Independent innovation analysis for nonlinear vector autoregressive process

Hiroshi Morioka  
RIKEN AIP, Japan  
ATR NIA, Japan

Aapo Hyvärinen  
Université Paris-Saclay, Inria, France  
Department of Computer Science and HIIT  
University of Helsinki, Finland

Abstract

The nonlinear vector autoregressive (NVAR) model provides an appealing framework to analyze multivariate time series obtained from a nonlinear dynamical system. However, the innovation (or error), which plays a key role by driving the dynamics, is almost always assumed to be additive. Additivity greatly limits the generality of the model, hindering analysis of general NVAR process which have nonlinear interactions between the innovations. Here, we propose a new general framework called independent innovation analysis (IIA), which estimates the innovations from completely general NVAR. We assume mutual independence of the innovations as well as their modulation by a fully observable auxiliary variable (which is often taken as the time index and simply interpreted as nonstationarity). We show that IIA guarantees the identifiability of the innovations with arbitrary nonlinearities, up to a permutation and component-wise invertible nonlinearities. We propose two practical estimation methods, both of which can be easily implemented by ordinary neural network training. We thus provide the first rigorous identifiability result for general NVAR, as well as very general tools for learning such models.

1 Introduction

Multivariate time series are of considerable interest in a number of domains, such as finance, economics, and engineering. Vector autoregressive (VAR) models have played a central role in capturing the dynamics hidden in such time series [24]. VAR models typically attempt to fit a multivariate time series with linear coefficients representing the dependencies of multivariate variables within limited number of lags, and innovation (or error) representing new information (impulses) fed to the process at a given time point. Although it has been common practice to maintain a linear functional form to achieve interpretability and tractability, recent studies have provided a growing body of evidence that nonlinearity often exists in time series, and allowing for nonlinearities can be valuable for uncovering important features of dynamics [13, 14, 17, 22, 23, 27, 28]. Many recent studies used a deep learning framework to model nonlinear processes in video [1, 19, 21, 26, 30, 32] or audio [29], for example, with neural networks.

The innovation plays a key role by driving time series, and it can have a concrete meaning, such as economic shocks in finance, external torques given to a mechanical system, or stimulation in neuroscience experiments. However, its estimation has a serious indeterminacy even with linear models, if only conventional statistical assumptions are made. To facilitate estimation, VAR typically assumes that the innovations are additive, multivariate Gaussian (not necessary uncorrelated), and temporally independent (or serially uncorrelated). A well-known consequence of this is that the innovations cannot be identified: Multiplication of such innovations by any orthogonal matrix will not change distribution of the observed data, which hinders their interpretation. Some studies proposed
We start by transforming the NV AR model to something similar to NICA. This leads us to consider a general NV AR model, which is first order (NV AR(1)) for simplicity:

\[ x_t = f(x_{t-1}, s_t), \]  

where \( f : \mathbb{R}^{2n} \to \mathbb{R}^n \) represents an NV AR (mixing) model, and \( x_t = [x_1(t), \ldots, x_n(t)]^T \) and \( s_t = [s_1(t), \ldots, s_n(t)]^T \) are observations and innovations (or errors) of the process at time point \( t \), respectively. As with ordinary VAR, the innovations are assumed to be temporally independent (serially uncorrelated). Importantly, this model includes potential nonlinear interaction between the observations and innovations, unlike ordinary linear VAR models and additive innovation nonlinear models. IIA can be seen as an extension of recently proposed NICA frameworks, and guarantees the identifiability of innovations up to permutation and component-wise nonlinearities.

We propose a novel VAR analysis framework called independent innovation analysis (IIA), which enables estimation of innovations hidden in unknown general NVAR. We first propose a model which allows for nonlinear interactions between innovations and observations, with very general nonlinearities. IIA can be seen as an extension of recently proposed NICA frameworks, and enables estimation of innovations hidden in unknown general NVAR. Our goal is to estimate the innovations (latent components) only from the observations \( x \) obtained from the unknown NVAR process. The model, learning algorithms, Theorems, and proofs below can be easily extended to higher order models NVAR(\( p \)) (\( p > 1 \)) by replacing \( x_{t-1} \) by \( [x_{t-1}, \ldots, x_{t-p}] \).

To estimate the innovation, we propose a new framework called IIA, which learns the inverse (demixing) of the NVAR (mixing) model from the observations in data-driven manner, based on some statistical assumptions on the innovations. The theory is related to ICA, which estimates a demixing from instantaneous mixtures of latent components, i.e., \( x_t = f(s_t) \), where \( f \) is usually a linear function. However, IIA includes a recurrent structure of the observations in the model (Eq. 1), which makes IIA theoretically distinct from ordinary ICA. Nevertheless, in the following we leverage the recently developed theory of NICA.

We start by transforming the NVAR model to something similar to NICA. This leads us to consider the following augmented NVAR (mixing) model

\[ \begin{bmatrix} x_t \\ x_{t-1} \end{bmatrix} = \tilde{f} \begin{bmatrix} s_t \\ x_{t-1} \end{bmatrix} = \begin{bmatrix} f(x_{t-1}, s_t) \\ x_{t-1} \end{bmatrix}, \]  

where \( \tilde{f} : \mathbb{R}^{2n} \to \mathbb{R}^{2n} \) is the augmented model, which includes the original NVAR model \( f \) in the half of the space, and an identity mapping of \( x_{t-1} \) in the remaining subspace. Importantly, this augmentation does not impose any particular constraint on the original model. We only assume that this augmented model is invertible (i.e. bijective; while \( f \) itself cannot be invertible) as well as sufficiently smooth, but we do not constrain it in any other way. The estimation of the innovation \( s \) can then be achieved by learning the inverse (demixing) of the augmented NVAR model \( \tilde{f} \):

\[ \begin{bmatrix} s_t \\ x_{t-1} \end{bmatrix} = \tilde{g} \begin{bmatrix} x_t \\ x_{t-1} \end{bmatrix} = \begin{bmatrix} g(x_t, x_{t-1}) \\ x_{t-1} \end{bmatrix}, \]  

where \( \tilde{g} : \mathbb{R}^{2n} \to \mathbb{R}^{2n} \) is the augmented demixing model of the (true) augmented NVAR model \( \tilde{f} \), and \( g(x_t, x_{t-1}) \in \mathbb{R}^n \) is the sub-space of the demixing model representing a mapping from two temporally consecutive observations to the innovation at the corresponding timing. This is simply a deduction from Eq. 2, and does not impose any additional assumptions on the original model.
2.2 Innovation model with auxiliary variable

The estimation of the demixing model in an unsupervised (or self-supervised) manner needs some assumptions on the innovations. Although some studies guaranteed the identifiability by assuming mutual independence of the innovations in linear VAR models \[11][18][20], it would not be enough in nonlinear cases, as can be seen in well-known indeterminacy of NICA with i.i.d. components \[9\]. Thus, we here adopt the framework recently proposed for NICA \[8][10\]; we assume that we can additionally observe auxiliary information about the innovation for each data point \(t\), represented by a random variable \(u_t\), which specifies the modulations of the distributions of the innovations as a function of \(t\). In practice, \(u_t\) can simply be time-index \(t\) \[10\] or a time-segment-index \[8\], thus incorporating information about nonstationarity. More specifically, we assume the followings:

A1. The innovations are temporally independent.

A2. Each \(s_i\) is statistically dependent on some fully-observed \(m\)-dimensional random auxiliary variable \(u\), but conditionally independent of the other \(s_j\), and has a univariate exponential family distribution conditioned on \(u\) (we omit data index \(t\) here):

\[
p(s|u) = \prod_{i=1}^{n} \frac{Q_i(s_i)}{Z_i(u)} \exp \left( \sum_{j=1}^{k} q_{ij}(s_i) \lambda_{ij}(u) \right),
\]

where \(Q_i\) is the base measure, \(Z_i\) is the normalizing constant, \(k\) is the model order, \(q_{ij}\) is the sufficient statistics, and \(\lambda_{ij}(u)\) is a parameter (scalar function) depending on \(u\)\[1\].

The temporal independence (A1) is the ordinary assumption for VAR. The A2 is related to the assumption of Gaussian innovations in ordinary VAR, but requires more specific properties represented by conditional independence and sufficient probabilistic modulation, determined by a fully observable auxiliary variable \(u\). Note that exponential families have universal approximation capabilities, so this assumption is not very restrictive \[25\].

3 Learning algorithms

Depending on the type of the auxiliary variable \(u\) in the innovation model (see A2), we can develop two learning algorithms; (IIA-GCL) generalized contrastive learning \[10\]-based framework for the general case of a possibly continuous-valued \(u\), and (IIA-TCL) time-contrastive learning \[8\]-based framework for the special case in which the auxiliary variable \(u\) is integer taking a finite number of values (e.g. a time segment index).

3.1 General contrastive learning framework (IIA-GCL)

In the general case, we develop a general contrastive learning (GCL) framework for IIA, based on the recently proposed NICA framework \[10\]. In IIA-GCL, we train a feature extractor and a logistic regression classifier, which discriminates a real dataset composed of the true observations of \((x_t, x_{t-1}, u_t)\), from a version where randomization is performed on \(u\). Thus we define two datasets

\[
\tilde{x} = (x_t, x_{t-1}, u_t) \text{ vs. } \tilde{x}' = (x_t, x_{t-1}, u^*_t),
\]

where \(u^*_t\) is a random value from the distribution of \(u\), but independent of \(x\), created in practice by random permutation of the empirical sample of \(u\). We learn a nonlinear logistic regression system using a regression function of the form

\[
r(\tilde{x}) = \sum_{i=1}^{n} \sum_{j=1}^{k} \psi_{ij}(h_i(x_t, x_{t-1})) \mu_{ij}(u_t) + \phi(x_{t-1}, u_t) + \alpha(u_t) + \beta(\tilde{h}(x_t, x_{t-1})) + \gamma(x_{t-1}),
\]

which gives the posterior probability of the first class \(\tilde{x}\) as \(1/(1 + \exp(-r(\tilde{x}))\). The scalar-valued functions \(\psi_{ij}, h_i, \mu_{ij}, \phi, \alpha, \beta,\) and \(\gamma\) take some of the \(x_t, x_{t-1}, u_t\) as input; note that the first
term has a special factorized form. The universal approximation capacity \[5\] is assumed for those functions; they would typically be learned by neural networks. This learning framework and the regression function are based on the following Theorem, proven in Supplementary Material \[A\].

**Theorem 1.** Assume the following:

1. We obtain observations from an NVAR model (Eq. 7), whose augmented model (Eq. 2) is invertible and sufficiently smooth.
2. The latent innovations of the process follow the assumptions \[A1\] and \[A2\] with \(k \geq 2\), and the sufficient statistics \(q_{ij}\) are twice differentiable.
3. (Assumption of Variability) There exist \(nk + 1\) distinct points \(u_0, \ldots, u_{nk}\) such that the matrix
   \[
   L = (\lambda(u_1) - \lambda(u_0), \ldots, \lambda(u_{nk}) - \lambda(u_0)) \tag{7}
   \]
   of size \(nk \times nk\) is invertible, where \(\lambda(u) = (\lambda_{11}(u), \ldots, \lambda_{nk}(u))^T \in \mathbb{R}^{nk}\).
4. We train a nonlinear logistic regression system with universal approximation capability to discriminate between \(\tilde{x}\) and \(\tilde{x}^*\) in Eq. 5 with regression function in Eq. 6.
5. The augmented function \(\hat{h}(x_t, x_{t-1}) = h(x_t, x_{t-1}), x_{t-1} : \mathbb{R}^{2n} \rightarrow \mathbb{R}^{2n}\) is invertible.
6. The scalar functions \(\psi_{ij}\) in Eq. 8 are twice differentiable, and for each \(i\), the following implication holds: \((\exists \theta \in \mathbb{R}^k | \forall y, \sum_{j=1}^{k} \psi_{ij}(y)\theta_j = \text{const}) \Rightarrow \theta = 0\).

Then, in the limit of infinite data, \(h\) in the regression function provides a consistent estimator of the IIA model: The functions \(h_i(x_t, x_{t-1})\) give the independent innovations, up to permutation and scalar (component-wise) invertible transformations.

This Theorem guarantees the convergence (consistency) of the learning algorithm. It immediately implies the identifiability of the innovations, up to a permutation and component-wise invertible nonlinearities. This kind of identifiability for innovations is stronger than any previous results in the literature. The estimation is based on the learning of nonlinear logistic regression function, and thus can be easily implemented based on ordinary neural network training. The Assumption of Variability requires the auxiliary variable \(u\) to have a sufficiently strong and diverse effect on the distributions of the innovations. The assumptions on the NVAR model are not too restrictive, and supposed to be satisfied in many applications. The assumption \[6\] indicates that \(\psi_{ij}\) are not functionally redundant; any \(\psi_{ij}\) cannot be represented by a linear combination of \(\psi_{il\neq j}\). Although the assumptions of the nonlinear functions to be trained (assumptions \[5\] and \[6\]) are not trivial, we assume they are only necessary to have a rigorous theory, and immaterial in any practical implementation.

### 3.2 Time-contrastive learning framework (IIA-TCL)

In the special case in which \(u_t\) is integer within a finite number of classes \([1, T]\), we can also develop a TCL-based framework for the estimation \[8\]. This special case includes time-segment-wise stationary process in which \(u_t\) represents the time segment index at time \(t\) \[8\].

Instead of the basic two-class logistic regression used in IIA-GCL, IIA-TCL uses a multinomial logistic regression (MLR) classifier for the learning. More specifically, we learn a nonlinear MLR using a softmax function which represents the posterior distribution of \(u_t\) by the form

\[
\pi(u_t = \tau | x_t, x_{t-1}) = \frac{\exp(\sum_{i=1}^{n} \sum_{j=1}^{k} w_{ij\tau} \psi_{ij}(h_i(x_t, x_{t-1})) + \phi(x_{t-1}, u_t = \tau) + b_{\tau})}{\sum_{l=1}^{T} \exp(\sum_{i=1}^{n} \sum_{j=1}^{k} w_{ijl} \psi_{ij}(h_i(x_t, x_{t-1})) + \phi(x_{t-1}, u_t = l) + b_{l})}, \tag{8}
\]

where \(w_{ij\tau}, b_{\tau}\) are the class-specific weight and bias parameters of the MLR, and \(\psi_{ij}, h_i\), and \(\phi\) are again scalar-valued functions assumed to have the universal approximation capacity \[5\]; they would typically be learned by neural networks. This learning framework and the regression function are justified on the following Theorem, proven in Supplementary Material \[B\].

**Theorem 2.** Assume the following:

1. We obtain observations from an NVAR model (Eq. 7), whose augmented model (Eq. 2) is invertible and sufficiently smooth.
2. The latent innovations of the process follow the assumptions $A_1$ and $A_2$ with $k \geq 2$, and the sufficient statistics $q_{ij}$ are twice differentiable.

3. The auxiliary variable $u$ is an integer in $[1, T]$, with $T$ the number of values it takes (classes).

4. (Assumption of Variability) The modulation matrix of size $nk \times (T - 1)$

$$L = (\lambda(2) - \lambda(1), \ldots, \lambda(T) - \lambda(1))$$

has full row rank $nk$, where $\lambda(\tau) = (\lambda_{11}(u = \tau), \ldots, \lambda_{nk}(u = \tau))^T \in \mathbb{R}^{nk}$.

5. We train a multinomial logistic regression with universal approximation capability to predict the class label (auxiliary variable) $u_t$ from $(x_t, x_{t-1})$ with regression function in Eq. 5.

6. The augmented function $\hat{h}(x_t, x_{t-1}) = [h(x_t, x_{t-1}), x_t] : \mathbb{R}^{2n} \rightarrow \mathbb{R}^{2n}$ is invertible.

7. The scalar functions $\psi_{ij}$ in Eq. 3 are twice differentiable, and for each $i$, the following implication holds: $(\exists \theta \in \mathbb{R}^k | \forall y, \sum_{j=1}^k \psi_{ij}(y)\theta_j = \text{const}) \Rightarrow \theta = 0$.

Then, in the limit of infinite data in each class, $h$ in the regression function provides a consistent estimator of the IIA model: The functions $h_t(x_t, x_{t-1})$ give the independent innovations, up to permutation and scalar (component-wise) invertible transformations.

Many of the assumptions are the same as those in IIA-TCL, except for the specifics of the innovation model (assumptions 3 and 4), and the learning algorithm (assumption 5). The estimation is based on self-supervised nonlinear MLR, and thus can be easily implemented based on ordinary neural network training, like IIA-GCL. Although the estimation methods are different, the identifiability result implied here by IIA-TCL is the same as above by IIA-GCL. Note that here the limit of infinite data takes the form that each class (value of $T$) has an infinite number of data points. In practice, each class is thus required to have a sufficient number of samples, so $T$ needs to be much smaller than the total number of data points; this would be natural if $T$ is a segment index.

4 Experiments

4.1 Simulation 1: IIA-GCL for artificial dynamics with nonstationary innovations

Data generation We generated data from an artificial NVAR process with nonstationary innovations. The innovations were randomly generated from a Gaussian distribution by modulating its mean and standard deviation across time $t$, i.e., $u_t = t$. The modulations were designed to be temporally smooth and continuous. The dimensions of the observations and innovations ($n_t$) were 20. As the NVAR model, we used a multilayer perceptron we call NVAR-MLP, which takes a concatenation of $x_{t-1}$ and $s_t$ as an input, then outputs $x_t$ (see Supplementary Material C for more details of the experimental settings). The goal of this simulation is to estimate the innovations $s$ only from the observable time series $x$, without knowing the parameters of the NVAR-MLP.

Training Considering the innovation model with $u_t = t$, we here used IIA-GCL for the estimation of the latent innovations. We adopted MLPs as the nonlinear scalar functions in Eq. 6. The nonlinear regression function was trained by back-propagation with a momentum term so as to discriminate the real dataset from its $u_t$-randomized version. (As a simple sanity check, we saw that it achieved higher classification accuracies than chance after the training, see Fig. 3 in Supplementary Material.) The performance was evaluated by the Pearson correlation between the true innovations and the estimated feature values $\hat{h}$. It was averaged over 10 runs, for each setting of the complexity (number of layers) $L \in [1, 3, 5]$ of the NVAR-MLP and the number of data points. For comparison, we also applied NICA based on GCL (NICA-GCL [10]), an NVAR with additive innovation model (AD-NVAR), and variational autoencoder (VAE) [16] to the same data. We fixed $L \in [1, 2]$ exceptionally for VAE because of the instability of training in high layer models. We additionally applied linear ICA [7] to the estimations by AD-NVAR and VAE for fair comparisons.

Result The IIA-GCL framework could reconstruct the innovations reasonably well even for the nonlinear mixture cases ($L > 1$) (Fig. 1b). We can see that a larger amount of data make it possible to achieve higher performance, and higher complexity of the NVAR model makes learning more difficult.
Figure 1: (Simulation) Estimation of the latent innovations from unknown artificial NVAR process by IIA. (a) (Simulation 1; IIA-GCL) Mean absolute correlation coefficients between innovations and their estimates by IIA-GCL (solid lines), with different settings of the complexity (number of layers $L$) of the NVAR models and data points. For comparison: NICA based on GCL (NICA-GCL, dashed line), NVAR with additive innovation model (AD-NVAR, dotted line), and variational autoencoder (VAE, dash-dot line). IIA-GCL generally has higher correlations than the baseline methods. (b) (Simulation 2; IIA-TCL) Mean absolute correlation coefficients between innovations and their estimates by the IIA-TCL framework (solid lines), evaluated by the same data used in Simulation 1. For comparison: NICA based on TCL (NICA-TCL, dashed line) and IIA-GCL shown in a (dotted line).

AD-NVAR performed well for the linear mixture case ($L = 1$) because the additive innovation model is equivalent to the general NVAR model in the linear case; however, it was much worse in the nonlinear case. As expected, the other methods performed worse than IIA-GCL because their model did not match well to the NVAR generation model.

4.2 Simulation 2: IIA-TCL for artificial dynamics with nonstationary innovations

Training Next, to evaluate the IIA-TCL framework, we applied it to the same data used in Simulation 1. For IIA-TCL, we first divided the time series into 256 equally-sized segments, and used the segment label as the auxiliary variable $u_t$; i.e., we assume that the data are segment-wise stationary. Although this assumption is not consistent with the real innovation model (Section 4.1), it is approximately true because the modulations were temporally smooth and continuous; we thus consider here data with a realistic deviation from model assumptions. We adopted MLPs as the nonlinear scalar functions in Eq. 8, which architectures were similar to those in Simulation 1. The training and evaluation methods follow those in Simulation 1. (Again as a sanity check, we saw that the MLR achieved higher classification accuracies than chance after the training, see Fig. 3b in Supplementary Material.) We discarded the cases of small data sets ($2^{10}$ and $2^{12}$) because of the instability of training. For comparison, we also applied NICA based on TCL (NICA-TCL [8]). See Supplementary Material [D] for more details of the training settings.

Result IIA-TCL performed better than NICA-TCL (Fig. 1b). In addition, even though the innovation model matches IIA-GCL better than IIA-TCL (the modulations are temporally smooth and continuous, and thus not segment-wise stationary), IIA-TCL achieved slightly better performances than IIA-GCL; this finding is consistent with the comparison of NICA-GCL and NICA-TCL by [10]. As with IIA-GCL, a larger number of data points leads to higher performance (i.e. the method seems to converge), and again, higher complexity of the NVAR models makes learning more difficult.

4.3 Experiments on real brain imaging data

To evaluate the applicability of IIA to real data, we applied it on multivariate time series of electrical activities of the human brain, measured by magnetoencephalography (MEG). In particular, we used a dataset measured during auditory or visual presentations of words [31]. Although ICA is often used to analyze brain imaging data, relying on the assumption of mutual independence of the hidden components, the event-related components (such as event-related potentials; ERPs) are not likely to be independent because they may have similar temporal patterns time-locked to the stimulation. However, the innovations generating the components should still be independent because
Figure 2: IIA-TCL on the electrical activity data measured by MEG from the human brain during auditory or visual stimuli of German nouns. (a) Decoding accuracies of the stimulus category predicted from the innovations extracted by IIA-TCL and the other baseline methods. The performance was measured by one-subject-out cross-validation (OSO-CV), with changing the number of layers $L$ for each method. Each point represents a testing accuracy on a target subject. The black horizontal line indicates the chance level. (b) The temporal pattern and the spatial specificity of each component trained by IIA-TCL ($L = 3$). (Left panel) The temporal patterns of the components averaged separately for auditory and visual trials (orange=auditory, blue=visual). 0 s is the onset of the stimulus, and the latter vertical line represents the average duration of the stimuli. (Right panel) The spatial topographies of the optimal input (MEG signal; top view) which maximizes (+) and minimizes (−) the component.

they would be generated by different brain sources, which motivates us to use IIA rather than ICA (see Supplementary Material E for the details of the data and settings).

Data and preprocessing We used a publicly available MEG dataset [31]. Briefly, the participants were presented with a random word selected from 420 unrelated German nouns either visually or auditorily, randomly for each trial. MEG signals were measured from twenty healthy volunteers by a 148-channel magnetometer (219.1±22.4 trials for each subject; 2,207 auditory and 2,174 visual trials in total for all subjects). We band-pass filtered the data between 4 Hz and 125 Hz (sampling frequency = 300 Hz). The dimension of the data was reduced to 30 by PCA.

IIA settings We used IIA-TCL for the training, by assuming an NVAR(3) model and the segment-wise-stationarity of the latent innovations. The trial data were divided into 84 equally sized segments of length of 8 samples (26.7 ms), and the segment label was used as the auxiliary variable $u_t$. The same segment labels were given across the trials; however, considering the possible stimulus-specific dynamics of the brain, we assigned different labels for the auditory and visual trials. In total, there are 168 segments (classes) to be discriminated by MLR. We used MLPs for the nonlinear scalar functions (Eq. 8), and fixed the number of components to 5. Considering the fast sampling rate of the data, we fixed the time lag between two consecutive samples to 3 (10 ms).

Evaluation methods For evaluation, we performed classification of the stimulus modality (auditory or visual) by using the estimated innovations. The classification was performed using a linear support vector machine (SVM) classifier trained on the stimulation label and sliding-window-averaged innovations (width=16 and stride=8 samples) obtained for each trial. The performance was evaluated by the generalizability of a classifier across subjects, i.e., one-subject-out cross-validation (OSO-CV); the feature extractor and the classifier were trained only from the training subjects, and then applied to the held-out subject. For comparison, we also evaluated NICA based on TCL [8] and AD-NVAR(3). We omitted $L = 1$ for IIA-TCL because of the instability of training. We visualized the spatial characteristics of each innovation component by estimating the optimal (maximal and minimal) input $x_t$ while fixing $x_{t-1}$ to zero.

Results Figure 2a shows the decoding accuracies of the stimulus categories, across different methods and the number of layers for each model. The performances by IIA-TCL with nonlinear
models ($L \geq 2$) were significantly higher than the other baseline methods ($p < 0.05$; Wilcoxon signed-rank test, FDR correction), which indicates the importance of the modeling of the MEG signals by NVAR, especially with the nonlinear (non-additive) interactions of the innovations.

The left panels of Fig. 2b show the temporal patterns of the innovations during the auditory and visual stimuli. Some components have clear differences between the stimulus modalities, which implies that those components are related to the stimulus-specific dynamics of the brain; e.g., C1 and C2 represent auditory- and visual-relevant innovations, respectively. Such stimulus-specificity can be also seen from the spatial characteristics of the components; C1 is strongly activated by the MEG signals around auditory areas of the brain, while C2 is more activated by the visual areas. C3 seems to represent stimulus-evoked activities on the parietal region caused by both categories. Those results show that IIA-TCL extracted reasonable components (innovations) relevant to the external stimuli automatically from the data in a self-supervised, data-driven manner.

5 Discussion

IIA can be seen as a generalization of the recently proposed NICA frameworks [8, 10], with the important difference that observations can have recurrent temporal structure. The theory strictly includes NICA as a special case, since the main assumptions can be satisfied even if the NVAR model (Eq. 1) does not actually depend on $x_{t-1}$, which corresponds to the instantaneous nonlinear mixture model of NICA: $x_t = f(s_t)$. This connection can be also seen by comparing the regression functions; by omitting the dependencies of Eqs. 6 and 8 on $x_{t-1}$, we can obtain the same algorithms as NICA ([8] with k=1, and [10]). This indicates that the regression functions of IIA can learn NICA models as a special case.

Applying IIA on time series data has some practical advantages compared to NICA. First, autoregressive structures are generally inherent in any kinds of dynamics, and their explicit modeling is beneficial for the estimation. Second, innovations are usually more independent mutually than the processes generated by them, because the independence of processes implies the independence of their innovations, but not vice versa, as argued in the linear case by [12]. Thus, innovations are likely to give a better fit to any model assuming independence of the latent variables.

While IIA estimates innovations from the observed time series, the NVAR model $f$ is left unknown, unlike in ordinary VAR analyses. In practice, we can estimate $f$ after IIA as a post-processing, by fitting a nonlinear function which outputs $x_t$ from $x_{t-1}$ and the estimated $s_t$. Since IIA guarantees the estimation of $s$ up to a permutation and element-wise invertible nonlinearities, this should be possible if the model to be fitted has universal approximation capability.

6 Conclusion

We proposed independent innovation analysis (IIA) as a new general framework to nonlinearly extract innovations hidden in a time series. In contrast to the common simplifying assumption of additive innovations, IIA can deal with a general nonlinear VAR model in which innovations are not additive. Any general nonlinear interactions between the innovations and the observations are allowed. To guarantee identifiability, IIA requires some assumptions on the innovations, in particular mutual independence conditionally on a fully observable auxiliary variable which also needs to modulate the distributions of the innovations. A typical case would be nonstationary innovations mutually independent at each time point.

We proposed two practical estimation methods, both of which were based on a self-supervised training of a nonlinear feature extractor by logistic regression, possibly multinomial. They can thus be easily implemented by ordinary neural network training. In both methods, the consistency of the estimation is guaranteed up to a permutation and component-wise invertible nonlinearity, which implies the strongest identifiability proof of general NVAR in the literature, by far. IIA can be seen as a generalization of recently proposed NICA frameworks, and includes them as special cases.

Experiments on real brain imaging data by MEG showed distinctive components relevant to the external-stimulus categories. This result suggests a wide applicability of the method to different kinds of time series such as video data, econometric data, and biomedical data, in which innovation plays an important role.
Acknowledgments and Disclosure of Funding

This research was supported in part by JSPS KAKENHI 18KK0284, 19K20355, and 19H05000. A.H. was funded by a Fellow Position from CIFAR as well as the DATAIA convergence institute as part of the “Programme d’Investissement d’Avenir”, (ANR-17-CONV-0003) operated by Inria.

References

[1] C. Finn, I. Goodfellow, and S. Levine. Unsupervised learning for physical interaction through video prediction. In Advances in Neural Information Processing Systems (NIPS) 29, pages 64–72. 2016.

[2] G. Gómez-Herrero, M. Atienza, K. Egiazarian, and J.L. Cantero. Measuring directional coupling between eeg sources. NeuroImage, 43(3):497 – 508, 2008.

[3] M. U. Gutmann and A. Hyvärinen. Noise-contrastive estimation of unnormalized statistical models, with applications to natural image statistics. Journal of Machine Learning Research, 13(11):307–361, 2012.

[4] T. Hastie, R Tibshirani, and J. Friedman. The Elements of Statistical Learning. Springer, New York, NY, 2001.

[5] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. Neural Networks, 2(5):359 – 366, 1989.

[6] A. Hyvärinen. Fast and robust fixed-point algorithms for independent component analysis. IEEE Trans. Neural Netw., 10(3):626–634, 1999.

[7] A. Hyvarinen. Blind source separation by nonstationarity of variance: a cumulant-based approach. IEEE Transactions on Neural Networks, 12(6):1471–1474, 2001.

[8] A. Hyvärinen and H. Morioka. Unsupervised feature extraction by time-contrastive learning and nonlinear ica. In Advances in Neural Information Processing Systems (NIPS) 29, pages 3765–3773, 2016.

[9] A. Hyvärinen and P. Pajunen. Nonlinear independent component analysis: Existence and uniqueness results. Neural Netw., 12(3):429 – 439, 1999.

[10] A. Hyvarinen, H. Sasaki, and R. Turner. Nonlinear ica using auxiliary variables and generalized contrastive learning. In AISTATS, pages 859–868, 2019.

[11] A. Hyvärinen, K. Zhang, S. Shimizu, and P. O. Hoyer. Estimation of a structural vector autoregression model using non-gaussianity. Journal of Machine Learning Research, 11(56):1709–1731, 2010.

[12] Aapo Hyvärinen. Independent component analysis for time-dependent stochastic processes. In ICANN 98, pages 135–140, 1998.

[13] I. Jeliazkov. Nonparametric Vector Autoregressions: Specification, Estimation, and Inference, volume 32, pages 327–359. Emerald Group Publishing Limited, 2013.

[14] M. Kalli and J. E. Griffin. Bayesian nonparametric vector autoregressive models. Journal of Econometrics, 203(2):267 – 282, 2018.

[15] I. Khemakhem, D. P. Kingma, R. P. Monti, and A. Hyvärinen. Variational autoencoders and nonlinear ica: A unifying framework. In AISTATS, 2020.

[16] D. P Kingma and M. Welling. Auto-encoding variational bayes. In ICLR 2014, 2014.

[17] G. Koop and D. Korobilis. Bayesian multivariate time series methods for empirical macroeconomics. Found. Trends Econ., 3(4):267–358, 2010.

[18] M. Lanne, M. Meitz, and P. Saikkonen. Identification and estimation of non-gaussian structural vector autoregressions. Journal of Econometrics, 196(2):288 – 304, 2017.

[19] W. Lotter, G. Kreiman, and D. Cox. Deep predictive coding networks for video prediction and unsupervised learning. In ICLR 2017, 2017.

[20] A. Moneta, D. Entner, P. O. Hoyer, and A. Coad. Causal inference by independent component analysis: Theory and applications. Oxford Bulletin of Economics and Statistics, 75(5):705–730, 2013.

[21] J. Oh, X. Guo, H. Lee, R. L Lewis, and S. Singh. Action-conditional video prediction using deep networks in atari games. In Advances in Neural Information Processing Systems (NIPS) 28, pages 2863–2871, 2015.
[22] G. E. Primiceri. Time varying structural vector autoregressions and monetary policy. *The Review of Economic Studies*, 72(3):821–852, 2005.

[23] Y. Shen, G. B. Giannakis, and B. Baingana. Nonlinear structural vector autoregressive models with application to directed brain networks. *IEEE Transactions on Signal Processing*, 67(20):5325–5339, 2019.

[24] C. A. Sims. Macroeconomics and reality. *Econometrica*, 48(1):1–48, 1980.

[25] B. Sriperumbudur, K. Fukunizu, A. Gretton, A. Hyvärinen, and R. Kumar. Density estimation in infinite dimensional exponential families. *Journal of Machine Learning Research*, 18(57):1–59, 2017.

[26] N. Srivastava, E. Mansimov, and R. Salakhutdinov. Unsupervised learning of video representations using lstms. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37*, ICML’15, pages 843–852, 2015.

[27] T. Teräsvirta. Specification, estimation, and evaluation of smooth transition autoregressive models. *Journal of the American Statistical Association*, 89(425):208–218, 1994.

[28] R. S. Tsay. Testing and modeling multivariate threshold models. *Journal of the American Statistical Association*, 93(443):1188–1202, 1998.

[29] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu. Wavenet: A generative model for raw audio. *CoRR*, abs/1609.03499, 2016.

[30] R. Villegas, J. Yang, S. Hong, X. Lin, and H. Lee. Decomposing motion and content for natural video sequence prediction. In *ICLR 2017*, 2017.

[31] B. U. Westner, S. S. Dalal, S. Hanslmayr, and T. Staudigl. Across-subjects classification of stimulus modality from human meg high frequency activity. *PLOS Computational Biology*, 4(3):1–14, 2018.

[32] N. Wichers, R. Villegas, D. Erhan, and H. Lee. Hierarchical long-term video prediction without supervision. In *Proceedings of the 35th International Conference on Machine Learning*, volume 80, pages 6038–6046, 2018.
Supplementary Material

A Proof of Theorem 1

The log-pdf of $\tilde{x}$ is given by, using the probability transformation formula,

$$
\log p(\tilde{x}(t)) = \log p(x_t, x_{t-1}, u_t)
$$

$$
= \log p_k(\tilde{g}(x_t, x_{t-1})|u_t) + \log p(u_t) + \log |\det Jg(x_t, x_{t-1})|
$$

$$
= \log p_k(\tilde{g}(x_t, x_{t-1}), x_{t-1}|u_t) + \log p(u_t) + \log |\det Jg(x_t, x_{t-1})|
$$

$$
= \log p_k(g(x_t, x_{t-1})|u_t) + \log p(x_t|x_{t-1}|u_t) + \log p(u_t) + \log |\det Jg(x_t, x_{t-1})|
$$

(10)

where $p_k$, $p_u$, and $p_x$ are the conditional pdfs of $(s, \tilde{x})$, $s$, and $x$, respectively, $J$ denotes the Jacobian, and $s_t = g_t(x_t, x_{t-1})$ by definition. The third equality is from the structure of the augmented demixing model (Eq. [5]), and the last equation is from the temporal independence of $s_t$ (A1).

By well-known theory [3][4], after convergence of logistic regression, with infinite data and a function approximator with universal approximation capability, the regression function (Eq. 6) will equal the difference of the log-pdfs in the two classes:

$$
\sum_{i=1}^{n} \sum_{j=1}^{k} \psi_{ij}(h_i(x_t, x_{t-1})) \mu_{ij}(u_t) + \phi(x_{t-1}, u_t) + \alpha(u_t) + \beta(h(x_t, x_{t-1})) + \gamma(x_{t-1})
$$

$$
= \log p_k(g(x_t, x_{t-1})|u_t) + \log p_{x}(x_{t-1}|u_t) + \log p(u_t) + \log |\det Jg(x_t, x_{t-1})|
$$

$$
- \log p_k(g(x_t, x_{t-1}) - log p_k(x_{t-1}) - log p(u_t) - log |\det Jg(x_t, x_{t-1})|
$$

$$
= \sum_{i=1}^{n} Q_i(g_i(x_t, x_{t-1})) - Z_i(u_t) + \sum_{j=1}^{k} Q_{ij}(g_i(x_t, x_{t-1})) \lambda_{ij}(u_t) + \log p_{x}(x_{t-1}|u_t)
$$

$$
- \log p_k(g(x_t, x_{t-1}) - log p_k(x_{t-1})
$$

(11)

where $p_k$ and $p_x$ are the marginal pdfs of the innovations and observations when $u$ is integrated out, and the last equation came from the conditional exponential pdf model of $s$ (A1). The Jacobians and marginals $log p(u)$ cancel out here. Considering its factorization form and the distinctive dependency of each term on $x_t$, $x_{t-1}$, and $u_t$, the approximation solution is possible as

$$
\psi_{ij}(h_i(x_t, x_{t-1})) = q_{ij}(g_i(x_t, x_{t-1}))
$$

$$\mu_{ij}(u_t) = \lambda_{ij}(u_t)
$$

$$\phi(x_{t-1}, u_t) = \log p_k(x_{t-1}|u_t)
$$

$$\alpha(u_t) = -\sum_{i=1}^{n} Z_i(u_t)
$$

$$\beta(h(x_t, x_{t-1})) = \sum_{i=1}^{n} Q_i(g_i(x_t, x_{t-1})) - log p_k(g(x_t, x_{t-1}))
$$

$$\gamma(x_{t-1}) = -log p_k(x_{t-1}).
$$

(12)

Next, we have to prove that this is the only solution up to the indeterminacies given in the Theorem. Let $u_0, \ldots, u_{nk}$ be the points given by assumption [3] in the Theorem. We plug each of those $u_t$ to obtain $nk + 1$ equations. By collecting those equations into rows, with subtracting the first equation for $u_0$ from the remaining $nk$ equations:

$$
M^T \psi(h(x_t, x_{t-1})) + \phi(x_{t-1}) + \alpha = L^T q(s_t) + p(x_{t-1}) + z,
$$

(13)

where $M \in \mathbb{R}^{nk \times nk}$ is a matrix of $\mu_{ij}(u_t) - \mu_{ij}(u_0)$, with the product of $i, j$ giving row index and $l$ column index, $L$ is a matrix of $\lambda_{ij}(u_t) - \lambda_{ij}(u_0)$ given in the assumption [3] in the Theorem, $\psi(h(x_t, x_{t-1})) = (\psi_{11}(h_1(x_t, x_{t-1})), \ldots, \psi_{nk}(h_{nk}(x_t, x_{t-1})))^T$, $q(s_t) = (q_{11}(s_1(t)), \ldots, q_{nk}(s_{nk}(t)))^T$, and the other vectors are the collection of the corresponding terms in Eq. [11] at the $nk$ points with all subtracting the one with $l = 0$; $\phi(x_{t-1}) = (\phi(x_{t-1}, u_1), \ldots, \phi(x_{t-1}, u_{nk}))^T - 1\phi(x_{t-1}, u_0)$, $1$ is a $nk \times 1$ vector of ones,
\[ \alpha = (\alpha(u_1), \ldots, \alpha(u_{nk}))^T - 1\alpha(u_0), p(x_{t-1}) = (\log p_x(x_{t-1} | u_1), \ldots, \log p_x(x_{t-1} | u_{nk}))^T - 1 \log p_x(x_{t-1} | u_0), \text{ and } z = (-\sum_{i=1}^h Z_i(u_1), \ldots, -\sum_{i=1}^h Z_i(u_{nk}))^T + 1 \sum_{i=1}^h Z_i(u_0). \]

In both sides of the equation, the terms not depending on \( u_i \) disappear by the subtraction with \( l = 0 \).

Let a compound demixing-mixing function \( v(s_i, x_{t-1}) = h \circ f(s_i, x_{t-1}) \), and change variables to \( y = [y_1, y_2] = [s_i, x_{t-1}] \), we then have

\[ M^T \phi(v(y)) + \phi(y_2) + \alpha = L^T q(y_1) + p(y_2) + z. \]

Firstly, we will show that \( M \) is invertible. From the definition of \( q(y_j) \), its partial derivative with respect to \( y_{1i} \) is \( q'(y_{1i}) = (0, \ldots, 0, q'_{1i}(y_{1i}), 0, \ldots, 0)^T \). According to Lemma 3 of [15], for \( y_{1i} \), which satisfies \( A_2 \), there exist \( k \) points \( \{\tilde{y}_{1i}^1, \ldots, \tilde{y}_{1i}^k\} \) such that \( q'(\tilde{y}_{1i}^1), \ldots, q'(\tilde{y}_{1i}^k) \) are linearly independent. By differentiating Eq. (14) with respect to \( y_{1i} \) and collecting their evaluations at such \( k \) distinctive points for all \( i \), we get

\[ M^T \tilde{Q} = L^T Q, \]

where \( Q \in \mathbb{R}^{nk \times nk} \) is a matrix collecting \( q'(\tilde{y}_{1i}^j) \) to the columns indexed by \( (i, l) \), and \( Q \) is a collection of partial derivatives of \( \phi(v(y)) \) evaluated at the same points. \( Q \) is invertible (through a combination of Lemma 3 of [15] and the fact that each component of \( Q \) is univariate), and thus the right-hand side is invertible because \( L \) is invertible (as assumption [5]). The invertibility of the right-hand side implies the invertibility of \( M \) and \( \tilde{Q} \).

Now, let an augmented compound demixing-mixing function \( \tilde{v}(y) = [\tilde{v}_1(y), \tilde{v}_2(y)] = \tilde{h} \circ \tilde{f}(y) \), where \( \tilde{h} \) is the augmented function defined in the assumption [5] in the Theorem. The \( \tilde{v}_1(y) \) corresponds to \( v(y) \) defined above. Note that \( \tilde{v} \) is invertible because both \( \tilde{h} \) and \( \tilde{f} \) are invertible. What we need to prove is that \( v \) is a block-wise invertible point-wise function, in the sense that \( \tilde{v}_1 \) is a function of only one \( y_{11} \), and not of any of \( y_{12} \), and vice versa. This can be done by showing that the product of two distinct partial derivatives of any component is always zero, and the Jacobian \( J_v \in \mathbb{R}^{2nk \times 2nk} \) is block diagonal; the upper and lower block correspond to \( y_1 \) and \( y_2 \) respectively. Along with invertibility, this means that each component depends exactly on one variable of the corresponding block \( (y_1 \text{ or } y_2) \). Below, we show that separately for \( J_v \in \mathbb{R}^{n \times 2n} \) and \( J_{y'}, \in \mathbb{R}^{n \times 2n} \), Firstly, this is obviously true for \( J_{y'} \) because \( y_{11} \) is just an identity mapping of \( y_2 \) from the definitions of \( \tilde{h} \) and \( \tilde{f} \), and does not depend on \( y_1 \); the lower non-zero block of \( J_{y'} \) is an identity matrix. Next, we will show that for \( J_v \). We differentiate Eq. (14) with respect to \( y_{1c} \), \( 1 \leq c \leq n \) (an element of \( y_1 = s_t \)), and \( y_{dt}, c < d \leq 2n \), and get

\[ M^T \frac{\partial^2}{\partial y_{1c} \partial y_{dc}} \psi(v(y)) = 0. \]

From the invertibility of \( M \) and the calculation of differentials, we get

\[ \frac{\partial^2}{\partial y_{1c} \partial y_{dc}} \psi(v(y)) = \Psi(y)^T \psi(y) = 0, \]

where \( \Psi(y) = (e(1,1)(y_1), \ldots, e(1,k)(y_1), \ldots, e(n,1)(y_n), \ldots, e(n,k)(y_n)) \in \mathbb{R}^{2nk \times nk}, e(a,b) = (0, \ldots, 0, \psi_{ab}(v_{1a}), \psi_{ab}(v_{2a}), 0, \ldots, 0)^T \in \mathbb{R}^{2n}, \) such that the non-zero entries are at indices \( (2a - 1, 2a) \), \( \psi(y) = (v_{c1}(y)v_{d1}(y), \ldots, v_{cn}(y)v_{dn}(y), v_{c1}(y)v_{dn}(y)-v_{cn}(y)v_{d1}(y)) \in \mathbb{R}^{2n}, \)

\[ v_{ci}^a = \frac{\partial v_{ci}}{\partial y_{ic}}(y), \text{ and } v_{ci}^a = \frac{\partial v_{ci}^a(y)}{\partial y_{ic}}(y). \]

From Lemma 4 and 5 of [15], assumption [5] implies that \( \Psi(y) \) has full row rank \( 2n \), and thus the pseudo-inverse of \( \Psi(y)^T \) fulfills \( \Psi(y)^T \Psi(y)^T = I \). We multiply the equation above from the left by such pseudo-inverse and obtain

\[ v(y) = 0. \]

In particular, \( v_{ci}^a(y)v_{di}^a(y) = 0 \) for all \( 1 \leq a \leq n, 1 \leq c \leq n, \) and \( c < d \leq 2n \). This means that a row of \( J_v \in \mathbb{R}^{n \times 2n} \) at each \( y \) has either 1) only one non-zero entry somewhere in the former half block (corresponding to the partial derivatives by \( y_1 \)), or 2) non-zero entries only in the latter half block (corresponding to the partial derivatives by \( y_2 \)). The latter case is contradictory because it means that the component \( v_{ci} \) is a function of only \( y_{1c} = x_{t-1} \), and cannot hold Eq. (14) which right-hand side is a function of all components of \( y_1 \) (and \( y_2 \)). Therefore, \( J_v \) should have only one non-zero entry in the former half block for each row. From the results of \( J_v \) and \( J_{y'} \), we deduce that \( J_v \) is a block diagonal matrix. Now, by invertibility and continuity of \( J_v \), we deduce that the location of the non-zero entries are fixed and do not change as a function of \( y \). This proves that \( v = h \circ f(y) \) is a block-wise invertible point-wise function, and \( v_i := h_i(x_t, x_{t-1}) \) is represented by only one \( y_{1j_i} \) \( (= s_j(t)) \) up to a scalar (component-specific) invertible transformation, and the Theorem is proven.
B Proof of Theorem 2

The conditional joint log-pdf of a data point \((x_t, x_{t-1})\) is given by, using the probability transformation formula,

\[
\log p(x_t, x_{t-1} | u_t = \tau) = \sum_{i=1}^{n} \sum_{j=1}^{k} \left( w_{ij\tau} - w_{ij1} \right) \psi_{ij} (x_t, x_{t-1}) + q_{ij} (g_i(x_t, x_{t-1})) \lambda_{ij} (u_t = 1) \\
+ \sum_{i=1}^{n} \left[ Q_i (g_i(x_t, x_{t-1})) - Z_i (u_t) + \sum_{j=1}^{k} q_{ij} (g_i(x_t, x_{t-1})) \lambda_{ij} (u_t) \right] \\
+ \log p_x(x_{t-1} | u_t) + \log | \det J(x_t, x_{t-1}) |
\]

where \(p_x, p_s, \) and \(p_e\) are the conditional pdfs of \((s, x), s, \) and \(x, \) respectively, \(J\) denotes the Jacobian, and \(s_t = g_t(x_t, x_{t-1})\) by definition. The second equation is from the structure of the augmented demixing model (Eq. 6) and the temporal independence of \(s (A1), \) and the last equation is from the conditional exponential family model of the innovation \( (A2) \). On the other hand, by applying Bayes rule on the optimal discrimination relation given by Eq. 8 after dividing all the exponential term by the one of \( \tau = 1 \) to avoid the well-known indeterminacy of the softmax function,

\[
\log p(x_t, x_{t-1} | u_t = \tau) = \sum_{i=1}^{n} \sum_{j=1}^{k} \left( w_{ij\tau} - w_{ij1} \right) \psi_{ij} (h_i(x_t, x_{t-1})) + \phi(x_{t-1}, u_t = \tau) \\
- \phi(x_{t-1}, u_t = 1) + \log p(x_t, x_{t-1} | u_t = 1) + \alpha_{\tau}, \tag{20}
\]

where \(\alpha_{\tau} = b_{\tau} - b_1 - \log p(u_t = \tau) + \log p(u_t = 1).\) Substituting Eq. 19 with \(u_t = 1\) into Eq. 20, we have:

\[
\log p(x_t, x_{t-1} | u_t = \tau) = \sum_{i=1}^{n} \sum_{j=1}^{k} \left( w_{ij\tau} - w_{ij1} \right) \psi_{ij} (h_i(x_t, x_{t-1})) + q_{ij} (g_i(x_t, x_{t-1})) \lambda_{ij} (u_t = 1) \\
+ \sum_{i=1}^{n} \left[ Q_i (g_i(x_t, x_{t-1})) - Z_i (u_t) + \phi(x_{t-1}, u_t = \tau) - \phi(x_{t-1}, u_t = 1) \right] \\
+ \log p_x(x_{t-1} | u_t = 1) + \log | \det J(g(x_t, x_{t-1})) | + \alpha_{\tau} \tag{21}
\]

Setting Eq. 21 and Eq. 19 with \(u_t = \tau\) to be equal for arbitrary \(\tau,\) we have:

\[
\sum_{i=1}^{n} \sum_{j=1}^{k} (w_{ij\tau} - w_{ij1}) \psi_{ij} (h_i(x_t, x_{t-1})) + \phi(x_{t-1}, u_t = \tau) - \phi(x_{t-1}, u_t = 1) + \alpha_{\tau} \\
= \sum_{i=1}^{n} \sum_{j=1}^{k} (\lambda_{ij} (u_t = \tau) - \lambda_{ij} (u_t = 1)) q_{ij} (g_i(x_t, x_{t-1})) + \log p_x(x_{t-1} | u_t = \tau) - \log p_x(x_{t-1} | u_t = 1) + z_{\tau} \tag{22}
\]

where \(z_{\tau} = \sum_{i=1}^{n} Z_i (u_t = 1) - Z_i (u_t = \tau).\) By collecting this equation for all the \(T\) labels into rows, except \(\tau = 1,\) which makes both-sides zero,

\[
W^T \psi(h(x_t, x_{t-1})) + \phi(x_{t-1}) + \alpha = L^T q(s_t) + p(x_{t-1}) + z, \tag{23}
\]

where \(W \in \mathbb{R}^{nk \times (T-1)}\) is a matrix of \(w_{ij\tau} - w_{ij1},\) with the product of \(i, j\) giving row index and \(\tau\) column index, \(L\) is a matrix of \(\lambda_{ij} (u_t = \tau) - \lambda_{ij} (u_t = 1)\) in the assumption 4 in the Theorem, \(\psi(h(x_t, x_{t-1})) = (\psi_{11}(h_1(x_t, x_{t-1})), \ldots, \psi_{nk}(h_n(x_t, x_{t-1})))^T,\)

\(q(s_t) = (q_{11}(s_1(t)), \ldots, q_{nk}(s_n(t)))^T,\) \(\phi(x_{t-1}) = (\phi(x_{t-1}, u_t = 2), \ldots, \phi(x_{t-1}, u_t = T))^T - 1 \phi(x_{t-1}, u_t = 1),\) \(1\) is a \((T - 1) \times 1\) vector of ones, \(\alpha = (\alpha_2, \ldots, \alpha_T)^T,\)

\(p(x_{t-1}) = (\log p_x(x_{t-1} | u_t = 2), \ldots, \log p_x(x_{t-1} | u_t = T))^T - 1 \log p_x(x_{t-1} | u_t = 1),\) and \(z = (z_2, \ldots, z_T)^T.\) Let a compound demixing-mixing function \(v(s_t, x_{t-1}) = h \circ f(s_t, x_{t-1}),\) and change variables to \(y = [y_1, y_2] = [s_t, x_{t-1}],\) we then have

\[
W^T \psi(v(y)) + \phi(y_2) + \alpha = L^T q(y_1) + p(y_2) + z. \tag{24}
\]
Firstly, we will show that $W$ has full row rank $nk$. From the definition of $q(y_1)$, its partial derivative with respect to $y_{1i}$ is $q'(y_{1i}) = (0, \ldots, 0, q_{1i}^t(y_{1i}), \ldots, q_{ik}^t(y_{1i}), 0, \ldots, 0)'$. According to Lemma 3 of [13], for $y_{1i}$ which satisfies $\mathbf{X}$, there exist $k$ points $(\bar{y}_{1i}^1, \ldots, \bar{y}_{1i}^k)$ such that $(q'(\bar{y}_{1i}^1), \ldots, q'(\bar{y}_{1i}^k))$ are linearly independent. By differentiating Eq. 24 with respect to $y_{1i}$ and collecting their evaluations at each $i$, we get

$$W^T \bar{Q} = L^T \bar{Q},$$

where $Q \in \mathbb{R}^{nk \times nk}$ is a matrix collecting $q'(\bar{y}_{1i}^l)$ to the columns indexed by $(i, l)$, and $\bar{Q}$ is a collection of partial derivatives of $\psi(v(y))$ evaluated at the same points. $Q$ is invertible (through a combination of Lemma 3 of [13] and the fact that each component of $q$ is univariate), and thus the right-hand side has full column rank $nk$ because $L$ has full row rank $nk$ (assumption $\text{H}$). The full column rank of the right-hand side implies the full row rank of $W$ and the invertibility of $Q$.

Now, let an augmented compound demixing-mixing function $\tilde{v}(y) = [\tilde{v}_1(y), \tilde{v}_2(y)] = \mathbf{h} \circ \mathbf{f}(y)$, where $\mathbf{h}$ is the augmented function defined in the assumption $\text{H}$ in the Theorem. The $\tilde{v}(y)$ corresponds to $v(y)$ defined above. Note that $\tilde{v}$ is invertible because both $\mathbf{h}$ and $\mathbf{f}$ are invertible. What we need to prove is that $\tilde{v}$ is a block-wise invertible point-wise function, in the sense that $\tilde{v}_{1i}$ is a function of only one $y_{1j}$, and not of any of $y_{2j}$, and vice versa. This can be done by showing that the product of any two distinct partial derivatives of any component is always zero, and the Jacobian $J_\mathbf{v} \in \mathbb{R}^{2n \times 2n}$ is block diagonal; the upper and lower block correspond to $y_1$ and $y_2$ respectively. Along with invertibility, this means that each component depends exactly on one variable of the corresponding block ($y_1$ or $y_2$). Below, we show that separately for $J_\mathbf{v} \in \mathbb{R}^{n \times 2n}$ and $\mathbf{J}_\mathbf{Q} = \mathbb{R}^{n \times 2n}$.

Firstly, this is obviously true for $J_\mathbf{v}$ because $v_2(y)$ is just an identity mapping of $y_2$ from the definitions of $\mathbf{h}$ and $\mathbf{f}$, and does not depend on $y_1$; the lower non-zero block of $\mathbf{J}_\mathbf{v}$ is an identity matrix. Next, we will show that for $J_\mathbf{v}$. We differentiate Eq. 24 with respect to $y_c, 1 \leq c \leq n$ (an element of $y_1 = s_t$), and $y_d, c < d \leq 2n$, and get

$$W^{T} \frac{\partial^2}{\partial y_c \partial y_d} \psi(v(y)) = 0. \quad (26)$$

From the full row rank of $W$ and the calculation of differentials, we get

$$\frac{\partial^2}{\partial y_c \partial y_d} \psi(v(y)) = \Psi(v(y))^T v(y) = 0, \quad (27)$$

where $\Psi(y) = (e^{(1,1)}(y_1), \ldots, e^{(1,k)}(y_1), \ldots, e^{(n,1)}(y_n), \ldots, e^{(n,k)}(y_n)) \in \mathbb{R}^{2nk \times nk}$, $e^{(a,b)} = (0, \ldots, 0, \psi_{ab}(v_a), \psi_{ab}'(v_a), 0, \ldots, 0)^T \in \mathbb{R}^{2n}$, such that the non-zero entries are at indices $(2a-1, 2a), \psi(v(y)) = (v_0^c(y), v_0^d(y), \ldots, v_0^{cd}(y), v_0^{cd}(y)) \in \mathbb{R}^{2n}, v_0 = \frac{\partial v}{\partial y_c}(y), \text{ and } v_0^{cd} = \frac{\partial^2 v}{\partial y_c \partial y_d}(y)$. From Lemma 4 and 5 of [13], assumption $\text{H}$ implies that $\Psi(y)$ has full row rank $2n$, and thus the pseudo-inverse of $\Psi(y)^T$ fulfills $\Psi(y)^T \Psi(y)^T = I$. We multiply the equation above from the left by such pseudo-inverse and obtain

$$v(y) = 0. \quad (28)$$

In particular, $v_0^c(y)v_0^d(y) = 0$ for all $1 \leq a \leq n, 1 \leq c \leq n$, and $c < d \leq 2n$. This means that a row of $J_\mathbf{v} \in \mathbb{R}^{n \times 2n}$ at each $y$ has either 1) only one non-zero entry somewhere in the former half block (corresponding to the partial derivatives by $y_1$) or 2) non-zero entries only in the latter half block (corresponding to the partial derivatives by $y_2$). The latter case is contradictory because it means that the component $v_i$ is a function of only $y_2 = x_{i-1}$, and cannot hold Eq. 24 which right-hand side is a function of all components of $y_1$ (and $y_2$). Therefore, $J_\mathbf{v}$ should have only one non-zero entry in the former half block for each row. From the results of $J_\mathbf{v}$ and $J_\mathbf{Q}$, we deduce that $J_\mathbf{v}$ is a block diagonal matrix. Now, by invertibility and continuity of $J_\mathbf{v}$, we deduce that the location of the non-zero entries are fixed and do not change as a function of $y$. This proves that $v = \mathbf{h} \circ \mathbf{f}(y)$ is a block-wise invertible point-wise function, and $v_i (= h_i(x_i, x_{i-1}))$ is represented by only one $y_{1j}$, ($= s_j(t)$) up to a scalar (component-specific) invertible transformation, and the Theorem is proven.

### C Implementation detail for Simulation 1

We give here more detail on the data generation, training, and evaluation for IIA-GCL in Simulation 1 (Section 4.1).
**Data generation** We generated data from an artificial NVAR process with time-index-parameterized nonstationary innovations. The nonstationary innovations were randomly generated from a Gaussian distribution by modulating its mean and standard deviation across time \( t \); i.e., the auxiliary variable \( u_t = t \), and \( \log p(s_i(t)) \propto -\lambda_{i1}(t)s_i(t)^2 - \lambda_{i2}(t)s_i(t) \), where \( \lambda_{i1}(t) \) and \( \lambda_{i2}(t) \) control the standard deviation and mean of the \( i \)-th component at time point \( t \), respectively. Each of \( \lambda_{i1}(t) \) and \( \lambda_{i2}(t) \) was modeled to be temporally smooth and continuous, by 1) obtaining a combination of Fourier basis functions spanning the whole time series (sine and cosine bases with 64 frequencies), which weights were randomly selected from uniform distribution, 2) normalizing to \([-2, 2]\), and 3) (only for \( \lambda_{i1}(t) \)) putting into exponential function. The dimensions of the observations and innovations \((n)\) were 20. As the NVAR model, we used a multilayer perceptron we call NVAR-MLP, which takes a concatenation of \( x_{t-1} \) and \( s_i \) as an input, then outputs \( x_t \). To guarantee the invertibility, we fixed the number of units of each layer to \( n \), and used leaky ReLU units for the nonlinearity except for the last layer which has no nonlinearity.

**Training** Considering the innovation model with \( u_t = t \), we here used IIA-GCL for the estimation of the latent innovations. We adopted MLPs as the nonlinear scalar functions in Eq. 6. The MLP for \( h (h-MLP) \) outputs \( n \)-dimensional feature values from an input \((x_t, x_{t-1})\), which is supposed to represent the latent innovations after the training. The number of layers was selected to be the same as that of the NVAR-MLP, and the number of node in each layer was \( 4n \) except for the output layer \((n)\), so as to make it have enough number of parameters as the demixing model. A maxout unit was used as the activation function in the hidden layers, which was constructed by taking the maximum across two affine fully connected weight groups, while no-linearity was applied at the last layer. The scalar functions \( \psi_{ij}, \mu_{ij}, \alpha(u_i) \) were modeled to be consistent with the NVAR model; i.e., we incorporated the information into the model that 1) the innovations were generated based on the Gaussian distribution with mean and std modulations by the log-pdf shown above, and 2) \( \lambda_{i1} \) and \( \lambda_{i2} \) were generated through a combination of Fourier basis functions with 64 frequencies, while their weights have to be estimated from the data. For \( \phi \), which has dependency on \( u_t \), we used the same structure as the combination of \( h, \psi_{ij}, \) and \( \mu_{ij} \) explained above, which we call \( \phi \)-MLP, except that the \( \phi \)-MLP takes a single data point \((x_{t-1})\) as an input, instead of a set of the consecutive points \((x_t, x_{t-1})\). The regression function also needs additional terms representing the marginal distributions of \( s \) and \( x \) \((\beta \) and \( \gamma)\), which were here modeled by the weighted squared sum of the output units of the \( h \)-MLP and \( \phi \)-MLP, respectively.

The nonlinear regression function was trained by back-propagation with a momentum term so as to discriminate the real dataset from its \( u_t \)-randomized version. The initial parameters were randomly drawn from a uniform distribution. The performance was evaluated by the Pearson correlation between the true innovations and the estimated feature values \( h \). It was averaged over 10 runs, for each setting of the complexity (number of layers) \( L \in [1, 3, 5] \) of the NVAR-MLP and the number of data points.

For comparison, we also applied NICA based on GCL (NICA-GCL [10]), an NVAR with additive innovation model (AD-NVAR), and variational autoencoder (VAE) [16] to the same data. In NICA based on GCL, we estimated the independent components by the similar architecture as IIA-GCL shown above, except that it assumed the instantaneous mixture model for the observation. In AD-NVAR, we trained an MLP with \( 4n \) maxout hidden units, which predicts \( x_t \) from \( x_{t-1} \), based on the mean squared prediction errors, and the error term was used as the estimation of the additive innovation; i.e., \( x_t = \text{MLP}(x_{t-1}) + \tilde{s}_t \), where \( \tilde{s}_t \) is the estimation of the innovation. In VAE, we trained an encoder based on an MLP, which nonlinearly embeds an input \((x_t, x_{t-1})\) into a 20 dimensional feature space representing the estimation of the innovation. The number of nodes in each layer was designed to linearly decrease from input \((40)\) to the output \((20)\). We fixed \( L \in [1, 2] \) exceptionally for VAE because of the instability of training in high layer models. We additionally applied linear ICA based on nonstationarity of variance (NSICA) [7] to the innovations estimated by AD-NVAR and VAE for fair comparisons because they do not assume the independence on the estimations. For all of them, we fixed the number of layers of the demixing model to be the same as that of the NVAR-MLP.

D Implementation detail for Simulation 2

We give here more detail on the training for IIA-TCL in Simulation 2 (Section 4.2).
Training  We applied IIA-TCL to the same data used in Simulation 1. For IIA-TCL, we first divided the time series into 256 equally-sized segments, and used the segment label as the auxiliary variable $u_i$; i.e., we assume that the data are segment-wise stationary. Although this assumption is not consistent with the real innovation model (Simulation 1), it is approximately true because the modulations were temporally smooth and continuous; we thus consider here data with a realistic deviation from model assumptions. We adopted MLPs as the nonlinear scalar functions in the regression function (Eq. [8]). The architecture of the MLP for $h$ ($h$-MLP) was the same as that in Simulation 1. Considering the log-pdf of the innovation, we fixed $\psi_1(y_t) = y_t^2$, and $\psi_2(y_t) = y_t$. For $\phi$, which has dependency on $u_i$, we used the same structure as the combination of $h$, $\psi_1$, and $w_{1:T}$, except that $\phi$ takes a single data point $(x_{t-1})$ as an input, instead of a set of consecutive points $(x_t, x_{t-1})$. The training and evaluation methods follow those in Simulation 1. We discarded the cases of small data sets (210 and 212, corresponding to 4 and 16 samples in a segment) because of the instability of training.

For comparison, we also applied NICA (TCL [8]), which estimates the independent components by the similar architecture as IIA-TCL shown above, except that it assumed the instantaneous mixture model for the observation.

E  Detail for experiments on real brain imaging data

Data and preprocessing  We used a publicly available MEG dataset [31] (https://doi.org/10.17605/OSF.IO/M25N4). Briefly, the participants were presented with a random word selected from 420 unrelated German nouns (duration $= 697 \pm 119$ ms) either visually (projected centrally on a screen) or auditorily (via nonferromagnetic tubes to both ears) randomly for each trial. The stimulus was followed by a visual fixation cross until the end of the trial (2000 ms after the stimulus onset). MEG signals were measured from twenty healthy volunteers by a 148-channel magnetometer (MAGNES 2500 WH, 4D Neuroimaging, San Diego, USA) inside a magnetically shielded room. The data were downsampled to 300 Hz, and epoched into trials. The contaminated trials were rejected by visual inspections, and thereafter the blinks, eye movements, and cardiac artifacts were corrected using ICA (see [31] for more details of the preprocessing). We further band-pass filtered the data between 4 Hz and 125 Hz, normalized them to have zero-mean and unit variance at the base line period ($-1,000$ ms to $0$ ms) for each channel and trial, and then cropped from $-300$ ms to $2,000$ ms after the onset for each trial. The dimension of the data was reduced to 30 by PCA. There were $219.1 \pm 22.4$ trials ($110.4 \pm 11.5$ for auditory and $108.7 \pm 11.9$ for visual) for each subject, and in total, 2,207 auditory and 2,174 visual trials in the whole dataset.

IIA settings  We used IIA-TCL for the training, by assuming an NVAR(3) model and the segment-wise-stationary of the latent innovations. The trial data were divided into 84 equally sized segments of length of 8 samples (26.7 ms), and the segment label was used as the auxiliary variable $u_i$. The same segment labels were given across the trials; however, considering the possible stimulus-specific dynamics of the brain, we assigned different labels for the auditory and visual trials. In total, there are 168 segments (classes) to be discriminated by MLR. The network architectures of the MLPs are the same with those in Simulation 2, except that $h$ and $\phi$ take $x_{t-3}$ and $x_{t-1:t-3}$ as inputs, respectively, the number of units of each layer was fixed to 30, and that of the last layer (number of components) was 5. The smaller number of components than the data dimension can be justified by assuming the stationarity of the remaining components (the remaining innovations do not depend on $u$) [8]. Considering the fast sampling rate of the data (300 Hz), we fixed the time lag between two consecutive samples to 3 (10 ms). The other settings were as in Simulation 2. The training of a four-layer model by IIA-TCL took about 2 hours (Intel Xeon 3.5 GHz 16 core CPUs, 376 GB Memory, NVIDIA Tesla V100 GPU).

Evaluation methods  For evaluation, we performed classification of the stimulus modality (auditory or visual) by using the estimated innovations. The classification was performed using a linear support vector machine (SVM) classifier trained on the stimulation label and sliding-window-averaged innovations (width=16 and stride=8 samples) obtained for each trial. The performance was evaluated by the generalizability of a classifier across subjects, i.e., one-subject-out cross-validation (OSO-CV); the feature extractor and the classifier were trained only from the training subjects, and then applied to the held-out subject. The hyperparameters of the SVM were determined by nested OSO-CV without using the test data. For comparison, we also applied NICA based on TCL [8] and
Figure 3: Mean classification accuracies of the (multinomial) logistic regression after the training by IIA in the simulations. (a) (Simulation 1; IIA-GCL) Mean classification accuracies of the real dataset and its randomized version by the nonlinear logistic regression trained by IIA-GCL, as a function of sample size and complexity (number of layers $L$) of the NVAR model. Solid lines: test data, dotted line: training data. The chance level is 50%. The logistic regression achieved higher classification accuracies than the chance level, which implies that the regression function learned the hidden temporal structure of the innovations from the observable time series. The higher complexity of the NVAR model (larger $L$) complicates the training, and the regression function suffered from overfitting when the number of data points was not sufficient. (b) (Simulation 2; IIA-TCL) Mean classification accuracies of the nonlinear MLR trained by IIA-TCL, as a function of sample size and complexity (number of layers $L$) of the NVAR model. Solid lines: test data, dotted line: training data. The chance level is 0.39% (256-class classification). The MLR achieved higher classification accuracies than the chance level, which implies that the regression function learned the nonstationarity of the innovations from the observable time series. As with IIA-GCL in a, the higher complexity of the NVAR models complicates the training, and the regression function suffered from overfitting when the number of data points was not sufficient.

AD-NVAR(3). In NICA based on TCL, we estimated the independent components by the similar architecture as IIA-TCL, except that it assumed the instantaneous mixture model for the observation. In AD-NVAR(3), we trained an MLP, which predicts $x_t$ from $x_{t-1:t-3}$, based on the mean squared prediction errors, and the error term was used as the estimation of the additive innovation; i.e., $x_t = \text{MLP}(x_{t-1:t-3}) + \tilde{s}_t$, where $\tilde{s}_t$ is the estimation of the innovation. The number of units of each layer was fixed to 30. We additionally applied linear ICA (NSVICA) to the estimations by AD-NVAR(3) so as to reduce the dimension to 5 for fair comparisons. We omitted $L = 1$ for IIA-TCL because of the instability of training.

We also visualized the spatial characteristics of each innovation component by estimating the optimal (maximal and minimal) input $x_t$ while fixing $x_{t-1:t-3}$ to zero. This method is commonly used in deep learning studies to visualize the input specificities of a hidden node of a neural network. We used $l_2$ regularization on the input to avoid overfitting.