

tvReg: Time-varying Coefficients in Multi-Equation Regression in R

by Isabel Casas and Rubén Fernández-Casal

Abstract This article explains the usage of R package **tvReg**, publicly available for download from the Comprehensive R Archive Network, via its application to economic and finance problems. The six basic functions in this package cover the kernel estimation of semiparametric panel data, seemingly unrelated equations, vector autoregressive, impulse response, and linear regression models whose coefficients may vary with time or any random variable. Moreover, this package provides methods for the graphical display of results, forecast, prediction, extraction of the residuals and fitted values, bandwidth selection and nonparametric estimation of the time-varying variance-covariance matrix of the error term. Applications to risk management, portfolio management, asset management and monetary policy are used as examples of these functions usage.

1 Introduction

A very popular research area has been brewing in the field of kernel smoothing statistics applied to linear models with time-varying coefficients. In econometrics, [Robinson \(1989\)](#) was the first to analyse these models for linear regressions with time-varying coefficients and stationary variables. Since then, this literature has extended to models with fewer restrictions in the dependence of the variables to models with time dependence in the error term and to multi-equation models. Although these models are potentially applicable to a large number of areas, no comprehensive computational implementation is, to our knowledge, formally available in any of the commercial programming languages. The package **tvReg** contains the aforementioned functionality, input and output interface, and user-friendly documentation.

Parametric multi-equation linear models have increased in popularity in the last decades due to an increase in access to multiple datasets. Their application extends to, perhaps, every field of quantitative research. Just to mention some, they are found in biostatistics, finance, economics, business, climate, linguistics, psychology, engineering and oceanography. Panel linear models (PLM) are widely used to account for the heterogeneity in the cross-section and time dimensions. Seemingly unrelated equations (SURE) and vector autoregressive models (VAR) are the extensions of linear regressions and autoregressive models to the multi-equation framework. Programs with these algorithms are found in all major programming languages. Particularly in R, the package **plm** ([Croissant and Millo, 2018, 2008](#)) contains a comprehensive functionality for panel data models. The package **systemfit** ([Henningesen and Hamann, 2007](#)) allows the estimation of coefficients in systems of linear regressions, both with equation error terms correlated among equations (SURE) or uncorrelated. Finally, the package **vars** ([Pfaff, 2008](#)) provides the tools to fit VAR models and impulse response functions (IRF). All these functions assume that the coefficients are constant. This assumption might not be true when a time series runs for a long period, and the relationships among variables do change. The package **tvReg** is relevant in this case.

In comparison to parametric models, the appeal of nonparametric models is their flexibility and robustness to functional form misspecification, with spline-based and kernel-based regression methods being the two main nonparametric estimation techniques, (e.g. [Eubank, 1999](#)). However, fully nonparametric models are not appropriate when many regressors are in play, as their rate of convergence decreases with the number of regressors, the infamous “curse of dimensionality”. In the case of cross-section data, a popular alternative to avoid this problem are the generalised additive models (GAM), introduced by [Hastie and Tibshirani \(1993\)](#). The GAM is a family of semiparametric models that extends parametric linear models by allowing for non-linear relationships of the explanatory variables and still retaining the additive structure of the model. In the case of time-series data, the most suitable alternative to nonparametric models is the linear models whose coefficients change over time or follow the dynamics of another random variable. This functionality is coded in R, within the single-equation framework, in packages **mgm** ([Haslbeck and Waldorp, 2020](#)), and **MARSS** ([Holmes et al., 2012](#)). Package **tvReg** uses the identical kernel smoothing estimation as package **mgm** when using a Gaussian kernel to estimate a VAR model with varying coefficients (TVVAR). However, the interpretation of their results is different because they are aimed at different audiences. The **mgm** focuses in the field of network models, producing network plots to represent relationships between current variables and their lags. Whereas the **tvReg** focuses in the field of economics where a direct interpretation of the TVVAR coefficients is not meaningful and may be done via the time-varying impulse response function (TVIRF) instead. Models with coefficients varying over time can also be

expressed in state space form, which assumes that the coefficients change over time in a determined way for example, as a Brownian motion. These models can be estimated using the Kalman filter or Bayesian techniques, for instance (Liu and Guo, 2020; Primiceri, 2005). Packages **MARSS** and **bvars** (Krueger, 2015) implement this approach based on the Carter and Kohn (1994) algorithm to estimate the TVVAR. On top of all this and as far as we can tell, the **tvReg** is the only package containing tools to estimate time-varying coefficients seemingly unrelated equation (TVSURE) and panel linear models (TVPLM) in R.

Simply, the main objective of the **tvReg** is to provide tools to estimate and forecast linear models with time-varying coefficients in the framework of kernel smoothing estimation, which may be difficult for the nonspecialised end-user to code. For completion, the **tvReg** also implements methods for the time-varying coefficients linear model (TVLM) and the time-varying coefficients autoregressive (TVAR) model. Often, these can be estimated using packages **gam** (Hastie, 2022) and **mgcv** (Wood, 2017), which combine (restricted) marginal likelihood techniques in combination with nonparametric methodologies. However, the advantage of using the **tvReg** is that it can handle dependency and any kind of distribution in the error term because it combines least squares techniques with nonparametric methodologies. An example of this is shown in Section [Standard usage of tvLM](#).

Summing up, this paper presents a review of the most common time-varying coefficient linear models studied in the econometrics literature during the last two decades, their estimation using kernel smoothing techniques, the usage of functions and methods in the package **tvReg**, and their latest applications. Along these lines, Table 1 offers a glimpse at the **tvReg** full functionality, displaying a summary of its methods, classes and functions.

Function	Class	Function and Methods for class	Based on
tvPLM	"tvPLM"	tvRE, tvFE, coef, confint, fitted, forecast, plot, predict, print, resid, summary	plm::plm
tvSURE	"tvsure"	tvGLS, bw, coef, confint, fitted, forecast, plot, predict, print, resid, summary	systemfit::systemfit
tvVAR	"tvvar"	tvAcoef, tvBcoef, tvIRF, tvOLS, tvPhi, tvPsi, bw, coef, confint, fitted, forecast, plot, predict, print, resid, summary	vars::VAR
tvIRF	"tvirf"	coef, confint, plot, print, summary	vars::irf
tvLM	"tvlm"	tvOLS, bw, coef, confint, fitted, forecast, plot, predict, print, resid, summary	stats::lm
tvAR	"tvar"	tvOLS, bw, coef, confint, fitted, forecast, plot, predict, print, resid, summary	stats::ar.ols

Table 1: Structure of the package **tvReg**.

2 Multi-equation linear models with time-varying coefficients

A multi-equation model formed by a set of linear models is defined when each equation has its own dependent variable and possible different regressors. Seemingly unrelated equations, panel data models and vector autoregressive models are included in this category.

Time-varying coefficients SURE

The SURE was proposed by Zellner (1962) and is referred to as the seemingly unrelated equations model (SURE). The SURE model is useful to exploit the correlation structure between the error terms of each equation. Suppose that there are N linear regressions of different dependent variables,

$$Y_t = X_t\beta(z_t) + U_t \quad i = 1, \dots, N \quad t = 1, \dots, T, \quad (1)$$

where $Y_t = (y_{1t} \dots y_{Nt})^\top$ with $y_i = (y_{i1}, \dots, y_{iT})^\top$ denotes the values over the recorded time period of the i -th dependent variable. Each equation in (1) may have a different number of exogenous

variables, p_i . The regressors matrix, $X_t = \text{diag}(x_{1t} \dots x_{Nt})$ with $X_i = (x_{i1}, \dots, x_{ip_i})$ for equation i and $\beta_{z_t} = (\beta_1(z_t)^\top, \dots, \beta_N(z_t)^\top)^\top$ is a vector of order $P = p_1 + p_2 + \dots + p_N$. The error vector, $U_t = (u_{1t} \dots u_{Nt})^\top$, has zero mean and covariance matrix $\mathbb{E}(U_t U_t^\top) = \Sigma_t$ with elements σ_{iit} .

It is important to differentiate between two types of smoothing variables: 1) $z_t = \tau = t/T$ is the rescaled time with $\tau \in [0, 1]$, and 2) z_t is the value at time t of the random variable $Z = \{z_t\}_{t=1}^T$. In other words, time-varying coefficients may be defined as unknown functions of time, $\beta(z_t) = f(\tau)$, or as unknown functions of a random variable, $\beta(z_t) = f(z_t)$. The estimation of the TVSURE has been studied by Henderson et al. (2015) when for a random z_t and by Orbe et al. (2005) and Casas et al. (2019) for $z_t = \tau$. These estimators are consistent and asymptotically normal under certain assumptions on the size of the bandwidth, kernel regularity and error moments, and dependency. Details are left out of this text as can be easily found in the related literature.

The estimation of system (1) may be done separately for each equation as if there is no correlation in the error term across equations, i.e. system (1) has a total of N different TVLM with possibly N different bandwidths, b_i . In this case, the time-varying coefficients are obtained by combining the ordinary least squares (OLS) and the local polynomial kernel estimator, which is extensively studied in Fan and Gijbels (1996). The result is the time-varying OLS denoted by TVOLS herein. Two versions of this estimator are implemented in **tvReg**: i) the TVOLS that uses the local constant (*lc*) kernel method, also known as the Nadaraya-Watson estimator; and ii) the TVOLS which uses the local linear (*ll*) method. Focussing in the single equation i , and assuming that $\beta_i(\cdot)$ is twice differentiable, an approximation of $\beta_i(z_t)$ around z is given by the Taylor rule, $\beta_i(z_t) \approx \beta_i(z) + \beta_i^{(1)}(z)(z_t - z)$, where $\beta_i^{(1)}(z) = d\beta_i(z)/dz$ is its first derivative. The estimates resolve the following minimisation:

$$(\hat{\beta}_i(z_t), \hat{\beta}_i^{(1)}(z_t)) = \arg \min_{\theta_0, \theta_1} \sum_{t=1}^T [y_i - X_i^\top \theta_0 - (z_t - z) X_i^\top \theta_1]^2 K_{b_i}(z_t - z).$$

Roughly, these methodologies fit a set of weighted local regressions with an optimally chosen window size. The size of these windows is given by the bandwidth b_i , and the weights are given by $K_{b_i}(z_t - z) = b_i^{-1} K(\frac{z_t - z}{b_i})$, for a kernel function $K(\cdot)$. The local linear estimator general expression is

$$\begin{pmatrix} \hat{\beta}_i(z_t) \\ \hat{\beta}_i^{(1)}(z_t) \end{pmatrix} = \begin{pmatrix} S_{T,0}(z_t) & S_{T,1}^\top(z_t) \\ S_{T,1}(z_t) & S_{T,2}(z_t) \end{pmatrix}^{-1} \begin{pmatrix} T_{T,0}(z_t) \\ T_{T,1}(z_t) \end{pmatrix} \tag{2}$$

with

$$S_{T,s}(z_t) = \frac{1}{T} \sum_{i=1}^T X_i^\top X_i (z_i - z_t)^s K\left(\frac{z_i - z_t}{b_i}\right)$$

$$T_{T,s}(z_t) = \frac{1}{T} \sum_{i=1}^T X_i^\top (z_i - z_t)^s K\left(\frac{z_i - z_t}{b_i}\right) y_i$$

and $s = 0, 1, 2$. The particular case of the local constant estimator is calculated by $\hat{\beta}_{i,t} = S_{T,0}^{-1}(z_t) T_{T,0}(z_t)$ and it is only necessary that $\beta_i(\cdot)$ has one derivative.

A second option is to use the correlation matrix of the error term in the estimation of system (1). This is called the time-varying generalised least squares (TVGLS) estimation. Its mathematical expression is the same as (2) with the following matrix components:

$$S_{T,s}(z_t) = \frac{1}{T} \sum_{i=1}^T X_i^\top K_{B,it}^{1/2} \Sigma_i^{-1} K_{B,it}^{1/2} X_i (Z_i - z_t)^s$$

$$T_{T,s}(z_t) = \frac{1}{T} \sum_{i=1}^T X_i^\top K_{B,it}^{1/2} \Sigma_i^{-1} K_{B,it}^{1/2} Y_i (Z_i - z_t)^s, \tag{3}$$

where $K_{B,it} = \text{diag}(K_{b_{1,it}}, \dots, K_{b_{N,it}})$ and $K_{b_i,it} = (Tb_i)^{-1} K((Z_i - z_t)/(Tb_i))$ is the matrix of weights introducing smoothness according to the vector of bandwidths, $B = (b_1, \dots, b_N)^\top$. Note that this minimisation problem accounts for the time-varying structure of the variance-covariance matrix of the errors, Σ_t .

The TVGLS assumes that the error variance-covariance matrix is known. In practice, this is unlikely and it must be estimated, resulting in the Feasible TVGLS estimator (TVFGLS). This estimator consists of two steps:

- Step 1 Estimate Σ_t based on the residuals of a line by line estimation (i.e, when Σ_t is the identity matrix). If Σ_t is known to be constant, the sample variance-covariance matrix from the residuals

is a consistent estimator of it. If Σ_t changes over time, a nonparametric estimator such the one explained in Section [Estimating a time-varying variance-covariance matrix](#) is a consistent alternative.

Step 2 Estimate the coefficients of the TVSURE by plugging in $\hat{\Sigma}_t$ from step 1 into Equation (3).

To ensure a good estimation of Σ_t , the iterative TVFLGS may be used. First, do steps 1-2 as above to obtain the residuals from step 2, and repeat step 2 until the estimates of Σ_t converge or the maximum number of iterations is reached.

Time-varying coefficients panel data models

Panel data linear models (PLM) are a particular case of SURE models with the same variables for each equation but measured for different cross-section units, such as countries, and for different points in time. All equations have the same coefficients apart from the intercept which can be different for different cross-sections. Therefore, the data from all cross-sections can be pooled together. The individual effects, α_i , account for the heterogeneity embedded in the cross-section dimension. This package only take into account balanced panel datasets, i.e. with the same number of data points for each cross-section unit.

Coefficient dynamics can be added to classical PLM models using a time-varying coefficients panel data model, TVPLM. Recent developments in this kind of models can be found in [Sun et al. \(2009\)](#); [Dong, C. Jiti Gao, J. and Peng, B. \(2015\)](#); [Casas et al. \(2021\)](#); [Dong et al. \(2021\)](#) among others, with general model,

$$y_{it} = \alpha_i + x_{it}^\top \beta(z_t) + u_{it} \quad i = 1, \dots, N, \quad t = 1, \dots, T. \quad (4)$$

Note that the smoothing variable only changes in the time dimension, not like in the SURE model where it changed over i and t . The three estimators of Equation (4) in the **tvReg** are:

1. The time-varying pooled ordinary least squares (TVPOLS) has the same expression than estimator (2) with the following terms:

$$\begin{aligned} S_{T,s}(z_t) &= X^\top K_{b,t}^* X (Z - z_t)^s \\ T_{T,s}(z_t) &= X^\top K_{b,t}^* Y (Z - z_t)^s, \end{aligned} \quad (5)$$

where $K_{b,t}^* = I_N \otimes \text{diag}\{K_b(z_1 - z_t), \dots, K_b(z_T - z_t)\}$. Note that it is not possible to ignore the panel structure in the semiparametric model because the coefficients change over time. The consistency and asymptotic normality of this estimator needs the classical assumptions about the kernel and the regularity of the coefficients, available in the related literature.

2. The time-varying random effects (TVRE) estimator is also given by Equation (5) with a non-identity Σ :

$$\begin{aligned} S_{T,s}(z_t) &= X^\top K_{b,t}^{*1/2} \Sigma_t^{-1} K_{b,t}^{*1/2} X (Z - z_t)^s \\ T_{T,s}(z_t) &= K_{b,t}^{*1/2} \Sigma_t^{-1} K_{b,t}^{*1/2} Y (Z - z_t)^s. \end{aligned} \quad (6)$$

Note that this is a simpler case of (3) with the same bandwidth for all equations. The variance-covariance matrix is estimated in the same way using the residuals from the TVPOLS and it may be an iterative algorithm until convergence of the coefficients.

3. The time-varying fixed effects (TVFE) estimator. Unfortunately, the transformation for the within estimation does not work in the time-varying coefficients model because the coefficients depend on time ([Sun et al., 2009](#), explain the issue in detail). Therefore, it is necessary to make the assumption that $\sum_{i=1}^N \alpha_i = 0$ for identification. The terms in the TVFE estimator are:

$$\begin{aligned} S_{T,s}(z_t) &= X^\top W_{b,t} X (Z - z_t)^s \\ T_{T,s}(z_t) &= X^\top W_{b,t} Y (Z - z_t)^s, \end{aligned} \quad (7)$$

where $W_{b,t} = D_t^\top K_{b,t}^* D_t$, $D_t = I_{NT} - D(D^\top K_{b,t}^* D)^{-1} D^\top K_{b,t}^*$, $D = (-1_{N-1}, I_{N-1})^\top \otimes 1_T$, and 1_k is the unity vector of length k . The fixed effects are given by, $\hat{\alpha} = (D^\top K_{b,t}^* D)^{-1} D^\top K_{b,t}^* (Y - X^\top \beta)$. Finally, $\hat{\alpha}_i = \frac{1}{T} \sum_{t=1}^T \alpha_{it}$ for $i = 2, \dots, N$.

Time-varying coefficient VAR model

Macroeconomic econometrics experienced a revolution when [Sims \(1980\)](#) presented the vector autoregressive (VAR) model: a new way of summarising relationships among several variables while

getting around the problem of endogeneity of structural models. The VAR model has lagged values of the dependent variable, y_t , as regressors to which further exogenous variables can be added as regressors. Unless the model is constrained, all variables are the same for every equation, which simplifies the algebra. The model coefficients and variance-covariance matrix may be estimated by maximum likelihood, OLS or GLS. VAR coefficients and the variance-covariance matrix do not have a direct economic interpretation. However, it is possible to use them to recover a structural model by imposing a number of restrictions and so analyse the transmission of a shock, for example, a new monetary policy, to the macroeconomy using the impulse response function (IRF). [Lütkepohl \(2005\)](#) dive into the theoretical properties of these models in detail.

The TVVAR(p) is an N -dimensional system of time-varying autoregressive processes of order p like

$$Y_t = A_{0,t} + A_{1,t}Y_{t-1} + \dots + A_{p,t}Y_{t-p} + U_t, \quad t = 1, 2, \dots, T. \quad (8)$$

In Equation (8), $Y_t = (y_{1t}, \dots, y_{Nt})^\top$ and coefficient matrices at each point in time $A_{j,t} = (a_{1t}^j, \dots, a_{Nt}^j)$, $j = 1, \dots, p$ are of dimension $N \times N$. Then, notation $A_{j,t}$ means that the elements of this matrix are unknown functions of either the rescaled time value, τ , or of a random variable at time t . The innovation, $U_t = (u_{1t}, \dots, u_{Nt})$, is an N -dimensional identically distributed random variable with $E(U_t) = 0$ and possibly a time-varying positive definite variance-covariance matrix, $E(U_t U_s^\top) = \Sigma_t$, for $t = s$, $E(U_t U_s^\top) = 0$ otherwise. Here, matrix $A_{j,t}$ is a function of τ , then process (8) is locally stationary in the sense of [Dahlhaus \(1997\)](#), which occurs when the functions in matrices $A_{j,t}$ are constant or change smoothly over time. Then, process (8) at time t has a well defined unique solution given by the Wold representation,

$$\bar{y}_t = \sum_{j=0}^{\infty} \Phi_{j,t} U_{t-j}, \quad (9)$$

such that $|Y_t - \bar{y}_t| \rightarrow 0$ almost surely. Matrix $\Phi_{0,t} = I_N$ and matrix $\Phi_{s,t} = \sum_{j=1}^s \Phi_{s-j,t} A_{j,t}$ for horizons $s = 1, 2, \dots$. As for the constant model, $\Phi_{s,t}$ are the time-varying coefficient matrices of the impulse response function (TVIRF). Its element (t, i, j) may be interpreted as the expected response of $y_{i,t+s}$ to an exogenous shock of $y_{j,t}$ ceteris paribus lags of y_t when the innovations are orthogonal. Otherwise, an orthogonal TVIRF can be found as $\Psi_{j,t} = \Phi_{j,t} P_t$ for $\Sigma_t = P_t P_t^\top$, the Cholesky decomposition of Σ_t at time t . More theoretical details in [Yan et al. \(2021\)](#).

In the macroeconomic literature, the Bayesian estimation of process (8) has attracted a lot of attention in recent years driven by results in [Cogley and Sargent \(2005\)](#); [Primiceri \(2005\)](#) and [Kapetanios et al. \(2012\)](#). In their approach, the coefficients are assumed to follow a random walk. Recently, [Kapetanios et al. \(2017\)](#) studied the inference of the local constant estimator of a TVVAR(p) for large sets, and they found an increase in the forecast accuracy in comparison to the forecast accuracy of the VAR(p).

3 Standard usage of tvSURE

The main argument of this function is a list of formulas, one for each equation. The formula follows the format of formula in the package `systemfit`, which implements estimators of parametric multi-equation models with constant coefficients. The tvSURE wraps the tvOLS and tvGLS methods to estimate the coefficients of system (1). The tvOLS method is used by default, calculating estimates for each equation independently with different bandwidths, `bw`. The user is able to enter a set of bandwidths or a single bandwidth to be used in the estimation instead. The tvGLS method has argument `Sigma` where a known variance-covariance matrix of the error can be entered in Equation (3). Otherwise, if `Sigma = NULL`, the variance-covariance matrix Σ_t is estimated using function `tvCov`, which is discussed in Section [Estimating a time-varying variance-covariance matrix](#).

In addition to `formula`, function tvSURE has other arguments to control and choose the desired estimation procedure:

Smoothing random variable

All methods assume by default that the coefficients are unknown functions of $\tau = t/T$ and therefore argument `z` is set to `NULL`. The user can modify this setting by entering a numeric vector in argument `z` with the values of the random smoothing variable over the corresponding time period. Note that the current version only allows one single smoothing random variable, z , common for all equations; and balanced panels.

Bandwidth

When argument `bw` is set to `NULL`, it is automatically selected by leave-one-out cross-validation. It is possible to select it by leave- k -out cross-validation ([Chu and Marron, 1991](#)) by setting argument `cv.block = k` ($k=0$ by default). This minimisation can be slow for large datasets, and

it should be avoided if the user knows an appropriate value of the bandwidth for the required problem.

Kernel type

The three choices for this argument are `tkernel = "Triweight"` (default), `tkernel = "Epa"` and `tkernel = "Gaussian"`. The first two options refer to the Triweight and Epanechnikov kernels, which are compact in $[-1, 1]$. The authors recommend the use of either of those two instead of the Gaussian kernel which, in general, requires more calculations.

Degree of local polynomial

The default estimation methodology is the Nadaraya-Watson or local constant, which is set as (`est = "lc"`) and it fits a constant at each interval defined by the bandwidth. The argument `est = "ll"` can be chosen to perform a local linear estimation (i.e., to fit a polynomial of order 1).

Singular fit

The `tvOLS` method used in the estimation wraps the `lm.wfit` method, which at default allows the fitting of a low-rank model, and the estimation coefficients can be *NAs*. The user can change the argument `singular.ok` to `FALSE`, so that the program stops in case of a low-rank model.

The user can restrict certain coefficients in the TVSURE model using arguments `R` and `r`. Note that the restriction is done by setting those coefficients to a constant. Furthermore, argument `method` defines the type of estimator to be used. The possible choices in argument `method` are:

1. "tvOLS" for a line by line estimation, i.e, with Σ the identity matrix.
2. "tvGLS" to estimate the coefficients of the system using Σ_t , for which the user must enter it in argument `Sigma`. Argument `Sigma` takes either a symmetric matrix or an array. If `Sigma` is a matrix (constant over time) then it must have dimensions $neq \times neq$, where neq is the number of equations in the system. If Σ_t changes with time, then argument `Sigma` is an array of dimension $neq \times neq \times obs$, where the last dimension measures the number of time observations. Note that if the user enters a diagonal variance-covariance matrix with diagonal values different from one, then a time-varying weighted least squares is performed. If `method = "tvGLS"` is entered but `Sigma = NULL`, then `tvSURE` is fitted as if `method = "tvOLS"` and a warning is issued.
3. "tvFGLS" to estimate the coefficients of the system using an estimate of Σ_t . By default, only one iteration is performed in the estimation, unless argument `control` indicates otherwise. The user can choose the maximum number of iterations or the level of tolerance in the estimation of Σ_t . See example the below for details.

The package `systemfit` contains the `Kmenta` dataset, which was first described in [Kmenta \(1986\)](#), to show the usage of the function `systemfit` to fit SURE models. This example has two equations: i) a demand equation, which explains how food consumption per capita, `consump`, depends on the ratio of food price, `price`; and disposable income, `income`; and ii) a supply equation, which shows how consumption depends on price, ratio prices received by farmers to general consumer prices, `farmPrice`; and a possible time trend, `trend`. Mathematically, this SURE model is

$$\begin{aligned} \text{consump}_t &= \beta_{10} + \beta_{11}\text{price}_t + \beta_{12}\text{income}_t + u_{1t} \\ \text{consump}_t &= \beta_{20} + \beta_{21}\text{price}_t + \beta_{22}\text{farmPrice}_t + \beta_{23}t + u_{2t}. \end{aligned} \quad (10)$$

The code below defines the system of equations using two formula calls which are put into a "list".

```
> data("Kmenta", package = "systemfit")
> eqDemand <- consump ~ price + income
> eqSupply <- consump ~ price + farmPrice + trend
> system <- list(demand = eqDemand, supply = eqSupply)
```

Two parametric models are fitted to the data using the function `systemfit`: one assuming that there is no correlation of the errors setting (the default), `OLS.fit` below; and another one assuming the existence of correlation in the system error term setting `method = "SUR"`, `FGLS1.fit` below. Arguing that the coefficients in (10) may change over time, the corresponding TVSUREs are fitted by using the the function `tvSURE` with the default in the argument `method` and by `method = "tvFGLS"`, respectively. They are denoted by `TVOLS.fit` and `TVFGLS1.fit`.

```
> OLS.fit <- systemfit::systemfit(system, data = Kmenta)
> FGLS1.fit <- systemfit::systemfit(system, data = Kmenta, method = "SUR")
> TVOLS.fit <- tvSURE(system, data = Kmenta)
> TVFGLS1.fit <- tvSURE(system, data = Kmenta, method = "tvFGLS")
```

In the previous chunk, the FGLS and TVFGLS estimators use only one iteration. However, the user can choose the iterative FGLS and the iterative TVFGLS models, which estimate the coefficients iteratively until convergence. The convergence level can be chosen with the argument `tol` (1e-05 by default) and the argument `maxiter` with the maximum number of iterations. The following chunk illustrates its usage:

```
> FGLS2.fit <- systemfit::systemfit(system, data = Kmenta, method = "SUR",
+                                 maxiter = 100)
> TVFGLS2.fit <- tvSURE(system, data = Kmenta, method = "tvFGLS",
+                       control = list(tol = 0.001, maxiter = 100))
```

Some of the coefficients can be restricted to have a certain constant value in `tvSURE`. This can aid statistical inference to test certain conditions. See an example of this below. Matrix `R` has as many rows as restrictions in `r` and as many columns as regressors in the model. In this case, Model (10) has 7 coefficients which are ordered as they appear in the list of formulas. Note that the time-varying coefficient of the variable `trend` is redundant when an intercept is included in the second equation of the TVSURE. Therefore, we want to restrict its coefficient to zero. For illustration, we also impose $\beta_{11,t} - \beta_{21,t} = 0.5$:

```
> Rrestr <- matrix(0, 2, 7)
> Rrestr[1, 7] <- 1; Rrestr[2, 2] <- 1; Rrestr[2, 5] <- -1
> qrestr <- c(0, 0.5)
> TVFGLS.rest <- tvSURE(system, data = Kmenta, method = "tvFGLS",
+                       R = Rrestr, r = qrestr,
+                       bw = TVFGLS1.fit$bw, bw.cov = TVFGLS1.fit$bw.cov)
```

Application to asset management

Several studies have argued that the three-factor model by [Fama and French \(1993\)](#) does not explain the whole variation in average returns. In this line, [Fama and French \(2015\)](#) added two new factors that measure the differences in profitability (robust and weak) and investment (conservative and aggressive), creating their five-factor model (FF5F). This model has been applied in [Fama and French \(2017\)](#) to analyse the international markets. A time-varying coefficients version of the FF5F has been studied in [Casas et al. \(2019\)](#), whose dataset is included in the `tvReg` under the name of FF5F. The TVFF5F model is

$$R_{it} - RF_{it} = a_{it} + b_{it} (RM_{it} - RF_{it}) + s_{it} SMB_{it} + h_{it} HML_{it} + r_{it} RMW_{it} + c_{it} CMA_{it} + u_{it}, \quad (11)$$

where R_{it} refers to the price return of the asset of certain portfolio for market i at time t , RF_t is the risk free return rate, and RM_t represents the total market portfolio return. Therefore, $R_{it} - RF_{it}$ is the expected excess return and $RM_{it} - RF_{it}$ is the excess return on the market portfolio. The other factors, SMB_t stands for "small minus big" and represents the size premium, HML_t stands for "high minus low" and represents the value premium, RMW_t is a profitability factor, and CMA_t accounts for the investment capabilities of the company. Finally, the error term structure is

$$E(u_{it}u_{js}) = \begin{cases} \sigma_{it} = \sigma_{it}^2 & i = j, \quad t = s \\ \sigma_{ijt} & i \neq j, \quad t = s \\ 0 & t \neq s. \end{cases}$$

The FF5F dataset has been downloaded from the Kenneth R. [French \(2016\)](#) data library. It contains the five factors from four different international markets: North America (NA), Japan (JP), Europe (EU), and Asia Pacific (AP). For the dependent variable, the excess returns of portfolios formed on size and book-to-market have been selected. The period runs from July 1990 to August 2016 and it has a monthly frequency. The data contains the Small/Low, Small/High, Big/Low and Big/High portfolios. The factors in the TVFF5F model explain the variation in returns well if the intercept is statistically zero. The lines of code below illustrate how to fit a TVSURE to the Small/Low portfolio.

```
> data("FF5F")
> eqNA <- NA.SMALL.LoBM - NA.RF ~ NA.Mkt.RF + NA.SMB + NA.HML + NA.RMW + NA.CMA
> eqJP <- JP.SMALL.LoBM - JP.RF ~ JP.Mkt.RF + JP.SMB + JP.HML + JP.RMW + JP.CMA
> eqAP <- AP.SMALL.LoBM - AP.RF ~ AP.Mkt.RF + AP.SMB + AP.HML + AP.RMW + AP.CMA
> eqEU <- EU.SMALL.LoBM - EU.RF ~ EU.Mkt.RF + EU.SMB + EU.HML + EU.RMW + EU.CMA
> system2 <- list(NorthA = eqNA, JP = eqJP, AP = eqAP, EU = eqEU)
```

```
> TVFF5F <- tvSURE(system2, data = FF5F, method = "tvFGLS",
+                 bw = c(0.56, 0.27, 0.43, 0.18), bw.cov = 0.12)
```

The package **tvReg** also includes the functionality to compute confidence intervals for the coefficients of class attributes "tv1m", "tvar", "tvplm", "tvsure" and "tvirf" by extending the `confint` method. The algorithm in [Fan and Zhang \(2000\)](#) and [Chen et al. \(2017\)](#) to calculate bootstrap confidence intervals has been adapted for all these class attributes. Argument `level` is set to 0.95 (95% confidence interval) by default. Argument `runs` (100 by default) is the number of resamples used in the bootstrapping calculation. Note that the calculation using `runs = 100` can take long, so we suggest to try a small value in `runs` first to get an initial intuition of the results. Because coefficients are time-varying, only wild bootstrap residual resampling is implemented. Two choices of wildbootstrap are allowed in argument `tboot`: the default one proposed in [Mammen \(1993\)](#) (`tboot = "wild"`); and the standard normal (`tboot = "wild2"`).

In the backend code, coefficient estimates from all replications are stored in the `BOOT` variable. In this way, calculations are not done again if the user chooses a different level for the same object. In the chunk below, the `confint` method calculates the 90% confidence interval of the object `TVFF5F`. Posteriorly, the 95% interval is calculated quickly because the resample calculations in the first interval are re-used for the second. Thus, the 90% confidence interval calculation takes around 318 seconds with a 2.2 GHz Intel Core i7 processor and the posterior 95% confidence interval takes only around 0.7 seconds.

```
> TVFF5F.90 <- confint(TVFF5F, level = 0.90)
> TVFF5F.95 <- confint(TVFF5F.90)
```

The `plot` method is implemented for each of the six class attributes in **tvReg**. For example, the 95% confidence intervals of the intercept for the North American, Japanese, Asia Pacific and European markets [Figure 1](#) are with `plot` statement below, that produces four independent plots of the first variable (the intercept in this case) in each equation due to argument `vars = 1`.

```
> plot(TVFF5F.95, vars = 1)
```

The user can also choose to plot the coefficients of several variables and/or equations. Plots will be grouped by equation, with a maximum of three variables per plot. The piece of code below show how to plot the coefficients of the second and third variables from the Japan market equation, which results can be seen in [Figure 2](#).

```
> plot(TVFF5F.95, vars = c(2, 3), eqs = 2)
```

4 Standard usage of tvPLM

The `tvPLM` method is inspired by the `plm` method from the package **plm**. It converts data into an object of the class attribute "pdata.frame" using argument `index` to define the cross-section and time dimensions. If `index = NULL` (default), the two first columns of data define the dimensions. The `tvPLM` wraps the `tvRE` and `tvFE` methods to estimate the coefficients of time-varying panel data models.

The user can provide additional optional arguments to modify the default estimation. See [section 2.3](#) for details on arguments `z`, `bw`, `est` and `tkernel`. Furthermore, argument `method` defines the estimator used. The possible choices based on package **plm** choices are: "pooling" (default), "random" and "within".

Application to health policy

The income elasticity of healthcare expenditure is defined as the percentage change in healthcare expenditure in response to the percentage change in income per capita. If this elasticity is greater than one, then healthcare expenditure grows faster than income, as luxury goods do, and is driven by market forces alone. The heterogeneity of health systems among countries and time periods have motivated the use of panel data models, for example in [Gerdtham et al. \(1992\)](#) who use a FE model. Recently, [Casas et al. \(2021\)](#) have investigated the problem from the time-varying panel models perspective using the TVFE estimation. In addition to the income per capita, measured by the log GDP, the authors use the proportion of population over 65 years old, the proportion of population under 15 years old and the share of public funding of healthcare. The income elasticity estimate with a FE implemented in the `plm` is greater than 1, a counter-intuitive result. This issue is resolved using the TVFE implemented in the **tvReg**. The code below estimates coefficients with the parametric and semiparametric models:

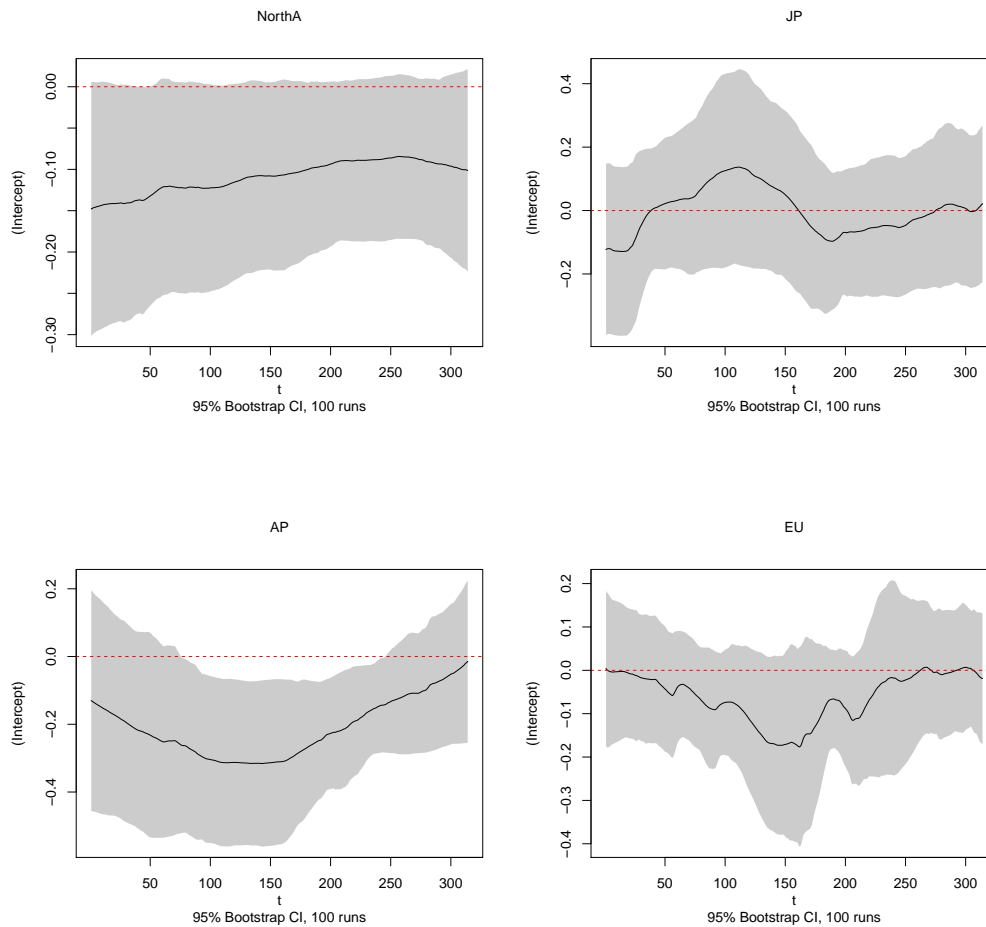


Figure 1: Intercept estimates of a Small/Low portfolio in the four markets (left to right, top to bottom: North America, Japan, Asia Pacific and Europe). The solid lines indicate the estimates, the grey bands are their 95% bootstrap confidence intervals and the red dashed lines indicate zero. Only the Asia Pacific market intercepts are statistically different from zero during a large period, implying that the FF5F does not explain excess returns well for the Asia Pacific market.

```

> data("OECD")
> elast.fe <- plm(lhe ~ lgdp + pop65 + pop14 + public, data = OECD,
+               index = c("country", "year"), model = "within")
> elast.tvfe <- tvPLM(lhe ~ lgdp + pop65 + pop14 + public, data = OECD,
+                   index = c("country", "year"), method = "within",
+                   bw = 0.67)
> elast.fe <- confint(elast.fe)
> elast.tvfe <- confint(elast.tvfe)

```

Figure 3 shows the elasticity estimates using the FE and TVFE estimators. The constant coefficients model (dashed line) suggests that healthcare is a luxury good (over 1), while the time-varying coefficients (solid line) model suggests it is a value under 0.8.

5 Standard usage of tvVAR and tvIRF

A TVVAR(p) model is a system of time-varying autoregressive equations of order p . The dependent variable, y , is of the class attribute "matrix" or "data.frame" with as many columns as equations. Regressors are the same for all equations and they contain an intercept if the argument type = "const" (default) or not if type = "none"; lagged values of y ; and other exogenous variables in exogen. Econometrically, the tvOLS method is called to calculate the estimates for each equation independently using one bandwidth per equation. The user can choose between automatic bandwidth selection; or

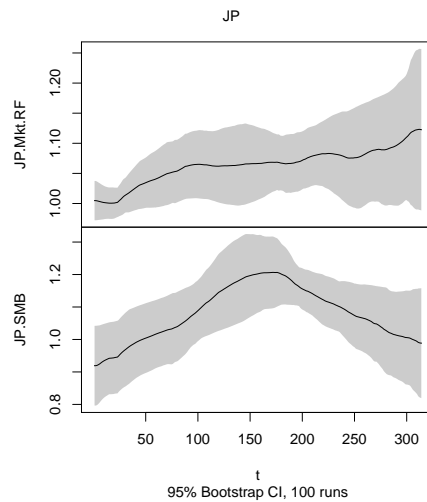


Figure 2: Coefficient estimates of excess returns on the market portfolio (JP.Mkt.RF) and JP.SMB factors for a Small/Low portfolio in the Japan market. The solid line indicates the estimates and the grey bands are their 95% bootstrap confidence intervals. It seems that the effect of the market return over the asset return increases slightly over time, while the effect of the size premium over the asset return has an inverted U shape over the time period.

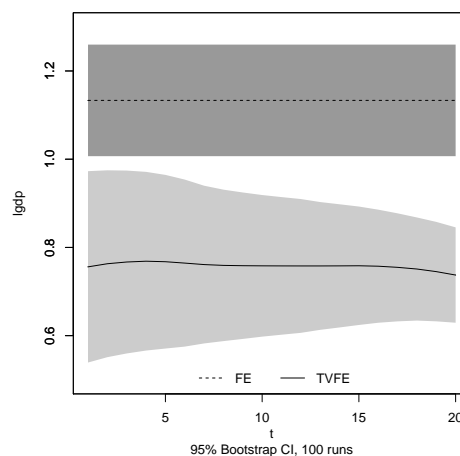


Figure 3: Comparison of income elasticity of healthcare expenditure in OECD countries. The dashed line with a dark grey band corresponds to the FE estimate and its 95% bootstrap confidence interval, while the solid line with a light band corresponds to the TVFE estimates and their 95% confidence intervals. There is a clear difference in the elasticity estimates of the two models.

entering a one value in `bw`, meaning that all equations will be estimated with the same bandwidth; or a vector of bandwidths, one for each equation. The `tvVAR` returns a `list` of the class attribute `tvvar`, which can be used to estimate the TVIRF model with the function `tvIRF`.

Application to monetary policy

The assessment and forecast of the effects of monetary policy on macroeconomic variables, such as inflation, economic output and employment is commonly modelled using the econometric framework of VAR and interpreted by the IRF. In recent years, scholars of macroeconometrics have searched intensely for a way to include time variation in the coefficients and covariance matrix of the VAR model. The reason for this is that the macroeconomic climate evolves over time and effects of monetary policy must be identified locally rather than globally. In the Bayesian framework, [Primiceri \(2005\)](#) used the [Carter and Kohn \(1994\)](#) algorithm to fit the TVP-VAR to this monetary policy problem. Results of the latter can be replicated with the functions in the package `bvars` and compared with results in the

tvReg that fits the following TVVAR(4):

$$\begin{aligned} \text{inf}_t &= a_t^1 + \sum_{i=1}^4 b_{it}^1 \text{inf}_{t-i} + \sum_{i=1}^4 c_{it}^1 \text{une}_{t-i} + \sum_{i=1}^4 d_{it}^1 \text{tbi}_{t-i} + u_t^1 \\ \text{une}_t &= a_t^2 + \sum_{i=1}^4 b_{it}^2 \text{inf}_{t-i} + \sum_{i=1}^4 c_{it}^2 \text{une}_{t-i} + \sum_{i=1}^4 d_{it}^2 \text{tbi}_{t-i} + u_t^2 \\ \text{tbi}_t &= a_t^3 + \sum_{i=1}^4 b_{it}^3 \text{inf}_{t-i} + \sum_{i=1}^4 c_{it}^3 \text{une}_{t-i} + \sum_{i=1}^4 d_{it}^3 \text{tbi}_{t-i} + u_t^3. \end{aligned}$$

Central banks commonly regulate the money supply by changing the interest rates to keep a stable inflation growth. The R code below uses macroeconomic data from the United States, exactly the one used in [Primiceri \(2005\)](#), with the following three variables: inflation rate (*inf*), unemployment rate (*une*) and the three months treasury bill interest rate (*tbi*). For illustration, a VAR(4) model is estimated using the function `VAR` from the package `vars`, a TVVAR(4) model is estimated using the function `tvVAR` from the package `tvReg` and a TVP-VAR(4) model is estimated using the function `bvar.sv.tvp` from the package `bvars`. Furthermore, their corresponding impulse response functions with horizon 20 are calculated to forecast how the inflation responds to a positive shock in interest rates. The TVVAR(4) can also be estimated with function `tvmlvar` from R package `mgm`, which will give the same coefficient estimates than the `tvVAR` for the Gaussian kernel and same bandwidth. However, package `mgm` does not have an impulse response function and, for this reason, it is left out of the example.

```
> data(usmacro, package = "bvars")
> VAR.usmacro <- vars::VAR(usmacro, p = 4, type = "const")
> TVVAR.usmacro <- tvVAR(usmacro, p = 4, bw = c(1.14, 20, 20), type = "const")
> TVPVAR.usmacro <- bvars::bvar.sv.tvp(usmacro, p = 4, pdrift = TRUE, nrep = 1000,
+                                     nburn = 1000, save.parameters = TRUE)
```

The user can provide additional optional arguments to modify the default estimation. See Section [Standard usage of tvSURE](#) to understand the usage of arguments `bw`, `tkernel`, `est` and `singular.ok`. In addition, the function `tvVAR` has the following arguments:

Number of lags

The number of lags is given by the model order set in the argument `p`.

Exogenous variables

Other exogenous variables can be included in the model using the argument `exogen`, which accepts a vector or a matrix with the same number of rows as the argument `y`.

Type

The default model contains an intercept (i.e., it has a mean different from zero). The user can set argument `type = "none"`, so the model has mean zero.

The variance-covariance matrix from the residuals of a TVVAR(p) can be used to calculate the orthogonal TVIRF. The `plot` method for object of class attribute `"tvvar"` displays as many plots as equations, each plot with the fitted and residuals values as it is shown in [Figure 4](#) obtained with:

```
> plot(TVVAR.usmacro)
```

[Figure 4](#) shows the residuals of the inflation equation that has a mean close to zero and the fitted values are fitting the observed values closely.

Function `tvIRF` estimates the TVIRF with main argument, `x`, which is an object of class attribute `"tvvar"` returned by the function `tvVAR`. The user can provide additional optional arguments to modify the default estimation as explained below.

Impulse and response variables

The user has the option to pick a subset of impulse variables and/or response variables using arguments `impulse` and `response`.

Horizon

The horizon of the TVIRF coefficients can be chosen by the user with argument `n.ahead`, the default is 10.

Orthogonal TVIRF

The orthogonalised impulse response function is computed by default (or `tho = TRUE`). In the orthogonal case, the estimation of the variance-covariance matrix of the errors is estimated as

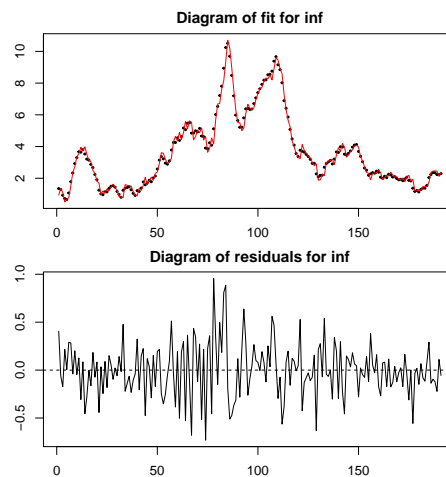


Figure 4: Returns and fitted values of object TVVAR.usmacro for the inflation equation. The dots in the top plot represent the observed values and the red line represents the fitted values, while the black line in the bottom plot represents the returns of the estimation. The model fits the observed values well and the returns appear to have zero mean and constant variance.

time-varying (ortho.cov = "tv") by default (see Section [Estimating a time-varying variance-covariance matrix](#) for theoretical details). Note that the user can enter a value of the bandwidth for the variance-covariance matrix estimation in bw.cov. It is possible to use a constant variance-covariance matrix by setting ortho.cov = "const".

Cumulative TVIRF

If the user desires to obtain the cumulative TVIRF values, then argument cumulative must be set to TRUE.

Following the previous example, the lines of code below estimate the IRF using the package **vars**, the TVP-IRF using the package **bvars** and the TVIRF using the package **tvReg**.

```
> IRF.usmacro <- vars::irf(VAR.usmacro, impulse = "tbi", response = "inf", n.ahead = 20)
> TVIRF.usmacro <- tvIRF(TVVAR.usmacro, impulse = "tbi", response = "inf", n.ahead = 20)
> TVPIRF.usmacro <- bvars::impulse.responses(TVPVAR.usmacro, impulse.variable = 3,
+                                           response.variable = 1, draw.plot = FALSE)
```

A comparison of impulse response functions from the three estimations is plotted in Figure 5, whose R code is shown below:

```
> irf1 <- IRF.usmacro$irf[["tbi"]]
> irf2 <- TVIRF.usmacro$irf[["tbi"]]
> irf3 <- TVPIRF.usmacro$irf
> ylim <- range(irf1, irf2[150,], irf3[50,])
> plot(1:20, irf1[-1], ylim = ylim, main = "Impulse variable: tbi from 1990Q2",
+      xlab = "horizon", ylab = "inf", type = "l", lwd = 2)
> lines(1:20, irf2[150,-1], lty = 2, lwd = 2)
> lines(1:20, irf3[50,], lty = 3, lwd = 2)
```

Figure 5 displays the IRF, the TVIRF and the TVP-IRF (the two latter at time 150 in our database, which corresponds to the second quarter of 1990) for horizons 1 to 20. The IRF and TVIRF follow a similar pattern: a positive shock of one unit in the short-term interest rates (tbi) during 1990Q2 results in an initial drop in inflation during the first three months, followed by an increase for two or three months and finally in a steady decrease until it plateaus one year after. The left plot shows an increase in inflation during the first three months and a drop after.

The confint method is also implemented for the class attribute "tvirf". Remember that the TVIRF model contains one impulse response function for each data time record. So, the full plot of TVIRF would have as many lines as the number of rows in the dataset. Instead, the plot method displays only one line by default, the mean value of all those impulse response functions and it issues a warning. The user can enter one or several values into argument obs.index to plot the IRF at the desire point(s) in time.

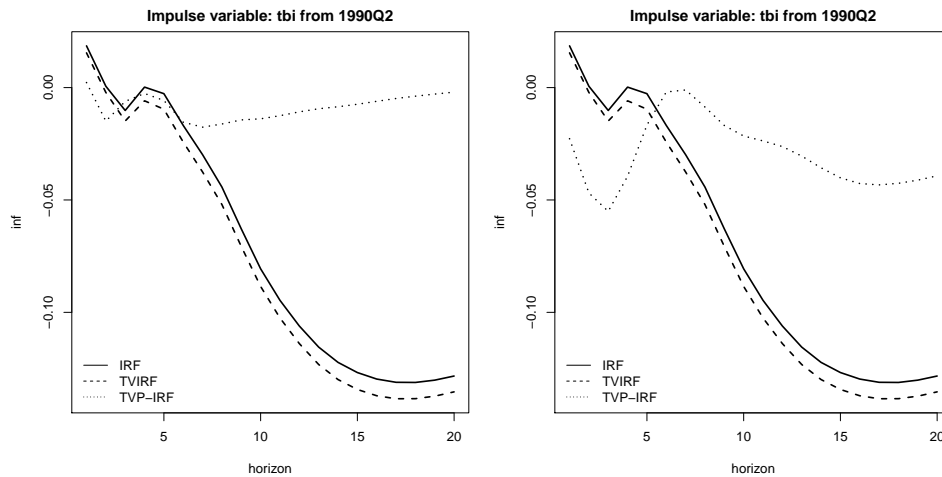


Figure 5: Estimated response of inflation (inf) to an increase in interest rates (tbi) of one unit during 1990Q2. The dashed line correspond to the IRF estimates, the solid line to the TVIRF and the dotted line to the Bayesian estimates. It appears that the Bayesian estimates are very different from those of the other two models.

6 Estimating a time-varying variance-covariance matrix

The time-varying variance-covariance matrix of two or more series is estimated nonparametrically in **tvReg**. Given a random process $y_i = (y_{i1}, \dots, y_{iT})^\top$, such that $E(y_{it}) = 0$ and $E(y_{it}y_{i't'}) = \sigma_{ii't}$ if $t = t'$ and zero otherwise. Thus, the variance-covariance matrix for time t is denoted by Σ_t with elements $\sigma_{ii't}$ with $1 \leq i, i' \leq N$. Given that Σ_t is locally stationary, its local linear estimator is defined by

$$\text{vech}(\hat{\Sigma}_\tau) = \sum_{t=1}^T \text{vech}(y_t^\top y_t) K_b(t - \tau) \frac{s_2 - s_1(\tau - t)}{s_0 s_2 - s_1^2} \quad (12)$$

where $s_j = \sum_{t=1}^T (\tau - t)^j K_b(\tau - t)$ for $j = 0, 1, 2$. As shown previously, $K_b(\cdot)$ is a symmetric kernel function heavily concentrated around the origin, $\tau = t/T$ is the focal point and b is the bandwidth parameter. Note that a single bandwidth is used for all co-movements, which ensures that $\hat{\Sigma}_\tau$ is positive definite.

The user must be aware that the local linear estimator can return non-positive definite matrices for small samples. Although the local constant estimator, calculated when $s_1 = s_2 = 1$ in (12), does not have as good asymptotic properties in the boundaries as the local linear estimator, it always provides positive definite matrices, which is a desirable property of an estimator of a variance-covariance matrix. Therefore, it is the default estimator in the function `tvCov`.

The function `tvCov` is called by the function `tvIRF` to calculate the orthogonal TVIRF, and by the function `tvSURE` for `method = "tvFGLS"` to estimate the variance-covariance matrix of the error term. The function `tvCov` can generally be used to estimate the time-varying covariance matrix of any two or more series.

Application to portfolio management

Aslanidis and Casas (2013) consider a portfolio of daily US dollar exchange rates of the Australian dollar (AUS), Swiss franc (CHF), euro (EUR), British pound (GBP), South African rand (RAND), Brazilian real (REALB) and Japanese yen (YEN), over the period from January 6, 1999 until May 7, 2010 ($T = 2855$ observations). This dataset contains the standardised rates after “devolatilisation”; i.e., after standardising the rates using the GARCH(1,1) estimates of the volatility and it is available in the **tvReg** under the names of CEES. A portfolio consisting of these currencies is well diversified containing some safe haven currencies, active and liquid currencies and currencies that perform well in times of high interest rates. The estimation of the correlation matrix among these currencies is essential for portfolio management. The model is

$$\begin{aligned} r_{p,t} &= \omega_t^\top r_t \\ h_{p,t} &= \omega_t^\top H_t \omega_t \end{aligned}$$

where $r_{p,t}$ and $h_{p,t}$ are the return and variance of the portfolio at time t . Variable ω_t is a vector with the weight of each currency in the portfolio strategy at time t . The portfolio variance-covariance matrix is denoted by H_t , and it may vary with time for a dynamic investment strategy. This matrix can be estimated using the function `tvCov` and then used in risk management, for example to calculate the Value-at-risk, denoted by VaR in the financial literature. The VaR, not be confused with the VAR, measures the level of financial risk of a portfolio, asset or firm. The VaR of an asset X , with distribution function F_X , at the confidence level α is defined as $\text{VaR}_\alpha = \inf\{x : F_X(x) > \alpha\}$. Commonly, the distribution function of X is assumed to be Gaussian with unknown variance. In a portfolio framework, the variance-covariance matrix is estimated to calculate the VaR of a portfolio together with the portfolio weights (`omega` in the code below). The portfolio weights are the percentage of the total portfolio investment in each asset and can be chosen to be constant or changing over time. In the code below, weights are calculated by minimum variance at each point in time. The estimated VaR of this example portfolio is shown in Figure 6.

```
> data(CEES)
> VaR <- numeric(nrow(CEES))
> Ht <- tvCov(CEES[, -1], bw = 0.12)
> e <- rep(1, ncol(CEES)-1)
> for (t in 1:nrow(CEES)){
+   omega <- solve(Ht[, , t])%*%e/((t(e)%*%solve(Ht[, , t])%*%e)[1])
+   VaR[t] <- abs(qnorm(0.05))*sqrt(max(t(omega)%*%Ht[, , t]%*%omega, 0))
+ }
> plot(as.Date(CEES[, "Date"]), VaR, type = "l", xlab = "year",
+       ylab = expression(VaR[t]), main="VaR of CEES over time")
```

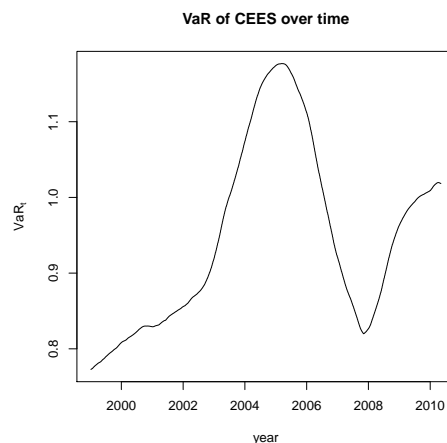


Figure 6: Dynamics of the Value-at-risk of the CEES exchange rates portfolio over time. The solid line represents the VaR. It appears that the risk of potential financial losses of this portfolio increased up to year 2005, decreasing then until 2008 and turn up again afterwards.

7 Single-equation linear models with time-varying coefficients

A varying coefficients linear model (TVLM) is generally expressed by

$$y_t = x_t^\top \beta(z_t) + u_t, \quad t = 1, \dots, T, \quad (13)$$

where y_t is the response or dependent variable, $x_t = (x_{1t}, x_{2t}, \dots, x_{dt})^\top$ is a vector of regressors at time t , $\beta(z_t)$ is the vector of coefficients at time t and u_t is the error term which satisfies $E(u_t|x_t) = 0$ and $E(u_t^2|x_t) = \sigma^2$. There are not enough degrees of freedom in the TVLM for a meaningful OLS estimation, but it may be estimated with the TVOLS displayed in Equation 2. The particular case of $x_t = (y_{t-1}, y_{t-2}, \dots, y_{t-p})$ corresponds to the time-varying autoregressive model, TVAR(p), which is also estimated with the TVOLS.

The case of $z_t = \tau = t/T$ was firstly studied in Robinson (1989) for stationary processes and generalised to nonstationary processes and correlated errors by Chang and Martinez-Chombo (2003) and Cai (2007) among others. Recently, Chen et al. (2017) apply it to the Heterogeneous Autoregressive (HAR) model of Corsi (2009) for the realized volatility of S&P 500 index returns. It is a very

flexible approach, but forecasts are not consistent because there is no information from the dependent variable at time $T + 1$. On the other hand, the case of a random z_t has been studied for iid or stationary processes by [Hastie and Tibshirani \(1993\)](#) and [Cai et al. \(2000\)](#); and nonstationary regressors or/and nonstationary z_t have been studied by [Chang and Martinez-Chombo \(2003\)](#), [Cai et al. \(2009\)](#), [Zhang and Wu \(2012\)](#), [Sun et al. \(2013\)](#) and [Gao and Phillips \(2013\)](#). [Das \(2005\)](#) and [Xiao \(2009\)](#) have used the approach for instrumental variables and cointegration. In summary, this estimator is consistent and asymptotically normal for several types of dependency of $\{(x_t, z_t, u_t)\}$.

8 Standard usage of tvLM

The function `tvLM` fits a TVLM using the `tvOLS` method. The `tvLM` follows the standards of the function `lm` with main arguments `formula` and `data`. The only mandatory argument is `formula`, which should be a single formula for a single-equation model. This arguments follows the standard regression formula in R. The function `tvLM` returns an object of the class attribute `tvlm`. This model is in some cases a GAM-type model which is implemented in the comprehensive and well-established `mgcv` package. The `mgcv` uses a methodology different from kernel smoothing to estimate the varying coefficients, involving splines and quasi-maximum likelihood estimation. The advantage of using kernel smoothing techniques to estimate the TVLM is that it can handle dependency and any kind of distribution in the error term. For illustration of this difference between the two packages in relation to the TVLM, the following model is generated:

$$y_t = \beta_{1t}x_{1t} + \beta_{2t}x_{2t} + u_t, \quad t = 1, \dots, T, \quad (14)$$

where $\beta_{1t} = \sin(2\pi\tau)$ and $\beta_{2t} = 2\tau$ with $\tau = t/T$ and $T = 1000$. The regressors, $x_{1t} \sim t_2$ (symmetric) and $x_{2t} \sim \chi_4^2$, are independent of the error term, $u_t \sim \chi_2^2$ which has an exponential dependency in the covariance matrix given by $Cov(u_t, u_{t+h}) = e^{-|h|/10}$ and does not follow an exponential-family distribution. The LM, TVLM and GAM models are fitted to the data. The process generation and the fitting of a classical LM, a TVLM and a GAM are shown in the following chunk. [Figure 7](#) compares the different estimates with the true β_{1t}, β_{2t} . As expected, the estimates from `lm` are constant and lie around the average of all β_{1t} and β_{2t} , while the estimates of `tvLM` and `gam` follow the dynamics of the varying coefficients. Besides the estimates of `gam` fit β_{1t} well, but not β_{2t} although the latter is a simple linear function. This issue is caused by the autocorrelated error term with a non-exponential distribution. On the other hand, the `tvLM`, although it requires for a longer computation time, it is able to fit both coefficients well.

```
> tau <- seq(1:1000)/1000
> d <- data.frame(tau, beta1 = sin(2 * pi * tau), beta2 = 2 * tau,
+               x1 = rt(1000, df = 2), x2 = rchisq(1000, df = 4))
> error.cov <- exp(-as.matrix(dist(tau))/10)
> error <- t(chol(error.cov)) %%% rchisq(N, df = 2)
> d <- transform(d, y = x1 * beta1 + x2 * beta2 + error)
> lm1 <- stats::lm(y ~ x1 + x2, data = d)
> TVLM1 <- tvLM(y ~ x1 + x2, data = d, bw = 0.05, est = "ll")
> gam1 <- mgcv::gam(y ~ s(tau, by = x1) + s(tau, by = x2), data = d)
```

In addition to `formula`, the function `tvLM` has the arguments described in [Section Standard usage of tvSURE](#) above. Also methods `confint`, `fitted`, `print`, `plot`, `residuals` and `summary` are implemented for class `"tvlm"`.

The `summary` method displays: (i) a summary of all coefficient values over the whole time period, (ii) the value of the bandwidth(s), and (iii) a measure of goodness-of-fit, pseudo- R^2 . The latter is printed for the class attributes `"tvsure"`, `"tvplm"`, `"tvvar"`, `"tvlm"` and `"tvar"` and it is calculated with the classical equation,

$$R^2 = 1 - \frac{\sum_{t=1}^T (y_t - \hat{y}_t)^2}{\sum_{t=1}^T (y_t - \bar{y})^2}$$

where y_t is the dependent variable, \bar{y} is its mean and \hat{y}_t are the fitted values. For multiple equation models, one pseudo- R^2 is calculated for each equation.

9 Standard usage of tvAR

A TVAR model is a particular case of TVLM whose regressors contain lagged values of the dependent variable, y . The number of lags is given by the model order set in the argument `p`. Other exogenous

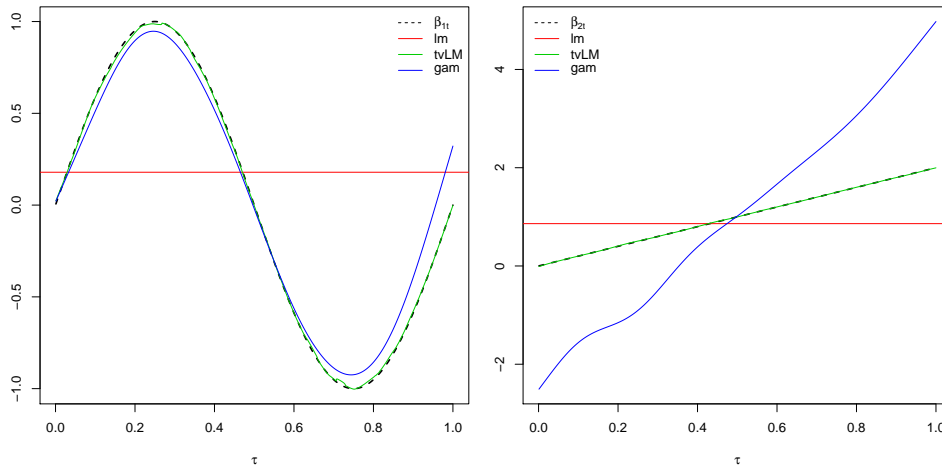


Figure 7: Comparison of the `lm`, `tvLM` and `gam` estimates of β_{1t} and β_{2t} . The true values are plotted in black, the red lines represent the `lm` estimates, the green lines refer to the `tvLM` estimates and the blue lines represent the `gam` estimates. This result suggests that the TVLM is preferable for modelling non-linear varying coefficients under strong dependency.

variables can be included in the model using the argument `exogen`, which accepts a vector or a matrix with the same number of rows as the argument `y`. An intercept is included by default unless the user enters `type = "none"` into the function call. Econometrically, this function also wraps the `tvOLS` estimator, which needs a bandwidth `bw` that is automatically selected when the user does not enter any number. An object of the class attribute `"tvar"` is returned by the function `tvAR`.

The user can provide additional optional arguments to modify the default estimation of the function `tvAR`. See Section [Standard usage of tvSURE](#) to understand the usage of arguments `bw`, `tkernel`, `est` and `singular.ok` and Section [Standard usage of tvVAR and tvIRF](#) to understand the usage of argument `type`. In addition, the function `tvAR` has the following argument:

Coefficient restrictions

An autoregressive process of order p does not necessarily contain all the previous p lags of y_t . Argument `fixed`, with the same format as in the function `arima` from the package `stats`, permits to impose these restrictions. The order of variables in the model is: intercept (if any), lag 1, lag 2, ..., lag p and exogenous variable (if any). By default, the argument `fixed` is a vector of `NA`s with length the number of coefficients in the model. The user can enter a vector in the argument `fixed` with zeros in the positions corresponding to the restricted coefficients.

Application to risk management

The realized variance (RV) model was popularised in the financial literature by [Andersen and Bollerslev \(1998\)](#), who show that the use of intraday data can offer an accurate forecast of daily variance. It is defined as $RV_t = \sum_{i=1}^N r_{it}^2$, where r_{it} is the price return at minute i of day t . The autocorrelation function of the RV also shows signs of long memory in the process, which can be accounted for by the heterogeneous RV (HAR) model of [Corsi \(2009\)](#):

$$RV_t = \beta_0 + \beta_1 RV_{t-1} + \beta_2 RV_{t-1|t-5} + \beta_3 RV_{t-1|t-22} + u_t. \tag{15}$$

Here, $RV_{t-1|t-k} = \frac{1}{k} \sum_{i=1}^k RV_{t-i}$. In this model, the current RV_t depends on its immediately previous value, RV_{t-1} , its medium-term memory factor, $RV_{t-1|t-5}$ and its long-term memory factor, $RV_{t-1|t-22}$. Basically, the HAR model may be seen as an AR(1) model with two exogenous variables.

It is likely that changes in the business cycles affect the coefficients in (15). [Chen et al. \(2017\)](#) coined the time-varying coefficient HAR, whose coefficients are functions of the rescaled time period. The RV dataset contains daily variables running from January 3, 1990 until December 19, 2007 that have been computed from 5 minute intraday data from [Store \(2017\)](#). This period coincides with the period in [Bollerslev et al. \(2009\)](#). The variable names in this dataset are `RV`, `RV_lag`, `RV_week`, `RV_month` and `RQ_lag_sqrt` and correspond to the RV_t , RV_{t-1} , $RV_{t-1|t-5}$, $RV_{t-1|t-22}$ and $RQ_{t-1}^{1/2}$ in Model (15).

```
> data("RV")
> RV2 <- head(RV, 2000)
```



```
> HAR <- with(RV2, arima(RV, order = c(1, 0, 0), xreg = cbind(RV_week, RV_month)))
> TVHAR<- with(RV2, tvAR(RV, p = 1, bw = 0.8, exogen = cbind(RV_week, RV_month)))
```

Bollerslev et al. (2016) extended the Model (15) to control for the effect of the realized quarticity (RQ) on the relationship between the future RV and its near past values. They present the HARQ model,

$$RV_t = \beta_0 + (\beta_1 + \beta_{1Q}RQ_{t-1}^{1/2}) RV_{t-1} + \beta_2 RV_{t-1|t-5} + \beta_3 RV_{t-1|t-22} + u_t. \tag{16}$$

The HARQ model is a HAR model whose RV_{t-1} term's coefficient is a linear function of the squared root of RQ at time $t - 1$. The RQ changes over time and it will be larger during periods of more uncertainty. Casas et al. (2018) appreciated that the variation of this coefficient may not be linear and proposed the TVHARQ model,

$$RV_t = \beta_0(z_t) + \beta_1(z_t) RV_{t-1} + \beta_2(z_t) RV_{t-1|t-5} + \beta_3(z_t) RV_{t-1|t-22} + u_t, \tag{17}$$

where the smoothing variable, $z_t = RQ_{t-1}^{1/2}$. This model is a TVAR(1) process and can be estimated with the function tvAR or with the function tvLM as it is shown in the chunk below.

```
> HARQ <- with(RV2, lm(RV ~ RV_lag + I(RV_lag*RQ_lag_sqrt) + RV_week + RV_month))
> TVHARQ <- with(RV2, tvAR(RV, p = 1, exogen = cbind(RV_week, RV_month),
+ z = RQ_lag_sqrt, cv.block = 10))
```

10 Prediction and forecast in time-varying coefficient models

Estimation is a useful tool to understand the patterns and processes hidden in known data. Prediction and forecast are the mechanisms to extend this understanding to unknown data. Although the two terms are often used indistinctively, the term prediction is broader than the term forecast which is reserved for time-series models and consists on using historical data to infer the future. For example, we speak of predicting values from a linear regression fitted to cross-sectional data and of forecasting future values from an AR(p) model.

The *prediction* of the dependent variable at time $T + h$ (horizon of length h) in a linear regression is $\hat{y}_{T+h} = x_{T+h}^\top \hat{\beta}$ for $h \geq 1$. Future values, x_{T+h} , must be known to calculate the prediction. In time series, the prediction of future values has a slightly different nature and then is when we use the word *forecast*. The regressors in the 1-step-ahead forecast are known, but they are effectively unknown for longer horizons and must be forecasted first. For example, given $y_t = 5 - 0.5y_{t-1} + u_t$ for $t = 1, \dots, T$; the 1-step-ahead forecast is $\hat{y}_{T+1}^* = 5 - 0.5y_T$ with known y_T . However, the 2-step-ahead forecast is $\hat{y}_{T+2}^* = 5 - 0.5\hat{y}_{T+1}^*$, which uses the previous forecast value, \hat{y}_{T+1}^* .

In the **tvReg**, we refer to prediction when z_t is a random variable and to forecast when $z_t = \tau$. Note that future values of the conditional variable, z_{T+h} , must be given for prediction. For example, the prediction problem $\hat{y}_{T+h} = x_{T+h}^\top \hat{\beta}(z_t)$ for $h \geq 1$ requires the future values x_{T+h} and z_{T+h} . Whereas, the forecast problem $\hat{y}_{T+h}^* = x_{T+h}^\top \hat{\beta}(T + h)$ requires only the future values x_{T+h} . Thus, the predict and forecast methods in **tvReg** are slightly different.

11 Standard usage of predict and forecast

The forecast method is implemented for the class attributes "tvsure", "tvplm", "tvar", "tvlm" and "tvar". As an example, the three days ahead forecast of model TVHAR, evaluated in Section [Application to risk management](#) using the first 2000 values of the dataset RV, is provided in the lines of code below. This is a TVAR(1) model with two exogenous variables, RV_week and RV_month. The argument newexogen requires three values of these exogenous variables and variable n.ahead = 3.

```
> newexogen <- cbind(RV$RV_week[2001:2003], RV$RV_month[2001:2003])
> forecast(TVHAR, n.ahead = 3, newexogen = newexogen)
```

```
[1] 2.200921e-05 2.566854e-05 2.466637e-05
```

The forecast method requires the argument object. In addition, other arguments are necessary, some of them depending on the class attribute of object.

Forecast horizon The argument n.ahead is a scalar with the forecast horizon. By default, it is set to 1.

Type of forecast It is possible to run either an increasing window forecast (default), when the argument `winsize = 0` or a rolling window forecast with a window size defined in the argument `winsize`.

newdata

These arguments belong to the forecast methods and it is a "vector", "data.frame" or "matrix" containing the new values of the regressors in the model. It is not necessary to enter the intercept. Note that this `newdata` does not refer to the variables in exogen which might be part of the "tvar" and "tvvar" objects. Those must be included in `newexogen`, if needed.

newexogen

This argument appears in the forecast method for the class attributes "tvar" and "tvvar" and it must be entered when the initial model contains exogen variables. It is a "vector", "data.frame" or "matrix".

The `predict` method is implemented for the same class attributes than the `forecast`. It does not require arguments `n.ahead` and `winsize`, but arguments `newdata` and `newexogen` are defined as in `forecast`. In addition, new values of the smoothing variable must be entered into the argument `newz`. This must be of the class attribute "vector" or "numeric". The code below, predicts three future values of the TVHAR model fitted above.

```
> newdata <- RV$RV_lag[2001:2003]
> newexogen <- cbind(RV$RV_week[2001:2003], RV$RV_month[2001:2003])
> newz <- RV$RQ_lag_sqrt[2001:2003]
> predict(TVHARQ, newdata, newz, newexogen = newexogen)
```

```
[1] 1.741663e-05 2.402516e-05 2.088794e-05
```

The example below shows the usage of the `forecast` and `predict` methods for the class attribute "tvsure".

The lines of code below forecast three values for model `TVOLS.fit` evaluated in Section [Standard usage of tvSURE](#). The method needs a set of new values in the argument `newdata`, which must have the same number of columns as the original dataset.

```
> newKmenta <- data.frame(consump = c(95, 100, 102), price = c(90, 100, 103),
+                         farmPrice = c(70, 95, 103), income = c(82, 94, 115),
+                         trend = c(21:23))
> forecast(TVOLS.fit, newdata = newKmenta, n.ahead = 3)
```

```
      demand    supply
[1,] 97.92300 95.32852
[2,] 98.94076 103.48589
[3,] 105.36951 106.26576
```

In case the smoothing variable in the model is a random variable, the `predict` method for the class attribute "tvsure" requires also a new set of values in argument `newz`. The chunk below first fits a TVSURE model, `tvOLS.z.fit`, to the `Kmenta` data with the same system of equations as in the `TVOLS.fit`, but with random variable as the smoothing variable, which is generated as an ARMA(2,2) process. Three values of the dependent variable are predicted with the `predict` method. In addition to new values in the argument `newdata`, it requires a set of new smoothing values in the argument `newz`. It returns the predicted values as a matrix with as many columns as equations in the system.

```
> nobs <- nrow(Kmenta)
> smoothing <- arima.sim(n = nobs + 3, sd = sqrt(0.1796),
+                      list(ar = c(0.8897, -0.4858), ma = c(-0.2279, 0.2488)))
> smoothing <- as.numeric(smoothing)
> tvOLS.z.fit <- tvSURE(system, data = Kmenta, z = smoothing[1:nobs])
> newSmoothing <- tail(smoothing, 3)
> predict(tvOLS.z.fit, newdata = newKmenta, newz = newSmoothing)
```

```
      demand    supply
[1,] 100.0195 96.50136
[2,] 100.3919 105.29293
[3,] 106.1426 107.97822
```

The `forecast` and `predict` methods for the rest of the class attributes in the package follow similar patterns, and further examples can be found in the documentation of the **tvReg**.

12 Summary

Research of time-varying coefficient linear models and their estimation using kernel smoothing methods has seen a great theoretical development during the last two decades. Our own work in this field has served as inspiration to code the **tvReg** because we encounter a lack of computer applications with this functionality. Indeed, we expect that this package empowers empirical researchers working with regression and time series models with a computing tool that allows for more flexible models.

Within the R framework: (i) the **tvReg** extends functions in the R packages **systemfit**, **plm** and **vars**; (ii) it extends functions **lm**, **ar.ols** and **arima** to allow for varying coefficients; (iv) it complements R packages **mgcv** and **gam** for the linear regression model by providing a consistent estimator of this model for in case of dependency and a general distribution in the error term; (v) it complements R package **mgm** by adding the time-varying impulse response (TVIRF) function which is commonly used in macroeconomics; and (vi) it complements R package **bvarsv** and **MARSS** which estimate the TVVAR and TVIRF within the state-space framework. In addition, the **confint**, **fitted**, **forecast**, **plot**, **predict**, **print**, **resid** and **summary** methods are implemented for all class attributes in the **tvReg** and will allow the user to conveniently produce their research output. In any case, the user is able to produce customised plots and summaries from the returns of the functions, whose elements are accessible in the same manner as other R "list" objects.

Finally, the **tvReg** shows multiple applications in economics and finance. Specifically in asset management, portfolio management, risk management, health policy and monetary policy. The methods and datasets permit to verify results in Aslanidis and Casas (2013); Casas et al. (2018, 2019, 2021). Models in this paper are used in other fields too. For example, Reikard (2009) uses the TVLM to forecast the wave energy flux and Haslbeck et al. (2021) uses the TVVAR in different applications in psychology. The **tvReg** is therefore not only relevant and original but also timely.

Acknowledgements

We thank the unknown referee for the constructive suggestions and comments on an earlier version of this paper. The first author would also like to thank her co-authors of related papers: Nektarios Aslanidis, Eva Ferreira, Jiti Gao, Stefano Grassi, Xiuping Mao, Susan Orbe, Bin Peng, Helena Veiga and Shangyu Xie whose comments, suggestions and collaboration have helped to shape this code. The first author acknowledges financial support from the European Research Council under the European Union's Horizon 2020 research and innovation program (grant no. 657182) and from the Ministerio de Economía y Competitividad (grant no. ECO2014-51914-P). The second author acknowledges financial support from the MINECO (grant no. MTM2017-82724-R), MICINN (grant no. PID2020-113578RB-I00), and from the Xunta de Galicia (Grupos de Referencia Competitiva ED431C-2020-14 and Centro de Investigación del Sistema Universitario de Galicia ED431G 2019/01), all of them through the ERDF.

Bibliography

- T. G. Andersen and T. Bollerslev. Answering the skeptics: Yes, standard volatility models do provide accurate forecasts. *International Economic Review*, 39:885–905, 1998. URL <https://doi.org/10.2307/2527343>. [p94]
- N. Aslanidis and I. Casas. Nonparametric correlation models for portfolio allocation. *Journal of Banking & Finance*, 37:2268–2283, 2013. URL <https://doi.org/10.1016/j.jbankfin.2013.01.010>. [p91, 97]
- T. Bollerslev, G. Tauchen, and H. Zhou. Expected stock returns and variance risk premia. *The Review of Financial Studies*, 22(11):44–63, 2009. URL <https://doi.org/10.1093/rfs/hhp008>. [p94]
- T. Bollerslev, A. J. Patton, and R. Quaedvlieg. Exploiting the errors: A simple approach for improved volatility forecasting. *Journal of Econometrics*, 192(1):1–18, 2016. URL <https://doi.org/10.1016/j.jeconom.2015.10.007>. [p95]
- Z. Cai. Trending time-varying coefficient time series with serially correlated errors. *Journal of Econometrics*, 136:163–188, 2007. URL <https://doi.org/10.1016/j.jeconom.2005.08.004>. [p92]
- Z. Cai, J. Fan, and Q. Yao. Functional-coefficient regression models for nonlinear time series. *Journal of the American Statistical Association*, 95:941–956, 2000. URL <https://doi.org/10.1080/01621459.2000.10474284>. [p93]
- Z. Cai, Q. Li, and J. Y. Park. Functional-coefficient models for nonstationary time series data. *Journal of Econometrics*, 148:101–113, 2009. URL <https://doi.org/10.1016/j.jeconom.2008.10.003>. [p93]

- C. K. Carter and R. Kohn. On gibbs sampling for state space models. *Biometrika*, 81:541–553, 1994. URL <https://doi.org/10.1093/biomet/81.3.541>. [p80, 88]
- I. Casas, X. Mao, and H. Veiga. Reexamining financial and economic predictability with new estimators of realized variance and variance risk premium. Working papers, CREATES, Aarhus University, 2018. URL http://econ.au.dk/fileadmin/site_files/filer_oekonomi/Working_Papers/CREATES/2018/rp18_10.pdf. [p95, 97]
- I. Casas, E. Ferreira, and S. Orbe. Time-varying coefficient estimation in SURE models. application to portfolio management. *Journal of Financial Econometrics*, 19:707–745, 2019. URL <https://doi.org/10.1093/jjfinec/nbz010>. [p81, 85, 97]
- I. Casas, J. Gao, B. Peng, and S. Xie. Time-varying income elasticities of healthcare expenditure for the OECD and Eurozone. *Journal of Applied Econometrics*, 36:328–345, 2021. URL <https://doi.org/10.1002/jae.2809>. [p82, 86, 97]
- Y. Chang and E. Martinez-Chombo. Electricity demand analysis using cointegration and error-correction models with time varying parameters: The mexican case. Working papers, Rice University, Department of Economics, 2003. URL <https://ideas.repec.org/p/ecl/riceco/2003-08.html>. [p92, 93]
- X. B. Chen, J. Gao, D. Li, and P. Silvapulle. Nonparametric estimation and forecasting for time-varying coefficient realized volatility models. *Journal of Business & Economic Statistics*, online:1–13, 2017. URL <https://doi.org/10.1080/07350015.2016.1138118>. [p86, 92, 94]
- C. K. Chu and J. S. Marron. Comparison of two bandwidth selectors with dependent errors. *Annals of Statistics*, 19:1906–1918, 1991. URL <https://doi.org/10.1214/aos/1176348377>. [p83]
- T. Cogley and T. J. Sargent. Drifts and volatilities: monetary policies and outcomes in the post wwii us. *Review of Economic Dynamics*, 8:262–302, 2005. URL <https://doi.org/10.1016/j.red.2004.10.009>. [p83]
- F. Corsi. A simple approximate long-memory model of realized volatility. *Journal of Financial Econometrics*, 7(2):174–196, nov 2009. URL <https://doi.org/10.1093/jjfinec/nbp001>. [p92, 94]
- Y. Croissant and G. Millo. Panel data econometrics in R: The plm package. *Journal of Statistical Software*, 27(2):1–43, 2008. URL <https://doi.org/10.18637/jss.v027.i02>. [p79]
- Y. Croissant and G. Millo. *Panel Data Econometrics with R: the plm package*. Wiley, 2018. [p79]
- R. Dahlhaus. Fitting time series models to nonstationary processes. *Annals of Statistics*, 25:1–37, 1997. URL <https://doi.org/10.1214/aos/1034276620>. [p83]
- M. Das. Instrumental variables estimators of nonparametric models with discrete endogenous regressors. *Journal of Econometrics*, 124:335–361, 2005. URL <https://doi.org/10.1016/j.jeconom.2004.02.001>. [p93]
- C. Dong, J. Gao, and B. Peng. Varying-coefficient panel data models with nonstationarity and partially observed factor structure. *Journal of Business & Economic Statistics*, 39:700–711, 2021. URL <https://doi.org/10.1080/07350015.2020.1721294>. [p82]
- Dong, C. Jiti Gao, J. and Peng, B. Semiparametric single-index panel data models with cross-sectional dependence. *Journal of Econometrics*, 188:301–312, 2015. URL <https://doi.org/10.1016/j.jeconom.2015.06.001>. [p82]
- R. L. Eubank. *Nonparametric Regression and Spline Smoothing*. New York: Marcel Dekker, 2nd edition, 1999. [p79]
- E. Fama and K. French. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33:3–56, 1993. URL [https://doi.org/10.1016/0304-405x\(93\)90023-5](https://doi.org/10.1016/0304-405x(93)90023-5). [p85]
- E. F. Fama and K. R. French. A five-factor asset pricing model. *Journal of Financial Economics*, 116(1): 1–22, 2015. URL <https://doi.org/10.1016/j.jfineco.2014.10.010>. [p85]
- E. F. Fama and K. R. French. International tests of a five-factor asset pricing model. *Journal of Financial Economics*, 123:441–463, 2017. URL <https://doi.org/10.1016/j.jfineco.2016.11.004>. [p85]
- J. Fan and I. Gijbels. *Local Polynomial Modeling and Its Applications*. Chapman and Hall, London, 1996. [p81]

- J. Fan and W. Zhang. Simultaneous confidence bands and hypothesis testing in varying-coefficient models. *Scandinavian Journal of Statistics*, 27:715–731, 2000. URL <https://doi.org/10.1111/1467-9469.00218>. [p86]
- K. R. French. Data library, 2016. URL http://mba.tuck.dartmouth.edu/pages/faculty/ken.french/data_library.html. Accessed: 2016-09-01. [p85]
- J. Gao and P. Phillips. Functional coefficient nonstationary regression. Cowles Foundation Discussion Papers 1911, Cowles Foundation for Research in Economics, Yale University, 2013. URL <https://ideas.repec.org/p/cwl/cwldpp/1911.html>. [p93]
- U.-G. Gerdtham, J. Soegaard, F. Andersson, and B. Jönsson. An econometric analysis of health care expenditure: A cross-section study of the OECD countries. *Journal of Health Economics*, 11:63–84, 1992. URL [https://doi.org/10.1016/0167-6296\(92\)90025-v](https://doi.org/10.1016/0167-6296(92)90025-v). [p86]
- J. Haslbeck, L. Bringmann, and L. Waldorp. A tutorial on estimating time-varying vector autoregressive models. *Multivariate Behavioral Research*, 56:120–149, 2021. URL <https://doi.org/10.1080/00273171.2020.1743630>. [p97]
- J. M. Haslbeck and L. J. Waldorp. mgm: Estimating time-varying mixed graphical models in high-dimensional data. *Journal of Statistical Software*, 93:1–46, 2020. URL <https://doi.org/10.18637/jss.v093.i08>. [p79]
- T. Hastie. *gam: Generalized Additive Models*, 2022. URL <https://CRAN.R-project.org/package=gam>. R package version 1.20.1. [p80]
- T. Hastie and R. Tibshirani. Varying-coefficient models. *Journal of the Royal Statistical Society: Series B (Methodological)*, 55:757–796, 1993. URL <https://doi.org/10.1111/j.2517-6161.1993.tb01939.x>. [p79, 93]
- D. J. Henderson, S. C. Kumbhakar, Q. Li, and C. F. Parmeter. Smooth coefficient estimation of a seemingly unrelated regression. *Journal of Econometrics*, 189:148–162, 2015. URL <https://doi.org/10.1016/j.jeconom.2015.07.002>. [p81]
- A. Henningsen and J. D. Hamann. systemfit: A package for estimating systems of simultaneous equations in R. *Journal of Statistical Software*, 23(4):1–40, 2007. URL <https://doi.org/10.18637/jss.v023.i04>. [p79]
- E. E. Holmes, E. J. Ward, and K. Wills. Marss: Multivariate autoregressive state-space models for analyzing time-series data. *The R Journal*, 4(1):30, 2012. URL <https://doi.org/10.32614/rj-2012-002>. [p79]
- G. Kapetanios, H. Mumtaz, I. Stevens, and K. Theodoridis. Assessing the economy-wide effects of quantitative easing. *The Economic Journal*, 122:316–347, 2012. URL <http://www.jstor.org/stable/23324226>. [p83]
- G. Kapetanios, M. Marcellino, and F. Venditti. Large time-varying parameter VARs: a non-parametric approach. Temi di discussione (economic working papers), Bank of Italy, Economic Research and International Relations Area, 2017. URL https://ideas.repec.org/p/bdi/wptemi/td_1122_17.html. [p83]
- J. Kmenta. *Elements of Econometrics*. New York: Macmillan, 2nd edition, 1986. [p84]
- F. Krueger. *bvarsv: Bayesian analysis of a vector autoregressive model with stochastic volatility and time-varying parameters*, 2015. URL <https://CRAN.R-project.org/package=bvarsv>. R package version 1.1. [p80]
- Y. Liu and F. Guo. A bayesian time-varying coefficient model for multitype recurrent events. *Journal of Computational and Graphical Statistics*, 29:383–395, 2020. URL <https://doi.org/10.1080/10618600.2019.1686988>. [p80]
- H. Lütkepohl. *New Introduction to Multiple Time Series Analysis*. Springer, 2005. [p83]
- E. Mammen. Bootstrap and wild bootstrap for high dimensional linear models. *The Annals of Statistics*, 21:255–285, 1993. URL <https://doi.org/10.1214/aos/1176349025>. [p86]
- S. Orbe, E. Ferreira, and J. Rodriguez-Poo. Nonparametric estimation of time varying parameters under shape restrictions. *Journal of Econometrics*, 126:53–77, 2005. URL <https://doi.org/10.1016/j.jeconom.2004.02.006>. [p81]

- B. Pfaff. VAR, SVAR and SVEC models: Implementation within R package vars. *Journal of Statistical Software, Articles*, 27(4):1–32, 2008. URL <https://www.jstatsoft.org/v027/i04>. [p79]
- G. E. Primiceri. Time varying structural vector autoregressions and monetary policy. *Review of Economic Studies*, 72:821–852, 2005. URL <https://doi.org/10.1111/j.1467-937x.2005.00353.x>. [p80, 83, 88, 89]
- G. Reikard. Forecasting ocean wave energy: Tests of time-series models. *Ocean Engineering*, 36:348–356, 2009. URL <https://doi.org/10.1016/j.oceaneng.2009.01.003>. [p97]
- P. Robinson. Nonparametric estimation of time-varying parameters. In P. Hackl, editor, *Statistical Analysis and Forecasting of Economic Structural Change*, pages 253–264. Springer, Berlin, 1989. URL https://doi.org/10.1007/978-3-662-02571-0_15. [p79, 92]
- C. A. Sims. Macroeconomics and reality. *Econometrica*, 48:1–48, 1980. URL <https://doi.org/10.2307/1912017>. [p82]
- H. S. M. D. Store. Intraday products, 2017. URL <http://download-stock-data.webs.com/>. Accessed: 2017-11-01. [p94]
- Y. Sun, R. Carroll, and D. Li. Semiparametric estimation of fixed-effects panel data varying coefficient models. *Advances in Econometrics*, 25:101–129, 12 2009. URL [https://doi.org/10.1108/s0731-9053\(2009\)0000025006](https://doi.org/10.1108/s0731-9053(2009)0000025006). [p82]
- Y. Sun, Z. Cai, and Q. Li. Semiparametric functional coefficient models with integrated covariates. *Econometric Theory*, 29:659–672, 2013. URL <https://doi.org/10.1017/s0266466612000710>. [p93]
- S. N. Wood. *Generalized Additive Models: An Introduction with R*. Chapman and Hall/CRC, 2 edition, 2017. [p80]
- Z. Xiao. Functional-coefficient cointegration models. *Journal of Econometrics*, 152:81–92, 2009. URL <https://doi.org/10.1016/j.jeconom.2009.01.008>. [p93]
- Y. Yan, J. Gao, and B. Peng. Nonparametric time-varying vector moving average (infinity) models. Technical report, SSRN Electronic Journal, 2021. URL <http://dx.doi.org/10.2139/ssrn.3729872>. [p83]
- A. Zellner. An efficient method of estimating seemingly unrelated regressions and tests for aggregation bias. *Journal of the American Statistical Association*, 57:348–368, 1962. URL <https://doi.org/10.1080/01621459.1962.10480664>. [p80]
- T. Zhang and W. B. Wu. Inference of time-varying regression models. *Annals of Statistics*, 40(3): 1376–1402, 2012. URL <https://doi.org/10.1214/12-aos1010>. [p93]

Isabel Casas
Department of Economics and Finance
University of Deusto
Spain
ORCID: 0000-0002-9063-9670
casasis@gmail.com

Rubén Fernández-Casal
Department of Mathematics
University of A Coruña
Spain
ORCID: 0000-0002-5785-3739
ruben.fcasal@udc.es