Journal of Statistical Software

March 2021, Volume 97, Issue 5. *doi: 10.18637/jss.v097.i05*

svars: An R Package for Data-Driven Identification in Multivariate Time Series Analysis

Alexander Lange
University of Goettingen

Bernhard Dalheimer
University of Goettingen

Helmut Herwartz
University of Goettingen

Simone Maxand
University of Helsinki

Abstract

Structural vector autoregressive (SVAR) models are frequently applied to trace the contemporaneous linkages among (macroeconomic) variables back to an interplay of orthogonal structural shocks. Under Gaussianity the structural parameters are unidentified without additional (often external and not data-based) information. In contrast, the often reasonable assumption of heteroskedastic and/or non-Gaussian model disturbances offers the possibility to identify unique structural shocks. We describe the R package **svars** which implements statistical identification techniques that can be both heteroskedasticity-based or independence-based. Moreover, it includes a rich variety of analysis tools that are well known in the SVAR literature. Next to a comprehensive review of the theoretical background, we provide a detailed description of the associated R functions. Furthermore, a macroeconomic application serves as a step-by-step guide on how to apply these functions to the identification and interpretation of structural VAR models.

Keywords: SVAR models, identification, independent components, non-Gaussian maximum likelihood, changes in volatility, smooth transition covariance, R.

1. Introduction

Particularly in macroeconometrics, structural vector autoregressive (SVAR) models have become a prominent tool to determine the impacts of different (economic) shocks in a system of variables. Within these models, the unobserved structural shocks represent information that is hidden in the reduced form vector autoregressive (VAR) model. Nevertheless, analysts might be interested in the system's reaction to exactly this type of isolated shocks, which is commonly visualized by means of impulse-response functions. For instance, policy makers could be interested in revealing the effects of an unexpected interest rate cut. Estimating the reduced form VAR by means of least squares (LS) or maximum likelihood methods (ML) is straightforward (see, e.g., Lütkepohl 2005), however, identifying the non-unique structural form is a controversial topic in the SVAR literature.

Beginning with the pioneering work of Sims (1980), two main types of identification strategies have been developed. On the one hand, following Sims (1980) original ideas such strategies refer to economic theory. Theory-based methods implement economic restrictions a-priori, e.g., short-run restrictions (Sims 1980), long-run restrictions (Blanchard and Quah 1989) or specific sign patterns (Uhlig 2005). On the other hand, statistical identification methods which have been developed more recently exploit the informational content of specific data features (heteroskedasticitiy of structural shocks, uniqueness of non-Gaussian independent components). The R package **svars**, which we describe in this paper, focuses on these statistical methods to identify the structural shocks.

The R (R Core Team 2021) archive network comprises several widely applied packages for multivariate time series models and, in particular, for analyzing VAR models. The **vars** package (Pfaff 2008) contains estimation techniques for reduced form VAR models, and functions to determine the lag order and to perform several diagnostic tests. Moreover, the **vars** package allows for the estimation of a basic structural form by means of theory-based short- and long-run restrictions. Further R packages for multivariate time series analysis and VAR estimation are **tsDyn** (Di Narzo, Aznarte, Stigler, and Tsung-wu 2020) and **MTS** (Tsay and Wood 2018). To the authors' knowledge, currently only the **VARsignR** package (Danne 2015) contains functions for SVAR identification by means of theory-based sign restrictions.

Given the lack of implementations of statistical identification techniques in R, the package **svars** has been explicitly developed to fill this gap by providing various recent statistical methods to estimate SVAR models. These methods build upon distinct but not mutually exclusive statistical properties of the data (i.e., covariance changes and the uniqueness of independent non-Gaussian distributed structural shocks). The **svars** package supports six identification techniques. Three identification methods make use of the assumption of heteroskedastic shocks, i.e., the identification (i) via changes in volatility (Rigobon 2003), (ii) via smooth transitions of covariances (Lütkepohl and Netsunajev 2017b) and (iii) via generalized autoregressive conditional heteroskedasticity (GARCH) (Normadin and Phaneuf 2004; Bouakez and Normandin 2010). Three further identification methods connect to the uniqueness of non-Gaussian independent components, namely the detection of least dependent innovations based on (iv) Cramér-von Mises (CVM) statistics (Herwartz 2018), (v) the distance covariances (Matteson and Tsay 2017) and (vi) a parametric non-Gaussian ML approach (Lanne, Meitz, and Saikkonen 2017).

By offering a variety of identification methods, the **svars** package can be applied in diverse data settings. Additionally, it extends the existing pool of SVAR techniques in R with more recent bootstrap procedures, further statistics and hypothesis tests directly related to inference in SVAR models. In this sense, the **svars** package is designed as a complete toolbox for the structural analysis of multivariate time series. Based on objects from reduced form estimations, **svars** is compatible with other packages such as **vars**, **tsDyn** and **MTS**. Moreover, computationally demanding modules are fully implemented in C++ and linked to R using the **Rcpp** (Eddelbuettel and François 2011) and **RcppArmadillo** (Eddelbuettel and Sanderson 2014) libraries. The package is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=svars`.

The article is organized as follows: Section 2 outlines the SVAR model and the alternative identification methods. In Section 3, we describe bootstrap methods and further diagnostic tools for SVAR analysis. Section 4 details the package design, and Section 5 provides an illustrative application of two identification schemes to a real world dataset. Lastly, a summary and an outlook on future extensions of the **svars** package complete this article.

2. Structural vector autoregressive models

Consider a K-dimensional VAR model of order p

$$\begin{aligned} y_t &= \mu + A_1 y_{t-1} + \cdots + A_p y_{t-p} + u_t, && (1)\\ &= \mu + A_1 y_{t-1} + \cdots + A_p y_{t-p} + B\varepsilon_t, \qquad t = 1, \ldots, T, && (2) \end{aligned}$$

where $y_t = [y_{1t}, \ldots, y_{Kt}]^\top$ is a vector of observable variables, $A_i, i = 1, \ldots, p$, are $(K \times K)$ coefficient matrices, and intercept parameters are collected in μ. We focus on the case of time invariant deterministic terms for notational clarity. Model augmentation with time-varying deterministic terms (e.g., breaks, linear trends), however, is straightforward. Furthermore, the VAR model is stationary (invertible) by assumption. The vector u_t consists of reduced-form residuals, which are serially uncorrelated with $\mathbb{E}(u_t) = 0$ and $\text{Cov}(u_t) = \Sigma_u$. The nonsingular matrix B captures the instantaneous effects of the structural shocks $\varepsilon_t = B^{-1}u_t$ on the variables of the system.

In the following, we briefly discuss the identification problem in SVAR analysis. Subsequently, we present six alternative statistical approaches to uniquely determine the structural shocks. Finally, we provide a short guidance on how to choose between these alternative identification approaches.

2.1. The identification problem

Cross-equation relations between the reduced-form residuals in Equation 1 are characterized by the covariance matrix

$$\text{Cov}(u_t) = \Sigma_u = B\Sigma_\varepsilon B^\top, \qquad (3)$$

where the covariance of the structural shocks $\text{Cov}(\varepsilon_t) = \Sigma_\varepsilon$ is a diagonal matrix. Thus, structural shocks are uncorrelated, which enables a meaningful impulse-response analysis (Lütkepohl 2005). Without any further model specification, Equation 3 holds for every matrix B which decomposes the covariance matrix Σ_u. Hence, additional restrictions are necessary to identify a (unique) matrix B.[1] In this paper, we focus on identification techniques which use the underlying data structure to determine the structural matrix. After estimating the model in Equation 1 by means of LS or ML methods, the resulting reduced form residual estimates $\hat{u}_t$ and the corresponding covariance estimate $\widehat{\Sigma}_u$ provide the starting point for the subsequent identification techniques. The following two Sections introduce the statistical identification methods which constitute the core functions of the **svars** package.

[1] The identification problem is described in more detail, for instance, in Chapter 1 of Lütkepohl (2005). Kilian and Lütkepohl (2017) resume a variety of traditional and more recent methods to identify the structural shocks.

2.2. Identification by means of heteroskedastic innovations

Time series are often characterized by time-varying covariance structures. Therefore, it is tempting to unravel the structural relationships by means of such changes in the second order moments (see, e.g., Sentana and Fiorentini 2001; Rigobon 2003). The **svars** package includes three alternative heteroskedasticity-based SVAR identification schemes. The first approach is built upon unconditional shifts in the covariance (Rigobon 2003), while the second procedure allows for a smooth transition between the covariance regimes (Lütkepohl and Netsunajev 2017b). The third scheme implements the identification of the structural shocks via conditional heteroskedasticity (Normadin and Phaneuf 2004).

Changes in volatility (CV)

Rigobon (2003) uses the presence of shifts in the time series' variance at known time points for the identification of structural shocks. He considers a model of exogenous covariance changes. More precisely, the changes of the covariance matrix occur at prespecified break dates implying

$$\mathbb{E}(u_t u_t^\top) = \Sigma_t = \Sigma_u(m) \;\text{ for }\; m = 1, \dots, M,\, t = 1, \dots, T.$$

Here, the index $m = 1, \dots, M$ indicates the respective variance regime. In the most simple framework of two volatility states (i.e., $M = 2$) with a structural break at time point $T_{sb} \in \{1, \dots, T\}$, the reduced form covariance matrix is

$$\mathbb{E}(u_t u_t^\top) = \begin{cases} \Sigma_1 \text{ for } t = 1, \dots, T_{sb} - 1 \\ \Sigma_2 \text{ for } t = T_{sb}, \dots, T, \end{cases}$$

where $\Sigma_1 \neq \Sigma_2$. The two covariance matrices can be decomposed as $\Sigma_1 = BB^\top$ and $\Sigma_2 = B\Lambda B^\top$, where Λ is a diagonal matrix with diagonal elements $\lambda_{ii} > 0, i = 1, \dots, K$. The matrix Λ formalizes the change of the variance of structural shocks ε_t in the second regime. In other words, the structural shocks have unit variance in the first regime, and variances $\lambda_{ii}, i = 1, \dots, K$, in the second regime. The structural shocks are uniquely identified if all diagonal elements in Λ are distinct. Under the assumption of Gaussian residuals u_t, the log-likelihood function for the estimation of B and Λ is

$$\begin{aligned} \log \mathcal{L} = T\frac{K}{2}\log 2\pi - \frac{T_{sb}-1}{2}\left[\log\det(BB^\top) + \operatorname{tr}\left(\widehat{\Sigma}_1 (BB^\top)^{-1}\right)\right] \\ - \frac{T - T_{sb} + 1}{2}\left[\log\det(B\Lambda B^\top) + \operatorname{tr}\left(\widehat{\Sigma}_2 (B\Lambda B^\top)^{-1}\right)\right], \end{aligned} \tag{4}$$

where $\widehat{\Sigma}_1$ and $\widehat{\Sigma}_2$ are retrieved from estimated residuals $\widehat{u}_t$, respectively, as

$$\widehat{\Sigma}_1 = \frac{1}{T_{sb}-1}\sum_{t=1}^{T_{sb}-1} \widehat{u}_t \widehat{u}_t^\top \qquad \text{and} \qquad \widehat{\Sigma}_2 = \frac{1}{T - T_{sb} + 1}\sum_{t=T_{sb}}^{T} \widehat{u}_t \widehat{u}_t^\top.$$

For the numerical log-likelihood optimization of (4), the initial matrix B is the lower triangular decomposition of $T^{-1}\sum_{t=1}^{T} \widehat{u}_t \widehat{u}_t^\top$, and the initial matrix Λ is set to the identity matrix. Lanne and Lütkepohl (2008) introduce an iterative procedure to improve the estimation precision of this routine. The matrices $\widetilde{B}$ and $\widetilde{\Lambda}$, which are obtained from maximizing the log-likelihood function, are used for iterative generalized least squares (GLS) estimation of the deterministic and autoregressive parameters

$$\begin{aligned}\widehat{\beta} &= \mathrm{vec}[\widehat{\mu}, \widehat{A}_1, \dots, \widehat{A}_p] \\ &= \left[\sum_{t=1}^{T_{sb}-1} \left(Z_t Z_t^\top \otimes (\widetilde{B}\widetilde{B}^\top)^{-1}\right) + \sum_{t=T_{sb}}^{T} \left(Z_t Z_t^\top \otimes (\widetilde{B}\widetilde{\Lambda}\widetilde{B}^\top)^{-1}\right)\right]^{-1} \\ &\quad \times \left[\sum_{t=1}^{T_{sb}-1} \left(Z_t \otimes (\widetilde{B}\widetilde{B}^\top)^{-1}\right) y_t + \sum_{t=T_{sb}}^{T} \left(Z_t \otimes (\widetilde{B}\widetilde{\Lambda}\widetilde{B}^\top)^{-1}\right) y_t\right],\end{aligned}$$

where $Z_t^\top = [1, y_{t-1}^\top, \dots, y_{t-p}^\top]$. Then, the GLS estimator $\widehat{\beta}$ is used to update the covariance estimates by means of $\widehat{u}_t = y_t - (Z_t^\top \otimes I_K)\widehat{\beta}$. This algorithm iterates until the log-likelihood converges. Furthermore, standard errors for the structural parameters can be obtained from the square root of the inverted information matrix (Hamilton 1994).

Identification through changes in volatility is conditional on the determination of the variance regimes. If available, the analyst might use external information for the selection of suitable break points (T_{sb}). Typically these are extraordinary events in history which can be associated with a change in data variation (see, e.g., Rigobon and Sack 2004). Alternatively, the model might be evaluated conditional on a range of alternative break point candidates from which the analyst selects the model with the highest log-likelihood as described in Lütkepohl and Schlaak (2018).

Smooth transition (co)variances (ST)

The implementation of identification via smooth transition covariances follows the descriptions in Lütkepohl and Netsunajev (2017b) and generalizes the identification via changes in volatility. The covariance matrix of the error terms u_t consists of several volatility states, and the transition from one state to another is formalized by means of a non-linear function. For two volatility regimes with distinct covariance matrices Σ_1 and Σ_2, the covariance structure at time t is

$$\mathbb{E}(u_t u_t^\top) = \Omega_t = (1 - G(s_t))\,\Sigma_1 + G(s_t)\Sigma_2, \quad t = 1, \dots, T. \tag{5}$$

In (5), $G(\cdot)$ is the transition function determined by the transition variable s_t. While the transition variable is usually deterministic (e.g., $s_t = t$), the model also allows for stochastic transition variables, for instance, lagged dependent variables (see Lütkepohl and Netsunajev 2017b, for more details). The most frequently employed transition function is the logistic function proposed by Maddala (1977), which is of the form

$$G(\gamma, c, s_t) = [1 + \exp(-\exp(\gamma)(s_t - c))]^{-1}. \tag{6}$$

The coefficient γ determines the slope of the function and c is the time point of transition. Based on the covariance structure in Equation 5 and Equation 6, and the assumption of normally distributed residuals u_t, the log-likelihood function reads as

$$\log \mathcal{L} = T\frac{K}{2}\log 2\pi - \frac{1}{2}\sum_{t=1}^{T} \log\det(\Omega_t) - \frac{1}{2}\sum_{t=1}^{T} u_t^\top \Omega_t^{-1} u_t. \tag{7}$$

Grid optimization enables the determination of the transition parameters γ and c. Lütkepohl and Netsunajev (2017b) suggest an iterative procedure for every pair of parameters (γ, c).

The first step is the maximization of the log-likelihood in (7) with respect to the structural parameters B and Λ. In the second step, the estimated matrices $\widetilde{B}$ and $\widetilde{\Lambda}$ are used to re-estimate the reduced form VAR parameters by means of GLS estimation

$$\widehat{\beta} = \left((Z_t^\top \otimes I_K)W_T(Z_t \otimes I_K)\right)^{-1} (Z_t^\top \otimes I_K)W_T y,$$

where W_T is a blockdiagonal $(KT \times KT)$ weighting matrix

$$W_T = \begin{bmatrix} \Omega_1^{-1} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \Omega_T^{-1} \end{bmatrix}.$$

The GLS step obtains $\widehat{\beta}$ to update the covariance estimates by means of $\widehat{u}_t = y_t - (Z_t^\top \otimes I_K)\widehat{\beta}$. The two steps are performed until the log-likelihood converges. The iterative procedure is evaluated at every parameter pair $(\gamma,\ c)$ within a prespecified range. The parameter pair which maximizes the log-likelihood in Equation 7 is considered to provide the best estimate for the true transition. For a more detailed discussion of the parameter choice see Lütkepohl and Netsunajev (2017b).

Conditional heteroskedasticity (GARCH)

As proposed by Normadin and Phaneuf (2004), Lanne and Saikkonen (2007) and Bouakez and Normandin (2010) structural shocks are unique if their conditional variances are of the GARCH type. For the formal exposition let $\mathcal{F}_t$ denote a filtration that summarizes systemic information which is available until time t. Accordingly, the time-varying covariance can be represented as

$$\mathbb{E}(u_t u_t^\top \mid \mathcal{F}_{t-1}) = \Sigma_{t|t-1} = B\Lambda_{t|t-1}B^\top, \tag{8}$$

where

$$\Lambda_{t|t-1} = \mathrm{diag}(\sigma^2_{1,t|t-1}, \ldots, \sigma^2_{K,t|t-1}) \tag{9}$$

is a $(K \times K)$ matrix with GARCH implied variances on the main diagonal. In the context of SVAR identification typically low order GARCH(1,1) specifications are assumed, such that the individual variances exhibit a dynamic structure as

$$\sigma^2_{k,t|t-1} = (1 - \gamma_k - g_k) + \gamma_k \varepsilon^2_{k,t-1} + g_k \sigma^2_{k,t-1|t-2}, \qquad k = 1, .., K. \tag{10}$$

Higher-order GARCH structures are rarely employed in practice, even though this can be done in principle. Under suitable distributional and parametric restrictions, $\gamma_k > 0$, $g_k \geq 0$ and $\gamma_k + g_k < 1$, the marginal GARCH processes $\varepsilon_{k,t}$ are covariance stationary (Milunovich and Yang 2013). Sentana and Fiorentini (2001) have shown that the structural parameters in B are uniquely identified, if there are at least $K-1$ GARCH-type variances present in $\Lambda_{t|t-1}$. The parameters γ_k and g_k can be estimated by means of standard univariate ML approaches. The multivariate Gaussian log-likelihood to obtain the structural parameters in B is

$$\log \mathcal{L} = T\frac{K}{2}\log 2\pi - \frac{1}{2}\sum_{t=1}^{T} \log\det(\Sigma_{t|t-1}) - \frac{1}{2}\sum_{t=1}^{T} u_t^\top \Sigma_{t|t-1} u_t. \tag{11}$$

For the practical implementation of identification through patterns of conditional heteroskedasticity, we follow the approach suggested by Lütkepohl and Milunovich (2016), and estimate the parameters in (10) and (11) iteratively until the log-likelihood in (11) converges.

2.3. Identification through independent components

As implied by a result of Comon (1994), independence of the components of ε_t could serve to identify the matrix B if at most one component ε_{it} exhibits a Gaussian distribution. Furthermore, partial identification of the non-Gaussian components is possible if the system contains multiple Gaussian components (cf. Maxand 2020). The **svars** package implements three distinct approaches for identification by means of independent components. Referring to principles of Hodges-Lehman (HL) estimation (Hodges and Lehmann 2006), the first two identification strategies allow for the detection of least dependent structural shocks by the minimization of nonparametric dependence criteria. More specifically, the first technique reveals the structural shocks by minimizing the CVM distance of Genest, Quessy, and Rémillard (2007). Following a suggestion of Matteson and Tsay (2017), the distance covariance statistic of Székely, Rizzo, and Bakirov (2007) is employed as a nonparametric independence diagnostic for the second estimator. The third identification scheme is a fully parametric ML approach for detecting independent Student-t distributed shocks (Lanne *et al.* 2017).

Least dependent innovations build on Cramér-von Mises statistics (CVM)

Under Gaussianity, the decomposition factor B of the covariance matrix Σ_u is not unique as Gaussian random vectors do not change their joint distribution under rotation. In contrast, assuming not more than one Gaussian distributed component ε_{it} in ε_t, the structural matrix B can be uniquely determined. Introducing the nonparametric identification scheme, let D denote a lower triangular Choleski factor of the covariance matrix of the reduced-form errors, $\Sigma_u = DD^\top$, which links the structural and reduced form errors by $\varepsilon_t = D^{-1}u_t$. Further candidate structural shocks can be generated as

$$\tilde{\varepsilon}_t = Q\varepsilon_t = QD^{-1}u_t, \tag{12}$$

where Q is a rotation matrix such that $Q \neq I_K$, $QQ^\top = I_K$. The rotation matrix could be parameterized as the product of $K(K-1)/2$ distinct forms of orthogonal Givens rotation matrices. In the case of $K = 3$, for instance, $Q(\theta)$ is defined as

$$Q(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_1) & -\sin(\theta_1) \\ 0 & \sin(\theta_1) & \cos(\theta_1) \end{bmatrix} \times \begin{bmatrix} \cos(\theta_2) & 0 & -\sin(\theta_2) \\ 0 & 1 & 0 \\ \sin(\theta_2) & 0 & \cos(\theta_2) \end{bmatrix} \times \begin{bmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 \\ \sin(\theta_3) & \cos(\theta_3) & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

with rotation angles $0 \leq \theta_i \leq \pi$, $i = 1, 2, 3$. By definition, the random vector $\tilde{\varepsilon}_t$ in Equation 12 is a rotation of ε_t. The set of possible structural matrices $B(\theta) = D\,Q(\theta)$ is defined in terms of the Choleski factor D and the vector of rotation angles θ of the Givens matrices $Q(\theta)$.

To avoid any restrictive assumption on the distribution of ε_t, nonparametric independence tests are applied to measure the degree of dependence. For instance, the copula-based CVM distance of Genest *et al.* (2007) has been successfully applied in the SVAR literature (Herwartz and Plödt 2016a; Herwartz 2018) to assess mutual dependence. The CVM distance is

$$\mathcal{B}_\theta = \int_{(0,1)^K} \left[\sqrt{T} \left(C(\tilde{\varepsilon}) - \prod_{i=1}^{K} U(\tilde{\varepsilon}_i) \right) \right]^2 d\tilde{\varepsilon}, \tag{13}$$

where C is the empirical copula and U is the distribution function of a uniformly distributed variable on $\{1/T, \ldots, T/T\}$. The CVM algorithm provides a matrix estimate $\widehat{B}$ such that the rotated structural shocks $\tilde{\varepsilon}_t$ minimize the CVM dependence criterion. Hence, the obtained structural shocks are least dependent according to the statistic in (13) and the corresponding structural matrix $\widehat{B}$ is the HL estimator. Standard errors for $\widehat{B}$ are obtained by means of bootstrap procedures as presented in Section 3.6.

Least dependent innovations build on distance covariance (DC)

There is a variety of nonparametric criteria available to measure the degree of dependence between random variables, one of which, namely the CVM distance, has been described before. The ICA algorithm by Matteson and Tsay (2017) provides a matrix estimate $\widehat{B}$ such that the respective structural shocks $\tilde{\varepsilon}_t = \widehat{B}^{-1}\hat{u}_t$ minimize the distance covariance of Székely *et al.* (2007), which we denote as $\mathcal{U}_T(\tilde{\varepsilon}_t)$, i.e., the elements in $\tilde{\varepsilon}_t$ are least dependent according to $\mathcal{U}_T(.)$. Similar to the procedure building on the CVM statistic, the set of possible structural matrices $B(\theta)$ is defined in terms of the Choleski factor D and the vector of rotation angles θ of $Q(\theta)$. The rotation angles $\tilde{\theta} = \text{argmin}_\theta \mathcal{U}_T(\tilde{\varepsilon}_t(\theta))$ determine the estimated structural matrix $\widehat{B} = B(\tilde{\theta})$.[2] In the **svars** package, we take advantage of the function `steadyICA` from the R package **steadyICA** (Risk, James, and Matteson 2015) to estimate $\widehat{B}$. The minimum is determined by means of a gradient algorithm.

Non-Gaussian maximum likelihood (NGML)

The identification technique described by Lanne *et al.* (2017) is also based on the assumption of non-Gaussian structural error terms. They propose ML estimation to determine the set of independent structural innovations, which are assumed to exhibit a Student t-distribution. Moreover, Lanne *et al.* (2017) suggest a three-step estimation method for computationally demanding situations. The first step consists of LS estimation of the VAR parameters $\beta = \text{vec}[\mu, A_1, \ldots, A_p]$ and of the reduced form residuals $u_t(\widehat{\beta}) = y_t - \widehat{\mu} - \widehat{A}_1 y_{t-1}, \ldots, -\widehat{A}_p y_{t-p}$. In the second step the log-likelihood function is maximized conditional on the first step estimates $\widehat{\beta}$. The log-likelihood function is

$$\log \mathcal{L}(\delta) = \log \mathcal{L}(\widehat{\beta}, \delta) = T^{-1} \sum_{t=1}^{T} l_t(\widehat{\beta}, \delta), \tag{14}$$

where

$$l_t(\widehat{\beta}, \delta) = \sum_{i=1}^{K} \log f_i(\sigma_i^{-1} \iota_i B(b)^{-1} u_t(\widehat{\beta}); df_i) - \log \det(B(b)) - \sum_{i=1}^{K} \log \sigma_i,$$

and ι_i is the i-th unit vector. The parameter vector of the log-likelihood function is composed of $\widehat{\beta}$ and $\delta = (b, \sigma, df)$. Regarding the latter, b is a $K(K-1) \times 1$ vector which contains the off-diagonal elements of the covariance decomposition matrix B. The parameters σ_i and df_i are the scale and the degrees of freedom parameters of the density function f_i of a Student t-distribution, respectively. In the third step, the parameter vector δ is replaced by the estimate $\widetilde{\delta}$ and the log-likelihood

$$\log \mathcal{L}(\beta) = \log \mathcal{L}(\beta, \widetilde{\delta}) = T^{-1} \sum_{t=1}^{T} l_t(\beta, \widetilde{\delta})$$

is maximized.

[2] For details on the exact minimization procedure and the empirical definition of the dependence measure we refer to Matteson and Tsay (2017).

	Assumptions on the variance of ε_t			Assumptions on the distribution of ε_t		
	Homoskedastic	Heteroskedastic		Gaussian	Non-Gaussian	
Model		Unconditional	Conditional		Arbitrary	t-distribution
Heteroskedasticity						
CV		✓		✓		
ST		✓	✓	✓		
GARCH			✓	✓		
Independence						
CVM	✓				✓	
DC	✓				✓	
NGML	✓					✓

Table 1: Overview of identification models and respective underlying assumptions on the error term ε_t. Note that depending on the choice of the transition variable, the ST model can capture unconditional as well as conditional heteroskedasticity (Lütkepohl and Netsunajev 2017b).

2.4. Choice of an adequate identification technique

In the face of a variety of statistical approaches available to model latent structural relationships, method selection becomes an important step of statistical identification. To facilitate this selection step Table 1 provides an overview of the assumptions on the error terms ε_t within the alternative identification models. Estimating the structural parameters by means of heteroskedasticity-based approaches necessitates the corresponding type of covariance structure. Contrarily, identification through independent components is only possible in non-Gaussian distributional frameworks. Note that we distinguish between nonparametric models (i.e., CVM and DC) where no further specification of the distribution of the innovations is required and fully parametric ML approaches. A more detailed discussion on method selection in the context of identification via heteroskedasticity can be found in Lütkepohl and Netsunajev (2017a) and Lütkepohl and Schlaak (2018). Moreover, Herwartz, Lange, and Maxand (2019) compare heteroskedasticity and independence-based models in a large scale simulation study. They show that identification by means of covariance changes provides precise estimation results if the log-likelihood is correctly specified, whereas under (co)variance misspecification such identification schemes lack efficiency or might suffer from estimation bias. In contrast, simulation-based evidence suggests that identification via independent components is more robust with respect to alternative distributional frameworks and heteroskedasticity as long as the innovations are non-Gaussian.

3. SVAR tests, tools and bootstrap methods

As a basis for the six identification techniques, the statistical analysis of SVAR models requires a diagnostic analysis of the underlying data structure. The presented package comprises two types of data-driven procedures where the first group assumes heteroskedasticity and the second one non-Gaussianity of the error terms. To decide on Gaussianity of the data a number of normality tests are available in respective R packages – see, e.g., **normtest** (Gavrilov and Pusev 2014) and **ICtest** (Nordhausen, Oja, Tyler, and Virta 2019). Furthermore, the **svars** package contains several useful tests for SVAR analysis which have not yet been implemented in R. Next we describe the diagnostics and discuss several tools which support the economic interpretations of SVAR estimation results.

3.1. Tests for structural breaks

As described in Section 2, identification-based on changes in volatility presumes at least one break point to occur in the covariance structure. To detect different types of breaks in the data, several tests that have been implemented in the **strucchange** package (Zeileis, Leisch, Hornik, and Kleiber 2002) are accessible for VAR analysis via the method `stability()` of the **vars** package. In the following, we consider two additional types of multivariate Chow tests, the sample split and the break point test. The sample split test addresses the null hypothesis of constant VAR parameters μ and A_i, $i = 1, 2, \dots, p$. The break point test works similarly, but also tests if the covariance matrix of the residuals u_t is constant over time (Lütkepohl and Kraetzig 2004, Chapter 3). To implement suitable likelihood ratio statistics, the VAR is estimated conditional on the full sample of T observations and conditional on the first $T_1 = T_{sb} - p - 1$ and the last $T_2 = T - p - T_{sb}$ observations with T_{sb} indicating the break point. The resulting residuals are denoted by $\widehat{u}_t$, $\widehat{u}_t^{(1)}$ and $\widehat{u}_t^{(2)}$. Then, the sample split and break point test statistic are defined, respectively, as

$$\lambda_{SP} = (T_1 + T_2)\left\{\log\det(\widehat{\Sigma}_{1,2}) - \log\det\left[\left(\frac{1}{T_1+T_2}(T_1\widehat{\Sigma}_1 + T_2\widehat{\Sigma}_2)\right)\right]\right\} \tag{15}$$

and

$$\lambda_{BP} = (T_1 + T_2)\log\det(\widehat{\Sigma}_{(1,2)}) - T_1\log\det(\widehat{\Sigma}_1) - T_2\log\det(\widehat{\Sigma}_2), \tag{16}$$

where the covariance estimators are

$$\widehat{\Sigma}_{(1,2)} = \frac{1}{T_1}\sum_{t=1}^{T_1}\widehat{u}_t\widehat{u}_t^\top + \frac{1}{T_2}\sum_{t=T-T_2+1}^{T_2}\widehat{u}_t\widehat{u}_t^\top,$$

$$\widehat{\Sigma}_{1,2} = \frac{1}{T_1+T_2}\left(\sum_{t=1}^{T_1}\widehat{u}_t\widehat{u}_t^\top + \sum_{t=T-T_2+1}^{T_2}\widehat{u}_t\widehat{u}_t^\top\right),$$

$$\widehat{\Sigma}_1 = \frac{1}{T_1}\sum_{t=1}^{T_1}\widehat{u}_t^{(1)}\widehat{u}_t^{(1)^\top}, \text{ and } \widehat{\Sigma}_2 = \frac{1}{T_2}\sum_{t=T_1+1}^{T}\widehat{u}_t^{(2)}\widehat{u}_t^{(2)^\top}.$$

Candelon and Lütkepohl (2001) show that both test statistics λ_{BP} and λ_{SP} converge to a non-pivotal asymptotic limit distribution. Hence, bootstrap procedures are a natural device to obtain critical values for the statistic at hand.

3.2. Testing for identical diagonal elements

Since the structural shocks are estimated by the volatility models under the assumption that the variance of the structural shocks change differently, respective diagnostic tests are frequently employed in the SVAR literature (see, e.g., Herwartz and Plödt 2016b; Lütkepohl and Velinov 2016; Lütkepohl and Netsunajev 2017a). A suitable Wald statistic to test the null hypothesis of proportional variance shifts, $H_0 : \lambda_{ii} = \lambda_{jj}$ is defined as

$$\lambda_{W,ij} = \frac{(\lambda_{ii} - \lambda_{jj})^2}{\mathrm{Var}(\lambda_{ii}) + \mathrm{Var}(\lambda_{jj}) - 2\mathrm{Cov}(\lambda_{ii}, \lambda_{jj})} \sim \chi^2_{(2)}, \tag{17}$$

where parameter estimates and (co)variances obtain from the ML estimation. The null hypothesis is rejected for large values of $\lambda_{W,ij}$.

3.3. Test for overidentifying restrictions

The non-Gaussian ML and heteroskedasticity-based models rest on a stylized log-likelihood optimization, which also allows for restricting the structural parameter space. Subsequently, the implied restrictions can be tested by means of likelihood ratio statistics

$$\lambda_{LR} = 2\left[\log \mathcal{L}\left(\mathrm{vec}(\widetilde{B})\right) - \log \mathcal{L}\left(\mathrm{vec}(\widetilde{B}_r)\right)\right] \sim \chi^2_{(N)}, \tag{18}$$

where $\widetilde{B}$ is the unrestricted ML estimator as defined in Equation 4, Equation 7 or Equation 14. Moreover, $\widetilde{B}_r$ denotes the restricted ML estimator, and N is the number of restrictions. The null hypothesis that the restricted model holds is rejected for large values of λ_{LR} (Lütkepohl 2005).

3.4. Test on joint parameter significance

To test joint hypotheses of parameter significance for non-likelihood-based models as in Herwartz (2018) the package provides a χ^2-test. The statistic for testing a number of J linearly independent hypotheses is defined as

$$\lambda_{JS} = \left(R\mathrm{vec}(\widehat{B}) - r\right)^\top \left[\widehat{\mathrm{Cov}}\left(\mathrm{vec}(\widehat{B}^{**})\right)\right]^{-1} \left(R\mathrm{vec}(\widehat{B}) - r\right) \approx \chi^2_{(J)}, \tag{19}$$

where R is a known $J \times K^2$ dimensional selection matrix of rank J, and r is a known $J \times 1$ vector, which represents the considered restrictions, such that the composite null hypothesis is $H_0 : R\mathrm{vec}(B) = r$. The matrix $\widehat{B}^{**}$ is the bootstrap version of the covariance decomposition matrix, and can be obtained from one of the bootstrap procedures described in Section 3.6 below.

3.5. Tools for SVAR analysis

The identified structural matrix B can help capturing the dynamic and instantaneous impacts of the structural shocks within the set of variables under consideration. Several tools to analyze these relations are described, for instance, in Kilian and Lütkepohl (2017) and Lütkepohl (2011). The **svars** package provides impulse-response functions, forecast error variance decompositions as well as historical decompositions.

Impulse-response functions

Impulse-response functions describe the impact of isolated unit shocks on the variables of the system with respect to a certain response delay (e.g., the zero delay gives the instantaneous impact). For the model formulation in Equation 1 the response matrices can be derived as follows (see, e.g., Lütkepohl 2005)

$$\begin{aligned} A(L)y_t &= \mu + B\varepsilon_t \\ y_t &= A(L)^{-1}\mu + A(L)^{-1}B\varepsilon_t \\ &= \nu + \Phi(L)B\varepsilon_t = \nu + \sum_{i=0}^{\infty} \Phi_i B\varepsilon_{t-i} = \nu + \sum_{i=0}^{\infty} \Theta_i \varepsilon_{t-i}, \end{aligned}$$

where ν is the unconditional mean of the series and $A(L) = I - A_1L - A_2L^2 - \ldots - A_pL^p$. The elements of $\Theta_i := \Phi_i B$ can be interpreted as the responses of the system to shocks ε_t which summarize the informational content of dynamic parameters in Φ_i, $i = 1, 2, 3, \ldots$ and of the structural matrix B. In particular, $\Theta_0 = B$.

Forecast error variance decompositions

Forecast error variance decompositions (FEVD) highlight the relative contribution of each shock to the variation a variable under scrutiny. For the multivariate series y_t, the corresponding h-step ahead forecast error is $y_{t+h} - y_{t|t}(h) = \Theta_0\varepsilon_{t+h} + \ldots + \Theta_h\varepsilon_{t+1}$, and the forecast error variance of the k-th variable is $\sigma_k^2(h) = \sum_{j=0}^{h-1}(\Theta_{k1,j}^2 + \ldots + \Theta_{kK,j}^2)$ (Lütkepohl 2005). Since $\Sigma_\varepsilon = I_K$ holds by assumption, the relative contribution of shock ε_{it} to the h-step forecast error variance of variable y_{kt} is

$$FEVD_{ki}(h) = (\Theta_{ki,0}^2 + \ldots + \Theta_{ki,h-1}^2)/\sigma_k^2(h).$$

Historical decompositions

Further information on the contribution of structural shocks to a variable of interest can be drawn from historical decompositions. The contribution of shock ε_{jt} to a variable y_{kt} in time period t is

$$y_{kt}^{(j)} = \sum_{i=0}^{t-1} \Theta_{kj,i}\varepsilon_{j,t-i} + \alpha_{j1}^{(t)}y_0 + \ldots + \alpha_{jp}^{(t)}y_{-p+1},$$

where $\alpha_{ji}^{(t)}$ is the j-th row of $A_i^{(t)}$, and $[A_1^{(t)}, \ldots, A_p^{(t)}]$ consists of the first K rows of the companion form matrix with exponent t, $\mathbf{A}^t$ (see Lütkepohl 2005, for more details).

3.6. Bootstrap methods

Wild bootstrap

Inferential issues (e.g., estimating standard errors of point estimates or confidence intervals of impulse-responses) might rely on the so-called wild bootstrap approach, which is robust in case of various forms of heteroskedasticity (Goncalves and Kilian 2004; Hafner and Herwartz 2009). For instance, under a fixed-design, bootstrap samples can be constructed as

$$y_t^* = \widehat{\mu} + \widehat{A}_1 y_{t-1} + \widehat{A}_2 y_{t-2} + \cdots + \widehat{A}_p y_{t-p} + u_t^*, \quad t = 1, \ldots, T, \tag{20}$$

where $\widehat{A}_j$, $j = 1, \ldots, p$, and $\widehat{\mu}$ are LS parameter estimates retrieved from the data. To determine bootstrap error terms $u_t^* = \omega_t \widehat{u}_t$, the scalar random variable ω_t is drawn from a distribution with zero mean and unit variance ($\omega_t \sim (0,1)$) which is independent of the observed data. A prominent distribution choice for sampling ω_t is the Gaussian distribution. Two other frequently considered approaches are drawing ω_t (i) from the so-called Rademacher distribution with ω_t being either unity or minus unity with probability 0.5 (Liu 1988), and (ii) from the distribution suggested by Mammen (1993), where $\omega_t = -(\sqrt{5}-1)/2$ with probability $(\sqrt{5}+1)/(2\sqrt{5})$ or $\omega_t = (\sqrt{5}-1)/2$ with probability $(\sqrt{5}-1)/(2\sqrt{5})$.

For the error terms $\widehat{u}_t^*$, estimated from (20), we determine the bootstrap structural parameter matrix as $\widehat{B}^{**} = \widehat{\Sigma}_u^{1/2}\widehat{\Sigma}_{\widehat{u}^*}^{-1/2}\widehat{B}^*$. Here, $\widehat{B}^*$ is a decomposition of $\widehat{\Sigma}_{\hat{u}^*}$ derived by the described identification procedures. The matrices $\widehat{\Sigma}_u^{1/2}$ and $\widehat{\Sigma}_{\widehat{u}^*}^{1/2}$ are symmetric eigenvalue decompositions of $\widehat{\Sigma}_u$ and $\widehat{\Sigma}_{\widehat{u}^*}$, respectively. Thus, $\widehat{B}^{**}$ provides a factorization of the sample covariance matrix $\widehat{\Sigma}_u$ such that it can be used for inference on the structural parameters as depicted, for instance, in (19).

Moving-block bootstrap

Brüggemann, Jentsch, and Trenkler (2016) suggest the moving-block bootstrap for inference in VAR models characterized by conditional heteroskedasticity. The moving-block bootstrap depends on a chosen block length $\ell < T$, which determines the number of blocks $n = T/\ell$ needed for data generation. The $(K \times \ell)$-dimensional blocks $M_{i,\ell} = (\widehat{u}_{i+1}, \ldots, \widehat{u}_{i+\ell})$, $i = 0, \ldots, T-\ell$, are laid randomly end-to-end together to obtain the bootstrap residuals $u_1^*, \ldots, u_T^*$. After centering the residuals, the bootstrap time series may be constructed recursively as

$$y_t^* = \widehat{\mu} + \widehat{A}_1 y_{t-1}^* + \widehat{A}_2 y_{t-2}^* + \cdots + \widehat{A}_p y_{t-p}^* + u_t^*, \quad t = 1, \ldots, T. \tag{21}$$

It is important to note that asymptotic theory for block bootstrap schemes is typically derived under the assumption that $\ell \to \infty$ as $T \to \infty$. Yet, there is no consensus in the literature on the choice of ℓ in finite samples and, hence, choosing a block length in practice is not straightforward. In general, the chosen block length should ensure that residuals being more than ℓ time points apart from each other are uncorrelated. A more thorough discussion on the choice of the block length can be found in Lahiri (2003). The bootstrap covariance decomposition $\widehat{B}^{**}$ is determined analogously to the case of wild bootstrap sampling described before. Note that both the wild bootstrap and the moving-block bootstrap can be implemented either under a fixed-design as in (20) or a recursive-design as in (21).

Bootstrap-after-bootstrap

Kilian (1998) proposes a bias-corrected bootstrap procedure to account for small sample biases. By means of the so-called bootstrap-after-bootstrap method, the true underlying data generating process (DGP) is not approximated by the bootstrap DGP as in Equation 20 and Equation 21, but rather by means of a bootstrap DGP with bias-corrected VAR parameters $\widehat{\beta}^{BC} = [\widehat{\mu}^{BC}, \widehat{A}_1^{BC}, \ldots, \widehat{A}_p^{BC}]$.

The approach consists of two stages. In the first stage, bootstrap replications for $\widehat{\beta}^* = [\widehat{\mu}^*, \widehat{A}_1^*, \ldots, \widehat{A}_p^*]$ are generated according to Equation 20 or Equation 21, and bias terms are approximated as $\widehat{\Psi} = \bar{\widehat{\beta}}^* - \widehat{\beta}$. Subsequently, the modulus of the largest root of the companion matrix associated with $\widehat{\beta}$ can be calculated, which is denoted by $m(\widehat{\beta})$. If $m(\widehat{\beta}) \geq 1$, $\widehat{\beta}^{BC} = \widehat{\beta}$ is set without any adjustment. However, if $m(\widehat{\beta}) < 1$, then the VAR parameters are corrected such that $\widehat{\beta}^{BC} = \widehat{\beta} - \widehat{\Psi}$.[3]

In the second stage, the actual bootstrap samples can be obtained from substituting $\widehat{\beta}^{BC}$ for $\widehat{\beta}$ in Equation 20 or Equation 21. Kilian (1998) shows by means of a simulation study that in small samples the bootstrap-after-bootstrap method tends to be more accurate than standard bootstrap approaches. Kilian and Lütkepohl (2017) provide more insights into the merits of bias adjustments in resampling, as well as a detailed overview of further bootstrap approaches in the context of SVAR models.

[3] The exact bias correction is an iterative procedure and described in Kilian (1998).

4. Package design

Table 2 summarizes the design of the **svars** package. The package is built around the six core functions for identification of the structural VAR form (`id.cv`, `id.cvm`, `id.dc`, `id.garch`, `id.ngml`, `id.st`). Moreover, various methods and further diagnostic tools are available for the resulting objects of class '`svars`' which have been described in Section 3. In the following, we describe the mandatory and optional input arguments of the implemented functions in a detailed manner.

4.1. Core functions for SVAR identification

To apply the implemented identification techniques the user needs to provide an estimated reduced form VAR or vector error correction model (VECM) object of class '`varest`' or '`vec2var`' from the **vars** package. Alternatively, an object of class '`nlVar`' or '`VECM`' from the **tsDyn** package or the `list` delivered by the function `VAR` of the **MTS** package can serve as an input argument for `id.cv`, `id.cvm`, `id.dc`, `id.garch`, `id.ngml` or `id.st`. Besides the estimated VAR objects, the identification procedures allow for further input arguments which differ across the techniques. In the following, we describe these options separately.

SVAR models built on (co)variance changes

For identification by means of changes in volatility the following command can be used

```
id.cv(x, SB, start = NULL, end = NULL, frequency = NULL, format = NULL,
  dateVector = NULL, max.iter = 50, crit = 0.001, restriction_matrix = NULL)
```

The function `id.cv()` requires the specification of a structural break point. Conditional on the data structure, the user may provide the breakpoint `SB` in various formats. Firstly, the sample can be separated into two parts by specifying the breakpoint in either integer or date formats. Secondly, single time instances can be assigned to a variance regime by passing a vector consisting of zeros and ones to the function. If the estimation of the reduced form VAR is based on a non-time series class object (e.g., '`ts`'), the user can add the information on the date and frequency by making use of the parameter `dateVector` or by specifying `start/end` and `format/frequency`. However, providing time series class objects or specifying dates is optional and the function also handles conventional observation numbers.

The log-likelihood and VAR coefficients are re-estimated in the algorithm until the log-likelihood changes by less than the value of `crit` or the maximum number of iterations (`max.iter`) is reached. Additionally, the function `id.cv()` allows for restricted ML estimation via the input argument `restriction_matrix`. There are two formats of specifying the restriction matrix, either pass (i) a $K \times K$ matrix, in which `NA` indicates unrestricted elements and 0 a restricted element, or (ii) a $K^2 \times K^2$ matrix of rank M, where M is the number of unrestricted coefficients (Lütkepohl 2005). In this case, unit (zero) values on the main diagonal refer to the unrestricted (restricted) coefficients. In case of over-identifying restrictions, `id.cv()` estimates the unrestricted and the restricted SVAR to perform the likelihood ratio test outlined in Section 3.3.

The function

```
id.garch(x, max.iter = 5, crit = 0.001, restriction_matrix = NULL)
```

Function or method	Class	Methods for class	Functions for class	Description
• Core functions for SVAR identification				
◇ SVAR models referred to (co)variance changes				
`id.cv`	‘svars’	`fevd`, `irf`, `print`, `summary`	`hd`, `mb.boot`, `wild.boot`	Estimates the structural shocks via unconditional (co)variance shifts
`id.garch`	‘svars’	`fevd`, `irf`, `print`, `summary`	`hd`, `mb.boot`, `wild.boot`	Estimates the structural shocks through conditional heteroskedasticity
`id.st`	‘svars’	`fevd`, `irf`, `print`, `summary`	`hd`, `mb.boot`, `wild.boot`	Estimates the structural shocks via smooth (co)variance transitions
◇ SVAR models based on independent components				
`id.cvm`	‘svars’	`fevd`, `irf`, `print`, `summary`	`hd`, `mb.boot`, `wild.boot`	Estimates the structural shocks via nonparametric CVM statistic
`id.dc`	‘svars’	`fevd`, `irf`, `print`, `summary`	`hd`, `mb.boot`, `wild.boot`	Estimates the structural shocks via nonparametric distance covariance statistic
`id.ngml`	‘svars’	`fevd`, `irf`, `print`, `summary`	`hd`, `mb.boot`, `wild.boot`	Estimates the structural shocks via parametric non-Gaussian ML
• Functions and methods for SVAR analysis				
◇ Pre-tests and joint significance tests				
`chow.test`	‘chow’	`print`, `summary`		Computes Chow test types on structural breaks
`stability`	‘chowpretest’	`plot`, `print`	`chow.test`	Performs multiple Chow tests in prespecified range
`js.test`	‘jstest’	`print`, `summary`		Performs chi-square test on joint parameter significance
◇ Further SVAR statistics				
`irf`	‘svarirf’	`plot`, `print`		Calculates impulse-response functions
`fevd`	‘svarfevd’	`plot`, `print`		Calculates forecast error variance decomposition
`hd`	‘hd’	`plot`, `print`		Computes historical decomposition
◇ Bootstrap procedures				
`mb.boot`	‘sboot’	`plot`, `print`, `summary`	`ba.boot`, `js.test`	Moving-block bootstrap for inferential analysis.
`wild.boot`	‘sboot’	`plot`, `print`, `summary`	`ba.boot`, `js.test`	Wild bootstrap for inferential analysis.
`ba.boot`	‘sboot’	`plot`, `print`, `summary`	`ba.boot`, `js.test`	Bootstrap-after-bootstrap for bias correction in inferential analysis

Table 2: Package design of **svars**. The first column lists all functions or methods contained in the package and the second column shows the class of the resulting object from these functions or methods. The third and fourth columns show which functions or methods can be used to further process the corresponding object class. In the last column a short description of the function or method is given.

provides model identification if structural shocks exhibit conditional heteroskedasticity. Identification proceeds in two steps. In the first step K univariate GARCH(1,1) models (see Equation 10) are estimated. In the second step a full, joint ML estimation of the parameters in B is performed. These two steps are executed until the multivariate log-likelihood changes by less than the value of `crit` or the maximum number of iterations (`max.iter`) is reached. Analogously to the `id.cv()` function, passing a `restriction_matrix` enables the user to estimate and test restricted models.

Identification by means of smooth covariance transitions is implemented as

```
id.st(x, nc = 1, c_lower = 0.3, c_upper = 0.7, c_step = 5, c_fix = NULL,
  transition_variable = NULL, gamma_lower = -3, gamma_upper = 2,
  gamma_step = 0.5, gamma_fix = NULL, max.iter = 5, crit = 0.001,
  restriction_matrix = NULL, lr_test = FALSE)
```

which entails several input arguments for adjustments. However, the user may run the function without any further specifications of input arguments only by passing the reduced form estimated VAR object. Since finding the optimal parameters γ and c as described in Section 2.2 is computationally demanding, the `id.st` function supports parallelization with `nc` determining the number of cores used. Grid optimization is optional. By default, the function searches for the transition point c to be located between $0.3T$ (`c_lower`) and $0.7T$ (`c_upper`) with a step width of 5 time points (`c_step`). If the user wants to specify the transition point in advance, she can pass an observation number to `c_fix`. Analogously, for the slope parameter γ the user can either specify a fixed slope parameter `gamma_fix`, or let the function optimize the transition coefficient between `gamma_lower` and `gamma_upper`.

Conditional on the location (c) and slope (γ) parameter the algorithm consists of an iterative procedure of log-likelihood optimization and GLS estimation until the improvement of the log-likelihood is smaller than `crit` or the maximum number of iterations (`max.iter`) is reached. By default, the transition variable corresponds to time, however, the user may choose another transition variable by passing a numeric vector to `transition_variable`. Note that the input argument for the location parameter has to be adjusted to the scale of the transition variable. Analogously to the previous functions, passing a `restriction_matrix` enables the estimation of restricted models. Due to the fact that the smooth transition covariance model is computationally demanding, it is possible to decide if the function performs a likelihood ratio test or not by specifying `lr_test` as either `TRUE` or `FALSE`.

SVAR models built on independent components

For identifying independent components by means of the CVM distance the function

```
id.cvm(x, dd = NULL, itermax = 500, steptol = 100, iter2 = 75)
```

can be employed. In Section 2 we have elaborated on how this approach evaluates a CVM test for rotated versions of the shocks. We use the implementation of the CVM test in the package **copula** (Kojadinovic and Yan 2010). The function `indepTestSim` from the **copula** package generates an independent sample to calculate the p value for the test statistic. The sample is passed to the **svars** function `id.cvm` as argument `dd`. If `dd = NULL` the sample is simulated within the `id.cvm` function. Simulating the independent sample in advance and passing the object to the `id.cvm` function may save computation time if the estimation is repeatedly applied to the same data set. The estimation of independent components through CVM statistics proceeds in two steps. The first stage is a global optimization using the differential evolution algorithm from the **DEoptim** package (Mullen, Ardia, Gil, Windover, and Cline 2011). In the second stage, the test statistic is optimized locally around the estimated parameters from the first stage. The precision of the algorithm can be determined by the input arguments `itermax` and `steptol` at the first stage (for more details see the help file of **DEoptim**) and `iter2` at the second stage.

The function

```
id.dc(x, PIT = FALSE)
```

identifies the structural shocks by means of distance covariance statistics. The implementation is built on the ICA algorithm from the package **steadyICA** (Risk *et al.* 2015). The function `steadyICA` therein applies a gradient algorithm to determine the minimum of the dependence criterion. The option `PIT` determines if probability integral transformation (PIT) is applied to transform the marginal densities of the structural shocks prior to the evaluation of the dependence criterion.

Estimating the structural shocks via non-Gaussian ML estimation is implemented with the function

```
id.ngml(x, stage3 = FALSE, restriction_matrix = NULL)
```

The input argument `stage3` indicates if the autoregressive parameters of the VAR model are estimated by maximizing the log-likelihood function which Lanne *et al.* (2017) describe as the third step of their model. Since this step does not change the result of the estimated covariance decomposition, and the estimation of the autoregressive parameter is computationally rather demanding, the default is set to `FALSE`. Analogously to the functions `id.cv`, `id.garch` and `id.st`, the user may run a restricted estimation by passing an appropriate `restriction_matrix` argument to `id.ngml`.

All identification functions (`id.cv`, `id.garch`, `id.st`, `id.cvm`, `id.dc`, `id.ngml`) return an object of class '`svars`'. The `summary` method for this class returns the estimated impact relation matrix with standard errors and various further information depending on the chosen identification method, while `print` only returns the covariance decomposition. The `plot` method is only applicable to objects from the function `id.st` and shows the optimized transition function of the variance from the first to the second volatility regime.

4.2. Functions and methods for SVAR analysis

The following functions and methods are built around the cornerstone functions which have been introduced in the last section. To obtain a user-friendly environment within the **svars** package, most of the implementations are feasible only by passing an object of class '`svars`' or '`sboot`' and leaving further specifications optional. Moreover, to facilitate compatibility with other `R` packages, we refer to the **vars** package (Pfaff 2008), and adapt methods for parameter tests, impulse-response analysis and forecast error variance decompositions.

Pre-tests and joint significance tests

For prior analysis of parameter stability, the function

```
chow.test(x, SB, nboot = 500, start = NULL, end = NULL,
  frequency = NULL, format = NULL, dateVector = NULL)
```

includes two versions of structural break tests. The input argument `x` needs to be a reduced form estimation result of class '`varest`', '`vec2var`' or '`nlVar`'. The time point of the assumed structural break has to be passed in `SB`. The user can work with date formats, in the same way as described for the `id.cv()` function above. To calculate the p values and critical values, the function employs a fixed-design wild bootstrap. The number of bootstrap replications needs to be provided by `nboot`. The `summary()` method returns the results from the sample split and break point tests. Additionally, the package includes an augmentation of the `stability()` method of the **vars** package (Pfaff 2008), which provides access to a variety of parameter stability analysis tools of **strucchange** (Zeileis *et al.* 2002). The method has been extended to contain multivariate Chow tests

```
stability(x, type = "mv-chow-test", h = 0.15)
```

By specifying `type = "mv-chow-test"` and `h = 0.15` the test statistics for all possible structural break points between (`h`/2)T and (1-`h`/2)T are calculated. The resulting object of class '`chowpretest`' from `stability()` can be used as an input argument for `x` in `chow.test()` afterwards without any further input specifications. Subsequently, the function provides the test results for the structural break at the observation with the corresponding highest break point test statistic resulting from `stability()`.

After obtaining point- and bootstrap estimates, the user can test joint hypotheses on the estimated elements in the structural matrix B by means of the function

```
js.test(x, R, r = NULL)
```

where `x` is an object of class '`sboot`'. If `r = NULL`, the function performs a test of the hypothesis $H_0 : R\text{vec}(B) = 0$.

Further SVAR statistics

Following the descriptions in Section 3, impulse-response functions can be calculated using

```
irf(x, n.ahead = 20)
```

where `x` is an object of class '`svars`'. The user can specify the time horizon of the impulse-response functions, which is 20 periods by default. The same input arguments are passed to calculate forecast error variance decompositions using

```
fevd(x, n.ahead = 10)
```

Historical decompositions are calculated by the function

```
hd(x, series = 1)
```

By default, the first series, i.e., the series in the first column of the original data set is decomposed. For all three analysis tools `plot` methods are available to visualize the resulting objects.

Bootstrap procedures

The bootstrap procedures described in Section 3 are implemented in the functions `mb.boot`, `wild.boot` and `ba.boot`. The required input object `x` is of class '`svars`'. Furthermore, it is possible to record how often one or multiple bootstrap shocks hold a specific sign pattern. This helps to evaluate the plausibility of the signs of instantaneous effects as described in Herwartz (2018). The appearance of specific sign patterns is documented by passing a list of vectors containing 1 and -1 to the input argument `signrest`. Every list entry represents the impact effects of a shock to the variables in the system. Thus, each list entry is of the same size as the VAR model, i.e., contains K elements. The list can consist of 1 up to K entries, one for each structural shock. By default, the bootstrap functions evaluate the occurrence of the sign pattern of the point estimate. The R function for the moving-block bootstrap is

```
mb.boot(x, design = "recursive", b.length = 15, n.ahead = 20,
  nboot = 500, nc = 1, dd = NULL, signrest = NULL, itermax = 300,
  steptol = 200, iter2 = 50)
```

where the user needs to specify the block length with input argument `b.length`. As described in Section 3.6 there is no consensus in the literature about the optimal block length in finite samples. In applied work, however, a typical block length is about 10% of the sample size (see, e.g., Brüggemann *et al.* 2016; Lütkepohl and Schlaak 2019). The wild bootstrap method is implemented as

```
wild.boot(x, design = "fixed", distr = "rademacher", n.ahead = 20,
  nboot = 500, nc = 1, dd = NULL, signrest = NULL, itermax = 300,
  steptol = 200, iter2 = 50)
```

The user can choose to draw ω_t from a Rademacher distribution with `distr = "rademacher"`, from a Gaussian distribution with `distr = "gaussian"` or from the distribution suggested in Mammen (1993) with `distr = "mammen"`. The remaining input arguments for the two bootstrap functions are identical, e.g., both can be called as fixed-design (`design = "fixed"`) or as recursive-design (`design = "recursive"`). Bootstrap impulse-responses are calculated in the functions for which the horizon needs to be determined via `n.ahead`. An integer for the number of bootstrap replications is supplied by the `nboot` argument. Parallelization is possible with a suitable choice of `nc`. The arguments `dd`, `itermax`, `steptol` and `iter2` correspond to the input arguments of the `id.cvm` model and are only applied if the point estimates have been derived by this method. Both bootstrap functions return an object of class '`sboot`' for which `summary` and `plot` methods can be applied.

Furthermore, the bootstrap-after-bootstrap procedure is implemented as

```
ba.boot(x, nc = 1)
```

In contrast to the other bootstrap functions of **svars**, `x` is of class '`sboot`', since the function only performs the bias correction and the second step of the procedure described above in Section 3.6. The necessary results from the first step of the algorithm are determined from the bootstrap object, which is passed to the function to obtain the most efficient implementation of this hierarchical bootstrap procedure. The second step bootstrap (after the bias correction) is executed with exactly the same specifications as in the first stage. Hence, no further input arguments are needed.

5. Example

To illustrate the functions and methods of the **svars** package, we replicate the empirical results of Herwartz and Plödt (2016b) obtained through the identification by means of unconditional covariance shifts (`id.cv()`). We augment their analysis by further statistics and complement the analysis with results from identification through independent components using the DC approach (`id.dc()`).[4] The main objective of the application in this Section is to present the usage of the functions rather than discussing the results in depth.

Herwartz and Plödt (2016b) apply identification by means of the CV approach to investigate the effects of a monetary policy shock on the economy of the United States (US). They consider three series: the output gap `"x"`, which is defined as the log-deviation of real gross domestic product (GDP) from the potential output, the inflation `"pi"` as quarter-on-quarter growth rates of the GDP deflator and the federal funds rate `"i"`. The data comes from the Federal Reserve Economic Data (FRED) database of the Federal Reserve Bank of St. Louis. The time series are sampled at the quarterly frequency and cover the time period from 1965Q1 until 2008Q3. The **svars** package contains this example data set labeled `"USA"`.

The first step of the analysis is to load the **svars** package into the workspace. Furthermore, the **ggplot2** (Wickham 2016) and **ggfortify** (Tang, Horikoshi, and Li 2016) packages enable to display the data in a convenient way.

```
R> library("svars")
R> library("ggplot2")
R> library("ggfortify")
R> data("USA", package = "svars")
```

In order to estimate the structural shocks via the `id.cv()` function, the user has to specify the time point of the variance shift in advance. An appropriate time point might be found by visual inspection of the series, historical information or previous analyses. Figure 1 depicts the three time series. Inflation data (`"pi"`) show less fluctuation during the second half of the data set.

```
R> autoplot(USA, facets = TRUE) + theme_bw() + ylab("")
```

Herwartz and Plödt (2016b) determine the break point at 1979Q3 due to a policy shift of the Federal Reserve Bank which caused a reduction of the volatility in US macroeconomic data (Stock and Watson 2003). The next step of the analysis is the estimation of the reduced form VAR, for instance, by means of the function `VAR()` from the **vars** package. We specify a VAR model with intercept of order $p = 6$. After model estimation, we can use the resulting `varest` object to estimate the structural form with the function `id.cv()`. We provide the structural break point with the function argument `SB` in `ts` date format.

```
R> plain.var <- vars::VAR(USA, p = 6, type = "const")
R> usa.cv <- id.cv(plain.var, SB = c(1979, 3))
R> summary(usa.cv)
```

[4] Estimation via the CVM criterion and DC deliver qualitatively the same results. Identifying independent components by means of NGML and ST models provide results that are comparable to those obtained from assuming covariance shifts. Identification via the assumption of GARCH-type variances obtains results which are qualitatively different from those of all other approaches.

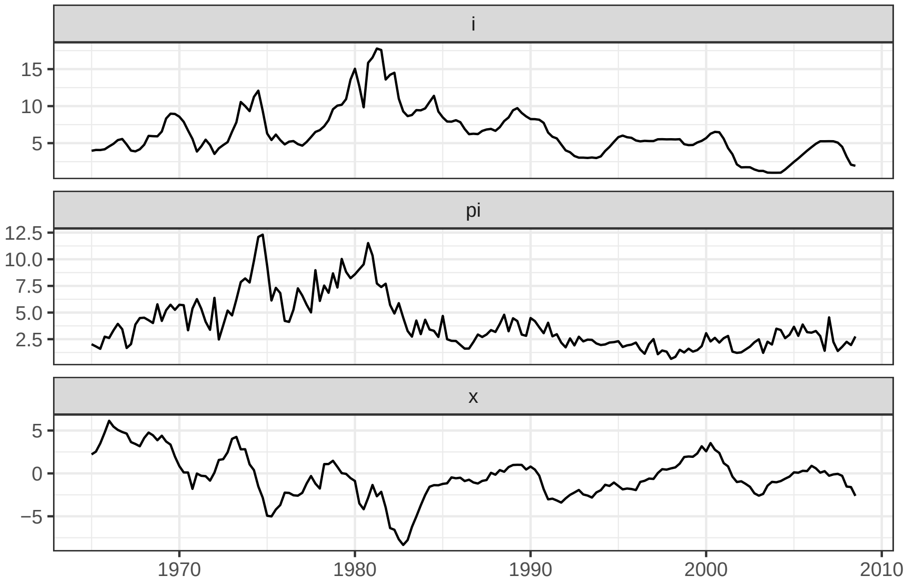

Figure 1: US macroeconomic data.

```
Identification Results
----------------------

Method: Changes in Volatility
Sample size: 169
Log-Likelihood: -564.2994
AIC: 1268.599
Structural Break: At Observation Number 59 during 1979 Q3
Number of GLS estimations: 22
Number of Restrictions: 0

Estimated unconditional Heteroscedasticity Matrix (Lambda):
        [,1]     [,2]     [,3]
x  0.3925906 0.000000 0.000000
pi 0.0000000 0.191641 0.000000
i  0.0000000 0.000000 1.244348

Standard Errors of Lambda:
         [,1]       [,2]      [,3]
x  0.09265819 0.00000000 0.0000000
pi 0.00000000 0.04527264 0.0000000
i  0.00000000 0.00000000 0.2935571

Estimated B Matrix (unique decomposition of the covariance matrix):
          [,1]       [,2]      [,3]
x   0.61193300 -0.5931964 0.2241237
```

```
pi  0.75559400  1.2987520 0.1131134
i  -0.02899916  0.1572953 0.7084709

Standard Errors of B:
        [,1]      [,2]       [,3]
x  0.1330924 0.1955350 0.07101215
pi 0.2498466 0.2600376 0.09960245
i  0.1559672 0.1213446 0.07004430

Pairwise Wald Test:
                  Test statistic p-value
lambda_1=lambda_2           3.80    0.05 *
lambda_1=lambda_3           7.66    0.01 **
lambda_2=lambda_3          12.56  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The summary of the identified object displays the estimated decomposition of the covariance matrix $\widehat{B}$, as well as the covariance shift matrix $\widehat{\Lambda}$ and their corresponding standard errors. Moreover, the summary provides the results of pairwise Wald-type tests for distinct diagonal elements of $\widehat{\Lambda}$ which is necessary for unique identification of the structural shocks. In the present case, all three tests statistics yield a rejection of the null hypotheses of equal diagonal elements with 10% significance. The ordering of the columns of $\widehat{B}$ is arbitrary and the user has to arrange them in an economically meaningful way. For instance, Herwartz and Plödt (2016b) order the columns according to a unique sign pattern which indicates the direction of the shocks on impact. The code below orders the columns in the same way.

```
R> usa.cv$B <- usa.cv$B[, c(3, 2, 1)]
R> usa.cv$B[, 3] <- usa.cv$B[, 3] * (-1)
R> usa.cv$B_SE <- usa.cv$B_SE[, c(3, 2, 1)]
R> usa.cv$Lambda <- diag(diag(usa.cv$Lambda)[c(3, 2, 1)])
R> usa.cv$Lambda_SE <- diag(diag(usa.cv$Lambda_SE)[c(3, 2, 1)])
R> round(usa.cv$B, 3)

    [,1]   [,2]   [,3]
x  0.224 -0.593 -0.612
pi 0.113  1.299 -0.756
i  0.708  0.157  0.029

R> round(usa.cv$Lambda, 3)

    [,1]  [,2]  [,3]
x  1.244 0.000 0.000
pi 0.000 0.192 0.000
i  0.000 0.000 0.393
```

Herwartz and Plödt (2016b) interpret the impact effects in the first column of the matrix $\widehat{B}$ to characterize a demand shock. Similarly, the effects in the second (third) column indicate a supply (monetary policy) shock. The authors argue that their shock labeling according to the estimated sign patterns is in line with the relevant literature. Since the matrix $\widehat{\Lambda}$ represents the variance of structural shocks in the second regime, Herwartz and Plödt (2016b) interpret the diagonal elements of $\widehat{\Lambda}$ such that the supply and monetary policy shocks have relatively lower variances and the demand shock a higher variance in regime two (i.e., for time instances $t > T_{SB} = 59$ or after the second quarter of 1979). The authors compare the results from this statistical identification scheme with a model structure implied by covariance decomposition matrix B which is lower triangular by assumption (Sims 1980). The `id.cv()` function enables the user to test for such restrictions by setting up a restriction matrix as described in the code below.

```
R> restMat <- matrix(rep(NA, 9), ncol = 3)
R> restMat[1, c(2, 3)] <- 0
R> restMat[2, 3] <- 0
R> restMat

     [,1] [,2] [,3]
[1,]   NA    0    0
[2,]   NA   NA    0
[3,]   NA   NA   NA

R> restricted.model <- id.cv(plain.var, SB = c(1979, 3),
+    restriction_matrix = restMat)
R> summary(restricted.model)

Identification Results
----------------------

Method: Changes in Volatility
Sample size: 169
Log-Likelihood: -568.6664
AIC: 1277.333
Structural Break: At Observation Number 59 during 1979 Q3
Number of GLS estimations: 23
Number of Restrictions: 3

Estimated unconditional Heteroscedasticity Matrix (Lambda):
        [,1]      [,2]      [,3]
x  0.3501948 0.0000000 0.0000000
pi 0.0000000 0.2346854 0.0000000
i  0.0000000 0.0000000 0.9420116

Standard Errors of Lambda:
         [,1]       [,2]     [,3]
x  0.08266738 0.00000000 0.000000
pi 0.00000000 0.05616318 0.000000
i  0.00000000 0.00000000 0.227189
```

```
Estimated B Matrix (unique decomposition of the covariance matrix):
         [,1]      [,2]      [,3]
x  0.87988465 0.0000000 0.0000000
pi 0.08137972 1.5306503 0.0000000
i  0.31518384 0.2606745 0.7378484

Standard Errors of B:
         [,1]       [,2]       [,3]
x  0.08638851 0.00000000 0.00000000
pi 0.10334026 0.15169565 0.00000000
i  0.08527442 0.08620187 0.07354585

Pairwise Wald Test:
                  Test statistic p-value
lambda_1=lambda_2           1.34    0.25
lambda_1=lambda_3           5.99    0.01 **
lambda_2=lambda_3           9.13  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Likelihood Ratio Test:
 Test statistic p-value
          8.734   0.033 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Since the structural shocks are just identified by the change in the covariance matrix, any further restriction on B over identifies the model and makes the restrictions testable. The function automatically performs a likelihood ratio test in case of such over identifying restrictions. The summary depicts the estimation results from the restricted model as well as the test statistics and p values. The likelihood ratio test indicates that the null hypothesis of a lower triangular structural impact matrix has to be rejected at the 5% significance level. Herwartz and Plödt (2016b) argue that identification by means of zero restrictions according to a lower triangular matrix lacks economic intuition which we can support with the obtained diagnostic. Therefore, the unrestricted model should be preferred for further analysis.

The next step is the calculation of impulse-response functions with bootstrap confidence bands to investigate future effects of the economically labeled structural shocks on the variables included in the model. Moreover, the implemented bootstrap functions allow for an evaluation of the significance of unique sign patterns in $\widehat{B}$ as described in Herwartz (2018). We define a list of sign restrictions and label them as demand, supply and monetary policy shock respectively.

```
R> signrest <- list(demand = c(1, 1, 1), supply = c(-1, 1, 1),
+    monetary_policy = c(-1, -1, 1))
```

For illustration, we use the wild bootstrap implemented with a Rademacher distribution, fixed-design and 1000 bootstrap replications as in Herwartz and Plödt (2016b). To reduce computation time we parallelize the bootstrap and specify a seed to obtain reproducible results. The time horizon for the impulse-response analysis has to be determined in advance using the argument `n.ahead`.

```
R> cores <- parallel::detectCores() - 1
R> set.seed(231)
R> usa.cv.boot <- wild.boot(usa.cv, design = "fixed", distr = "rademacher",
+   nboot = 1000, n.ahead = 15, nc = cores, signrest = signrest)
R> summary(usa.cv.boot)

Bootstrap Results
-----------------

Method: Wild bootstrap
Bootstrap iterations: 1000
Distribution used: rademacher
Design: fixed

Point estimates:
        [,1]       [,2]        [,3]
x  0.2241237 -0.5931964 -0.61193300
pi 0.1131134  1.2987520 -0.75559400
i  0.7084709  0.1572953  0.02899916

Bootstrap means:
         [,1]        [,2]       [,3]
x  0.09580768 -0.50304819 -0.6303871
pi 0.09639670  1.13109064 -0.7720154
i  0.70100300  0.03648028 -0.1675660

Bootstrap standard errors:
         [,1]      [,2]      [,3]
x  0.14143455 0.3077347 0.2494870
pi 0.17500253 0.4556690 0.5925733
i  0.07511815 0.2273603 0.2178136

Identified sign patterns:
=========================
Specified sign pattern:

   demand supply monetary_policy
x       1     -1              -1
pi      1      1              -1
i       1      1               1

Unique occurrence of single shocks according to sign pattern:
demand : 65.5 %
```

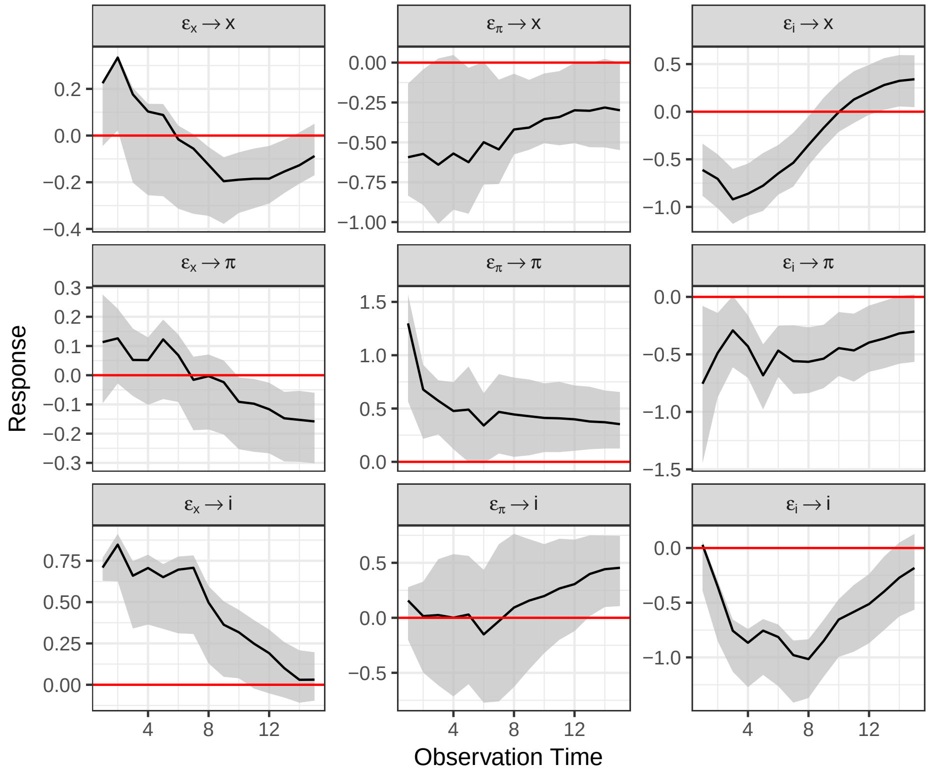

Figure 2: Impulse-response functions with 68% confidence bands based on 1000 bootstrap replications. Structural shocks identified through unconditional shift in the covariance structure.

```
supply : 66 %
monetary_policy : 28.4 %

Joint occurrence of specified shocks: 12.7 %

R> plot(usa.cv.boot, lowerq = 0.16, upperq = 0.84)
```

The summary reveals that only 12.7% of all bootstrap estimates are in line with all economically motivated sign patterns jointly. The sign pattern of the monetary policy shock appears in only 28.4% of all bootstrap samples. Furthermore, the bootstrap means indicate that the third shock is more in line with the sign pattern of the demand shock. This result is plausible noting that the point estimate in the lower right corner is close to zero and, therefore, lacks a significantly positive effect on the interest rate. Figure 2 shows the impulse-response functions of normalized shocks having unit variance in the first regime. Herwartz and Plödt (2016b) argue that the negative reaction of the interest rate to a monetary policy shock after the initial period is implausible, and puts the interpretation of this shock as a monetary policy shock into question. The results from the bootstrap support the authors' argumentation with regard to the shock labeling.

Furthermore, we can calculate the forecast error variance decomposition to investigate the contribution of each shock to the prediction mean squared error of the variables. The `fevd()`

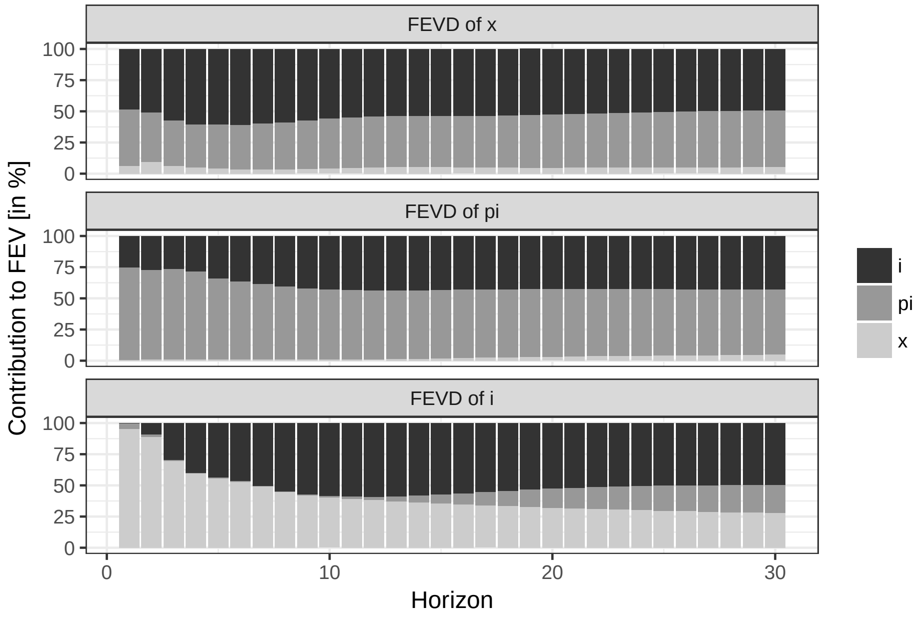

Figure 3: Forecast error variance decomposition for 30 periods. Structural shocks identified by means of the CV model.

method creates an object for visual inspection of the forecast error variance decomposition by means of the `plot` function.

```
R> fev.cv <- fevd(usa.cv, n.ahead = 30)
R> plot(fev.cv)
```

Figure 3 depicts the forecast error variance decomposition. It is evident that the monetary policy shock accounts for more than 50% of the prediction mean squared error of the output gap, whereas the demand shock constantly accounts for only about 5% of the prediction mean squared error. Moreover, the demand shock contributes almost 100% of the forecast error variance of the interest rates on impact. Thus, the forecast error decompositions hint at a shock labeling which differs from the one developed above on the basis of sign patterns of $\widehat{B}$. Furthermore, they confirm the conclusion of Herwartz and Plödt (2016b) that the empirical model fails to identify a monetary policy shock according to its theoretical effect patterns.

We re-estimate the structural form with the DC method under the assumption of independent non-Gaussian shocks.[5]

```
R> usa.dc <- id.dc(plain.var, PIT = FALSE)
R> summary(usa.dc)

Identification Results
----------------------
```

[5] Component-wise kurtosis and skewness tests as implemented in the package **normtest** (Gavrilov and Pusev 2014) as well as fourth-order blind identification-based tests from the package **ICtest** (Nordhausen *et al.* 2019) show no indication for Gaussian components.

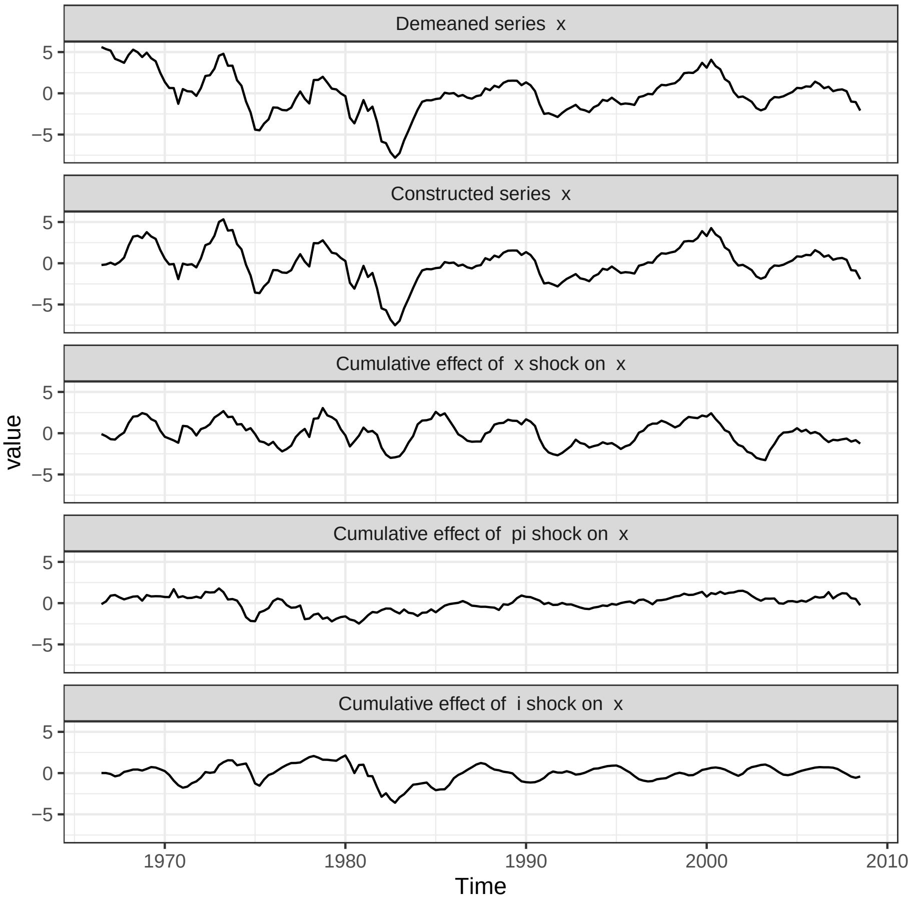

Figure 4: Historical decomposition of the US output gap in percent deviations from the mean. Structural shocks are identified by means of the DC algorithm.

```
Method: Distance covariances
Sample size: 169

Estimated B Matrix (unique decomposition of the covariance matrix):
           [,1]        [,2]      [,3]
x  0.541926899 -0.36707854 0.1964223
pi 0.508827712  0.92428628 0.1967426
i  0.003560267  0.02576151 0.8194037
```

The estimated structural matrix differs from the estimated matrix obtained from the CV approach. The matrix identified by means of the DC method does not allow for a labeling of the shocks that accords with a unique sign pattern. Nevertheless, it is possible to label the shocks in a meaningful way, since one could assume that the loading of the structural shocks on reduced form errors is stronger for own effects in comparison with cross variable effects. The finding that a positive monetary policy shock has a positive effect on output and inflation might seem to be at odds with intuition at first, although this mechanism can be observed rather frequently in the literature (e.g., Lütkepohl and Netsunajev 2017a) and is usually referred to as a so-called price puzzle (Eichenbaum 1992). Conditional on the estimate $\widehat{B}$, we construct the historical decomposition. As an example, we decompose the output into its underlying determinants over the sample period. In the data set output is the first column and, hence, `series = 1` is the provided option.

```
R> hd.cv.1 <- hd(usa.dc, series = 1)
R> plot(hd.cv.1)
```

Figure 4 indicates that output fluctuations are mainly explained by demand shocks rather than supply or monetary policy shocks.

6. Summary

The R package **svars** provides a vast set of estimation techniques that build on several assumptions on the data and a variety of input arguments. In the present article we describe how the implemented identification techniques for SVAR models depend on assumptions of heteroskedasticity and independence coupled with non-Gaussianity to retrieve the structural shocks from the reduced form VAR model. Furthermore, we provide a set of auxiliary functions which complement the cornerstone identification methods, and thereby offer a complete toolbox for structural analysis in a multivariate time series context.

We give a step-by-step guideline on how to use the functions on a real dataset comparing one representative from both groups of heteroskedasticity and independence-based identification. Even though the estimation results are similar, identification by means of covariance shifts might imply a misleading sign pattern which is indicated in the forecast error variance decomposition. Moreover, we illustrate how to test sign and zero restrictions by means of restricted log-likelihood estimation and bootstrap methods.

The **svars** package contains six alternative and recent SVAR identification techniques. Besides these, further popular data-driven identification approaches include, e.g., the heteroskedastic model with Markov switching mechanisms (Lanne, Lütkepohl, and Maciejowska 2010; Herwartz and Lütkepohl 2014) or pseudo ML estimation (Gourieroux, Monfort, and Renne 2017). Moreover, an option to test for long run restrictions by means of likelihood-based identification schemes is a possible augmentation of the package. We regard both directions as promising for future developments of **svars**.

Acknowledgments

We thank the editors and two anonymous reviewers for helpful comments and suggestions. Furthermore, we thank Bernhard Pfaff for his kind help. Financial support of the DFG through project HE 2188/12-1 and 192626868 as well as the Academy of Finland (308628) are gratefully acknowledged.

References

Blanchard OJ, Quah D (1989). "The Dynamic Effects of Aggregate Demand and Supply Disturbances." *American Economic Review*, **79**(4), 655–673. URL `http://www.jstor.org/stable/1827924`.

Bouakez H, Normandin M (2010). "Fluctuations in the Foreign Exchange Market: How Important Are Monetary Policy Shocks?" *Journal of International Economics*, **81**(1), 139–153. doi:10.1016/j.jinteco.2009.11.007.

Brüggemann R, Jentsch C, Trenkler C (2016). "Inference in VARs with Conditional Heteroskedasticity of Unknown Form." *Journal of Econometrics*, **191**(1), 69–85. doi:10.1016/j.jeconom.2015.10.004.

Candelon B, Lütkepohl H (2001). "On the Reliability of Chow-Type Tests for Parameter Constancy in Multivariate Dynamic Models." *Economics Letters*, **73**(2), 155–160. ISSN 0165-1765. doi:10.1016/s0165-1765(01)00478-5.

Comon P (1994). "Independent Component Analysis, A New Concept?" *Signal Processing*, **36**(3), 287–314. ISSN 0165-1684. doi:10.1016/0165-1684(94)90029-9.

Danne C (2015). ***VARsignR**: Sign Restrictions, Bayesian, Vector Autoregression Models.* R package version 0.1.3, URL https://CRAN.R-project.org/package=VARsignR.

Di Narzo AF, Aznarte JL, Stigler M, Tsung-wu H (2020). ***tsDyn**: Nonlinear Time Series Models with Regime Switching.* R package version 10-1.2, URL https://CRAN.R-project.org/package=tsDyn.

Eddelbuettel D, François R (2011). "**Rcpp**: Seamless R and C++ Integration." *Journal of Statistical Software*, **40**(8), 1–18. doi:10.18637/jss.v040.i08.

Eddelbuettel D, Sanderson C (2014). "**RcppArmadillo**: Accelerating R with High-Performance C++ Linear Algebra." *Computational Statistics & Data Analysis*, **71**, 1054–1063. doi:10.1016/j.csda.2013.02.005.

Eichenbaum M (1992). "Interpreting the Macroeconomic Time Series Facts: The Effects of Monetary Policy': By Christopher Sims." *European Economic Review*, **36**(5), 1001–1011. ISSN 0014-2921. doi:10.1016/0014-2921(92)90042-u.

Gavrilov I, Pusev R (2014). ***normtest**: Tests for the Composite Hypothesis of Normality.* R package version 1.1, URL https://CRAN.R-project.org/package=normtest.

Genest C, Quessy JF, Rémillard B (2007). "Asymptotic Local Efficiency of Cramér-Von Mises Tests for Multivariate Independence." *The Annals of Statistics*, **35**(1), 166–1991. ISSN 00905364. URL http://www.jstor.org/stable/25463552.

Goncalves S, Kilian L (2004). "Bootstrapping Autoregressions with Conditional Heteroskedasticity of Unknown Form." *Journal of Econometrics*, **123**(1), 89–120. ISSN 0304-4076. doi:10.1016/j.jeconom.2003.10.030.

Gourieroux C, Monfort A, Renne JP (2017). "Statistical Inference for Independent Component Analysis: Application to Structural VAR Models." *Journal of Econometrics*, **196**(1), 111–126. ISSN 0304-4076. doi:10.1016/j.jeconom.2016.09.007.

Hafner CM, Herwartz H (2009). "Testing for Linear Vector Autoregressive Dynamics under Multivariate Generalized Autoregressive Heteroskedasticity." *Statistica Neerlandica*, **63**(3), 294–323. doi:10.1111/j.1467-9574.2009.00424.x.

Hamilton JD (1994). *Time Series Analysis*. 1st edition. Princeton University Press.

Herwartz H (2018). "Hodges Lehmann Detection of Structural Shocks – An Analysis of Macroeconomic Dynamics in the Euro Area." *Oxford Bulletin of Economics and Statistics*, **80**(4), 736–754. doi:10.1111/obes.12234.

Herwartz H, Lange A, Maxand S (2019). "Statistical Identification in SVARs – Monte Carlo Experiments and a Comparative Assessment of the Role of Economic Uncertainties for the US Business Cycle." *CEGE Discussion Paper 375*, Center for European Governance and Economy. doi:10.2139/ssrn.3418405.

Herwartz H, Lütkepohl H (2014). "Structural Vector Autoregressions with Markov Switching: Combining Conventional with Statistical Identification of Shocks." *Journal of Econometrics*, **183**(1), 104–116. ISSN 0304-4076. doi:10.1016/j.jeconom.2014.06.012.

Herwartz H, Plödt M (2016a). "The Macroeconomic Effects of Oil Price Shocks: Evidence from a Statistical Identification Approach." *Journal of International Money and Finance*, **61**(C), 30–44. ISSN 0261-5606. doi:10.1016/j.jimonfin.2015.11.001.

Herwartz H, Plödt M (2016b). "Simulation Evidence on Theory-Based and Statistical Identification under Volatility Breaks." *Oxford Bulletin of Economics and Statistics*, **78**(1), 94–112. ISSN 1468-0084. doi:10.1111/obes.12098.

Hodges J, Lehmann E (2006). *Hodges-Lehmann Estimators*. Encyclopedia of Statistical Sciences.

Kilian L (1998). "Small-Sample Confidence Intervals for Impulse Response Functions." *The Review of Economics and Statistics*, **80**(2), 218–230. doi:10.1162/003465398557465.

Kilian L, Lütkepohl H (2017). *Structural Vector Autoregressive Analysis*. Cambridge University Press.

Kojadinovic I, Yan J (2010). "Modeling Multivariate Distributions with Continuous Margins Using the **copula** R Package." *Journal of Statistical Software*, **34**(9), 1–20. doi:10.18637/jss.v034.i09.

Lahiri SN (2003). *Resampling Methods for Dependent Data*. 1st edition. Springer-Verlag.

Lanne M, Lütkepohl H (2008). "Identifying Monetary Policy Shocks via Changes in Volatility." *Journal of Money, Credit and Banking*, **40**(9), 1131–1149. doi:10.1111/j.1538-4616.2008.00151.x.

Lanne M, Lütkepohl H, Maciejowska K (2010). "Structural Vector Autoregressions with Markov Switching." *Journal of Economic Dynamics and Control*, **34**(2), 121–131. ISSN 0165-1889. doi:10.1016/j.jedc.2009.08.002.

Lanne M, Meitz M, Saikkonen P (2017). "Identification and Estimation of Non-Gaussian Structural Vector Autoregressions." *Journal of Econometrics*, **196**(2), 121–131. ISSN 0304-4076. doi:10.1016/j.jeconom.2016.06.002.

Lanne M, Saikkonen P (2007). "A Multivariate Generalized Orthogonal Factor GARCH Model." *Journal of Business & Economic Statistics*, **25**(1), 61–75. doi:10.1198/073500106000000404.

Liu RY (1988). “Bootstrap Procedures under Some Non-i.i.d. Models.” *The Annals of Statistics*, **16**(4), 1696–1708. ISSN 0378-3758. doi:10.1016/0378-3758(94)00008-j.

Lütkepohl H (2005). *New Introduction to Multiple Time Series Analysis.* 1st edition. Springer-Verlag, Berlin. doi:10.1007/978-3-540-27752-1.

Lütkepohl H (2011). “Vector Autoregressive Models.” *EUI Working Paper ECO 2011/30*, European University Institute, Florence.

Lütkepohl H, Kraetzig M (2004). *Applied Time Series Econometrics.* Cambridge University Press. doi:10.1017/CBO9780511606885.

Lütkepohl H, Milunovich G (2016). “Testing for Identification in SVAR-GARCH Models.” *Journal of Economic Dynamics and Control*, **73**(C), 241–258. doi:10.1016/j.jedc.2016.09.01.

Lütkepohl H, Netsunajev A (2017a). “Structural Vector Autoregressions with Heteroskedasticity: A Review of Different Volatility Models.” *Econometrics and Statistics*, **1**(C), 2–18. ISSN 2452-3062. doi:10.1016/j.ecosta.2016.05.001.

Lütkepohl H, Netsunajev A (2017b). “Structural Vector Autoregressions with Smooth Transition in Variances.” *Journal of Economic Dynamics and Control*, **84**(C), 43–57. ISSN 0165-1889. doi:10.1016/j.jedc.2017.09.001.

Lütkepohl H, Schlaak T (2018). “Choosing between Different Time-Varying Volatility Models for Structural Vector Autoregressive Analysis.” *Oxford Bulletin of Economics and Statistics*, **80**(4), 715–735. doi:10.1111/obes.12238.

Lütkepohl H, Schlaak T (2019). “Bootstrapping Impulse Responses of Structural Vector Autoregressive Models Identified through GARCH.” *Journal of Economic Dynamics and Control*, **101**, 41–61. ISSN 0165-1889. doi:10.1016/j.jedc.2019.01.008.

Lütkepohl H, Velinov A (2016). “Structural Vector Autoregressions: Checking Identifying Long-Run Restrictions via Heteroskedasticity.” *Journal of Economic Surveys*, **30**(2), 377–392. doi:10.1111/joes.12100.

Maddala GS (1977). *Econometrics.* 1 edition. McGraw-Hill, New York.

Mammen E (1993). “Bootstrap and Wild Bootstrap for High Dimensional Linear Models.” *The Annals of Statistics*, **21**(1), 255–285. ISSN 00905364. URL http://www.jstor.org/stable/3035590.

Matteson DS, Tsay RS (2017). “Independent Component Analysis via Distance Covariance.” *Journal of the American Statistical Association*, **112**(518), 623–637. doi:10.1080/01621459.2016.1150851.

Maxand S (2020). “Identification of Independent Structural Shocks in the Presence of Multiple Gaussian Components.” *Econometrics and Statistics*, **16**, 55–68. ISSN 2452-3062. doi:10.1016/j.ecosta.2018.10.005.

Milunovich G, Yang M (2013). “On Identifying Structural VAR Models via ARCH Effects.” *Journal of Time Series Econometrics*, **5**(2), 117–131. doi:10.1515/jtse-2013-0010.

Mullen K, Ardia D, Gil D, Windover D, Cline J (2011). "**DEoptim**: An R Package for Global Optimization by Differential Evolution." *Journal of Statistical Software*, **40**(6), 1–26. doi:10.18637/jss.v040.i06.

Nordhausen K, Oja H, Tyler DE, Virta J (2019). ***ICtest**: Estimating and Testing the Number of Interesting Components in Linear Dimension Reduction.* R package version 0.3-2, URL https://CRAN.R-project.org/package=ICtest.

Normadin M, Phaneuf L (2004). "Monetary Policy Shocks: Testing Identification Conditions under Time-Varying Conditional Volatility." *Journal of Monetary Economics*, **51**(6), 1217–1243. ISSN 0304-3932. doi:10.1016/j.jmoneco.2003.11.002.

Pfaff B (2008). "VAR, SVAR and SVEC Models: Implementation Within R Package **vars**." *Journal of Statistical Software*, **27**(4). doi:10.18637/jss.v027.i04.

R Core Team (2021). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Rigobon R (2003). "Identification through Heteroskedasticty." *Review of Economics and Statistics*, **28**(1), 777–792. ISSN 00346535, 15309142. URL http://www.jstor.org/stable/3211805.

Rigobon R, Sack B (2004). "The Impact of Monetary Policy on Asset Prices." *Journal of Monetary Economics*, **51**(8), 1553–1575. doi:10.1016/j.jmoneco.2004.02.004.

Risk BB, James NA, Matteson DS (2015). ***steadyICA**: ICA and Tests of Independence via Multivariate Distance Covariance.* R package version 1.0, URL https://CRAN.R-project.org/package=steadyICA.

Sentana E, Fiorentini G (2001). "Identification, Estimation and Testing of Conditionally Heteroskedastic Factor Models." *Journal of Econometrics*, **102**(2), 143–164. doi:10.1016/S0304-4076(01)00051-3.

Sims CA (1980). "Macroeconomics and Reality." *Econometrica*, **48**(1), 1–48. ISSN 00129682, 14680262. URL http://www.jstor.org/stable/1912017.

Stock J, Watson M (2003). "Has the Business Cycle Changed and Why?" *NBER Macroeconomics Annual 2002*, **17**, 159–230.

Székely GJ, Rizzo ML, Bakirov NK (2007). "Measuring and Testing Dependence by Correlation of Distances." *The Annals of Statistics*, **35**(6), 2769–2794. doi:10.1214/009053607000000505.

Tang Y, Horikoshi M, Li W (2016). "**ggfortify**: Unified Interface to Visualize Statistical Result of Popular R Packages." *The R Journal*, **8**(2). doi:10.32614/RJ-2016-060.

Tsay RS, Wood D (2018). ***MTS**: All-Purpose Toolkit for Analyzing Multivariate Time Series (MTS) and Estimating Multivariate Volatility Models.* R package version 1.0, URL https://CRAN.R-project.org/package=MTS.

Uhlig H (2005). "What Are the Effects of Monetary Policy on Output? Results from an Agnostic Identification Procedure." *Journal of Monetary Economics*, **52**(2), 381–419. ISSN 0304-3932. doi:10.1016/j.jmoneco.2004.05.007.

Wickham H (2016). ***ggplot2**: Elegant Graphics for Data Analysis*. Springer-Verlag, New York. ISBN 978-3-319-24277-4. URL https://ggplot2.tidyverse.org/.

Zeileis A, Leisch F, Hornik K, Kleiber C (2002). "**strucchange**: An R Package for Testing for Structural Change in Linear Regression Models." *Journal of Statistical Software*, **7**(2), 1–38. doi:10.18637/jss.v007.i02.

Affiliation:

Alexander Lange
Chair of Econometrics
Faculty of Economic Science
Universität Göttingen
Humboldtallee 3
37073 Göttingen, Germany
E-mail: alexander.lange@uni-goettingen.de

Journal of Statistical Software http://www.jstatsoft.org/
published by the Foundation for Open Access Statistics http://www.foastat.org/
March 2021, Volume 97, Issue 5 *Submitted:* 2018-06-07
doi:10.18637/jss.v097.i05 *Accepted:* 2019-11-10