MULTIATSM: AN R PACKAGE FOR ARBITRAGE-FREE MULTICOUNTRY AFFINE TERM STRUCTURE OF INTEREST RATES MODELS WITH UNSPANNED MACROECONOMIC RISK

Rubens Moura

LIDAM Discussion Paper LFIN
2022 / 01

UCLouvain

LFIN | LiDAM Louvain Institute of Data Analysis and Modeling in economics and statistics

LFIN

Voie du Roman Pays 34, L1.03.01 B-1348 Louvain-la-Neuve

Tel (32 10) 47 43 04

Email: lidam-library@uclouvain.be

https://uclouvain.be/en/research-institutes/lidam/lfin/publications.html

MultiATSM: An R Package for Arbitrage-free Multicountry Affine Term Structure of Interest Rates Models with Unspanned Macroeconomic Risk

Rubens Moura
UCLouvain

Abstract

The R package **MultiATSM** provides several estimation routines and additional outputs for eight classes of affine term structure of interest rates models (ATSMs). All the ATSMs from this package build on the single-country unspanned macroeconomic risk framework by Joslin, Priebsch, and Singleton (2014). The **MultiATSM** package also features alternative multicountry extensions based on the settings of Jotikasthira, Le, and Lundblad (2015), which imposes the existence of a dominant (global) economy, and Candelon and Moura (2021), where the joint dynamics of the risk factors are captured by a GVAR setup. For each ATSM, the **MultiATSM** package produces a set of model outputs that includes: *(i)* the graphical representations from the model fit, the orthogonalized and generalized versions of impulse response and forecast error variance decomposition from bond yields and risk factors; *(ii)* a number of bootstrap procedures for constructing confidence intervals, and *(iii)* out-of-sample forecasting of bond yields.

Keywords: term structure of interest rates models, macrofinance, international finance, financial-economic connectedness, R, **MultiATSM**.

1. Introduction

The term structure of interest rates (or yield curve) describes the relationship between bond yields and different investment maturities. The workhorse in yield curve modelling is referred to as affine term structure models (ATSMs). Based on the assumption of absence of arbitrage opportunities, ATSMs provide a simple framework to assess how markets price different sources of risk and generates predictions for the price of any bond.[1]

Early ATSMs have gained popularity in the academic literature due to their capability of capturing nearly all term structure movements.[2] Despite producing accurate statistical descriptions of the yield curve, this class of models provide little insights into the underlying economic forces that steer interest rates fluctuations. In view of this limitation, a large body of research has emerged to study the interplay between the term structure and macroeconomic developments.[3]

[1]See Piazzesi (2010) and Gürkaynak and Wright (2012) for a comprehensive summary of this literature.

[2]This line of research includes, for instance, Vasicek (1977), Duffie and Kan (1996), and Dai and Singleton (2002).

[3]See, for instance, the seminal works of Ang and Piazzesi (2003) and Rudebusch and Wu (2008).

The unspanned economic risk framework developed by Joslin *et al.* (2014) (henceforth JPS, 2014) is a recent and referenced macrofinance ATSM. In essence, this model assumes the absence of arbitrage opportunities and considers a linear state space representations of the yield curve dynamics. With respect to previous macrofinance ATSMs, JPS (2014) offer a more tractable estimation approach that simultaneously combine the traditional yield curve factors (spanned factors) along with economic and financial variables (unspanned factors).

The work of Joslin *et al.* (2014) lays the foundations for the modelling frameworks from the R package **MultiATSM** (Moura 2022), which is available from the Comprehensive R Archive Network (CRAN) at `https://CRAN.R-project.org/package=MultiATSM`. Aside from the original single-country framework of JPS (2014), this same package also features the multicountry extensions of Jotikasthira *et al.* (2015) (henceforth JLL, 2015) and Candelon and Moura (2021) (henceforth CM, 2021). In total, the package provides several functions to construct 8 types of ATSMs that include the original versions and variations of these three setups. In addition to providing complete estimation routines for these models, the **MultiATSM** package generates a myriad of other outputs. Specifically, in terms of graphical representations, one can obtain the model fit of the country term structures in addition to both the orthogonalized and generalized versions of the *(i)* impulse response functions of bond yields on shocks to the risk factors and *(ii)* forecast error variance decomposition of bond yields and risk factors. The confidence intervals of this set of outputs can be computed via residual-based, block or wild bootstrap procedures. Furthermore, the package can also be used to produce out-of-sample forecasts of bond yields. The goal of this paper is to provide detailed guidance on the use of the **MultiATSM** package.

There are a few notable packages for term structure modelling in R. **YieldCurve** (Guirreri 2015) and **fBonds** (Setz 2017) provide a collection of functions to compute term structures based on the frameworks of Nelson and Siegel (1987) and Svensson (1994). These yield curve methods have gained the attention among financial practitioners because they offer a parsimonious model parametrization while granting a good fit of the term structure. However, these approaches are subject to criticism for allowing arbitrage opportunities, a feature that can be avoided with ATSMs. Moreover, the functionalities of **YieldCurve** and **fBonds** are limited to the estimation of the model parameters and the yield curve fit. As such, other popular model outputs, such as the ones available at **MultiATSM**, are not contemplated in these packages. Relatedly, the R package **EWS** (Hasse and Lajaunie 2021) uses information from the short and the long-end of the term structure (yield spreads) to forecast future recessions. Yet, this package does not intend to model the whole maturity spectrum of term structures.

The remainder of the paper is as follows. Section 2 provides the theoretical foundations from the ATSMs of the **MultiATSM** package, and Section 3 provides a detailed description of the characteristics of each ATSM. In the subsequent sections, the focus is on the description of the various necessary pieces to implement ATSMs in practice. Specifically, Section 4 describes the dataset available at this package and the set of functions that are useful to retrieve data from Excel files. Section 5 exposes the necessary inputs that have to be specified by the user. In Section 6 the estimation procedure is detailed. Section 7 shows how to use the **MultiATSM** package to estimate ATSMs from scratch, as well as to closely reproduce some empirical features of academic papers. To ease the exposition, throughout the next sections, the database used by CM (2021) is employed to illustrate the use of the various functions available in this package.

2. ATSMs with unspanned economic risks: theoretical background

In this section, we outline the arbitrage-free ATSMs with unspanned macroeconomic risk frameworks. One convenient aspect of these setups is that they enable a clear split of the yield curves into cross-sectional (governed by the Q-dynamics model parameters) and time series (formed from the P-dynamics parameters) dimensions. In light of this characteristic of the models, we present the single and the multicountry Q-dynamics model dimensions in Section 2.1. Next, we expose the specific features of the risk factor dynamics under the P-measure of the various restricted and unrestricted VARs settings in Section 2.2.

2.1. Cross-sectional dimension of the term structure (Q-dynamics)

Single-country specifications (individual Q-dynamics models)

ATSMs cross-sectional settings are based on two central equations. The first one assumes that the country i short-term interest rate, $r_{i,t}$, is an affine function of N unobserved (latent) country-specific factors, $X_{i,t}$:

$$r_{i,t} = \delta_{i,0} + \delta_{i,1}^{\top} X_{i,t}. \tag{1}$$

The second equation refers to the state dynamics. By assumption, for each country i, the unobserved factors dynamics follow a maximally flexible affine VAR(1) model under the Q-measure:

$$X_{i,t} = \mu_{i,X}^{Q} + \Phi_{i,X}^{Q} X_{i,t-1} + \Sigma_{i,X}\varepsilon_{i,t}^{Q}, \qquad \varepsilon_{i,t}^{Q} \sim \mathcal{N}(0, I_N). \tag{2}$$

Supported by Equations (1) and (2), Dai and Singleton (2000) show that the country-specific yield of a zero-coupon bond with maturity of n periods, $y_{i,t}^{(n)}$, is affine in $X_{i,t}$:

$$y_{i,t}^{(n)} = A_{i,n}(\Theta_n) + B_{i,n}(\Theta_n) X_{i,t}, \tag{3}$$

where $A_{i,n}(\Theta_n)$ and $B_{i,n}(\Theta_n)$ are restricted to preclude arbitrage opportunities in this bond market. For notational simplicity, we collect J yields into the vector $Y_{i,t} = [y_{i,t}^{(1)}, y_{i,t}^{(2)}, ..., y_{i,t}^{(J)}]^{\top}$, the J intercepts into $A_X(\Theta_i) = [A_{i,1}(\Theta_1), A_{i,2}(\Theta_2), ..., A_{i,J}(\Theta_J)]^{\top}$, and the N slope coefficients into a $J \times N$ matrix $B_X(\Theta_i) = [B_{i,1}^{\top}(\Theta_1), B_{i,2}^{\top}(\Theta_2), ..., B_{i,N}^{\top}(\Theta_N)]^{\top}$. Accordingly, we outline the country's i yield curve cross-section dimension as:

$$Y_{i,t} = A_X(\Theta_i) + B_X(\Theta_i) X_{i,t}. \tag{4}$$

It follows from Equations (1) and (2) that the parameter set $\Theta_i = \{\mu_{i,X}^{Q}, \Phi_{i,X}^{Q}, \Sigma_{i,X}, \delta_{i,0}, \delta_{i,1}\}$ fully characterizes the cross-section of country's i term structure. Importantly, Dai and Singleton (2000) show that this system is not identified without the imposition of restrictions.[4] To circumvent this problem, JPS (2014) employ the three sets of (minimal) restrictions proposed by Joslin *et al.* (2011). Firstly, they impose the latent factors to be zero-mean processes,

[4] The authors demonstrate that $X_{i,t}$ and any invertible affine transformation of $X_{i,t}$ have observationally equivalent representations. The model restrictions imposed by Joslin, Singleton, and Zhu (2011) ensure that no additional invariant rotation is possible.

namely $\mu^Q_{i,X} = 0$. Secondly, they choose $\delta_{i,1}$ to be a N-dimensional vector with all entries equal to one. Lastly, $\Phi^Q_{i,X}$ is a diagonal matrix, the elements of which are the real and distinct eigenvalues, λ^Q_i, of the matrix of eigenvectors of $\Phi^Q_{i,X}$.

Joslin *et al.* (2011) also demonstrate that a rotation from latent factors $X_{i,t}$ to portfolios of yields, the spanned factors $P_{i,t}$, leads to an observationally equivalent term structure representation. This invariant transformation assumes that N portfolios of yields are perfectly priced and observed without errors, while the remaining $J - N$ portfolios are priced and observed imperfectly. Specifically, the spanned factors are computed as $P_{i,t} = V_i Y_{i,t}$, for some full-rank matrix V_i. Supported by this definition, we can rewrite the vector of country-specific yields as an affine function of $P_{i,t}$ as follows:[5]

$$Y_{i,t} = A_P(\Theta_i) + B_P(\Theta_i)P_{i,t}. \tag{5}$$

The rotation from $X_{i,t}$ to $P_{i,t}$ is convenient for at least two reasons. First, in contrast to $X_{i,t}$, $P_{i,t}$ is directly observable, and its N time series elements can be interpreted as the traditional yield curve factors.[6] Second, the model representation in terms of $P_{i,t}$ allows for an opportune decomposition of the likelihood function that makes the model estimation tractable and the interpretation of each parameter straightforward.

Multicountry specifications (joint Q-dynamics models)

To build the multicountry yield curve cross-section, we simply stack the sets of country yields, spanned factors, and intercepts from Equation (5) into, respectively, $Y_t = [Y^\top_{1,t}, Y^\top_{2,t}, ..., Y^\top_{C,t}]^\top$, $P_t = [P^\top_{1,t}, P^\top_{2,t}, ..., P^\top_{C,t}]^\top$, and $A_P(\Theta) = [A^\top_P(\Theta_1), A^\top_P(\Theta_2), ..., A^\top_P(\Theta_C)]^\top$. Additionally, we set $B_P(\Theta)$ as block diagonal, $B_P(\Theta) = diag(B_P(\Theta_1), B_P(\Theta_2), ..., B_P(\Theta_C))$. Accordingly, we can write

$$Y_t = A_P(\Theta) + B_P(\Theta)P_t. \tag{6}$$

2.2. Time series dimension of the term structure (P-dynamics)

In the modelling frameworks available at the **MultiATSM** package, the risk factor dynamics under the P-measure may include N domestic spanned variables ($P_{i,t}$), in addition to M domestic ($M_{i,t}$) and G global unspanned factors (M^W_t). The dynamics of this same risk factors may evolve as either an unrestricted or a restricted VAR models. While the former refer to JPS related settings, the latter feature the GVAR and JLL frameworks.

It is worth stressing the role of unspanned factors in the yield curve developments. Although unspanned factors are absent in the cross-section dimension of the models, they influence the dynamics of the spanned factors which, in turn, affect directly bond yields.

JPS-based models

The country-specific state vector, $Z_{i,t}$, is formed from stacking the global and domestic (unspanned and spanned) risk factors: $Z_{i,t} = [M^{W\top}_t,\ M^\top_{i,t},\ P^\top_{i,t}]^\top$. In JPS-based setups, $Z_{i,t}$

[5]The loadings of $Y_{i,t}$ are computed as $A_P(\Theta_i) = I - B_X(\Theta_i)(V_i B_X(\Theta_i))^{-1} V_i A_X(\Theta_i)$ and $B_P(\Theta_i) = B_X(\Theta_i)(V_i B_X(\Theta_i))^{-1}$.

[6]For instance, for $N = 3$ and V_i being the weight matrix that results from a principal component analysis, the portfolios of yields $P_{i,t}$ are commonly referred to as the level, slope, and curvature factors (see Section 6.1 for a more detailed discussion).

follows a standard unrestricted Gaussian VAR(1):

$$Z_{i,t} = C_i^{\mathbb{P}} + \Phi_i^{\mathbb{P}} Z_{i,t-1} + \Sigma_i \varepsilon_{Z,t}^{\mathbb{P}}, \qquad \varepsilon_{Z,t}^{\mathbb{P}} \sim \mathcal{N}(0, I_{G+M+N}). \tag{7}$$

JLL-based models

In JLL-based frameworks, the economic system comprises the global economy, one worldwide large (dominant) economy,[7] and another set of smaller economies. We denote by C the number of countries in this economic system, and by D the dominant economy.

In these models, the state vector is formed from a number of linear projections. This approach is intended to build purely country-specific risk factors, *i.e.*, risk factors that are free from the influence of the variables from other countries and/or from the global economy.

The construction of the domestic spanned factors follows a two-step procedure. First, for each economy i, $P_{i,t}$ is projected on $M_{i,t}$ of this same country

$$P_{i,t} = b_i M_{i,t} + P_{i,t}^e, \tag{8}$$

where the residuals $P_{i,t}^e$ are orthogonal to the economic fundamentals of the country i.

Second, for the non-dominant economies, $P_{i,t}^e$ is additionally projected on the orthogonalized spanned factors of the dominant country as follows:

$$P_{i,t}^e = c_i^D P_{D,t}^e + P_{i,t}^{e*}, \tag{9}$$

where $P_{i,t}^{e*}$ corresponds to the non-dominant country i set of residuals.

The design of the domestic unspanned factors also features two steps: for the dominant economy, $M_{D,t}$ is projected on the global economic factors

$$M_{D,t} = a_D^W M_t^W + M_{D,t}^e \tag{10}$$

and, for the other economies, the residuals of the previous regression are used to compute

$$M_{i,t} = a_i^W M_t^W + a_i^D M_{D,t}^e + M_{i,t}^{e*}. \tag{11}$$

Accordingly, the state vector is formed by $Z_t^e = [M_t^{W^\top}, M_{D,t}^{e^\top}, P_{D,t}^{e^\top}, M_{2,t}^{e*^\top}, P_{2,t}^{e*^\top} ... M_{C,t}^{e*^\top}, P_{C,t}^{e*^\top}]^\top$ and its dynamics evolve as a restricted VAR(1) of the following form:

$$Z_t^e = C_Y^e + \Phi_Y^e Z_{t-1}^e + \Sigma_Y^e \varepsilon_{Z,t}^e, \qquad \varepsilon_{Z,t}^e \sim \mathcal{N}(0, I_F). \tag{12}$$

where $F = CK + G$ and the set of restrictions imposed in Φ^e and Σ_Y^e are detailed in Jotikasthira *et al.* (2015).

GVAR-based models

In the **MultiATSM** package, the GVAR setup is formed from two parts: the $VARX^*$ and the marginal models. The former captures the developments from the domestic (spanned and unspanned) factors, whereas the latter describes the joint dynamics of the global economy.[8]

[7] Noticeably, the model type *JLL NoDomUnit* is the only exception (see the detailed description of the different ATSMs in Section 3).

[8] See the work of Chudik and Pesaran (2016) for a thorough description of GVAR models.

VARX* models are small-scale local VAR augmented by the *star* and global factors. The star factors are a weakly exogenous weighted average of foreign variables constructed as follows:

$$Z_{i,t}^{*\top} = \sum_{j=1}^{C} w_{i,j} Z_{j,t}^{\top}, \qquad \sum_{j=1}^{C} w_{i,j} = 1, \quad w_{ii} = 0 \quad \forall i \in \{1, 2..C\}, \tag{13}$$

where $Z_{j,t}$ is a vector of domestic factors $Z_{j,t} = [M_{j,t}^{\top},\ P_{j,t}^{\top}]^{\top}$ with dimension $K = M + N$ and $w_{i,j}$ measures the degree of connectedness of country i with country j.

In terms of functional form, these models assume the structure VARX*(p, q, r) where p, q and r are the number of lags from, respectively, the domestic, the star, and the global risk factors. In its current version, the **MultiATSM** package provides the estimates for the case $p = q = r = 1$. In such a case, the dynamics of $Z_{i,t}$ is described as a VARX* of the following form:

$$Z_{i,t} = C_i^X + \Phi_i^X Z_{i,t-1} + \Phi_i^{X^*} Z_{i,t-1}^* + \Phi_i^{X^W} M_{t-1}^W + \Sigma_i^X \varepsilon_{i,t}^X, \qquad \varepsilon_{i,t}^X \sim \mathcal{N}(0, I_K). \tag{14}$$

The marginal model is a standard unrestricted VAR(1) featuring exclusively the global factors:

$$M_t^W = C^W + \Phi^W M_{t-1}^W + \Sigma^W \epsilon_t^W, \qquad \varepsilon_t^W \sim \mathcal{N}(0, I_G). \tag{15}$$

GVAR models require the specification of country-specific link matrices, W_is, to unify the individual VARX*'s. Formally, W_is are computed as

$$\begin{bmatrix} Z_{i,t} \\ Z_{i,t}^* \end{bmatrix}_{2K\times 1} \equiv \underset{2K\times CK}{W_i} \begin{bmatrix} Z_{1,t} \\ Z_{2,t} \\ \vdots \\ Z_{C,t} \end{bmatrix}_{CK\times 1} \tag{16}$$

Lastly, we compose the GVAR state vector by gathering the global economic variables and the country-specific risk factors, as $Z_t = [M_t^{W^\top},\ Z_{1,t}^{\top},\ Z_{2,t}^{\top}, \dots Z_{C,t}^{\top}]^{\top}$. As such, we can form a first order GVAR process as

$$Z_t = C_y + \Phi_y Z_{t-1} + \Sigma_y \epsilon_{y,t}, \qquad \varepsilon_{y,t} \sim \mathcal{N}(0, I_F), \tag{17}$$

where $C_y = [C^{W^\top},\ C_1^{X^\top},\ C_2^{X^\top}, \dots\ C_C^{X^\top}]^{\top}$, $\varepsilon_{y,t} = [\varepsilon_t^{W^\top},\ \varepsilon_{1,t}^{X^\top},\ \varepsilon_{2,t}^{X^\top}\ \dots\ \varepsilon_{C,t}^{X^\top}]^{\top}$, $\Sigma_y = diag(\Sigma^W,$ $\Sigma_1^X,\ \Sigma_2^X\ \dots\ \Sigma_C^X)$, and

$$\Phi_y = \begin{bmatrix} \Phi^W & 0_{G\times CK} \\ \Phi^{X^W} & G_1 \end{bmatrix}_{F\times F}, \tag{18}$$

where $\Phi^{X^W} = \begin{bmatrix} \Phi_1^{X^W} \\ \Phi_2^{X^W} \\ \vdots \\ \Phi_C^{X^W} \end{bmatrix}_{CK\times G}$ and $G_1 = \begin{bmatrix} \Phi_1 W_1 \\ \Phi_2 W_2 \\ \vdots \\ \Phi_C W_C \end{bmatrix}_{CK\times CK}$, for $\Phi_i = [\Phi_i^X,\ \Phi_i^{X^*}]$ and $i = 1, 2, \dots C$.

	P-dynamics					**Q-dynamics**		Σ		**Dom. Eco.**
	Individual		Joint			Individual	Joint	P	P and Q	
	UR	R	UR	R						
				JLL	*GVAR*					
Unrestricted VAR										
JPS	x					x			x	
JPS jointP			x			x			x	
VAR jointQ			x				x		x	
Restricted VAR (GVAR)										
GVAR sepQ					x	x			x	
GVAR jointQ					x		x		x	
Restricted VAR (JLL)										
JLL original				x			x	x		x
JLL NoDomUnit				x			x	x		
JLL jointSigma				x			x		x	x

Table 1: Summary of model estimation features. Under the P-measure, the dynamics of the risk factors can be unrestricted (*UR* columns) or restricted (*R* columns). The set of restrictions present in the JLL-based and GVAR-based models are described in Jotikasthira *et al.* (2015) and Candelon and Moura (2021), respectively. The estimation of the Σ matrix is done either exclusively with the other parameters of the P-dynamics (P column) or jointly under both P- and Q-parameters (P *and* Q column). The entries featuring X indicate that the referred characteristic is part of the model.

3. The ATSMs available at the MultiATSM package

In its current version, the **MultiATSM** package can generate the outputs of 8 different classes of ATSMs. Overall, the ATSMs differ in terms of the form of the P and Q dynamics, the model estimation procedure and the presence (or not) of a dominant economy. In Table 1, we summarize the general features of each model available at the package. We briefly comment below on the characteristics of the different frameworks.

As exposed in Section 2, under both the P and the Q-dynamics, the ATSMs can be cast on a country-by-country basis - as in JPS (2014) - or jointly for all the countries of the economic system - as in JLL(2015) and CM (2021). Further, under the P-measure, the risk factors dynamics follow a VAR(1) model of some sort (restricted or unrestricted). In terms of the model estimation process, most of the ATSMs follow the method described in Joslin *et al.* (2014). Specifically, the authors show that, due to the peculiar features of their specification, an efficient estimation of the parameters governing the Q and the P-measures can be carried out rather independently. The only exception is the variance-covariance matrix term, Σ, which is a common element to both the P and the Q likelihood functions.

The ATSMs in which the estimation is performed individually for each country are labeled as *JPS*, *JPS jointP* and *GVAR sepQ*. In the *JPS* setup, the set of risk factors includes exclusively each country's domestic factors and the global unspanned factors, whereas *JPS jointP* and *GVAR sepQ* also incorporate domestic risk factors of the other countries of the economic system. Noticeably, the difference between *JPS jointP* and *GVAR sepQ* stem from the restrictions imposed under the P-dynamics.

In terms of the multicountry frameworks, some aspects are worth highlighting. As for the models based on the setup of JLL (2015), the version *JLL original* follows closely the seminal work of JLL (2015), *i.e.*, it is assumed an economic cohort containing a worldwide dominant economy and a set of smaller countries, in addition to the estimation of the Σ matrix be performed exclusively under the P-measure.[9] The two other alternative versions assume the absence of a dominant country (*JLL NoDomUnit*) and the estimation of Σ under both the P and Q measures (*JLL jointSigma*), as in the standard JPS (2014) model. As for the remaining multicountry ATSMs, it is considered that the dynamics of the risk factors under the P-measure evolve according to a GVAR model. The version labeled *GVAR jointQ* is the one presented in CM (2021).

4. Risk factors dataset

4.1. Package dataset

MultiATSM package contains the four datasets used in CM (2021). The first set of data comprises several time series of zero-coupon bonds yields from four emerging markets: China, Brazil, Mexico, and Uruguay. It is worth noting that this package requires *(i)* for estimation purposes, the maturities of the bond yields to be the same for all countries (although the function `DataForEstimation()` handles different maturities across countries as inputs, the outputs generated by this same function are common bond yields for all the economies); *(ii)* bond yields to be expressed in percentage terms (and not in basis points) per annum. **MultiATSM** package does not support a routine to bootstrap zero-coupon yields from coupon bonds. As such, this data manipulation procedure, if necessary, must be handled by the user herself.

```
R> data("CM_Yields")
```

The second group of data concerns the time series of the risk factors. Specifically, along the terminology defined in JPS (2014), this dataset contains *(i)* country-specific spanned factors and *(ii)* an array of country-specific and global unspanned factors (namely, some measure of economic growth and inflation). Similarly to the case of the bond yields data, the measures of economic and financial variables must be constructed by the user.

```
R> data("CM_Factors")
```

The two last blocks of data are necessary for the estimation of the GVAR-based models. The trade flows database presents the sum of the value of all goods imports and exports between any two countries of the sample on a yearly basis since 1948. All the values are free on board and are expressed in U.S. dollars. This data are used to construct the transition matrix from the GVARs models.

```
R> data("CM_Trade")
```

The GVAR factors database casts country-specific lists of all the factors that are used in the estimation of each country's VARX. A specific function for computing the star risk factors is detailed in the Section 6.2.

[9] Jotikasthira *et al.* (2015) claim that, although this method is not fully efficient, it makes limited impact empirically.

```
R> data("CM_Factors_GVAR")
```

4.2. Importing data from Excel files

This package also offers an automatized procedure to extract data from Excel files and to, subsequently, prepare the risk factors database that are directly used in the estimation of the models. The use of the package functions requires that the databases *(i)* are constructed in separate Excel files for the unspanned factors and the term structure data. Additionally, for the GVAR-based models, the measures of interdependence must be compiled in a supplementary file; *(ii)* contain, in each Excel file, one separate tab per country. Further, in the case of the unspanned factors' database, one must include a separate tab for the global variables; and *(iii)* have identical variable labels across all the tabs within each file. For illustration, see the Excel file available at the package. One example of list of inputs to be provided is

```
R> Initial_Date <- "2006-09-01"
R> Final_Date <- "2019-01-01"
R> DataFrequency <- "Monthly"
R> GlobalVar <- c("GBC", "VIX")
R> DomVar <- c("Eco_Act", "Inflation", "Com_Prices", "Exc_Rates")
R> N <- 3
R> Economies <- c("China", "Mexico", "Uruguay", "Brazil", "Russia")
R> ModelType <- "JPS"
```

Based on these inputs, one can construct the variable *ZZfull* which contains the complete set of the model risk factors.

```
R> FactorLabels <- LabFac(N, DomVar, GlobalVar, Economies, ModelType)
R> ZZfull <- DataForEstimation(Initial_Date, Final_Date, Economies, N,
+                              FactorLabels, ModelType, DataFrequency)
```

5. Required user inputs

5.1. Basic inputs used in the model estimation

In order to estimate any ATSM, the user needs to specify several general model inputs, namely:

- Model type: a string-vector containing the label of the model to be estimated (as per described in Table 1);
- Frequency of the data: a string-vector that specifies the frequency of the time series data. The available options are: *Annually*, *Quarterly*, *Monthly*, *Weekly*, *Daily Business Days*, and *Daily All Days*;
- Economies: a string-vector containing the names of the economies which are part of the economic system;

- Number of spanned factors (N): number of country-specific spanned factors. In this package, N is assumed to be equal across countries;

- Global variables: a string-vector containing the labels of the global unspanned factors used in the model estimation;

- Domestic variables: a string-vector containing the labels of the domestic unspanned factors used in the model estimation;

- Bond yield set of maturities: a numerical-vector containing the maturities of the yields used in the model estimation. The units of this vector must be expressed in years. Alternatively, the user can rely on the `Maturities()` routine to extract the bond maturities from the time series of bond yields (see the documentation of the `Maturities()` function). In this case, the user must specify the unit in which the label of the bond yields are expressed, *e.g.*, if the label of the 10-years bond is selected as *Y120M* (*Y10y*), then the option *Month* (*Year*) must be selected;

- Factor labels: a string-list based which contains the labels of all the variables present in the model. This input is particularly useful for constructing the plots. The routine `LabFac()` provides the desired list of factor labels;

- Stationarity constraint under the Q-dynamics: a binary variable which takes value equal to 1 if the user wishes the largest eigenvalue under the Q-measure to be strictly smaller than 1. Otherwise, this same variable must be set to 0;

- Output label: a single element string-vector which contains the name that appears in the file name that stores the model outputs.

One possible example of the basic model inputs is

```
R> ModelType <- "JPS"
R> DataFrequency <- "Monthly"
R> Economies <- c("China", "Brazil", "Mexico", "Uruguay")
R> N <- 3
R> GlobalVar <- c("GBC", "CPI_OECD")
R> DomVar <- c("Eco_Act", "Inflation")
R> mat <- c(0.25, 0.5, 1, 3, 5, 10)
R> FactorLabels <- LabFac(N, DomVar,GlobalVar, Economies, ModelType)
R> StationarityUnderQ <- 0
R> OutputLabel <- "Model_demo"
```

5.2. GVARinputs and JLLinputs

Some additional inputs are required if the user intends to estimate the GVAR-ATSM or JLL-ATSM related setups. For a convenient use, the extra outputs should be stored in a separate list for each model. The GVAR model list of inputs must contain:

- Economies: a string-vector containing the names of the economies as described above;

- GVAR list of risk factors: a list of risk factors sorted by country in addition to the global variables. See `DatabasePrep()`;
- VARX type: a string-vector containing the desired estimated form of the VARX*. Two possibilities are available. The *unconstrained* form estimates the model without any constrains, using standard OLS regressions for each one of the model equations. The *constrained: Spanned Factors* form prevents foreign-spanned-factors to impact any domestic risk factor in the feedback matrix, whereas *constrained: '* followed by the name of the risk factor restricts this same factor to be influenced only by its own lagged values and the lagged values of its own star variables. In the last two cases, the VARX* is estimated by restricted least squares.

```
R> GVARinputs <- list()
R> GVARinputs$Economies <- Economies
R> GVARinputs$GVARFactors <- FactorsGVAR
R> GVARinputs$VARXtype <- "constrained: Inflation"
```

Furthermore, the GVAR estimation requires the specification of the necessary inputs to build the transition matrix (see Section 6.2 for more details). One example of this list of inputs is

```
R> t_First <- "2000"
R> t_Last <-  "2015"
R> W_type <- 'Sample Mean'
```

Concerning the JLL-ATSM frameworks, the list of inputs includes:

- JLL type: a string-vector containing the label of the JLL model to be estimated (as described in Table 1);
- Economies: a string-vector containing the names of the economies as described above;
- Dominant Country: a string-vector containing the name of the economy which is assigned as the dominant country (applicable for the *JLL original* and *JLL jointSigma* models) or *None* (applicable for the *JLL NoDomUnit*).
- Wish the estimation of the sigmas matrices: this is a binary variable which assumes value equal to 1 if the user wishes the estimation of all JLL sigma matrices (*i.e.*, variance-covariance and the Cholesky factorization matrices) and, 0 otherwise. As the sigma matrices must be evaluated numerically, the estimation process can take several minutes;
- Sigma of the non-orthogonalized variance-covariance matrix: to save time, the user may provide the variance-covariance matrix from the non-orthogonalized dynamics. Otherwise, this input should be set as *NULL*.

One possible composition for the *JLLinputs* is as follows:

```
R>   JLLinputs <- list()
R>   ModelType <- "JLL original"
R>   JLLinputs$Economies <- Economies
```

```
R>    JLLinputs$DomUnit <- "China"
R>    JLLinputs$WishSigmas <- 1
R>    JLLinputs$SigmaNonOrtho <- NULL
R>    JLLinputs$JLLModelType <- ModelType
```

5.3. Additional inputs for numerical and graphical outputs

Once the model parameters from the ATSM have been estimated, the **MultiATSM** package enables the numerical and graphical compilation of the following additional outputs:

- model fit of the bond yields;
- orthogonalized impulse response functions (IRFs);
- orthogonalized forecast error variance decompositions (FEVDs);
- generalized impulse response functions (GIRFs);
- generalized forecast error variance decompositions (GFEVDs).

The methods used in the computation of the orthogonalized and generalized outputs are described in Pesaran and Shin (1998).

Furthermore, for the IRFs, GIRFs, FEVDs or GFEVDs, a horizon of analysis has to be specified, *e.g.*,:

```
R> Horiz <- 100
```

For the graphical outputs, the user must indicate the desired types of graphs in a string-based vector. Available options are: *Fit*, *IRF*, *FEVD*, *GIRF*, *GFEVD*. One example is

```
R> DesiredGraphs <- c("Fit", "GIRF", "GFEVD")
```

Moreover, the user must select the types of variables of interest (yields, risk factors or both) and, for the JLL type of models, whether the orthogonalized version should be additionally included. These variables take value 1 if the graphs of interest are to be generated and 0, otherwise.

```
R> WishGraphRiskFac <- 0
R> WishGraphYields <- 1
R> WishOrthoJLLgraphs <- 0
```

The function `InputsForOutputs()` can provide some guidance for customizing the features of the wished outputs. Conditional to these settings, individual folders are created at the user's temporary directory to store the different types of the desired graphical outputs.

Bootstrap settings

If the user intends to generate confidence intervals through some bootstrap procedure, an additional list of inputs is required. Specifically:

- *methodBS*: the desired bootstrap procedure. Available options are *(i)* standard residual bootstrap (*bs*); *(ii)* wild bootstrap (*wild*) and block bootstrap (*block*). If the latest is selected, then the block length must be indicated;
- *ndraws*: the number of bootstrap draws;
- *pctg*: the confidence level expressed in percentage points.

```
R> Bootlist <- list()
R> Bootlist$methodBS <- 'block'
R> Bootlist$BlockLength <- 4
R> Bootlist$ndraws <-  3
R> Bootlist$pctg <-  95
```

Out-of-sample forecast settings

To perform out-of-sample forecasts, the following list-based features have to be detailed:

- *ForHoriz*: number of periods-ahead that the forecasts are to be generated;
- *t0Sample*: time-dimension index of the first observations belonging to the information set;
- *t0Forecast*: time-dimension index of the last observation of the information set used to perform the first forecast set.

```
R> ForecastList <- list()
R> ForecastList$ForHoriz <- 12
R> ForecastList$t0Sample <- 1
R> ForecastList$t0Forecast <- 70
```

6. Model estimation

Having gathered bond yields and other economic time series data, the estimation of the ATSM of interest is carried out in three steps. Firstly, one needs to provide the number of country-specific spanned factors to be included in the global ATSM. Secondly, it is decided on the form of the risk factor dynamics under the P-measure. Finally, the user needs to select general features for the model optimization.

6.1. Spanned Factors

The spanned factors are yield-related variables that are used to fit the cross-section dimensions of the term structures. Typically, the spanned factors of country i, $P_{i,t}$, are computed as the N first principal components (PCs) of the set of observed bond yields. Formally, $P_{i,t}$ is constructed as $P_{i,t} = w_i Y_{i,t}$ where w_i is the PC weight matrix and $Y_{i,t}$ is a country-specific column-vector of yields with increasing maturities.

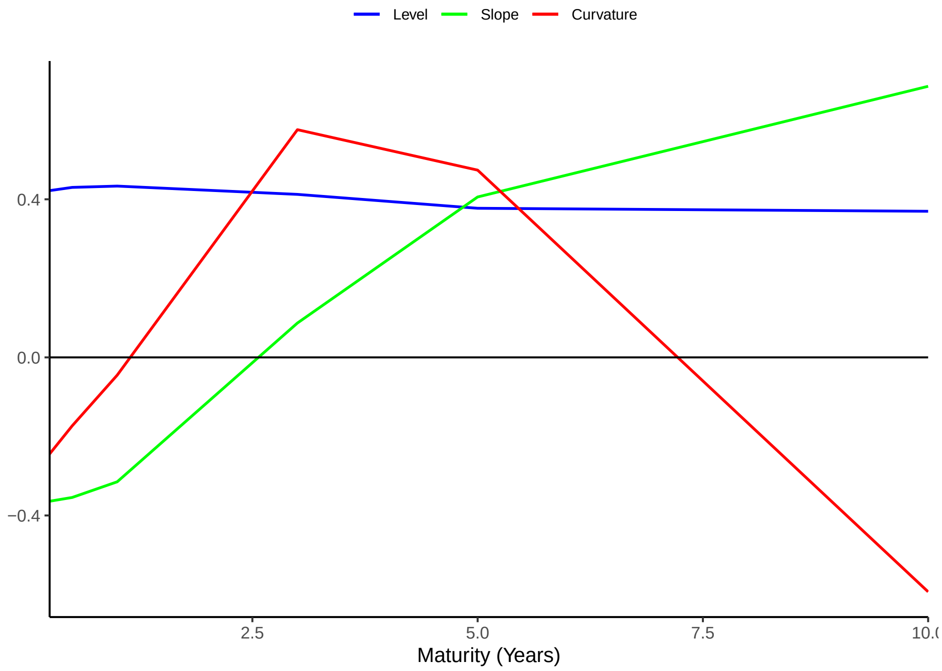

Figure 1: Yield loading on the spanned factors.

For $N = 3$, the spanned factors are traditionally interpreted as level, slope, and curvature. The nature of such interpretability results from the features of the PC weight matrix as illustrated below:[10]

```
R> w <- pca_weights_one_country(Yields, Economy = "Uruguay")
```

In matrix w, each row stores the weights used for constructing each spanned factor. The entries of the first row are linked to the composition of the level factor in that they load rather equally on all yields. Accordingly, high (low) values of the level factor indicate an overall high (low) value of yields across all maturities. In the second row, the weights monotonically increase with the maturities and, therefore, they capture the degree of steepness (slope) of the term structure. High slope factor values imply a steep yield curve, whereas low ones entail flat (or, possibly, downward) curves. In the third row, the weights of the curvature factor are presented. The name of this factor follows from the fact that the weights have a more pronounced effect on the middle range maturities of the curve. These concepts are graphically illustrated in Figure 1.

To directly obtain the time series of the country-specific spanned factors, the user can simply use the `Spanned_Factors()` as follows:

```
R> data('CM_Yields')
R> Economies <- c("China", "Brazil", "Mexico", "Uruguay")
R> N <- 2
R> SpaFact <- Spanned_Factors(Yields, Economies, N)
```

[10] See the seminal work of Litterman and Scheinkman (1991) for a supplementary discussion.

6.2. The P-dynamics estimation

As presented in Table 1, the dynamics of the risk factors under the P-measure evolves according to a VAR(1) model, which may be fully unrestricted (as in the case of the JPS-related models) or somewhat restricted (as in the GVAR and JLL frameworks). Below, the usage of each one of these model dimensions are illustrated.

VAR

Using the `VAR()` function of this package requires simply selecting the appropriate set of risk factors for the desired estimated model. For instance, for the models *JPS jointP* and *VAR jointQ*, the estimation of the P-dynamics parameters is obtained as

```
R> data("CM_Factors")
R> PdynPara <- VAR(RiskFactors, VARtype= "unconstrained")
```

whereas the estimation of a *JPS* model for China is

```
R> FactorsChina <- RiskFactors[1:7,]
R> PdynPara <- VAR(FactorsChina, VARtype= "unconstrained")
```

In both cases, the outputs generated are the vector of intercepts in addition to the feedback and the variance-covariance matrices.

GVAR

As exposed in Section 2.2, the estimation of a GVAR model requires defining a measure of interdependence among the countries of the economic system. This information is reported in the transition matrix, the entries of which reflect the degree of interconnections of two entities of this same economic system. In this package, the illustration of the transition matrix is based on the average of the cross-border trade flow weights for the period spanning the years from 2006 to 2019. Note that each row sums up to 1.

```
R> data("CM_Trade")
R> t_First <- "2006"
R> t_Last <- "2019"
R> Economies <- c("China", "Brazil", "Mexico", "Uruguay")
R> type <- "Sample Mean"
R> W_gvar <- Transition_Matrix(t_First, t_Last, Economies, type,
+                      DataPath = NULL, TradeFlows)
R> round(W_gvar, digits= 4)

         China Brazil Mexico Uruguay
China   0.0000 0.6549 0.3155  0.0296
Brazil  0.8269 0.0000 0.1234  0.0497
Mexico  0.8596 0.1326 0.0000  0.0078
Uruguay 0.3811 0.5498 0.0691  0.0000
```

Having defined the form of the transition matrix, one can complete the *GVARinputs* variable along the lines discussed in Section 5.2.

```
R> data("CM_Factors_GVAR")
R> GVARinputs <- list()
R> GVARinputs$Economies <- Economies
R> GVARinputs$GVARFactors <- FactorsGVAR
R> GVARinputs$VARXtype <- "unconstrained"
R> GVARinputs$Wgvar <- W_gvar
R> N <- 3
R> GVARpara <- GVAR(GVARinputs, N)
```

A separate routine is provided for computing the star variables used in the estimation of the VARX* models.

```
R> data('CM_Factors')
R> StaFac <- StarFactors(RiskFactors, Economies, W_gvar)
```

JLL

Calculating the P-dynamics parameters in the form proposed by JLL (2015) requires the following inputs: *(i)* the time series of the risk factors in non-orthogonalized form; *(ii)* the number of country-specific spanned factors, and *(iii)* the specification of the *JLLinputs* as presented in Section 5.2. See, for instance:

```
R> data("CM_Factors")
R> N <- 3
R> JLLinputs <- list()
R> ModelType <- "JLL original"
R> JLLinputs$Economies <- Economies
R> JLLinputs$DomUnit <- "China"
R> JLLinputs$WishSigmas <- 1
R> JLLinputs$SigmaNonOrtho <- NULL
R> JLLinputs$JLLModelType <- ModelType
R> JLLpara <- JLL(RiskFactors, N, JLLinputs)
```

6.3. ATSM estimation

The model estimation involves defining the model inputs from the log-likelihood function (LLK) and, subsequently, the parameters used in its optimization process. The structure proposed in this part of the code is in a great extent based on the term structure package by Le and Singleton (2018).

The log-likelihood function

For the sake of simplicity, we illustrate the construction of a LLK based on a *VAR jointQ* model. For both the JLL and GVAR-based models, one must further specify the *JLLinputs* or *GVARinputs* along the lines described in the Section 5.2. We divide the exposition in four parts:

1. Inputs to be specified by the user (see Section 5)

```
R> data("CM_Yields")
R> data("CM_Factors")
R> ModelType <- "VAR jointQ"
R> Yields <- Yields
R> ZZ <- RiskFactors
R> Economies <- c("China", "Brazil", "Mexico", "Uruguay")
R> mat <-  Maturities(Yields, Economies, UnitYields = "Month")
R> DataFrequency <- "Monthly"
R> GlobalVar <- c("GBC", "CPI_OECD")
R> DomVar <- c("Eco_Act", "Inflation")
R> N <- 3
```

2. Generate the *Factor Labels* list

```
R> FactorLabels <- LabFac(N, DomVar, GlobalVar, Economies, ModelType)
```

3. Prepare the inputs for LLK

```
R> ATSMInputs <- InputsForMLEdensity(ModelType, Yields, ZZ, FactorLabels,
+                                    mat, Economies, DataFrequency)
```

4. Set the objective function

```
R> f <- Functionf(ATSMInputs, Economies, mat,
+                 DataFrequency, FactorLabels, ModelType)
```

Optimization parameters

JPS (2014) requires the estimation of a set of parameters containing 6 elements, namely: the risk-neutral long-run mean of the short rate ($r0$), the risk-neutral feedback matrix ($K1XQ$), the variance-covariance matrix (SSZ) from the VAR processes, the standard deviation of the errors from the portfolios of yields observed with error (se), in addition to the intercept ($K0Z$) and the feedback ($K1Z$) matrices of the physical dynamics. Each one of these parameters must be cast in an individual list that should contain the *(i)* the starting value of the parameter (if any); *(ii)* the variable label followed by a ':' and a type of constraint; *(iii)* a lower bound (if any), and *(iv)* an upper bound (if any). The variable labels of $r0$, se, $K0Z$ and $K1Z$ should be preceded by the symbol @ as a manner to account for the fact that the solution of these parameters are known in closed-form. For the remaining ones ($K1XQ$ and SSZ), Le and Singleton (2018) provide standardized routines (already incorporated in this package) to set good initial values so that the optimization process runs faster.

The type of constraint for the parameters $r0$, se, $K0Z$, and $K1Z$ is typically *bounded* and are used for bounded matrices. For $K1XQ$ and SSZ, the **MultiATSM** package currently provides the following options:

- $K1XQ$: *Jordan* (or *Jordan MultiCountry*) for a matrix of Jordan type for single (multi) country specifications. These labels can be extended by *'; stationary'*, if one wishes to impose the largest eigenvalue of the risk-neutral feedback matrix to be strictly smaller than 1;

- *SSZ*: *psd* for a positive semi-definite matrix; *diag* for a diagonal matrix; *BlockDiag* for a block diagonal matrix (typical from the GVAR-based models) and *JLLstructure* for the models containing the restrictions along the lines of JLL (2015).

One example for the list of parameters specification is

```
R> K1XQinputs <- list(NULL, "K1XQ: Jordan", NULL , NULL)
R> SSZinputs <- list(NULL, "SSZ: psd", NULL, NULL)
R> r0inputs <- list(NULL, "@r0: bounded", NULL, NULL)
R> seinputs <- list(NULL, "@se: bounded", 1e-6, NULL)
R> K0Zinputs <- list(NULL, "@K0Z: bounded", NULL, NULL)
R> K1Zinputs <- list(NULL, "@K1Z: bounded", NULL, NULL)
```

To complete the optimization setting, the user needs to specify the level of convergence tolerance (usually, $1e-4$ is a reasonable value) and the vector *OptRun*. The first element of this vector must be filled by the word *iter off*, if the user wishes to switch off the printouts of the numerical optimization routines, or simply by *iter* otherwise. The second element of the vector concerns the algorithm that is used in the optimization: the option *fminunc only* only uses `fminunc()`, whereas *fminsearch only* only applies `fminsearch()`. If *OptRun* is formed exclusively by *iter off*, then both `fminunc()` and `fminsearch()` are employed in the optimization.

7. Example of full implementation of ATSMs

This section presents an example on how to use the **MultiATSM** package to fully implement ATSMs. We use the *JPS* framework to illustrate the example. The implementation steps are described below.

1. Defining user inputs along the lines presented in Section 5

   (a) Load database data

   ```
   R> data("CM_Factors")
   R> data('CM_Factors_GVAR')
   R> data('CM_Trade')
   R> data('CM_Yields')
   ```

   (b) Decide on general model inputs

   ```
   R> ModelType <- "JPS"
   R> StationarityUnderQ <- 0
   R> Economies <- c("China", "Brazil", "Mexico", "Uruguay")
   R> GlobalVar <- c("GBC", "CPI_OECD")
   R> DomVar <- c("Eco_Act", "Inflation")
   R> N <- 3
   R> OutputLabel <- "Test"
   R> DataFrequency <- "Monthly"
   R> UnitMatYields <- "Month"
   ```

i. Decide on specific model inputs

A. GVAR-based models (transition matrix and VARX inputs)

```
R> if (ModelType == 'GVAR sepQ' ||
+       ModelType == 'GVAR jointQ'){
+     t_First <- "2006"
+     t_Last <- "2019"
+     W_type <- 'Sample Mean'
+     VARXtype <- "unconstrained"
+  }
```

B. JLL-based models

```
R> if (ModelType =="JLL original" ||
+       ModelType == "JLL NoDomUnit" ||
+       ModelType == "JLL jointSigma"){
+     DominantCountry <- "China" }
```

(c) Decide on Settings for numerical outputs

```
R> Horiz <- 20
R> DesiredGraphs <- c("IRF", "FEVD")
R> WishGraphRiskFac <- 0
R> WishGraphYields <- 1
R> WishOrthoJLLgraphs <- 0
```

(d) Bootstrap settings

```
R> WishBootstrap <- 1
R> Bootlist <- list()
R> Bootlist$methodBS <- 'wild'
R> Bootlist$BlockLength <- 4
R> Bootlist$ndraws <-  30
R> Bootlist$pctg <-  95
```

(e) Out-of-sample forecast features

```
R> WishForecast <- 1
R> ForecastList <- list()
R> ForecastList$ForHoriz <- 12
R> ForecastList$t0Sample <- 1
R> ForecastList$t0Forecast <- 145
```

2. Minor preliminary work. From this point of the code onward, no changes are needed from the user

```
R> C <- length(Economies)
R> FactorLabels <- LabFac(N, DomVar,GlobalVar, Economies, ModelType)
R> mat <- Maturities(Yields, Economies, UnitYields = UnitMatYields)
R> ZZ <- RiskFactors
```

(a) Build the GVARinputs

```
R> if (ModelType == 'GVAR sepQ' || ModelType == 'GVAR jointQ'){
+     GVARinputs <- list()
```

```
+    GVARinputs$Economies <- Economies
+    GVARinputs$GVARFactors <- FactorsGVAR
+    GVARinputs$VARXtype <- VARXtype
+    GVARinputs$Wgvar <- Transition_Matrix(t_First, t_Last, Economies,
+                                          W_type, DataPath = NULL,
+                                          Data = TradeFlows)
+  } else { GVARinputs <- NULL }
```

(b) Build the JLLinputs

```
R> if (ModelType =="JLL original" || ModelType == "JLL NoDomUnit"
+      || ModelType == "JLL jointSigma"){
+    JLLinputs <- list()
+    JLLinputs$Economies <- Economies
+    JLLinputs$DomUnit <- DominantCountry
+    JLLinputs$WishSigmas <- 1
+    JLLinputs$SigmaNonOrtho <- NULL
+    JLLinputs$JLLModelType <- ModelType
+  }else{
+    JLLinputs <- NULL
+  }
```

3. Obtaining initial guesses for $K1XQ$ and SSZ, build the objective function and perform the model optimization according to the procedure described in Section 6

```
R> ModelParaList <- list()
R> for (i in 1:C){
+    if (( ModelType == "GVAR jointQ" || ModelType == "VAR jointQ"
+          || ModelType == "JLL original" || ModelType == "JLL NoDomUnit"
+          || ModelType == "JLL jointSigma" )   & i >1 ){break}
+
+    ATSMInputs <- InputsForMLEdensity(ModelType, Yields, ZZ, FactorLabels,
+                                      mat, Economies, DataFrequency,
+                                      JLLinputs, GVARinputs)
+
+    K1XQ <- ATSMInputs$K1XQ
+    if (ModelType == "JLL original" || ModelType == "JLL NoDomUnit" ){
+      SSZ <- NULL} else {SSZ <- ATSMInputs$SSZ}
+
+    f <- Functionf(ATSMInputs, Economies, mat, DataFrequency,
+                   FactorLabels, ModelType)
+
+    VarLab <- ParaLabels(ModelType, StationarityUnderQ)
+    varargin <- list()
+    varargin$K1XQ <- list(K1XQ, VarLab[[ModelType]][["K1XQ"]], NULL, NULL)
+    varargin$SSZ <- list(SSZ, VarLab[[ModelType]][["SSZ"]], NULL, NULL)
+    varargin$r0 <- list(NULL, VarLab[[ModelType]][["r0"]], NULL, NULL)
+    varargin$se <- list(NULL, VarLab[[ModelType]][["se"]], 1e-6, NULL)
+    varargin$K0Z <- list(NULL, VarLab[[ModelType]][["K0Z"]], NULL, NULL)
```

```
+    varargin$K1Z <- list(NULL, VarLab[[ModelType]][["K1Z"]], NULL, NULL)
+    varargin$OptRun <- c("iter off")
+
+    LabelVar <- c('Value', 'Label', 'LB', 'UB')
+    for (d in 1:(length(varargin)-1)){ names(varargin[[d]]) <-  LabelVar}
+    tol <- 1e-4
+
+    if (ModelType == 'JPS' || ModelType == 'JPS jointP' ||
+        ModelType == "GVAR sepQ"){
+  ModelParaList[[ModelType]][[Economies[i]]] <- Optimization(f, tol,
+                                                   varargin, FactorLabels,
+                                                   Economies, ModelType,
+                                                  JLLinputs,
+                                                  GVARinputs)$Summary
+    }else{ ModelParaList[[ModelType]] <- Optimization(f, tol, varargin,
+                                                    FactorLabels,
+                                                    Economies, ModelType,
+                                                    JLLinputs,
+                                                    GVARinputs)$Summary}
+
+  }
```

4. Generate the numerical and graphical outputs

```
R> InputsForOutputs <- InputsForOutputs(ModelType, Horiz, DesiredGraphs,
+                                      OutputLabel, StationarityUnderQ,
+                                      WishGraphYields, WishGraphRiskFac,
+                                      WishOrthoJLLgraphs, WishBootstrap,
+                                      Bootlist, WishForecast, ForecastList)
```

   (a) Fit, IRFs, FEVDs, GIRFs and GFEVDs

```
R> NumericalOutputs <- NumOutputs(ModelType, ModelParaList,
+                                 InputsForOutputs,
+                                 FactorLabels, Economies)
```

   (b) Bootstrap

```
R> Bootstrap <- Bootstrap(ModelType, ModelParaList, NumericalOutputs,
+                         mat, Economies, InputsForOutputs,
+                         FactorLabels, DataFrequency, varargin,
+                         JLLinputs, GVARinputs)
```

   (c) Out-of-sample forecasting

```
R> Forecasts <- ForecastYields(ModelType, ModelParaList,
+                              InputsForOutputs, FactorLabels,
+                              Economies, DataFrequency, JLLinputs,
+                              GVARinputs)
```

Out of illustration, we present some of the model graphical outputs in Figures 2 and 3. In particular, the former shows the IRFs for Brazilian bonds to a global economic growth shock and the latter exhibits the FEVD of the 3 month bond yield in China.

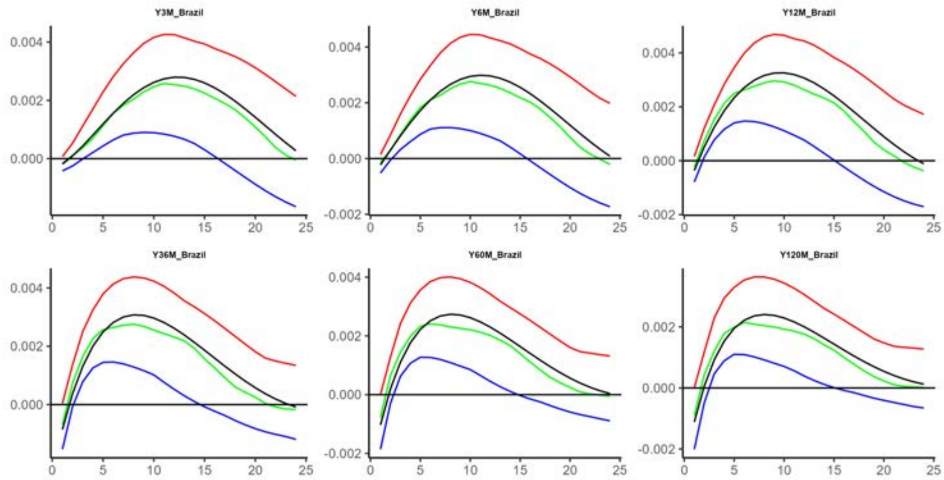

Figure 2: IRFs from the Brazilian bond yields to global economic growth shock. Maturity of bond yields are 3, 6, 12, 36, 60, and 120 months. Size of the shock is one-standard deviation. Red, green and blue lines are, respectively the upper bound, median, and the lower bounds of the 95% confidence intervals. The black lines are the point estimate. The dataset is the same used in Candelon and Moura (2021).

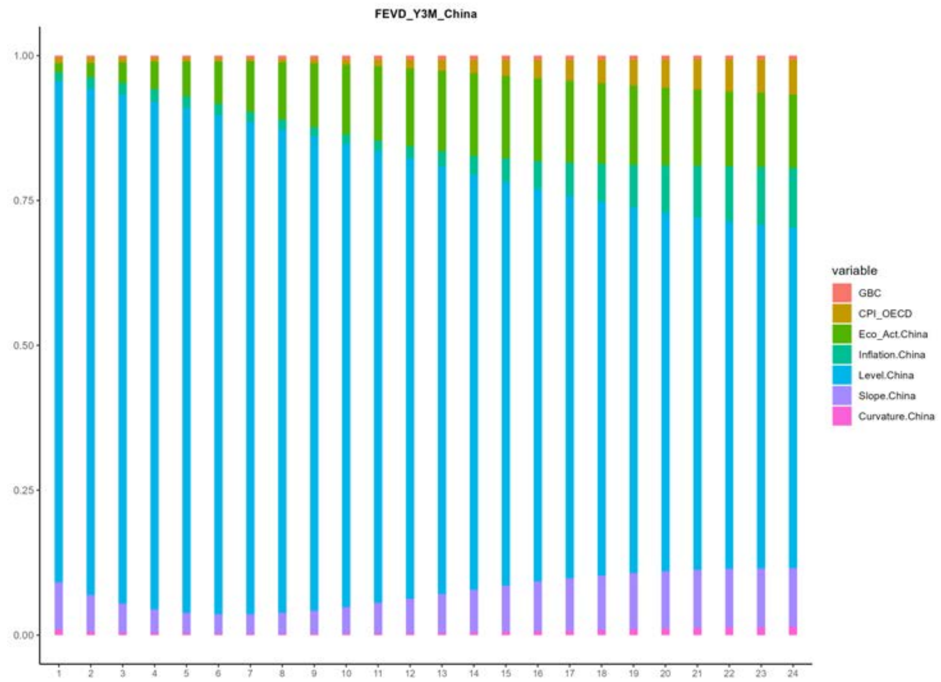

Figure 3: FEVD from the Chinese bond yield with maturity 3 months. The dataset is the same used in Candelon and Moura (2021).

Acknowledgments

The author is indebted to Bertrand Candelon for many helpful discussions. The author is responsible for the remaining errors.

References

Ang A, Piazzesi M (2003). "A No-Arbitrage Vector Autoregression of Term Structure Dynamics with Macroeconomic and Latent Variables." *Journal of Monetary Economics*, **50**(4), 745–787.

Bauer MD, Rudebusch GD (2017). "Resolving the Spanning Puzzle in Macro-Finance Term Structure Models." *Review of Finance*, **21**(2), 511–553.

Candelon B, Moura R (2021). "A Multicountry Model of the Term Structures of Interest Rates with a GVAR."

Chudik A, Pesaran MH (2016). "Theory and Practice of GVAR Modelling." *Journal of Economic Surveys*, **30**(1), 165–197.

Dai Q, Singleton KJ (2000). "Specification Analysis of Affine Term Structure Models." *Journal of Finance*, **55**(5), 1943–1978.

Dai Q, Singleton KJ (2002). "Expectation Puzzles, Time-Varying Risk Premia, and Affine Models of the Term Structure." *Journal of financial Economics*, **63**(3), 415–441.

Duffie D, Kan R (1996). "A Yield-Factor Model of Interest Rates." *Mathematical finance*, **6**(4), 379–406.

Guirreri SS (2015). ***YieldCurve**: Modelling and estimation of the yield curve*. R package version 4.1, URL https://CRAN.R-project.org/package=YieldCurve.

Gürkaynak RS, Wright JH (2012). "Macroeconomics and the term structure." *Journal of Economic Literature*, **50**(2), 331–67.

Hasse JB, Lajaunie Q (2021). ***EWS**: Early Warning System*. R package version 0.2.0, URL https://CRAN.R-project.org/package=EWS.

Joslin S, Priebsch M, Singleton KJ (2014). "Risk Premiums in Dynamic Term Structure Models with Unspanned Macro Risks." *Journal of Finance*, **69**(3), 1197–1233.

Joslin S, Singleton KJ, Zhu H (2011). "A New Perspective on Gaussian Dynamic Term Structure Models." *Review of Financial Studies*, **24**(3), 926–970.

Jotikasthira C, Le A, Lundblad C (2015). "Why Do Term Structures in Different Currencies Co-move?" *Journal of Financial Economics*, **115**, 58–83.

Le A, Singleton K (2018). "A Small Package of Matlab Routines for the Estimation of Some Term Structure Models."

Litterman R, Scheinkman J (1991). "Common Factors Affecting Bond Returns." *Journal of Fixed Income*, **1**, 54–61.

Moura R (2022). ***MultiATSM**: Multicountry Term Structure of Interest Rates Models*. R package version 0.1-1, URL https://CRAN.R-project.org/package=MultiATSM.

Nelson CR, Siegel AF (1987). "Parsimonious Modeling of Yield Curves." *Journal of business*, pp. 473–489.

Pesaran HH, Shin Y (1998). "Generalized Impulse Response Analysis in Linear Multivariate Models." *Economics letters*, **58**(1), 17–29.

Piazzesi M (2010). "Affine Term Structure Models." In *Handbook of financial econometrics: Tools and Techniques*, pp. 691–766. Elsevier.

Rudebusch GD, Wu T (2008). "A Macro-Finance Model of the Term Structure, Monetary Policy and the Economy." *The Economic Journal*, **118**(530), 906–926.

Setz T (2017). ***fBonds**: Rmetrics - Pricing and Evaluating Bonds*. R package version 3042.78, URL https://CRAN.R-project.org/package=fBonds.

Svensson LE (1994). "Estimating and Interpreting Forward Interest Rates: Sweden 1992-1994."

Vasicek O (1977). "An Equilibrium Characterization of the Term Structure." *Journal of financial economics*, **5**(2), 177–188.

A. Paper replications

In this section, we use the **MultiATSM** package to replicate some of the empirical results presented in the original papers of JPS (2014) and CM (2021).

A.1. Joslin, Priebisch and Singleton (2014)

To the best of our knowledge, the original dataset used in Joslin *et al.* (2014) is not publicly available. Nevertheless, there exists an alternative source of data that closely matches JPS (2014) in terms of the construction method and the sample period. This database was constructed by Bauer and Rudebusch (2017) (henceforth BR, 2017) and is available at Bauer's website (`https://www.michaeldbauer.com/publication/spanning-puzzle/`). In that paper, BR (2017) investigate whether macrofinance term structure models better suit the unspanned macro risk framework by JPS (2014) or the earlier traditional spanned settings as the one by Ang and Piazzesi (2003). Accordingly, BR (2017) intend to replicate some of the empirical results reported in JPS (2014). The R-code used by BR (2017) is also available at Bauer's webpage.

Supported by BR (2017)'s dataset, the code below uses the **MultiATSM** package to estimate the key model parameters from the ATSM of JPS (2014).

1. Load database data and set the model inputs

```
R> data("BR_jps_gro_R3")
R> ModelType <- "JPS"
R> StationarityUnderQ <- 0
R> Economies <- c("U.S.")
R> GlobalVar <- c()
R> DomVar <- c("GRO", "INF")
R> N <- 3
R> DataFrequency <- "Monthly"
R> UnitMatYields <- "Month"
R> mat <- c(0.25, 0.5, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10)
```

2. Minor preliminary work

```
R> C <- length(Economies)
R> M <- length(DomVar)
R> FactorLabels <- LabFac(N, DomVar, GlobalVar, Economies, ModelType)
R> Yields <- t(BR_jps_out$Y)
R> W <- BR_jps_out$W[1:N,]
R> SpaFac <- W%*%Yields
R> rownames(SpaFac) <- FactorLabels$Tables$U.S.[-(1:M)]
R> ZZ <- rbind(t(BR_jps_out$M.o), SpaFac)
```

3. Build the objective function and perform the model optimization

```
R> for (i in 1:C){
+  ATSMInputs <- InputsForMLEdensity(ModelType, Yields, ZZ, FactorLabels,
+                                    mat, Economies, DataFrequency)
```

```
+  K1XQ <- ATSMInputs$K1XQ
+  SSZ <- ATSMInputs$SSZ
+  ATSMInputs$Wpca <- W
+  ATSMInputs$We <- t(pracma::null(W))
+  ATSMInputs$WpcaFull <- rbind(ATSMInputs$Wpca, ATSMInputs$We)
+  ATSMInputs$PP <- SpaFac
+
+  f <- Functionf(ATSMInputs, Economies, mat, DataFrequency,
+                 FactorLabels, ModelType)
+
+  VarLab <- ParaLabels(ModelType, StationarityUnderQ)
+
+  varargin <- list()
+  varargin$K1XQ <- list(K1XQ, VarLab[[ModelType]][["K1XQ"]], NULL, NULL)
+  varargin$SSZ <- list(SSZ, VarLab[[ModelType]][["SSZ"]], NULL, NULL)
+  varargin$r0 <- list(NULL, VarLab[[ModelType]][["r0"]], NULL, NULL)
+  varargin$se <- list(NULL, VarLab[[ModelType]][["se"]], 1e-6, NULL)
+  varargin$K0Z <- list(NULL, VarLab[[ModelType]][["K0Z"]], NULL, NULL)
+  varargin$K1Z <- list(NULL, VarLab[[ModelType]][["K1Z"]], NULL, NULL)
+  varargin$OptRun <-  c("iter off")
+
+  LabelVar <- c('Value', 'Label', 'LB', 'UB')
+  for (d in 1:(length(varargin)-1)){ names(varargin[[d]]) <-  LabelVar}
+  tol <- 1e-4
+
+  ModelPara <- Optimization(f, tol, varargin, FactorLabels,
+                            Economies, ModelType)$Summary
+  }
```

The tables below compare the ATSM parameter estimates generated from BR (2017) and the **MultiATSM**. Overall, one can note that the differences in the estimates are economically modest.

	MultiATSM	BR (2017)
$r0$	0.00055	-0.00016
λ_1	0.99670	0.99682
λ_2	0.91488	0.95945
λ_3	0.91488	0.87174

Table 2: Q-dynamics parameters. λ's are the eigenvalues from the risk-neutral feedback matrix and $r0$ is the long-run mean of the short rate under Q.

It is relevant to highlight that the script above makes use of the principal component weights provided by BR (2017). Such a matrix is simply a scaled-up version of the one provided by the function `pca_weights_one_country()`. Accordingly, despite the numerical differences on the weight matrices, both methods generate time series of spanned factors which are perfectly correlated. Another difference between the two approaches relates to the construction form of the LLK: while in the BR (2017) code the LLK is expressed in terms of a portfolio of yields,

	$K0Z$	$K1Z$				
		PC1	PC2	PC3	GRO	INF
BR (2017)						
PC1	0.07811	0.93691	-0.01307	-0.02181	0.10457	0.10033
PC2	0.02100	0.00582	0.97814	0.17031	-0.16719	-0.04016
PC3	0.10047	-0.01037	-0.00625	0.78346	-0.03987	0.04369
GRO	0.06904	-0.00483	0.01801	-0.11117	0.88177	-0.00247
INF	0.04996	0.00185	0.00642	-0.05920	0.02767	0.98593
MultiATSM						
PC1	0.07811	0.93691	-0.01307	-0.02181	0.10457	0.10033
PC2	0.02100	0.00582	0.97814	0.17031	-0.16719	-0.04016
PC3	0.10047	-0.01037	-0.00625	0.78346	-0.03987	0.04369
GRO	0.06904	-0.00483	0.01801	-0.11117	0.88177	-0.00247
INF	0.04996	0.00185	0.00642	-0.05920	0.02767	0.98593

Table 3: P-dynamics parameters. $K0Z$ is the intercept and $K1Z$ is feedback matrix from the P-dynamics.

	MultiATSM	BR (2017)
se	0.0000546	0.0000550

Table 4: Portfolio of yields with errors. *se* is the standard deviation of the portfolio of yields observed with errors.

the **MultiATSM** package generates this same input directly as a function of observed yields (*i.e.*, both procedures lead to equivalent LLK up to the Jacobian term).

A.2. Candelon and Moura (2021)

We now provide the lines of code that replicate the empirical results presented in Candelon and Moura (2021). Specifically, setting *ModelType* as *GVAR jointQ* generates the outputs of the model labeled as *GVAR-ATSM*, whereas *JLL original* builds the results for the *JLL-ATSM*. To save space, we report only the overall model features outlined in the step 1 of Section 7. The remaining part of the code is identical to the steps 2 through 4 described in this same section.

1. Load database data and the general and specific model inputs:

```
R> data("CM_Factors")
R> data('CM_Factors_GVAR')
R> data('CM_Trade')
R> data('CM_Yields')
R> ModelType <- "GVAR jointQ"
R> StationarityUnderQ <- 0
R> Economies <- c("China", "Brazil", "Mexico", "Uruguay")
R> GlobalVar <- c("GBC", "CPI_OECD")
R> DomVar <- c("Eco_Act", "Inflation")
```

```
R> N <- 3
R> OutputLabel <- "CM_2021"
R> DataFrequency <- "Monthly"
R> UnitMatYields <- "Month"
R> if (ModelType == 'GVAR jointQ'){
+  t_First <- "2006"
+  t_Last <-  "2019"
+  W_type <- 'Sample Mean'
+  VARXtype <- "unconstrained"
+  }
R> if (ModelType =="JLL original"){DominantCountry <- "China"}
```

2. Defining the settings of the numerical outputs, bootstrap, and the out-of-sample forecast:

```
R> Horiz <- 25
R> DesiredGraphs <- c("Fit", "GIRF", "GFEVD")
R> WishGraphRiskFac <- 0
R> WishGraphYields <- 1
R> WishOrthoJLLgraphs <- 1
R> WishBootstrap <- 0
R> Bootlist <- list()
R> Bootlist$methodBS <- 'bs'
R> Bootlist$BlockLength <- c()
R> Bootlist$ndraws <-  1000
R> Bootlist$pctg   <-  95
R> WishForecast <- 0
R> ForecastList <- list()
R> ForecastList$ForHoriz <- 12
R> ForecastList$t0Sample <- 1
R> ForecastList$t0Forecast <- 90
```

Affiliation:

Rubens Moura
LFIN/LIDAM
UCLouvain
Voie du Roman Pays, 34
1348 Louvain-la-Neuve, Belgium
E-mail: rubens.guimaraes@uclouvain.com