p-1}$ *is the zero vector of size* $p-1$, $\hat{z}^{(1)} = \mathbf{0}_{p-1}$, *and*

$$\hat{z}^{(i)} = \begin{cases} \frac{\widehat{\beta}^{(i)} - \widehat{\beta}^{(i-1)}}{\|\widehat{\beta}^{(i)} - \widehat{\beta}^{(i-1)}\|_2} & \text{if } \widehat{\beta}^{(i)} - \widehat{\beta}^{(i-1)} \neq 0, \\ \in \mathcal{B}_2(0,1) & \text{otherwise}, \end{cases}$$

$$\hat{y}^{(i)} = \begin{cases} \text{sign}(\widehat{\beta}^{(i)}) & \text{if } x \neq 0, \\ \in \mathcal{B}_1(0,1) & \text{otherwise}. \end{cases}$$

Proof. The proof is given in the Appendix. It consists in writing the sub-differential of the objective function and say, thanks to the convexity, that 0 belongs to it. □

Theorem 1. *(Change-point consistency) Let* $\{x_i\}_{i=1}^n$ *be a sequence of observations drawn from the model presented in Sec. 2. Suppose (A1-A3) hold, and assume that* $\lambda_1 \asymp \lambda_2 = \mathcal{O}(\sqrt{\log(n)/n})$. *Let* $\{\delta_n\}_{n\geq 1}$ *be a non-increasing sequence that converges to* 0, *and such that* $\forall n > 0$, $\Delta_{\min} \geq n\delta_n$, *with* $n\delta_n \to +\infty$. *Assume further that* (i) $\frac{\lambda_1}{n\delta_n\xi_{\min}} \to 0$, (ii) $\frac{\sqrt{p-1}\lambda_2}{\xi_{\min}} \to 0$, *and* (iii) $\frac{\sqrt{p\log(n)}}{\xi_{\min}\sqrt{n\delta_n}} \to 0$. *Then, if the correct number of change-points are estimated, we have* $\widehat{D} = D$ *and:*

$$\mathbb{P}(\max_{j=1,...,D} |\hat{T}_j - T_j| \leq n\delta_n) \underset{n\to\infty}{\longrightarrow} 1. \quad (8)$$

Proof. We extend the proof given in (Harchaoui & Lévy-Leduc, 2010; Kolar & Xing, 2012) to the particular case of the Ising model. While some important steps are essentially similar, the main difference between our proof and the previous one are Lemmas 2, 3, 4, and 5 included in the Appendix, which provide concentration bounds adapted to the Ising model setting. Moreover, we had to employ several new tricks that were not used in the previous papers. We give here a sketch of the proof.

Thanks to the union bound, the probability of the complementary in Eq. (8) can be upper bounded by:

$$\mathbb{P}(\max_{j=1,...,D} |\hat{T}_j - T_j| > n\delta_n) \leq \sum_{j=1}^{D} \mathbb{P}(|\hat{T}_j - T_j| > n\delta_n).$$

To prove Eq. (8), it is now sufficient to show $\forall j = 1, ..., D$ that $\mathbb{P}(|\hat{T}_j - T_j| > n\delta_n) \to 0$. Let us define the event $C_n = \{|\hat{T}_j - T_j| < \frac{\Delta_{\min}}{2}\}$ and its complementary C_n^c. The rest of the proof is divided in two parts: bounding the good scenario, i.e. show that $\mathbb{P}(\{|\hat{T}_j - T_j| > n\delta_n\} \cap C_n) \to 0$, and doing the same for the bad scenario, i.e $\mathbb{P}(\{|\hat{T}_j - T_j| > n\delta_n\} \cap C_n^c) \to 0$.

To bound the good scenario, the proof applies Lemma 1 to bound the considered probability by three others probabilities. These latter are then asymptotically bounded by 0, thanks to a combination of Assumptions (A1-A3), assumptions of the theorem and concentration inequalities related to the considered time-varying Ising model (given by the lemmas of the Appendix).

To bound the bad case scenario, the three following comple-

mentary events are defined:

$$D_n^{(l)} \triangleq \left\{\exists j \in [D], \widehat{T}_j \leq T_{j-1}\right\} \cap C_n^c,$$
$$D_n^{(m)} \triangleq \left\{\forall j \in [D], T_{j-1} < \widehat{T}_j < T_{j+1}\right\} \cap C_n^c,$$
$$D_n^{(r)} \triangleq \left\{\exists j \in [D], \widehat{T}_j \geq T_{j+1}\right\} \cap C_n^c.$$

Thus, it suffices to prove that $\mathbb{P}(\{|\hat{T}_j - T_j| > n\delta_n\} \cap D_n^{(l)})$, $\mathbb{P}(\{|\hat{T}_j - T_j| > n\delta_n\} \cap D_n^{(m)})$, and $\mathbb{P}(\{|\hat{T}_j - T_j| > n\delta_n\} \cap D_n^{(r)}) \to 0$ as $n \to \infty$. To prove this, similar arguments to those used for the good case are employed. □

Note that with $\delta_n = \log(n)^\gamma / n$, for any $\gamma > 1$ and $\xi_{\min} = \Omega(\sqrt{\log(n)/\log(n)^\gamma})$, the conditions of the theorem are met. With this parameterization, we obtain a convergence rate of order $\mathcal{O}(\log(n)^\gamma / n)$ for the estimation of the change-points. More precisely, for any $\delta > 0$ and sufficiently large n, we have with probability at least $1 - \delta$ that

$$\frac{1}{n} \max_{j=1,\dots,D} |\hat{T}_j - T_j| \leq \frac{1}{n} \log(n)^\gamma.$$

In conclusion, we obtain the same rate of convergence to that of the single change-point detection method given in (Roy et al., 2017). It is almost optimal up to a logarithmic factor. The main drawback of the previous theorem is that it assumes that the number of change-points have been correctly estimated. In practice this is complicated to verify, while proving that the right number of change-points are consistently estimated is also difficult to get for this type of methods (Harchaoui & Lévy-Leduc, 2010). Nevertheless, in practice we may have an idea about an upper bound on the true number of change-points.

The next proposition provides a consistency result when the number of change-points is overestimated. Let us first introduce the metric $d(A\|B)$ defined as:

$$d(A\|B) = \sup_{b \in B} \inf_{a \in A} |b - a|. \tag{9}$$

Proposition 1. *Let $\{x_i\}_{i=1}^n$ be a sequence of observations drawn from the model presented in Sec. 2. Assume the conditions of Theorem 1 are respected. Then, if for a fix $D_{\max} < \infty$, we have $D \leq \widehat{D} \leq D_{\max}$ then:*

$$\mathbb{P}(d(\widehat{\mathcal{D}}\|\mathcal{D}) \leq n\delta_n) \underset{n\to\infty}{\longrightarrow} 1.$$

Proof. A detailed proof is provided in Appendix. The proof applies multiple times the different tricks used to prove Theorem 1 and the Lemmas also given in the Appendix. □

Proposition 1 is of fundamental importance as it tells us that, even though the number of change-points has been overestimated, asymptotically, all the true change-points belong to the set of estimated change-points.

5. Experimental study

This section provides numerical arguments showing the empirical performance of TVI-FL. All the experiments were implemented using Python and conducted on a personal laptop. The code of TVI-FL is available online[1] and provided in the supplementary material, so as a Jupyter Notebook reproducing results and figures of the real-world example.

5.1. Optimization procedure

Despite being non-differentiable, the convexity of the objective function allows the use of existing convex optimization algorithms of the literature. In this work, we use the python package CVXPY (Diamond & Boyd, 2016) that allows us to solve our problem efficiently. Note also that the optimization for each node is independent to the other nodes, and hence the approach allows efficient parallel implementations.

In the situation where more than one data vector is observed at each timestamp, one has simply to replace the node-wise negative log-likelihood in Eq. 5 with:

$$-\sum_{i=1}^{n}\sum_{l=1}^{n^{(i)}} \log\left(\mathbb{P}_{\beta^{(i)}}(x_a^{(il)}|x_{\setminus a}^{(il)})\right), \tag{10}$$

where $n^{(i)}$ stands for the number of data vectors observed at timestamp i, and $x^{(il)}$ for the l-th observed vector at time i.

Tuning the hyperparameters λ_1 and λ_2. As stated in Sec. 3.1, it is possible to employ any model selection technique suited for logistic regression. In the experiments, we use and compare two techniques. The first is the *Akaike Information Criterion* (AIC) that computes the average of the following quantity for all nodes:

$$\text{AIC}(\widehat{\beta}_a) \triangleq 2\mathcal{L}_a(\widehat{\beta}_a) + 2\,\text{Dim}(\widehat{\beta}_a). \tag{11}$$

$$\text{Dim}(\widehat{\beta}_a) = \sum_{i=1}^{n}\left(\mathbb{1}\{\widehat{\beta}_a^{(i)} \neq \widehat{\beta}_a^{(i-1)}\} \sum_{b \in V\setminus a} \mathbb{1}\{\text{sign}(\widehat{\beta}_{ab}^{(i)}) \neq 0\}\right)$$

counts the number of parameters that are estimated. By convention, $\widehat{\beta}_a^{(0)} = \widehat{\beta}_a^{(1)}$. In this case, set of hyperparameters that minimize the AIC are finally selected.

The second technique, based on cross-validation (CV), assumes that more than one sample is observed at each moment in time $i = 1, ..., n$. Thus, the time-series can be split in a part for the learning phase and another part for the testing phase. In our experiments, we selected the hyperparameters maximizing the AUC i.e. the area under the ROC-curve associated to the classification score (the probability to be equal to either 1 or -1).

[1] https://github.com/BatisteLB/TVI-FL

For both model selection techniques, AIC and CV, the hyperparameters are found using either the standard random-search or grid-search strategies.

5.2. Experimental setup

Baseline method. As mentioned in Sec. 1, no existing work in the literature deals properly with the considered multiple change-points detection task. Several methods deal with varying Gaussian graphical models (Kolar & Xing, 2012; Yang & Peng, 2019), varying Ising models with smooth structural changes over time (Kolar et al., 2010), or the detection of a single change-point in the varying Ising model (Roy et al., 2017). The closest work we can compare with is the Tesla method (Ahmed & Xing, 2009; Kolar et al., 2010). Its major difference to our approach is the use of the ℓ_1-norm instead of the ℓ_2-norm as a fused-penalty. This difference is very significant, theoretically and practically.

Indeed, *using an ℓ_1-norm fused-penalty does not encourage the recovery of a graphical model that evolves piece-wise constantly as a whole*, which makes it less adaptable to recover change-points. More specifically, such a term does not encourage two consecutive parameter vectors to be equal at every dimensions: the regularization only affects each dimension independently. Thus, despite the edge weights may evolve *independently* in a piece-wise constant fashion, those changes occur at arbitrary timestamps and does not aggregate to a globally piece-wise constant behavior. An illustration of this phenomena and a comparison with the ℓ_2-norm can be found in the Appendix. Nonetheless, the same way the standard linear regression can be used to recover sparse parameters, Tesla can still be used to recover change-points in practice. Hence, we choose this method as our baseline because, despite the lack of any theoretical guarantee, it can still be applicable, provided a sufficiently large sample size and appropriately tuned regularization.

Performance metrics. We use two suitable metrics to evaluate the quality of TVI-FL on the learned graphs and change-points. The first one, very standard in change-point detection tasks (Truong et al., 2019) and known as the *Hausdorff metric*, measures the longest temporal distance between a change-point in $\mathcal{D}$ and its prediction in $\widehat{\mathcal{D}}$:

$$h(\mathcal{D}, \widehat{\mathcal{D}}) \triangleq \frac{1}{n} \max \left\{ \max_{t \in \mathcal{D}} \min_{\hat{t} \in \widehat{\mathcal{D}}} |t - \hat{t}|,\ \max_{\hat{t} \in \widehat{\mathcal{D}}} \min_{t \in \mathcal{D}} |t - \hat{t}| \right\}.$$

The lower this metric is, the better is the estimation. The second one, the F_1*-score*, measures the goodness of the learned graphs structures (high value is better) by the quantity:

$$F_1 = \frac{2 \times precision \times recall}{precision + recall},$$

which combines the two following classic measures:

$$precision = \frac{1}{n} \sum_{i=1}^{n} \sum_{a<b} \frac{\mathbb{1}\{(a,b) \in \hat{E}_i \wedge (a,b) \in E_i\}}{\mathbb{1}\{(a,b) \in \hat{E}_i\}},$$

$$recall = \frac{1}{n} \sum_{i=1}^{n} \sum_{a<b} \frac{\mathbb{1}\{(a,b) \in \hat{E}_i \wedge (a,b) \in E_i\}}{\mathbb{1}\{(a,b) \in E_i\}}.$$

5.3. Application to synthetic data

Simulation design. We compare the performance of our method TVI-FL against Tesla using several independent synthetic datasets. We first fix certain characteristics for all generated datasets: each of them has $n = 100$ timestamps, $|D| = 2$ change-points at the 51-st and 81-st timestamps, hence resulting in 3 submodels being valid respectively for 50, 30, and 20 timestamps. We consider the graph structure of each submodel to be an independent random d-regular graph of $p = 20$ nodes, where at each time the degree of the all nodes can be $d \in \{2, 3, 4\}$.

To generate a piece-wise constant Ising model:

- We first pick a degree value $d \in \{2, 3, 4\}$ and draw independently 3 random d-regular graphs, one for each submodel. Same as in (Ahmed & Xing, 2009), the edge weights are drawn from a uniform distribution taken over $[-1, -0.5] \cup [0.5, 1]$.
- For each submodel, we draw observations using Gibbs sampling with a burn-in period of 1000 samples. Moreover, we collect one observation every 20 samples (lag) to avoid dependencies between them. In fact, instead of a single observation, for each timestamp $i \in \{1, ..., n\}$ we generate multiple observations $n^{(i)}$ in $\{4, 6, 8\}$, which requires to use the likelihood of Eq. (10). Besides, to be able to perform CV, we also sample 5 more observations per timestamp and use them only in the testing phase.

With the above procedure we generate 10 different piece-wise constant Ising models for each degree d, which makes 30 models to learn in total. In addition, for each model, we generate 3 different sets of observations, one for each $n^{(i)} \in \{4, 6, 8\}$, that constitute the individual learning problems of our evaluation. This results in 90 experiments in total.

For each experiment, we use a random-search strategy to find the best pair of hyperparameters (λ_1, λ_2) in $[4, 15] \times [30, 40]$. This is done individually for the TVI-FL and Tesla methods. The selected hyperparameters are those minimizing the AIC or maximizing the AUC (see Sec. 5.1).

Results. The average value and standard deviation of the corresponding h-score and F_1-score over each group of 10 experiments are reported in Tab. 1. The results clearly show that TVI-FL outperforms Tesla, regardless which model selection criterion we consult. This was expected as Telsa is not designed to recover Ising models that are evolving

Degree	Observations per timestamps	Method	AIC h-score ↓	AIC F_1-score ↑	AIC $\widehat{D}$	AUC h-score ↓	AUC F_1-score ↑	AUC $\widehat{D}$
$d=2$	$n^{(i)}=4$	TVI-FL	**0.046 ± (0.024)**	**0.694 ± (0.103)**	7.400 ± (3.137)	0.221 ± (0.035)	**0.876 ± (0.030)**	26.100 ± (7.739)
		Tesla	0.106 ± (0.087)	0.649 ± (0.190)	12.700 ± (7.682)	**0.184 ± (0.051)**	0.841 ± (0.041)	25.100 ± (4.784)
	$n^{(i)}=6$	TVI-FL	**0.129 ± (0.058)**	**0.816 ± (0.073)**	9.700 ± (2.759)	**0.147 ± (0.071)**	**0.875 ± (0.027)**	15.300 ± (3.378)
		Tesla	0.178 ± (0.130)	0.748 ± (0.167)	12.900 ± (5.540)	0.164 ± (0.062)	0.841 ± (0.048)	19.000 ± (2.530)
	$n^{(i)}=8$	TVI-FL	**0.082 ± (0.081)**	0.833 ± (0.095)	7.400 ± (3.040)	**0.099 ± (0.073)**	**0.891 ± (0.024)**	11.000 ± (3.873)
		Tesla	0.124 ± (0.071)	**0.846 ± (0.047)**	13.600 ± (2.010)	0.178 ± (0.066)	0.853 ± (0.039)	14.700 ± (3.348)
$d=3$	$n^{(i)}=4$	TVI-FL	**0.080 ± (0.069)**	**0.563 ± (0.089)**	7.000 ± (2.683)	**0.204 ± (0.035)**	**0.734 ± (0.024)**	23.100 ± (6.715)
		Tesla	0.278 ± (0.319)	0.353 ± (0.072)	3.200 ± (2.891)	0.208 ± (0.029)	0.611 ± (0.041)	29.200 ± (3.187)
	$n^{(i)}=6$	TVI-FL	**0.055 ± (0.064)**	**0.617 ± (0.161)**	6.300 ± (3.494)	**0.130 ± (0.051)**	**0.743 ± (0.034)**	12.800 ± (2.821)
		Tesla	0.302 ± (0.241)	0.346 ± (0.060)	2.000 ± (1.183)	0.173 ± (0.044)	0.616 ± (0.041)	22.600 ± (2.245)
	$n^{(i)}=8$	TVI-FL	**0.091 ± (0.073)**	**0.714 ± (0.130)**	8.000 ± (2.530)	**0.127 ± (0.073)**	**0.764 ± (0.032)**	10.400 ± (2.154)
		Tesla	0.311 ± (0.231)	0.361 ± (0.098)	2.600 ± (2.615)	0.162 ± (0.052)	0.633 ± (0.045)	18.700 ± (3.716)
$d=4$	$n^{(i)}=4$	TVI-FL	**0.101 ± (0.082)**	**0.453 ± (0.111)**	6.500 ± (3.324)	**0.232 ± (0.026)**	**0.644 ± (0.041)**	29.400 ± (4.317)
		Tesla	0.444 ± (0.273)	0.347 ± (0.044)	2.875 ± (1.900)	0.234 ± (0.017)	0.518 ± (0.046)	34.625 ± (1.654)
	$n^{(i)}=6$	TVI-FL	**0.099 ± (0.064)**	**0.501 ± (0.130)**	5.667 ± (2.309)	**0.183 ± (0.044)**	**0.664 ± (0.041)**	16.778 ± (3.258)
		Tesla	0.258 ± (0.236)	0.355 ± (0.035)	2.500 ± (1.118)	0.215 ± (0.032)	0.503 ± (0.040)	26.000 ± (4.472)
	$n^{(i)}=8$	TVI-FL	**0.077 ± (0.076)**	**0.528 ± (0.158)**	5.556 ± (3.624)	**0.169 ± (0.064)**	**0.678 ± (0.049)**	12.444 ± (4.524)
		Tesla	0.251 ± (0.230)	0.357 ± (0.044)	2.625 ± (0.696)	0.219 ± (0.027)	0.518 ± (0.054)	24.000 ± (2.398)

Table 1: Results for the model with the lowest AIC, and that with the highest AUC. The average ± (std) of the metrics is reported. The best results between TVI-FL and Tesla, for each metric, are specified in bold.

piece-wise constantly (see Sec. 5.2). Furthermore, while in some cases Tesla finds a number of change-points closer to the true number, the associated h-scores are still higher than those of TVI-FL. Yet, Tesla is still not irrelevant to the task and in fact there are cases in which it reaches competitive performance scores to those of TVI-FL. Another finding is that AIC seems to favor a low number of estimated changes-points. It achieves better h-scores for this simulated process, while the AUC criterion seems to give priority to the recovery of the graph structure, illustrated by higher F_1-scores.

We show that empirically it is possible to obtain both low h-score and high F_1-score via better hyperparameters tuning. Specifically, for each experiment and for each degree d, we select the model with the highest F_1-score when the associated h-score $\leq h_{\text{min}}$, with $h_{\text{min}} \in \{0, .01, .02, .03\}$. This allows, respectively at most 0 to 3 timestamps of offset error between an estimated and a real change-point. In the results of Fig. 1 we observe that even with very low h-score, high F_1-score are reachable. Furthermore, the TVI-FL method always provides better F_1-score than Tesla, confirming once again its superior performance.

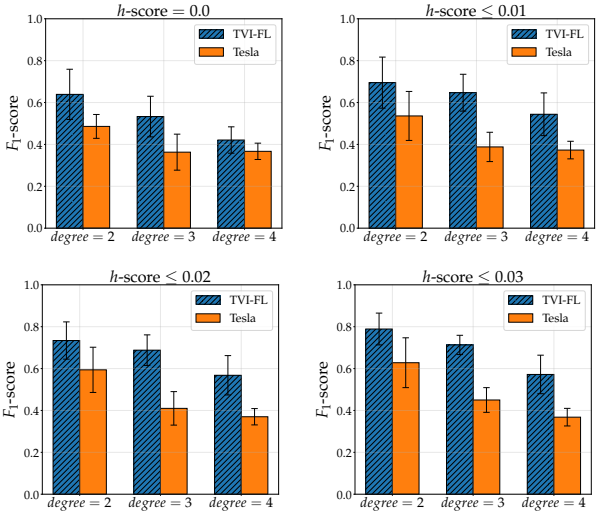

Figure 1: The average value of the best F_1-score obtained when the h-score is below a certain threshold. These thresholds are respectively (from top left to bottom right), 0.0, 0.01, 0.02, 0.03, i.e. at most 0, 1, 2, or 3 timestamps of offset error between an estimated and a real change-point. Each pairs of bars corresponds to different d-regular graphs, with $d \in \{2, 3, 4\}$. The error bars correspond to ± (std).

5.4. Finding change-points in the real world

Dataset and setup. In this section we evaluate the empirical performance of the TVI-FL method in a real-world use case. In particular, we analyze the different votes of the Illinois House of Representatives during the period of the 114-th and 115-th US Congresses (2015-2019), which are available at voteview.com (Lewis et al., 2020). The Illinois House of Representatives has 18 seats (one per district), each one corresponds to a US Representative belonging to the Democratic or the Republican parties. A Representative may or may not get reelected at the end of a Congress, which affects if he/she will retain his/her seat in the new Congress. The specific dataset we used contains 1264 votes, each of them represented by a vector of size $p = 18$, where a dimension is equal to 1 if the respective Representative of that seat has voted *Yes*, and -1 if it has voted *No*. When no

information is provided about the vote of a seat (e.g. due to an absence), we impute the majority vote of its party.

It is always difficult to interpret a large number of change-points. For this reason, we choose to use the AIC criterion, which was found in Sec. 5 to favor smaller number of change-points. As for model tuning, we use a grid-search strategy to find the best values for the hyperparameters.

Results. Fig. 2 (bottom) shows the cumulative function of the votes of each of the 18 seats, in temporal order, and the three change-points (dashed vertical line) detected by TVI-FL. The first two change-points are difficult to interpret; it seems though that the second one corresponds to the pre-election period when a Congress comes to its end and votes get usually less polarized. Nevertheless, it must be noted that the structural changes of the first two change-points are significantly lower compared to the third one. In fact, this last estimated change-point corresponds exactly to the time at which the Congress has changed. This significant change-point seems due to the non-reelection of some Representatives. More specifically, the Representative of 10-th seat was the only one who was not reelected at the end of the 114-th Congress: the Republican Robert Dold, who was replaced by the Democrat Brad Schneider. This switch apparently lead to a significant variation in the structure of the underlying graph. Fig. 2 (top) shows the graphs of positive weights, before and after this significant change-point. As expected, two clusters appear, one with the seats of Democrats and the other with those of the Republicans. Moreover, the 10-th seat becomes more connected with the cluster of Democrats after the time of change: the node loses 3 connections to the Republican cluster and gains 5 connections to Democrats and gets connected with all of them. More generally, all its weights with the Republican cluster decreases, contrarily to its weights with the Democratic cluster that do increase. This observation explains the origins of the structural change. Finally, it is interesting to observe that before and after the change-point, the 10-th seat is the only one well-connected to both political groups. This makes us to conclude that this seat is represented by a *super-collaborator*, a role that some Representatives get by acting more independently and position themselves in the middle of the parties (Andris et al., 2015). Similarly, it is not surprising for Dan Lipinski, who had the 3-rd seat, to present in the learned graphs 2 connections with Republicans, as he is known to be a conservative Democrat.

Overall, this experiment shows that TVI-FL is suited to find change-points in a real-world binary dataset, while also to recover the underlying evolving graph structure. This way, it increases the interpretability of the detected change-points. After applying the Telsa method on the same problem, we observed similar results and for this reason we omit them from the presentation.

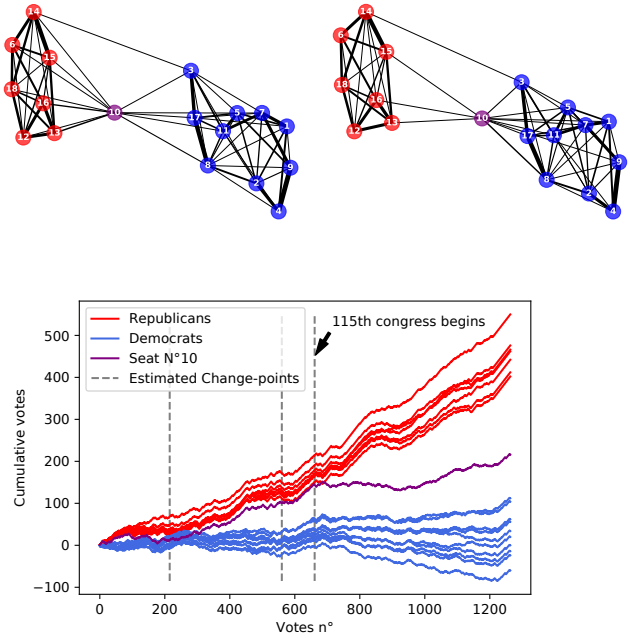

Figure 2: (Top) The two graphs before and after the strongest estimated change-point: the third one that indeed corresponds to the end of the 114-th Congress. (Bottom) The cumulative functions of votes of the 18 seats over the two Congresses.

6. Conclusion and Future work

This paper proposed TVI-FL, an efficient way to learn a time-varying Ising model with piece-wise constantly evolving structure. Our method is able to both detect the change-points at which the structure of the model changes and the structure themselves. Our work is the first to provide change-point consistency theorems in this context. Those theoretical guarantees are reinforced by an empirical study. Using two different model selection criteria, the proposed method is showed to outperform the closest baseline algorithm. Directions of future research may include the investigation of a more adapted and refined way to solve the optimization program, the proof of consistent graph structure recovery (*sparsistency*) or the use of the recent Interaction Screening Objective (Vuffray et al., 2016) in place of the standard conditional likelihood.

Acknowledgments

This work was funded by the IdAML Chair hosted at ENS Paris-Saclay, Université Paris-Saclay.

References

Ahmed, A. and Xing, E. P. Recovering time-varying networks of dependencies in social and biological studies. *Proc. of the National Academy of Sciences*, 106(29):11878–11883, 2009.

Andris, C., Lee, D., Hamilton, M. J., Martino, M., Gunning, C. E., and Selden, J. A. The rise of partisanship and super-

cooperators in the US House of Representatives. *PLOS ONE*, 10(4):e0123507, 2015.

Banerjee, O., Ghaoui, L. E., and d'Aspremont, A. Model selection through sparse maximum likelihood estimation for multivariate Gaussian or binary data. *Journal of Machine Learning Research*, 9(Mar):485–516, 2008.

Bleakley, K. and Vert, J.-P. The group fused lasso for multiple change-point detection. *arXiv preprint arXiv:1106.4199*, 2011.

Bybee, L. and Atchadé, Y. Change-point computation for large graphical models: a scalable algorithm for Gaussian graphical models with change-points. *Journal of Machine Learning Research*, 19(1):440–477, 2018.

Choi, M. J., Lim, J. J., Torralba, A., and Willsky, A. S. Exploiting hierarchical context on a large database of object categories. In *IEEE Computer Society Conf. on Computer Vision and Pattern Recognition*, pp. 129–136, 2010.

Diamond, S. and Boyd, S. CVXPY: A Python-embedded modeling language for convex optimization. *Journal of Machine Learning Research*, 17(83):1–5, 2016.

Dong, X., Thanou, D., Frossard, P., and Vandergheynst, P. Learning Laplacian matrix in smooth graph signal representations. *Trans. Signal Processing*, 64(23):6160–6173, 2016.

Du, N., Song, L., Yuan, M., and Smola, A. Learning networks of heterogeneous influence. In *Advances in Neural Information Processing Systems*, pp. 2780–2788, 2012.

Fazayeli, F. and Banerjee, A. Generalized direct change estimation in ising model structure. In *Int. Conf. on Machine Learning*, pp. 2281–2290, 2016.

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

Gibberd, A. J. and Nelson, J. D. Regularized estimation of piecewise constant Gaussian graphical models: The group-fused graphical lasso. *Journal of Computational and Graphical Statistics*, 26(3):623–634, 2017.

Gibberd, A. J. and Roy, S. Multiple changepoint estimation in high-dimensional Gaussian graphical models. *arXiv preprint arXiv:1712.05786*, 2017.

Goel, S., Kane, D. M., and Klivans, A. R. Learning ising models with independent failures. In *Conf. on Learning Theory*, pp. 1449–1469, 2019.

Hallac, D., Park, Y., Boyd, S., and Leskovec, J. Network inference via the time-varying graphical lasso. In *Proc. of the ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, pp. 205–213. ACM, 2017.

Harchaoui, Z. and Lévy-Leduc, C. Multiple change-point estimation with a total variation penalty. *Journal of the American Statistical Association*, 105(492):1480–1493, 2010.

Höfling, H. and Tibshirani, R. Estimation of sparse binary pairwise Markov networks using pseudo-likelihoods. *Journal of Machine Learning Research*, 10(Apr):883–906, 2009.

Keshavarz, H., Michailidis, G., and Atchade, Y. Sequential change-point detection in high-dimensional Gaussian graphical models. *arXiv preprint arXiv:1806.07870*, 2018.

Kolar, M. and Xing, E. P. Estimating networks with jumps. *Electronic Journal of Statistics*, 6:2069, 2012.

Kolar, M., Song, L., Ahmed, A., Xing, E. P., et al. Estimating time-varying networks. *The Annals of Applied Statistics*, 4(1): 94–123, 2010.

Koller, D., Friedman, N., and Bach, F. *Probabilistic graphical models: principles and techniques*. MIT press, 2009.

Le Bars, B., Humbert, P., Oudre, L., and Kalogeratos, A. Learning Laplacian matrix from bandlimited graph signals. In *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, pp. 2937–2941, 2019.

Lewis, J. B., Poole, K., Rosenthal, H., Boche, A., Rudkin, A., and Sonnet, L. Voteview: Congressional roll-call votes database. *https://voteview.com/*, 2020.

Londschien, M., Kovács, S., and Bühlmann, P. Change point detection for graphical models in presence of missing values. *arXiv preprint arXiv:1907.05409*, 2019.

Marbach, D., Costello, J. C., Küffner, R., Vega, N. M., Prill, R. J., Camacho, D. M., Allison, K. R., Aderhold, A., Bonneau, R., Chen, Y., et al. Wisdom of crowds for robust gene network inference. *Nature methods*, 9(8):796, 2012.

Meinshausen, N., Bühlmann, P., et al. High-dimensional graphs and variable selection with the lasso. *The Annals of Statistics*, 34(3):1436–1462, 2006.

Ravikumar, P., Wainwright, M. J., Lafferty, J. D., et al. High-dimensional ising model selection using $\ell 1$-regularized logistic regression. *The Annals of Statistics*, 38(3):1287–1319, 2010.

Ren, Z., Sun, T., Zhang, C.-H., Zhou, H. H., et al. Asymptotic normality and optimalities in estimation of large Gaussian graphical models. *The Annals of Statistics*, 43(3):991–1026, 2015.

Rodriguez, M., Balduzzi, D., and Schölkopf, B. Uncovering the temporal dynamics of diffusion networks. In *Proc. of the Int. Conf. on Machine Learning*, pp. 561–568, 2011.

Roy, S., Atchadé, Y., and Michailidis, G. Change point estimation in high dimensional Markov random-field models. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 79(4):1187–1206, 2017.

Truong, C., Oudre, L., and Vayatis, N. Selective review of offline change point detection methods. *Signal Processing*, pp. 107299, 2019.

Vuffray, M., Misra, S., Lokhov, A., and Chertkov, M. Interaction screening: Efficient and sample-optimal learning of ising models. In *Advances in Neural Information Processing Systems*, pp. 2595–2603, 2016.

Wang, B., Qi, Y., et al. Fast and scalable learning of sparse changes in high-dimensional Gaussian graphical model structure. In *Int. Conf. on Artificial Intelligence and Statistics*, pp. 1691–1700, 2018.

Xue, L., Zou, H., Cai, T., et al. Nonconcave penalized composite conditional likelihood estimation of sparse ising models. *The Annals of Statistics*, 40(3):1403–1429, 2012.

Yang, J. and Peng, J. Estimating time-varying graphical models. *Journal of Computational and Graphical Statistics*, pp. 1–12, 2019.

Yuan, M. and Lin, Y. Model selection and estimation in the Gaussian graphical model. *Biometrika*, 94(1):19–35, 2007.