Zero-Coupon Yield Curve Estimation with the Package termstrc

Robert Ferstl
University of Regensburg

Josef Hayden
University of Regensburg

Abstract

Since zero-coupon rates are rarely directly observable, they have to be estimated from market data. In this paper we review several widely-used parametric term structure estimation methods. We propose a weighted constrained optimization procedure with analytical gradients and a globally optimal start parameter search algorithm. Moreover, we introduce the R package **termstrc**, which offers a wide range of functions for term structure estimation based on static and dynamic coupon bond and yield data sets. It provides extensive summary statistics and plots to compare the results of the different estimation methods. We illustrate the application of the package through practical examples using market data from European government bonds and yields.

Keywords: fixed income, term structure estimation, global optimization, R.

1. Introduction

The term structure of interest rates defines the relationship between the yield of a fixed income investment and the time to maturity of its cash flows. Accordingly, the zero-coupon yield curve provides the relationship for investments with only one payment at maturity. It serves as the basis for the valuation of other fixed income instruments and as an input for various models, e.g., for risk management, monetary policy, derivatives pricing. Although zero-coupon prices can be directly used to construct the term structure, the lack of market liquidity and the limited available maturity spectrum necessitates the estimation based on observed coupon bond prices.

In this paper we briefly review important term structure estimation methods. We focus on the cubic splines approach of McCulloch (1971, 1975) and the Nelson and Siegel (1987) method with extensions by Svensson (1994), Diebold and Li (2006) and De Pooter (2007). According to a survey of Bank for International Settlements (2005) they are widely applied at central

banks. We give a detailed discussion of the model estimation, and propose a nonconvex weighted global optimization procedure with linear constraints, analytical gradients and an appropriate start parameter search algorithm. Moreover, we introduce the package **termstrc** which is written in the R system for statistical computing (R Development Core Team 2010). It is available from the Comprehensive R Archive Network at `http://CRAN.R-project.org/package=termstrc` and from the R-Forge development platform (Theußl and Zeileis 2009) at `http://R-Forge.R-project.org/projects/termstrc/`. The package provides a wide range of functions for estimating and analyzing of the term structure of interest rates. All methods, except the cubic splines approach, can be applied statically and iteratively on dynamic coupon bond price and zero yield data sets.

In the following two subsections we review the relevant literature and the currently available term structure estimation software. In Section 2 we introduce the notation and discuss different models and estimation procedures. The structure of the package **termstrc** (i.e., the available functions, S3 classes and methods) is presented in Section 3. Several examples for the different models based on dynamic/static bond/yield data sets are presented in Section 4. Section 5 concludes the paper.

1.1. Literature review

The first term structure model goes back to Vasicek (1977, Vasicek in the following), in which he assumed that the short rate followed a univariate continuous Markovian diffusion process and all riskless zero-coupon bond prices were functions of the short rate, the current time and the maturity date of the bond. Whereas Vasicek used an Uhlenbeck and Ornstein (1930) process for the short rate, Cox, Ingersoll, and Ross (1985, CIR in the following) proposed a one-dimensional square root diffusion process, which ensured positive short rates. CIR derived their model by applying an equilibrium from utility-maximizing identical individuals with a log-utility function. Nawalkha, Beliaeva, and Soto (2007) classified term structure models, which assume a time-homogeneous short rate process and require and explicit market price of risk specification as *fundamental* models. In addition to the Vasicek and CIR model, the multifactor affine term structure models (ATSMs) of Duffie and Kan (1996) and Dai and Singleton (2000) also belong to the class of fundamental models. To estimate these models a time series of a cross section of fixed income security prices is required. The estimation of multifactor ATSMs can be challenging, especially with nonlinear market price of risk specifications. Moreover, the model implied prices may or may not converge to the observed market prices. However, for the purpose of pricing fixed income derivatives, models which exactly fit the term structure are required. Based on Vasicek and CIR, models were proposed, which assume an endogenous term structure. They price the initially observed zero-bond prices without errors by allowing time inhomogeneity in the parameters of the stochastic differential equation for the short rate. Examples of so-called *endogenous* or *no-arbitrage* models are Ho and Lee (1986), Hull and White (1990) and Black and Karasinski (1991).

Another class of models, which has not been deduced from equilibrium conditions or/and no-arbitrage conditions, takes a more empirical approach. In general these models assume a parametric form of the spot rate, forward rate or discount function. The unknown parameters are then estimated by minimizing the error between theoretical and observed prices of a cross section of coupon bonds at a certain point in time. The method of Fama and Bliss (1987) iteratively extracts the forward rates by extending the discount function at each step of the

calculation, i.e., the forward rates are computed to price bonds with increasing maturity given the discount function fitted to the previous bonds with shorter maturity. The obtained spot or forward rates are referred to as unsmoothed Fama/Bliss yields. The discount function is piecewise linear with the number of parameters equal to the number of included securities. The procedure works well only if all cash flows have the same maturity intervals (see, e.g., Hagan and West 2006). McCulloch (1971, 1975) proposed using splines to fit the discount function of the segmented term structure. Several different types of splines have been suggested as well as the use of penalty functions, e.g., the penalized spline model of Fisher, Nychka, and Zervos (1995). Nelson and Siegel (1987, NS in the following) proposed a four parametric parsimonious function from the family of general exponential-polynomial functions for the forward curve, which was later extended by Svensson (1994, SV in the following). Both specifications make it possible to reproduce a wide range of possible term structure shapes. The estimation of the parameters typically involves a nonconvex optimization procedure and is based on coupon bond prices. However, it also possible to use interpolated zero or Fama/Bliss yields. Several papers compare the performance of empirical term structure estimation methods, (see, e.g., Bliss 1997; Bolder and Streliski 1999; Ioannides 2003). The models of NS and SV are not formulated in a dynamic framework and the first model is not consistent with arbitrage-free pricing theory (see, Björk and Christensen 1999; Filipovic 1999). The first disadvantage was addressed by Diebold and Li (2006), who iteratively applied a simplified NS model (DL in the following) to a dynamic zero yield data set. Subsequently, they estimated common time series models for the parameter time series in order to forecast future zero-coupon yield curves. Christensen, Diebold, and Rudebusch (2007, 2009) tried to correct for the second drawback. They developed two ATSMs which belonged to the class of Dai and Singleton (2000), where the spot rate function is similar to the version of NS and SV.

1.2. Software review

The package **sde** developed by Iacus (2009) provides functions for the parametric estimation and numerical methods for the simulation of univariate stochastic processes. The afore mentioned Vasicek and CIR term structure model assumes that the instantaneous spot rate follows an one-dimensional diffusion process. The former model requires the short rate under the real measure to follow an Uhlenbeck and Ornstein (1930) process with constant coefficients, whereas the latter postulates a square-root diffusion process. The package makes it possible to simulate the short rate based on the underlying differential equation and to estimate the parameters. However, it is not possible to estimate the term structure based on coupon bonds or to price fixed income securities with the offered functions. Moreover, only univariate processes are covered. The **yuima** project is currently developing a more general framework for the simulation and estimation of multivariate stochastic processes. For more details concerning **sde** we refer to Iacus (2009) and concerning the **yuima** project to Iacus (2010).

The existing software, which allows the estimation of the mentioned empirical term structure models can be distinguished according to the type of data used. The R package **YieldCurve** developed by Guirreri (2010) offers functions for the estimation based on zero yields. The available models include NS, SV and DL. The functions require a user-specified range of certain parameters of the spot rate function. All other parameters are subsequently obtained by a linear grid search. Unfortunately, nececessary constraints on the parameters were not considered. The package **fBonds** of Wuertz (2009) requires forward rates for the estimation

and the available methods are NS and SV. In contrast to the **YieldCurve** package, **fBonds** obtains globally optimal start parameters from a grid search and afterwards an unconstrained estimation procedure leads to the optimal parameters. Neither optimization procedures allow the user to specify the step size of the grid. The previous two packages consist of a limited number of highly similar functions which do not allow the estimation based on observed prices of coupon bonds and ignore constraints on the parameters. Moreover, they provide no standardized S3 methods. In contrast, the library **QuantLib** and its R interface **RQuantLib** developed by Eddelbuettel and Nguyen (2010) offers several methods to estimate the term structure based on coupon bonds (i.e., simple polynomial fitting, various types of splines and the NS model). Additionally, popular one- and two-factor short rate models are covered. For the estimation of the NS model **RQuantLib** offers a weighted optimization algorithm. The start parameters of the optimization are selected automatically, however the underlying logic is not documented. The estimation is currently limited to fixed-coupon bonds. Moreover, the ability in **RQuantLib** to analyse and compare the results of the estimation appears to be limited, e.g., it is impossible to obtain the optimal parameters of the NS spot rate function. The commercial software MATLAB provides in the **Fixed-Income** toolbox functions (The MathWorks, Inc. 2010) which allow the term structure estimation based on observed fixed income prices and yields. The available models include NS, SV and smoothed splines. The estimation is carried out by applying a constrained weighted optimization procedure with box constraints. However, a start parameter search algorithm and linear constraints are not provided.

2. Zero-coupon yield curve estimation

Before we discuss the problem of model specific zero-coupon yield curve estimation, we introduce the definitions of a few basic terms used in the fixed income literature and the associated notation. Two very fundamental fixed income securities are *discount* or *zero-coupon* bonds and *fixed-coupon* bonds. The first type of bond is a fixed income investment with only one payment at maturity, whereas the second type provides fixed periodical coupons and a redemption payment at maturity. For a coupon bond j we denote the occurring cash flows, i.e., the coupons and the redemption payment, with c_{ij} and the associated maturities with m_{ij}. For a group of k bonds we summarize the cash flows and maturities in the matrices $\boldsymbol{M} = \{m_{ij}\}$ and $\boldsymbol{C} = \{c_{ij}\}$ with t rows and k columns, with $i = 1, \ldots, t$ and $j = 1, \ldots, k$. The number of rows t is determined by the number of cash flows of the j-th bond with the longest maturity. Dates after the maturity of each bond j are completed with zeros until the maturity date of the bond with the longest maturity. One element m_{ij} of $\boldsymbol{M}$ refers, therefore, to the time to occurrence (in years) of the i-th cash flow of the j-th bond. The maturities of the last cash flows m_j of the bonds are collected in the row vector $\boldsymbol{m} = \{m_j\}$ of dimension k.

The *spot rate, zero-coupon rate* or *zero yield* $s(m_{ij})$ denotes the interest paid on a discount bond for a certain maturity m_{ij}. With continuous compounding and in the absence of arbitrage opportunities the fair price of a discount bond p_j paying one Euro at maturity m_j is given by

$$p_j = e^{-s(m_j)m_j}. \tag{1}$$

The *spot curve* (or *zero-coupon yield curve*) shows the spot rates for different maturities. The

forward rate $f(r,t)$ is the interest contracted now to be paid for a future investment between the time r and t $(r < t)$, denoted by m_r and m_t. The forward rate as a function of maturity is the *forward curve*. Assuming continuous compounding, we observe the following relationship between spot rates and forward rates

$$e^{s(m_r)m_r}e^{f(r,t)(m_t - m_r)} = e^{s(m_t)m_t}.$$

We can solve this for the forward rate

$$f(r,t) = \frac{s(m_t)m_t - s(m_r)m_r}{m_t - m_r}.$$

The *instantaneous forward rate* describes the return for an infinitesimal investment period after r.

$$f(r) = \lim_{t \to r} f(r,t)$$

Another interpretation is the marginal increase in the total return from a marginal increase in the length of the investment period. Thus, the spot rate can be seen as the average of the instantaneous forward rates

$$s(m) = \frac{1}{m} \int_0^m f(x)dx, \tag{2}$$

where m is the the time to maturity.

At the market the following information is typically available for a bond j: the clean price p_j^c (given as percentage of the nominal value), the cash flows c_{ij} (coupons and redemption payment) and their maturity dates m_{ij}. When an investor buys a bond j, she obtains the right to receive all its future cash flows. If the purchase occurs between two coupon dates, the seller must be compensated for the fraction of the next coupon, the so-called *accrued interest* a_j. Depending on the market specific day-count convention, e.g., 30/360, Actual/360, a basic form for the accrued interest of bond j is

$$a_j = \frac{\text{number of days since last coupon payment}}{\text{number of days in current coupon period}} \times c_{ij}.$$

Therefore, the purchase price also named *dirty price* p_j is the sum of the quoted market price (*clean price*) p_j^c and the accrued interest a_j. We again summarize the dirty and clean prices and accrued interest of a group of bonds into the k-dimensional row vectors $\boldsymbol{p}, \boldsymbol{p}^c, \boldsymbol{a}$, where $\boldsymbol{p} = \boldsymbol{p}^c + \boldsymbol{a} = \{p_j = p_j^c a_j\}$.

By considering the accrued interest, similar to (1), the bond pricing equation for a bond j under continuous compounding is the present value of all cash flows

$$p_j^c + a_j = \sum_{i=1}^t c_{ij}\delta(m_{ij}),$$

where $\delta(m_{ij})$ is the *discount factor* for a certain maturity m_{ij}. Applying the concept of spot rates the discount factors can be expressed as $\delta(m_{ij}) = e^{-s(m_{ij})m_{ij}}$. The $t \times k$ matrix

$D = \{\delta(m_{ij})\}$ consists of all discount factors for a group of bonds and the bond pricing equation in matrix notation is therefore given by

$$p = \iota^\top(C \cdot D).$$

Throughout the paper "·" denotes an element-wise multiplication, $()^\top$ is the transpose of a vector or matrix and ι is a column vector filled with ones. Another possibility to express the bond pricing equation, is to use the internal rate of return of the cash flows. The so-called *yield to maturity* (YTM) of a bond j is the solution for y_j in the following equation

$$p_j^c + a = \sum_{i=1}^{t} c_{ij}e^{-y_j m_{ij}}.$$

We summarize the yield to maturities of a group of bonds in the k-dimensional row vector $y = \{y_j\}$. The YTM for a discount bond is equal to the spot rate. However, this is not valid for coupon bonds. Plotting just the yield to maturity for coupon bonds with different maturities does not result in a yield curve which can be used to discount cash flows or price any other fixed income security, except the bond from which it was calculated. Therefore, estimating the term structure of interest rates from a set of coupon bonds can not be seen as a simple curve-fitting of the YTMs.

To quantify the sensitivity of a bond's price against changes in the interest rate, one needs to account for the fact that coupons are paid during the lifetime of a coupon bond. A standard measure of risk is the (Macaulay) *duration* which computes the average maturity of a bond using the present values of its cash flows as weights. The k-dimensional vector d consists of the individual durations and is calculated by

$$d = \frac{\iota^\top(C \cdot M \cdot D)}{\iota^\top(C \cdot D)},$$

where the discount matrix D contains the discount factors calculated with the yield to maturity of each bond, i.e., $\delta(m_{ij}) = e^{-y_j m_{ij}}$ (the division is applied elementwise).

The objective of term structure estimation is to extract the spot, forward or discount curve out of a set of coupon bonds. The simplest methods to derive the zero-coupon yield curve are direct methods, e.g., bootstrapping. Such techniques price the bonds without error, which leads to overfitting. The direct methods require bonds with nearly identical cash flow dates and lead to nonsmooth spot, forward or discount curves. Therefore, to overcome the mentioned disadvantages indirect methods have been proposed. They postulate a parametric parsimonious form of the spot, forward or discount function. In the following we denote the vector which contains the parameters of the indirect methods by β. Because bond prices are observed with idiosyncratic errors we define the *theoretical* bond prices as

$$\hat{p} = \iota^\top(C \cdot D).$$

Thus, the market prices of set of coupon bonds p can be expressed as the sum of the theoretical bond prices $\hat{p}$ plus an idiosyncratic error vector ϵ. The estimated parameter vector $\hat{\beta}$ is then obtained by a possible weighted nonconvex optimization procedure. So far we have only mentioned the estimation of the zero-coupon yield curve based on a set of coupon bonds. However, another application is based on zero yields, which may be obtained from a bootstrap

procedure. The optimization then aims at minimizing the error between the observed $\boldsymbol{y}$ and the theoretical zero yields $\hat{\boldsymbol{y}}$.

The following two subsections focus on two popular indirect estimation methods, i.e., the method of NS and SV which propose a specific parametric forward and spot rate function and the cubic splines method which models the discount function.

2.1. Nelson/Siegel and Svensson method

Spot curve parameterizations

Nelson and Siegel (1987) propose a parsimonious function for modelling the instantaneous forward rate as a solution to a second-order differential equation for the case of equal roots

$$f(m_{ij}, \boldsymbol{\beta}) = \beta_0 + \beta_1 \exp\left(-\frac{m_{ij}}{\tau_1}\right) + \beta_2 \left[\left(\frac{m_{ij}}{\tau_1}\right) \exp\left(-\frac{m_{ij}}{\tau_1}\right)\right],$$

with the parameter vector $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2, \tau_1)$. As in (2), the spot rate is the average of the instantaneous forward rates

$$s(m_{ij}, \boldsymbol{\beta}) = \frac{1}{m_{ij}} \int_0^{m_{ij}} f(m_{ij}, \boldsymbol{\beta}) \, dm_{ij},$$

resulting in

$$s(m_{ij}, \boldsymbol{\beta}) = \beta_0 + \beta_1 \frac{1 - \exp(-\frac{m_{ij}}{\tau_1})}{\frac{m_{ij}}{\tau_1}} + \beta_2 \left[\frac{1 - \exp(-\frac{m_{ij}}{\tau_1})}{\frac{m_{ij}}{\tau_1}} - \exp\left(-\frac{m_{ij}}{\tau_1}\right)\right]. \tag{3}$$

Instead of one specific maturity m_{ij} the spot rate function can also be written in terms of a maturity vector $\boldsymbol{m}$, i.e., $s(\boldsymbol{m}, \boldsymbol{\beta})$. Consequently, all calculations in (3) have to be applied element by element. The specification of NS can produce a wide range of possible curve shapes, including monotonic, humped, U-shapes or S-shapes. Svensson (1994) adds the term

$$\beta_3 \left[\frac{1 - \exp(-\frac{m_{ij}}{\tau_2})}{\frac{m_{ij}}{\tau_2}} - \exp\left(-\frac{m_{ij}}{\tau_2}\right)\right], \tag{4}$$

with two new parameters β_3 and τ_2 to increase the flexibility. The parameter vector is then given by $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2, \tau_1, \beta_3, \tau_2)$. This specification of the spot rate allows for a second hump in the curve. Based on the proposed spot rate function of NS and SV several extensions have been developed. In order to simplify the estimation procedure DL suggest to reduce the parameter vector to $\boldsymbol{\beta} = (\beta_0, \beta_1, \beta_2)$ by fixing $\tau_1 = \frac{1}{\lambda}$ to a prespecified value. A potential multicollinearity problem in the SV model arises if the decay parameters τ_1 and τ_2 have similar values. As a result only the sum of β_2 and β_3 can be estimated efficiently. To circumvent the multicollinearity problem De Pooter (2007) proposes similar to Björk and Christensen (1999) to replace the last term in (4), i.e., $-\exp\left(-\frac{m_{ij}}{\tau_2}\right)$ by $-\exp\left(-\frac{2m_{ij}}{\tau_2}\right)$. In the following we refer to the SV model with the replaced term as adjusted SV. Because the named models are nested in the formulation of the (adjusted) SV spot rate function the following parameter interpretations also hold:

- β_0 is the asymptotic value of the spot rate function $\lim_{m_{ij} \to \infty} s(m_{ij}, \boldsymbol{\beta})$, which can be seen as the long-term interest rate. Due to the assumption of positive interest rates it is required that $\beta_0 > 0$.

- β_1 determines the rate of convergence with which the spot rate function approaches its long-term trend. The slope will be negative if $\beta_1 > 0$ and vice versa.

- The instantaneous short rate is given by $\lim_{m_{ij} \to 0} s(m_{ij}, \boldsymbol{\beta}) = \beta_0 + \beta_1$, which is constrained to be greater zero.

- β_2 determines the size and the form of the hump. $\beta_2 > 0$ results in a hump at τ_1, whereas $\beta_2 < 0$ produces a U-shape.

- τ_1 specifies the location of the first hump or the U-shape on the curve.

- β_3, analogously to β_2, determines the size and form of the second hump.

- τ_2 specifies the position of the second hump.

- It is required that $\tau_1, \tau_2 > 0$. Moreover, we restrict $\tau_2 - \tau_1 > \Delta\tau$ to ensure the interpretation of the first and second hump position (for the adjusted SV $\Delta\tau = 0$).

For the NS (DL), SV (adjusted SV) spot rate function defined in (3) and (4) the elements of the discount matrix $\boldsymbol{D}$ can be calculated as follows:

$$\delta(m_{ij}, \boldsymbol{\beta}) = e^{-m_{ij}s(m_{ij}, \boldsymbol{\beta})/100} \, .$$

Weighted least squares and globally optimal parameter estimation

The unknown parameter vector of the spot rate functions $\boldsymbol{\beta}$ can be obtained by minimizing the error between the observed bond prices $\boldsymbol{p}$ and the theoretical bond prices $\hat{\boldsymbol{p}}$. However, when minimizing the unweighted price errors, bonds with a longer maturity obtain a higher weighting, due to higher degree of price sensitivity, which leads to a less accurate fit at the short end. Therefore, a weighting of the price errors has to be introduced to solve this problem or at least to reduce the degree of heteroscedasticity. As Martellini, Priaulet, and Priaulet (2003) point out, the choice of the weights is a crucial part of the optimization procedure. We use a specification which is based on the inverse of the duration and was proposed by Bliss (1997). The weight ω_j for a bond j is given by

$$\omega_j = \frac{d_j^{-1}}{\sum_{i=1}^{k} d_i^{-1}} \, .$$

Alternative weighting schemes also make use of the duration (see, e.g., Vasicek and Fong 1982) or the bid-ask spread (see, e.g., Nawalkha, Soto, and Beliaeva 2005). We summarize the individual weights for k bonds into the k-dimensional row vector $\boldsymbol{\omega}$. The consideration of the weights leads to the first objective function, which minimizes the sum of the weighted squared coupon bond price errors The second minimizes the sum of the squared zero yield errors. The objective function $F(\boldsymbol{\beta})$ is given by

$$F(\boldsymbol{\beta}) = \begin{cases} \left(\boldsymbol{p} - \boldsymbol{\iota}^\top (\boldsymbol{C} \cdot \boldsymbol{D}) \right)^2 \boldsymbol{\omega}^\top & \text{coupon bond data} \\ \left(\boldsymbol{y} - s(\boldsymbol{m}, \boldsymbol{\beta}) \right)^2 \boldsymbol{\iota} & \text{zero yield data.} \end{cases} \tag{5}$$

The entire optimization problem for the SV spot rate function considering all mentioned constraints can be written as

$$\min_{\boldsymbol{\beta} \in \mathbb{R}^6} \quad F(\boldsymbol{\beta}) \tag{6}$$

$$\text{s.t.} \quad \boldsymbol{A}\boldsymbol{\beta} \quad \leq \quad \boldsymbol{b} \tag{7}$$

$$\boldsymbol{l} < \boldsymbol{\beta} \quad \leq \quad \boldsymbol{u}. \tag{8}$$

The estimation problems using the NS, DL and adjusted SV spot rate function are nested in the above formulation. Therefore, we refrain from reporting upon them. We derive the analytical gradients $\boldsymbol{\nabla} F(\boldsymbol{\beta})$ for both objective functions and report them in Appendix A. The optimization problem stated in (6)–(8) is nonconvex and may has multiple local optima, which increases the dependence of the numeric solution on the starting values considerably. Choosing the start parameters arbitrarily and possibly not reaching a global optimum can be avoided by using the following reformulation of the problem based on an idea of Werner and Ferenczi (2006). The parameter vector $\boldsymbol{\beta}$ is split in a local component $\boldsymbol{\beta}^*$ and a global component $\boldsymbol{\tau}$. Then the original optimization problem in (6)–(8) is equivalent to:

$$\min_{\boldsymbol{\tau} \in \mathbb{R}^2} \quad H(\boldsymbol{\tau}) \tag{9}$$

$$\text{s.t.} \quad \boldsymbol{A}_o\boldsymbol{\tau} \quad \leq \quad \boldsymbol{b}_o \tag{10}$$

$$\boldsymbol{l}_o < \boldsymbol{\tau} \quad \leq \quad \boldsymbol{u}_o \tag{11}$$

where

$$H(\boldsymbol{\tau}) = \min_{\boldsymbol{\beta}^* \in \mathbb{R}^4} \quad F(\boldsymbol{\beta}^*, \boldsymbol{\tau}) \tag{12}$$

$$\text{s.t.} \quad \boldsymbol{A}_i\boldsymbol{\beta}^* \quad \leq \quad \boldsymbol{b}_i \tag{13}$$

$$\boldsymbol{l}_i < \boldsymbol{\beta}^* \quad \leq \quad \boldsymbol{u}_i. \tag{14}$$

The inner problem (12)–(14) is a (weighted) least squares problem with four variables under linear constraints, which is convex and therefore easy to solve numerically. In contrast, the outer problem (9)–(11) is nonconvex in two variables. Werner and Ferenczi (2006) formulate the optimization problem with the usual parameter constraints given in Svensson (1994), i.e., with only box-constraints for the τ-parameters. Under these constraints the nonconvex outer problem is solved with a grid search method. Werner and Ferenczi (2006) use various global search algorithms (e.g., sparse grids, HCP algorithm) for this purpose. We apply a full grid search in which we are able to reduce the size by at least the half. The reduction is a consequence of the imposed minimum distance constraint Δ_τ between τ_1 and τ_2. The additional linear constraint for the τ-parameters was proposed by De Pooter (2007) and avoids identification problems that could arise from the similar factor loading structure associated with the parameters β_2 and β_3. In a static setting, the only consequence of not imposing this additional constraint can be rather extreme parameter estimates, where the total factor contributions virtually cancel each other out. However, this does not influence the accuracy of the fit. De Pooter (2007) notes that the real advantage of the additional constraint are smoother parameter time series when performing a two-step dynamic estimation. Furthermore, the interpretation of τ_1 and τ_2 as the position of the humps is ensured, because they are not able to switch their positions. We can confirm these results and have included the

necessary constraints in the code of the optimization algorithm. To sum up, the local optimization problem in six parameters defined in (6)–(8) is solved with the globally optimal starting point from the grid search procedure. An advantage of our method is that the user does not have to specify start paramaters in advance. We apply the above algorithm, with a reduced parameter space ($\boldsymbol{\beta} \in \mathbb{R}^4, \boldsymbol{\beta}^* \in \mathbb{R}^3, \boldsymbol{\tau} \in \mathbb{R}^1$), also for the Nelson/Siegel model.

2.2. Cubic splines

The cubic splines based term structure estimation method divides the term structure into segments using a series of so-called *knot points*. Cubic polynomial functions are then used to fit the term structure over these segments. The polynomial functions ensure the continuity and smoothness of the discount function within each interval. McCulloch (1971, 1975) use the following definition of the discount factors:

$$\delta(m_{ij}, \boldsymbol{\beta}) = 1 + \sum_{l=1}^{n} \beta^l g^l(m_{ij}), \tag{15}$$

Where $g^l(m_{ij})$ ($l = 1, \dots, n$) defines a set of piecewise cubic functions, the so-called basis functions, which satisfy $g^l(0) = 0$. The functions have to be twice-differentiable at each knot point to ensure a smooth and continuous curve around the points. The unknown parameter vector $\boldsymbol{\beta}$ can be estimated with ordinary least squares (OLS).

Knot point selection

McCulloch (1975) defines a n-parameter spline with $n - 1$ knot points q_l. We sort the cash flow matrix $\boldsymbol{C}$ and the maturity matrix $\boldsymbol{M}$ such that the k bonds are arranged in ascending order by their maturity dates $\boldsymbol{m}$. The following specification places an approximately equal number of bonds between adjacent knots. For $1 < l < n - 1$ the knot points are defined as

$$q_l = m_h + \theta(m_{h+1} - m_h),$$

with $h = \left\lceil \frac{(l-1)k}{n-2} \right\rceil$ and $\theta = \frac{(l-1)k}{n-2} - h$. The first knot point $q_1 = 0$ and the last one is equal to the maximum maturity, i.e., $q_{n-1} = m_k$. McCulloch (1971) proposes to set the number of basis functions n to the integer nearest to the square root of the number of observed bonds k, i.e., $n = \left\lfloor \sqrt{k} + 0.5 \right\rfloor$.

Basis functions for cubic splines

For $l < n$ the set of basis functions is defined by

$$g^l(m_{ij}) = \begin{cases} 0 & m_{ij} < q_{l-1}, \\ \frac{(m_{ij} - q_{l-1})^3}{6(q_l - q_{l-1})} & q_{l-1} \leq m_{ij} < q_l, \\ \frac{(q_l - q_{l-1})^2}{6} + \frac{(q_l - q_{l-1})(m_{ij} - q_l)}{2} + \frac{(m_{ij} - q_l)^2}{2} - \frac{(m_{ij} - q_l)^3}{6(q_{l+1} - q_l)} & q_l \leq m_{ij} < q_{l+1}, \\ (q_{l+1} - q_{l-1}) \left[\frac{2q_{l+1} - q_l - q_{l-1}}{6} + \frac{m_{ij} - q_{l+1}}{2} \right] & q_{l+1} \leq m_{ij}. \end{cases}$$

For $l = 1$ we set $q_{l-1} = q_l = 0$. When $l = n$, the basis function is given by $g^l(m_{ij}) = m_{ij}$. For a set of bonds we summarize the basis functions for all cash flows m_{ij} in the $t \times k$ matrix $\boldsymbol{G}^l = \{g_{ij}^l(m_{ij})\}$.

Regression fitting of the discount function

For a set of k coupon bonds the dirty price vector can be expressed as the sum of the discounted cash flows, i.e., the theoretical prices plus an idiosyncratic error

$$\boldsymbol{p} = \boldsymbol{\iota}^\top (\boldsymbol{C} \cdot \boldsymbol{D}) + \boldsymbol{\epsilon}, \tag{16}$$

where the discount factor matrix is defined as the weighted sum of the $l = 1 \ldots n$ basis functions

$$\boldsymbol{D} = 1 + \beta^1 \boldsymbol{G}^1 + \cdots + \beta^n \boldsymbol{G}^n. \tag{17}$$

We substitute (17) in (16), rearrange the terms and obtain an expression which is linear in the parameter vector $\boldsymbol{\beta} = (\beta^1, \ldots, \beta^n)$.

$$\boldsymbol{p} - \boldsymbol{\iota}^\top \boldsymbol{C} = \beta^1 \boldsymbol{\iota}^\top \boldsymbol{C} \cdot \boldsymbol{G}^1 + \cdots + \beta^n \boldsymbol{\iota}^\top \boldsymbol{C} \cdot \boldsymbol{G}^n + \boldsymbol{\epsilon}$$

We construct a $t \times n$ matrix $\boldsymbol{X}$, where the l-th column consists of $\left(\boldsymbol{\iota}^\top \boldsymbol{C} \cdot \boldsymbol{G}^l \right)^\top$ and $l = 1, \ldots, n$. The difference between the observed price vector $\boldsymbol{p}$ and the sum of the cash flows is summarized by a t-dimensional column vector $\boldsymbol{z}$, which is given by $\left(\boldsymbol{p} - \boldsymbol{\iota}^\top \boldsymbol{C} \right)^\top$. The unknown parameters are denoted by the $t \times 1$ vector $\boldsymbol{\beta}$. The multivariate linear regression equation is therefore given by $\boldsymbol{z} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$. The parameter estimates are obtained by simple OLS, i.e., the $n \times 1$ vector $\hat{\boldsymbol{\beta}} = \left(\boldsymbol{X}^\top \boldsymbol{X} \right)^{-1} \boldsymbol{X}^\top \boldsymbol{z}$.

We can use the resulting parameters to calculate the discount function in (15) for any given maturity m_{ij} between the first and the last knot point. The discount factors can then be converted into the spot rates by the following relationship

$$s(m_{ij}, \boldsymbol{\beta}) = \frac{-\ln \delta(m_{ij}, \boldsymbol{\beta})}{m_{ij}}.$$

Confidence intervals for the discount function

McCulloch (1975) plots error bands one standard error above and below the best estimate. However, it is also possible to derive a confidence interval for the estimated discount function. Under the assumption of n.i.i.d. disturbances $\boldsymbol{\epsilon}$ with variance σ^2, the ordinary least squares coefficient estimator is normally distributed with mean $\boldsymbol{\beta}$ and variance-covariance matrix $\sigma^2 \boldsymbol{X}^\top \boldsymbol{X}^{-1}$. If the disturbances exhibit heteroscedasticity or/and autocorrelation the estimated covariance matrix $\sigma^2 \boldsymbol{X}^\top \boldsymbol{X}^{-1}$ will be inappropriate. The appropriate matrix is then given by $\boldsymbol{\Psi} = \sigma^2 \boldsymbol{X}^\top \boldsymbol{X}^{-1} (\boldsymbol{X}^\top \boldsymbol{\Omega} \boldsymbol{X})(\boldsymbol{X}\boldsymbol{X})^{-1}$, where $\boldsymbol{\Omega}$ is the weights matrix. Therefore, autocorrelation and heteroscedasticity consistent estimators for $\boldsymbol{\Psi}$ have been proposed. Within the package and for the calculations in this paper we use an estimator developed by Andrews (1991) and provided by the R package **sandwich** of Zeileis (2004, 2006), .

According to Greene (2002), the confidence interval for a linear combination of OLS coefficients follows a normal distribution. Therefore, the discount function in (15) is normally distributed with mean $\mu_d = 1 + \boldsymbol{g}(m_{ij})^\top \boldsymbol{\beta}$ and variance-covariance matrix $\sigma_d^2 = \boldsymbol{g}(m_{ij})^\top \boldsymbol{\Psi} \boldsymbol{g}(m_{ij})$, where $\boldsymbol{g}(m_{ij})$ is a n-dimensional column vector with consists of the elements $g^l(m_{ij})$ ($l = 1, \ldots, n$). The $1 - \alpha$ confidence interval for the mean of the discount function can now be constructed as follows

$$\mathrm{P}\left[\delta(m_{ij}) - t_{\alpha/2} s_d \leq \mu_d \leq \delta(m_{ij}) + t_{\alpha/2} s_d \right] = 1 - \alpha,$$

where s_d is the estimate for σ_d, $t_{\alpha/2}$ the appropriate critical value from the t distribution with $k - n$ degrees of freedom and $1 - \alpha$ the desired level of confidence.

3. Software implementation in R

In this section, we describe the structure of the implementation of the package **termstrc** in the R system for statistical computing. The matrix-oriented notation introduced in Section 2 can efficiently be realized with R base functions. However, a few functions depend on other packages, which we appropriately reference. Table 1 lists the basic structure of the package. It contains of three data classes and two associated estimation methods. Subsequently, we explain in detail the structure of our data sets, the core functions and describe the available standard and generic S3 classes and methods provided by the package.

3.1. Data structure

The package **termstrc** makes it possible to estimate the term structure based on observed data of coupon bonds and yields. Accordingly, we have defined two basic types of data sets and corresponding classes, i.e., the class `"couponbonds"` for a set of coupon bonds and the class `"zeroyields"` for zero yields . An object of the class `"couponbonds"` has to be organized in a list form, which contains sublists for each country or group of bonds. The structure of the sublist is illustrated in Table 2.

The first part consists of the general specifications of the bonds, its market prices and accrued interest. The sublist `CASHFLOWS` contains all cash flows for the available bonds sorted by their ISIN and their maturity date. The list structure allows us to add additional data easily, e.g., bid and ask prices, volumes. Due to the different day-count-conventions, business holidays and market conventions, no functions for the calculation of accrued interests, maturity dates and cash flows are provided. The data structure provides itself to be convenient when building the cash flow matrices and the maturity matrices defined in Section 2. Moreover, the creation of dynamic data sets can be easily achieved by merging static `"couponbonds"` data sets. For the resulting object we have invented the class `"dyncouponbonds"`. Objects of the class `"zeroyields"` are structured in a similar way. However, only sublists with the yields and the associated maturities are required. Due to the simple data structure the package provides no separate class for static zero yield data. Moreover, because of the different data formats of the data providers no general class constructor functions, except for the class `"zeroyields"`, are included. For the classes `"couponbonds"` and `"dyncouponbonds"` the package offers a `rm_bond()` method, which allows us to remove a specified vector of bonds and their associated

Data class	Method	Returned object	Method for class
`"couponbonds"`	`estim_cs()`	`"termstrc_cs"`	`print()`, `plot()`,
	`estim_nss()`	`"termstrc_nss"`	`summary()`
`"dyncouponbonds"`	`estim_nss()`	`"dyntermstrc_nss"`	`print()`, `plot()`,
			`summary()`, `param()`
`"zeroyields"`	`estim_nss()`	`"dyntermstrc_yields"`	`print()`, `plot()`,
			`summary()`, `param()`

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

Sublist	Description
`ISIN`	International Securities Identifying Number (ISIN)
`MATURITYDATE`	maturity dates
`ISSUEDATE`	issue dates
`COUPONRATE`	coupon rates as percentage of nominal values
`PRICE`	observed market prices (clean prices)
`ACCRUED`	accrued interest
`TODAY`	observation date
`CASHFLOWS$ISIN`	International Securities Identifying Number (ISIN)
`CASHFLOWS$CF`	all cash flows for a given ISIN
`CASHFLOWS$DATE`	maturity dates of all cash flows

Table 2: Structure of a sublist of an object of the class `"couponbonds"`.

data from the data set.

3.2. Auxiliary functions

The term structure estimation functions are based on several functions which perform typical fixed income mathematics operations. The majority of them can be applied independently. The most important functions are `create_cashflows_matrix()` and `create_maturities_matrix()` for the creation of the cash flows and maturities matrices. Based on the obtained matrices the theoretical prices and yield to maturities of the bonds can be calculated with `bond_prices()` and `bond_yields()`. The implemented spot rate and forward rate functions for the different methods allow the calculation of the required interest rate type based on the parameter and maturity vector. The function `duration()` calculates basic interest rate risk measures, i.e., the Macauly and modified duration.

3.3. Core functions

The package offers functions for the estimation of the term structure based on NS, SV, DL, adjusted SV implemented in the method `estim_nss()` and cubic splines implemented in `estim_cs()`. The first method is available for objects of the class `"couponbonds"`, `"zeroyields"` and `"dyncouponbonds"`, whereas the latter one only for `"couponbonds"` objects. All methods require the appropriate data set as an first input argument. The methods available for coupon bond data sets allow the restriction of the estimation up to a certain maturity range and group of bonds. Further inputs of the `estim_nss()` method enable the user to provide constraints for the start parameter grid search, to set the options of the used optimizer appropriately and to define the fixed parameter $\lambda = \frac{1}{\tau_1}$ of the DL spot rate function. The `estim_nss()` method for `"couponbonds"` objects gives the user the option to provide start parameters for the optimization. Without them the grid search algorithm is automatically applied. For the iterative estimation of the term structure the package includes the `estim_nss()` methods for `"dyncouponbonds"` and `"zeroyields"` objects. The first calls at every estimation step the static `estim_nss()` estimation method for `"couponbonds"` objects, because an object of the class `"dyncouponbonds"` consists of subobjects of the class `"couponbonds"`. For the iterative estimation a global optimization at every or the first point in time is possible. For the second alternative the results of the previous estimation are used

as start parameters for the current one. All the mentioned functions follow the same internal structure, i.e., data preprocessing, start parameter grid search, optimization and postprocessing of the results. In order to improve the speed of the term structure estimation based on coupon bonds and the associated start parameter grid search we outsourced performance-critical functions, i.e., the objective functions and gradients for the NS and SV methods, into C++ by applying the package **Rcpp** of Eddelbuettel and Francois (2010).

3.4. Exploring the estimation results

The methods for objects of the class `"couponbonds"` and `"dyncouponbonds"` return objects of the class `"termstrc_nss"` or `"termstrc_cs"` and `"dyntermstrc_nss"`, whereas the method applied on the class `"zeroyields"` returns an object of the class `"dyntermstrc_yields"`. All the obtained objects from a certain term structure estimation method contain the estimation results, pricing errors, input parameters, pre and postprocessed data, results of the start parameter grid search and method specific information in a list format. The sublists follow the structure of the input data sets, i.e., they are classified according to countries or an other group argument. For all the mentioned result classes we provide appropriate `print(), plot()` and `summary()` methods. The `summary()` method gives goodness of fit measures, i.e., the RMSE and AABSE for the pricing and yield errors and a convergence information from the solver is printed. For static results objects the `print()` method shows the estimated parameters and method specific information, whereas for dynamic result objects the `print()` method offers aggregated summary statistics on the estimated parameters. The parameters of the cubic splines approach are estimated with OLS by using the `lm()` function. Consequently, the `print()` method for `"termstrc_cs"` also provides statistics for the OLS estimation by applying the `summary()` method for `"lm"` objects or in case of `rse = TRUE` robust standard errors are used for the calculation of the statistics. The method applies the function `coeftest()` from the package **lmtest** of Zeileis and Hothorn (2002). Additionally, we implemented a `param()` method for objects of the class `"dyntermstrc_yields"` and `"dyntermstrc_nss"`. The method allows to conveniently extract the estimated parameter time series and an object of the class `"dyntermstrc_param"` is returned. Using the `summary()` method leads to an overview about the correlation and possible unit roots of the estimated parameter time series for the levels and the first differences. The unit root tests are performed with functions form the **urca** package of Pfaff (2008).

3.5. Visualization of the results

The package contains several `S3 plot()` methods which allow us to visualize different interest rate curves, estimation errors, start parameter grid search results and the parameter estimates. The `plot()` methods for `"termstrc_cs"` and `"termstrc_nss"` objects offer the following possibilities:

- plot zero-coupon, forward, discount or spread curves,

- return single/multiple plots for the estimated group of bonds,

- show error plots for pricing/yield errors to identify outliers.

Plots of the zero-coupon yield curve for a single country include additional information. In detail, the yield to maturities are also plotted and for objects of the class `"termstrc_cs"`,

the knot points used for the estimation and the 95% confidence interval, based on robust standard errors, of the zero-coupon yield curve are added to the figure. The robust standard errors are calculated by applying functions of the package **sandwich** of Zeileis (2004). Both `plot()` methods depend on methods of the classes `"spot_curves"`, `"fwr_curves"` and `"df_curves"` which are itself based on the `plot()` method of the class `"ir_curve"`, which is the most granular class. An object of the class `"spot_curves"`, `"fwr_curves"` or `"df_curves"` can therefore consist of several objects of the class `"ir_curve"`. The advantage for the user is that when exploring a `"termstrc_nss"` or `"termstrc_cs"` object, she can plot the different curves at every hierarchical level of the object. For example, the command `plot(<object>$spot$<COUNTRY>)` creates a figure of the spot curve of <COUNTRY>, while `plot(<object>$spot)` plots the spot curves of all available countries. The `plot()` method for dynamic estimation results objects of the class `"dyntermstrc_nss"` and `"dyntermstrc_yields"` allows us to illustrate the evolution of the spot curves over time. The three-dimensional plots depend on the **rgl** package of Adler and Murdoch (2010). Objects of the class `"dyntermstrc_param"` contain the extracted time series of the estimated parameters. By applying the associated `plot()` method the time series of the levels and first differences of the parameters or the empirical autocorrelation function can be plotted. An inspection of the factor contribution plot enables the user to identify possible identification problems. The method `fcontrib()` is available for `"dyntermstrc_param"` objects. Moreover, the package offers a `plot()` method for `"spsearch"` objects, which include results on the start parameter grid search. Every static and dynamic results object except `"termstrc_cs"` includes a `"spsearch"` object. The command `plot(<object>$spsearch$<COUNTRY>)` plots the objective function of the start parameter grid search and a contour plot. The inspection of both helps us to understand the optimization problem in terms of multiple local minima.

4. Examples

In the following section we will estimate zero-coupon yield curves in R from various data sets with the afore mentioned methods. The examples are available as demos in the package.

4.1. Nelson/Siegel and Svensson method

After loading the package **termstrc**, we explore a static data set consisting of price information on coupon bonds from several countries.

```
R> library("termstrc")
R> data("govbonds")
R> govbonds


This is a data set of coupon bonds for:
GERMANY AUSTRIA FRANCE ,
observed at 2008-01-30.


R> class(govbonds)


[1] "couponbonds"
```

It includes price data for government bonds of three European countries. The bonds are classified by their *International Securities Identifying Number (ISIN)*, and all the necessary information on the future cash flows is provided. Next, we see the structure described in Table 2 for the sublist of German bonds.

```
R> str(govbonds$GERMANY)
```

```
List of 8
 $ ISIN       : chr [1:52] "DE0001141414" "DE0001137131" "DE0001141422" ...
 $ MATURITYDATE:Class 'Date'  num [1:52] 13924 13952 13980 14043 ...
 $ ISSUEDATE   :Class 'Date'  num [1:52] 11913 13215 12153 13298 ...
 $ COUPONRATE  : num [1:52] 0.0425 0.03 0.03 0.0325 ...
 $ PRICE       : num [1:52] 100 99.9 99.8 99.8 ...
 $ ACCRUED     : num [1:52] 4.09 2.66 2.43 2.07 ...
 $ CASHFLOWS   :List of 3
  ..$ ISIN: chr [1:384] "DE0001141414" "DE0001137131" "DE0001141422" ...
  ..$ CF  : num [1:384] 104 103 103 103 ...
  ..$ DATE:Class 'Date'  num [1:384] 13924 13952 13980 14043 ...
 $ TODAY       :Class 'Date'  num 13908
```

Lets suppose we want to estimate the zero-coupon yield curve for the three included countries with the Nelson and Siegel (1987) method minimizing the duration weighted pricing errors. The sample of bonds is restricted to a maximum maturity of 30 years. We perform the estimation with the `estim_nss()` method available for the `"couponbonds"` class. The optional argument `tauconstr` defines the box constraint $0.2 < \tau_1 \leq 5$ and a grid step size of 0.1 years for the start parameter search procedure. If not supplied by the user, economically meaningful constraints based on the available maturities in the data set are chosen.

```
R> ns_res <- estim_nss(govbonds, c("GERMANY", "AUSTRIA", "FRANCE"),
+    matrange = c(0, 30), method = "ns", tauconstr = list(c(0.2, 5, 0.1),
+    c(0.2, 5, 0.1), c(0.2, 5, 0.1)))
```

```
[1] "Searching startparameters for  GERMANY"
    beta0     beta1     beta2      tau1
 5.132836 -1.274357 -3.208435  2.700100
[1] "Searching startparameters for  AUSTRIA"
    beta0     beta1     beta2      tau1
 5.050193 -1.327244 -2.629411  2.500100
[1] "Searching startparameters for  FRANCE"
    beta0     beta1     beta2      tau1
 5.108886 -1.217795 -3.068065  2.500100
```

The optimal start parameters chosen for each country are displayed during the estimation. The results for the grid search are available as sublists for each country and belong to the `"spsearch"` class. In the following, we call the corresponding `plot()` method and the resulting plot is displayed in Figure 1.
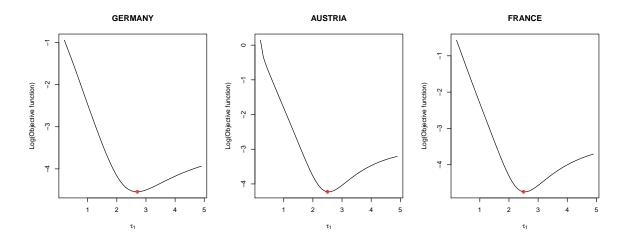
Figure 1: Start parameter grid search objective function for Nelson/Siegel.

```
R> class(ns_res$spsearch$GERMANY)

 [1] "spsearch"

R> par(mfrow = c(1, 3))
R> plot(ns_res$spsearch$GERMANY, main = "GERMANY")
R> plot(ns_res$spsearch$AUSTRIA, main = "AUSTRIA")
R> plot(ns_res$spsearch$FRANCE, main = "FRANCE")
```

Next, we see the globally optimal estimated parameters for each country. They have the same order of magnitude, which implies similar shapes for the zero-coupon yield curves.

```
R> ns_res

----------------------------------------------------
Estimated Nelson/Siegel parameters:
----------------------------------------------------


        GERMANY   AUSTRIA   FRANCE
beta_0  5.13067   5.05562   5.11861
beta_1 -1.26939  -1.35185  -1.24247
beta_2 -3.21445  -2.58210  -3.03670
tau_1   2.69026   2.53985   2.54291
```

The `summary()` method gives goodness of fit measures for the pricing and the yield errors, i.e., the root mean squared error (RMSE) and the average absolute error (AABSE). Moreover, it shows the convergence information from the solver to check whether a solution to the nonconvex optimization problem has been found.

```
R> summary(ns_res)
```

```
-----------------------------------------------------
Goodness of fit:
-----------------------------------------------------


                    GERMANY    AUSTRIA    FRANCE
RMSE-Prices         0.3582035 0.1801092 0.2214614
AABSE-Prices        0.1992017 0.1224709 0.1182058
RMSE-Yields (in %)  0.0848693 0.0185987 0.0392390
AABSE-Yields (in %) 0.0498134 0.0155659 0.0275052



-----------------------------------------------------
Startparameters:
-----------------------------------------------------


        beta0     beta1     beta2     tau1
GERMANY 5.13284  -1.27436  -3.20844  2.70010
AUSTRIA 5.05019  -1.32724  -2.62941  2.50010
FRANCE  5.10889  -1.21779  -3.06807  2.50010



-----------------------------------------------------
Convergence information:
-----------------------------------------------------


        optim() convergence info
GERMANY                         0
AUSTRIA                         0
FRANCE                          0


        optim() solver message
GERMANY NULL
AUSTRIA NULL
FRANCE  NULL
```

Our package offers several options to plot the results, e.g., spot rate, forward rate, discount and spread curves. Figure 2 shows the estimated zero-coupon yield curves. The dashed lines indicate that the curve was extrapolated to match the longest available maturity.

```
R> plot(ns_res, multiple = TRUE)
```

4.2. Cubic splines

In this section, we demonstrate how to estimate the term structure of interest rates with the McCulloch (1975) cubic splines approach applied to French governement bonds.

```
R> cs_res <- estim_cs(govbonds, "FRANCE", matrange = c(0, 30))
R> cs_res
```
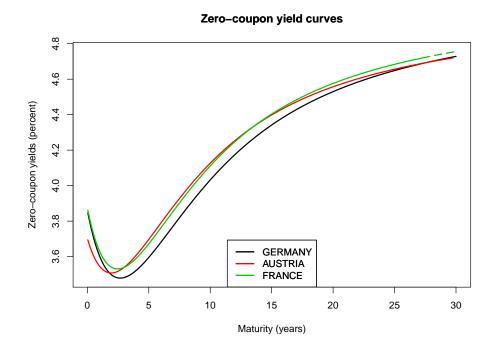
Figure 2: Zero-coupon yield curves estimated with the Nelson/Siegel model.

```
----------------------------------------------------
Estimated parameters and robust standard errors:
----------------------------------------------------
[1] "FRANCE:"

t test of coefficients:

          Estimate  Std. Error  t value   Pr(>|t|)
alpha 1  0.01366206  0.00144839   9.4326  2.889e-11 ***
alpha 2 -0.00180226  0.00061908  -2.9112   0.006144 **
alpha 3 -0.00037729  0.00041331  -0.9129   0.367392
alpha 4  0.00018673  0.00024763   0.7541   0.455709
alpha 5  0.00106333  0.00012055   8.8204  1.589e-10 ***
alpha 6  0.00146938  0.00015525   9.4646  2.646e-11 ***
alpha 7 -0.04088792  0.00052860 -77.3509 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The `summary()` method returns the usual goodness of fit measures.

`R> summary(cs_res)`

```
----------------------------------------------------
Goodness of fit:
```

FRANCE
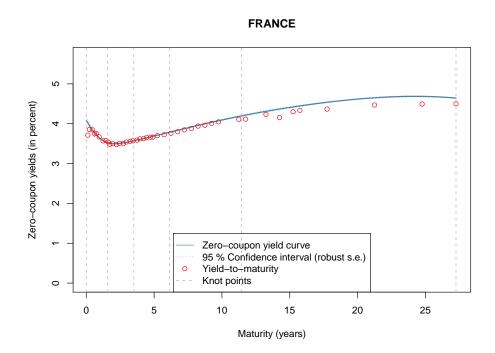
Figure 3: Zero-coupon yield curve for French government bonds estimated with cubic splines.

```
----------------------------------------------------

                       FRANCE
RMSE-Prices            0.1819771
AABSE-Prices           0.0820363
RMSE-Yields (in %)     0.0528660
AABSE-Yields (in %)    0.0225692
```

Compared to goodness of fit obtained in Section 4.1 for the French government bonds the cubic splines estimation leads to smaller errors. However, the number of estimated parameters is nearly twice as high and moreover assigning a clear economic interpretation to the parameters appears complicated. Therefore, fitting the zero-coupon yield curve could be preferable to the discount curve. Figure 3 shows the yield to maturities and the estimated zero-coupon yield curve together with the automatically selected knot points.

```
R> plot(cs_res)
```

As we can see from the plotted pricing errors in Figure 4, there seems to be a misspricing of two bonds.

```
R> plot(cs_res,errors="price")
```

They can be removed by applying the `rm_bond()` method and the estimation is redone. As expected, the goodness of fit improves.
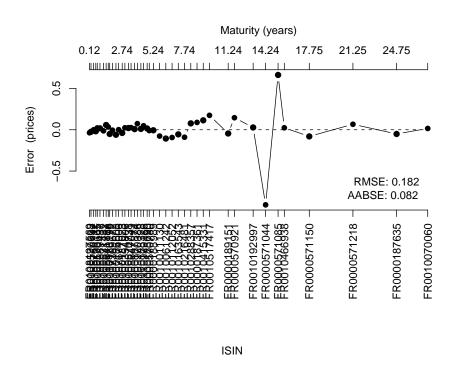
FRANCE

Figure 4: Pricing errors of the term structure estimation based on French government bonds.

```
R> cs_res2 <- estim_cs(rm_bond(govbonds, "FRANCE",
+    c("FR0000571044", "FR0000571085")), "FRANCE", matrange = c(0, 30))
R> summary(cs_res2)


----------------------------------------------------
Goodness of fit:
----------------------------------------------------


                     FRANCE
RMSE-Prices          0.0615631
AABSE-Prices         0.0514044
RMSE-Yields (in %)   0.0345521
AABSE-Yields (in %)  0.0199479
```

4.3. Rolling estimation procedure

In the last section, we will perform a Nelson/Siegel and Svensson estimation on data sets that store price/yield information for several days, i.e., for objects of the `"zeroyields"` or the `"dyncouponbonds"` class.

Zero yield data

The **termstrc** package includes a data set of German zero yields in CSV (comma-separated values) format. We load it and construct a `"zeroyields"` object.

```
R> data("zyields")
R> x <- zyields
R> maturities <- c(1/12, 3/12, 6/12, 9/12, 1:12)
R> yields <- as.matrix(x[, 2:ncol(x)])
R> dates <- as.Date(x[, 1], format = "%d.%m.%Y")
R> datazeroyields <- zeroyields(maturities, yields, dates)
R> datazeroyields
```

```
This is a data set of zero-coupon yields.
Maturities range from 0.0833333333333333 to 12 years.
There are 80 observations between 2004-01-01 and 2005-07-07 .
```

For this class an `estim_nss()` method is available. We estimate all different model parameterizations, and by default a global start parameter search is performed for the first observation. Later time stages use the optimal parameter vector of the previous estimation as a starting value. With this procedure, the estimation tends to stay in a global optimum over time. As mentioned in Section 2.1, choosing an appropriate upper bound for the τ_1 and τ_2 parameter and a minimum distance between them, avoids identification problems and leads to smooth parameter time series. The argument `tauconstr` has the following structure: (lower bound τ_1, τ_2; upper bound τ_1, τ_2; grid step size; minimum distance $\Delta\tau$ between τ_1 and τ_2). Performing the estimation with a start parameter search in each time step by setting `optimtype = "allglobal"` leads to only marginally different parameter values (for a small number of observations) in our examples. Therefore, we set `optimtype = "firstglobal"` as default value, and recommend performing a start parameter search only if an abrupt jump in the parameter values occurs or if the solver encounters convergence problems. We observed that for looser constraints on the τ-parameters, the solution can alternate between two optima. However, the improvement in the fit is negligible, and the smooth time evolution of the parameters is lost. Therefore, we set the upper bound tight enough to avoid alternating optima but ensured that the optimal solution does not actually hit the bound. In particular for the (adjusted) Svensson model, limiting the grid to a sufficient small size is desirable for performance reasons.

```
R> dl_res <- estim_nss(datazeroyields, "dl", lambda = 1/2)
R> ns_res <- estim_nss(datazeroyields, "ns", tauconstr = c(0.2, 6, 0.1))
R> sv_res <- estim_nss(datazeroyields, "sv", tauconstr = c(0.2, 5, 0.1, 0.5))
R> asv_res <- estim_nss(datazeroyields, "asv", tauconstr =  c(0.2, 7, 0.1))
```

For example, we can look at the parameter evolution of the Nelson/Siegel model illustrated in Figure 5.

```
R> plot(param(ns_res))
```

Directly plotting the results object (`ns_res`) returns a 3D plot of the zero-coupon yield curve evolution created with the **rgl** package. We choose this over the usual `persp()` command,
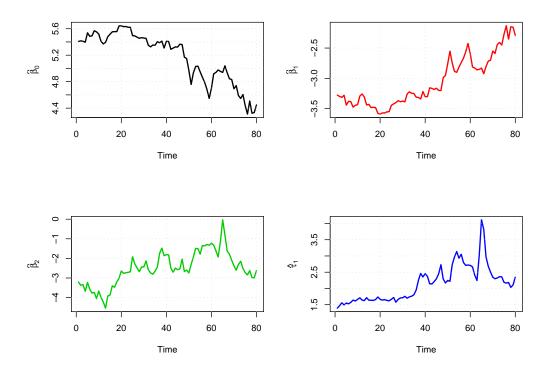
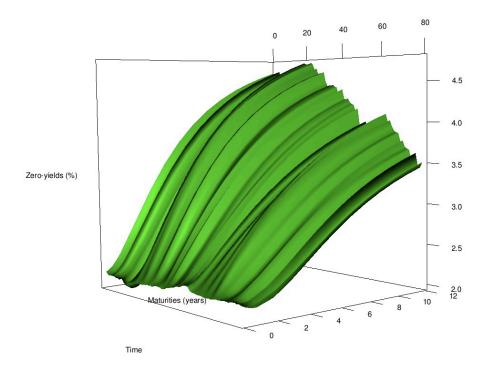Figure 5:   Evolution of the estimated Nelson/Siegel parameters.

Figure 6:   Evolution of the estimated zero-coupon yield curve.

because it is very convenient to freely zoom in and out and rotate the plot. Figure 6 shows the estimated three-dimensional zero-coupon yield curve.

```
R> plot(ns_res)
```

Finally, we compare the goodness of fit for the four estimated spot curve parameterizations.

```
R> allgof <- cbind(summary(dl_res)$gof, summary(ns_res)$gof,
+    summary(sv_res)$gof, summary(asv_res)$gof)
R> colnames(allgof) <- c("Diebold/Li", "Nelson/Siegel", "Svensson",
+    "Adj. Svensson")
R> print(allgof)
```

	Diebold/Li	Nelson/Siegel	Svensson	Adj. Svensson
RMSE-Yields (in %)	0.02273301	0.01516032	0.005816803	0.005921090
AABSE-Yields (in %)	0.01668111	0.01212619	0.004315130	0.004483215

Coupon bond data

The data set `GermanBonds` contains daily price observations for German coupon bonds for several months.

```
R> data("GermanBonds")
R> datadyncouponbonds
```

```
This is a dynamic data set of coupon bonds.
There are 65 observations between 2009-07-31 and 2009-11-02 .
```

```
R> class(datadyncouponbonds)
```

```
[1] "dyncouponbonds"
```

```
R> class(datadyncouponbonds[[1]])
```

```
[1] "couponbonds"
```

We begin with the estimation of the Diebold/Li and the Nelson/Siegel model.

```
R> dl_res <- estim_nss(datadyncouponbonds, "GERMANY", method = "dl",
+    lambda = 1/3)
R> ns_res <- estim_nss(datadyncouponbonds, "GERMANY", method = "ns",
+    tauconstr = list(c(0.2, 7, 0.2)), optimtype = "allglobal")
```

Here, we have choosen `optimtype = "allglobal"` because for this data set the β_2 parameter becomes very small, indicating a low factor factor contribution from the curvature and meaning that there is not much improvement in fit compared to the Diebold/Li parametrization with a fixed λ parameter, as we will later see in a plot of the factor contribution. We continue with the Svensson model. Again, the globally optimal start parameters are displayed during the estimation.

Figure 7: Start parameter grid search objective function for Svensson.

```
R> sv_res <- estim_nss(datadyncouponbonds, "GERMANY", method = "sv",
+    tauconstr = list(c(0.2, 7, 0.2, 0.5)))

[1] "Searching startparameters for  GERMANY"
    beta0      beta1      beta2       tau1      beta3       tau2
 5.832975  -4.405985  -7.373311   0.600100  -6.460257   3.200100
```

Figure 7 visualizes the start parameter grid search for the Svensson method by calling the `plot()` method of the `"spsearch"` object.

```
R> class(sv_res[[1]]$spsearch$GERMANY)
```

```
[1] "spsearch"
```

```
R> plot(sv_res[[1]]$spsearch$GERMANY)
```

Figure 8 exhibits the parameter evolution of the Svensson model. Again, we can obtain smooth time series and a reasonable performance in the grid search by setting appropriate constraints on the τ-parameters.
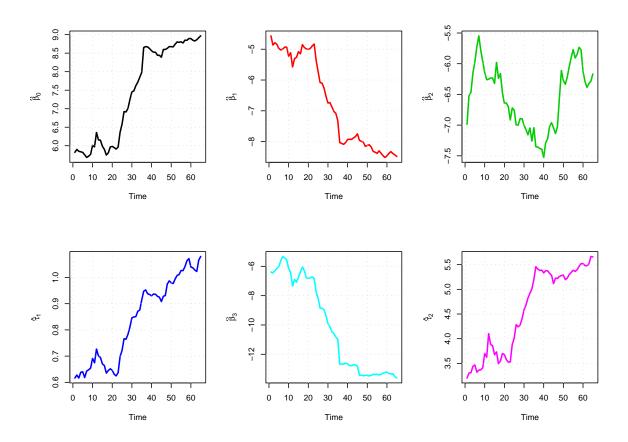
Figure 8: Evolution of the estimated Svensson parameters.

```
R> plot(param(sv_res))
```

Finally, we also estimate the Adjusted Svensson version for the dynamic coupon bonds data set.

```
R> asv_res <- estim_nss(datadyncouponbonds, "GERMANY", method = "asv",
+     tauconstr = list(c(0.2, 10, 0.2)))

  [1] "Searching startparameters for  GERMANY"
    beta0     beta1     beta2      tau1     beta3      tau2
 5.907375 -4.544134 -7.227484  0.600100 -4.029217  5.600100
```

An interesting method for detecting possible overparameterization is plotting the factor contributions. Figure 9 shows this for the first observation in the dynamic data set. The first panel shows the separation in level, slope and curvature of the Diebold/Li model. Comparing this to the Nelson/Siegel method in the second panel could let us conclude that including more parameters would just lead to an identification problem due to the small contribution of the second factor. However, we can see in the third and fourth panel that the six parameter
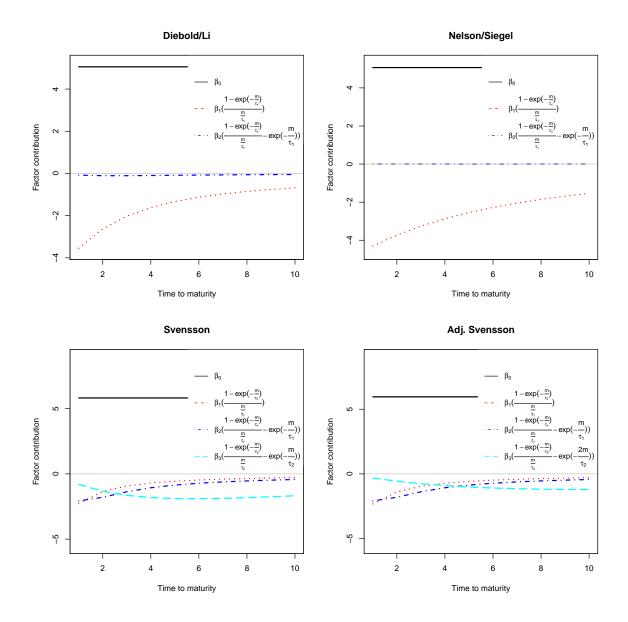
Figure 9: Factor contribution of several models.

models split the individual factor contributions in a different way. This is primarily caused by the added flexibility for a second hump in the curve.

```
R> par(mfrow = c(2, 2))
R> fcontrib(param(dl_res), index = 1, method = "dl")
R> fcontrib(param(ns_res), index = 1, method = "ns")
R> fcontrib(param(sv_res), index = 1, method = "sv")
R> fcontrib(param(asv_res), index = 1, method = "asv")
```

Finally, we compare the in-sample-fit. Even though it does not improve to a large degree when allowing a flexible τ_1 as in the Nelson/Siegel method, we see (as expected) much smaller

errors when increasing the the number of parameters to six. As seen before in the factor contribution plots, the model is not overparameterized in the Svensson case, and therefore, the RMSE improves drastically.

```
R> allgof <- cbind(summary(dl_res)$gof, summary(ns_res)$gof,
+    summary(sv_res)$gof, summary(asv_res)$gof)
R> colnames(allgof) <- c("Diebold/Li", "Nelson/Siegel", "Svensson",
+    "Adj. Svensson")
R> print(allgof)
```

	Diebold/Li	Nelson/Siegel	Svensson	Adj. Svensson
RMSE-Prices	0.239824311	0.186386090	0.0308522921	0.0308900064
AABSE-Prices	0.166937159	0.149307034	0.0228355326	0.0228308060
RMSE-Yields (in %)	0.056483846	0.052567693	0.0118737517	0.0118751464
AABSE-Yields (in %)	0.003190425	0.002763362	0.0001409860	0.0001410191

5. Conclusion

In this paper we discussed the two most widely-used parametric term structure estimation methods. We covered their estimation based on a set of coupon bonds which resulted in a nonconvex optimization problem with linear constraints. Due to the presence of multiple local optima, a global start parameter search algorithm was developed. The presented R extension package **termstrc** provides functions for the estimation of the zero-coupon yield curve based on static and dynamic coupon bond and zero yield data sets. Detailed examples illustrated the usage and functionality of the package.

Computational details

Acknowledgments

References

Adler D, Murdoch D (2010). *rgl: 3D Visualization Device System (OpenGL)*. R package version 0.91, URL `http://CRAN.R-project.org/package=rgl`.

Andrews DWK (1991). "Heteroskedasticity and Autocorrelation Consistent Covariance Matrix Estimation." *Econometrica*, **59**, 817–858.

Bank for International Settlements (2005). "Zero-Coupon Yield Curves: Technical Documentation." *BIS Papers 25*, Monetary and Economic Department. URL http://www.bis.org/publ/bppdf/bispap25.htm.

Björk T, Christensen BJ (1999). "Interest Rate Dynamics and Consistent Forward Rate Curves." *Mathematical Finance*, **9**(4), 323–348.

Black F, Karasinski P (1991). "Bond and Option Pricing when Short Rates Are Lognormal." *Financial Analysts Journal*, **46**, 52–59.

Bliss RR (1997). "Testing Term Structure Estimation Methods." *Advances in Futures and Options Research*, **9**, 197–231.

Bolder DJ, Streliski D (1999). "Yield Curve Modelling at the Bank of Canada." *Technical Report 84*, Bank of Canada. URL http://www.bankofcanada.ca/en/res/tr/1999/tr84-e.html.

Christensen JHE, Diebold FX, Rudebusch GD (2007). "The Affine Arbitrage-Free Class of Nelson-Siegel Term Structure Models." *Working Paper 2007-20*, Federal Reserve Bank of San Francisco. URL http://www.frbsf.org/publications/economics/papers/2007/wp07-20bk.pdf.

Christensen JHE, Diebold FX, Rudebusch GD (2009). "An Arbitrage-Free Generalized Nelson-Siegel Term Structure Model." *The Econometrics Journal*, **12**, 33–64.

Cox JC, Ingersoll JE, Ross SA (1985). "A Theory of the Term Structure of Interest Rates." *Econometrica*, **53**, 385–407.

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

De Pooter M (2007). "Examining the Nelson-Siegel Class of Term Structure Models: In-Sample Fit versus Out-of-Sample Forecasting Performance." *SSRN eLibrary*. http://ssrn.com/paper=992748.

Diebold FX, Li C (2006). "Forecasting the Term Structure of Government Bond Yields." *Journal of Econometrics*, **130**(2), 337–364.

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

Eddelbuettel D, Francois R (2010). ***Rcpp**: R/C++ Interface Package*. R package version 0.8.5, URL http://CRAN.R-project.org/package=Rcpp.

Eddelbuettel D, Nguyen K (2010). ***RQuantLib**: R Interface to the **QuantLib** Library*. R package version 0.3.2, URL http://CRAN.R-project.org/package=RQuantLib.

Fama EF, Bliss RR (1987). "The Information in Long-Maturity Forward Rates." *American Economic Review*, **77**(4), 680–692.

Ferstl R, Hayden J (2010). **termstrc:** *Zero-Coupon Yield Curve Estimation.* R package version 1.3.2, URL http://CRAN.R-project.org/package=termstrc.

Filipovic D (1999). "A Note on the Nelson-Siegel Family." *Mathematical Finance,* **9**(4), 349–359.

Fisher M, Nychka D, Zervos D (1995). "Fitting the Term Structure of Interest Rates with Smoothing Splines." *Finance and Economics Discussion Series 95-1*, Board of Governors of the Federal Reserve System (U.S.).

Greene WH (2002). *Econometric Analysis.* 5th edition. Prentice Hall, Upper Saddle River, NJ.

Guirreri SS (2010). **YieldCurve:** *Modelling and Estimation of the Yield Curve.* R package version 3.0, URL http://CRAN.R-project.org/package=YieldCurve.

Hagan PS, West G (2006). "Interpolation Methods for Curve Construction." *Applied Mathematical Finance,* **13**(2), 89–129.

Ho TSY, Lee SB (1986). "Term Structure Moevements and the Pricing of Interest Rate Contingent Claims." *The Journal of Finance,* **41**, 1011–1029.

Hull J, White A (1990). "Pricing Interest-rate Derivative Securities." *The Review of Financial Studies,* **3**(4), 573–592.

Iacus SM (2009). **sde:** *Simulation and Inference for Stochastic Differential Equations.* R package version 2.0.10, URL http://CRAN.R-project.org/package=sde.

Iacus SM (2010). **yuima:** *The Project for Simulation and Inference of Multidimensional Stochastic Differential Equations with Abstract Noise.* URL http://R-Forge.R-project.org/projects/yuima/.

Ioannides M (2003). "A Comparison of Yield Curve Estimation Techniques Using UK Data." *Journal of Banking & Finance,* **27**(1), 1–26.

Martellini L, Priaulet P, Priaulet S (2003). *Fixed-Income Securities: Valuation, Risk Management and Portfolio Strategies.* John Wiley & Sons, Hoboken.

McCulloch JH (1971). "Measuring the Term Structure of Interest Rates." *The Journal of Business,* **44**(1), 19–31.

McCulloch JH (1975). "The Tax-Adjusted Yield Curve." *The Journal of Finance,* **30**(3), 811–830.

Nawalkha SK, Beliaeva NA, Soto GM (2007). *Dynamic Term Structure Modeling.* John Wiley & Sons, Hoboken.

Nawalkha SK, Soto GM, Beliaeva NA (2005). *Interest Rate Risk Modeling : The Fixed Income Valuation Course.* John Wiley & Sons, Hoboken.

Nelson CR, Siegel AF (1987). "Parsimonious Modeling of Yield Curves." *The Journal of Business,* **60**(4), 473–489.

Pfaff B (2008). *Analysis of Integrated and Cointegrated Time Series with R.* 2nd edition. Springer-Verlag, New York.

R Development Core Team (2010). *R: A Language and Environment for Statistical Computing.* R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, URL http://www.R-project.org/.

Svensson LEO (1994). "Estimating and Interpreting Forward Interest Rates: Sweden 1992–1994." *Technical Reports 4871*, National Bureau of Economic Research, Inc.

The MathWorks, Inc (2010). ***Fixed-Income** Toolbox 1.9: Model and Analyze Fixed-Income Securities.* The MathWorks, Inc., Natick, Massachusetts. URL http://www.mathworks.com/products/fixedincome/.

Theußl S, Zeileis A (2009). "Collaborative Software Development Using R-Forge." *The R Journal*, **1**(1), 9–14. URL http://journal.R-project.org/2009-1/RJournal_2009-1_Theussl+Zeileis.pdf.

Uhlenbeck GE, Ornstein LS (1930). "On the Theory of Brownian Motion." *Physical Review*, **36**, 823–841.

Vasicek O (1977). "An Equilibrium Characterisation of the Term Structure." *Journal of Financial Economics*, **5**, 177–188.

Vasicek O, Fong HG (1982). "Term Structure Modeling Using Exponential Splines." *The Journal of Finance*, **37**(2), 339–348.

Werner R, Ferenczi I (2006). "Calibration of the Svensson Model to Simulated Yield Curves." In *MathFinance Workshop, Frankfurt*. URL http://workshop.mathfinance.com/2006/papers/werner/slides.pdf.

Wuertz D (2009). ***fBonds**: Bonds and Interest Rate Models.* R package version 2100.75, URL http://CRAN.R-project.org/package=fBonds.

Zeileis A (2004). "Econometric Computing with HC and HAC Covariance Matrix Estimators." *Journal of Statistical Software*, **11**(10), 1–17. URL http://www.jstatsoft.org/v11/i10/.

Zeileis A (2006). "Object-Oriented Computation of Sandwich Estimators." *Journal of Statistical Software*, **16**(9), 1–16. URL http://www.jstatsoft.org/v16/i09/.

Zeileis A, Grothendieck G (2005). "**zoo**: S3 Infrastructure for Regular and Irregular Time Series." *Journal of Statistical Software*, **14**(6), 1–27. URL http://www.jstatsoft.org/v14/i06/.

Zeileis A, Hothorn T (2002). "Diagnostic Checking in Regression Relationships." *R News*, **2**(3), 7–10. URL http://CRAN.R-project.org/doc/Rnews/.

A. Analytical gradients of the objective functions

We consider the two different objective functions defined in (5) and report upon the associated elements of the gradient $\boldsymbol{\nabla} F(\boldsymbol{\beta})$. In the following "·" denotes an element wise multiplication. Divisions and the function $\exp(\cdot)$ are also applied to each element of a matrix or vector. If not explicitly stated the elements are valid for the DL, NS, SV and adjusted SV model.

In the case of a term structure estimation based on coupon bond prices the elements of the gradient $\boldsymbol{\nabla} F(\boldsymbol{\beta})$, with $F(\boldsymbol{\beta}) = \left(\boldsymbol{p} - \boldsymbol{\iota}^\top (\boldsymbol{C} \cdot \boldsymbol{D}) \right)^2 \boldsymbol{\omega}^\top$, are given by

$$\frac{\partial F}{\partial \beta_1} = \boldsymbol{\iota}^\top \left(\boldsymbol{b} \cdot \left(-\boldsymbol{A} \cdot \boldsymbol{C} \cdot \boldsymbol{M} \frac{1}{100} \right) \right),$$

$$\frac{\partial F}{\partial \beta_2} = \boldsymbol{\iota}^\top \left(\boldsymbol{b} \cdot \left(-\boldsymbol{A} \cdot \boldsymbol{C} \frac{1}{100} \tau_1 \left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) \right) \right),$$

$$\frac{\partial F}{\partial \beta_3} = \boldsymbol{\iota}^\top \left(\boldsymbol{b} \cdot \left(-\boldsymbol{A} \cdot \boldsymbol{C} \cdot \boldsymbol{M} \frac{1}{100} \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) \tau_1}{\boldsymbol{M}} - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) \right) \right),$$

$$\frac{\partial F}{\partial \tau_1} = \boldsymbol{\iota}^\top \left(\boldsymbol{b} \cdot \left(-\boldsymbol{A} \cdot \boldsymbol{C} \cdot \boldsymbol{M} \frac{1}{100} \left(-\beta_2 \frac{\exp(-\frac{\boldsymbol{M}}{\tau_1})}{\tau_1} + \beta_2 \frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right)}{\boldsymbol{M}} + \right.\right.\right.$$
$$\left.\left.\left. +\beta_3 \left(\frac{1 - \exp(-\frac{\boldsymbol{M}}{\tau_1})}{\boldsymbol{M}} - \frac{\exp(-\frac{\boldsymbol{M}}{\tau_1})}{\tau_1} - \frac{\exp(-\frac{\boldsymbol{M}}{\tau_1}) \cdot \boldsymbol{M}}{\tau_1^2} \right) \right) \right) \right),$$

$$\frac{\partial F}{\partial \beta_4} = \begin{cases} \boldsymbol{\iota}^\top \left(\boldsymbol{b} \cdot \left(-\boldsymbol{A} \cdot \boldsymbol{C} \cdot \boldsymbol{M} \frac{1}{100} \left(-\exp(-\frac{\boldsymbol{M}}{\tau_2}) + \frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_2}) \right) \tau_2}{\boldsymbol{M}} \right) \right) \right) & \text{SV,} \\[2em] \boldsymbol{\iota}^\top \left(\boldsymbol{b} \cdot \left(-\boldsymbol{A} \cdot \boldsymbol{C} \cdot \boldsymbol{M} \frac{1}{100} \left(-\exp(-\frac{2\boldsymbol{M}}{\tau_2}) + \frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_2}) \right) \tau_2}{\boldsymbol{M}} \right) \right) \right) & \text{adj. SV,} \end{cases}$$

$$\frac{\partial F}{\partial \tau_2} = \begin{cases} \boldsymbol{\iota}^\top \left(\boldsymbol{b} \cdot \left(-\boldsymbol{A} \cdot \boldsymbol{C} \cdot \boldsymbol{M} \frac{1}{100} \beta_4 \left(-\frac{\exp(-\frac{\boldsymbol{M}}{\tau_1})}{\tau_1} + \frac{1 - \exp(-\frac{\boldsymbol{M}}{\tau_1})}{\boldsymbol{M}} - \frac{\exp(-\frac{\boldsymbol{M}}{\tau_1}) \cdot \boldsymbol{M}}{\tau_1^2} \right) \right) \right) & \text{SV,} \\[2em] \boldsymbol{\iota}^\top \left(\boldsymbol{b} \cdot \left(-\boldsymbol{A} \cdot \boldsymbol{C} \cdot \boldsymbol{M} \frac{1}{100} \beta_4 \left(-\frac{\exp(-\frac{\boldsymbol{M}}{\tau_1})}{\tau_1} + \frac{1 - \exp(-\frac{\boldsymbol{M}}{\tau_1})}{\boldsymbol{M}} - 2\frac{\exp(-\frac{2\boldsymbol{M}}{\tau_1}) \cdot \boldsymbol{M}}{\tau_1^2} \right) \right) \right) & \text{adj. SV.} \end{cases}$$

$\boldsymbol{b}$ and $\boldsymbol{A}$ are defined as

$$\boldsymbol{b} = -2\boldsymbol{\omega} \cdot \left(\boldsymbol{p} - \boldsymbol{\iota}^\top (\boldsymbol{A} \cdot \boldsymbol{C}) \right),$$

$$\boldsymbol{A} = \begin{cases} \exp \left(-\frac{\boldsymbol{M}}{100} \cdot \left(\beta_1 + \beta_2 \tau_1 \frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right)}{\boldsymbol{M}} + \beta_3 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) \tau_1}{\boldsymbol{M}} - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) \right) \right) & \text{NS/DL,} \\[2em] \exp \left(-\frac{\boldsymbol{M}}{100} \cdot \left(\beta_1 + \beta_2 \tau_1 \frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right)}{\boldsymbol{M}} + \beta_3 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) \tau_1}{\boldsymbol{M}} - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) + \right.\right. & \text{SV,} \\[1em] \left.\left. \quad +\beta_4 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_2}) \right) \tau_2}{\boldsymbol{M}} - \exp(-\frac{\boldsymbol{M}}{\tau_2}) \right) \right) \right) \\[2em] \exp \left(-\frac{\boldsymbol{M}}{100} \cdot \left(\beta_1 + \beta_2 \tau_1 \frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right)}{\boldsymbol{M}} + \beta_3 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) \tau_1}{\boldsymbol{M}} - \exp(-\frac{\boldsymbol{M}}{\tau_1}) \right) + \right.\right. & \text{adj. SV.} \\[1em] \left.\left. \quad + \beta_4 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{M}}{\tau_2}) \right) \tau_2}{\boldsymbol{M}} - \exp(-\frac{2\boldsymbol{M}}{\tau_2}) \right) \right) \right) \end{cases}$$

For a term structure estimation based on zero yields the elements of the gradient $\nabla F(\boldsymbol{\beta})$, with $F(\boldsymbol{\beta}) = (\boldsymbol{y} - s(\boldsymbol{m}, \boldsymbol{\beta}))^2 \, \boldsymbol{\iota}$, are given by

$$\frac{\partial F}{\partial \beta_1} = (-2(\boldsymbol{y} - \boldsymbol{b})) \, \boldsymbol{\iota},$$

$$\frac{\partial F}{\partial \beta_2} = \left(\frac{-2\tau_1(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1}))}{\boldsymbol{m}} \cdot (\boldsymbol{y} - \boldsymbol{b}) \right) \boldsymbol{\iota},$$

$$\frac{\partial F}{\partial \beta_3} = \left(-2 \left(\frac{\tau_1 \left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1}) \right)}{\boldsymbol{m}} - \exp(-\frac{\boldsymbol{m}}{\tau_1}) \right) \cdot (\boldsymbol{y} - \boldsymbol{b}) \right) \boldsymbol{\iota},$$

$$\frac{\partial F}{\partial \tau_1} = \left(2 \left(-\frac{\beta_2 \left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1}) \right)}{\boldsymbol{m}} + \frac{\exp(-\frac{\boldsymbol{m}}{\tau_1})\beta_2}{\tau_1} - \right. \right.$$
$$\left. \left. -\beta_3 \left(\frac{1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})}{\boldsymbol{m}} - \frac{\exp(-\frac{\boldsymbol{m}}{\tau_1})}{\tau_1} - \frac{\exp(-\frac{\boldsymbol{m}}{\tau_1}) \cdot \boldsymbol{m}}{\tau_1^2} \right) \right) \cdot (\boldsymbol{y} - \boldsymbol{b}) \right) \boldsymbol{\iota},$$

$$\frac{\partial F}{\partial \beta_4} = \begin{cases} \left(-2 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_2})\right)\tau_2}{\boldsymbol{m}} - \exp(-\frac{\boldsymbol{m}}{\tau_2}) \right) \cdot (\boldsymbol{y} - \boldsymbol{b}) \right) \boldsymbol{\iota} & \text{SV}, \\[3ex] \left(-2 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_2})\right)\tau_2}{\boldsymbol{m}} - \exp(-\frac{2\boldsymbol{m}}{\tau_2}) \right) \cdot (\boldsymbol{y} - \boldsymbol{b}) \right) \boldsymbol{\iota} & \text{adj. SV}, \end{cases}$$

$$\frac{\partial F}{\partial \tau_2} = \begin{cases} \left(-2\beta_4 \left(\frac{1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})}{\boldsymbol{m}} - \frac{\exp(-\frac{\boldsymbol{m}}{\tau_1})}{\tau_1} - \frac{\exp(-\frac{\boldsymbol{m}}{\tau_1}) \cdot \boldsymbol{m}}{\tau_1^2} \right) \cdot (\boldsymbol{y} - \boldsymbol{b}) \right) \boldsymbol{\iota} & \text{SV}, \\[3ex] \left(-2\beta_4 \left(\frac{1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})}{\boldsymbol{m}} - \frac{\exp(-\frac{\boldsymbol{m}}{\tau_1})}{\tau_1} - \frac{\exp(-\frac{2\boldsymbol{m}}{\tau_1}) \cdot \boldsymbol{m}}{\tau_1^2} \right) \cdot (\boldsymbol{y} - \boldsymbol{b}) \right) \boldsymbol{\iota} & \text{adj. SV}. \end{cases}$$

$\boldsymbol{b}$ is defined as

$$\boldsymbol{b} = \begin{cases} -\beta_1 - \beta_2\tau_1 \frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})\right)}{\boldsymbol{m}} - \beta_3 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})\right)\tau_1}{\boldsymbol{m}} - \exp(-\frac{\boldsymbol{m}}{\tau_1}) \right) & \text{NS/DL}, \\[3ex] -\beta_1 - \beta_2\tau_1 \frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})\right)}{\boldsymbol{m}} - \beta_3 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})\right)\tau_1}{\boldsymbol{m}} - \exp(-\frac{\boldsymbol{m}}{\tau_1}) \right) - & \text{SV}, \\ \quad -\beta_4 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_2})\right)\tau_2}{\boldsymbol{m}} - \exp(-\frac{\boldsymbol{m}}{\tau_2}) \right) \\[3ex] -\beta_1 - \beta_2\tau_1 \frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})\right)}{\boldsymbol{m}} - \beta_3 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_1})\right)\tau_1}{\boldsymbol{m}} - \exp(-\frac{\boldsymbol{m}}{\tau_1}) \right) - & \text{adj. SV}. \\ \quad -\beta_4 \left(\frac{\left(1 - \exp(-\frac{\boldsymbol{m}}{\tau_2})\right)\tau_1}{\boldsymbol{m}} - \exp(-\frac{2\boldsymbol{m}}{\tau_2}) \right) \end{cases}$$

Affiliation:

Robert Ferstl, Josef Hayden
Department of Finance
University of Regensburg
93053 Regensburg, Germany
E-mail: robert.ferstl@wiwi.uni-regensburg.de
        josef.hayden@wiwi.uni-regensburg.de
URL: http://www-finance.uni-regensburg.de/