MSc Thesis

Time-Varying Source Separation by Joint Diagnolization on Autocovariances

Yan Pan

JYVÄSKYLÄN YLIOPISTO
UNIVERSITY OF JYVÄSKYLÄ

Department of Mathematics and Statistics
Statistics
29 February 2020

UNIVERSITY OF JYVÄSKYLÄ, Faculty of Mathematics and Science
Department of Mathematics and Statistics
Statistics

Yan Pan:	Time-Varying Source Separation by Joint Diagnolization on Autocovariances
MSc thesis:	33 p., 4 appendices (5 p.)
Supervisor:	University lecturer Sara Taskinen
Inspectors:	Professor Juha Karvanen
February 2020	

Keywords: Blind Source Separation, Second-Order Blind Identification, SOBI, Time-Varying Second-Order Blind Identification, TV-SOBI.

Blind Source Separation (BSS) seeks to recover the true signals from the observed ones when only limited information about the mixing matrix and the original sources are available. There are various methodologies established to solve the BSS problems, and notably, Second-Order Blind Identification (SOBI) identifies sources through second-order statistics (Tong et al., 1994). This thesis stretches the Second-Order Source Separation (SOS) model in terms of latent time variation in the mixing mechanism that was initially proposed by Yeredor (2003). An improved algorithm, Linearly Time-Varying SOBI (LTV-SOBI), together with alternatives attempts to estimate mixing parameters and ultimately derives latent independent sources employing sample autocovariance decomposition and joint diagonalization. The performance of LTV-SOBI is analyzed with simulated data by extending the performance metric Signal-to-interference ratio (SIR, Yeredor, 2003) into the time-varying case. Simulation results suggest the superiority of the new LTV-SOBI algorithm compared with Yeredor's TV-SOBI algorithm, despite overall results are still non-optimal. In addition to the full implementation of LTV-SOBI algorithm in `R`, an interactive dashboard is designed to enable further outlook of algorithm performance.

JYVÄSKYLÄN YLIOPISTO, Matemaattis-luonnontieteellinen tiedekunta
Matematiikan ja tilastotieteen laitos
Tilastotiede

Yan Pan:	Ajassa muuttuvien signaalien erottelu autokovarianssimatriisien yhteisdiagonalisoinnilla
Pro-gradu -tutkielma:	33 s., 4 liitettä (5 s.)
Työn ohjaaja:	Yliopistonlehtori Sara Taskinen
Tarkastajat:	Professori Juha Karvanen
Helmikuu 2020	

Hakusanat: Sokea signaalinerottelu, toisen asteen sokea signaalinerottelu, SOBI, ajassa muuttuvien signaalien erottelu, TV-SOBI

Sokealla signaalinerottelulla (Blind Source Separation, BSS) pyritään erottelemaan todelliset signaalit havaituista signaaleista, kun ennakkotietoja sekoitusmatriisista ja todellisista signaaleista on vain vähän saatavilla. BSS-ongelmien ratkaisemiseksi on kehitetty erilaisia menetelmiä. Näistä toisen asteen sokea signaalinerottelu (Second-Order Blind Identification, SOBI) tunnistaa lähteet toisen asteen tunnuslukujen avulla (Tong et al., 1994). Tässä opinnäytetyössä tarkastellaan toisen asteen sokean signaalinerottelumallin laajennusta (Yeredor, 2003), jossa sekoitusmatriisi muuttuu ajassa. Työssä esitellään paranneltu versio Yeredorin TV-SOBI (time-varying SOBI) algoritmista sekä sen variaatioita. Algoritmit pyrkivät estimoimaan sekoitusmatriisin ja edelleen latentit signaalit otosautokovarianssimatriisin hajotelman sekä yhteisdiagonalisoinnin avulla. Kehitetyn algoritmin (linearly time-varying SOBI, LTV-SOBI) suorituskysyä arvioidaan simulointien avulla. Suorituskyvyn mittarina käytetään tässä työssä kehitettyä signaali-häiriö suhteen (Signal-to-Inference Ratio, SIR, Yeredor, 2003) laajennusta aikamuuttuvan signaalin tapaukseen. Simulaatiotulokset osoittavat uuden LTV-SOBI-algoritmin paremmuuden verrattuna Yeredorin TV-SOBI-algoritmiin. Tulokset eivät tosin ole vielä optimaalisia. Lisäksi työssä esitellään LTV-SOBI algorithmin `R` implementointi sekä interaktiivinen `R Shiny` sovellus, jonka avulla algoritmien suorituskykyä voidaan vertailla.

Contents

1 Introduction **2**

2 Blind Source Separation **5**

2.1 Ambiguities and Assumptions 6

2.2 Stationary Time-Series Source Separation Using Autocovariance Matrices 6

3 Time-Varying Second-Order Model Formulation and Yeredor's Solution **8**

3.1 TV-SOS Model and Assumptions 8

3.2 Yeredor's TV-SOBI Algorithm 9

4 Algorithm for Linearly Time-Varying Second Order Blind Identification **12**

4.1 Decomposition of Autocovariance Structure 12

4.2 Finding $\mathbf{\Omega}_0$ with Approximate Joint Diagnolization 14

4.3 Finding $\boldsymbol{\mathcal{E}}$ through $\widehat{\boldsymbol{\beta}_2}$ and $\widehat{\mathbf{\Omega}_0}$ 15

4.4 Signal Restoration 16

4.5 Minor Alternatives for LTV-SOBI 16

4.6 Summary of Algorithms 17

5 LTV-SOBI Performance Measures and Simulation Studies **19**

5.1 Extension of Minimal Distance Index 19

5.2 Extension of Signal-to-Inference Ratio 20

5.3 Simulation Study 21

5.4 Simulation Study Results 23

5.5 Performance Consistency over Singal Types and Time-Varying Rates 24

6 Discussion **30**

Appendix **32**

A Implementaion of LTV-SOBI in *R* 32

B Introduction to LTV-SOBI Performance Metric Explorer 32

C *R* Code for Simulation Study 33

D Supplementary Simulation Results 35

References **37**

1 Introduction

Imagine a situation that a small musical band composed of a piano, a bass, and a drum is playing at the stage, while three microphones are recording its performance. Rather than each microphone records one specific instrument, it can be expected that each microphone captures a slightly different mixture of the original sounds from the trio. In a real-world situation, there exist diversified needs for extracting structured information from observable mixtures of unknown signals. For example, a recording of speech may contain external noise from nearby road traffic, minor discussion among the audience and constant electronic interference in addition to the speech voice itself.

In most cases, the mixing mechanism is unknown or too costly to measure. Thus several statistical methods which try to recover the source signals given the observed ones based on some statictical assumptions, such as independence, have been developed (Hyvärinen & Oja, 2000). Such methods are called as Blind Source Separation (BSS, Jutten & Herault, 1991) methods. Further, the mixing mechanism is not necessarily static; in recording, the microphones can be moving slowly towards or away from the artists due to some relative movement (dancing, walking, etc.). Figure 1 provides a simple illustration of such time-varying BSS problem.

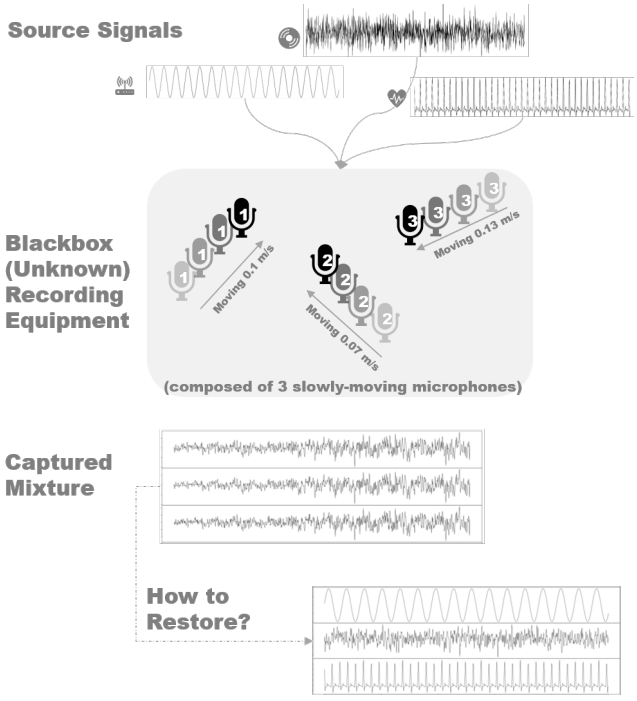

Figure 1: Illustration of Blind Source Separation with time-varying mixing mechanism

BSS methods include a set of unsupervised machine learning algorithms which

take input as a single data matrix. The characteristics of output usually cannot be accurately foreseen in advance due to lack of other relevant evidence and such algorithms mostly serve as exploratory purposes (Hyvärinen, 2013). As compared with supervised learning methods, such as regression and classification tree, unsupervised ones are more challenging and tend to be more subjective in the absence of a clear goal. Nevertheless, BSS and other exploratory data analysis methods gain increasing importance especially in the online marketing and healthcare industry (James et al., 2013). The modern information technology makes the massive quantity of data available, but subtracting structural insights can be an enormous challenge. The "blind" approaches endeavor to provide a unique perspective of data if computation resources are sufficiently available with rather limited human interferences.

This thesis expands the static blind source separation problem into linearly time-varying one (Yeredor, 2003) employing relatively modern tools. A full functional implementation and associated utilities in R will be produced. The thesis is organized as follows. Section 2 will first formulate the BSS problems and review established solutions, and then Section 3 elaborates the linearly time-varying structure of second-order source separation. Section 4 presents the new algorithms in detail, followed by Section 5, which will discuss the performance measures and provides simulation studies. Finally, the thesis concludes with a discussion of potential extension and performance-related topics in Section 6.

Regarding the notations, vectors and matrices are always marked as bold symbols while a lower case letter stands for a real-valued number. The most commonly used symbols are summarized below.

Table 1: Notation

Symbol	Meaning	Note
$\boldsymbol{I}$	identity matrix	compatible dimensions are assigned
$\boldsymbol{z}$	source signals	p-vector; a realization of (unobservable) stochastic process (Z_t)
$\boldsymbol{x}$	observed signals	p-vector; an observable stochastic process (X_t)
$\boldsymbol{\mu}$	location parameter	p-vector
$\boldsymbol{\Omega}_t$	mixing matrix at given time point t	$p \times p$ matrix. $\boldsymbol{\Omega}_t = (\boldsymbol{I} + t\boldsymbol{\mathcal{E}})\boldsymbol{\Omega}$
$\boldsymbol{\Gamma}_t$	unmixing matrix at t	$\boldsymbol{\Gamma}_t \stackrel{\text{def}}{=} \boldsymbol{\Omega}_t^{-1}$
$\boldsymbol{\Omega}$	mixing matrix at $t = 0$	$\boldsymbol{\Omega} \stackrel{\text{def}}{=} \boldsymbol{\Omega}_0$
$\boldsymbol{\mathcal{E}}$	time-varying mixing factor	$p \times p$ matrix

Symbol	Meaning	Note
$\boldsymbol{\Lambda}_\tau$	autocovariance given stationarity	$p \times p$ matrix that depends only on lag τ and assumed to be diagonal
$\boldsymbol{W}$	affine tranformation matrix used in joint diagonalization	not confused with $\boldsymbol{\Gamma}$
$\boldsymbol{R}_\tau = \boldsymbol{\Omega}\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}'$	partially mixed autocovariance	$p \times p$, a short-hand notation
$\boldsymbol{K}^{(p,p)}$	commutation matrix	$p^2 \times p^2$
$t = 0, 1, 2, \ldots, T$	index of time	use in subscript. $t = 1,\ t = T$ are the first and last observation in the time series correspondingly
$\tau \in \{0\} \bigcup L$	pre-selected time lag	An integer. 0 is included for covenience
$L = \{\tau_1, \tau_2, \ldots, \tau_l\}$	set of pre-selected lags	l-item set of positive integers
$\mathcal{C}$	set of permutation matrices	all real-valued matrices that contain exactly 1 non-zero element in each row
$\boldsymbol{C}$	a permutation matrix	$\boldsymbol{C} \in \mathcal{C}$
Operator $\boldsymbol{A}'$	matrix transpose of $\boldsymbol{A}$	
Subscript $\boldsymbol{A}_t$	can be matrix or t-th column vector	representing A at time t
Function vec($\boldsymbol{A}$)	vectorization (by-column) of $\boldsymbol{A}$	
Operator $\otimes$	Kronecker product	
Index $\boldsymbol{A}[i, j]$	value at the i-th row j-th column of matrix $\boldsymbol{A}$	

2 Blind Source Separation

Blind Source Separation (BSS) assumes that an individual source signal $\boldsymbol{z} = (\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_p)'$ is mixed by a $p \times p$ matrix $\boldsymbol{\Omega}$ and thus produces the mixture $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_p)'$, where the signals themselves can be multidimensional. The mixing mechanism shall satisfy $\boldsymbol{x} = \boldsymbol{\mu} + \boldsymbol{\Omega}\boldsymbol{z}$, where $\boldsymbol{\mu}$ stands for a p-variate static location parameter, usually the mean value (Belouchrani et al., 1997). Without loss of generality, it can be further assumed that $\boldsymbol{x}$ embeds the zero-mean property, which is always achievable through subtracting the mean from the mixture, leading to further simplified mathematical representation.

Relying on certain (assumed) property/properties in the mixing mechanism, there exist different approaches to solve the BSS problem in terms of identifying the mixing matrix and source signal series. Independent Component Analysis (ICA) is perhaps the most well-known methodology to tackle the BSS problem, which was introduced by Hérault and Ans (1984) and became widely popular in early 1990s. Since Central Limit Theorem suggests that sufficiently many independent signals together approximate the Gaussian distribution, ICA finds the independent components by maximizing the non-Gaussianity. Kurtosis and negative entropy (negentropy) are among the best measures for non-Gassianity (Comon, 1994; Hyvärinen & Oja, 2000; Tharwat, 2018). Another common BSS solution, Second Order Blind Identification (SOBI, Belouchrani et al., 1997) is based on autocovariance, which is conceptualized by the theorem that diagonal autocovariance matrices suggest uncorrelated components. Later sections will elaborate SOBI in further detail. SOBI differs from ICA mainly from two aspects: (1) ICA uses fourth-order statistics like kurtosis while SOBI uses second-order statistics; (2) SOBI seeks to recover uncorrelated components while ICA recovers independent components.

Albeit mathematical unnecessity, $\boldsymbol{\Omega}$ is defined as a full-rank $p \times p$ matrix. A decrease in dimension of $\boldsymbol{\Omega}$ will allow fewer signal series (less-than-p-variate time series) generated from the same mixture $\boldsymbol{x}$. However, it is not necessary due to the fact that the outcome carries real-world information, and the usefulness of a series would be more properly determined with the support of further evidence from both the outcome and source. Consequently, it is a justified choice to force the mixing matrix to be of full-rank.

The ultimate goal of BSS is to restore the source signals based on the observed data, and the above model assures that finding either the mixing matrix $\boldsymbol{\Omega}$ or the unmixing matrix $\boldsymbol{\Gamma} = \boldsymbol{\Omega}^{-1}$ suffices, where the full-rank assumption guarantees the existence of inverse matrix. For notation simplification and clarity, the following paragraphs shall only use $\boldsymbol{\Omega}$.

2.1 Ambiguities and Assumptions

The BSS model underlies the impossibility to identify $\boldsymbol{\Omega}$ and $\boldsymbol{z}$ unambiguously because of merely one known item, and thus BSS solution tends to incur (Belouchrani et al., 1997),

- Permutation ambiguity: the a-th series of signal may be confused into b-th series in the restored one; however, they are always 1-1 correspondent after sign-correction. The permutation can also include sign ambiguity.
- Scale ambiguity: restored signals can be scaled since for any scale constant $||a|| : \ \boldsymbol{x} \equiv \big(\boldsymbol{\Omega}||a||^{-1}\big)\big(||a||\boldsymbol{z}\big)$. The scale constant is usually unknown.

The ambiguity can be summarised mathematically as $\boldsymbol{\Gamma} = \boldsymbol{C}\boldsymbol{\Omega}^{-1}$, where $\boldsymbol{C} \in \mathcal{C}$ is a set of matrices that contain exactly one non-zero element in each row and column.

Figure 2 illustrates the ambiguity. The observed source is a 4-dimensional signal combining auto-regressive, moving average, sinusoidal and Electrocardiogram ECG-like time series. Comparing two plots, source series 2 is clearly mapped into series 1 in restored signals, while series 4 become series 2. The scale of the y-axis intimates the scale ambiguity, though the shape and waveform are highly analogous.

2.2 Stationary Time-Series Source Separation Using Autocovariance Matrices

Focusing on the autocovariance matrices, that is, second order statistics, the Second Order Source separation (SOS, Belouchrani et al., 1997; Miettinen et al., 2016) model seeks to extract original source signals based on the property of uncorrelatedness. Meanwhile, the source signals are assumed to be weakly stationary, implying that the autocovariances vary only on the lag τ but not on the time point t, which are mathematically expressed as $\mathbb{E}(\boldsymbol{z}_t\boldsymbol{z}'_{t+\tau})$ being invariant given τ for all t. The SOS model states the discrete p-variate stochastic process $(\boldsymbol{z}_t)_{t=0,\pm1,\pm2,\dots}$ as (unobservable) source signals and the (observable) mixture $(\boldsymbol{x}_t)_{t=0,\pm1,\pm2,\dots}$ such that,

$$\begin{aligned} &\boldsymbol{x}_t = \boldsymbol{\Omega}\boldsymbol{z}_t, \text{ where } \boldsymbol{z}_t \text{ satisfies} \\ &\qquad (A1)\ \mathbb{E}(\boldsymbol{z}_t) = \boldsymbol{0} \\ &\qquad (A2)\ \mathbb{E}(\boldsymbol{z}_t\boldsymbol{z}'_t) = \boldsymbol{I}_p \\ &\qquad (A3)\ \mathbb{E}(\boldsymbol{z}_t\boldsymbol{z}'_{t+\tau}) = \boldsymbol{\Lambda}_\tau \text{ diagonal for all } \tau = 1, 2, \dots \end{aligned} \tag{1}$$

(Miettinen et al., 2016). The condition $(A1)$ simplifies the model with pre-centered observation; $(A2)$ further restrains the source signals in unit scale,

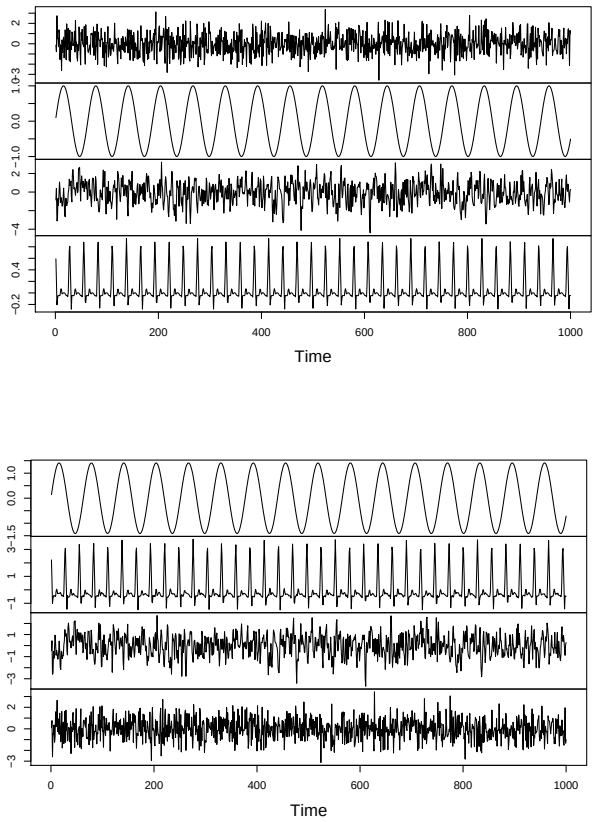

Figure 2: Observed (upper) vs. restored (lower) sources

solving the scale ambiguity; and ($A3$) ensures both stationarity and uncorrelatedness. Given realization of the stochastic process $(\boldsymbol{x}_1, \boldsymbol{x}_2, \dots)$ (a p-variate time-series), this semi-parametric model can be solved using sample autocovariance matrices by joint optimization for the diagonal properties in ($A3$) under the restriction of ($A2$) in (1), which yield to a unique constrained optimization that can be solved using Lagrange method after proper whitening and/or normalization procedures (Miettinen et al., 2016).

The Second Order Blind Identification (SOBI) solves the BSS problem using second-order statistics when the signal and mixing mechanism obey the SOS model (1). The major second-order blind source separation approaches include Algorithm for Multiple Unknown Signals Extraction (AMUSE, Tong et al., 1990) and Second-Order Blind Identification (SOBI, Belouchrani et al., 1997). Nordhausen (2014) expanded the algorithm to non-stationary time series using locally stationary intervals and further robustifying the method with average spatial-sign autocovariances on such intervals.

3 Time-Varying Second-Order Model Formulation and Yeredor's Solution

Time-varying Second-Order Source Separation (TV-SOS, Yeredor, 2003) refers to the existence of subtle change in mixing matrix $\boldsymbol{\Omega}$. Within the SOS model (1), linear time variation can be represented as $\boldsymbol{\Omega}_t = (\boldsymbol{I} + t\boldsymbol{\mathcal{E}})\boldsymbol{\Omega}_0$, which is clearly non-constant despite $\boldsymbol{\mathcal{E}}$ and $\boldsymbol{\Omega}_0$ are. The initial mixing $\boldsymbol{\Omega}_0$ is the $p \times p$ mixing matrix at time $t = 0$; the time-varying factor $\boldsymbol{\mathcal{E}}$ is another $p \times p$ matrix that measures the scale of linear variation in mixing matrix over the change of time. In addition to this linear variation, other time-varying structures include periodical (Weisman & Yeredor, 2006), geometric curved (Kaftory & Zeevi, 2007), etc. Figure 3 illustrates the difference between an ordinary time-invariant mixture and linearly time-varying one. It can be discovered that the time-varying mixture has trends and it does not demonstrate the stationary property. In fact, the introduction of time-varying factor invalids the stationary property in the observation $\boldsymbol{x}$ almost surely even though the source signals are stationary. This is because $\boldsymbol{\mathcal{E}}$ changes the scale (second-order statistics) over time t. Nevertheless, the aforementioned SOBI and the upcoming LTV-SOBI algorithms do not require such property on the observations; only the source signals are bind to stationarity.

3.1 TV-SOS Model and Assumptions

TV-SOS model serves as an extension to the SOS model (1) and is also a special case of general-TV-SOS where the time-dependent variation is assumed to be linear. Focusing on the realization of stochastic processes, an observable p-variate time series in TV-SOS satisfies

$$\begin{aligned} \boldsymbol{x}_t = (\boldsymbol{I} + t\boldsymbol{\mathcal{E}})\boldsymbol{\Omega}_0\boldsymbol{z}_t, \text{ where } \boldsymbol{z}_t &\text{satisfies} \\ (B1)\ \mathbb{E}(\boldsymbol{z}_t) &= \boldsymbol{0} \\ (B2)\ \mathrm{Cov}(\boldsymbol{z}_t) &= \boldsymbol{I} \\ (B3)\ \mathrm{Cov}(\boldsymbol{z}_t, \boldsymbol{z}'_{t+\tau}) &= \boldsymbol{\Lambda}_\tau \text{ diagonal for all } \tau = 1, 2, \ldots \\ (B4*)\ \boldsymbol{\mathcal{E}} &<< \boldsymbol{I} \end{aligned} \tag{2}$$

Similar to SOS, the first assumption $(B1)$ is non-restrictive and achievable through data transformation; $(B2)$ is required to tackle the ambiguity of BSS, while $(B3)$ states both stationary and uncorrelated characteristics in source signals. Further, the diagonal elements in $\boldsymbol{\Lambda}_\tau$ have to be different to ensure identifiability of each series. The final and optional assumption $(B4)$ ensures that the change of mixing is rather slow so that the meaningfulness of BSS is not greatly compromised, and this assumption can simplify BSS process in some algorithms (Yeredor, 2003). It should be noted that the model assumes uncorrelatedness

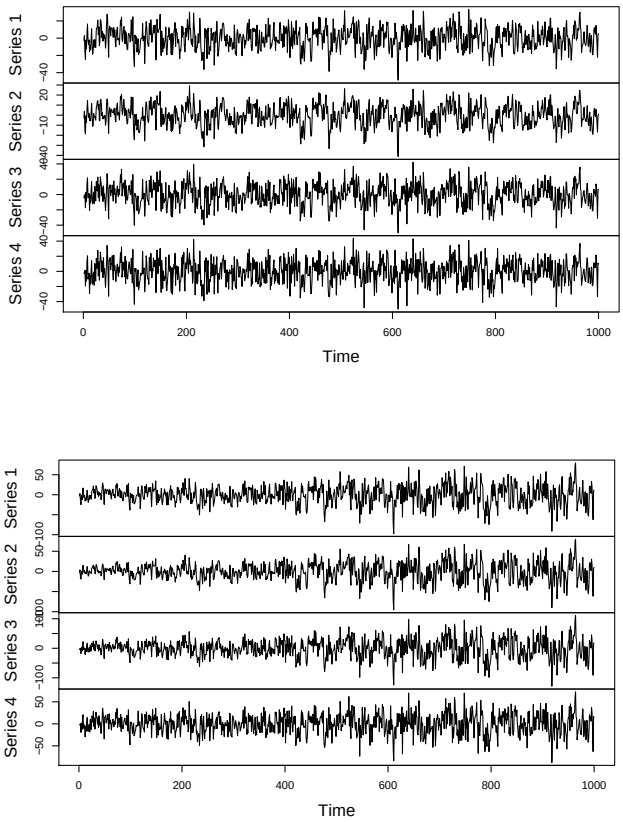

Figure 3: Two types of 4-dimensional signal mixture example: ordinary mixing (upper) and time-varying mixing (lower).

instead of independence in pre-centered source signals, formally, the Pearson sample correlation between different series is always 0, or $\mathrm{Cov}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \mathbf{0}$ for all $i \neq j$. It is a relatively less-restrictive condition as independence implies uncorrelatedness, while the opposite is not true in general (e.g. Papoulis & Pillai, 2002).

3.2 Yeredor's TV-SOBI Algorithm

TV-SOBI is the original algorithm provided by Yeredor (2003) that solves the above TV-SOS model. While the SOS model can be identified by obtaining one single matrix $\boldsymbol{\Omega}$, TV-SOS demands at least one more matrix to be effectively estimated, the linear time-varying mixing factor $\boldsymbol{\mathcal{E}}$ in addition to the initial mixing matrix $\boldsymbol{\Omega}_0$. Similar to model fitting in time series analysis, a prerequisite is that the desired lags must be chosen beforehand based on the data characteristics (for example with the help of `acf` function; a proper choice of

lags could be challenging, and this topic would not be elaborated in this thesis), and suppose $L = \{\tau_1, \tau_2, \ldots, \tau_l\}$ is the set of pre-defined lags. For convenience, let $\tau \in \{0\} \bigcup L$. Yeredor's algorithm first finds the approximate three-item expression of autocovariances as,

$$\begin{aligned}
\mathbb{E}(\boldsymbol{x}_t \boldsymbol{x}'_{t+\tau}) &= \mathbb{E}[(\boldsymbol{I} + t\boldsymbol{\mathcal{E}})\boldsymbol{\Omega}_0 \boldsymbol{z}_t \ \boldsymbol{z}'_{t+\tau} \boldsymbol{\Omega}'_0 [\boldsymbol{I} + (t+\tau)\boldsymbol{\mathcal{E}}]'] \\
&= \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 + t(\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 + \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}') + t^2 (\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}') \\
&\quad + \tau(\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0) + t\tau(\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}') \\
&= \underline{\boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0} + t(\underline{\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 + \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}'}) \\
&\quad + t(t+\tau)(\underline{\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}'}) + \tau(\underline{\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0}) \\
&\approx \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 + t(\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 + \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}') + t^2 (\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}') \\
&:= \boldsymbol{R}_\tau^{(1)} + t\, \boldsymbol{R}_\tau^{(2)} + t^2\, \boldsymbol{R}_\tau^{(3)}
\end{aligned} \quad , \quad (3)$$

where the shorthand notations are defined as $\boldsymbol{R}_\tau^{(1)} = \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0$, $\boldsymbol{R}_\tau^{(2)} = \boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 + \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}'$ and $\boldsymbol{R}_\tau^{(3)} = \boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}'$; those three items are the sample autocovariance decomposition of the observed mixture. Since $\boldsymbol{\mathcal{E}}$ is assumed to be rather small in quantity, the items $\tau(\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0)$ and $t\tau(\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}')$ would also be insignificant due to their scale transformation by $\boldsymbol{\mathcal{E}}$; Yeredor hence argues that they are negligible.

Then, Yeredor tried to estimate the $p \times p$ matrices of $\boldsymbol{R}_\tau^{(1)}$, $\boldsymbol{R}_\tau^{(2)}$ and $\boldsymbol{R}_\tau^{(3)}$ through a linear least-squares model. The model is achieved by casting matrix equation (3) to element-wise real-valued equations of

$$\begin{bmatrix} \boldsymbol{x}_1 \boldsymbol{x}'_{1+\tau}\ [i,j] \\ \boldsymbol{x}_2 \boldsymbol{x}'_{2+\tau}\ [i,j] \\ \vdots \\ \boldsymbol{x}_{T-\tau} \boldsymbol{x}'_T\ [i,j] \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1^2 \\ 1 & 2 & 2^2 \\ \vdots & \vdots & \vdots \\ 1 & T-\tau & T^2 \end{bmatrix} \begin{bmatrix} \boldsymbol{R}_\tau^{(1)}[i,j] \\ \boldsymbol{R}_\tau^{(2)}[i,j] \\ \boldsymbol{R}_\tau^{(3)}[i,j] \end{bmatrix} + \text{residuals.} \quad (4)$$

The LS-estimation leads to

$$\begin{aligned}
\begin{bmatrix} \widehat{\boldsymbol{R}}_\tau^{(1)}[i,j] \\ \widehat{\boldsymbol{R}}_\tau^{(2)}[i,j] \\ \widehat{\boldsymbol{R}}_\tau^{(3)}[i,j] \end{bmatrix} = &\left(\begin{bmatrix} 1 & 1 & 1^2 \\ 1 & 2 & 2^2 \\ \vdots & \vdots & \vdots \\ 1 & T-\tau & T^2 \end{bmatrix}' \begin{bmatrix} 1 & 1 & 1^2 \\ 1 & 2 & 2^2 \\ \vdots & \vdots & \vdots \\ 1 & T-\tau & T^2 \end{bmatrix} \right)^{-1} \\
&\begin{bmatrix} 1 & 1 & 1^2 \\ 1 & 2 & 2^2 \\ \vdots & \vdots & \vdots \\ 1 & T-\tau & T^2 \end{bmatrix}' \begin{bmatrix} \boldsymbol{x}_1 \boldsymbol{x}'_{1+\tau}\ [i,j] \\ \boldsymbol{x}_2 \boldsymbol{x}'_{2+\tau}\ [i,j] \\ \vdots \\ \boldsymbol{x}_{T-\tau} \boldsymbol{x}'_T\ [i,j] \end{bmatrix}.
\end{aligned} \quad (5)$$

Yeredor (2003) proposed a practical approach to optimize the best solution for

$\boldsymbol{\Omega}_0$ and $\boldsymbol{\mathcal{E}}$. Denote the affine transformation (aka. whitening) matrix $\boldsymbol{W} = \left(\widehat{\boldsymbol{R}}_0^{(1)}\right)^{-\frac{1}{2}}$. The idea is then to apply sequential Jacobi rotations to optimize

$$\min_{V,\Lambda_{\tau_1},\ldots,\Lambda_{\tau_l}} \left(\sum_{\tau=\tau_1}^{\tau_l} ||\boldsymbol{W}\widehat{\boldsymbol{R}}_\tau^{(1)}\boldsymbol{W}' - \boldsymbol{V}\boldsymbol{\Lambda}_\tau\boldsymbol{V}'||^2 \right), \tag{6}$$

where $\Lambda_{\tau_1}, \ldots, \Lambda_{\tau_l}$ are diagonal. This diagonal property is sufficient for the optimization algrithm, and the values of $\Lambda_{\tau_1}, \ldots, \Lambda_{\tau_l}$ will become available right after the optimization. This procedure is similar to the joint diagnolization procedure that will be detailed in section 4.2. The $\boldsymbol{\mathcal{E}}$ can be found through optimization,

$$\min_{\boldsymbol{\mathcal{E}}} \left(\sum_{\tau=\tau_1}^{\tau_l} ||\widehat{\boldsymbol{R}}_\tau^{(2)} - \boldsymbol{\mathcal{E}}\widehat{\boldsymbol{R}}_\tau^{(1)} - \widehat{\boldsymbol{R}}_\tau^{(1)}\boldsymbol{\mathcal{E}}'||^2 \right). \tag{7}$$

Finally, Yeredor's TV-SOBI concludes with $\widehat{\boldsymbol{\Omega}}_0 = \boldsymbol{W}^{-1}\boldsymbol{V}$ and $\widehat{\boldsymbol{\mathcal{E}}}$. It can be noticed in the optimization steps that the value of $\boldsymbol{R}_\tau^{(3)}$ is not used. In fact, Yeredor even provided an alternative that excludes it from (3). Nonetheless, notice that $\boldsymbol{R}_\tau^{(3)}$ is closely associating with $\boldsymbol{R}_\tau^{(1)}$ and $\boldsymbol{R}_\tau^{(2)}$ as shown in equation (4), suggesting that inclusion or exclusion would truly impact the output of TV-SOBI. It is theoretically possible to include the expression of $\boldsymbol{R}_\tau^{(3)}$ in optimization procedures, but it would be mathematically too complicated and computationally too costly to find a solution, and the relatively naive approach described above suffices to solve the TV-SOS problem.

4 Algorithm for Linearly Time-Varying Second Order Blind Identification

This section introduces the new Linearly Time-Varying Second Order Blind Identification (LTV-SOBI) algorithm pursuing improved mathematical accuracy given TV-SOS model (2). This algrorithm is developed from Yeredor's (2003) original TV-SOBI algorithm, but does not require the assumption $(B4)$ in model (2). LTV-SOBI mainly includes three steps which use sample autocovariance matrices and applying joint diagonalization after applicable decomposition. Various matrix operations are heavily utilized in all steps.

4.1 Decomposition of Autocovariance Structure

Being a second-order approach, the LTV-SOBI algorithm is based on the sample autocovariances of pre-centered observation. As previously demonstrated in equation (3), the autocovariances are

$$\begin{aligned}\mathrm{Cov}(\boldsymbol{x}_t, \boldsymbol{x}'_{t+\tau}) &= \mathbb{E}(\boldsymbol{x}_t \boldsymbol{x}'_{t+\tau}) \\ &= \mathbb{E}[(\boldsymbol{I} + t\boldsymbol{\mathcal{E}})\boldsymbol{\Omega}_0 \boldsymbol{z}_t \ \boldsymbol{z}'_{t+\tau} \boldsymbol{\Omega}'_0 [\boldsymbol{I} + (t+\tau)\boldsymbol{\mathcal{E}}]'] \\ &= \underline{\boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0} + t(\underline{\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 + \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}'}) \\ &\quad + t(t+\tau)(\underline{\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 \boldsymbol{\mathcal{E}}'}) + \tau(\underline{\boldsymbol{\mathcal{E}} \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0})\end{aligned} \quad , \tag{8}$$

where $\tau \in \{0\} \bigcup L$, and $L = \{\tau_1, \tau_2, \ldots, \tau_l\}$ are the set of pre-defined lags. The assumption of stationarity is demonstrated as $\boldsymbol{\Lambda}_\tau = \mathbb{E}(\boldsymbol{z}_t \boldsymbol{z}'_{t+\tau})$ being invariant for all $\tau \in L$ and unreliant on any $t = 1, 2, \ldots, T - \tau$. Instead of Yeredor's approximation methodology, the proposed LTV-SOBI algorithm seeks to improve accuracy by preserving all terms in autocovariance matrices.

After decomposition, the observation is summarized into $l + 1$ matrices of autocovariance. The number reflects the quantity of pre-selected lags plus the one for the sample covariance matrix $\mathbb{E}(\boldsymbol{x}_t \boldsymbol{x}'_t)$. Consider element-wise equivalence, for $i, j = 1, 2, \ldots, p$, the autocovariance structure in (8) is equivalent to $\boldsymbol{S}_\tau = \boldsymbol{H}^*_\tau \boldsymbol{\beta}^*_\tau$ defined as,

$$\underbrace{\begin{bmatrix} \boldsymbol{x}_1 \boldsymbol{x}'_{1+\tau}\ [i,j] \\ \boldsymbol{x}_2 \boldsymbol{x}'_{2+\tau}\ [i,j] \\ \vdots \\ \boldsymbol{x}_{T-\tau} \boldsymbol{x}'_T\ [i,j] \end{bmatrix}}_{:=\boldsymbol{S}_\tau} = \underbrace{\begin{bmatrix} 1 & 1 & 1(1+\tau) & \tau \\ 1 & 2 & 2(2+\tau) & \tau \\ \vdots & \vdots & \vdots & \vdots \\ 1 & T-\tau & (T-\tau)T & \tau \end{bmatrix}}_{:=\boldsymbol{H}^*_\tau} \underbrace{\begin{bmatrix} (\boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0)\ [i,j] \\ (\boldsymbol{\mathcal{E}}\, \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0 + \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0\, \boldsymbol{\mathcal{E}}')\ [i,j] \\ (\boldsymbol{\mathcal{E}}\, \boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0\, \boldsymbol{\mathcal{E}}')\ [i,j] \\ (\boldsymbol{\Omega}_0 \boldsymbol{\Lambda}_\tau \boldsymbol{\Omega}'_0\, \boldsymbol{\mathcal{E}}')\ [i,j] \end{bmatrix}}_{:=\boldsymbol{\beta}^*_\tau} . \tag{9}$$

As the p-vector $\boldsymbol{x}$ is the only available observation, it seems that element-wise linear regression is an unpretentious solution to decompose the autocovariances

into the structure as in (8). The challenge of equation (9) is that the last column of $\boldsymbol{H}_\tau^*$ is constant given τ and fortunately, the modified form in (10) can comfortably tackle it, where the fourth element in $\boldsymbol{\beta}_\tau^*$ are merged into the first row within the same matrix.

$$\underbrace{\begin{bmatrix} \boldsymbol{x}_1\boldsymbol{x}'_{1+\tau}\ [i,j] \\ \boldsymbol{x}_2\boldsymbol{x}'_{2+\tau}\ [i,j] \\ \vdots \\ \boldsymbol{x}_{T-\tau}\boldsymbol{x}'_{T}\ [i,j] \end{bmatrix}}_{:=\boldsymbol{S}_\tau} = \underbrace{\begin{bmatrix} 1 & 1 & 1(1+\tau) \\ 1 & 2 & 2(2+\tau) \\ \vdots & \vdots & \vdots \\ 1 & T-\tau & (T-\tau)T \end{bmatrix}}_{:=\boldsymbol{H}_\tau} \underbrace{\begin{bmatrix} (\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0' + \tau\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}')\ [i,j] \\ (\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0' + \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}')\ [i,j] \\ (\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}')\ [i,j] \end{bmatrix}}_{:=\boldsymbol{\beta}_\tau}. \tag{10}$$

Therefore, a proper linear regression can be applied for each $i,j = 1,2,\ldots,p$ and each $\tau \in \{0\} \cup L$ in the form of $\boldsymbol{S}_\tau[i,j] = \boldsymbol{H}_\tau[i,j]\boldsymbol{\beta}_\tau[i,j]$, where $\boldsymbol{S}_\tau$ are known and $\boldsymbol{H}_\tau$ are design matrices as expressed in (10); ultimately, matrices of $\boldsymbol{\beta}_\tau$ are fully estimated in an element-wise manner after looping. For better efficiency, the vectorization form as follows is generally recommended as the alternative for looping over each i,j.

$$\underbrace{\begin{bmatrix} \mathrm{vec}(\boldsymbol{x}_1\boldsymbol{x}'_{1+\tau}) \\ \mathrm{vec}(\boldsymbol{x}_2\boldsymbol{x}'_{2+\tau}) \\ \vdots \\ \mathrm{vec}(\boldsymbol{x}_{T-\tau}\boldsymbol{x}'_{T}) \end{bmatrix}}_{:=\mathrm{vec}(\boldsymbol{S}_\tau)} = \underbrace{\begin{bmatrix} 1 & 1 & 1(1+\tau) \\ 1 & 2 & 2(2+\tau) \\ \vdots & \vdots & \vdots \\ 1 & T-\tau & (T-\tau)T \end{bmatrix} \otimes \boldsymbol{I}_{p^2}}_{:=\boldsymbol{H}_\tau\otimes\boldsymbol{I}_{p^2}} \underbrace{\begin{bmatrix} \mathrm{vec}(\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0' + \tau\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}') \\ \mathrm{vec}(\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0' + \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}') \\ \mathrm{vec}(\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}') \end{bmatrix}}_{:=\mathrm{vec}(\boldsymbol{\beta}_\tau)}. \tag{11}$$

The classical linear model theory suggests the unique maximum likelihood estimator of $\boldsymbol{\beta}_\tau$ in (11) to be $\mathrm{vec}(\widehat{\boldsymbol{\beta}}_\tau) = \left[(\mathbf{H}_\tau \otimes \boldsymbol{I}_{p^2})'(\mathbf{H}_\tau \otimes \boldsymbol{I}_{p^2})\right]^{-1}(\mathbf{H}_\tau \otimes \boldsymbol{I}_{p^2})'\mathrm{vec}(\mathbf{S}_\tau)$. The estimator coincides with least squared ones (e.g. Myers & Myers, 1990), and the inverse of vectorization is straightforward. In conclusion, the first step of LTV-SOBI decomposes autocovariance matrices into $3(\tau+1)$ matrices as defined in (12) that contains second-order information on the observation, where $\boldsymbol{\beta}_{2,\tau}$ can be viewed as partially time-varying autocovariance, and $\boldsymbol{\beta}_{3,\tau}$ as time varying autocovariances. For all $\tau \in \{0\} \cup L$, it becomes fully estimated that

$$\begin{cases} \widehat{\boldsymbol{\beta}}_{1,\tau} = \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0' + \tau\,\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}' \\ \widehat{\boldsymbol{\beta}}_{2,\tau} = \boldsymbol{\mathcal{E}}\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0' + \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}' \\ \widehat{\boldsymbol{\beta}}_{3,\tau} = \boldsymbol{\mathcal{E}}\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\ \boldsymbol{\Omega}_0'\boldsymbol{\mathcal{E}}' \end{cases}. \tag{12}$$

4.2 Finding $\boldsymbol{\Omega}_0$ with Approximate Joint Diagnolization

In TV-SOS model (2), the mathematical properties of $\widehat{\boldsymbol{\beta}}_{1,\tau}$, $\widehat{\boldsymbol{\beta}}_{2,\tau}$ and $\widehat{\boldsymbol{\beta}}_{3,\tau}$ are not particular except for the latter two's symmetry. Hence, this step tries to further process the results in (12) and then match the assumptions, especially the diagonal property. Since $\widehat{\boldsymbol{\beta}}_{1,\tau}+\widehat{\boldsymbol{\beta}}'_{1,\tau} = \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\ \boldsymbol{\Omega}'_0+\tau\,\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}'_0\,\boldsymbol{\mathcal{E}}'+\boldsymbol{\Omega}\ \boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}'_0+\tau\,\boldsymbol{\mathcal{E}}\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}'_0 = 2\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}'_0+\tau\,\widehat{\boldsymbol{\beta}}_{2,l}$, it is possible to find the representing formula as in (13), where $\boldsymbol{R}_\tau$ is a short-hand notation for $\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\ \boldsymbol{\Omega}'_0$, that is,

$$\boldsymbol{R}_\tau \overset{\text{def}}{=} \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\ \boldsymbol{\Omega}'_0 = \frac{1}{2}\left(\widehat{\boldsymbol{\beta}}_{1,\tau}+\widehat{\boldsymbol{\beta}}'_{1,\tau}-\tau\widehat{\boldsymbol{\beta}}_{2,\tau}\right) \text{ for all } \tau\in\{0\}\cup L\ . \tag{13}$$

Despite both $\boldsymbol{\Omega}_0$ and $\boldsymbol{\Lambda}_\tau$ are unknown, the items $\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}'_0$ are analytic using approximate joint diagonalization (JADE, Clarkson & Jennrich, 1988), which is possible because of diagonal property in $\boldsymbol{\Lambda}_\tau$ (the uncorrelatedness assumption in sources) (Belouchrani et al., 1997; Li & Zhang, 2007; Miettinen et al., 2017; Yeredor, 2002). Unlike the sample covariance matrix used in SOBI, LTV-SOBI has to use $\boldsymbol{R}_0$ due to the complex mixing mechanism. The JADE task in this step can be formulated as optimization for $\boldsymbol{\Omega}_0$ using known matrices $\boldsymbol{R}_{\tau_1},\ \boldsymbol{R}_{\tau_1},\ldots,\boldsymbol{R}_{\tau_l}$ with the conditions set in (14).

$$\begin{cases} \boldsymbol{R}_0 = \boldsymbol{\Omega}_0\boldsymbol{\Omega}'_0 & \Leftrightarrow\ \boldsymbol{\Omega}_0^{-1}\boldsymbol{R}_0{\boldsymbol{\Omega}_0^{-1}}' = \boldsymbol{I} \quad \text{(whitening restriction)} \\ \boldsymbol{R}_{\tau_1} = \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_{\tau_1}\boldsymbol{\Omega}'_0 & \Leftrightarrow\ \boldsymbol{\Omega}_0^{-1}\boldsymbol{R}_{\tau_1}{\boldsymbol{\Omega}_0^{-1}}' = \boldsymbol{\Lambda}_{\tau_1}, \ \text{ and } \boldsymbol{\Lambda}_1 \text{ is diagonal} \\ \boldsymbol{R}_{\tau_2} = \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_{\tau_2}\boldsymbol{\Omega}'_0 & \Leftrightarrow\ \boldsymbol{\Omega}_0^{-1}\boldsymbol{R}_{\tau_2}{\boldsymbol{\Omega}_0^{-1}}' = \boldsymbol{\Lambda}_{\tau_2}, \ \text{ and } \boldsymbol{\Lambda}_2 \text{ is diagonal} \\ \vdots & \\ \boldsymbol{R}_{\tau_l} = \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_{\tau_l}\boldsymbol{\Omega}'_0 & \Leftrightarrow\ \boldsymbol{\Omega}_0^{-1}\boldsymbol{R}_{\tau_l}{\boldsymbol{\Omega}_0^{-1}}' = \boldsymbol{\Lambda}_{\tau_l}, \ \text{ and } \boldsymbol{\Lambda}_3 \text{ is diagonal} \end{cases} . \tag{14}$$

It is not possible to exactly diagonolize all $\boldsymbol{\Lambda}_{\tau_1},\ \boldsymbol{\Lambda}_{\tau_2},\ldots,\boldsymbol{\Lambda}_{\tau_l}$ simultaneously. However, JADE is shown to provide good estimates for $\boldsymbol{\Lambda}_{\tau_1},\boldsymbol{\Lambda}_{\tau_2},\ldots$ (Miettinen et al., 2016). This method is based on the idea of minimizing the sum of off-diagonal elements $\sum_{\tau=\tau_1}^{\tau_l}||\text{off}(\boldsymbol{\Lambda}_\tau)||^2$. The established JADE algorithm by Miettinen et al. (2016) efficiently finds an orthogonal matrix $\boldsymbol{V}$ such that $\sum_{\tau=\tau_1}^{\tau_l}||\text{off}(\boldsymbol{V}\boldsymbol{R}_\tau\boldsymbol{V}')||^2$ is minimized given a set of matrices $\boldsymbol{R}_{\tau_1},\ldots,\boldsymbol{R}_{\tau_l}$. To achieve the goal of finding non-orthogonal matrix $\boldsymbol{\Omega}_0$, the LTV-SOBI algorithm requires a whitening step, such that for all $\tau\in L\setminus\{0\}$, $\boldsymbol{R}_\tau$ is whitened by $\boldsymbol{R}_0^{-1/2}$; that is the whitened items $\tilde{\boldsymbol{R}}_\tau = \boldsymbol{R}_0^{-1/2}\ \boldsymbol{R}_\tau\ {\boldsymbol{R}_0^{-1/2}}'$. Joint diagonalization procedure can then be applied to find an orthogonal $p\times p$ matrix $\boldsymbol{V}$ that maximize the diagonality of $\boldsymbol{\Lambda}_{\tau_1},\ \boldsymbol{\Lambda}_{\tau_2},\ldots,\boldsymbol{\Lambda}_{\tau_l}$ that maximizes

$$\begin{cases} \boldsymbol{V}\tilde{\boldsymbol{R}}_{\tau_1}\boldsymbol{V}' &= \boldsymbol{\Lambda}_{\tau_1} \\ \boldsymbol{V}\tilde{\boldsymbol{R}}_{\tau_2}\boldsymbol{V}' &= \boldsymbol{\Lambda}_{\tau_2} \\ & \vdots \\ \boldsymbol{V}\tilde{\boldsymbol{R}}_{\tau_l}\boldsymbol{V}' &= \boldsymbol{\Lambda}_{\tau_L} \end{cases} . \tag{15}$$

Joint diagonalization steps are concluded with unwhitening, that is, $\boldsymbol{\Omega}_0$ is found by $\boldsymbol{\Omega}_0 = \boldsymbol{R}_0^{1/2}\boldsymbol{V}$.

The whitening requires $\boldsymbol{R}_0$ to be positive semi-definite, and mathematically, it is also embedded with this property as it represents the unobservable sample covariance matrix of the true signal. However, $\boldsymbol{R}_0$ can only be found through the autocovariance decomposition result (12), which is achieved by a linear estimator from equation (10), and naturally, the estimator is not guaranteed to be positive semi-definite due to residuals and linear fitting. Practically, in such a case, the Nearest Positive Definite Matrix (nearPD) algorithm by Bates and Martin Maechler (2018) is appointed to replace the problematic $\boldsymbol{R}_0$ with its nearest positive definite neighbor, where the nearest distance is measured by Frobenius norm. Details about correcting non-positive semi-definite matrix were proposed by Knol and ten Berge (1989) and have been further developed by Cheng and Higham (1998). Meanwhile, it should be noted that unlike in SOBI, the whitening is achievable in observed data, it is not possible to conduct similar whitening on the observation for LTV-SOBI since the sample covariance used in whitening is estimated after autocovariance decomposition.

After the above steps, the following matrices are entirely solved

$$\begin{cases} \widehat{\boldsymbol{\Omega}}_0 \\ \widehat{\boldsymbol{\Lambda}}_{\tau_1},\ \widehat{\boldsymbol{\Lambda}}_{\tau_2}, \ldots, \widehat{\boldsymbol{\Lambda}}_{\tau_l} \end{cases} . \tag{16}$$

4.3 Finding $\boldsymbol{\mathcal{E}}$ through $\widehat{\beta_2}$ and $\widehat{\Omega_0}$

Unlike ordinary SOBI, $\widehat{\boldsymbol{\Omega}}_0$ alone would not identify TV-SOS model, and thus LTV-SOBI procedure continues using outcomes from previous steps (16) and (12). Observing $\widehat{\boldsymbol{\beta}}_{2,\tau} \approx \boldsymbol{\mathcal{E}}\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0' + \widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0'\boldsymbol{\mathcal{E}}'$ and with the help of commutation matrix $\boldsymbol{K}^{(m,n)}$ that ensures $\boldsymbol{K}^{(m,n)}\mathrm{vec}(\boldsymbol{A}) = \mathrm{vec}(\boldsymbol{A}')$ for any $m \times n$ matrix $\boldsymbol{A}$, the vectorization leads to

$$
\begin{aligned}
\operatorname{vec}(\widehat{\boldsymbol{\beta}}_{2,\tau}) &= \operatorname{vec}(\boldsymbol{\mathcal{E}}\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'}) + \operatorname{vec}(\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'}\boldsymbol{\mathcal{E}}') \\
&= \operatorname{vec}(\boldsymbol{I}\boldsymbol{\mathcal{E}}(\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'})) + \operatorname{vec}((\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'})\boldsymbol{\mathcal{E}}'\boldsymbol{I}) \\
&= \Big((\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'})' \otimes \boldsymbol{I}\Big)\operatorname{vec}(\boldsymbol{\mathcal{E}}) + \Big(\boldsymbol{I}' \otimes (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'})\Big)\operatorname{vec}(\boldsymbol{\mathcal{E}}') \\
&= \Big((\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'}) \otimes \boldsymbol{I}\Big)\operatorname{vec}(\boldsymbol{\mathcal{E}}) + \Big(\boldsymbol{I} \otimes (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'})\Big)\boldsymbol{K}^{(p,p)}\operatorname{vec}(\boldsymbol{\mathcal{E}}) \\
&= \Big((\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'}) \otimes \boldsymbol{I} + (\boldsymbol{I} \otimes (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_\tau\widehat{\boldsymbol{\Omega}}_0^{'}))\boldsymbol{K}^{(p,p)}\Big)\operatorname{vec}(\boldsymbol{\mathcal{E}})
\end{aligned} \quad . \quad (17)
$$

Stacking (row-binding) over all $\tau \in L$ will provide the equation with only one unknown matrix $\boldsymbol{\mathcal{E}}$, that is,

$$
\begin{bmatrix} \operatorname{vec}(\widehat{\boldsymbol{\beta}}_{2,\tau_1}) \\ \operatorname{vec}(\widehat{\boldsymbol{\beta}}_{2,\tau_2}) \\ \vdots \\ \operatorname{vec}(\widehat{\boldsymbol{\beta}}_{2,\tau_l}) \end{bmatrix} = \begin{bmatrix} (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_{\tau_1}\widehat{\boldsymbol{\Omega}}_0^{'}) \otimes \boldsymbol{I} + (\boldsymbol{I} \otimes (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_{\tau_1}\widehat{\boldsymbol{\Omega}}_0^{'}))\boldsymbol{K}^{(p,p)} \\ (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_{\tau_2}\widehat{\boldsymbol{\Omega}}_0^{'}) \otimes \boldsymbol{I} + (\boldsymbol{I} \otimes (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_{\tau_2}\widehat{\boldsymbol{\Omega}}_0^{'}))\boldsymbol{K}^{(p,p)} \\ \vdots \\ (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_{\tau_l}\widehat{\boldsymbol{\Omega}}_0^{'}) \otimes \boldsymbol{I} + (\boldsymbol{I} \otimes (\widehat{\boldsymbol{\Omega}}_0\widehat{\boldsymbol{\Lambda}}_{\tau_l}\widehat{\boldsymbol{\Omega}}_0^{'}))\boldsymbol{K}^{(p,p)} \end{bmatrix} \operatorname{vec}(\boldsymbol{\mathcal{E}}) \ . \quad (18)
$$

The solution of $\operatorname{vec}(\boldsymbol{\mathcal{E}})$ can be found directly through the matrix inverse.

4.4 Signal Restoration

The aforementioned three steps lead to closed-form estimates of parameters $\boldsymbol{\Omega}_0$ and $\boldsymbol{\mathcal{E}}$, and and the signals can be restored simply by using the inverse of the time-varying mixing. Yet, unlike the SOBI and most other BSS methodologies, the source signals would not be restored by a single matrix calculation. As the time index t still persists in equation (19), the restoration has to be conducted one-by-one. Luckily, the restoration procedure itself is straightforward and computation resource-friendly, as

$$
\widehat{\boldsymbol{z}}_t = \Big((\boldsymbol{I} + t\,\widehat{\boldsymbol{\mathcal{E}}})\widehat{\boldsymbol{\Omega}}_0\Big)^{-1}\boldsymbol{x}_t \ . \quad (19)
$$

4.5 Minor Alternatives for LTV-SOBI

Sections 4.1, 4.2 and 4.3 composed the three steps of LTV-SOBI algorithm, and after signal restoration, the TV-SOS problem is fully identified. It can also be perceived that in the decomposition of autocovariance matrices (step 1, section 4.1), the estimated $\widehat{\boldsymbol{\beta}}_{3,\tau}$ is never used in later steps within the proposed LTV-SOBI algorithm (note that it affects the estimation of other components). To

resolve such redundancy, one naive way is to avoid estimating it by replacing (11) with the approximate equation, and this alternative is referred as linear form LTV-SOBI, programmed in *R*-function `ltvsobi` as a binary optional parameter `quadratic = FALSE`. The naming of quadratic and linear form originates from the greatest power of t. The approximate equation is now

$$\begin{bmatrix} \mathrm{vec}(\boldsymbol{x}_1\boldsymbol{x}_{1+\tau}') \\ \mathrm{vec}(\boldsymbol{x}_2\boldsymbol{x}_{2+\tau}') \\ \vdots \\ \mathrm{vec}(\boldsymbol{x}_{T-\tau}\boldsymbol{x}_T') \end{bmatrix} \approx \begin{bmatrix} 1 & 1 & 1(1+\tau) \\ 1 & 2 & 2(2+\tau) \\ \vdots & \vdots & \vdots \\ 1 & T-\tau & (T-\tau)T \end{bmatrix} \otimes \boldsymbol{I}_{p^2} \begin{bmatrix} \mathrm{vec}(\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0' + \tau\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}') \\ \mathrm{vec}(\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0' + \boldsymbol{\Omega}_0\boldsymbol{\Lambda}_\tau\boldsymbol{\Omega}_0'\,\boldsymbol{\mathcal{E}}') \end{bmatrix}. \tag{20}$$

Another option with LTV-SOBI is to prioritize $\widehat{\boldsymbol{\beta}}_{3,\tau}$, namely the LTV-SOBI-alt algorithm. By writing $\widehat{\boldsymbol{\beta}}_{3,\tau} = (\boldsymbol{\mathcal{E}}\boldsymbol{\Omega}_0)\boldsymbol{\Lambda}_\tau(\boldsymbol{\mathcal{E}}\boldsymbol{\Omega}_0)'$, notice that $\boldsymbol{\mathcal{E}}\boldsymbol{\Omega}_0$ can be estimated by approximate joint diagonalization following the algorithm in (15). The preceeding step with this alternative is to find $\boldsymbol{\Omega}_0$ and $\boldsymbol{\mathcal{E}}$ separately. The solution is based on a similar idea as in (17), but the known item is different; the expression is presented as,

$$\begin{aligned} \mathrm{vec}(\widehat{\boldsymbol{\beta}}_{2,\tau}) &= \mathrm{vec}((\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau})\boldsymbol{\Omega_0}') + \mathrm{vec}(\boldsymbol{\Omega_0}(\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau})') \\ &= \mathrm{vec}((\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau})\boldsymbol{\Omega_0'I'}) + \mathrm{vec}(\boldsymbol{I\Omega_0}(\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau})') \\ &= (\boldsymbol{I} \otimes (\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau}))\mathrm{vec}(\boldsymbol{\Omega}_0') + ((\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau}) \otimes \boldsymbol{I})\mathrm{vec}(\boldsymbol{\Omega}_0) \\ &= (\boldsymbol{I} \otimes (\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau}))\boldsymbol{K}^{(p,p)}\mathrm{vec}(\boldsymbol{\Omega}_0) + ((\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau}) \otimes \boldsymbol{I})\mathrm{vec}(\boldsymbol{\Omega}_0) \\ &= \Big((\boldsymbol{I} \otimes (\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau}))\boldsymbol{K}^{(p,p)} + (\widehat{\boldsymbol{\mathcal{E}}\,\boldsymbol{\Omega}_0}\ \widehat{\boldsymbol{\Lambda}_\tau}) \otimes \boldsymbol{I}\Big)\mathrm{vec}(\boldsymbol{\Omega}_0) \end{aligned}. \tag{21}$$

By stacking over $\tau \in L$ for (21), $\boldsymbol{\Omega}_0$ can be estimated through inverse matrix or maximal likelihood estimator, and $\widehat{\boldsymbol{\mathcal{E}}}$ is obtained as $\widehat{\boldsymbol{\mathcal{E}}} = \widehat{\boldsymbol{\mathcal{E}}\boldsymbol{\Omega}_0}(\widehat{\boldsymbol{\Omega}_0})^{-1}$.

4.6 Summary of Algorithms

In the above sections, several statistical approaches have been presented to solve linearly time-varying blind source separation problem using sample autocovariance. Although the flows of the algorithm are analogous, there are so distinguishable as to almost surely result in diversified outcomes. Figure 4 visualizes the different approaches in the manner of flow chart. The original algorithm proposed by Yeredor (2003) is labeled as "Y-TVSOBI" for reference.

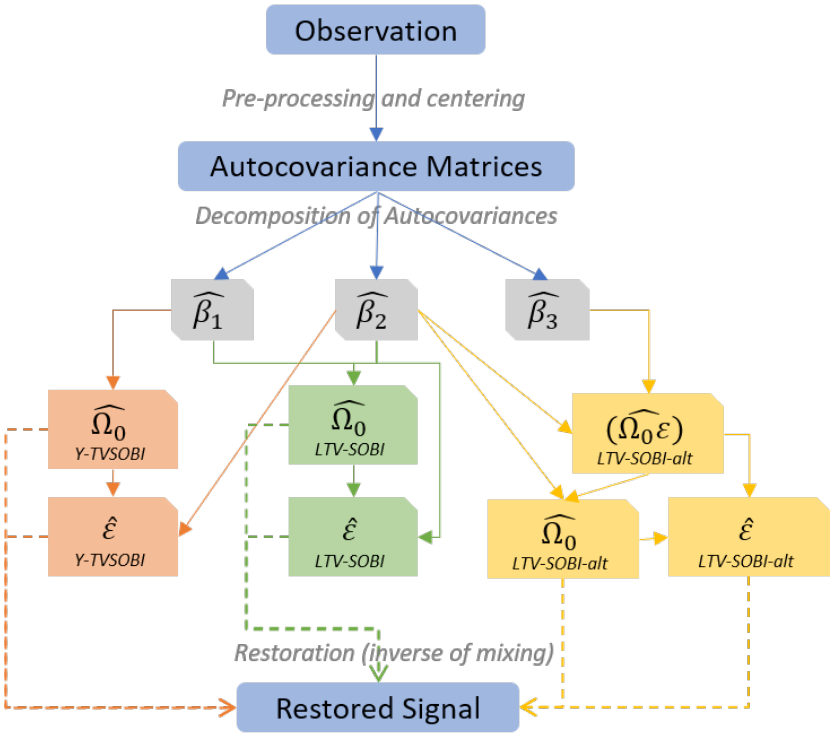

Figure 4: Comparison of Y-TVSOBI, LTV-SOBI and its alternative, briefly showing the different paths on how singnals are restored given observation

5 LTV-SOBI Performance Measures and Simulation Studies

The accuracy and effectiveness of the LTV-SOBI algorithm principally depend on the source signals and observed mixture themselves, especially on the assumptions of uncorrelatedness and stationarity. Nevertheless, when compared with SOBI in non-time-varying structure, LTV-SOBI is expected to be embedded with extra accuracy losses due to its relatively more complex mixing mechanism and associating algorithms. In spite of mathematically negligible residuals/errors in each step of the algorithm, the inherent losses can accumulate in sample autocovariance decomposition, approximate joint diagonalization and deriving $\boldsymbol{\mathcal{E}}$ and $\boldsymbol{\Omega}_0$. Precision can be further compromised when nearPD has to enforce in case of non-positive semi-definite matrices. To compare different estimation algorithms, performance indices should be used. However, to address the time-varying characteristics of the mixing matrix, modification to existing BSS performance measures is needed.

5.1 Extension of Minimal Distance Index

The minimal distance index (MD) was initially introduced by Ilmonen (2010) to measure the ICA algorithm performance by comparing true mixing matrix $\boldsymbol{\Omega}$ and estimated unmixing matrix $\widehat{\boldsymbol{\Gamma}}$. For a non-time-varying model, MD-index is defined as,

$$\mathrm{MD}(\widehat{\boldsymbol{\Gamma}}) = \frac{1}{\sqrt{p-1}} \inf_{\boldsymbol{C}\in\mathcal{C}} ||\boldsymbol{C}\widehat{\boldsymbol{\Gamma}}\boldsymbol{\Omega} - \boldsymbol{I}_p||, \tag{22}$$

where $\mathcal{C}$ is the set of matrices that permits permutation and scale ambiguity, i.e. each row and column contain exactly one non-zero element. Even though MD-index is a well-designed measure for the majority of the BSS algorithms in addition to ICA, the time-varying case requires adaptation.

The true mixing matrix $\boldsymbol{\Omega}_t$ is varying over time $t = 1, 2, \ldots, T$ in TV-SOS model, and same holds for $\widehat{\boldsymbol{\Gamma}_t}$. Therefore, this thesis proposes a time-varying version of the MD-index as defined in equation (23).

$$\mathrm{tvMD}(\{\widehat{\boldsymbol{\Gamma}}_1, \widehat{\boldsymbol{\Gamma}}_2, \ldots, \widehat{\boldsymbol{\Gamma}}_T\}) = \frac{1}{T}\sum_{t=1}^{T} \mathrm{MD}(\widehat{\boldsymbol{\Gamma}}_t)\ . \tag{23}$$

Since TV-SOS model (2) determines all $\widehat{\boldsymbol{\Gamma}_t} = \widehat{\boldsymbol{\Omega}_t}^{-1}$ by the pseudo initial mixing matrix $\boldsymbol{\Omega}_0$ and the time varying factor $\boldsymbol{\mathcal{E}}$, the tvMD can be determined once LTV-SOBI algorithm is completed and the true mixing is known.

As tvMD is officially a mean value of MD-indies over a time span, it has a value between 0 and 1 that is identical to original MD by central limit theorem, and

the smaller value suggests the better algorithm.

5.2 Extension of Signal-to-Inference Ratio

In information processing, researchers tend to decompose the restored signals into four parts: (1) target signals; (2) interference from other sources; (3) noise and (4) artifacts that are originated from separation and evaluation algorithm. The decomposition can be written as (Na & Chai, 2013; Vincent et al., 2006),

$$\boldsymbol{z} = \boldsymbol{s}_{\text{signal}} + \boldsymbol{s}_{\text{interf}} + \boldsymbol{e}_{\text{noise}} + \boldsymbol{e}_{\text{artif}} \ . \tag{24}$$

The target signal $\boldsymbol{s}_{\text{signal}}$ is not necessarily the exact source signal. Instead, carefully-selected and evaluated transformations of a source are permissible in response to the identifiability issue in BSS, and it is common to tolerate certain transformation. In particular, permutation and scaling do not usually affect signal interpretation. Some literature (e.g. Vincent et al., 2006) denotes interference as $\boldsymbol{e}_{\text{interf}}$ when falsely mixing of sources is regarded as an error even though it is originated from sources. For example, the restored signal series II mainly corresponds to source series IV, but also have a partial mixture from source series I. In this case, the former is undoubtful $\boldsymbol{s}_{\text{signal}}$, and the later should be treated as erroneous interference. For this reason, several BSS researches, especially under the information processing domain, use Signal-to-Inference Ratio (SIR) to measure the similarity between true and restored signals (Eriksson et al., 2000; Vincent et al., 2006). The ratio is defined as

$$\text{SIR} = 10 \log_{10} \frac{||\boldsymbol{s}_{\text{target}}||^2}{||\boldsymbol{s}_{\text{interf}}||^2} \ . \tag{25}$$

The LTV-SOBI algorithm by nature does not involve any external noise, and even if the noise is present in source signals, it shall become a part of true signals. Further, $\boldsymbol{e}_{\text{artif}}$ is assumed to be $\mathbf{0}$ for simplicity. Assume the restored signal to be $\widehat{\boldsymbol{z}}$ and the permutation/scaling matrix to be $\boldsymbol{C}$ as defined in (22); the signal decomposition of (24) is then

$$\begin{aligned} \widehat{\boldsymbol{z}} &= \boldsymbol{s}_{\text{signal}} + \boldsymbol{s}_{\text{interf}} \\ &= \boldsymbol{C}\boldsymbol{z} + (\widehat{\boldsymbol{x}} - \boldsymbol{C}\boldsymbol{z}) \end{aligned} . \tag{26}$$

Without doubt, time-varying factor should be considered, and the extension can be achieved by introducing a time index, that is, $\widehat{\boldsymbol{z}}_t = \boldsymbol{C}_t \boldsymbol{z}_t + (\widehat{\boldsymbol{z}}_t - \boldsymbol{C}_t \boldsymbol{z}_t)$. The SIR-index should also be slightly modified to include convolution over $t = 1, 2, \ldots, T$.

The SIR-index for LTV-SOBI can be simplified by taking $\boldsymbol{C}_t = \text{diag}(\boldsymbol{\Omega}_t \widehat{\boldsymbol{\Omega}}_t^{-1})$ after a permutation fix. In practice, the permutation is found by arranging the

numerically largest values to diagonal position either row-by-row or column-or-column in $\boldsymbol{\Omega}_t\widehat{\boldsymbol{\Omega}}_t^{-1}$. Finally, the time-varying SIR-index is the measure of all diagonal values against off-diagonal items, and it is formulated as,

$$\text{tvSIR} = 10\log_{10}\frac{\sum_{t=1}^{T}||\text{diag}(\boldsymbol{\Omega}_t\widehat{\boldsymbol{\Omega}}_t^{-1})||^2}{\sum_{t=1}^{T}||\text{off}(\boldsymbol{\Omega}_t\widehat{\boldsymbol{\Omega}}_t^{-1})||^2}, \tag{27}$$

where, $\boldsymbol{\Omega}_t\widehat{\boldsymbol{\Omega}}_t^{-1} = (\boldsymbol{I} + t\boldsymbol{\mathcal{E}})\boldsymbol{\Omega}_0\big[(\boldsymbol{I} + t\widehat{\boldsymbol{\mathcal{E}}})\widehat{\boldsymbol{\Omega}}_0\big]^{-1}$.

SIR and tvSIR do not have a direct mathematical connection to each other, but the values are comparable since they both measure the similarity between the true source and the restored one. SIR and tvSIR range from $-\infty$ to ∞, and the larger the better. It should also be noted that correlation-based SIR, that is, the implementation in `JADE` package (Miettinen et al., 2017) does not require true mixing parameters to be known, but tvSIR will always require so. This is a result of different mixing mechanism and tvSIR definition. As $\boldsymbol{\mathcal{E}}$ is assumed to be very small in value, the Pearson's correlation between restored and original can potentially encounter extreme values and therefore, makes the correlation-based SIR unrobust.

5.3 Simulation Study

As previously discussed, the performance of LTV-SOBI is expected to be significantly influenced by signal inherited properties, dimension, length and the scale of mixing matrix. Despite the impossibility to inscribe exact factors that impair the performance, simulations have been devided into four cohorts and conducted in *R*. Intending to minimize potential bias, the four cohorts have similar sources of 3-dimensional signal that involve sinusoidal and electrocardiograph (ECG) time-series, together with a moving-average or auto-regressive series. The simulated signals are analogous to those in Figure 2 with the dissimilarity in that the simulation study has 1 less dimension of either moving-average or auto-regressive series. The reduction of dimension is due to computational efficiency and visual similarity of such two signals. For convenience, define two matrix constants to be,

$$\boldsymbol{\Omega}_{\text{sim}} = \begin{bmatrix} 2 & -6 & 0.5 \\ -9 & 5 & 3 \\ -4 & 6 & 8 \end{bmatrix} \text{ and } \boldsymbol{M} = \begin{bmatrix} -3 & 6 & -6 \\ -4 & 2.5 & 6 \\ 9 & 2.1 & 7 \end{bmatrix}. \tag{28}$$

In simulation configuration, the initial mixing matrix $\boldsymbol{\Omega}_0$ is first arbitrarily fixed to $\boldsymbol{\Omega}_{\text{sim}}$ as in (28). Then, four cohorts of the simulation are parameterized with differed time-varying factors and signal length, while the difference is only in

terms of the scale with details given in Table 2. In each cohort, the true source signal and mixture are thus fixed.

Table 2: Key Configuration Parameters of Simulation Cohorts

	I	II	III	IV
$\boldsymbol{\mathcal{E}}$	$\boldsymbol{M} \times 10^{-5}$	$\boldsymbol{M} \times 10^{-4}$	$\boldsymbol{M} \times 10^{-5}$	$\boldsymbol{M} \times 10^{-4}$
Simulated Total Length	100000 (10^5)	100000 (10^5)	10000 (10^4)	10000 (10^4)
Sampling Frequency	1:1 - 1024:1	1:1 - 1024:1	1:1 - 512:1	1:1 - 512:1
Observed Length	100000 - 98	100000 - 98	10000 - 40	10000 - 40

In the realistic scenario, true source is unobservable while mixture can be observed only at certain sampling rates (aka. sampling frequencies). For example, a piece of sound shall include 4800k samples per second as a signal mixture, but the recording equipment can only sample at 48kHz, which means that merely 1 out of every 100 sources is sampled as an observation. Aiming to simulate in accordance with such scenario and to evaluate how the observation length could affect LTV-SOBI performance, the simulation study further generates artificial observed mixture and corresponding source upon different sampling rates based on the same source and mixture. Figure 5 illustrates the sampling mechanism, and Table 2 summarizes the sampling rates in each cohort. Consequently, multiple mixtures of different lengths (as a result of sampling rates) are simulated within each cohort. In brief, instead of generating new pseudo sources and mixtures, the simulation study considers the sampling frequencies as a more robust alternative.

In the next step, LTV-SOBI algorithms along with various alternatives are applied to each generated observed mixture, and the results are compared against the true mixing matrix using tvMD and the true signals using tvSIR. Each observed mixture has seven different algorithms applied, including two types of Yeredor's TV-SOBI (with and without quadratic form), four types of LTV-SOBI (with and without symmetry fix, with and without quadratic form) and LTV-SOBI-alt.

Finally, over 1000 similar simulations are performed to further eliminate potential bias and outliers and enable reporting simulation results. Figure 6 overviews the simulation study in the manner of progress flow, and the full R code is attached in Appendix C.

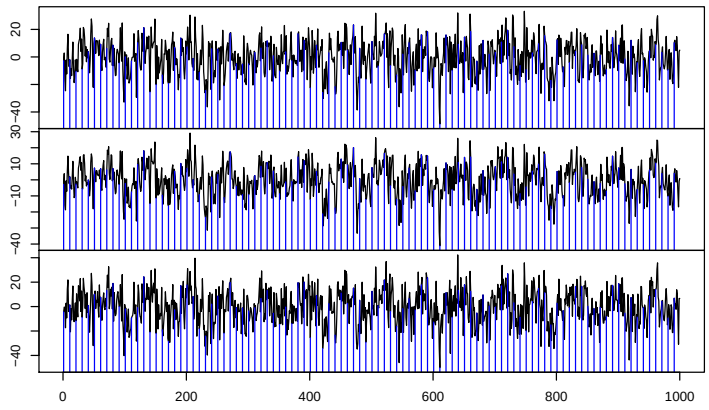

Figure 5: Illustration of 10:1 sampling rate - only signals at blue vertical bars are sampled and observed

5.4 Simulation Study Results

The aforementioned tvSIR and tvMD, served as a time-varying application of SIR and MD correspondingly, are applied to measure the algorithm performance in terms of accuracy and capability of restoring original mixing matrices and signals. The results are reported in Figures 7-10. Each sub-graph reports a specific simulation cohort with a defined lag parameter used in the signal restoration algorithm. The curves visualize the performance influenced by sampling frequencies, which are equivalent to observed signal lengths. Different algorithms can be distinguished by the line color.

Simulations showed that minor alternatives do not significantly affect the performance of a given algorithm. For example, the symmetry correction for estimated matrices in LTV-SOBI does not affect the overall performance metric. Therefore, further results presented below are aggregated, for convenience and simplicity, only over the major algorithms, namely LTV-SOBI, LTV-SOBI-alt, and Y-TVSOBI. Consequently, a point in Figure 7 and 10 represent the mean value of simulations under same settings. Besides, there is a separate web-based interactive dashboard, which is available at http://bss.yan.fi, to enable investigation of performance metrics from various perspectives, including those that are not directly reported in the sections.

Figures 7 and 8 presumes to exhibit better performance measured by tvSIR with a larger value; the higher the points and curves the better. The missing values and other deviations will be deliberated in Section 6. As seen in the results, the new LTV-SOBI outperformed Yereodr's original TV-SOBI in most cases, while the comparable advantage seems to be diminished with increased

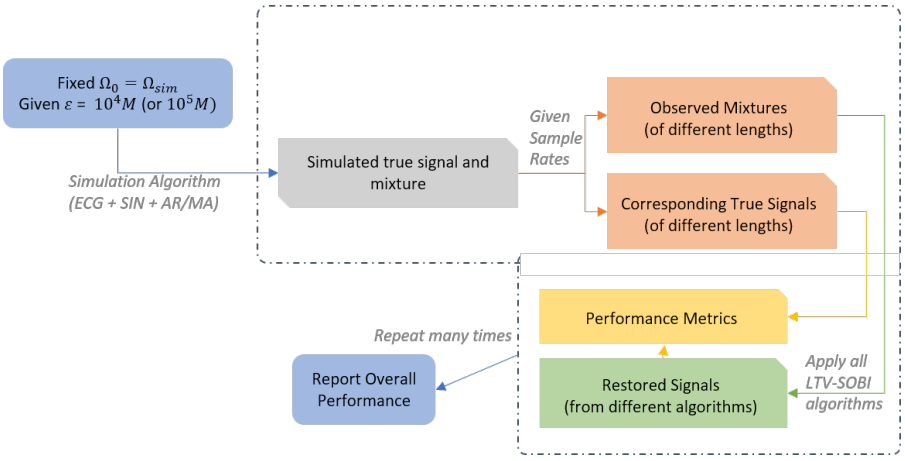

Figure 6: Overview of simulation study settings

observed length. On the other hand, the LTV-SOBI-alt algorithm is relatively insensitive to length, though the benefits from larger observation size are also less significant.

Using tvMD, Figures 9 and 10 demonstrate better performance with a smaller value, and LTV-SOBI-alt is comparably the best algorithm in most cases except for simulation cohort III; larger sample size often leads to improved performance. The difference among the three major algorithms is essentially minor.

In conclusion, the extension of MD and SIR grants insights into algorithm performance; the simulation results suggest that time-varying blind source separation problems are yet rather challenging as the tvSIR and tvMD values are not sufficiently good. LTV-SOBI algorithm is a comparably more applicable and effective approach. Nonetheless, the metric is not robust enough under all scenarios, and Section 6 will attempt to detail underlying issues. Finally, the simulation results have to exclude the comparison with ordinary SOBI because of non-compatible metrics; SOBI would be penalized by the time-varying model while benefiting from its ignorance of possible extreme values caused by $\boldsymbol{\mathcal{E}}$. Supposing the metric is acceptable with SOBI, the results prefer SOBI under large sample sizes and LTV-SOBI when the length of the signal is relatively small. Appendix D presents a few simulation results with SOBI, and the interactive dashboard provides full SOBI results.

5.5 Performance Consistency over Singal Types and Time-Varying Rates

The above simulation includes a series of ECG-signal with doubtful stationarity. Further simulation has been performed to check the performance of LTV-SOBI when source signals are composed of three different moving-average series, and

the effect of time-varying speed ($\boldsymbol{\mathcal{E}}$) is also briefly examined. The results suggest that LTV-SOBI performance stays relatively consistent under different signal types and rates/speed of time-varying. When signal length is set to be 10000, lag selected to be 3, the tvMD averages to 4.33, 4.29 and 4.32 for $\boldsymbol{\mathcal{E}} = \boldsymbol{M} \times 10^{-2}$, $\boldsymbol{M} \times 10^{-3}$ and $\boldsymbol{M} \times 10^{-4}$ correspondingly using LTVI-SOBI. Performance is also tested to be insensitive to the signal length. The results are similar to those reported in Figure 7 and 8.

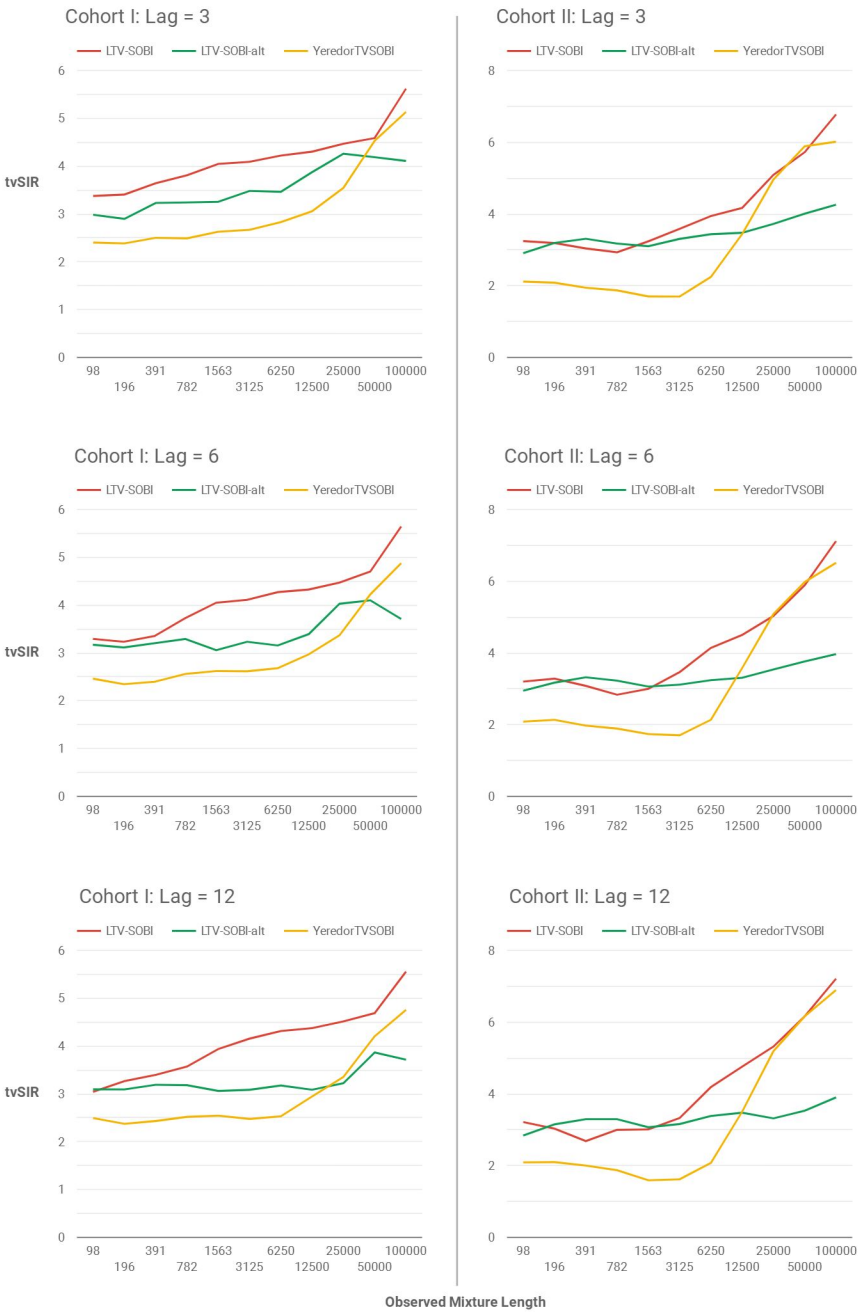

Figure 7: Signal separation performance measured by tvSIR for Cohort I and II when the sources are separted using LTV-SOBI, LTV-SOBI-alt and Yeredor TVSOBI. Larger value indicates better separation performance.

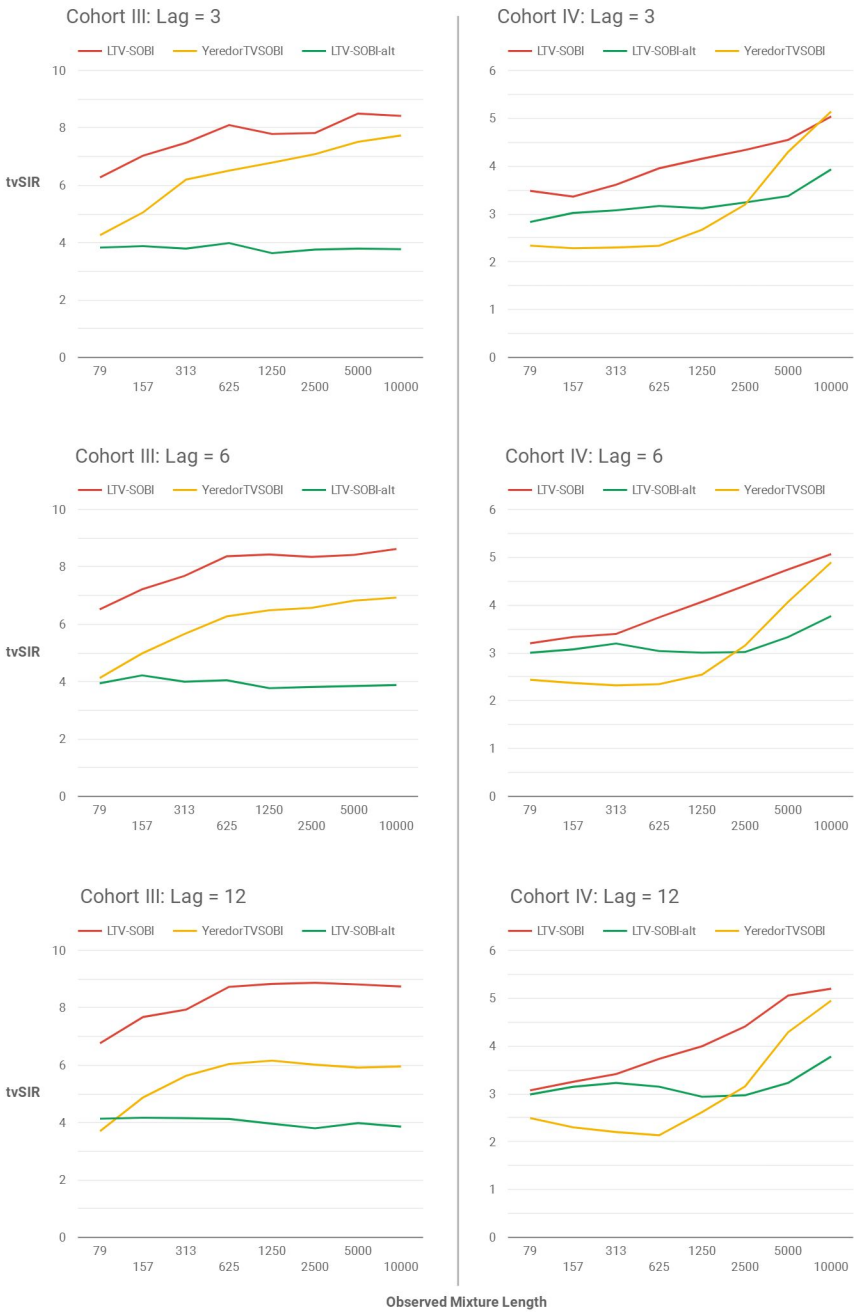

Figure 8: Signal separation performance measured by tvSIR for Cohort III and IV when the sources are separted using LTV-SOBI, LTV-SOBI-alt and Yeredor TVSOBI. Larger value indicates better separation performance.

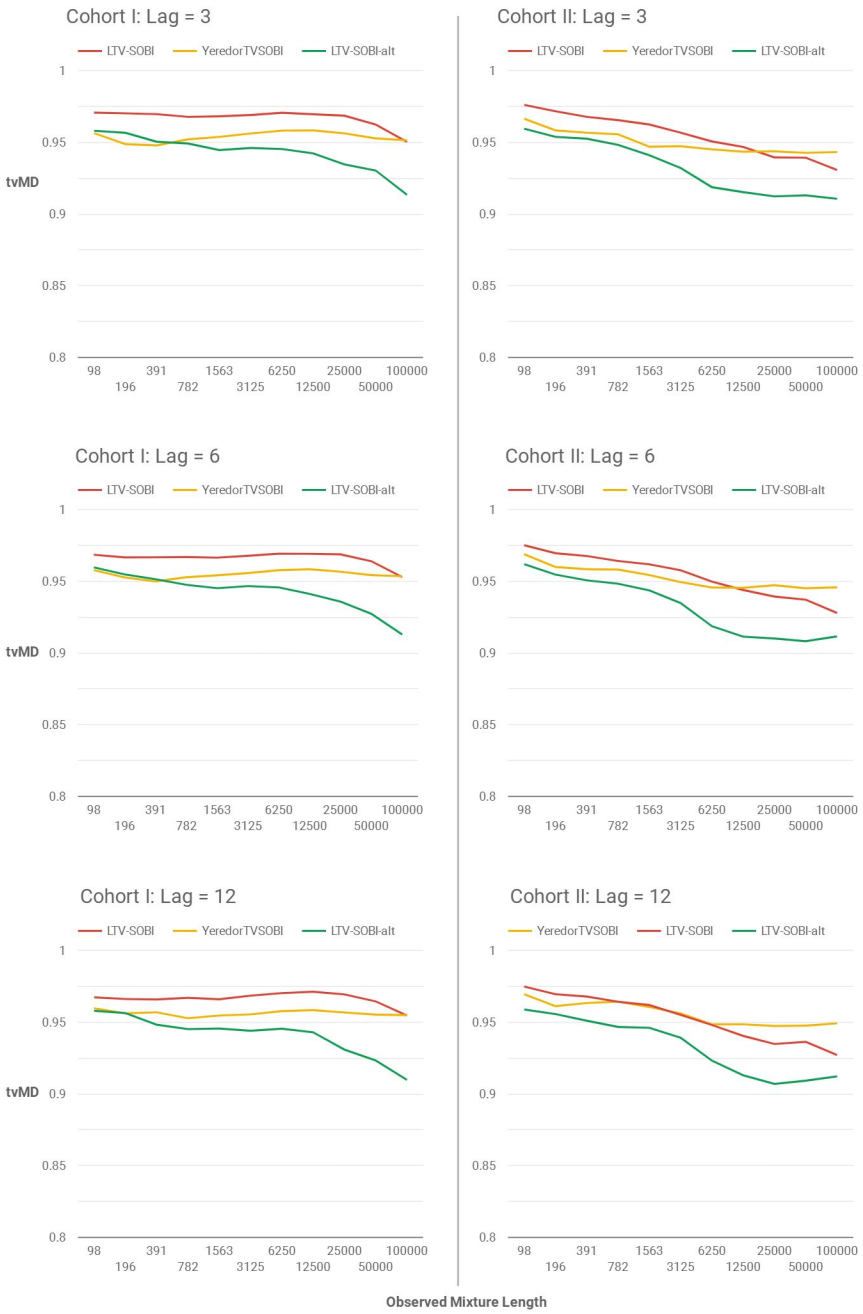

Figure 9: Signal separation performance measured by tvMD for Cohort I and II when the sources are separted using LTV-SOBI, LTV-SOBI-alt and Yeredor TVSOBI. Smaller value indicates better separation performance.

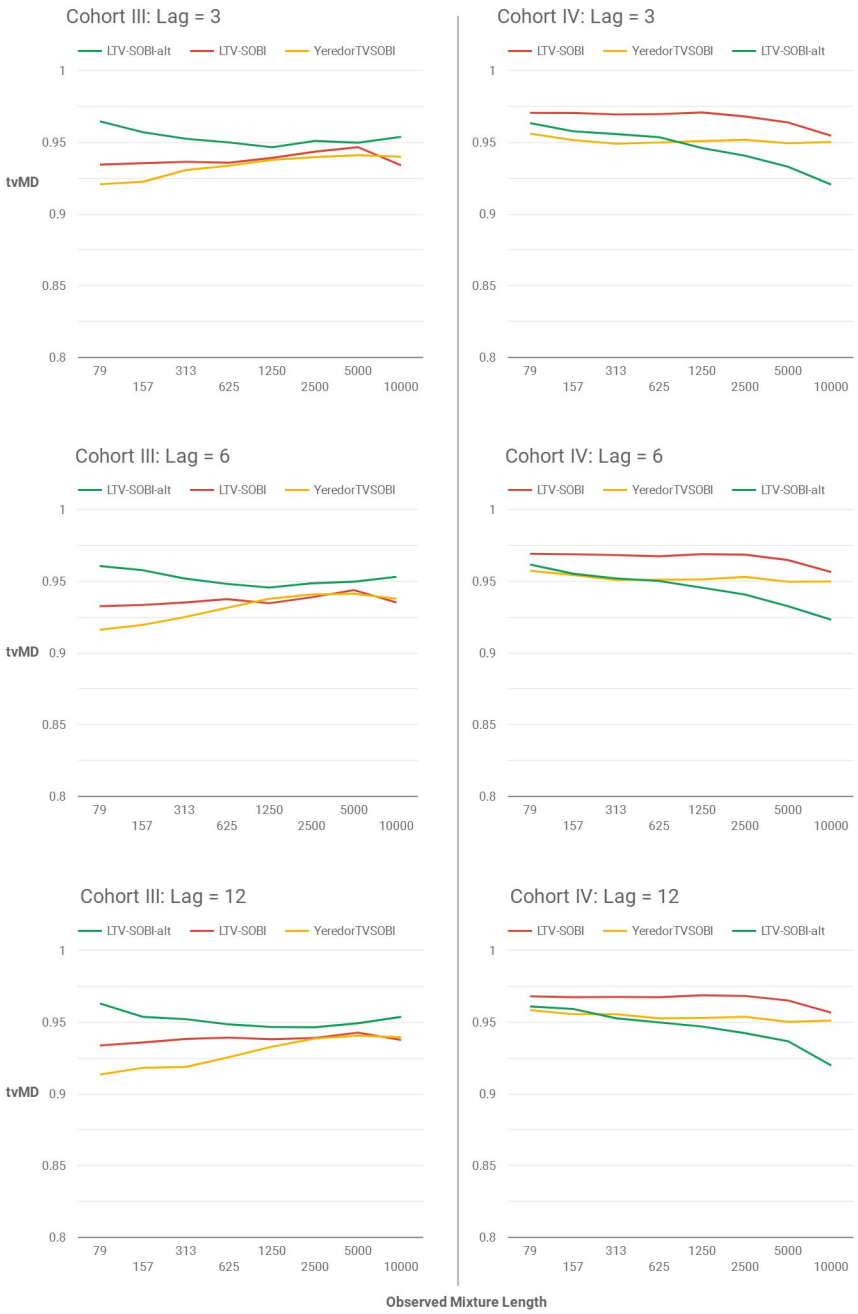

Figure 10: Signal separation performance measured by tvMD for Cohort III and IV when the sources are separted using LTV-SOBI, LTV-SOBI-alt and Yeredor TVSOBI. Smaller value indicates better separation performance.

6 Discussion

Previous sections have reviewed the TV-SOS model together with Yeredor's original TV-SOBI algorithm, while new approaches of LTV-SOBI and its alternatives are presented, together with a simulation study. Compared with Yeredor's original TVSOBI algorithm, LTV-SOBI eliminates one major problem of the estimated covariance matrix being non-positive semi-definite as detailed in Section 4, and the assumption of $\boldsymbol{\mathcal{E}} << \boldsymbol{I}$ is no longer required. Yet, the LTV-SOBI algorithm does not always guarantee a solution and is particularly dependent on the observation. The most significant cause is the computational singularity, while in rather few cases, a negative-definite estimator also prohibits the algorithm of finding the nearest positive-definite matrix and ultimately fails the whole separation process. Despite the mixing matrix $\boldsymbol{\Omega}_0$ is assumed to be of full-rank, the non-singularity of every $\boldsymbol{\Omega}_t$ could not be secured as a result of small valued $\boldsymbol{\mathcal{E}}$. Sadly, the computational singular issue could occur in almost every step.

The performance of newly proposed algorithm is generally consistent given different signal types and time-varying rates, but, as illustrated in Figure 11, the simulation results indicate probable stability issues. There exists a considerable amount of simulations that yield significantly better results (tvSIR ≈ 20) when compared with average, while there barely exists poor results (tvSIR < 0). The issue could arise from both algorithm compatibility and metric robustness. Since $\boldsymbol{\mathcal{E}}$ is supposed to be small, extreme values can often occur when performing the inverse, and these can also be observed in the restored signal. Neither tvSIR nor tvMD tackles outliers in a truly robust manner. Additionally, the simulation study uses an ECG signal to reflect practical scenarios, but this signal's stationarity is in question. In fact, the simulation study results presented in Section 5.4 are conservative, meaning that the true performance of LTV-SOBI should be at least no worse. Furthermore, LTV-SOBI does not provide the option to utilize robust second-order statistics due to its reliance on autocovariance structure in equation (2). Stationarity, seasonality, the scale of time-variance, observation length and signal dimension are all possible factors that affect LTV-SOBI's performance. Unfortunately, the exact causes and mechanisms have not been identified thoroughly.

This thesis offers the application, extension, and improvement to Yeredor's original time-varying blind source identification, together with modified BSS metrics, while remarking the challenges in robustness and imperfection of measurement. Further studies would be demanded in terms of tensor observation, consistent performance evaluation and statistical property of TV-SOS results.

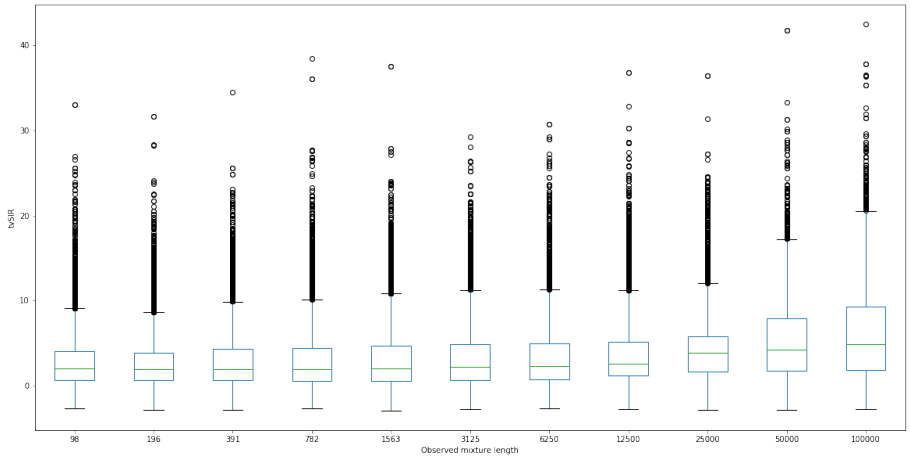

Figure 11: Boxplot for tvSIR measures in Cohort I simulation when sources are separated by LTV-SOBI

Appendix

A Implementaion of LTV-SOBI in *R*

The primary `R`-functions that support LTV-SOBI are listed as following,

- `ltvsobi(x, lags = 12, quadratic = TRUE, fix_symmetry = TRUE, verbose = FALSE)` returns list of class `tvbss` that is compatible with `JADE` pacakge's `bss` class;
- `ltvsobi2(x, lags = 12, quadratic = TRUE, fix_symmetry = TRUE, verbose = FALSE)` implements the LTV-SOBI-alt in the same manner as `ltvsobi`
- `SIR_all(bss_res, Omega, Epsilon, S)` calculates all applicable performance metric, supporting both `tvbss` class and `bss` class;
- `tvmix(z, Omega, Epsilon, x_only = TRUE)` and `tvunmix <- function(x, Omega_hat, Epsilon_hat)` serve as TV-SOS utility that generates and restores time-varying mixture based on source signal and mixing parameters.

For major computer platforms, linear estimation is faster and more reliable compared with the inverse matrix. Consequently, inversion is minimized in `LTV-SOBI`. It is possible to use a vectorization version of `ltvsobi`, and it indeed reduces the CPU time in processing compared with looping. However, vectorization and Kronecker product significantly raise the dimension of matrices; therefore, the improvement in CPU time is at the cost of RAM usage, which becomes a problem if the dimension is too high and `R` crash might occur. It is advisable to use non-vectorized version for large p and T ($p \geq 10,\ T \geq 50000$).

B Introduction to LTV-SOBI Performance Metric Explorer

As Section 5 stated the complexity in presents LTV-SOBI performance due to a large number of factors, an interactive dashboard is designed to enable customizable performance exploring. The explorer is available at http://bss.yan.fi. The simulation results are conducted in *R*, and the data is stored in structured database while the dashboard is prepared in an open platform. The explorer allows the user to view and compare performance metrics from different perspectives.

There are 5 dynamic pages designed to aid performance evaluation of multiple algorithms, involving a trend of performance change over the number of observations. The "Simple View" page allows users to select one single cohort and aggregate over 1 or more lags, and the "View All Corhots" will display all 4

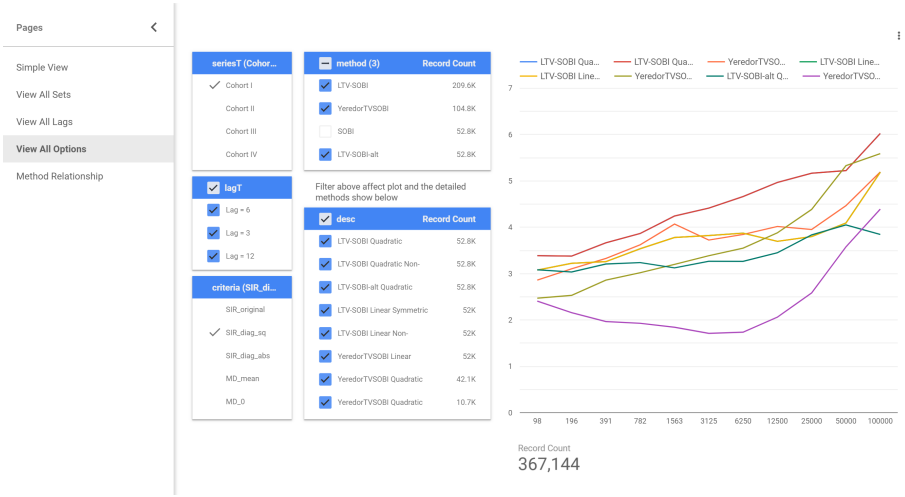

Figure 12: Screeshot of BSS Explorer, an interactive dashboard

plots that representing each cohort on the same page. "View All Lags" shows 3 plots of different lags given user selection of 1 or more cohorts. Control on all possible parameters is available in the "View All Options" page. In addition, the "Method Relationships" attempts to overview the hierarchies of all methods discussed, where the size of blocks shall indicate the number of valid simulations done.

C *R* Code for Simulation Study

```
path <- paste0(getwd(), "/sim/")
library(tidyverse)
source("rfun.R")  # contains algorithms and utilities of LTV-SOBI
source("rsim.R")  # contains functions that simulate various sources
source("rlab_x.R")# an implementation of Yeredors' TV-SOBI

# simulation sampling interval ---------------------------------------------

do_it_once <- function(x, z, lll = 6, id = "ID", Omega, Epsilon){

  for (i in 1:8) {
    flag <- TRUE
    tryCatch({
      if(i == 1) bss_res <- JADE::SOBI(x, k = lll)
      if(i == 2) bss_res <- tvsobi  (x, lag.max = lll, TRUE)
      if(i == 3) bss_res <- tvsobi  (x, lag.max = lll, FALSE)
      if(i == 4) bss_res <- ltvsobi (x, lags = lll,
                                     quadratic = TRUE,
                                     fix_symmetry = TRUE)
```

```
      if(i == 5) bss_res <- ltvsobi (x, lags = lll,
                                     quadratic = TRUE,
                                     fix_symmetry = FALSE)
      if(i == 6) bss_res <- ltvsobi (x, lags = lll,
                                     quadratic = FALSE,
                                     fix_symmetry = TRUE)
      if(i == 7) bss_res <- ltvsobi (x, lags = lll,
                                     quadratic = FALSE,
                                     fix_symmetry = FALSE)
      if(i == 8) bss_res <- ltvsobi2(x, lags = lll)
    }, error = function(e) {
      print(paste("skip to next method due to", e))
      flag <<- FALSE
    })

    if(flag) {
      #save_estimator(bss_res, id) #save_restored(bss_res, id)
      benchmarks <- SIR_all(bss_res, Omega, Epsilon, z)
      remove(bss_res)
      save_eval(benchmarks, id)
    }
  }
}

save_eval <- function(benchmarks, id){
  df <- NULL
  for(i in 2:length(benchmarks)){
    df <- data.frame(criteria = attributes(benchmarks)$names[i],
                     value    = benchmarks[[i]]) %>% rbind(df, .)
  }
  df$detail <- benchmarks$method
  df$method <- word(benchmarks$method)
  df$id     <- id
  df$desc   <- str_remove(benchmarks$method, " NearestSPD")
  df$N      <- benchmarks$N
  df$p      <- benchmarks$p

  df <- df %>% filter(criteria != "N" & criteria != "p")

  Sys.sleep(runif(1))
  fname <- paste0(path, "benchmarks-", id, ".rds")
  if (file.exists(fname)) saveRDS(rbind(readRDS(fname), df), file = fname)
  else  saveRDS(df, file = fname)
}

# function to run the repeated simulation---------------------------------

multido <- function(E, N, sn){
  for(i in 1:100){
```

```
    Omega <- matrix(c(2, -9, -4, -6, 5, 6, 0.5, 3, 8), ncol =3)
    Epsilon <- 10^(-E) * matrix(c(-3, - 4, 9  ,
                                  6, 2.5, 2.1,
                                 -6, 6  , 7  ), ncol = 3)
    zall <- sim_good_sources(N = 1e4, 3)
    xall <- tvmix(zall, Omega, Epsilon)

    # loop for freqs
    freq_list <- 2^(0:10)
    for(freq in freq_list){
      for(l in c(3,6,12,1)){
        ids  <- seq(from = 1, to = nrow(xall), by = freq)
        x <- xall[ids,]
        z <- zall[ids,]
        do_it_once(x, z, lll = l,
                   id = paste0("seq", sn, "_fixed_freq_E",
                               E, "N", N, "_Boot_lag", l), Omega, Epsilon)
      }
    }
  }
}

# submitting the job using multi-core----------------------------------------

library(parallel) # enable parallel processing
mclapply(
  as.list(1:500),
  function(seq) {
    multido(5,5,seq)
    multido(5,4,seq);
    multido(4,5,seq);
    multido(4,4,seq);
  },
  mc.cores = detectCores()
)
```

D Supplementary Simulation Results

Figure 13 contains the identical configuration as Figure 8 with an extra comparison with the original SOBI algorithm. Further views can be found in the interactive performance metric explorer (Appendix B).

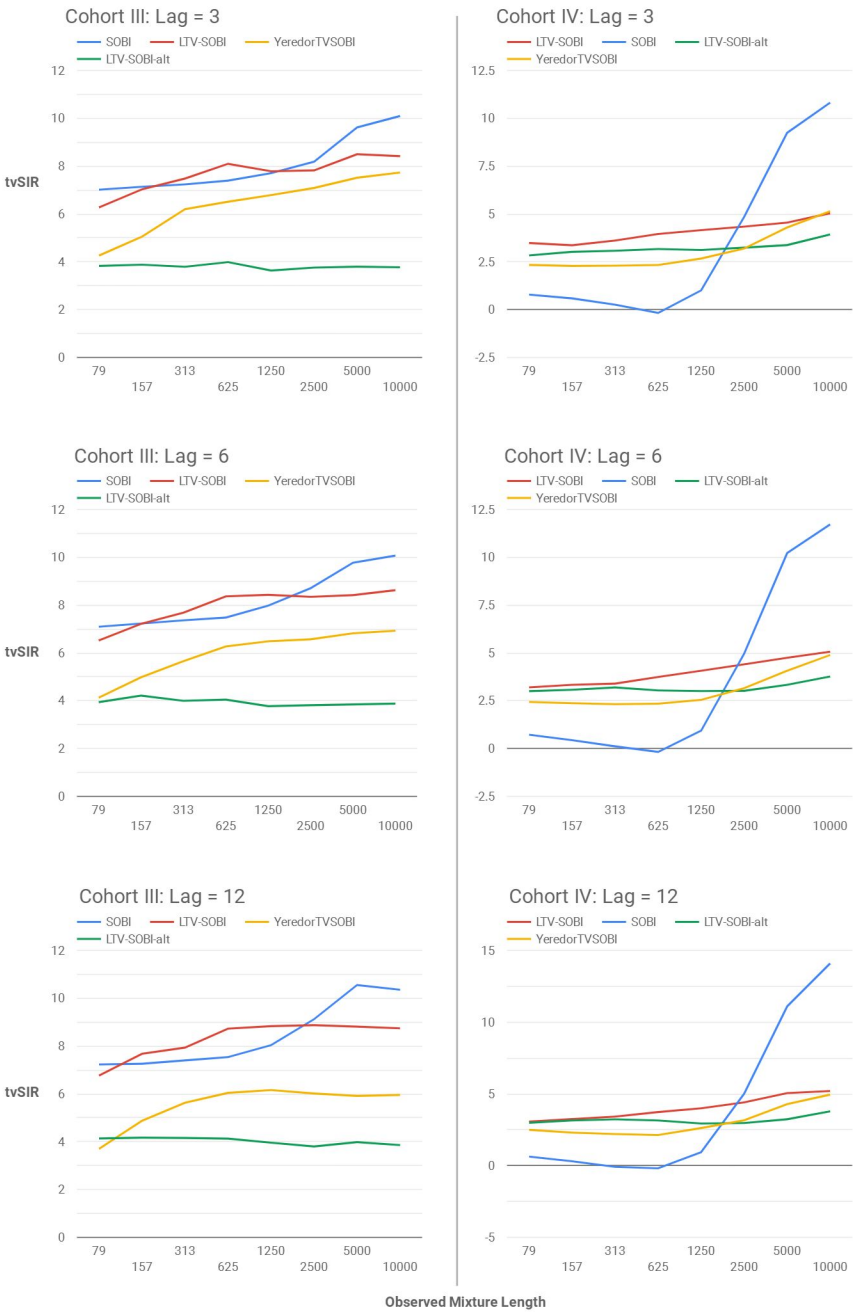

Figure 13: Signal separation performance measured by tvSIR for Cohort III and IV when the sources are separted using SOBI, LTV-SOBI, LTV-SOBI-alt and Yeredor TVSOBI. Larger value indicates better separation performance.

References

Bates, D., & Maechler, M. (2018). *Matrix: Sparse and dense matrix classes and methods*. https://CRAN.R-project.org/package=Matrix

Belouchrani, A., Abed-Meraim, K., Cardoso, J.-F., & Moulines, E. (1997). A blind source separation technique using second-order statistics. *IEEE Transactions on Signal Processing*, *45*(2), 434–444.

Cheng, S. H., & Higham, N. J. (1998). A modified Cholesky algorithm based on a symmetric indefinite factorization. *SIAM Journal on Matrix Analysis and Applications*, *19*(4), 1097–1110.

Clarkson, D. B., & Jennrich, R. I. (1988). Quartic rotation criteria and algorithms. *Psychometrika*, *53*(2), 251–259.

Comon, P. (1994). Independent component analysis, a new concept? *Signal Processing*, *36*(3), 287–314.

Eriksson, J., Karvanen, J., & Koivunen, V. (2000). Source distribution adaptive maximum likelihood estimation of ICA model. *Proceedings of the 2nd International Conference on ICA and BSS*, 227–232.

Hérault, J., & Ans, B. (1984). Réseau de neurones à synapses modifiables: Décodage de messages sensoriels composites par apprentissage non supervisé et permanent. *Comptes Rendus Des Séances de L'Académie Des Sciences. Série 3, Sciences de La Vie*, *299*(13), 525–528.

Hyvärinen, A. (2013). Independent component analysis: Recent advances. *Philosophical Transactions of the Royal Society A*, *371*(1984), 20110534.

Hyvärinen, A., & Oja, E. (2000). Independent component analysis: Algorithms and applications. *Neural Networks*, *13*(4-5), 411–430.

Ilmonen, P., Nordhausen, K., Oja, H., & Ollila, E. (2010). A new performance index for ICA: Properties, computation and asymptotic analysis. *International Conference on Latent Variable Analysis and Signal Separation*, 229–236.

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2013). *An introduction to statistical learning* (Vol. 112). Springer.

Jutten, C., & Herault, J. (1991). Blind separation of sources, part i: An adaptive algorithm based on neuromimetic architecture. *Signal Processing*, *24*(1), 1–10.

Kaftory, R., & Zeevi, Y. Y. (2007). Probabilistic geometric approach to blind separation of time-varying mixtures. *International Conference on Independent Component Analysis and Signal Separation*, 373–380.

Knol, D. L., & Berge, J. M. ten. (1989). Least-squares approximation of an improper correlation matrix by a proper one. *Psychometrika*, *54*(1), 53–61.

Li, X.-L., & Zhang, X.-D. (2007). Nonorthogonal joint diagonalization free of degenerate solution. *IEEE Transactions on Signal Processing*, *55*(5), 1803–1814.

Miettinen, J., Illner, K., Nordhausen, K., Oja, H., Taskinen, S., & Theis, F. J. (2016). Separation of uncorrelated stationary time series using autocovariance matrices. *Journal of Time Series Analysis*, *37*(3), 337–354.

Miettinen, J., Nordhausen, K., & Taskinen, S. (2017). Blind source separation based on joint diagonalization in R: The packages JADE and BSSasymp. *Journal of Statistical Software*, *76*.

Myers, R. H., & Myers, R. H. (1990). *Classical and modern regression with applications* (Vol. 2). Duxbury Press Belmont, CA.

Na, Y., & Chai, B. (2013). Performance evaluation for frequency domain blind source separation algorithms. *Journal of Computational Information Systems*, *9*(18), 7369–7379.

Nordhausen, K. (2014). On robustifying some second order blind source separation methods for nonstationary time series. *Statistical Papers*, *55*(1), 141–156.

Nordhausen, K., Cardoso, J.-F., Miettinen, J., Oja, H., Ollila, E., & Taskinen, S. (2019). *JADE: Blind source separation methods based on joint diagonalization and some bss performance criteria*. https://CRAN.R-project.org/package=JADE

Papoulis, A., & Pillai, S. U. (2002). *Probability, random variables, and stochastic processes*. Tata McGraw-Hill Education.

Tharwat, A. (2018). Independent component analysis: An introduction. *Applied Computing and Informatics*.

Tong, L., Soon, V., Huang, Y., & Liu, R. (1990). AMUSE: A new blind identification algorithm. *Circuits and Systems, 1990., IEEE International Symposium on*, 1784–1787.

Tong, L., Xu, G., & Kailath, T. (1994). Blind identification and equalization based on second-order statistics: A time domain approach. *IEEE Transactions on Information Theory*, *40*(2), 340–349.

Vincent, E., Gribonval, R., & Févotte, C. (2006). Performance measurement in blind audio source separation. *IEEE Transactions on Audio, Speech, and Language Processing*, *14*(4), 1462–1469.

Weisman, T., & Yeredor, A. (2006). Separation of periodically time-varying mixtures using second-order statistics. *International Conference on Independent Component Analysis and Signal Separation*, 278–285.

Yeredor, A. (2002). Non-orthogonal joint diagonalization in the least-squares sense with application in blind source separation. *IEEE Transactions on Signal Processing*, *50*(7), 1545–1553.

Yeredor, A. (2003). TV-sobi: An expansion of SOBI for linearly time-varying mixtures. *Proc. 4th International Symposium on Independent Component Analysis and Blind Source Separation (ICA'03), Nara, Japan*.