NUS RMI Working Paper Series – No. 2020-01

TriSNAR: A Three-Layer Sparse Estimator for Large-Scale Network AutoRegressive Models

Simon TRIMBORN, Ying CHEN and Ray-Bing CHEN

11 April 2020

NUS Risk Management Institute
21 HENG MUI KENG TERRACE, #04-03 I3 BUILDING, SINGAPORE 119613
www.rmi.nus.edu.sg/research/rmi-working-paper-series

TriSNAR: A Three-Layer Sparse Estimator for Large-Scale Network AutoRegressive Models

Simon Trimborn *[1], Ying Chen[1,2,3], and Ray-Bing Chen[4]

[1]Department of Mathematics, National University of Singapore
[2]Risk Management Institute, National University of Singapore
[3]Department of Economics, National University of Singapore
[4]Department of Statistics, National Cheng Kung University, Taiwan

April 11, 2020

Abstract

Understanding multi-market interactions and identifying leading markets in the global financial network is of interest to investors, regulators and policymakers. To discover the essential dynamic dependencies of digital currency exchanges, we propose TriSNAR, a three-layer sparse estimator for large-scale network autoregressive models, which imposes a structure on the lag-, network/group- and individual-level effects. We determine the asymptotic properties of the sparse estimator and investigate its finite-sample performance in extensive simulations. Numerical analysis shows that TriSNAR obtains a higher accuracy with less computational time per model contestant. We explore the applicability of TriSNAR on a network of 26 cryptocurrency exchanges with hourly pricing information. TriSNAR not only provides good out-of-sample prediction accuracy, but also exactly detects each leading exchange in North America, Europe and Asia.

Keywords: High-Dimensions, Dimension Reduction, Structure Detection, Network Analysis, Bitcoin Exchanges

JEL classification: C01, C52, C53, C55, C58

*Corresponding author, phone: +65 6516-4163, E-Mail: simon.trimborn@nus.edu.sg

1 Introduction

Advances in technology and globalization have started to reshape the trading and investment ecosystem. Among other things, this has resulted in an increased cross-listing of financial products in multiple exchanges, domestic and international, for raising funds and liquidity. This market fragmentation, along with the modern developments in the infrastructures of the exchanges, has been accompanied by billions of orders and executions. When considering each exchange as an individual in the market network, it is subject to local price formation with high time resolution to receive price information from its competing markets. Some exchanges, being recognized as signalers of dynamic patterns, have become more influential and thus lead future price movements. It has become essential for a broad community of investors, regulators and policymakers to assess and understand the dynamic effects of multi-market interactions and information flows in global large-scale financial networks.

The dynamic interactions of a variety of international networks have been investigated in the economic and econometric literature, such as economic exposure networks (Pesaran et al., 2004), the EU network (Dees et al., 2007), risk networks spanned by financial firms (Diebold and Yılmaz, 2014; Härdle et al., 2016), exchange rates and credit risk ratings (Creal et al., 2013), company networks (Rapach et al., 2013), social networks (Kline and Tamer, 2017; Zhu et al., 2017) and the Euro-zone bank network (Bonaldi et al., 2015). Gagnon and Karolyi (2010) studied the trading behavior in multi-markets and the resulting arbitrage opportunities. Gagnon and Karolyi (2013) found that cross-listing in multi-markets matters for price discovery, whereas Chen et al. (2013) investigated the particular price discovery in two markets. Halling et al. (2013) studied cross-listed firms for the similarities of markets that exhibit higher trading volume. Although there are different underlying kinds, the presence of dominating entities and mutual interactions has led to investigations of the lead-lag time effect in the spanning of these networks. This has motivated the adoption of Vector AutoRegressive (VAR) models for describing the network dynamics among multiple dependent series and facilitating their interpretation, providing valuable information for inference and helping the accuracy of the predictions (Tsay, 2016).

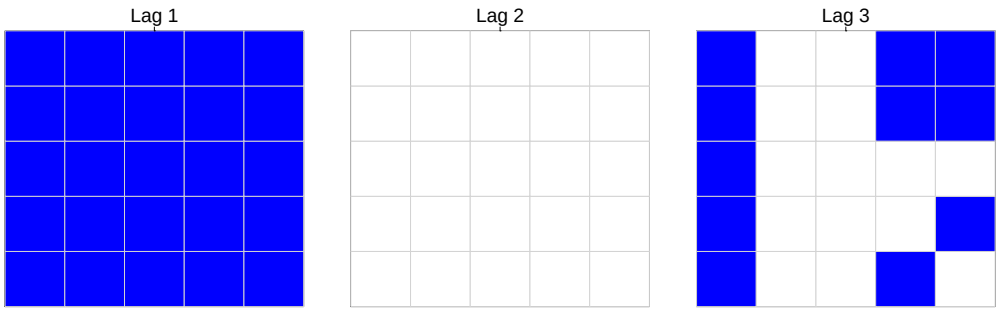
However, for high dimensions or when several lags are involved, VAR models quickly run into overparameterization and have difficulties in their convergence. Even for a moderate number of dimensions, the model structure is often overparametrized. This further impairs their understanding and interpretation, leading to possible in-

accurate estimation. A meaningful (interpretable) estimation and inference of large-scale VAR models is often impossible without imposing some structural assumptions, Basu and Michailidis (2015). Additionally, empirical evidence and domain knowledge support the existence of a sparse structure in economic and social networks (though not for all), (de Paula, 2017). Figure 1 depicts example parameter matrices for a 5-dimensional network described in a VAR(3) framework. It illustrates a dependence structure with three kinds of sparsity in

- lags, accounting for the time-dependent effect of the entire multi-market network on all exchanges,
- columns within an active NAR parameter matrix, reflecting the group effect of one exchange on all others, and
- elements within an active group for the individual effect, namely, the interactions between any two exchanges.

The Figure shows that there is a serial dependence between the system and lags 1 and 3, while lag 2 has no effect on the entire system. Among the two active matrices, lag 1 is dense with all parameters active, and lag 3 contains partially active groups in the 1st, 4th and 5th columns, indicating the only active lagged groups that have effects. The individual sparsity in the last two columns of lag 3, with some null elements, implies that the two active lagged processes have only a partial influence. The location and type of sparsity are usually unknown. This promotes the adoption of regularized estimation for the network models.

Figure 1: NAR specification with $p = 3$ lags and $d = 5$ processes. The blue squares represent active parameters with non-zero magnitude, white squares are those with null magnitude.



A plethora of estimators has been developed to uncover sparse structures. Tibshirani (1996) proposed the Least Absolute Shrinkage and Selection Operator (LASSO),

using a soft-thresholding operator for sparse estimation. Other approaches suggest hard-thresholding functions; see Antoniadis (1997). While a soft-thresholding operator is biased, a hard-thresholding operator is not continuous. Overcoming these drawbacks, a number of unbiased and continuous regularized estimators have been proposed, e.g., Smoothly Clipped Absolute Deviation (SCAD) (Fan and Li, 2001) and an adaptive LASSO (Zou, 2006). All these estimators have been derived for regression models. For multivariate dependent data, Song and Bickel (2011) proposed a two-stage VAR approach, which separately penalizes the lagged parameter matrices, columns and individual parameters. Davis et al. (2016) developed a two-step approach for Network AutoRegressive (NAR) models to investigate the dynamic interconnection in a large-scale network under sparsity. Both approaches translate to a hard-thresholding operator. Nicholson et al. (2017) considered a variety of structured VAR models and constructed soft-thresholding estimators with LASSO and the sparse-group penalty functions (Simon et al., 2013). Basu et al. (2019) proposed a lag-one VAR model for reduced rank and parameter matrices with an underlying structure, e.g., a network structure. Lin and Michailidis (2017) investigated block VAR models, whereas Skripnikov and Michailidis (2019) combined a group LASSO and LASSO to jointly estimate several VAR models. Bayesian VAR models have been widely studied; see, e.g., Ghosh et al. (2019). A VAR model under LASSO estimation was studied in Medeiros and Mendes (2016).

We propose a 3-layer Sparse estimator for large-scale Network AutoRegressive models (TriSNAR). The estimation is conducted under the sparsity of the lags, groups and individuals. We develop an efficient and fast algorithm, sequentially optimizing the large-scale estimation for each of the three layers. An approximate optimization algorithm is also provided for the fast convergence of the regularized estimation. We determine the asymptotic properties of the sparse estimator and show that TriSNAR possesses the three properties of a good sparse estimator defined in Fan and Li (2001), namely, unbiasedness, sparsity and continuity. In an extensive simulation experiment, TriSNAR delivers good accuracy. By identifying the statistically significant lags, groups and individuals in the network, TriSNAR outperforms in terms of uncovering the true structure and can enhance the interpretability of the dynamic structure of the network.

As an illustration, we explore the applicability of TriSNAR to a network of 26 cryptocurrency exchanges with hourly Bitcoin pricing information from 4 July 2018 to 4 July 2019. Powered by blockchain technology, Bitcoin is transacted as a bor-

derless decentralized digital currency and has grown into an active global virtual money network with millions of accounts. The transactions occur in cryptocurrency exchanges all over the globe, where the assets are traded 24 hours, on weekends, and on holidays. However, there are many unknowns concerning the dynamic structure of the booming cryptocurrency markets. The choice of Bitcoin as an asset for the empirical study is motivated by a number of reasons. First, the continuous trading of Bitcoin (BTC) allows for a comparison of exchanges in different time zones. This is challenging for traditional exchanges that usually have closing hours. As a result, equity exchanges in America and Asia are not open at the same time. Second, we chose exchanges that trade BTC/USD; hence, we consider the same trading pair despite considering exchanges from all over the world. On traditional equity exchanges, trading usually takes place in the local currency, which would challenge a global comparison of a particular asset. Third, BTC is the most widely traded cryptocurrency against fiat currency, although it also features no concentration of trading volume at a particular exchange. Consequently, the identification of the leading exchanges is more challenging and requires a more thorough analysis than a pure judgment based on trading volume concentration. We find a meaningful and insightful dependence structure in the global BTC multi-market network. The serial dependence effect on the entire system exists only in the first two lags. There is a sparse-group effect, with only three exchanges being identified as leading markets, namely, Kraken in the US, Bitfinex in the UK, and Cex.io in Hong Kong, and sparse individual effects between the leading exchanges and all others. In comparison, the alternative estimators suggest a much denser structure for three lags, whereas their AIC is worse. TriSNAR also has good out-of-sample prediction accuracy.

There are several existing sparse estimators for VAR/NAR models. TriSNAR differs from them in the following aspects. Davis et al. (2016) and Kock and Callot (2015) focus on individual effects between two processes. The sparse estimators neither cover the group effect of one process on the entire network nor the serial dependence effect of the system. Chen et al. (2018) and Nicholson et al. (2017) consider either two of the three effects but do not cover all the effects. Song and Bickel (2011) also facilitates the triple sparsity; however, their approach does not have the three properties of a good estimator.

We present the model framework and provide a detailed description of the penalization operator with its theoretical properties in Section 2. Section 2.2 introduces the algorithms for TriSNAR, as well as the technique for generating the penalty pa-

rameters and the method for selecting the best model. In Section 3, we investigate the finite-sample performance in a simulation study. In Section 4, we illustrate the performance of the models on real data. Section 5 concludes. The code used in this paper is available at GitHub.

2 TriSNAR for Large-Scale Network AutoRegressive Models

Let $Y_t \in \mathcal{R}^d$ denote a vector of observations of a network with d-dimensional processes at time point $t \in \{1, \ldots, T\}$, with the length of the time period T. Assume there are p parameter matrices $\boldsymbol{A}_k$ with $k \in \{1, \ldots, p\}$, which are of dimension $d \times d$ and measure the serial dependence of the d processes. The Network AutoRegressive (NAR) model describes the serially linear dependencies between these processes. It is defined as

$$Y_t = A_0 + \sum_{k=1}^{p} \boldsymbol{A}_k Y_{t-k} + \epsilon_t \tag{1}$$

where $A_0 = (a_{1,0}, \ldots, a_{d,0})^\top$ is the intercept and $\epsilon_t = (\epsilon_{1,t}, \cdots, \epsilon_{d,t})^\top$ is a vector that is assumed to be independently and identically distributed with $\epsilon_t \sim (0, \Sigma)$. We assume that the model is stationary and ergodic, with all roots of the polynomial $\boldsymbol{I}_d - \sum_{k=1}^{p} \boldsymbol{A}_k Z^k$ lying outside the unit ball.

Our interest is in detecting the dynamic dependence in the network to help us understand the multi-market interactions. For an insightful interpretation, we assume a sparse network structure. In other words, the parameter matrices $\boldsymbol{A}_k$ for $k = \{1, \ldots, p\}$ are sparse, where the location and form of the sparsity are not predetermined. To perform regularized estimation, we introduce a penalty function $p_{\lambda_1,\lambda_2,\lambda_3}(\cdot)$ imposed on the lags, groups and individual parameters of $\boldsymbol{A}_k$ and estimate the model (1) by solving a regularized least squares optimization problem,

$$\min_{\mathbf{A}} \sum_{t=p+1}^{T} \frac{1}{2} ||Y_t - \sum_{k=1}^{p} \boldsymbol{A}_k Y_{t-k}||_F^2 + \sum_{k=1}^{p} p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_k), \tag{2}$$

where λ_1, λ_2 and λ_3 are tuning parameters for the sparsity. The penalty function, for a suitable choice of λs, should permit the estimator to have the properties of unbiasedness, sparsity and continuity and, further, the oracle property.

2.1 Estimator and Asymptotic Properties

Write $A_{k;ij}$ for the i,jth entry of the matrix $\boldsymbol{A}_k$ and write $||\cdot||_F$ for the Frobenius norm, defined by $||\boldsymbol{A}_k||_F = \sqrt{\sum_{i=1}^{d}\sum_{j=1}^{d} A_{k;ij}^2}$. We extract the diagonal of $\boldsymbol{A}_k$ and consider the autoregressive parameters separately as the $(d+1)$th group, $A_{d+1;j}$. In other words, we describe the autoregressive effects disentangled from the network effects. We use $A_{k;j}$ to denote the column (group) j with $j = 1, \ldots, d$ within the parameter matrix $\boldsymbol{A}_k$, yet without the jth parameter on the diagonal. Hence, $A_{k;j} = (A_{k;1j}, \ldots, A_{k;(j-1)j}, A_{k;(j+1)j}, \ldots, A_{k;dj})^\top$. As such, the groups have $(d-1)$ parameters except for the group of the autoregressive parameters, which has d parameters. We introduce a scaling parameter $d_j = (d-1)$ for $j = 1, \ldots, d$ and $d_j = d$ for $j = (d+1)$ to offset the impact of a mismatch between the number of parameters in the columns $(d-1)$ and the diagonal d and apply it to the regularization parameter λ_2 accordingly. We define the penalty function by

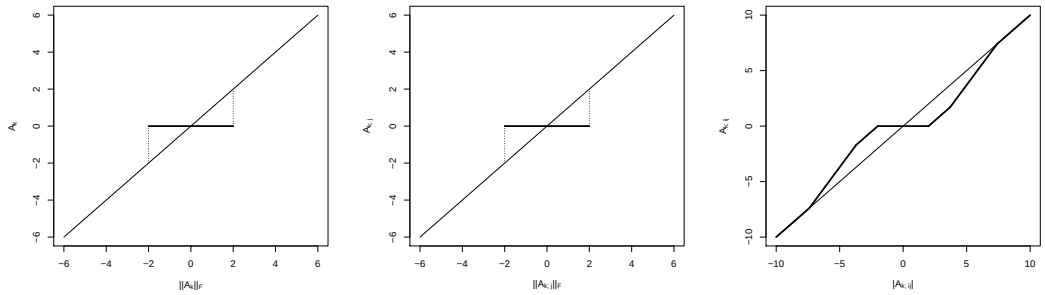
$$p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_k) = \begin{cases} d^2\lambda_1||\boldsymbol{A}_k||_F & ||\boldsymbol{A}_k||_F \leq d^2\lambda_1 \\ d_j\lambda_2||A_{k;j}||_F & ||A_{k;j}||_F \leq d_j\lambda_2 \wedge d^2\lambda_1 < ||\boldsymbol{A}_k||_F \\ \lambda_3|A_{k;ij}| & |A_{k;ij}| \leq \lambda_3 \wedge d_j\lambda_2 < ||A_{k;j}||_F \wedge \\ & d^2\lambda_1 < ||\boldsymbol{A}_k||_F \\ \frac{2b\lambda_3|A_{k;ij}| - |A_{k;ij}|^2 - \lambda_3^2}{2(b-1)} & \lambda_3 < |A_{k;ij}| \leq b\lambda_3 \wedge d_j\lambda_2 < ||A_{k;j}||_F \wedge \\ & d^2\lambda_1 < ||\boldsymbol{A}_k||_F \\ \frac{\lambda_3^2(b+1)}{2} & b\lambda_3 < |A_{k;ij}| \wedge d_j\lambda_2 < ||A_{k;j}||_F \wedge \\ & d^2\lambda_1 < ||\boldsymbol{A}_k||_F \end{cases}. \quad (3)$$

Note that the first case $d^2\lambda_1||\boldsymbol{A}_k||_F$ applies to the layer of lags. It is regularized by the magnitude of all parameters within $\boldsymbol{A}_k$ and scales the regularization parameter λ_1 by the number of parameters, d^2. The second case, $d_j\lambda_2||A_{k;j}||_F$, regularizes each group j of $\boldsymbol{A}_k$. The 3rd to 5th cases build the regularization operator for the individual parameters first, the soft- and tapering-off threshold, $\lambda_3|A_{k;ij}|$ and $\frac{2b\lambda_3|A_{k;ij}| - |A_{k;ij}|^2 - \lambda_3^2}{2(b-1)}$, and the non-regularized case $\frac{\lambda_3^2(b+1)}{2}$. Here, we require $b > 2$.

The penalty function (3) combines the advantages of two hard-thresholding functions and a soft-thresholding and a tapering-off function. Figure 2 shows the performance of the penalization for examples of sequences of λ by setting the scaled tuning

Figure 2: The three plots illustrate the three states of TriSNAR. The first plot shows the penalization of each k lag $\boldsymbol{A}_k$, which is penalized with a hard-thresholding function. When the penalizing value λ_1 is reached, the values are unpenalized. The second penalization function for the group steps is illustrated in the second figure, which is also a hard-thresholding function. Next, the individual penalization function becomes active, which corresponds to the form of SCAD for regression models.

(a) Lag penalization (b) Group penalization (c) Individual penalization



parameters to 2 and $b = 3.7$. The hard-thresholding functions are applied to the lagged parameter matrices (Fig. 2a) and the groups (Fig. 2b). The soft-thresholding and tapering-off functions, similar to the SCAD penalty for regression models, are used for the individual parameters (Fig. 2c). This function ensures that $\boldsymbol{A}_k$ and $A_{k;j}$ are only penalized until the values $d^2\lambda_1$ and $d_j\lambda_2$ are reached. This favors the unbiasedness of the resulting NAR model. This formulation is also grounded in the rationale that a group inside of a lagged matrix can only be unpenalized when the entire matrix is not subject to the hard-thresholding parameter λ_1.

Applying the three-layer penalty function (3), we obtain the estimator of $\boldsymbol{A}_k$ according to several cases:

$$\boldsymbol{A}_k = \begin{cases} 0 & ||\boldsymbol{A}_k||_F \leq d^2\lambda_1 \\ 0 & ||A_{k;j}||_F \leq d_j\lambda_2 \ \wedge \ d^2\lambda_1 < ||\boldsymbol{A}_k||_F \\ sgn(A_{k;ij})(|A_{k;ij}|-\lambda_3)_+ & |A_{k;ij}| \leq 2\lambda_3 \ \wedge \ d_j\lambda_2 < ||A_{k;j}||_F \wedge d^2\lambda_1 < ||\boldsymbol{A}_k||_F \\ \frac{(b-1)A_{k;ij}-sgn(A_{k;ij})b\lambda_3}{(b-2)} & 2\lambda_3 < |A_{k;ij}| \leq b\lambda_3 \ \wedge \ d_j\lambda_2 < ||A_{k;j}||_F \wedge d^2\lambda_1 < ||\boldsymbol{A}_k||_F \\ A_{k;ij} & b\lambda_3 < |A_{k;ij}| \wedge d_j\lambda_2 < ||A_{k;j}||_F \wedge d^2\lambda_1 < ||\boldsymbol{A}_k||_F \end{cases} \tag{4}$$

We derive the theoretical properties of the estimator (4). Denote $b_T = \max(\frac{\partial p}{\partial A_{k;ij}} : A_{k;ij} \neq 0)$. The proofs are given in detail in the Appendix.

Theorem 1 *Assume that the assumptions for model (1) hold. If* $\max\{\frac{\partial^2 p}{\partial A^2_{k;ij}(A_{k;ij})} : A_{k;ij} \neq 0\} \to 0$, *then there exists a local maximizer* $\widehat{\boldsymbol{A}}$ *for (2) such that* $||\widehat{\boldsymbol{A}} - \boldsymbol{A}_0||_F = \mathcal{O}(T^{-1/2} + b_T)$.

When the hypotheses of Theorem 1 are fulfilled, the regularization parameters λ_1, λ_2, λ_3 converge to 0 faster than $T^{-1/2}$. Hence, there exists a local maximizer of (2), which converges at speed $\sqrt{T}$.

Next, we show that the estimator possesses the sparsity property and hence is capable of selecting the model parameters in a sparse system. Denote by $CL(\cdot)$ the constrained likelihood.

Lemma 1 *Assume that the assumptions for model (1) hold. If* $\lambda_{1,T}, \lambda_{2,T}, \lambda_{3,T} \to 0$ *and* $\sqrt{T}b_T \to \infty$ *as* $T \to \infty$, *then with probability tending to 1, for any given* $\boldsymbol{A}_{k;d_1d_1}$ *satisfying* $||\boldsymbol{A}_{k;d_1d_1} - \boldsymbol{A}_{k;d_1d_1;0}||_F = O_p(T^{-1/2})$ *and any constant* Q,

$$CL(\boldsymbol{A}_{k;d_1d_1}, 0) = \underset{||\boldsymbol{A}_{k;-d_1-d_1}||_F \leq QT^{-1/2}}{max} CL(\boldsymbol{A}_{k;d_1d_1}, \boldsymbol{A}_{k;-d_1-d_1}),$$

hence

$$P(\boldsymbol{A}_{k;-d_1-d_1} = 0) \to 1.$$

Finally, we show that the estimator possesses the oracle property, i.e., it chooses the true model as if it were a theoretical estimator that knows the true model structure.

We define

$$F = \left[p''_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{1;11}), \cdots, p''_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{p;d_1d_1})\right]$$

as a $pd_1 \times pd_1$ symmetric matrix containing the second derivatives of the penalty function and

$$G = \left[p'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A_{1;11})sgn(A_{1;11}), \cdots, p'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A_{p;d_1d_1})sgn(A_{p;d_1d_1})\right]$$

as a $d_1 \times pd_1$ matrix containing the first derivatives of the penalty function.

Theorem 2 *Assume that the assumptions for model (1) hold. If* $\lambda_{1,T}, \lambda_{2,T}, \lambda_{3,T} \to 0$ *and* $\sqrt{T}b_T \to \infty$ *as* $T \to \infty$*, then with probability tending to 1, the root-T consistent local maximizer* $\boldsymbol{A} = [A_{\cdot;d_1 d_1}, A_{\cdot;-d_1-d_1}]$ *from Theorem 1 must satisfy*

1. *Sparsity:* $A_{\cdot;-d_1-d_1} = \mathbf{0}$
2. *Asymptotic normality:*

$$\sqrt{T}((\boldsymbol{A}_{\cdot;d_1 d_1} - \boldsymbol{A}^0_{\cdot;d_1 d_1})(I(\boldsymbol{A}^0_{\cdot;d_1 d_1}) + F) + G) \xrightarrow{d} N(0, I(\boldsymbol{A}^0_{\cdot;d_1 d_1})) \tag{5}$$

in distribution, where $I(\boldsymbol{A}^0_{\cdot;d_1 d_1})$ *is the Fisher Information knowing that* $A^0_{\cdot;-d_1-d_1} = \mathbf{0}$.

2.2 Algorithm

We develop two algorithms to implement the estimation: an active-set algorithm based on coordinate-wise descent and an approximating algorithm with early termination for fast solutions. The latter switches from the active-set approach after a given number of iterations and continues searching for a solution with the FISTA algorithm, Beck and Teboulle (2009), solely for the active parameters already discovered.

In the following, we describe the algorithms in more detail. The vector Y_t without the jth process is denoted by $Y_{t,-j}$, and recall that $Y_{t,j}$ represents the jth process. Define $sort(\cdot)$ as the operator that sorts the variables in decreasing order. We use an active-set algorithm to sequentially evaluate the three-layer parameters with the order of lags, groups and individuals. The algorithm is initialized with completely sparse parameter matrices, meaning that all parameters are set equal to 0. Since we assume that the parameter matrix is sparse, this can be considered an appropriate starting point. First, we sort the lags, groups and individual parameters according to the proportion of variance unexplained by them. The sorting ensures that, the algorithm optimizes first the parameters that explain more of the variability of the system. In the algorithm, we iterate over active lags in an outer optimization loop. We iterate over the active groups in each of the identified active lags in a similar manner. The active individual elements are identified and optimized only from the active group sets. In each iteration, we construct the residuals ϵ_k, $\epsilon_{k;-j}$ and $\epsilon_{k;ij}$, reflecting the unexplained variance, based on which we optimize the parameters for the active sets. While iterating, non-active lags, groups and individual parameters

are removed from further analysis.

The implementations are formulated as Algorithm **TriSNAR.lag**, Algorithm **TriSNAR.group** and Algorithm **TriSNAR.individual**.

1. TriSNAR.lag is the first outer loop algorithm. It evaluates the tuning parameter λ_1 to identify the lag parameter matrix carrying sufficient information. We sort the matrices in decreasing order according to the explained variance as reflected in the residuals ϵ_k. In each iteration step (m_1), $\boldsymbol{A}_k$ with little or no explanatory power is forced to be 0. Otherwise, with a sufficiently large explained contribution to the variance, i.e., $\epsilon_k > d^2\lambda_1$, we continue to estimate the lag parameter matrix with Algorithm TriSNAR.group.

2. TriSNAR.group evaluates the groups on the sequence λ_2. Similarly, we order them according to the explained variance, and the algorithm is iterated with the residual $\epsilon_{k;-j}$. In each iteration step (m_2), $A_{k;j}$ is set to 0 in the case of little or no explanatory power. Otherwise, if $\epsilon_{k;-j} > d\lambda_2$, we continue the implementation with Algorithm TriSNAR.individual.

3. TriSNAR.individual is used to optimize the individual parameters inside an active group. It is a coordinate-wise descent optimization under the sequence λ_3 and with residual $\epsilon_{k;ij}$ according to estimator (4). In each iteration step (m_3), the contribution to the variance is evaluated. In case there is little or no explanatory power, $A_{k;ij}$ is set to 0. Otherwise, the non-zero parameter is estimated.

4. The algorithms are repeated with iteration steps m_1, m_2, m_3 until all parameter matrices have converged.

The implementation depends on the hyperparameters η_1, η_2, η_3, which are user specified. The parameter b for the estimator of the individual parameters can also be set as a sequence. However, this part of the estimator corresponds to SCAD; hence, we follow the recommendation of Fan and Li (2001) and set $b = 3.7$. The regularization sequences remain to be selected, i.e., the values of the tuning parameters λ_1, λ_2, and λ_3. Usually, cross-validation is used to determine the sequence. However, due to the time dependence in our model, cross-validation is not very suitable. We choose the tuning parameters using the out-of-sample AIC. This is consistent with Bańbura et al. (2010), Song and Bickel (2011), Nicholson et al. (2017) and Chen et al. (2018).

The run time depends on the size of the sequence of tuning parameters λ_1, λ_2 and λ_3. Naturally, a more granular penalization sequence leads to a longer runtime of the optimization procedure. In our case, it is a halving sequence approaching 0 for the individual parameter penalization (λ_3) and a diminishing sequence by 1/5 for the group (λ_2) and lag regularization (λ_1), also approaching 0. This default setting provides stable performance in a later simulation study.

For very high dimensions or difficult specifications, we also propose an approximating algorithm that can find a faster solution but may lead to a local optimizer. In the approximating algorithm, the described procedure interrupts after a specified number of iterations (s). At this point, the by then identified lagged matrices and groups are considered active, and the remaining ones are set to 0. Then, we apply the FISTA algorithm to the parameters that are considered active. FISTA is not applied in the active-set algorithm because the convergence takes longer over the active sets. Coordinate-wise descent, delivering similar parameter estimates, is beneficial because an easier parameter-updating step improves the speed due to iterating over the sets of parameters. However, FISTA performs well for the approximating algorithm because it can optimize faster over all sets at once instead of iterating over them, which improves the estimation time of the parameters. In the numerical analysis, the results derived with the active-set optimization algorithm are denoted by TriSNAR$_G$ and the approximate result by TriSNAR$_A$.

3 Simulation

We investigate the finite-sample performance of the proposed TriSNAR estimator with simulations. We consider various scenarios, ranging from simple cases with only one active lag to relatively complex cases with mixed lag, group and individual sparsity. We evaluate the ability to detect sparsity, the accuracy of the parameter estimation and prediction, as well as the runtime. We compare TriSNAR, derived with the active-set (TriSNAR$_G$) and approximating algorithm (TriSNAR$_A$), with a number of competing models. Two of the models, namely, LASSO and SCAD, only penalize for the individual parameters, while one other model, 'tapered LASSO' (TLASSO), Nicholson et al. (2014), penalizes for the lag structure by tapering off the effect of the individual parameter penalization. Consequently, it regularizes the lags and individual parameters. The estimation is implemented in the BigVAR package, Nicholson et al. (2019).

Algorithm 1 : TriSNAR.lag

Input: Data Y_t for all $t = 1, \ldots, T$
Output: Adjacency matrix $\boldsymbol{A}$

Initialization $\boldsymbol{A} = 0$, $m_1 = 1$
for $k = 1, \ldots, p$ **do**
  $\epsilon.lag_k = \sqrt{\sum_{t=p+1}^{T}(Y_{t-k}^{\top}(Y_t - \sum_{l=1\setminus k}^{p} \boldsymbol{A}_l Y_{t-l}))^2}$
end for
$order.lag = sort(\{\epsilon.lag_k\}_{k=1}^{p})$

while $\text{vec}\{\boldsymbol{A}^{(m_1)} - \boldsymbol{A}^{(m_1-1)}\} < \eta_1$ and $m_1 \leq s$ **do**
  for $k \in order.lag$ **do**
    $m_2 = 1$
    $\epsilon_k = \sqrt{\sum_{t=p+1}^{T}(Y_{t-k}^{\top}(Y_t - \sum_{l=1\setminus k}^{p} \boldsymbol{A}_l^{(m_2)} Y_{t-l}))^2}$
    if $\epsilon_k \leq d^2\lambda_1$ **then** $\boldsymbol{A}_k^{(m_2)} = 0$
    else
      while $\boldsymbol{A}_k^{(m_2)} - \boldsymbol{A}_k^{(m_2-1)} < \eta_2$ **do**
        TriSNAR.group$(\{Y_t\}_{t=1}^{T}, \boldsymbol{A}_k^{(m_2)})$
        $m_2 = m_2 + 1$
      end while
    end if
    $m_1 = m_1 + 1$
  end for
end while

Algorithm 2 : TriSNAR.group

Input: Data Y_t for all $t = 1, \ldots, T$; $\boldsymbol{A}_k$
Output: Adjacency matrix $\boldsymbol{A}_k$

for $j = 1, \ldots, d$ **do**
  $\epsilon.group_j = \sqrt{\sum_{t=p+1}^{T} (Y_{t-k,-j}^{\top} (Y_t - \sum_{l=1\setminus k}^{p} \boldsymbol{A}_l Y_{t-l})_{-j})^2}$
end for
$order.group = sort(\{\epsilon.group_j\}_{j=1}^{d})$

for $j \in order.group$ **do**
  $m_3 = 1$
  $\epsilon_{k;-j} = \sqrt{\sum_{t=p+1}^{T} (Y_{t-k,-j}^{\top} (Y_t - \sum_{l=1\setminus k}^{p} \boldsymbol{A}_l^{(m_2)} Y_{t-l})_{-j})^2}$
  if $\epsilon_{k;-j} \leq d\lambda_2$ **then** $A_{k;j}^{(m_2)} = 0$
  else
    while $A_{k;j}^{(m_3)} - A_{k;j}^{(m_3-1)} < \eta_3$ **do**
      TriSNAR.individual$(\{Y_t\}_{t=1}^{T}, A_{k;j}^{(m_3)})$
      $m_3 = m_3 + 1$
    end while
  end if
end for

Algorithm 3 : TriSNAR.individual

Input: Data Y_t for all $t = 1, \ldots, T$; $A_{k;j}$
Output: Adjacency matrix $A_{k;j}$

for $i = 1, \ldots, d$ **do**
  $\epsilon_{k;ij} = \sqrt{\sum_{t=p+1}^{T} (Y_{t-k,j}^{\top} (Y_{t,i} - \sum_{l=1\setminus k}^{p} A_{l,ij}^{(m_2)} Y_{t-l,j}))^2}$
  if $|\epsilon_{k;ij}| \leq 2\lambda_3$ **then** $z = sgn(\epsilon_{k;ij})(|\epsilon_{k;ij}| - \lambda_3)_+$
  else if $2\lambda_3 < |\epsilon_{k;ij}| \leq b\lambda_3$ **then** $z = \frac{(b-1)\epsilon_{k;ij} - sgn(\epsilon_{k;ij})b\lambda_3}{(b-2)}$
  else if $b\lambda_3 < |\epsilon_{k;ij}|$ **then** $z = \epsilon_{k;ij}$
  end if
  $A_{k;ij} = z / \sum_{t=p+1}^{T} (Y_{t-k,j}^{\top} Y_{t-k,j})$
end for

In the TriSNAR$_A$ optimization, we fix the identified lags, groups and parameters and activate FISTA after $m_1 = 5$ iterations, which provided frequently similar results in the simulation study. This value may not be sufficient for a different simulation study setting or a different dataset. With dimensions larger than $d = 50$, we only derive the results for the approximating algorithm due to a slow optimization speed in high dimensions. The code used in this paper as well as files containing the exact settings required for a replication of the simulation study are available at GitHub.

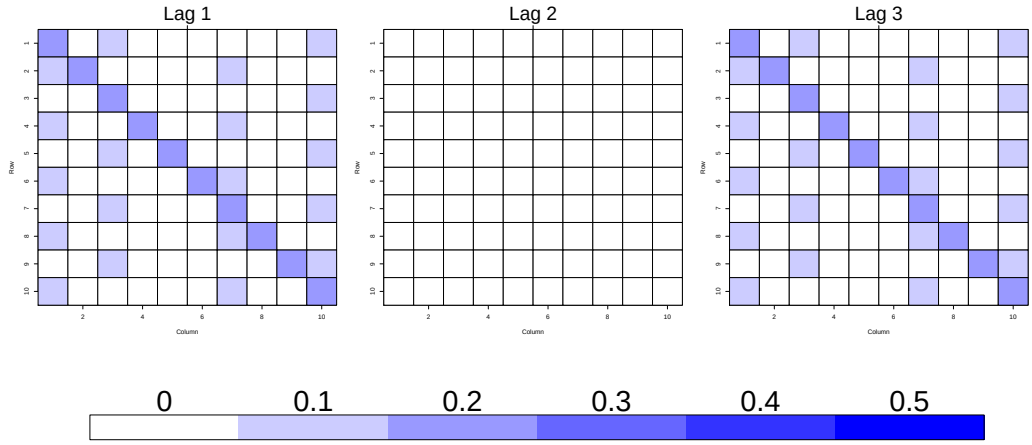
3.1 Set up

We consider networks with $d \in \{10, 20, 50, 100\}$ time series and lengths of $T \in \{100, 200, 500, 1,000\}$ data points. As we fix the active parameter matrices up to $p = 3$ lags, we need to derive 300, 1,200, 7,500 and 30,000 unknown parameters of the adjacency matrices $\boldsymbol{A}_k$, $k = 1, \cdots, 3$. We design 6 model specifications and refer to each scenario by an abbreviation. We assign the capital letter D for specifications with a diagonal parameter matrix and the capital letter M for a medium persistent case containing group- and individual sparsity. A digit postfixed to the abbreviation indicates the number of active lag(s) in the specification. For example, an NAR model with the first and third lag being active with medium persistence is referred to as M1/M3. The details are listed below.

- *D1*: The first lag is a diagonal matrix with all diagonal parameters being 0.5. The other two lags are zero. In other words, there only exists autocorrelation in the network.
- *D2*: Similar specification as D1, except the active parameters appear in lag 2. The first and third lags are nonactive.
- *M1*: The first lag is active, with all diagonal parameters being of magnitude 0.5. There are 4 active columns $(1, 3, 7, 10)$ with alternating values of 0 and 0.15. The companion matrix of this specification has the largest eigenvalue of 0.75, yielding a medium persistence. The simultaneous evaluation of the lag, group and individual effects are required.
- *M2*: Similar to specification M1, except the active parameters occur in lag 2.
- *M1/M3*: Two lags are active, namely, the first and third lags. Both have the diagonal parameters of 0.2. The columns $1, 3, 7, 10$ are active with alternating

values 0 and 0.1. The companion matrix of the specification has the largest eigenvalue of 0.87, indicating a medium persistence (see Figure 3).

Figure 3: Model specification: M1/M3



- *NS1*: We also consider a non-sparse specification, where only the first lag is active, yet there is no group and individual sparsity in the active matrix. The magnitude of the parameter decays exponentially away from the diagonal. It starts at 0.4 on the diagonal, and the off-diagonal parameters take on the values resulting from the formula $A_{i,j} = (-1)^{|i-j|}0.4^{|i-j|+1}$. In other words, all parameters are active; however, those far from the diagonal become quite small.

For specifications featuring only individual effects, e.g., NS1, the classic regularizations LASSO and SCAD are expected to perform well since both models are designed for such settings. When group effects are added, e.g., M1 and M2, TriSNAR is expected to excel since its three-layer design is appropriate, whereas the other models do not consider group effects. In the complex case of, e.g., M1/M3 with lag, group and individual effects present, TriSNAR and TLASSO should behave well as both regularize the lag and individual effects, though TriSNAR has the advantage of also considering the group effects.

In the data generation, the innovations are assumed to be i.i.d. Gaussian with $\epsilon \sim N(0, I_d)$. In all, there are $4(d) \times 4(T) \times 6(specifications) = 96$ experiments. For each experiment, we carried out 100 simulations and computed the average performance. Moreover, we split each dataset into a training, validation and testing dataset, each with length T. We estimated the models for all combinations of λ_1, λ_2, and λ_3-

sequences on the training dataset. The best performing model was evaluated based on the validation dataset with AIC. The resulting model was then implemented for the test sample for its forecasting accuracy.

3.2 Evaluation Criteria

The performance was evaluated in three aspects: pattern, accuracy and speed.

- To evaluate the pattern identification, we computed the False Negative (FN) and False Positive (FP) rates on the estimated sets. FN refers to active set's being falsely identified as null, namely, under-detection or overly sparse. FP refers to the set's being wrongly identified as active, namely, overdetection or overly dense. It is natural that the lower these two measures are, the better the performance. Given the three-layer sparsity, there are then 6 metrics: FN.l and FP.l for lags, FN.g and FP.g for groups, and FN.e and FP.e for individual elements. In the case of perfect detection, namely, all 6 metrics are zero, we conclude that the true pattern was identified.

- Accuracy is measured using the Mean Absolute Error (MAE). Again, there are three different metrics. MAE.para refers to the estimation accuracy, computed based on the difference between the true and estimated parameters. MAE.res refers to the prediction accuracy, which is calculated based on the residuals between the true values of the time series and the predicted values based on the model. In other words, it evaluates in-sample on the training dataset. MAFE.res refers to the forecast error, an out-of-sample measure based on the testing dataset. In all the accuracy metrics, a low value indicates good accuracy.

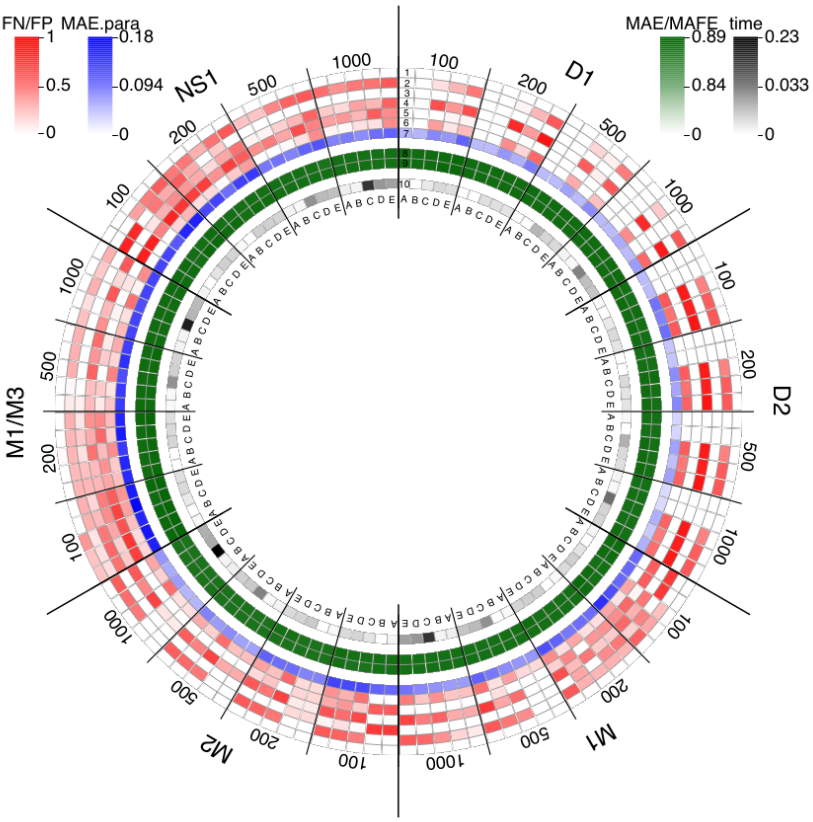
- Time/combination is an indicator to measure the speed. We report the time in seconds it took on average to derive the model per combination of penalization sequences.

3.3 Roseplots: Summary of the Results

The Figures 4, 5, 6 and 7 summarize the performance of TriSNAR and competing estimators along with the 10 measurements in the 96 experiments, separated according to model specification. Each of the 4 roseplots shows the performance of the

simulation study for a given dimension for all 6 scenarios, all number of observations and the 10 evaluation criteria. Each roseplot is separated into 6 sections, one for each scenario. Within each section, 4 subsections are assigned for the number of observations: $T = \{100, 200, 500, 1000\}$. For each of these subsections, 5 columns of rectangles are provided, named A, B, C, D, E. The naming convention refers to TriSNAR$_G$ (A), TriSNAR$_A$ (B), TLASSO (C), SCAD (D) and LASSO (E).

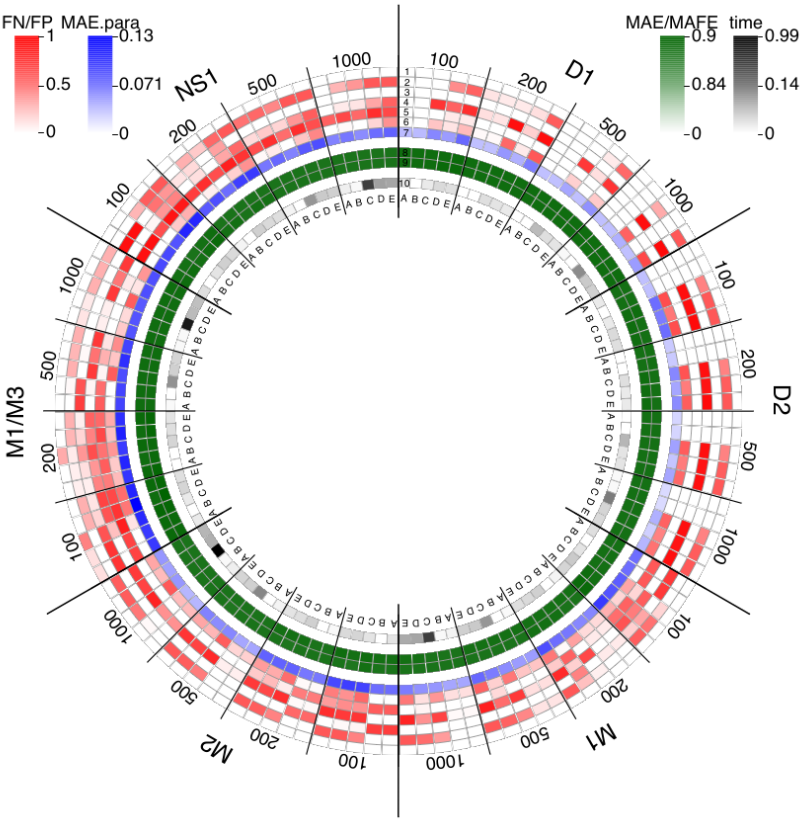
Figure 4: Roseplot for $d = 10$



The entire circle has 10 tracks, each of which represents another evaluation criterion. The most outer track is referred to as 1, and the most inner track is referred to as 10. The 6 most outer tracks are for the False Negative and False Positive criteria: FN.l (1), FP.l (2), FN.g (3), FP.g (4), FN.e (5), FP.e (6). The FN and FP rates vary between 0 and 1, whereas the color palette goes from white (0) to red (1). No False Negatives and no False Positives are the best possible outcomes; hence, the more white or shallow red the rectangles are, the better. Track 7 reports the MAE.para with a color palette from white (0) to blue (maximum value of MAE.para). Again, the lower the value, the better; hence, white or shallow blue rectangles are preferable. The MAE.res is reported via track 8 and the MAFE.res via track 9. The color

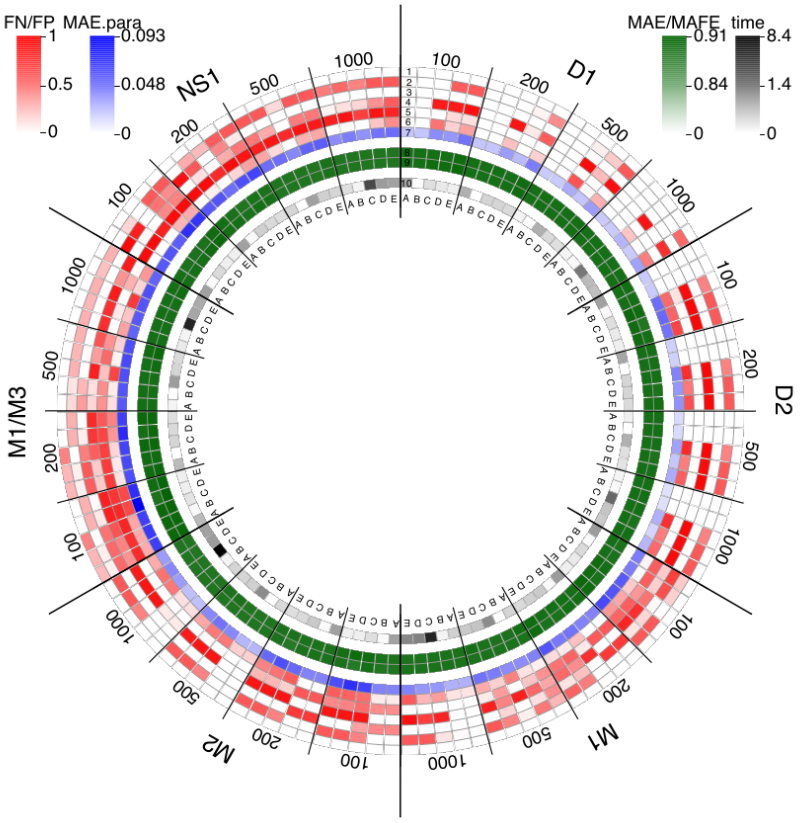
palette goes from white (0) to green (maximum value of MAE.res/MAFE.res). Since these evaluation criteria reflect the error terms, the smaller the values are, the better. Thus, white and shallow green is preferable. The most inner track, 10, reports the runtime per combination of λ values. The color palette ranges from white (0) to black (maximum value of runtime per comparison). Certainly, a faster runtime of the code is preferred; hence, white and gray rectangles are better.

Figure 5: Roseplot for $d = 20$



It can be observed that the rectangles associated with TriSNAR$_G$ and TriSNAR$_A$ with $d = \{10, 20, 50\}$ for the FN and FP values are always white for D1 and D2. For $d = 100$, the rectangles become all white for $T \leq 200$. However, all the other models show mostly dark red rectangles on tracks 2, 4, and 6, which reflect the FP rates. Hence, the competing estimators overparametrize the models; therefore, they do not provide the true model. For D1 and a higher number of observations, SCAD also returned the true model, although only TriSNAR identified the true structure in both cases and for a small number of observations. For scenarios M1 and M2, overall, TriSNAR$_G$ and TriSNAR$_A$ performed best in terms of uncovering the true structure. For $T = 100$, both models provided a too sparse model, while the competing estima-

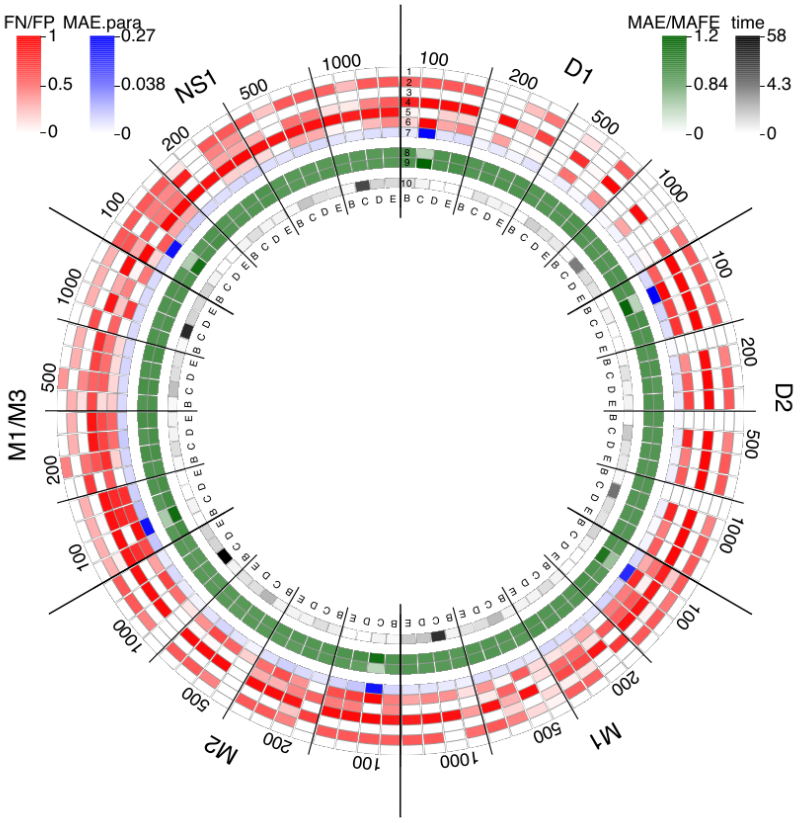
Figure 6: Roseplot for $d = 50$



tors selected mostly the incorrect parameters and the incorrect structure. We observe that all rectangles of C, D, and E are shaded red, which indicates that incorrect parameters were chosen, whereas the true parameters were not included in the model. Hence, the TriSNAR solution is preferable even though the solution was too sparse. For $T \leq 200$, TriSNAR$_G$ and TriSNAR$_A$ became continuously better, whereas estimators C, D, and E overparameterized. For M1, M3 and NS1 scenarios, exploring the underlying structure became more difficult for all involved estimators. It can be observed that the TriSNAR estimators gave a more accurate estimate, which was inferred from the observation that the respective rectangles are white or have more shallow red compared to the competing estimators. TriSNAR$_G$ and TriSNAR$_A$ also overparameterized, however, less intense than the competing estimators.

Considering the MAE.para, track 7, it can be observed that both TriSNAR estimators provided more accurate parameter estimations in most cases. When they did not outperform, the performance was similar to the other models. This result corresponds with the observation that TriSNAR$_G$ and TriSNAR$_A$ provided more accurate

Figure 7: Roseplot for $d = 100$



solutions in terms of identifying the underlying model structure.

Tracks 8 and 9 provided the results for MAE.res and MAFE.res. Obviously, for all models within one subsection, the results were comparable for these evaluation criteria. This infers that all models provide similar model performance, though, as was discussed before, models C, D, and E tend to overparameterize the models. In contrast, TriSNAR$_G$ and TriSNAR$_A$ can identify the true model structure more often while providing the same model performance.

Track 10 compares the runtime per combination of λ sequences. We observe that the TriSNAR estimators had the fastest runtime, whereas the TLASSO model (C) frequently had a much longer runtime than the other estimators. SCAD and LASSO performed comparably in terms of runtime, and they were derived faster than TLASSO.

The good performance of TriSNAR over all evaluation metrics was possible due to its properties. With the oracle property, it has the potential to find the true pattern,

especially with a large sample size. Moreover, TriSNAR is an unbiased estimator, which allows it to achieve low FN and FP. In summary, TriSNAR outperformed in the majority of the scenarios and cases, illustrating its applicability to large-scale networks.

3.4 Results for a Detailed Example

To provide more insights into the performance, we illustrate the average performance of the estimators, TriSNAR and the alternatives, in an example: the case $d = 50$ and $T = 100$ in Table 2. The detailed results of the other specifications are available in the Appendix. This shows that TriSNAR is superior in most cases, except specifications *M1* and *NS1*. For the easiest specifications, *D1* and *D2*, where only the first lag contains active parameters while the algorithm searches in an NAR(3) framework, TriSNAR$_{G/A}$ found the true pattern with even a small sample size $T = 100$, outperforming the TLASSO, LASSO and SCAD methods. Moving to a more challenging case with one or two active lags and all kinds of sparsity in the lags, groups and individuals, *M2* and *M1/M3*, TriSNAR$_{G/A}$ outperformed again and was close to identifying the true pattern, indicated by the low FN and FP. With *M1/M3*, we observe that TriSNAR$_G$ and TriSNAR$_A$ returned the best performance. From FN.l and FP.l shown for *M2*, we see that TriSNAR$_{G/A}$ tended to regularize the third lag, while the other models tended to include the second lag, which was non-active. In specification *M1*, TLASSO outperformed by having the lowest FNs for the lags, groups and individual parameters, whereas TriSNAR provided better FPs.

It would probably be interesting to compare the performance in specification *NS1*, where LASSO or SCAD would be expected to outperform, as only the first lag is active and dense. It turns out that TLASSO gave the best result. Figure 8 allows a detailed visual inspection for the D1 case, where we depict the true pattern and the estimated patterns using different estimators. It can be observed that TriSNAR returned the true model for each of the 100 simulated datasets, whereas TLASSO and LASSO strongly overparametrized. In particular, LASSO returned a poor result by identifying all 3 lags as important instead of only the first lag.

3.4.1 Comparison of derivation time

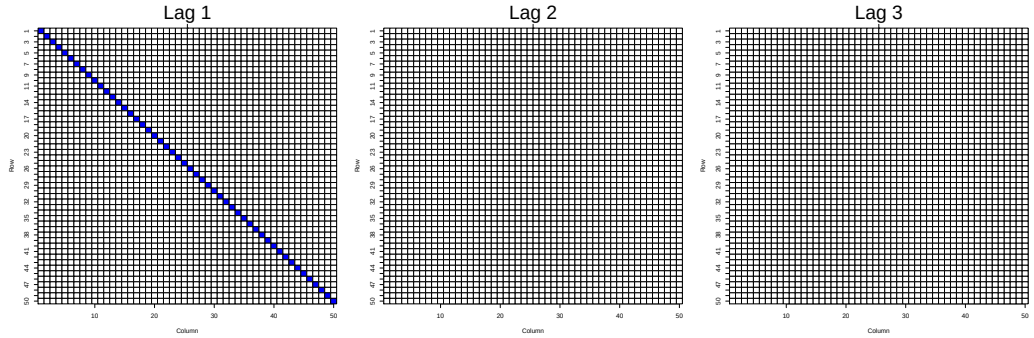
Table 1 shows the time the estimation took for each sample size $T \in \{100, 200, 500, 1000\}$ for TriSNAR$_G$. The results are compared with the simulations with $d = 10$ time series for 3, 5 and 7 lags in the algorithmic specifications. The computational time increases when more lags are involved; however, with a higher number of observations, the computation time mostly decreases. This is particularly true for 5 and 7 lags. One observes that the computational time for those models only involving the diagonal, *D1* and *D2*, is comparable to models involving network effects of medium persistence, *M1*, *M2*, *M1.M3*. Interestingly, the estimation time of the models is lower when the third lag is also active, meaning *M1. M3*, than for its peers with only the first or second lag active (*M1*, *M2*). Compared with the computational time with the competing models in the simulation study, it can be observed that TriSNAR$_{G/A}$ has a considerably lower computational time for each combination of λ's than the competing models; see, e.g., Table 9. The granularity of the λ sequences defines the performance. The lesser combinations, namely, a less granular grid of λ, reduce the overall runtime of the models and are therefore crucial for applying any of the reported methods.

Table 1: Average duration of derivation in seconds for simulations with 100, 200, 500, 1000 observations and for 3, 5, 7 lags for the specifications D1, D2, M1, M1/M3, M2, NS1. For the sake of brevity only the simulations for 10 time series are shown, however the results for larger systems give comparable results.

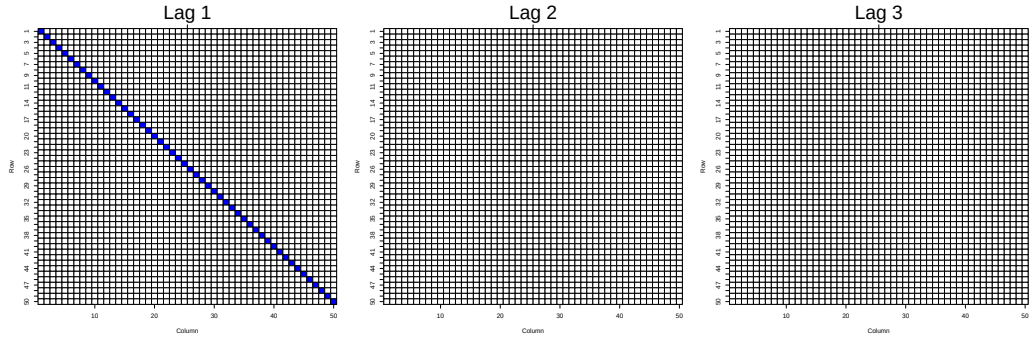
	100			200			500			1000		
	3	5	7	3	5	7	3	5	7	3	5	7
D1	68	317	623	52	145	335	73	142	267	122	391	387
D2	67	255	722	50	121	252	69	159	314	107	211	364
M1	98	275	737	87	209	420	107	219	441	160	307	520
M1/M3	69	240	682	44	113	253	89	165	283	147	282	482
M2	88	275	749	72	172	351	102	224	397	143	300	527
NS1	99	284	777	93	239	506	121	240	410	176	347	627

Figure 8: Comparison of the modeling performance of TriSNAR, TLASSO and LASSO for scenario *D1* with $d = 50$ and $T = 100$. The solution for SCAD was comparable to LASSO. The blue entries in the matrices represent parameters that are active in at least one simulated dataset.

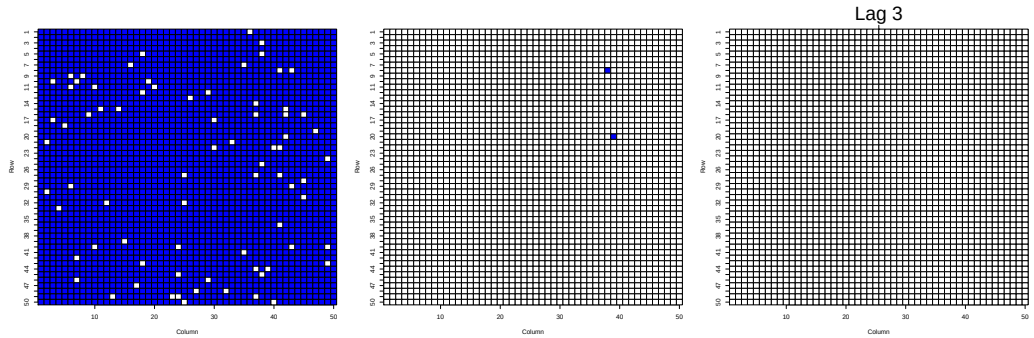
(a) True Model



(b) TriSNAR



(c) TLASSO



(d) LASSO

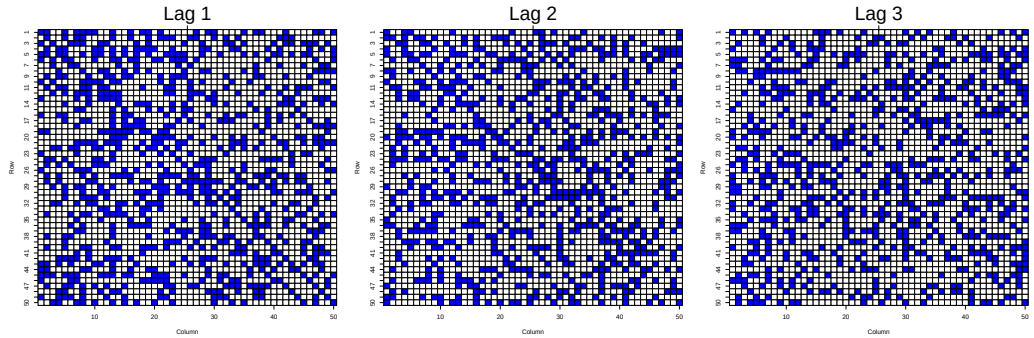


Table 2: Simulation $d = 50$ with 3 lags, $T = 100$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**3.04**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**0.33**
	TLASSO	0.00	0.01	0.00	0.90	0.00	0.57	0.04	0.85	0.86	1.27
	SCAD	0.00	0.65	0.00	0.90	0.18	0.22	0.04	0.85	0.87	0.89
	LASSO	0.00	0.67	0.00	0.95	0.15	0.38	0.05	0.87	0.88	0.79
D2	TriSNAR$_G$	**0.00**	**0.01**	**0.00**	**0.02**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**3.24**
	TriSNAR$_A$	**0.00**	**0.01**	**0.00**	**0.02**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**0.44**
	TLASSO	0.00	0.50	0.00	0.97	0.13	0.61	0.06	0.87	0.88	1.27
	SCAD	0.00	0.67	0.00	0.98	0.05	0.63	0.06	0.81	0.87	0.91
	LASSO	0.00	0.67	0.00	0.99	0.03	0.75	0.05	0.84	0.86	0.81
M1	TriSNAR$_G$	0.00	0.22	0.78	0.24	0.65	0.06	0.05	0.84	0.84	3.51
	TriSNAR$_A$	0.00	0.20	0.72	0.20	0.64	0.06	0.05	0.83	0.84	0.46
	TLASSO	**0.00**	**0.00**	**0.05**	**0.58**	**0.44**	**0.23**	**0.06**	**0.85**	**0.86**	**1.35**
	SCAD	0.00	0.66	0.20	0.84	0.66	0.37	0.06	0.83	0.86	0.95
	LASSO	0.00	0.67	0.13	0.87	0.63	0.43	0.07	0.86	0.87	0.84
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.50**	**0.00**	**0.46**	**0.12**	**0.05**	**0.82**	**0.83**	**3.32**
	TriSNAR$_A$	**0.00**	**0.00**	**0.50**	**0.00**	**0.46**	**0.12**	**0.05**	**0.82**	**0.83**	**0.50**
	TLASSO	0.00	0.50	0.38	0.93	0.59	0.60	0.07	0.85	0.86	1.22
	SCAD	0.00	0.67	0.02	0.93	0.51	0.62	0.08	0.78	0.86	0.82
	LASSO	0.00	0.67	0.01	0.94	0.50	0.64	0.07	0.83	0.85	0.73
M1/M3	TriSNAR$_G$	**0.29**	**0.09**	**0.61**	**0.44**	**0.84**	**0.34**	**0.07**	**0.88**	**0.89**	**2.50**
	TriSNAR$_A$	0.30	0.09	0.61	0.51	0.83	0.35	0.07	0.87	0.89	0.34
	TLASSO	0.50	0.00	0.57	0.69	0.87	0.36	0.07	0.89	0.89	1.15
	SCAD	0.00	0.33	0.11	0.88	0.85	0.72	0.09	0.83	0.91	0.81
	LASSO	0.00	0.33	0.06	0.90	0.81	0.74	0.08	0.87	0.89	0.71
NS1	TriSNAR$_G$	0.00	0.04	0.98	0.03	0.98	0.01	0.06	0.84	0.84	3.34
	TriSNAR$_A$	0.00	0.20	0.88	0.05	0.98	0.02	0.06	0.83	0.84	0.40
	TLASSO	**0.00**	**0.02**	**0.01**	**0.00**	**0.90**	**0.00**	**0.06**	**0.82**	**0.84**	**1.38**
	SCAD	0.00	0.67	0.56	0.49	0.97	0.31	0.08	0.82	0.86	0.83
	LASSO	0.00	0.67	0.47	0.52	0.97	0.34	0.07	0.86	0.88	0.73

4 Application

The introduction in 2009 of Bitcoin as the first cryptocurrency initiated a new era of digital or virtual currency payment and ignited the Initial Coin Offerings (ICOs) with a series of new cryptocurrencies hitting the global monetary market. Bitcoin and other cryptocurrencies offer an easy-to-use, digital alternative to fiat currencies. Their loosely correlated nature makes cryptocurrencies a potential hedge against risk, in that these digital currencies can be added to diversified portfolios. With the establishment of exchanges and strong price gains, more trading interest arose in 2014. The number of exchanges has grown constantly; some went bankrupt (e.g., Mt. Gox), while at the same time, several exchanges are active in the market.

The market nature of cryptocurrencies, namely, a global phenomenon without country ties or direct linkage to single economies, has triggered considerable research interest in the network structure of the cryptocurrency market; see, e.g., Guo et al. (2019). Chen et al. (2018) consider the connections within the BTC blockchain, and Giudici and Pagnottoni (2019) investigate the connections between the exchanges. Among others, these publications are interested in the dynamic dependencies in the network and leading nodes with an impact on the future movement of the cryptocurrencies. These studies have used daily data. Given the system of high-frequency order submission and execution in modern exchanges, an analysis using higher frequency data can convey more information on the price information flows and market interactions between the exchanges.

We employ TriSNAR$_G$ to analyze the time-dependent network between hourly Bitcoin time series denoted in USD. Under the assumption that investors observe the price evolution on all the exchanges and incorporate the corresponding information into their trading decisions, we address the following questions:

- The inter-exchange network between traders on different exchanges is not directly observable, although respective traders should be able to observe the price evolution on other exchanges. This raises the question of whether there is information flow such that traders on certain exchanges influence those on others on a regular basis, and thus, certain exchanges essentially serve as leading network nodes.

- Is time a factor, or, more specifically, do multi-market interactions last for only one time unit (lag) or are they continuously present over several periods?

- Does the dynamic span of the price of an exchange have different impacts on exchanges in terms of the level of connectivity? For example, does a US exchange have a different effect on traders in Europe than in Asia?

Among the exchanges, not all allow trading against the USD, e.g., Poloniex and Binance. We only consider exchanges offering trading for BTC/USD. As such, the longest continuous period with the largest number of exchanges is from 2018-07-04 until 2019-07-04, covering 26 exchanges. We employ TriSNAR$_G$ to study the dynamic connections within the exchange network by splitting the data into training, validation and testing (out-of-sample) periods. In particular, we estimate TriSNAR$_G$ on the training data for the period 2018-07-04 until 2018-12-31 (6 months) and evaluate the regularization parameters λ_1, λ_2, λ_3 on the validation data for 2019-01-01 until 2019-03-31 (3 months). We evaluate the out-of-sample forecast accuracy on the testing data for 2019-04-01 until 2019-07-04 (3 months). To make an easy and interpretable comparison of the estimated parameters, we adopt a GARCH(1,1) model to scale the demeaned data to unit variance, where the magnitudes of the parameters become comparable between exchanges and over time. After data preparation, an ADF test rejects the H_0 of non-stationarity.

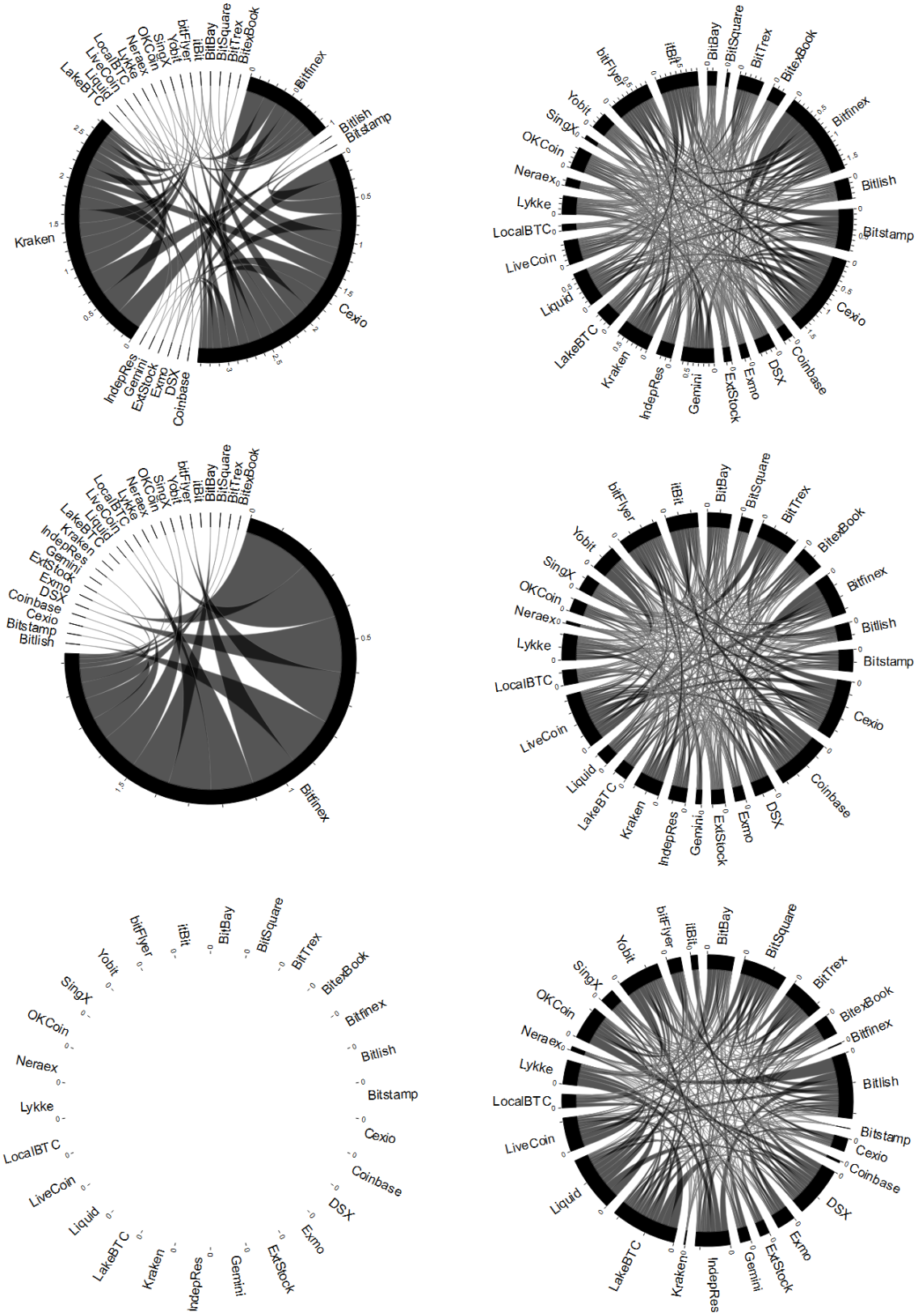
A model allowing for sparsity in the lags, columns, and individual parameters has not yet been used to study the network of BTC exchanges. With the help of TriSNAR, we studied which exchanges are leaders in the formation of the series of returns. For this purpose, we used TriSNAR$_G$ to study whether a time-dependent connection is present and which exchanges influence the dynamics of the return series of the exchanges. We follow the rationale that the trading activity on certain exchanges would have a stronger influence on the price than others. We investigate whether price changes on certain exchanges influence the return series on other exchanges in a time-dependent manner.

We illustrate the estimated network structure with chord diagrams (see Figure 9). A chord diagram displays the direction and magnitude of the influence of each node (Bitcoin exchange) by showing the magnitude by means of the circle and the destination of the signal by the chord. The wider the space in the circle, the larger the magnitude and hence the higher the dynamic impact on the network. A chord diagram does not differentiate between positive and negative influences. The sum of the absolute values of the parameters (magnitude) is displayed on the circle.

Figure 9: Chord diagrams of TriSNAR and TLASSO for the Bitcoin exchanges.

(a) TriSNAR with lags 1–3

(b) TLASSO with lags 1–3



The graphs demonstrate the essential dynamic connectivity in the network spanned by the Bitcoin exchanges. We found that TriSNAR$_G$ detected a meaningful and insightful network structure that was not only sparse but also had a clear interpretation. TriSNAR$_G$ detected three exchanges – Kraken in America, Cex.io in Asia, and Bitfinex in Europe – leading network movement in the next hour with a significant first lag. In the second lag, only Bitfinex (Europe) was influential, and network effects disappeared in the third lag. The three leading exchanges were located in the three major time zones[1] of financial markets (America, Europe, Asia) and have a high transaction volume; compare Figure 10. Kraken is one major exchange located in San Francisco, CA, USA. Bitfinex is based in London, and Cex.io has its headquarters in Hong Kong, which are also traditional financial centers. Considering the fact that traders in different time zones have very distinct working hours, one can infer that each exchange is a leader for the return series movements. It is worth mentioning that the identified leading exchanges are not always the exchanges with the highest trading activity (measured in trading volume of the BTC/USD), see Figure 9a. Bitfinex and Kraken are the top exchanges in their respective continents, while Cex.io features slightly less trading volume in Asia. Note that the trading volume is of similar size on several exchanges, which challenges the visual detection of the underlying network structure. However, TriSNAR detected the leading exchanges from the exchanges with reasonably high trading volume.

In contrast, the TLASSO method from the BigVAR package showed a rather dense structure, where all three lags were active, with vanishing magnitude on the parameters, yet all exchanges were connected to the others. Considering the individual parameters, we observed that TriSNAR assigned a greater magnitude to the selected parameters, whereas many parameters of TLASSO were very close to null. In particular, TriSNAR assigned a combined magnitude of roughly 3.3 to all parameters of Cex.io in the first lag, whereas most individual parameters were greater than 0.1 in magnitude. TLASSO, however, assigned only 1.5 of the combined magnitude, and most parameters were close to null. This observation was even more pronounced for, e.g., Lykke. TriSNAR estimated it as null, whereas TLASSO' considered it active with a combined magnitude of roughly 0.45 and most parameters were very close to null. Additionally, itBit and Bitstamp could be considered strong, with others following close in terms of magnitude and connectivity. In other words, not only a small

[1]With the term 'three major time zones', we refer to the fact that the time differences within America, Europe and Asia are substantially smaller than those between them. Therefore, the business hours within the 'major time zones' are aligned, whereas substantial differences are apparent between them.

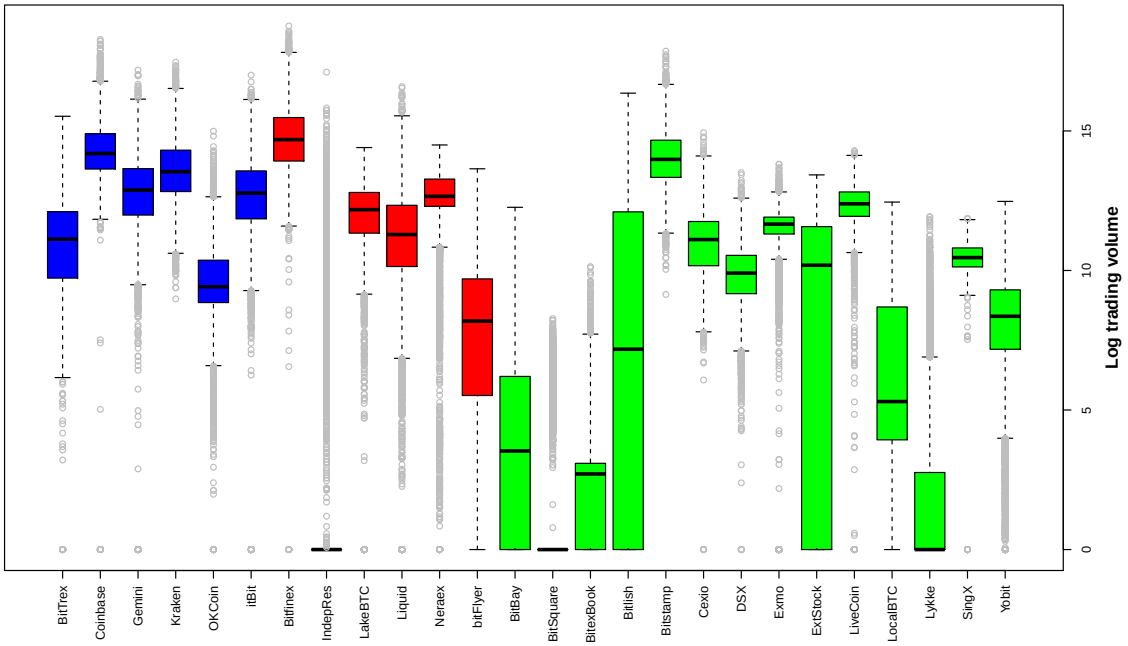
number of exchanges have a network effect. This unclear structure makes the goal of identifying the leading exchanges a challenge. SCAD and LASSO did not discover network effects and hence produced a structure similar to that found by TLASSO. We omit their chord diagrams and only show those of TLASSO as illustrations.

Table 3 summarizes the results for a maximum of three lags, where TriSNAR$_G$ was the sparsest model in terms of the numbers of lags and parameters selected. Although considerably sparser, the MAE.res of TriSNAR was approximately the same as for SCAD or LASSO, and in terms of MAFE.res, all models achieved similar accuracy. This suggests that LASSO, SCAD and TLASSO overparametrized the model since similar results could be achieved with much fewer parameters, as illustrated by TriSNAR$_G$. Similar to the simulation study, we evaluated the best model with AIC. In this respect, TriSNAR performed considerably better due to the sparser model structure. Moreover, we observed the spread of the identified parameters over the parameter matrices. All of the methods allocated the most active parameters to the first lag. They identified considerably fewer parameters in lag 3, though only TriSNAR set lag 3 to null. The outperformance of TriSNAR was balanced across the three continents, though with better accuracy in Asia and Europe. In summary, TriSNAR$_G$ deepened our understanding and improved the interpretability of the global network between the exchanges while giving a comparable forecasting accuracy.

Table 3: Comparison of TriSNAR$_G$, SCAD, LASSO and TLASSO. The best model was chosen based on the observations from the validation period and evaluated in terms of the AIC. MAE.res is measured on the observations in the estimation period and MAFE.res is derived from the out-of-sample period. We also show the number of parameters in each adjacency matrix individually, as well as the MAFE.res for each of the three continents in the dataset.

	#Parameters			MAFE.res				
	Lag 1	Lag 2	Lag 3	America	Asia	Europe	MAE.res	AIC
TriSNAR	76	42	0	0.652	0.523	0.568	0.483	-50,467
SCAD	145	76	43	0.652	0.527	0.573	0.486	-48,945
LASSO	232	149	120	0.651	0.521	0.568	0.481	-48,406
TLASSO	584	460	304	0.649	0.529	0.566	0.491	-45,802

Figure 10: Hourly log trading volume denoted in USD on the 26 exchanges in America, Asia/Oceania and Europe.



5 Conclusion

In this study, we introduced a three-layer Sparse estimator for large-scale Network AutoRegressive models (TriSNAR), which regularized an NAR model for the lags, groups and individual parameters. TriSNAR was developed to investigate the interactions between markets and identify leading network nodes. TriSNAR has the three properties of a good estimator: it is unbiased, has the sparsity property and is a continuous estimator. With an extensive simulation study, we showed that TriSNAR has higher accuracy in terms of uncovering the true pattern of the system. In addition, TriSNAR allows an interpretation of the network. For the three-layer structure of TriSNAR, we developed two algorithms that solve for TriSNAR efficiently. Both the global and local algorithms have a faster runtime per regularization-sequence combination than competitors, such as LASSO. We used TriSNAR to study the network structure by analyzing the market interactions of a set of Bitcoin pricing series from 26 exchanges and by identifying the leading exchanges. Compared to tapered LASSO, SCAD and LASSO, TriSNAR had the best forecasting (MAFE) and modeling (MAE) accuracy, as well as the sparsest solution. TriSNAR identified three exchanges, one each from Europe, Asia, and North America, as important network nodes located

in important financial places: London, Hong Kong, and San Francisco. TriSNAR's properties of being an unbiased and sparse estimator with a continuous optimization function enabled it to outperform the other contestants in an empirical study. The three competing methods were not capable of discovering the structure, which highlights the advantages of TriSNAR in this model setting: highest accuracy, reasonable runtime, and the ability to discover the network's structure.

References

Antoniadis, A. (1997). "Wavelets in Statistics: A Review". *Journal of the Italian Statistical Society* 6.2, pp. 97–130.

Bańbura, M., Giannone, D., and Reichlin, L. (2010). "Large Bayesian Vector Autoregressions". *Journal of Applied Econometrics* 25.1, pp. 71–92.

Basu, S., Li, X., and Michailidis, G. (2019). "Low Rank and Structured Modeling of High-Dimensional Vector Autoregressions". *IEEE Transactions on Signal Processing* 67.5, pp. 1207–1222.

Basu, S. and Michailidis, G. (2015). "Regularized Estimation in Sparse High-Dimensional Time Series Models". *The Annals of Statistics* 43.4, pp. 1535–1567.

Beck, A. and Teboulle, M. (2009). "A Fast Iterative Shrinkage-Thresholding Algorithm for Linear Inverse Problems". *SIAM Journal on Imaging Sciences* 2.1, pp. 183–202.

Bonaldi, P., Hortaçsu, A., and Kastl, J. (2015). *An Empirical Analysis of Funding Costs Spillovers in the EURO-zone with Application to Systemic Risk*. Working Paper 21462. National Bureau of Economic Research.

Chen, H., Choi, P. M. S., and Hong, Y. (2013). "How smooth is price discovery? Evidence from cross-listed stock trading". *Journal of International Money and Finance* 32, pp. 668–699.

Chen, Y., Trimborn, S., and Zhang, J. (2018). "Discover Regional and Size Effects in Global Bitcoin Blockchain via Sparse-Group Network AutoRegressive Modeling". available at SSRN ID 3245031.

Creal, D., Koopman, S. J., and Lucas, A. (2013). "Generalized Autoregressive Score Models with Applications". *Journal of Applied Econometrics* 28.5, pp. 777–795.

Davis, R. A., Zang, P., and Zheng, T. (2016). "Sparse Vector Autoregressive Modeling". *Journal of Computational and Graphical Statistics* 25.4, pp. 1077–1096.

de Paula, Á. (2017). "Econometrics of Network Models". *Advances in Economics and Econometrics: Eleventh World Congress*. Ed. by B. Honoré, A. Pakes, M. Piazzesi, and L. Samuelson. Cambridge University Press, pp. 268–323.

Dees, S., di Mauro, F., Pesaran, M. H., and Smith, L. V. (2007). "Exploring the International Linkages of the EURO Area: A Global VAR Analysis". *Journal of Applied Econometrics* 22.1, pp. 1–38.

Diebold, F. X. and Yılmaz, K. (2014). "On the Network Topology of Variance Decompositions: Measuring the Connectedness of Financial Firms". *Journal of Econo-*

metrics 182.1. Special Issue on Causality, Prediction, and Specification Analysis: Recent Advances and Future Directions, pp. 119–134.

Fan, J. and Li, R. (2001). "Variable Selection via Nonconcave Penalized Likelihood and its Oracle Properties". *Journal of the American Statistical Association* 96.456, pp. 1348–1360.

Gagnon, L. and Karolyi, G. (2013). "International Cross-Listings". *The Evidence and Impact of Financial Globalization*. Ed. by G. Caprio, T. Beck, S. Claessens, and S. L. Schmukler. San Diego, CA, USA: Academic Press, pp. 155–180.

Gagnon, L. and Karolyi, G. A. (2010). "Multi-market trading and arbitrage". *Journal of Financial Economics* 97.1, pp. 53–80.

Ghosh, S., Khare, K., and Michailidis, G. (2019). "High-Dimensional Posterior Consistency in Bayesian Vector Autoregressive Models". *Journal of the American Statistical Association* 114.526, pp. 735–748.

Giudici, P. and Pagnottoni, P. (2019). "Vector Error Correction Models to Measure Connectedness of Bitcoin Exchange Markets". *Applied Stochastic Models in Business and Industry*, pp. 1–15.

Guo, L., Tao, Y., and Härdle, W. K. (2019). *A Dynamic Network Perspective on Cryptocurrencies*. SSRN Paper ID 3185594. Rochester, NY, USA: Social Science Research Network.

Halling, M., Moulton, P. C., and Panayides, M. (2013). "Volume Dynamics and Multimarket Trading". *Journal of Financial and Quantitative Analysis* 48.2, pp. 489–518.

Härdle, W. K., Wang, W., and Yu, L. (2016). "TENET: Tail-Event driven NETwork Risk". *Journal of Econometrics* 192.2. Special Issue on Innovations in Multiple Time Series Analysis, pp. 499–513.

Kline, B. and Tamer, E. (2017). "Econometric Analysis of Models with Social Interactions". available at semanticscholar.org.

Kock, A. B. and Callot, L. (2015). "Oracle Inequalities for High Dimensional Vector Autoregressions". *Journal of Econometrics* 186.2. Special Issue on High Dimensional Problems in Econometrics, pp. 325–344.

Lin, J. and Michailidis, G. (2017). "Regularized Estimation and Testing for High-Dimensional Multi-Block Vector-Autoregressive Models". *Journal of Machine Learning Research* 18.1, pp. 4188–4236.

Medeiros, M. C. and Mendes, E. F. (2016). "L1-Regularization of High-Dimensional Time-Series Models with Non-Gaussian and Heteroskedastic Errors". *Journal of Econometrics* 191.1, pp. 255–271.

Nicholson, W., Matteson, D., and Bien, J. (2019). *BigVAR: Dimension Reduction Methods for Multivariate Time Series*. R package version 1.0.4.

Nicholson, W., Matteson, D., and Bien, J. (2017). "VARX-L: Structured Regularization for Large Vector Autoregressions with Exogenous Variables". *International Journal of Forecasting* 33.3, pp. 627–651.

Nicholson, W. B., Wilms, I., Bien, J., and Matteson, D. S. (2014). "High Dimensional Forecasting via Interpretable Vector Autoregression". arXiv:1412.5250 [stat].

Pesaran, M. H., Schuermann, T., and Weiner, S. M. (2004). "Modeling Regional Interdependencies Using a Global Error-Correcting Macroeconometric Model". *Journal of Business & Economic Statistics* 22.2, pp. 129–162.

Rapach, D. E., Strauss, J. K., and Zhou, G. (2013). "International Stock Return Predictability: What Is the Role of the United States?" *The Journal of Finance* 68.4, pp. 1633–1662.

Simon, N., Friedman, J., Hastie, T., and Tibshirani, R. (2013). "A Sparse-Group Lasso". *Journal of Computational and Graphical Statistics* 22.2, pp. 231–245.

Skripnikov, A. and Michailidis, G. (2019). "Regularized joint estimation of related vector autoregressive models". *Computational Statistics & Data Analysis* 139, pp. 164–177.

Song, S. and Bickel, P. J. (2011). "Large Vector Autoregressions". arXiv:1106.3915 [stat.ML].

Tibshirani, R. (1996). "Regression Shrinkage and Selection via the Lasso". *Journal of the Royal Statistical Society. Series B (Methodological)* 58.1, pp. 267–288.

Tsay, R. S. (2016). "Some Methods for Analyzing Big Dependent Data". *Journal of Business & Economic Statistics* 34.4, pp. 673–688.

Wang, H., Li, G., and Tsai, C.-L. (2007). "Regression Coefficient and Autoregressive Order Shrinkage and Selection via the Lasso". *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 69.1, pp. 63–78.

Zhu, X., Pan, R., Li, G., Liu, Y., and Wang, H. (2017). "Network Vector Autoregression". *The Annals of Statistics* 45.3, pp. 1096–1123.

Zou, H. (2006). "The Adaptive Lasso and Its Oracle Properties". *Journal of the American Statistical Association* 101.476, pp. 1418–1429.

I Proof Appendix

In this section we prove the consistency and oracle property of the estimator. We prove the theorems under the assumptions made for model (1). The proofs follow Fan and Li (2001), Song and Bickel (2011) and Wang et al. (2007).

In what follows, assume without loss of generality that the true parameter matrix $\boldsymbol{A}_0$ contains p lag matrices. Each matrix has a submatrix of dimension $d_1 \times d_1$ whose elements are different from 0 in the upper left corner. The remaining elements are equal to 0. Let $\boldsymbol{A}_{k;d_1 d_1}$ indicate the respective submatrix for all k and $\boldsymbol{A}_{k;-d_1-d_1}$ the remaining elements of the respective matrix. We denote by $\boldsymbol{A}_{\cdot;d_1 d_1}$ the combined parameter matrices $\boldsymbol{A}_{k;d_1 d_1}$ over all k, and let $\boldsymbol{A}_{\cdot;-d_1-d_1}$ denote the respective combined parameter matrices $\boldsymbol{A}_{k;-d_1-d_1}$ over all k.

We further define $O_M(\cdot)$ as big O notation for elementwise convergence within a matrix and $O_V(\cdot)$ as big O notation for elementwise convergence within a vector. Likewise we define $o_M(\cdot)$ and $o_V(\cdot)$ as small o notation for matrices and vectors. Let $vec(\cdot)$ denote the vectorizing operator to convert a matrix to a vector. Further, we denote the Fisher information matrix by $I(\cdot)$.

I.1 Proof of Theorem 1

Denote by $CL(\cdot)$ the constrained likelihood and by $L(\cdot)$ the likelihood. Further define $b = T^{-1/2} + b_T$ and $\boldsymbol{U}$ coordinates around $\boldsymbol{A}_0$. For a large constant Q, it holds that $\{\boldsymbol{A}_0 + b\boldsymbol{U} : ||\boldsymbol{U}||_F \leq Q\}$ is the ball around $\boldsymbol{A}_0$ and we intend to show that a local maximum with maximizer $\widehat{\boldsymbol{A}}$ lies in the ball. So we intend to show that on the surface of the ball, $||\boldsymbol{U}||_F = Q$, for any $\epsilon > 0$,

$$P\{\sup_{||\boldsymbol{U}||_F = Q} CL(\boldsymbol{A}_0 + b\boldsymbol{U}) < CL(\boldsymbol{A}_0)\} \geq 1 - \epsilon. \tag{6}$$

The difference between the two penalized likelihoods $CL(\boldsymbol{A}_0 + b\boldsymbol{U})$ and $CL(\boldsymbol{A}_0)$ can be bounded from above by the likelihood and the penalization on $\widehat{\boldsymbol{A}}$ only for the pd_1^2 parameters different from 0. So we use the property $p_{\lambda_1,\lambda_2,\lambda_3}(0) = 0$. In case no parameter in $\boldsymbol{A}_0$ is 0, it will be equal, otherwise larger:

$$CL(\boldsymbol{A}_0 + b\boldsymbol{U}) - CL(\boldsymbol{A}_0) \leq L(\boldsymbol{A}_0 + b\boldsymbol{U}) - L(\boldsymbol{A}_0) \tag{7}$$
$$- T\sum_{k=1}^{p}\sum_{i=1}^{d_1}\sum_{j=1}^{d_1}\{p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0} + b\boldsymbol{U}_{k;ij;0}) - p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})\}.$$

Approximating by a Taylor expansion for $\boldsymbol{A}_0 + b\boldsymbol{U}$ around $\boldsymbol{A}_0$ gives

$$L(\boldsymbol{A}_0 + b\boldsymbol{U}) = L(\boldsymbol{A}_0) + (\boldsymbol{A}_0 + b\boldsymbol{U} - \boldsymbol{A}_0)L^{'}(\boldsymbol{A}_0) \tag{8}$$
$$+ \frac{1}{2}L^{''}(\boldsymbol{A}_0)(\boldsymbol{A}_0 + b\boldsymbol{U} - \boldsymbol{A}_0)^{\top}(\boldsymbol{A}_0 + b\boldsymbol{U} - \boldsymbol{A}_0)$$
$$+ o\{\frac{1}{2}L^{''}(\boldsymbol{A}_0)(\boldsymbol{A}_0 + b\boldsymbol{U} - \boldsymbol{A}_0)^{\top}(\boldsymbol{A}_0 + b\boldsymbol{U} - \boldsymbol{A}_0)\}$$

$$T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0} + b\boldsymbol{U}_{k;ij}) = T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0}) \tag{9}$$
$$+ T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1}(\boldsymbol{A}_{k;ij;0} + b\boldsymbol{U}_{k;ij} - \boldsymbol{A}_{k;ij;0})$$
$$p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})^{'} sgn(\boldsymbol{A}_{k;ij;0})$$
$$+ T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1}(\boldsymbol{A}_{k;ij;0} + b\boldsymbol{U}_{k;ij} - \boldsymbol{A}_{k;ij;0})^2$$
$$p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})^{''}$$
$$+ T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} o(\boldsymbol{A}_{k;ij;0} + b\boldsymbol{U}_{k;ij} - \boldsymbol{A}_{k;ij;0})^2$$
$$p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})^{''}$$

Hence

$$L(\boldsymbol{A}_0 + b\boldsymbol{U}) - L(\boldsymbol{A}_0) = bL^{'}(\boldsymbol{A}_0)\mathrm{vec}(\boldsymbol{U}) + \frac{1}{2}b^2\mathrm{vec}(\boldsymbol{U})^{\top}L^{''}(\boldsymbol{A}_0)\mathrm{vec}(\boldsymbol{U}) \tag{10}$$
$$+ \frac{1}{2}b^2\mathrm{vec}(\boldsymbol{U})^{\top}L^{''}(\boldsymbol{A}_0)\mathrm{vec}(\boldsymbol{U})o\{1\}$$

and

$$T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0}+b\boldsymbol{U}_{k;ij}) - T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0}) \tag{11}$$

$$= T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} (b\boldsymbol{U}_{k;ij}) p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})' sgn(\boldsymbol{A}_{k;ij;0}) \tag{12}$$

$$+ T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} (b\boldsymbol{U}_{k;ij})^2 p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})''$$

$$+ T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} b^2\boldsymbol{U}_{k;ij}^2 o(1)^2 p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})''$$

Recall that $L''(\boldsymbol{A}_0) = -TI(\boldsymbol{A}_0)$.

Hence,

$$\begin{aligned} CL(\boldsymbol{A}_0+b\boldsymbol{U}) - CL(\boldsymbol{A}_0) \leq & bL'(\boldsymbol{A}_0)^\top \mathrm{vec}(\boldsymbol{U}) \\ & - \frac{1}{2}Tb^2\mathrm{vec}(\boldsymbol{U})^\top I(\boldsymbol{A}_0)\mathrm{vec}(\boldsymbol{U})(1+o(1)) \\ & - T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} bp_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})' sgn(\boldsymbol{A}_{k;ij;0})\boldsymbol{U}_{k;ij} \\ & - T\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} (b\boldsymbol{U}_{k;ij})^2 p_{\lambda_1,\lambda_2,\lambda_3}(\boldsymbol{A}_{k;ij;0})''(1+o(1)) \end{aligned} \tag{13}$$

Now, $T^{-1/2}L(\boldsymbol{A}_0)' = \mathcal{O}_V(1)$. It follows that the first term on the right-hand side is of order $\mathcal{O}_V(T^{1/2}b)$. If $\boldsymbol{A}_0$ is large enough or the penalty on it small, then by writing out the inner term it goes to 1. The same holds for $\mathcal{O}(Tb^2)$, hence $\mathcal{O}(T^{1/2}b) = \mathcal{O}(Tb^2)$. For a sufficiently large Q, the second term dominates the first term uniformly in $||\boldsymbol{U}||_F = Q$. The third and fourth term are bounded by

$$Tb\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} b_T\boldsymbol{U}_{k;ij} + Tb^2\sum_{k=1}^{p}\sum_{i,j=1}^{d_1} \boldsymbol{U}_{k;ij}^2 \max\left(\frac{\partial^2 p_{\lambda_1,\lambda_2,\lambda_3}}{\partial A_{k;ij}^2} : A_{k;ij}\neq 0\right)(1+o(1)), \tag{14}$$

and therefore are $\mathcal{O}(Tb)$ and $\mathcal{O}(Tb^2)$ and hence dominated by the second term too in the case of a large Q. Hence (6) holds. This implies that there exists a local maximizer $\widehat{\boldsymbol{A}}$ for which $||\widehat{\boldsymbol{A}} - \boldsymbol{A}_0||_F = \mathcal{O}(b)$. This completes the proof of the theorem.

I.2 Proof of Lemma 1

We carry out the proof by showing that all parameters in $\boldsymbol{A}_{k;-d_1-d_1}$ for all k cannot be different from 0 since this would be a contradiction. One has

$$\frac{\partial CL(\widehat{\boldsymbol{A}}_k)}{\partial A_{k;ij}} = \frac{\partial L(\widehat{\boldsymbol{A}}_k)}{\partial A_{k;ij}} - Tp'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(\widehat{A}_{k;ij})sgn(\widehat{A}_{k;ij}), \tag{15}$$

hence for a consistent selection of $\boldsymbol{A}_{k;-d_1-d_1}$ all parameters have to be 0. Otherwise the first derivative of the constrained likelihood would not equal the unconstrained one, which is 0.

It is sufficient to show that $\frac{\partial CL(\boldsymbol{A}_k)}{\partial A_{k;ij}} \neq 0$ if and only if $A_{k;ij} \neq 0$. Hence we will show that with probability tending to 1 for $T \to \infty$, for any $\boldsymbol{A}_{k;d_1d_1}$ satisfying $\boldsymbol{A}_{k;d_1d_1} - \boldsymbol{A}_{k;d_1d_1;0} = O_M(T^{-1/2})$ and for some small $\epsilon_T = QT^{-1/2}$ and $i, j = 1, \ldots, d_1$,

$$\frac{\partial CL(\boldsymbol{A}_k)}{\partial A_{k;ij}} < 0 \quad \text{for } 0 < A_{k;ij} < \epsilon_T \tag{16}$$

$$> 0 \quad \text{for } -\epsilon_T < A_{k;ij} < 0 \tag{17}$$

By Taylor's expansion,

$$\frac{\partial CL(\widehat{\boldsymbol{A}}_k)}{\partial A_{k;ij}} = \frac{\partial L(\widehat{\boldsymbol{A}}_k)}{\partial A_{k;ij}} - Tp'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(\widehat{A}_{k;ij})sgn(\widehat{A}_{k;ij}) \tag{18}$$

$$= \frac{\partial L(\boldsymbol{A}_k^0)}{\partial A_{k;ij}} + \sum_{l_1=1}^{d_2}\sum_{l_2=1}^{d_2} \frac{\partial^2 L(\boldsymbol{A}_k^0)}{\partial A_{k;ij}\partial A_{k;l_1l_2}}(\widehat{A}_{k;l_1l_2} - A^0_{k;l_1l_2}) \tag{19}$$

$$+ \sum_{l_1=1}^{d_2}\sum_{l_2=1}^{d_2}\sum_{l_3=1}^{d_2}\sum_{l_4=1}^{d_2} \frac{\partial^3 L(\boldsymbol{A}_k^*)}{\partial A_{k;ij}\partial A_{k;l_1l_2}\partial A_{k;l_3l_4}} \times (\widehat{A}_{k;l_1l_2} - A^0_{k;l_1l_2})(\widehat{A}_{k;l_3l_4} - A^0_{k;l_3l_4})$$

$$- Tp'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(\widehat{A}_{k;ij})sgn(\widehat{A}_{k;ij})$$

with $\boldsymbol{A}_k^*$ lying between $\widehat{\boldsymbol{A}}_k$ and $\boldsymbol{A}_k^0$.

Recall that

$$T^{-1}\frac{\partial L(\boldsymbol{A}_k^0)}{\partial A_{k;ij}} = O(T^{-1/2})$$
$$T^{-1}\frac{\partial^2 L(\boldsymbol{A}_k^0)}{\partial A_{k;ij}\partial A_{k;l_1l_2}} = \mathrm{E}\left(\frac{\partial^2 L(\boldsymbol{A}_k^0)}{\partial A_{k;ij}\partial A_{k;l_1l_2}}\right) + o(1)$$

The first term is therefore of order $O(T^{1/2})$. The second term is also of order $O(T^{1/2})$ because it consists of the Fisher information matrix and $o(1)$, where the latter is negligible because o goes to 0 faster than O. The third term is obviously faster at 0 due to the squared O_M, meaning it is bounded by $O_P(T^{-1/2})^2$, hence it goes faster to 0 than the first and second term. It follows that

$$\frac{\partial CL(\boldsymbol{A}_k)}{\partial A_{k;ij}} = -Tp'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A_{k;ij})sgn(A_{k;ij}) + O(T^{1/2}) \tag{20}$$

The first term dominates, because $\sqrt{T}b_T \to \infty$. Hence the sign of $A_{k;ij}$ determines the sign of $\frac{\partial CL(\boldsymbol{A}_k)}{\partial A_{k;ij}}$. Hence the inequalities (16) and (17) hold, which implies that $\frac{\partial CL(\boldsymbol{A}_k)}{\partial A_{k;ij}}$ can only be 0 if and only if $A_{k;ij} = 0$. This completes the proof.

I.3 Proof of Theorem 2

From Lemma 1 there follows 1. It can be easily shown that there exists an $\widehat{\boldsymbol{A}}_{\cdot;d_1d_1}$ in Theorem 1 that is a root-T consistent local maximizer of $CL((\boldsymbol{A}_{\cdot;d_1d_1}, \boldsymbol{0}))$ that satisfies the likelihood equations

$$\left.\frac{\partial CL(\boldsymbol{A})}{\partial A_{k;ij}}\right|_{\boldsymbol{A}=[\boldsymbol{A}_{\cdot;d_1d_1},\boldsymbol{A}_{\cdot;-d_1-d_1}]} = 0 \qquad \text{for } i = 1,\cdots,d_1; j = 1,\cdots,d_1 \tag{21}$$

Recall that $\boldsymbol{A}_{\cdot;d_1d_1}$ is a consistent estimator,

$$\frac{\partial L(\boldsymbol{A})}{\partial A_{k;ij}}\Big|_{\boldsymbol{A}=[\boldsymbol{A}_{\cdot;d_1d_1},\boldsymbol{A}_{\cdot;-d_1-d_1}]} - Tp'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A_{k;ij})sgn(A_{k;ij}) \tag{22}$$

$$= \frac{\partial L(\boldsymbol{A}^0)}{\partial A_{k;ij}} + \sum_{l_1=1}^{p}\sum_{l_2=1}^{d_1}\sum_{l_3=1}^{d_1}\left(\frac{\partial^2 L(\boldsymbol{A}^0)}{\partial A_{k;ij}\partial A_{l_1;l_2l_3}} + o_P(1)\right)(A_{k;ij} - A^0_{k;ij}) \tag{23}$$

$$- T\Big(p'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{k;ij})sgn(A^0_{k;ij})$$

$$+(p''_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{k;ij}) + o_P(1))(A_{k;ij} - A^0_{k;ij})\Big).$$

Setting the first derivative equal to 0 and rearranging terms gives

$$\begin{aligned}(A_{k;ij} - A^0_{k;ij}) &= -\frac{\frac{\partial L(\boldsymbol{A}^0)}{\partial A_{k;ij}} - Tp'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{k;ij})sgn(A^0_{k;ij})}{H - TK} \\ &= -\frac{\frac{1}{T}\frac{\partial L(\boldsymbol{A}^0)}{\partial A_{k;ij}} - p'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{k;ij})sgn(A^0_{k;ij})}{\frac{1}{T}H - K},\end{aligned}$$

whereas $H = \sum_{l_1=1}^{p}\sum_{l_2=1}^{d_1}\sum_{l_3=1}^{d_1}\left(\frac{\partial^2 L(\boldsymbol{A}^0)}{\partial A_{k;ij}\partial A_{l_1;l_2l_3}} + o_P(1)\right)$ and $K = p''_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{k;ij}) + o_P(1)$.

The nominator converges in distribution by the Central Limit Theorem to

$$\frac{1}{T}\frac{\partial L(\boldsymbol{A}^0)}{\partial A_{k;ij}} - p'_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{k;ij})sgn(A^0_{k;ij}) \xrightarrow{d} N(0, \frac{I(\boldsymbol{A}^0_{;d_1d_1})_{k;ij}}{T}) - G_{k;ij} \tag{24}$$

By Slutsky's Theorem, the denominator goes to

$$-\frac{1}{T}\sum_{l_1=1}^{p}\sum_{l_2=1}^{d_1}\sum_{l_3=1}^{d_1}\left(\frac{\partial^2 L(\boldsymbol{A}^0)}{\partial A_{k;ij}\partial A_{l_1;l_2l_3}} + o_P(1)\right)$$

$$+ (p''_{\lambda_{1,T},\lambda_{2,T},\lambda_{3,T}}(A^0_{k;ij}) + o_P(1)) \to I(\boldsymbol{A}^0_{;d_1d_1})_{k;ij} + F_{k;ij} \tag{25}$$

Combining the two results and writing this in matrix form gives

$$(\boldsymbol{A}_{\cdot;d_1d_1} - \boldsymbol{A}^0_{\cdot;d_1d_1}) \xrightarrow{d} N(0, \frac{I(\boldsymbol{A}^0_{\cdot;d_1d_1}}{T}(I(\boldsymbol{A}^0_{\cdot;d_1d_1}) + F)^{-2}) - G(I(\boldsymbol{A}^0_{\cdot;d_1d_1}) + F)^{-1}$$

$$(\boldsymbol{A}_{\cdot;d_1d_1} - \boldsymbol{A}^0_{\cdot;d_1d_1}) + G(I(\boldsymbol{A}^0_{\cdot;d_1d_1}) + F)^{-1} \xrightarrow{d} N(0, \frac{I(\boldsymbol{A}^0_{\cdot;d_1d_1})}{T}(I(\boldsymbol{A}^0_{\cdot;d_1d_1}) + F)^{-2})$$

$$\sqrt{T}((\boldsymbol{A}_{\cdot;d_1d_1} - \boldsymbol{A}^0_{\cdot;d_1d_1})(I(\boldsymbol{A}^0_{\cdot;d_1d_1}) + F) + G) \xrightarrow{d} N(0, I(\boldsymbol{A}^0_{\cdot;d_1d_1}))$$

Hence by applying Slutsky's Theorem and the Central Limit Theorem, we find

$$\sqrt{T}((\boldsymbol{A}_{\cdot;d_1d_1} - \boldsymbol{A}^0_{\cdot;d_1d_1})(I(\boldsymbol{A}^0_{\cdot;d_1d_1}) + F) + G) \xrightarrow{d} N(0, I(\boldsymbol{A}^0_{\cdot;d_1d_1})) \tag{26}$$

This completes the proof.

II Tables Appendix

Table 4: Simulation $d = 50$ with 3 lags, $T = 200$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**2.70**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**0.33**
	TLASSO	0.00	0.00	0.00	0.93	0.00	0.28	0.04	0.85	0.85	1.57
	SCAD	0.00	0.07	0.00	0.11	0.02	0.01	0.03	0.85	0.85	1.64
	LASSO	0.00	0.46	0.00	0.66	0.01	0.20	0.05	0.86	0.87	1.36
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**3.06**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**0.31**
	TLASSO	0.00	0.50	0.00	0.98	0.00	0.84	0.05	0.84	0.85	1.63
	SCAD	0.00	0.67	0.00	0.98	0.00	0.56	0.04	0.83	0.85	1.67
	LASSO	0.00	0.67	0.00	0.99	0.00	0.71	0.04	0.84	0.85	1.40
M1	TriSNAR$_G$	0.00	0.00	0.80	0.00	0.66	0.00	0.05	0.84	0.84	3.39
	TriSNAR$_A$	0.00	0.00	0.79	0.00	0.66	0.00	0.05	0.84	0.84	0.50
	TLASSO	**0.00**	**0.00**	**0.00**	**0.84**	**0.11**	**0.28**	**0.05**	**0.83**	**0.84**	**1.98**
	SCAD	0.00	0.54	0.28	0.48	0.62	0.10	0.05	0.83	0.84	1.73
	LASSO	0.00	0.63	0.13	0.62	0.57	0.17	0.06	0.86	0.86	1.45
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.80**	**0.00**	**0.66**	**0.00**	**0.05**	**0.82**	**0.83**	**3.22**
	TriSNAR$_A$	**0.00**	**0.00**	**0.80**	**0.00**	**0.66**	**0.00**	**0.05**	**0.82**	**0.83**	**0.42**
	TLASSO	0.00	0.50	0.00	0.92	0.21	0.66	0.06	0.82	0.83	1.89
	SCAD	0.00	0.67	0.00	0.93	0.32	0.56	0.07	0.80	0.83	1.51
	LASSO	0.00	0.67	0.00	0.93	0.30	0.51	0.06	0.82	0.83	1.26
M1/M3	TriSNAR$_G$	0.30	0.10	0.65	0.24	0.78	0.13	0.07	0.87	0.87	2.36
	TriSNAR$_A$	**0.30**	**0.10**	**0.66**	**0.22**	**0.80**	**0.11**	**0.07**	**0.86**	**0.87**	**0.31**
	TLASSO	0.50	0.00	0.52	0.65	0.78	0.19	0.07	0.88	0.89	1.64
	SCAD	0.00	0.33	0.09	0.81	0.74	0.43	0.08	0.85	0.88	1.45
	LASSO	0.00	0.33	0.04	0.85	0.68	0.50	0.07	0.87	0.89	1.20
NS1	TriSNAR$_G$	0.00	0.02	0.96	0.01	0.98	0.01	0.06	0.84	0.84	3.36
	TriSNAR$_A$	0.00	0.36	0.67	0.16	0.97	0.09	0.06	0.83	0.84	0.47
	TLASSO	**0.00**	**0.29**	**0.00**	**0.05**	**0.85**	**0.01**	**0.05**	**0.81**	**0.82**	**1.74**
	SCAD	0.00	0.66	0.50	0.34	0.97	0.19	0.07	0.82	0.84	1.49
	LASSO	0.00	0.67	0.12	0.54	0.93	0.39	0.07	0.83	0.84	1.25

Table 5: Simulation $d = 50$ with 3 lags, $T = 500$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**0.64**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**0.22**
	TLASSO	0.00	0.10	0.00	0.98	0.00	0.68	0.03	0.85	0.85	2.75
	SCAD	0.00	0.22	0.00	0.30	0.00	0.05	0.02	0.85	0.85	1.36
	LASSO	0.00	0.67	0.00	0.95	0.00	0.34	0.04	0.85	0.85	1.21
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**0.71**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**0.28**
	TLASSO	0.00	0.50	0.00	0.98	0.00	0.84	0.04	0.84	0.84	2.82
	SCAD	0.00	0.66	0.00	0.97	0.00	0.54	0.02	0.83	0.84	1.31
	LASSO	0.00	0.66	0.00	0.98	0.00	0.65	0.03	0.84	0.84	1.17
M1	TriSNAR$_G$	0.00	0.32	0.48	0.36	0.45	0.09	0.05	0.83	0.84	1.55
	TriSNAR$_A$	0.00	0.55	0.15	0.61	0.25	0.12	0.04	0.83	0.83	0.32
	TLASSO	**0.00**	**0.08**	**0.00**	**0.90**	**0.00**	**0.51**	**0.04**	**0.82**	**0.83**	**3.90**
	SCAD	0.00	0.51	0.20	0.51	0.41	0.20	0.05	0.83	0.83	2.04
	LASSO	0.00	0.67	0.00	0.89	0.10	0.35	0.05	0.83	0.84	1.86
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.21**	**0.03**	**0.81**	**0.81**	**1.36**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.21**	**0.03**	**0.81**	**0.81**	**0.34**
	TLASSO	0.00	0.50	0.00	0.92	0.00	0.55	0.04	0.81	0.82	3.95
	SCAD	0.00	0.67	0.00	0.94	0.04	0.54	0.05	0.80	0.82	1.75
	LASSO	0.00	0.67	0.00	0.91	0.04	0.38	0.05	0.82	0.82	1.60
M1/M3	TriSNAR$_G$	0.00	0.17	0.42	0.18	0.46	0.14	0.06	0.85	0.85	0.53
	TriSNAR$_A$	**0.00**	**0.13**	**0.32**	**0.15**	**0.47**	**0.12**	**0.06**	**0.85**	**0.85**	**0.24**
	TLASSO	0.00	0.33	0.01	0.89	0.27	0.74	0.07	0.85	0.85	3.43
	SCAD	0.00	0.31	0.06	0.42	0.61	0.08	0.07	0.85	0.86	1.61
	LASSO	0.00	0.33	0.01	0.70	0.45	0.22	0.07	0.87	0.88	1.45
NS1	TriSNAR$_G$	0.00	0.67	0.00	0.55	0.88	0.38	0.06	0.81	0.82	1.52
	TriSNAR$_A$	0.00	0.67	0.01	0.35	0.93	0.17	0.06	0.81	0.82	0.30
	TLASSO	**0.00**	**0.16**	**0.00**	**0.01**	**0.86**	**0.00**	**0.04**	**0.81**	**0.81**	**3.30**
	SCAD	0.00	0.67	0.01	0.53	0.92	0.33	0.06	0.80	0.82	1.77
	LASSO	0.00	0.67	0.00	0.55	0.91	0.37	0.06	0.82	0.83	1.61

Table 6: Simulation $d = 50$ with 3 lags, $T = 1000$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**0.61**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**0.42**
	TLASSO	0.00	0.00	0.00	0.93	0.00	0.24	0.03	0.85	0.85	4.72
	SCAD	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**2.43**
	LASSO	0.00	0.61	0.00	0.84	0.00	0.53	0.04	0.85	0.85	2.26
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.01**	**0.84**	**0.84**	**0.61**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.01**	**0.84**	**0.84**	**0.41**
	TLASSO	0.00	0.50	0.00	0.98	0.00	0.83	0.03	0.84	0.84	5.15
	SCAD	0.00	0.66	0.00	0.97	0.00	0.54	0.02	0.84	0.84	2.18
	LASSO	0.00	0.67	0.00	0.98	0.00	0.62	0.03	0.84	0.84	2.00
M1	TriSNAR$_G$	**0.00**	**0.04**	**0.00**	**0.01**	**0.00**	**0.13**	**0.03**	**0.82**	**0.82**	**1.76**
	TriSNAR$_A$	0.00	0.04	0.00	0.01	0.00	0.13	0.03	0.82	0.82	0.60
	TLASSO	0.00	0.12	0.00	0.88	0.00	0.41	0.04	0.82	0.83	7.39
	SCAD	0.00	0.67	0.00	0.78	0.04	0.32	0.04	0.82	0.83	4.23
	LASSO	0.00	0.67	0.00	0.96	0.00	0.67	0.05	0.82	0.83	4.06
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.02**	**0.10**	**0.02**	**0.14**	**0.03**	**0.81**	**0.81**	**1.68**
	TriSNAR$_A$	**0.00**	**0.00**	**0.02**	**0.10**	**0.02**	**0.14**	**0.03**	**0.81**	**0.81**	**0.59**
	TLASSO	0.00	0.50	0.00	0.92	0.00	0.55	0.04	0.81	0.81	8.43
	SCAD	0.00	0.67	0.00	0.94	0.00	0.58	0.05	0.80	0.81	3.48
	LASSO	0.00	0.67	0.00	0.90	0.00	0.35	0.04	0.81	0.81	3.31
M1/M3	TriSNAR$_G$	0.00	0.33	0.00	0.83	0.00	0.40	0.06	0.84	0.85	0.85
	TriSNAR$_A$	0.00	0.32	0.01	0.82	0.10	0.25	0.06	0.84	0.85	0.63
	TLASSO	0.00	0.33	0.00	0.91	0.03	0.78	0.06	0.84	0.85	7.30
	SCAD	**0.00**	**0.26**	**0.02**	**0.45**	**0.33**	**0.18**	**0.07**	**0.85**	**0.85**	**3.09**
	LASSO	0.00	0.33	0.00	0.91	0.06	0.43	0.06	0.85	0.86	2.87
NS1	TriSNAR$_G$	0.00	0.54	0.00	0.06	0.89	0.02	0.04	0.81	0.81	1.27
	TriSNAR$_A$	0.00	0.62	0.00	0.15	0.93	0.06	0.05	0.81	0.81	0.54
	TLASSO	**0.00**	**0.50**	**0.00**	**0.20**	**0.77**	**0.02**	**0.04**	**0.81**	**0.81**	**6.08**
	SCAD	0.00	0.67	0.00	0.46	0.92	0.25	0.05	0.81	0.81	3.48
	LASSO	0.00	0.67	0.00	0.65	0.88	0.48	0.05	0.81	0.81	3.31

Table 7: Simulation $d = 10$ with 3 lags, $T = 100$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.00**
	TLASSO	0.00	0.15	0.00	0.72	0.00	0.40	0.09	0.85	0.85	0.02
	SCAD	0.00	0.32	0.00	0.43	0.07	0.14	0.08	0.85	0.86	0.04
	LASSO	0.00	0.55	0.00	0.71	0.06	0.29	0.11	0.87	0.87	0.03
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.84**	**0.84**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.84**	**0.84**	**0.01**
	TLASSO	0.00	0.51	0.00	0.91	0.00	0.68	0.10	0.84	0.85	0.02
	SCAD	0.00	0.65	0.00	0.89	0.02	0.56	0.09	0.83	0.85	0.04
	LASSO	0.00	0.66	0.00	0.93	0.00	0.68	0.11	0.84	0.85	0.03
M1	TriSNAR$_G$	0.00	0.01	0.79	0.01	0.65	0.00	0.11	0.84	0.84	0.01
	TriSNAR$_A$	0.00	0.02	0.78	0.02	0.65	0.01	0.11	0.84	0.84	0.01
	TLASSO	**0.00**	**0.27**	**0.00**	**0.55**	**0.07**	**0.36**	**0.11**	**0.83**	**0.83**	**0.03**
	SCAD	0.00	0.37	0.51	0.32	0.62	0.12	0.12	0.84	0.85	0.05
	LASSO	0.00	0.62	0.20	0.61	0.47	0.38	0.14	0.85	0.86	0.04
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.80**	**0.00**	**0.65**	**0.00**	**0.11**	**0.83**	**0.83**	**0.01**
	TriSNAR$_A$	0.00	0.01	0.78	0.01	0.65	0.01	0.11	0.82	0.83	0.01
	TLASSO	0.00	0.53	0.06	0.72	0.22	0.57	0.13	0.82	0.83	0.03
	SCAD	0.00	0.63	0.14	0.58	0.43	0.34	0.13	0.81	0.84	0.04
	LASSO	0.00	0.66	0.07	0.69	0.34	0.45	0.13	0.82	0.84	0.04
M1/M3	TriSNAR$_G$	0.26	0.15	0.57	0.24	0.77	0.19	0.16	0.87	0.88	0.01
	TriSNAR$_A$	0.25	0.15	0.61	0.21	0.80	0.17	0.16	0.87	0.89	0.01
	TLASSO	0.42	0.13	0.54	0.37	0.70	0.27	0.15	0.87	0.88	0.02
	SCAD	0.00	0.33	0.28	0.48	0.75	0.39	0.18	0.85	0.89	0.04
	LASSO	**0.00**	**0.33**	**0.18**	**0.53**	**0.66**	**0.43**	**0.16**	**0.87**	**0.89**	**0.03**
NS1	TriSNAR$_G$	0.00	0.03	0.86	0.01	0.88	0.01	0.13	0.85	0.85	0.01
	TriSNAR$_A$	0.00	0.03	0.85	0.01	0.89	0.01	0.13	0.85	0.85	0.01
	TLASSO	**0.00**	**0.39**	**0.01**	**0.13**	**0.50**	**0.04**	**0.12**	**0.83**	**0.84**	**0.03**
	SCAD	0.00	0.54	0.54	0.34	0.87	0.19	0.15	0.83	0.86	0.05
	LASSO	0.00	0.65	0.35	0.46	0.81	0.30	0.16	0.85	0.87	0.04

Table 8: Simulation $d = 10$ with 3 lags, $T = 200$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.00**
	TLASSO	0.00	0.08	0.00	0.85	0.00	0.46	0.08	0.85	0.85	0.04
	SCAD	0.00	0.30	0.00	0.40	0.00	0.19	0.06	0.85	0.85	0.04
	LASSO	0.00	0.66	0.00	0.92	0.00	0.61	0.09	0.85	0.85	0.03
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.01**	**0.00**	**0.00**	**0.04**	**0.84**	**0.84**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.01**	**0.00**	**0.00**	**0.04**	**0.84**	**0.84**	**0.01**
	TLASSO	0.00	0.50	0.00	0.90	0.00	0.65	0.08	0.84	0.84	0.04
	SCAD	0.00	0.65	0.00	0.90	0.00	0.56	0.07	0.83	0.84	0.03
	LASSO	0.00	0.66	0.00	0.92	0.00	0.64	0.09	0.84	0.84	0.03
M1	TriSNAR$_G$	0.00	0.30	0.55	0.22	0.46	0.15	0.10	0.83	0.83	0.01
	TriSNAR$_A$	0.00	0.32	0.46	0.24	0.43	0.15	0.11	0.83	0.83	0.01
	TLASSO	**0.00**	**0.31**	**0.00**	**0.56**	**0.01**	**0.37**	**0.10**	**0.82**	**0.83**	**0.05**
	SCAD	0.00	0.34	0.38	0.30	0.48	0.14	0.11	0.83	0.84	0.05
	LASSO	0.00	0.67	0.01	0.72	0.20	0.46	0.13	0.83	0.84	0.04
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.17**	**0.12**	**0.19**	**0.17**	**0.09**	**0.81**	**0.82**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.17**	**0.12**	**0.19**	**0.17**	**0.09**	**0.81**	**0.82**	**0.01**
	TLASSO	0.00	0.56	0.00	0.74	0.02	0.60	0.10	0.81	0.82	0.05
	SCAD	0.00	0.62	0.03	0.54	0.25	0.28	0.11	0.81	0.82	0.04
	LASSO	0.00	0.66	0.00	0.66	0.17	0.38	0.11	0.81	0.82	0.04
M1/M3	TriSNAR$_G$	0.00	0.32	0.30	0.32	0.38	0.27	0.15	0.85	0.86	0.01
	TriSNAR$_A$	**0.00**	**0.30**	**0.26**	**0.28**	**0.45**	**0.22**	**0.15**	**0.85**	**0.86**	**0.00**
	TLASSO	0.04	0.31	0.14	0.58	0.29	0.46	0.14	0.85	0.86	0.05
	SCAD	0.00	0.31	0.22	0.34	0.62	0.20	0.16	0.86	0.87	0.04
	LASSO	0.00	0.33	0.13	0.46	0.49	0.29	0.15	0.87	0.88	0.03
NS1	TriSNAR$_G$	0.00	0.55	0.65	0.43	0.78	0.15	0.13	0.83	0.84	0.01
	TriSNAR$_A$	0.00	0.54	0.35	0.34	0.77	0.17	0.13	0.83	0.84	0.01
	TLASSO	**0.00**	**0.32**	**0.00**	**0.17**	**0.45**	**0.05**	**0.11**	**0.82**	**0.83**	**0.05**
	SCAD	0.00	0.57	0.24	0.37	0.76	0.23	0.14	0.82	0.84	0.05
	LASSO	0.00	0.67	0.03	0.57	0.65	0.40	0.14	0.83	0.84	0.04

Table 9: Simulation $d = 10$ with 3 lags, $T = 500$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.01**
	TLASSO	0.00	0.19	0.00	0.65	0.00	0.36	0.07	0.85	0.85	0.07
	SCAD	0.00	0.09	0.00	0.10	0.00	0.02	0.05	0.85	0.85	0.04
	LASSO	0.00	0.52	0.00	0.66	0.00	0.27	0.08	0.85	0.85	0.04
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.84**	**0.84**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.84**	**0.84**	**0.01**
	TLASSO	0.00	0.51	0.00	0.90	0.00	0.64	0.07	0.84	0.84	0.08
	SCAD	0.00	0.65	0.00	0.89	0.00	0.56	0.05	0.83	0.84	0.04
	LASSO	0.00	0.66	0.00	0.92	0.00	0.62	0.07	0.84	0.84	0.03
M1	TriSNAR$_G$	0.00	0.02	0.00	0.01	0.00	0.17	0.07	0.82	0.83	0.01
	TriSNAR$_A$	**0.00**	**0.01**	**0.00**	**0.00**	**0.00**	**0.17**	**0.07**	**0.82**	**0.83**	**0.01**
	TLASSO	0.00	0.48	0.00	0.59	0.00	0.39	0.08	0.82	0.83	0.10
	SCAD	0.00	0.55	0.00	0.38	0.16	0.16	0.10	0.82	0.83	0.06
	LASSO	0.00	0.67	0.00	0.79	0.00	0.57	0.11	0.82	0.83	0.06
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.01**	**0.35**	**0.01**	**0.27**	**0.07**	**0.81**	**0.81**	**0.01**
	TriSNAR$_A$	0.00	0.01	0.01	0.35	0.01	0.27	0.07	0.81	0.81	0.01
	TLASSO	0.00	0.56	0.00	0.75	0.00	0.59	0.08	0.81	0.81	0.12
	SCAD	0.00	0.60	0.00	0.51	0.03	0.22	0.07	0.81	0.81	0.05
	LASSO	0.00	0.65	0.00	0.66	0.01	0.35	0.09	0.81	0.82	0.05
M1/M3	TriSNAR$_G$	0.00	0.10	0.09	0.28	0.12	0.26	0.14	0.84	0.85	0.01
	TriSNAR$_A$	**0.00**	**0.05**	**0.04**	**0.27**	**0.24**	**0.12**	**0.14**	**0.84**	**0.85**	**0.01**
	TLASSO	0.00	0.33	0.00	0.63	0.04	0.51	0.13	0.84	0.85	0.10
	SCAD	0.00	0.33	0.01	0.43	0.24	0.20	0.15	0.84	0.85	0.05
	LASSO	0.00	0.33	0.00	0.59	0.11	0.35	0.14	0.85	0.86	0.04
NS1	TriSNAR$_G$	**0.00**	**0.25**	**0.01**	**0.06**	**0.44**	**0.02**	**0.10**	**0.82**	**0.83**	**0.01**
	TriSNAR$_A$	0.00	0.24	0.01	0.07	0.52	0.03	0.10	0.82	0.83	0.01
	TLASSO	0.00	0.49	0.00	0.21	0.37	0.05	0.09	0.82	0.83	0.10
	SCAD	0.00	0.63	0.04	0.30	0.71	0.16	0.11	0.82	0.83	0.06
	LASSO	0.00	0.67	0.00	0.63	0.51	0.47	0.12	0.82	0.83	0.06

Table 10: Simulation $d = 10$ with 3 lags, $T = 1000$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.01**
	TLASSO	0.00	0.23	0.00	0.86	0.00	0.47	0.06	0.85	0.85	0.12
	SCAD	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.05**	**0.85**	**0.85**	**0.06**
	LASSO	0.00	0.66	0.00	0.91	0.00	0.61	0.07	0.85	0.85	0.06
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.01**	**0.00**	**0.00**	**0.03**	**0.84**	**0.84**	**0.01**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.01**	**0.00**	**0.00**	**0.03**	**0.84**	**0.84**	**0.01**
	TLASSO	0.00	0.51	0.00	0.90	0.00	0.63	0.06	0.84	0.84	0.14
	SCAD	0.00	0.65	0.00	0.89	0.00	0.54	0.04	0.84	0.84	0.05
	LASSO	0.00	0.66	0.00	0.91	0.00	0.59	0.06	0.84	0.84	0.04
M1	TriSNAR$_G$	**0.00**	**0.12**	**0.00**	**0.35**	**0.00**	**0.28**	**0.07**	**0.82**	**0.82**	**0.02**
	TriSNAR$_A$	0.00	0.19	0.00	0.36	0.00	0.24	0.07	0.82	0.82	0.01
	TLASSO	0.00	0.46	0.00	0.63	0.00	0.45	0.08	0.82	0.82	0.18
	SCAD	0.00	0.61	0.00	0.53	0.01	0.25	0.09	0.82	0.83	0.10
	LASSO	0.00	0.67	0.00	0.75	0.00	0.53	0.10	0.82	0.83	0.09
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.11**	**0.00**	**0.15**	**0.05**	**0.81**	**0.81**	**0.02**
	TriSNAR$_A$	0.00	0.01	0.00	0.12	0.00	0.15	0.05	0.81	0.81	0.02
	TLASSO	0.00	0.55	0.00	0.75	0.00	0.59	0.07	0.81	0.81	0.23
	SCAD	0.00	0.59	0.00	0.50	0.00	0.20	0.06	0.81	0.81	0.08
	LASSO	0.00	0.66	0.00	0.68	0.00	0.37	0.08	0.81	0.81	0.07
M1/M3	TriSNAR$_G$	0.00	0.28	0.03	0.46	0.04	0.28	0.13	0.84	0.85	0.02
	TriSNAR$_A$	0.00	0.14	0.01	0.29	0.07	0.16	0.13	0.84	0.85	0.02
	TLASSO	0.00	0.33	0.00	0.66	0.00	0.56	0.12	0.84	0.85	0.20
	SCAD	**0.00**	**0.24**	**0.00**	**0.17**	**0.14**	**0.06**	**0.13**	**0.84**	**0.85**	**0.07**
	LASSO	0.00	0.33	0.00	0.66	0.01	0.46	0.13	0.84	0.85	0.07
NS1	TriSNAR$_G$	0.00	0.59	0.00	0.24	0.39	0.06	0.09	0.82	0.82	0.02
	TriSNAR$_A$	**0.00**	**0.44**	**0.00**	**0.17**	**0.43**	**0.05**	**0.09**	**0.82**	**0.82**	**0.02**
	TLASSO	0.00	0.56	0.00	0.38	0.30	0.14	0.08	0.82	0.82	0.19
	SCAD	0.00	0.57	0.00	0.30	0.65	0.16	0.10	0.82	0.83	0.10
	LASSO	0.00	0.67	0.00	0.63	0.47	0.46	0.11	0.82	0.83	0.09

Table 11: Simulation $d = 20$ with 3 lags, $T = 100$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.04**	**0.85**	**0.85**	**0.11**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.04**	**0.85**	**0.85**	**0.02**
	TLASSO	0.00	0.01	0.00	0.80	0.00	0.24	0.07	0.85	0.86	0.10
	SCAD	0.00	0.48	0.00	0.62	0.11	0.13	0.06	0.85	0.86	0.17
	LASSO	0.00	0.62	0.00	0.84	0.10	0.29	0.08	0.87	0.88	0.14
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.84**	**0.84**	**0.11**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.84**	**0.84**	**0.02**
	TLASSO	0.00	0.50	0.00	0.95	0.00	0.71	0.08	0.85	0.86	0.10
	SCAD	0.00	0.67	0.00	0.94	0.03	0.55	0.07	0.82	0.85	0.18
	LASSO	0.00	0.67	0.00	0.97	0.01	0.70	0.08	0.84	0.86	0.14
M1	TriSNAR$_G$	0.00	0.00	0.80	0.00	0.66	0.00	0.08	0.84	0.84	0.15
	TriSNAR$_A$	0.00	0.00	0.80	0.00	0.66	0.00	0.08	0.84	0.84	0.03
	TLASSO	**0.00**	**0.19**	**0.00**	**0.66**	**0.18**	**0.37**	**0.09**	**0.83**	**0.84**	**0.12**
	SCAD	0.00	0.57	0.37	0.54	0.64	0.18	0.09	0.83	0.85	0.23
	LASSO	0.00	0.65	0.21	0.69	0.59	0.33	0.11	0.86	0.87	0.20
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.77**	**0.00**	**0.64**	**0.02**	**0.08**	**0.82**	**0.83**	**0.14**
	TriSNAR$_A$	**0.00**	**0.00**	**0.77**	**0.00**	**0.64**	**0.02**	**0.08**	**0.82**	**0.83**	**0.03**
	TLASSO	0.00	0.50	0.07	0.81	0.45	0.61	0.10	0.83	0.84	0.12
	SCAD	0.00	0.66	0.06	0.78	0.48	0.45	0.10	0.80	0.84	0.20
	LASSO	0.00	0.67	0.02	0.84	0.41	0.54	0.10	0.82	0.84	0.16
M1/M3	TriSNAR$_G$	**0.14**	**0.00**	**0.82**	**0.01**	**0.74**	**0.01**	**0.10**	**0.86**	**0.87**	**0.10**
	TriSNAR$_A$	**0.14**	**0.00**	**0.82**	**0.01**	**0.74**	**0.01**	**0.10**	**0.86**	**0.87**	**0.02**
	TLASSO	0.50	0.00	0.68	0.27	0.89	0.14	0.11	0.89	0.89	0.12
	SCAD	0.00	0.33	0.19	0.70	0.81	0.55	0.13	0.84	0.90	0.19
	LASSO	0.00	0.33	0.12	0.74	0.73	0.58	0.12	0.87	0.89	0.16
NS1	TriSNAR$_G$	0.00	0.00	0.95	0.00	0.95	0.00	0.10	0.84	0.84	0.16
	TriSNAR$_A$	0.00	0.00	0.95	0.00	0.95	0.00	0.10	0.84	0.84	0.03
	TLASSO	**0.00**	**0.31**	**0.01**	**0.07**	**0.71**	**0.01**	**0.09**	**0.82**	**0.83**	**0.13**
	SCAD	0.00	0.65	0.56	0.42	0.94	0.24	0.11	0.83	0.86	0.21
	LASSO	0.00	0.66	0.43	0.51	0.91	0.33	0.11	0.86	0.87	0.17

Table 12: Simulation $d = 20$ with 3 lags, $T = 200$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	0.00	0.12	0.00	0.14	0.00	0.04	0.04	0.85	0.85	0.04
	TriSNAR$_A$	**0.00**	**0.12**	**0.00**	**0.14**	**0.00**	**0.04**	**0.04**	**0.85**	**0.85**	**0.01**
	TLASSO	0.00	0.12	0.00	0.94	0.00	0.62	0.06	0.85	0.85	0.16
	SCAD	0.00	0.15	0.00	0.21	0.01	0.10	0.04	0.85	0.85	0.13
	LASSO	0.00	0.65	0.00	0.94	0.00	0.67	0.07	0.85	0.86	0.11
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.84**	**0.84**	**0.04**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.84**	**0.84**	**0.02**
	TLASSO	0.00	0.50	0.00	0.95	0.00	0.69	0.06	0.84	0.85	0.15
	SCAD	0.00	0.66	0.00	0.94	0.00	0.52	0.05	0.83	0.84	0.12
	LASSO	0.00	0.67	0.00	0.96	0.00	0.64	0.06	0.84	0.85	0.10
M1	TriSNAR$_G$	**0.00**	**0.02**	**0.10**	**0.01**	**0.12**	**0.18**	**0.07**	**0.82**	**0.83**	**0.06**
	TriSNAR$_A$	**0.00**	**0.02**	**0.10**	**0.01**	**0.12**	**0.18**	**0.07**	**0.82**	**0.83**	**0.02**
	TLASSO	0.00	0.10	0.00	0.73	0.02	0.38	0.07	0.83	0.83	0.21
	SCAD	0.00	0.27	0.52	0.20	0.62	0.05	0.08	0.84	0.84	0.17
	LASSO	0.00	0.67	0.01	0.86	0.23	0.55	0.09	0.83	0.84	0.15
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.62**	**0.05**	**0.53**	**0.05**	**0.07**	**0.82**	**0.82**	**0.06**
	TriSNAR$_A$	0.00	0.01	0.61	0.06	0.52	0.06	0.07	0.82	0.82	0.02
	TLASSO	0.00	0.51	0.00	0.84	0.10	0.60	0.08	0.82	0.82	0.21
	SCAD	0.00	0.65	0.01	0.73	0.29	0.34	0.08	0.80	0.83	0.15
	LASSO	0.00	0.66	0.00	0.79	0.23	0.41	0.08	0.82	0.83	0.13
M1/M3	TriSNAR$_G$	**0.02**	**0.08**	**0.09**	**0.52**	**0.53**	**0.28**	**0.11**	**0.87**	**0.88**	**0.04**
	TriSNAR$_A$	0.02	0.09	0.16	0.43	0.66	0.22	0.11	0.86	0.88	0.02
	TLASSO	0.34	0.11	0.42	0.71	0.56	0.44	0.10	0.86	0.87	0.20
	SCAD	0.00	0.32	0.19	0.52	0.72	0.26	0.12	0.86	0.88	0.15
	LASSO	0.00	0.33	0.09	0.63	0.61	0.35	0.11	0.87	0.88	0.13
NS1	TriSNAR$_G$	**0.00**	**0.02**	**0.13**	**0.01**	**0.77**	**0.00**	**0.09**	**0.82**	**0.83**	**0.07**
	TriSNAR$_A$	0.00	0.33	0.17	0.15	0.82	0.08	0.10	0.82	0.83	0.02
	TLASSO	0.00	0.22	0.00	0.03	0.70	0.01	0.08	0.82	0.82	0.21
	SCAD	0.00	0.60	0.40	0.38	0.90	0.21	0.10	0.82	0.84	0.17
	LASSO	0.00	0.67	0.03	0.60	0.81	0.45	0.10	0.82	0.84	0.15

Table 13: Simulation $d = 20$ with 3 lags, $T = 500$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.85**	**0.85**	**0.03**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.85**	**0.85**	**0.02**
	TLASSO	0.00	0.00	0.00	0.79	0.00	0.21	0.05	0.85	0.85	0.29
	SCAD	0.00	0.11	0.00	0.14	0.00	0.02	0.04	0.85	0.85	0.15
	LASSO	0.00	0.61	0.00	0.81	0.00	0.24	0.06	0.85	0.85	0.14
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**0.03**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**0.02**
	TLASSO	0.00	0.50	0.00	0.95	0.00	0.66	0.05	0.84	0.84	0.30
	SCAD	0.00	0.66	0.00	0.93	0.00	0.53	0.03	0.83	0.84	0.13
	LASSO	0.00	0.66	0.00	0.96	0.00	0.61	0.05	0.84	0.84	0.12
M1	TriSNAR$_G$	0.00	0.14	0.00	0.15	0.00	0.24	0.06	0.82	0.83	0.06
	TriSNAR$_A$	**0.00**	**0.13**	**0.00**	**0.14**	**0.00**	**0.24**	**0.06**	**0.82**	**0.83**	**0.03**
	TLASSO	0.00	0.49	0.00	0.79	0.00	0.55	0.06	0.82	0.83	0.42
	SCAD	0.00	0.64	0.00	0.59	0.17	0.22	0.07	0.82	0.83	0.22
	LASSO	0.00	0.67	0.00	0.89	0.02	0.64	0.08	0.82	0.83	0.20
M2	TriSNAR$_G$	0.00	0.00	0.03	0.09	0.03	0.20	0.05	0.81	0.81	0.06
	TriSNAR$_A$	**0.00**	**0.01**	**0.03**	**0.08**	**0.03**	**0.20**	**0.05**	**0.81**	**0.81**	**0.03**
	TLASSO	0.00	0.54	0.00	0.86	0.00	0.66	0.07	0.81	0.81	0.49
	SCAD	0.00	0.65	0.00	0.75	0.04	0.31	0.06	0.81	0.81	0.20
	LASSO	0.00	0.66	0.00	0.80	0.02	0.37	0.07	0.81	0.82	0.18
M1/M3	TriSNAR$_G$	**0.00**	**0.00**	**0.77**	**0.01**	**0.64**	**0.01**	**0.10**	**0.85**	**0.86**	**0.04**
	TriSNAR$_A$	0.00	0.04	0.69	0.08	0.60	0.03	0.10	0.85	0.86	0.04
	TLASSO	0.00	0.33	0.00	0.79	0.07	0.64	0.10	0.84	0.85	0.44
	SCAD	0.00	0.31	0.04	0.56	0.34	0.25	0.11	0.84	0.86	0.20
	LASSO	0.00	0.33	0.00	0.80	0.14	0.46	0.10	0.85	0.86	0.18
NS1	TriSNAR$_G$	0.00	0.57	0.00	0.17	0.71	0.05	0.07	0.81	0.82	0.06
	TriSNAR$_A$	0.00	0.61	0.02	0.24	0.82	0.11	0.08	0.81	0.82	0.03
	TLASSO	**0.00**	**0.49**	**0.00**	**0.19**	**0.59**	**0.03**	**0.07**	**0.81**	**0.82**	**0.43**
	SCAD	0.00	0.67	0.02	0.44	0.83	0.26	0.09	0.81	0.82	0.23
	LASSO	0.00	0.67	0.00	0.64	0.72	0.49	0.09	0.81	0.82	0.21

Table 14: Simulation $d = 20$ with 3 lags, $T = 1000$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.85**	**0.85**	**0.05**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.85**	**0.85**	**0.04**
	TLASSO	0.00	0.35	0.00	0.95	0.00	0.63	0.05	0.85	0.85	0.47
	SCAD	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.03**	**0.85**	**0.85**	**0.23**
	LASSO	0.00	0.67	0.00	0.97	0.00	0.72	0.05	0.85	0.85	0.22
D2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**0.05**
	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.84**	**0.84**	**0.04**
	TLASSO	0.00	0.50	0.00	0.94	0.00	0.64	0.04	0.84	0.84	0.53
	SCAD	0.00	0.66	0.00	0.93	0.00	0.51	0.03	0.84	0.84	0.19
	LASSO	0.00	0.66	0.00	0.94	0.00	0.57	0.04	0.84	0.84	0.18
M1	TriSNAR$_G$	**0.00**	**0.01**	**0.00**	**0.04**	**0.00**	**0.15**	**0.05**	**0.82**	**0.82**	**0.08**
	TriSNAR$_A$	0.00	0.03	0.00	0.04	0.00	0.14	0.05	0.82	0.82	0.05
	TLASSO	0.00	0.41	0.00	0.76	0.00	0.44	0.06	0.82	0.82	0.77
	SCAD	0.00	0.60	0.00	0.45	0.07	0.15	0.06	0.82	0.83	0.37
	LASSO	0.00	0.67	0.00	0.87	0.00	0.55	0.07	0.82	0.83	0.35
M2	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.45**	**0.00**	**0.29**	**0.04**	**0.81**	**0.81**	**0.08**
	TriSNAR$_A$	0.00	0.04	0.00	0.46	0.00	0.29	0.04	0.81	0.81	0.05
	TLASSO	0.00	0.55	0.00	0.87	0.00	0.68	0.06	0.81	0.81	0.99
	SCAD	0.00	0.65	0.00	0.74	0.00	0.30	0.05	0.81	0.81	0.33
	LASSO	0.00	0.66	0.00	0.79	0.00	0.36	0.06	0.81	0.81	0.31
M1/M3	TriSNAR$_G$	0.00	0.12	0.02	0.08	0.02	0.22	0.09	0.84	0.85	0.05
	TriSNAR$_A$	**0.00**	**0.11**	**0.00**	**0.10**	**0.03**	**0.18**	**0.09**	**0.84**	**0.85**	**0.04**
	TLASSO	0.00	0.33	0.00	0.80	0.01	0.65	0.09	0.84	0.85	0.90
	SCAD	0.00	0.33	0.00	0.54	0.11	0.14	0.10	0.84	0.85	0.30
	LASSO	0.00	0.33	0.00	0.78	0.03	0.44	0.09	0.85	0.85	0.28
NS1	TriSNAR$_G$	**0.00**	**0.00**	**0.00**	**0.00**	**0.62**	**0.00**	**0.07**	**0.81**	**0.82**	**0.09**
	TriSNAR$_A$	0.00	0.31	0.00	0.12	0.72	0.05	0.07	0.81	0.82	0.05
	TLASSO	0.00	0.43	0.00	0.12	0.61	0.03	0.06	0.81	0.82	0.77
	SCAD	0.00	0.66	0.00	0.37	0.83	0.20	0.08	0.81	0.82	0.38
	LASSO	0.00	0.67	0.00	0.63	0.72	0.45	0.08	0.81	0.82	0.36

Table 15: Simulation $d = 100$ with 3 lags, $T = 100$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_A$	**0.00**	**0.63**	**0.00**	**0.88**	**0.07**	**0.19**	**0.03**	**0.86**	**0.86**	**7.08**
	TLASSO	0.00	0.67	0.00	1.00	0.00	1.00	0.27	0.36	1.21	3.97
	SCAD	0.00	0.67	0.00	0.97	0.22	0.37	0.03	0.87	0.87	1.31
	LASSO	0.00	0.67	0.00	0.98	0.19	0.50	0.04	0.88	0.88	1.37
D2	TriSNAR$_A$	**0.00**	**0.67**	**0.00**	**0.99**	**0.04**	**0.64**	**0.04**	**0.86**	**0.87**	**1.77**
	TLASSO	0.00	0.67	0.00	1.00	0.00	1.00	0.27	0.35	1.21	4.04
	SCAD	0.00	0.67	0.00	0.99	0.07	0.74	0.05	0.87	0.89	1.31
	LASSO	0.00	0.67	0.00	1.00	0.05	0.80	0.04	0.86	0.87	1.38
M1	TriSNAR$_A$	0.00	0.66	0.22	0.89	0.56	0.35	0.05	0.82	0.85	11.05
	TLASSO	**0.00**	**0.55**	**0.00**	**0.93**	**0.09**	**0.84**	**0.23**	**0.49**	**1.09**	**4.15**
	SCAD	0.00	0.67	0.08	0.94	0.66	0.52	0.05	0.82	0.86	1.50
	LASSO	0.00	0.67	0.05	0.94	0.65	0.52	0.05	0.86	0.88	1.50
M2	TriSNAR$_A$	**0.00**	**0.65**	**0.06**	**0.91**	**0.49**	**0.53**	**0.05**	**0.79**	**0.85**	**2.33**
	TLASSO	0.00	0.67	0.00	0.98	0.00	0.99	0.25	0.36	1.13	4.19
	SCAD	0.00	0.67	0.00	0.97	0.53	0.76	0.07	0.76	0.87	1.52
	LASSO	0.00	0.67	0.00	0.97	0.53	0.72	0.05	0.82	0.85	1.53
M1/M3	TriSNAR$_A$	0.00	0.33	0.09	0.92	0.82	0.74	0.07	0.82	0.91	1.50
	TLASSO	**0.00**	**0.33**	**0.00**	**0.97**	**0.01**	**0.98**	**0.25**	**0.43**	**1.16**	**3.94**
	SCAD	0.00	0.33	0.06	0.95	0.87	0.83	0.07	0.81	0.92	1.39
	LASSO	0.00	0.33	0.04	0.95	0.86	0.83	0.06	0.86	0.89	1.42
NS1	TriSNAR$_A$	0.00	0.66	0.69	0.34	0.99	0.19	0.05	0.82	0.86	10.67
	TLASSO	**0.00**	**0.67**	**0.00**	**0.67**	**0.00**	**0.67**	**0.25**	**0.39**	**1.12**	**4.05**
	SCAD	0.00	0.67	0.53	0.54	0.99	0.38	0.06	0.81	0.86	1.51
	LASSO	0.00	0.67	0.49	0.55	0.98	0.38	0.05	0.86	0.88	1.51

Table 16: Simulation $d = 100$ with 3 lags, $T = 200$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**2.30**
	TLASSO	0.00	0.00	0.00	0.98	0.00	0.38	0.03	0.85	0.85	7.79
	SCAD	0.00	0.26	0.00	0.33	0.03	0.01	0.02	0.85	0.85	4.03
	LASSO	0.00	0.55	0.00	0.74	0.03	0.05	0.04	0.87	0.87	3.62
D2	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.01**	**0.84**	**0.84**	**3.35**
	TLASSO	0.00	0.50	0.00	0.98	0.00	0.48	0.03	0.86	0.86	8.00
	SCAD	0.00	0.67	0.00	0.99	0.00	0.65	0.04	0.82	0.86	4.11
	LASSO	0.00	0.67	0.00	0.99	0.00	0.75	0.03	0.84	0.85	3.71
M1	TriSNAR$_A$	0.00	0.00	0.80	0.00	0.67	0.00	0.03	0.84	0.84	3.52
	TLASSO	**0.00**	**0.00**	**0.00**	**0.93**	**0.12**	**0.38**	**0.04**	**0.83**	**0.84**	**8.78**
	SCAD	0.00	0.63	0.14	0.74	0.62	0.18	0.04	0.83	0.84	4.22
	LASSO	0.00	0.64	0.09	0.73	0.61	0.14	0.05	0.87	0.87	3.86
M2	TriSNAR$_A$	**0.00**	**0.00**	**0.77**	**0.00**	**0.64**	**0.02**	**0.03**	**0.82**	**0.83**	**5.10**
	TLASSO	0.00	0.50	0.26	0.93	0.51	0.46	0.04	0.84	0.85	9.08
	SCAD	0.00	0.67	0.00	0.97	0.33	0.72	0.06	0.78	0.84	4.25
	LASSO	0.00	0.67	0.00	0.97	0.35	0.59	0.04	0.82	0.83	3.89
M1/M3	TriSNAR$_A$	**0.01**	**0.00**	**0.79**	**0.01**	**0.67**	**0.00**	**0.04**	**0.85**	**0.86**	**2.58**
	TLASSO	0.50	0.00	0.57	0.68	0.83	0.21	0.05	0.89	0.89	8.13
	SCAD	0.00	0.33	0.04	0.92	0.76	0.59	0.06	0.84	0.89	4.09
	LASSO	0.00	0.33	0.02	0.94	0.73	0.63	0.05	0.87	0.89	3.69
NS1	TriSNAR$_A$	0.00	0.13	0.88	0.06	0.99	0.03	0.04	0.84	0.84	3.71
	TLASSO	**0.00**	**0.01**	**0.00**	**0.00**	**0.95**	**0.00**	**0.04**	**0.82**	**0.83**	**8.39**
	SCAD	0.00	0.67	0.51	0.36	0.98	0.21	0.05	0.84	0.84	4.31
	LASSO	0.00	0.67	0.43	0.36	0.98	0.20	0.05	0.87	0.87	3.90

Table 17: Simulation $d = 100$ with 3 lags, $T = 500$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**1.56**
	TLASSO	0.00	0.14	0.00	0.83	0.00	0.57	0.02	0.85	0.85	11.86
	SCAD	0.00	0.04	0.00	0.06	0.00	0.01	0.02	0.85	0.85	6.95
	LASSO	0.00	0.67	0.00	0.98	0.00	0.45	0.03	0.85	0.85	6.03
D2	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.01**	**0.84**	**0.84**	**1.75**
	TLASSO	0.00	0.50	0.00	0.99	0.00	0.87	0.03	0.84	0.84	12.79
	SCAD	0.00	0.67	0.00	0.99	0.00	0.54	0.02	0.83	0.84	7.26
	LASSO	0.00	0.67	0.00	0.99	0.00	0.68	0.02	0.84	0.84	6.32
M1	TriSNAR$_A$	**0.00**	**0.06**	**0.17**	**0.03**	**0.14**	**0.17**	**0.03**	**0.82**	**0.83**	**3.76**
	TLASSO	0.00	0.17	0.00	0.95	0.00	0.64	0.03	0.82	0.83	16.19
	SCAD	0.00	0.50	0.18	0.34	0.62	0.05	0.03	0.84	0.84	8.19
	LASSO	0.00	0.67	0.00	0.96	0.11	0.43	0.04	0.83	0.84	7.33
M2	TriSNAR$_A$	**0.00**	**0.01**	**0.00**	**0.00**	**0.00**	**0.21**	**0.02**	**0.81**	**0.81**	**3.91**
	TLASSO	0.00	0.50	0.00	0.96	0.01	0.64	0.03	0.81	0.82	18.27
	SCAD	0.00	0.67	0.00	0.98	0.04	0.73	0.05	0.80	0.82	8.49
	LASSO	0.00	0.67	0.00	0.96	0.05	0.45	0.03	0.82	0.82	7.56
M1/M3	TriSNAR$_A$	**0.00**	**0.33**	**0.14**	**0.72**	**0.17**	**0.23**	**0.04**	**0.84**	**0.85**	**3.40**
	TLASSO	0.42	0.01	0.48	0.64	0.66	0.19	0.05	0.87	0.88	15.03
	SCAD	0.00	0.33	0.02	0.76	0.59	0.17	0.05	0.85	0.86	8.20
	LASSO	0.00	0.33	0.00	0.83	0.54	0.20	0.05	0.88	0.88	7.12
NS1	TriSNAR$_A$	0.00	0.66	0.01	0.33	0.97	0.17	0.04	0.81	0.81	3.01
	TLASSO	**0.00**	**0.24**	**0.00**	**0.01**	**0.92**	**0.00**	**0.03**	**0.81**	**0.81**	**14.86**
	SCAD	0.00	0.67	0.00	0.57	0.96	0.37	0.04	0.80	0.81	8.13
	LASSO	0.00	0.67	0.00	0.54	0.96	0.35	0.04	0.82	0.83	7.10

Table 18: Simulation $d = 100$ with 3 lags, $T = 1000$ for D1, D2, M1, M2, M1/M3, NS1. The best performing model in terms of lowest FN and FP for each number of observations is indicated in bold.

		FN.l	FP.l	FN.g	FP.g	FN.e	FP.e	MAE.para	MAE.res	MAFE.res	time (in s)/combination
D1	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.02**	**0.85**	**0.85**	**2.11**
	TLASSO	0.00	0.01	0.00	0.98	0.00	0.38	0.02	0.85	0.85	30.18
	SCAD	0.00	0.02	0.00	0.02	0.00	0.00	0.02	0.85	0.85	8.62
	LASSO	0.00	0.55	0.00	0.66	0.00	0.06	0.03	0.85	0.85	7.66
D2	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.01**	**0.84**	**0.84**	**2.56**
	TLASSO	0.00	0.50	0.00	0.99	0.00	0.87	0.02	0.84	0.84	32.85
	SCAD	0.00	0.67	0.00	0.99	0.00	0.54	0.01	0.84	0.84	8.39
	LASSO	0.00	0.67	0.00	0.99	0.00	0.64	0.02	0.84	0.84	7.44
M1	TriSNAR$_A$	**0.00**	**0.60**	**0.00**	**0.41**	**0.00**	**0.15**	**0.02**	**0.82**	**0.82**	**4.50**
	TLASSO	0.00	0.02	0.00	0.94	0.00	0.44	0.03	0.82	0.83	47.46
	SCAD	0.00	0.67	0.00	0.94	0.02	0.49	0.03	0.82	0.83	14.58
	LASSO	0.00	0.67	0.00	0.88	0.02	0.37	0.04	0.83	0.83	13.52
M2	TriSNAR$_A$	**0.00**	**0.00**	**0.00**	**0.00**	**0.00**	**0.12**	**0.02**	**0.81**	**0.81**	**4.17**
	TLASSO	0.00	0.50	0.00	0.96	0.00	0.59	0.03	0.81	0.81	57.68
	SCAD	0.00	0.67	0.00	0.98	0.00	0.76	0.04	0.80	0.82	11.25
	LASSO	0.00	0.67	0.00	0.95	0.00	0.39	0.03	0.81	0.81	10.39
M1/M3	TriSNAR$_A$	0.00	0.00	0.71	0.00	0.62	0.00	0.04	0.85	0.85	2.57
	TLASSO	0.00	0.33	0.00	0.93	0.25	0.80	0.05	0.85	0.85	48.72
	SCAD	**0.00**	**0.32**	**0.01**	**0.34**	**0.51**	**0.03**	**0.05**	**0.85**	**0.85**	**10.41**
	LASSO	0.00	0.33	0.00	0.96	0.07	0.55	0.04	0.85	0.86	9.43
NS1	TriSNAR$_A$	0.00	0.63	0.00	0.16	0.97	0.06	0.04	0.81	0.81	2.97
	TLASSO	**0.00**	**0.29**	**0.00**	**0.11**	**0.89**	**0.01**	**0.03**	**0.81**	**0.81**	**41.72**
	SCAD	0.00	0.67	0.00	0.51	0.96	0.30	0.04	0.81	0.81	11.18
	LASSO	0.00	0.67	0.00	0.65	0.93	0.49	0.04	0.81	0.81	10.17