The Monotone Convex Method Of Interpolation-Applications to the Construction of the Yield Curve Using Octave

Phumlani Lawrence Zondi (phumlanizondi@aims.ac.za)

African Institute for Mathematical Sciences (AIMS)

Supervised by: Emeritus Professor Ronnie Becker
AIMS, South Africa

22 May 2014

Submitted in partial fulfillment of a structured masters degree at AIMS South Africa

Abstract

The yield curve is one of the most significant tools used by financial institutions in modern days. Building such a curve is a process that requires a careful approach. The problem when building yield curves is the scarcity of market data. In most cases we rely on the approach known as *bootstrapping* which takes as inputs the few instruments that can be observed in markets and utilizes them to build yield curves. Bootstrapping is strongly controlled by the process of interpolation which is used as the mechanism for constructing new data points. The problem is that well-known interpolation methods are not compatible with bootstrapping. This essay is devoted into providing an in-depth review of the most suitable and promising method of interpolation known as the monotone convex method. We also analyse this method and prove its suitability in the construction of yield curves.

Key Words: Yield curves, bootstrapping, zero-coupon bonds, spot rates, interpolation.

Declaration

I, the undersigned, hereby declare that the work contained in this research project is my original work, and that any work done by others or by myself previously has been acknowledged and referenced accordingly.

Phumlani Lawrence Zondi, 22 May 2014

Contents

1. The purpose and introduction

The yield curve is a plot showing interest rates across different contract lengths known as maturities. In normal circumstances the yield curve is an increasing function of time. When interest rates change they affect the shape and slope of the yield curve. By studying various aspects of the yield curves analysts gain insight about the future of interest rates. For instance if the yield curve is upwardly sloping it suggests that there would be high interest rates in the future.

Yield curves have extensive applications in finance. They play roles in predicting interest rates, pricing of bonds, establishing fixed income trading strategies, gauging of risks and many other areas.

1.1 Interpolation problem and its significance in the yield curve construction

In many branches of science like mathematics, physics, and computer sciences we often encounter situations where only a limited amount of data is obtainable and it is necessary to estimate intermediate values between two consecutive given data points. In many instances this is overcome by using the mathematical tool known as interpolation. Interpolation is a methodology of constructing new points between known data points.

The basic interpolation problem states: Given $N+1$ set of data points $(x_i, y_i) \in \mathbb{R}^2$ for $i = 0, ..., N$, find a function $h : [x_0, x_N] \to \mathbb{R}$ such that $h(x_i) = y_i$ for all i. The function which results from interpolation is used to estimate a structure hidden behind a set of data. There is a rich variety of interpolation methods available. Amongst these are linear interpolation, polynomial interpolation and spline interpolation.

When constructing yield curves we encounter the problem of interpolation. In financial markets there is a very limited number of securities that are suitable for constructing yield curves. Thus, there is a need to come up with the interpolation method that will construct the curve using the limited data available. The purpose of this dissertation is to give a comprehensive review of the method of interpolating yield curves called the monotone convex method introduced by (Hagan and West, 2006). This method falls under the piecewise polynomial interpolations.

Interpolating yield curve is a very critical process. It is required that the curve resulting from interpolation be a smooth curve. We know from calculus that one condition for smoothness is to ensure the continuity in derivatives. Another desirable feature of the interpolation method in the case of yield curve construction is that it be able to result in a curve that will be able to exactly price back all the input securities that were used to construct it. The development of the monotone convex method was inspired by the need to satisfy these constraints and others which we will explore in the next chapters.

1.2 Basic concepts and the mathematics of yield curves

Bond-is a type of investment that represents a loan between a borrower and a lender.

Maturity date-is the date that the debt will cease to exist, at which time the borrower pays to the lander the amount borrowed plus interest.

Coupon rate/ nominal rate-is the interest rate that the borrower agrees to pay the lender each year. When the coupon rate is not fixed during the life of a bond it is known as the floating rate. It is called floating rate in a sense that it changes with future changes in interest rates.

Coupon-paying bond-is an instrument that pays periodic interest payments (called coupons) to the holder of the bond, and an additional value at maturity called the face value, par value, or principal value.

Settlement date-is the date by which a buyer settles the owed amount to the seller.

Accrued interest-is the interest that has already been earned on an investment but is yet to be paid by a debtor.

Clean price of a bond-is the price of a bond that is calculated without taking accrued interest into accounts.

A swap- is a type of contract between two parties that involves an agreement to exchange cash flows in the future. The payment date is specified when the agreement takes place. (Hull, 2006).

Notional principal amount-is a fixed amount on which the the fixed rat and floating rate are calculated in a Swap contract.

A Forward Rate Agreement (FRA)-"is an over-the-counter agreement that a certain interest rate will apply to either borrowing or lending a certain principal during a specified future period of time" (Hull, 2006).

The yield curve is connected to the concept of interest rate. Interest rates can either be discretely compounded or continuously compounded.

Continuously compounding

We know from basic financial mathematics that

$$FV = P_0 \left(1 + \frac{i}{m}\right)^{mt}. \tag{1.2.1}$$

Where FV denotes the future amount of an investment, P_0 is the present value, i is the rate of return on an investment, m is the number of compounding periods each year, and t is the number of years that an investment takes. Continuous compounding arises when we take the limit as m approaches infinity in our formula above. So this means that the interest is compounded at every instant, rather than fixed intervals (Lee. C and Lee. A, 2006). We know by the standard formula from calculus that:

$$e^i = \lim_{m\to\infty} \left(1 + \frac{i}{m}\right)^m, \tag{1.2.2}$$

so

$$\lim_{m\to\infty} P_0 \left(1 + \frac{i}{m}\right)^{mt} = P_0 \left(\lim_{m\to\infty} \left(1 + \frac{i}{m}\right)^m\right)^t \tag{1.2.3}$$

$$= P_0 e^{it}. \tag{1.2.4}$$

Equation (1.2.4) is known as the future-value formula for the loan that is subject to the continuously compounded interest rate. Much of the mathematics associated with zero-coupon yield curves is clearer when working with continuously compounded rates. Thus, throughout this dissertation, unless explicitly stated otherwise, we will assume that all rates are continuously compounded. Time is measured in years, the current time is denoted by t_0 (we will take t_0 to be zero for simplicity), and the time to maturity is denoted by t.

Zero-coupon bond/ Discount factor-is the instrument that pays 1 unit of currency at time $t \geq t_0$. We denote the price of a zero-coupon bond by $Z(t_0, t)$. Mathematically it is expressed as

$$Z(t_0, t) = \exp(-r(t_0, t)t), \tag{1.2.5}$$

which can rewritten as

$$r(t_0, t) = -\frac{1}{t}\ln\left(Z(t_0, t)\right). \tag{1.2.6}$$

$r(t_0, t)$ is called the zero-coupon bond yield or zero-coupon bond rate.

Forward rates

Suppose we can borrow at a known rate at time t_0 to time t_1, and we can borrow from t_1 to t_2 at a rate known and fixed rate at time t_0, it is possible then to borrow at a known rate at time t_0 until t_2. It follows by the no arbitrage argument that we should have:

$$Z(t_0, t_1)Z(t_0; t_1, t_2) = Z(t_0, t_2), \tag{1.2.7}$$

where $Z(t_0; t_1, t_2)$ is the forward discount factor for the period from t_1 to t_2. The forward rate governing the period from t_1 to t_2, denoted $f(t_0; t_1, t_2)$ satifies

$$\exp(-f(t_0; t_1, t_2))(t_2 - t_1) = Z(t_0; t_1, t_2). \tag{1.2.8}$$

Immediately, we see that forward rates are positive (West, 2011). We can combine (1.2.7) and (1.2.8) to obtain

$$\exp(-f(t_0; t_1, t_2)(t_2 - t_1)) = \frac{Z(t_0, t_2)}{Z(t_0, t_1)} \tag{1.2.9}$$

$$\Rightarrow f(t_0; t_1, t_2) = -\frac{\ln\left(Z(t_0, t_2)\right) - \ln\left(Z(t_0, t_1)\right)}{t_2 - t_1} \tag{1.2.10}$$

$$= \frac{r(t_0, t_2)t_2 - r(t_0, t_1)t_1}{t_2 - t_1}. \tag{1.2.11}$$

For simplicity we can write $r(t_0, t_1)$ and $r(t_0, t_2)$ as r_1 and r_2, respectively. So that (1.2.11) becomes

$$f(t_0; t_1, t_2) = \frac{r_2 t_2 - r_1 t_1}{t_2 - t_1}. \tag{1.2.12}$$

Instantaneous forward rate

The above defines the discretely compounded forward interest rate. We can also define the continuously compounded or instantaneous forward rate. Let ϵ be a small change in t, we define the instantaneous forward rate of t to be

$$
\begin{aligned}
f(t) &= \lim_{\epsilon\to 0} f(t_0; t, t+\epsilon) && (1.2.13)\\
&= \lim_{\epsilon\to 0} \frac{r(t_0, t+\epsilon)(t+\epsilon) - r(t_0,t)t}{t+\epsilon-t} && (1.2.14)\\
&= \lim_{\epsilon\to 0} \frac{r(t_0, t+\epsilon)\epsilon + t(r(t_0,t+\epsilon) - r(t_0,t))}{\epsilon} && (1.2.15)\\
&= \left[\frac{r(t_0,t+\epsilon)\epsilon}{\epsilon}\right]_{\epsilon=0} + t\lim_{\epsilon\to 0}\frac{r(t_0,t+\epsilon)-r(t_0,t)}{\epsilon} && (1.2.16)\\
&= r(t_0,t) + \frac{dr(t_0,t)}{dt}t && (1.2.17)\\
&= \frac{d}{dt}\left(r(t_0,t)t\right) \quad . && (1.2.18)
\end{aligned}
$$

This is true only if $\left(\frac{dr(t_0,t)}{dt}\right)$ exists for all time. Integrating from 0 to t on both sides we obtain

$$r(t_0,t)t = \int_0^t f(s)ds.$$

By $(1.2.1)$ we have

$$Z(t_0,t) = \exp\left(-\int_0^t f(s)ds\right). \tag{1.2.19}$$

We can change the limits of our integral so that

$$r(t_0,t)t - r(t_0,t_{i-1})t_{t_{i-1}} = \int_{t_{i-1}}^t f(s)ds.$$

Which gives

$$r(t_0,t)t = r(t_0,t_{i-1})t_{t_{i-1}} + \int_{t_{i-1}}^t f(s)ds,$$

and consequently

$$\frac{r_i t_i - r_{i-1}t_{i-1}}{t_i - t_{i-1}} = \frac{1}{t_i - t_{i-1}}\int_{t_{i-1}}^{t_i} f(s)ds. \tag{1.2.20}$$

Note that the left hand side of (1.2.20) is similar to the equation for the discrete forward rate we established in (1.2.12) above, the only difference is that (1.2.20) is defined on the generic interval $[t_{i-1}, t_i]$. We can therefore define

$$f_i^d := \frac{r_i t_i - r_{i-1}t_{i-1}}{t_i - t_{i-1}} = \frac{1}{t_i - t_{i-1}}\int_{t_{i-1}}^{t_i} f(s)ds. \tag{1.2.21}$$

From (1.2.21) it transpires that if you average the instantaneous forward rates over any arbitrary interval $[t_{i-1}, t_i]$, you would get the discrete forward rate corresponding to that interval.

1.3 Ensuring arbitrage-free yield curve construction

At any given time, the discount factor is lower, the longer the maturity t. That is, given two dates t_1 and t_2 with $t_1 \leq t_2$, it is always the case that

$$Z(t_0, t_1) \geq Z(t_0, t_2). \tag{1.3.1}$$

The violation of relation (1.3.1) generates an arbitrage opportunity (Veronesi, 2010). To prove our assertion we consider the following situation: Suppose $Z(t_0, t)$ satisfies $Z(t_0, t_1) < Z(t_0, t_2)$ for some $t_1 < t_2$. Then an arbitrageur can buy a zero-coupon bond for t_1 and sell it for t_2, for an immediate profit of $(Z(t_0, t_2) - Z(t_0, t_1))$. At time t_1 they would receive 1 unit of currency from the bond they have bought, which they could keep until time t_2, and pay it to the buyer of the time t_2 bond, at time t_2. Requiring a decreasing zero-coupon curve is synonymous to requiring positive forwards. So we impose the positivity of forward rates as a condition to guarantee no arbitrage opportunities (West, 2011).

2. Bootstrapping the yield curve

2.1 Zero-coupon yield curve models

The determination of the yield curve is a difficult process. One problem that emerges when constructing yield curves, as we have mentioned in Chapter 1 is that we do not know beforehand the yield curve function for a sufficiently wide range of maturities. In many instances they are known only for a few maturities. There is a need for a mathematical model that will interpolate between the few data points available and build a smooth zero-coupon yield curve. The models of this kind have been the subject of research for some time. Two types of models are parametric and spline-based models.

2.1.1 Parametric models. The idea is to describe the yield curve as a function of some parameters. These parameters are estimated using techniques such as least-squares regression. Some prominent works in this field are (Nelson, 1987) and (Svensson, 1992).

In practice, when parametric models are employed, they can give unreasonable results, so financial institutions rarely use them. For the remainder of this dissertation we will confine ourselves on the spline- based models.

2.1.2 Spline-Based Models. The yield curve is made up of piecewise polynomials. These polynomials join together at some points in time called knot points. So the idea is to choose the knot points (these knot points are maturity times), and after that to extract the set of spot rates corresponding to the chosen maturities and then eventually interpolate to get the yield curve. Consequently this presents us with two challenges; finding the right method to extract spot rates and choosing the right method of interpolation. The major part of this dissertation is devoted in discussing these challenges and showing how to tackle them.

2.2 Using bootstrapping to extract zero-coupon rates

Different methods for computing spot rates from the set of maturity dates have been proposed. (McCulloch J, 1975) suggested the use of a multivariate optimisation routine to establish the set of spot rates. This routine can be combined with an appropriate method of interpolation to produce a yield curve. The problem with this method is that it fails to price back all the inputs and it is complex. Alternatively, we describe another convenient and efficient approach known as *bootstrapping*. This method employs an iterative technique which converges to the required spot rates corresponding to the specified maturity dates. Bootstrapping was first introduced by (Fama and Bliss, 1987). Their work was later generalised by (Hagan and West, 2006) among others.

What bootstrapping does is it uses the available coupon-bearing securities to estimate zero-coupon bond prices. Bootstrapping is based on the notion that any security can be regarded as collection of zero-coupon bonds. A par rate can be seen as the average discount rate of many cash flows over many periods, so the bootstrapping process involves breaking apart these par rates to uncover spot rates. If we want to apply this technique we need to decide on a set of benchmark securities from which we will extract spot rates. Furthermore, it is worth noting that if a yield curve has been constructed using a certain security (through bootstrapping) then it would in turn serve as a benchmark for pricing that particular security.

2.3 Choosing benchmark securities

We distinguish between sovereign yield curves, and interbank yield curves. The standard approach when calibrating sovereign yield curves is to make use of government bonds. For this reason, these curves are often referred to as bond curves. When calibrating interbank yield curves, the standard approach is to use FRAs and swaps. For this reason, interbank curves are often referred to as Swap curves (Du Preez, 2011). So among others, the coupon bonds, swaps and FRAs are found to be convenient benchmark securities for the construction of yield curves. Below we study these securities and observe their link to yield curves.

2.4 Coupon bonds and their connection to yield curves

The purchasers of coupon bonds get periodic coupon payments until the bond reach maturity. At the end of bond's lifetime they receive the face value of the bond and the last coupon payment. The price of a bond that pays annual coupons can therefore be given by (2.4.1).

$$\mathbb{A} = \sum_{j=1}^{n-1} p\frac{1}{(1+r)^j} + (W+p)\frac{1}{(1+r)^n}, \tag{2.4.1}$$

for all $j \in \mathbb{R}$. where:

$\mathbb{A}$ is the market price of the bond.

p is the annual coupon payment.

r is the interest rate

W is the par value of the bond n is the number of years to maturity

Note that equation (2.4.1) calculates the clean price on a coupon payment date, so there is no accrued interest incorporated into the price. The accrued interest can be added to the clean price to obtain the market value of bond, known as the dirty price. In this essay we will only consider the clean price of a coupon bond.

We can discount continuously in (2.4.1) to obtain (2.4.2) below.

$$\mathbb{A} = \sum_{j=1}^{n-1} pe^{-rj} + (W+p)e^{-rn}. \tag{2.4.2}$$

Suppose the coupon payments are made at times $t_1, t_2, ..., t_n$, equation (2.4.2) can be reformulated into (2.4.3)

$$\mathbb{A} = \sum_{j=1}^{n-1} pe^{-r(t_j)t_j} + (W+p)e^{-r(t_n)t_n}. \tag{2.4.3}$$

Therefore the price of an arbitrary coupon-bearing bond is equal to the sum of n cashflows, discounted at the particular zero-coupon rate r. From (2.4.3) we have that

$$e^{-r(t_n)t_n} = \left[\frac{\mathbb{A} - \sum_{j=1}^{n-1} pe^{-r(t_j)t_j}}{W+p}\right], \tag{2.4.4}$$

which can be rewritten in terms of discount factors as

$$Z(t_n) = \left[\frac{\mathbb{A} - \sum_{j=1}^{n-1} pZ(t_j)}{W + p} \right]. \tag{2.4.5}$$

Let us analyse what equation (2.4.5) actually says. It says that if we know the zero-coupon rate $r(t_n)$ corresponding to t_n and if we know the information on the right hand side we can determine the zero-coupon bond price $Z(t_n)$ corresponding to the time t_n. Therefore we have shown that it is possible to find zero-coupon bond price from the coupon-bearing bonds. But the problem with equation (2.4.5) is that it assumes we have all the zero coupon rates for all times $t_1, t_2, ..., t_n$. In fact this is not the case. Usually for short term maturities like 0.5 years and 1 year, bootstrapping relies on using instruments known as money market instruments. These are types of debt instruments with maturities less than or equal to 1 year. Money markets instruments do not receive coupons, instead they receive a fixed interest rates at their maturities. This means that we can use their interest rates as input data in (2.4.5). The rest of the zero-coupon rates corresponding to maturities longer than 1 year may be guessed, as a starting point to bootstrapping.

Note that we can also solve for $r(t_n)$ in (2.4.4) to obtain

$$r(t_n) = -\frac{1}{t_n} \ln \left[\frac{\mathbb{A} - \sum_{j=1}^{n-1} pe^{-r(t_j)t_j}}{W + p} \right]. \tag{2.4.6}$$

The bootstrapping algorithm will involve using equation (2.4.6) iteratively to find zero-coupon rates.

2.5 Swaps and their connection to yield curves

A Swap is an agreement between two parties to exchange sequences of cash flows for a period of time in future. The simplest and widely known type of Swaps is the Plain Vanilla Interest Rate Swap. In this Swap one party agrees to pay coupons for a certain period of time on a fixed rate. This fixed rate is calculated using a notional principal amount. In return the party receives from another counter party cash at a floating rate which is calculated using the same notional principal for the same period of time (Hull, 2006). So this Swap consists of two legs of transaction; the fixed-rate leg and the floating-rate leg. To price this Swap one needs to calculate first the present value of cash flows for each leg of transaction.

The fixed-rate leg of a Swap- The coupon rate is set at the time of agreement and never changes. So the present value of a fixed leg is defined to be

$$V_{fix} := \sum_{k=1}^{n} N_a.R.\alpha_k.Z(t_0, t_k). \tag{2.5.1}$$

where:

V_{fix}- is the present value of cash flows for the fixed-rate leg.

N_a- is the notional principal amount of a Swap.

n- is the number of coupons payable between value date and maturity date.

R- is the fixed coupon rate.

α_k- is the fraction of a year between $(k-1)th$ and $(k)th$ payment (calculated on an agreed day-convention).

$Z(t_0, t_k)$- is the factor discounting each cash flow on cash flow k.

For simplicity we assume that the notional amount N_a is equal to 1, and hence we have

$$V_{fix} = \sum_{k=1}^{n} R\alpha_k Z(t_0, t_k). \tag{2.5.2}$$

The floating-rate leg of a Swap- is given by

$$V_{float} = \sum_{k=1}^{n} N_a \alpha_k f_k Z(t_0, t_k), \tag{2.5.3}$$

where:

N_a- is the notional amount on a Swap.

F_k- is the forward rate from date $k-1$ to k.

α_k- is the fraction of a year between $k-1$ and k payment.

n- is the number of floating coupons payable between value date and maturity date.

$Z(t_0, t_k)$- is the discount factor on cash flow date k.

It can be shown that there exists a replicating strategy which can be applied to the cash flows associated with the floating leg of a Swap. The value at time t_0 for this strategy is one and results in the value of a floating leg expressed by (2.5.4) below (Du Preez, 2011).

$$V_{float} = 1 - Z(t_0, t_n). \tag{2.5.4}$$

If V_{swap} denotes the value of a Swap then we have the following: The value of a Swap to the receiver of a fixed rate is $V_{swap}^{fixed} = V_{fixed} - V_{float}$ and to the receiver of a floating rate it is $V_{swap}^{floating} = V_{float} - V_{fixed}$. At initiation the value of a Swap is said to be fair for each side and is equal to zero (Tuckman and Serrat, 2011). For simplicity let us assume we are at the initiation of a Swap and the value is zero. Then from (2.5.2) and (2.5.4) we obtain

$$R \sum_{i=1}^{n} \alpha_i Z(t_0, t_i) = 1 - Z(t_0, t_n). \tag{2.5.5}$$

We can rewrite (2.5.5) as

$$R = \frac{1 - Z(t_0, t_n)}{\sum_{i=1}^{n} \alpha_i Z(t_0, t_i)} \tag{2.5.6}$$

$$= \frac{1 - Z(t_0, t_n)}{\sum_{i=1}^{n-1} \alpha_i Z(t_0, t_i) + \alpha_n Z(t_0, t_n)}. \tag{2.5.7}$$

which gives

$$1 - Z(t_0, t_n) = R\left(\sum_{i=1}^{n-1} \alpha_i Z(t_0, t_i) + \alpha_n Z(t_0, t_n)\right), \tag{2.5.8}$$

from which we get

$$Z(t_0, t_n)(1 + R\alpha_n) = 1 - R\sum_{i=1}^{n-1} \alpha_i Z(t_0, t_i). \tag{2.5.9}$$

Solving for $Z(t_0, t_n)$ in (2.5.9) we get

$$Z(t_0, t_n) = \frac{1 - R\sum_{i=1}^{n-1} \alpha_i Z(t_0, t_i)}{1 + R\alpha_n}. \tag{2.5.10}$$

Equation 2.5.10 shows that it is possible to extract zero-coupon bond prices when given information about the Swap. As in the case of coupon bonds in Section 2.4, the result in (2.5.10) assumes that we know all the inputs to the curve for all maturities, which is not true. Usually in the case of Swaps we use deposit rates for short term maturities. We may also use Forward rates agreements (FRAs) or futures for instruments with maturities starting from 3 months up to 2 years, and for maturities longer than that we may use Swap rates (Hagan and West, 2006).

Now, from (1.2.5) and (2.5.10) it follows that

$$\exp(-r(t_0, t_n)t_n) = \frac{1 - R\sum_{i=1}^{n-1} \alpha_i Z(t_0, t_i)}{1 + R\alpha_n}, \tag{2.5.11}$$

taking logs on both sides and solving for $r(t_0, t_n)$ in (2.5.11) we obtain

$$r(t_0, t_n) = -\frac{1}{t_n} \ln\left[\frac{1 - R\sum_{i=1}^{n-1} \alpha_i Z(t_0, t_i)}{1 + R\alpha_n}\right]. \tag{2.5.12}$$

Bootstrapping algorithm would then use (2.5.12) iteratively to find the implied zero coupon rates from Swaps and initial guess of spot rates.

2.6 Bootstrapping algorithm

First of all the algorithm requires the set of zero-coupon rates as inputs. In most cases for bonds with maturity dates less than 1 year we may use money market rates as inputs and or maturities greater than 1 year we may guess zero-coupon rates or use FRAs, futures or Swaps as inputs rates, it depends on the type of a benchmark security we are using. With this information the algorithm can then start executing the following steps iteratively:

1. The algorithm would start off by interpolating in between the input set of zero-coupon rates to find new extra rates which correspond to intermediate dates.
2. The algorithm would then use (2.4.6) which will give the new estimates for zero-coupon rates.
3. Repeat (1) and (2) until convergence is reached.

The algorithm will converge to the true set of spot rates for our input bonds. Interpolating these final spot rates would give a yield curve.

2.7 An illustration of bootstrapping

Below we discuss an example of bootstrapping yield curves. In our example we use bonds and for the time being we are going to use linear interpolation. We also use r_t to denote the yield at time t.

Problem

Suppose we have 2 coupon-bearing bonds that mature in 3 and 5 years, respectively. Suppose they both have the par value of 100 units and pay semi annual coupons. Let us assume the following information is known about them:

For bond 1: $\mathbb{A}_1 = 99.49$, $p_1 = 9.7\%$

For bond 2: $\mathbb{A}_2 = 92.24$, $p_2 = 10.8\%$.

The problem is to find the zero-coupon rates of these two bonds, i.e., we must strip out r_3 and r_5 from the given information.

Solution

The approach we take in solving this problem was presented in (Smith.L, 2000), which is similar to the bootstrapping algorithm we described above. We start off by choosing the set of maturity dates. The choice we make should ensure that some points match with the maturities for the two bonds in question. To simplify things we choose only 4 maturity dates. Those are:

$0.5 years$, $1.0 year$, $3 years$, $5 years$.

We then assume the existence of two money-market securities maturing in 0.5 years and 1 year, respectively. Their interest rates are:

$r_{0.5} = 4.0\%$, $r_1 = 4.15\%$.

Next we guess the initial zero-coupon rates for the t_3 and t_5 dates as follows:

$r_3 = 4.894\%$, $r_5 = 5.511\%$.

Iteration 1

We now interpolate between our input zero-coupon rates to obtain spot rates at intermediate points. Since we know the rates $r_{0.5}$, r_1, r_3 and r_5 and we assume that we are compounding semi-annually, we can use linear interpolation to find the rates $r_{1.5}$, r_2, $r_{2.5}$, $r_{3.5}$, r_4 and $r_{4.5}$. So we have

$$\frac{r_{1.5} - r_1}{1.5 - 1} = \frac{r_3 - r_1}{3 - 1} \tag{2.7.1}$$

$$\Rightarrow r_{1.5} = \frac{0.5(4.894 - 4.15)}{2} + 4.15 = 4.336, \tag{2.7.2}$$

$$\frac{r_2 - r_1}{2 - 1} = \frac{r_3 - r_1}{3 - 1} \tag{2.7.3}$$

$$\Rightarrow r_2 = \frac{(4.894 - 4.15)}{2} + 4.15 = 4.522, \tag{2.7.4}$$

$$\frac{r_{2.5} - r_1}{2.5 - 1} = \frac{r_3 - r_1}{3 - 1} \tag{2.7.5}$$

$$\Rightarrow r_{2.5} = \frac{1.5(4.894 - 4.15)}{2} + 4.15 = 4.708, \tag{2.7.6}$$

$$\frac{r_{3.5} - r_3}{3.5 - 3} = \frac{r_5 - r_3}{5 - 3} \tag{2.7.7}$$

$$\Rightarrow r_{3.5} = \frac{0.5(5.511 - 4.894)}{2} + 4.894 = 5.048, \tag{2.7.8}$$

$$\frac{r_4 - r_3}{4 - 3} = \frac{r_5 - r_3}{5 - 3} \tag{2.7.9}$$

$$\Rightarrow r_4 = \frac{(5.511 - 4.894)}{2} + 4.894 = 5.20, \tag{2.7.10}$$

$$\frac{r_{4.5} - r_3}{4.5 - 3} = \frac{r_5 - r_3}{5 - 3} \tag{2.7.11}$$

$$\Rightarrow r_{4.5} = \frac{1.5(5.511 - 4.894)}{2} + 4.894 = 5.35675. \tag{2.7.12}$$

The maturity dates for our two bonds are 3 and 5 years, respectively. So we use (2.4.6) to calculate new r_3 and r_5.

For Bond 1: $W = 100$ since we assumed the par value to be 100. So we have

$$r_3 = -\frac{1}{3}\ln\left[\frac{\mathbb{A}_1 - p_1\left(e^{-r_{0.5}(0.5)} + e^{-r_1(1)} + e^{-r_{1.5}(1.5)} + e^{-r_2(2)} + e^{-r_{2.5}(2.5)}\right)}{100 + p_1}\right]$$
$$= -\frac{1}{3}\ln\left[\frac{99.49 - 9.7\left(e^{-4(0.5)} + e^{-4.15(1)} + e^{-4.336(1.5)} + e^{-4.522(2)} + e^{-4.708(2.5)}\right)}{109.7}\right].$$

which gives $r_3 = 3.756$.

For Bond 2 we plug $\mathbb{A}_2$, p_2 and all the rates in (2.4.6) again to compute the value of r_5, so we have

$$\begin{aligned} r_5 &= -\frac{1}{5}\ln\left(\frac{92.24}{100 + 10.8}\right) - \frac{1}{5}\ln\left[-p_2\frac{e^{-r_{0.5}(0.5)} + e^{-r_1(1)} + e^{-r_{1.5}(1.5)}}{100 + 10.8}\right] \\ &\quad -\frac{1}{5}\ln\left[-p_2\frac{e^{-r_2(2)} + e^{-r_{2.5}(2.5)} + e^{-r_3(3)} + e^{-r_{3.5}(3.5)} + e^{-r_4(4)} + e^{-r_{4.5}(4.5)}}{100 + 10.8}\right], \end{aligned}$$

which gives $r_5 = 4.0275$.

Iteration 2

So now we have the new values of r_3 and r_5, we proceed to the next iteration with these values and interpolate again between them to find the new rates. So we do the calculations as above

$$\frac{r_{1.5}-r_1}{1.5-1} = \frac{r_3-r_1}{3-1} \tag{2.7.13}$$

$$\Rightarrow r_{1.5} = \frac{0.5(3.756-4.15)}{2} + 4.15 = 4.015, \tag{2.7.14}$$

$$\frac{r_2-r_{1.5}}{2-1} = \frac{r_3-r_1}{3-1} \tag{2.7.15}$$

$$\Rightarrow r_2 = \frac{(3.756-4.15)}{2} + 4.15 = 3.953, \tag{2.7.16}$$

$$\frac{r_{2.5}-r_1}{2.5-1} = \frac{r_3-r_1}{3-1} \tag{2.7.17}$$

$$\Rightarrow r_{2.5} = \frac{1.5(3.756-4.15)}{2} + 4.15 = 3.85, \tag{2.7.18}$$

$$\frac{r_{3.5}-r_3}{3.5-3} = \frac{r_5-r_3}{5-3} \tag{2.7.19}$$

$$\Rightarrow r_{3.5} = \frac{0.5(4.0275-3.756)}{2} + 3.756 = 3.82, \tag{2.7.20}$$

$$\frac{r_4-r_3}{4-3} = \frac{r_5-r_3}{5-3} \tag{2.7.21}$$

$$r_4 = \frac{(4.0275-3.756)}{2} + 3.756 = 3.890, \tag{2.7.22}$$

$$\frac{r_{4.5}-r_3}{4.5-3} = \frac{r_5-r_3}{5-3} \tag{2.7.23}$$

$$r_{4.5} = \frac{1.5(4.0275-3.756)}{2} + 3.756 = 3.9596, \tag{2.7.24}$$

Again, using these rates in (2.4.6) we get new the zero-coupon rates for our two bonds to be:

$r_3 = 3.76,\ r_5 = 4.03.$

Iteration 3

We have new r_3 and r_5, so we proceed and do interpolation to obtain new estimates. Similar calculations as above give the following results:

$r_{1.5} = 4.05,\ r_2 = 3.955, r_{2.5} = 3.857575,\ r_{3.5} = 3.827575,\ r_4 = 3.895,\ r_{4.5} = 3.96.$

The rest of the results is summarised in table 2.1 below.

Iterations	$r_{0.5}$	r_1	$r_{1.5}$	r_2	$r_{2.5}$	r_3	$r_{3.5}$	r_4	$r_{4.5}$	r_5
1	4.0	4.15	4.336	4.522	4.708	4.894	5.04825	5.20	5.35675	5.511
2	4.0	4.15	4.05	3.95	3.85	3.756	3.82	3.89	3.9596	4.0275
3	4.0	4.15	4.05	3.955	3.857575	3.7601	3.827575	3.895	3.96	4.030
4	4.0	4.15	4.05	3.955	3.8575	3.760	3.8275	3.895	3.96	4.030
5	4.0	4.15	4.05	3.955	3.8575	3.760	3.8275	3.895	3.96	4.030

Table 2.1: Table showing the convergence of bootstrapping algorithm

As we see in table 2.1, our algorithm has converged. So we have managed to uncover the two required zero-coupon rates. The zero-coupon rates for the two bonds above are $r_3 = 3.760$ and $r_5 = 4.030$, respectively. The reason why this algorithm converged to the set of required rates is due to the theorem which was proved by (Smith.L, 2000), which claims that if we are given any guess of zero-coupon bonds, the approximated linear interpolation fit would converge to the zero-coupon yield curve. It can also be shown that the convergence would hold for other methods of interpolation. Please note that we chose the linear interpolation method for demonstration purposes and not because it is a suitable method for constructing yield curves.

2.8 Choosing a good interpolation method

As we have observed above, the bootstrap algorithm requires the interpolation method to construct new data points. Choosing a suitable interpolation method is a very crucial aspect of bootstrapping that needs care. Hagan and West (2006) judge interpolation methods according to the following criteria:

- It must give positive forward rates which imply a decreasing discount factor function.It must also ensure continuous forward curve. A discontinuous forward curve may for instance depicts improbable expectations about the future of interest rates.
- Interpolation method should be local. That is to say that if we make a change in our inputs the effect should be local.
- The method should give rise to stable forward rates, i.e,. forwards that do not change much when inputs are changed.
- The method must be able to price back all the input data from which the yield curve was constructed.

Well-known methods of interpolation do not meet all these requirements. Hagan and West (2006) developed the monotone convex method of interpolation. This method was observed to offset all the problems that arise from other methods of interpolation. In the next chapter we give a comprehensive review of the monotone convex method. We also assess its suitability for interpolating yield curve data by taking some of the above requirements into consideration.

3. The monotone convex method of interpolation

3.1 Forward monotone convex interpolator

In the monotone convex method the interpolation is not done on the interest rate curve itself, instead it takes as inputs the discrete forwards belonging to intervals. If instead of discrete forward rates we have zero rates $r_1, r_2, ..., r_n$, the method will convert them into discrete forward using (1.2.21). In the monotone convex method the discrete forward rates f_i^d and f_{i+1}^d are assigned to midpoint values of the intervals $[t_{i-1}, t_i]$ and $[t_i, t_{i+1}]$, respectively. So we have the points $\left(\frac{t_i+t_{i-1}}{2}, f_i^d\right)$ and $\left(\frac{t_i+t_{i+1}}{2}, f_{i+1}^d\right)$. We regard (t_i, f_i) as an arbitrary point between these two points. To obtain f_i we apply linear interpolation on f_i^d and f_{i+1}^d. So this leads us to the following:

$$\frac{f_i - f_i^d}{t_i - \frac{t_i+t_{i-1}}{2}} = \frac{f_{i+1}^d - f_i^d}{\frac{t_{i+1}+t_i}{2} - \frac{t_i+t_{i-1}}{2}}, \tag{3.1.1}$$

which implies that

$$f_i = \frac{(t_i - t_{i-1})\left(f_{i+1}^d - f_i^d\right)}{t_{i+1} - t_{i-1}} + f_i^d. \tag{3.1.2}$$

From which we obtain

$$f_i = \frac{t_i - t_{i-1}}{t_{i+1} - t_{i-1}} f_{i+1}^d + \frac{t_{i+1} - t_i}{t_{i+1} - t_{i-1}} f_i^d \tag{3.1.3}$$

for $i = 1, 2, ..., n-1$.

The instantaneous forwards at the end points are chosen to be

$$f_0 = f_1^d - \frac{1}{2}\left(f_1 - f_1^d\right), \tag{3.1.4}$$

$$f_n = f_n^d - \frac{1}{2}\left(f_{n-1} - f_n^d\right). \tag{3.1.5}$$

These choices were motivated by the fact that we require $f_0^{'} = 0 = f_n^{'}$.

We want an interpolating function $f(t)$ defined on the domain $[0, t_n]$ for $f_0, f_1, ..., f_n$. Let us take a subinterval $[t_{i-1}, t_i]$ of $[0, t_n]$, so $f(t)$ should satisfy the following conditions:

1. It is able to reproduce the input set of forward rates, i.e.,

$$f(t_{i-1}) = f_{i-1}, \tag{3.1.6}$$

$$f(t_i) = f_i, \tag{3.1.7}$$

   for all i.

2. It is able to recover the discrete forward, i.e.,

$$\frac{1}{t_i - t_{i-1}} \int_{t_{i-1}}^{t_i} f(t)dt = f_i^d \tag{3.1.8}$$

3. f must be positive.
4. f must be continuous.
5. Lastly the function must be monotonic. This is achieved by requiring that if $f_{i-1} < f_i^d < f_i$ then $f(t)$ is increasing on $[t_{i-1}, t_i]$, and if $f_{i-1} > f_i^d > f_i$ then $f(t)$ is decreasing on $[t_{i-1}, t_i]$.

For simplicity, we would like to deal with an interpolation function on $[0, 1]$ rather than on $[t_{i-1}, t_i]$. We therefore define a function g_i for $x \in [0, 1]$ by requiring that

$$g_i(x) = f(t_{i-1} + (t_i - t_{i-1})x) - f_i^d, \tag{3.1.9}$$

where $(t_{i-1} + (t_i - t_{i-1})x) \in [t_{i-1}, t_i]$ for $x \in [0, 1]$.

We write g for g_i in what follows for the simplicity of our notation. We define

$$f(t_{i-1}) = f_{i-1} \tag{3.1.10}$$
$$f(t_i) = f_i, \tag{3.1.11}$$

so requirement 1 is satisfied. Then

$$g(0) = f_{i-1} - f_i^d \tag{3.1.12}$$
$$g(1) = f_i - f_i^d. \tag{3.1.13}$$

We first let

$$x = \frac{t - t_{i-1}}{t_i - t_{i-1}}, \tag{3.1.14}$$

so (3.1.9) and (3.1.14) implies

$$f(t) = g\left(\frac{t - t_{i-1}}{t_i - t_{i-1}}\right) + f_i^d. \tag{3.1.15}$$

To establish requirement 2 we need a condition on $g(x)$ so that (3.1.15) gives us (3.1.8). To find this condition we combine (3.1.8) and (3.1.15) so that

$$\frac{1}{t_i - t_{i-1}} \int_{t_{i-1}}^{t_i} f(t)dt = \frac{1}{t_i - t_{i-1}} \int_{t_i}^{t_{i-1}} \left(g\left(\frac{t - t_{i-1}}{t_i - t_{i-1}}\right) + f_i^d\right) dt. \tag{3.1.16}$$

We let $u = \frac{t-t_{i-1}}{t_i-t_{i-1}}$, and so $du = \frac{dt}{t_i-t_{i-1}}$. Combining this with (3.1.16) implies

$$\frac{1}{t_i - t_{i-1}} \int_{t_{i-1}}^{t_i} f(t)dt = \int_{t=t_{i-1}}^{t=t_i} g(u)du + f_i^d \tag{3.1.17}$$
$$= \int_0^1 g(u)du + f_i^d \tag{3.1.18}$$
$$= \int_0^1 g(x)dx + f_i^d. \tag{3.1.19}$$

So for (3.1.8) to hold we must have

$$\int_0^1 g(x)dx = 0 \tag{3.1.20}$$

3.2 Monotonicity and continuity analysis of the monotone convex method

We now choose the function $g(x)$ so as to satisfy monotonicity and continuity. We know $g(0)$ and $g(1)$ from the input data. Depending on these values we will identify eight different cases which will require four different functions g. These functions will be piecewise quadratic. In the simplest case, we will be able to take g as a quadratic polynomial on the whole interval $[0, 1]$:

$$g(x) = K + Lx + Mx^2. \tag{3.2.1}$$

In order to satisfy $\int_0^1 g(x) = 0$ we will calculate K, L, M as functions of $g(0)$ and $g(1)$.

By (3.1.12), (3.1.13) and (3.2.1) we have

$$g(0) = K = f_{i-1} - f_i^d \tag{3.2.2}$$

$$g(1) = K + L + M = f_i - f_i^d. \tag{3.2.3}$$

For $g(x)$ in (3.2.1) to satisfy $\int_0^1 g(x) = 0$ we must have

$$\int_0^1 (K + Lx + Mx^2)dx = 0 \tag{3.2.4}$$

$$= \left[Kx + \frac{Lx^2}{2} + \frac{Mx^3}{3}\right]_0^1 \tag{3.2.5}$$

$$\Rightarrow K + \frac{L}{2} + \frac{M}{3} = 0. \tag{3.2.6}$$

By (3.2.2) and (3.2.3) we have

$$L = f_i - f_{i-1} - M \tag{3.2.7}$$

and putting (3.2.7) in (3.2.6) gives

$$M = 3f_i + 3f_{i-1} - 6f_i^d \tag{3.2.8}$$

which implies that

$$L = 6f_i^d - 4f_{i-1} - 2f_i. \tag{3.2.9}$$

We plug these values of K, L and M in (3.2.1) to obtain

$$g(x) = g(0)[1 - 4x + 3x^2] + g(1)[-2x + 3x^2] \tag{3.2.10}$$

In analysing monotonicity of an interpolating function, we said that we require f to be such that if $f_{i-1} < f_i^d < f_i$, then $f(t)$ should be increasing on $[t_{i-1}, t_i]$. Whilst if $t_{i-1} > f_i^d > f_i$, then $f(t)$ should be decreasing on $[t_{i-1}, t_i]$. This is equivalent to requiring that if $g(0)$ and $g(1)$ are of opposite signs then g is monotone. We analyse monotonicity of $g(x)$ by studying the behaviour of $g'(x)$ at 1 and 0. So we calculate the derivative of $g(x)$,

$$g'(x) = g(0)(-4 + 6x) + g(1)(-2 + 6x), \tag{3.2.11}$$

which implies that

$$g'(0) = -4g(0) - 2g(1) \tag{3.2.12}$$
$$g'(1) = 2g(0) + 4g(1). \tag{3.2.13}$$

The most significant cases correspond to $g'(0) = 0$ and $g'(1) = 0$, so we consider them below: when $g'(0) = 0$ we have

$$g(1) = -2g(0), \tag{3.2.14}$$

and when $g'(1) = 0$ we have

$$g(0) = -2g(1). \tag{3.2.15}$$

Equations (3.2.14) and (3.2.15) represent two straight lines which devide the $g(0)/g(1)$ space into eight regions as shown in Figure 3.1.

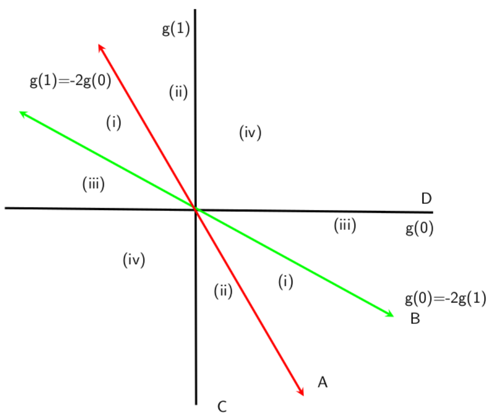

Figure 3.1: The eight regions

These eight regions are then divided into four groups, labelled (i) to (iv), which we describe as follows:

(i) $g(0) > 0,\ -\frac{1}{2}g(0) \geq g(1) \geq -2g(0)$ or $g(0) < 0,\ -\frac{1}{2}g(0) \leq g(1) \leq -2g(0)$.

The first region of this group is in the second quadrant while the second region is in the fourth quadrant, therefore the signs of $g(0)$ and $g(1)$ are not the same. The slopes $g'(0)$ and $g'(1)$ are of the same signs since the two regions are bounded by the lines $g(1) = -2g(0)$ and $g(0) = -2g(1)$. Thus, in this group $g(x)$ is monotonic as required. So this is the simplest case, i.e., the choice of $g(x)$ we made in (3.2.1) is sufficient to guarantee us monotonicity, and since it is a quadratic it is continuous also.

(ii) $g(0) < 0,\ g(1) > -2g(0)$ or $g(0) > 0,\ g(1) < -2g(0)$.

The first region of this group is in the second quadrant while the second region is in the fourth quadrant, hence $g(0)$ and $g(1)$ have opposite signs. The slopes $g'(0)$ and $g'(1)$ have different signs since both regions of this group are bounded by the vertical axis $g(1)$ and by the line $g(1) = -2g(0)$. Thus, $g(x)$ is not monotonic, we therefore have to fine-tune the interpolation function to ensure monotonicity. But before we do that, notice that (i) and (ii) are bounded by the same boundary which we named A in our figure. To ensure that continuity holds, we need to make sure that the formulas for groups (i) and (ii) agree on boundary A. We proceed as follows:

At A we have $g(1) = -2g(0)$. We plug this in (3.2.10) and obtain

$$g(x) = g(0)[1 - 4x + 3x^2] - 2g(0)[-2x + 3x^2] \quad (3.2.16)$$

$$\Rightarrow g(x) = g(0)(1 - x^2). \quad (3.2.17)$$

We can now change our interpolation function for (ii) so that it reduces to equation (3.2.17). We follow the same approach taken by (Hagan and West, 2006). We modify $g(x)$ by introducing extra knot points between $x = 0$ and $x = 1$ and inserting new functions. We redefined $g(x)$ as follows:

$$g(x) = \begin{cases} g(0) & \text{for } 0 \leq x \leq \eta \\ g(0) + (g(1) - g(0)) \left(\frac{x-\eta}{1-\eta}\right)^2 & \text{for } \eta < x \leq 1 \end{cases}, \quad (3.2.18)$$

where η is chosen as in (3.2.19).

$$\eta = \frac{g(1) + 2g(0)}{g(1) - g(0)}. \quad (3.2.19)$$

Let us now check if our interpolation function reduces to (3.2.17). We take a limit as η approaches zero in (3.1.9), therefore we have

$$\lim_{\eta \to 0} \eta(g(1) - g(0)) = g(1) + 2(0) \quad (3.2.20)$$

$$\Rightarrow g(1) = -2g(0) \quad (3.2.21)$$

Therefore the interpolation formula across A will reduce to (3.2.17) as required. Hence, $g(x)$ is smooth across the boundary A.

Upon redefining our interpolation function we need to ensure that the original property that $\int_0^1 g(x)dx = 0$ is preserved. Let us check if our new $g(x)$ satisfies this. From (3.2.18) we have

$$\int_0^1 g(x)dx = \int_0^\eta g(0)dx + \int_\eta^1 g(0)dx + \int_\eta^1 (g(1) - g(0)) \left(\frac{x-\eta}{1-\eta}\right)^2 dx \quad (3.2.22)$$

$$= g(0)\eta + g(0) - g(0)\eta + \int_\eta^1 (g(1) - g(0)) \left(\frac{x-\eta}{1-\eta}\right)^2 dx \quad (3.2.23)$$

If we let

$u = x - \eta \Rightarrow du = 1 - \eta$,

at $x = 1$ $u = 1 - \eta$, at $x = \eta$ $u = 0$. Therefore

$$\frac{(g(1) - g(0))}{(1-\eta)^2} \int_0^{1-\eta} u^2 du = \frac{1}{3} \frac{(g(1) - g(0))}{(1-\eta)^2} \left[u^3\right]_0^{1-\eta} = \frac{(g(1) - g(0))}{3(1-\eta)^2}(1-\eta)^3 \quad (3.2.24)$$

$$= \frac{(g(1) - g(0))}{3}(1-\eta) \quad (3.2.25)$$

We substitute (3.2.19) in (3.2.25) to obtain

$$\frac{(g(1) - g(0))}{3}\left(1 - \frac{g(1) + 2g(0)}{g(1) - g(0)}\right) = -g(0) \quad (3.2.26)$$

We plug this in (3.2.23),

$$\int_0^1 g(x)dx = g(0)\eta + g(0) - g(0)\eta - g(0) = 0. \tag{3.2.27}$$

Therefore our new $g(x)$ is fine.

(iii) $g(0) > 0,\ 0 > g(1) > -\frac{1}{2}g(0)$ or $g(0) < 0,\ 0 < g(1) < -\frac{1}{2}g(0)$.

The first region of this group is in the second quadrant while the second region is in the fourth quadrant. Hence, the sign of $g(0)$ and $g(1)$ are different. The slopes $g'(0)$ and $g'(1)$ have different signs since both regions are bounded by the horizontal axis $g(0)$ and by the line $g(0) = -2g(1)$. Thus, $g(x)$ is not monotonic. We therefore have to fine-tune the interpolation function. But before we do that, notice that the region (i) and (iii) share the same boundary B. Therefore it is necessary to ensure that the formulas for regions (i) and (iii) agree on boundary B to ensure continuity. So we proceed as follows:

At A we have $g(1) = -\frac{1}{2}g(0)$. We plug this in (3.2.10).

$$g(x) = g(0)[1 - 4x + 3x^2] - 2g(0)[-2x + 3x^2] \tag{3.2.28}$$

$$g(x) = g(0)(1 - 3x + \frac{3}{2}x^2) \tag{3.2.29}$$

We carry on and redefine $g(x)$ as follows:

$$g(x) = \begin{cases} g(1) + (g(0) - g(1))\left(\frac{\eta - x}{\eta}\right)^2 & \text{for } 0 < x < \eta \\ g(1) & \text{for } \eta \le x < 1 \end{cases}, \tag{3.2.30}$$

where η is chosen as in (3.2.31).

$$\eta = 3\frac{g(1)}{g(1) - g(0)}. \tag{3.2.31}$$

Let us check if the interpolation function reduces to (3.2.29). We take the limit as η approaches 1 in (3.2.31). So we have

$$\lim_{\eta \to 1}(g(1) - g(0)) = g(1) \tag{3.2.32}$$

$$g(1) = -\frac{1}{2}g(0). \tag{3.2.33}$$

Therefore the interpolation function will reduce to (3.2.29) as required. Hence, $g(x)$ is smooth across the boundary B.

Again, as in group (ii) we need to check if $\int_0^1 g(x)dx = 0$. From (3.2.30) we have,

$$\int_0^1 g(x)dx = \int_0^\eta g(1)dx + \int_0^\eta (g(0) - g(1))\left(\frac{\eta - x}{\eta}\right)^2 dx + \int_\eta^1 g(1)dx \tag{3.2.34}$$

$$= g(1)\eta + g(1) - g(1)\eta + \int_0^\eta \frac{(g(0) - g(1))}{\eta^2}(\eta - x)^2 dx. \tag{3.2.35}$$

If we let:

$u = \eta - x \Rightarrow du = -dx.$

At $x = \eta$, $u = 0$, at $x = 0$, $u = \eta$. Therefore we have

$$\frac{(g(0) - g(1))}{\eta^2} \int_0^\eta u^2 du = \frac{1}{3}(g(0) - g(1))\eta \tag{3.2.36}$$

substituting (3.2.31) in (3.2.36) we obtain

$$\frac{1}{3}(g(0) - g(1)) \left(\frac{3g(1)}{g(1) - g(0)} \right) = -g(1), \tag{3.2.37}$$

we plug this in (3.2.35),

$$\int_0^1 g(x)dx = g(1)\eta + g(1) - g(1)\eta - g(1) = 0. \tag{3.2.38}$$

Therefore $g(x)$ satisfies our requirement.

(iv) $g(0) \geq 0,\ g(1) \geq 0$ or $g(0) \leq 0,\ g(1) \leq 0$.

The signs of $g(0)$ and $g(1)$ are the same since the first region of this group is in the first quadrant whilst the second region is in the third quadrant. Therefore $g(x)$ is not required to be monotone here. But it is still necessary to ensure that the formulas for groups (iv) and (ii) agree on C which corresponds to $g(1) = 0$ and that the formulas for groups (iv) and (iii) agree on boundary D which corresponds to $g(0) = 0$. To ensure this we redefine $g(x)$ on group (iv) as the piecewise function of two quadratics, i.e,

$$g(x) = \begin{cases} \omega + (g(0) - \omega) \left(\frac{\eta - x}{\eta} \right)^2 & \text{for } 0 < x < \eta \\ \omega + (g(1) - \omega) \left(\frac{x - \eta}{1 - \eta} \right)^2 & \text{for } \eta < x < 1 \end{cases}. \tag{3.2.39}$$

Where η and ω are defined as in (3.2.40) and (3.2.41), respectively.

$$\eta = \frac{g(1)}{g(1) + g(0)} \tag{3.2.40}$$

$$\omega = -\frac{g(0)g(1)}{g(0) + g(1)} \tag{3.2.41}$$

Let us check if this reformulation satisfies our requirements. Notice that at D (3.2.39) is,

$$g(1) = 0 = \begin{cases} \omega + (g(0) - \omega) \left(\frac{\eta - 1}{\eta} \right)^2 & \text{for } 0 < x < \eta \\ 0 & \text{for } \eta < x < 1 \end{cases}, \tag{3.2.42}$$

whereas the formula for group (iii) in D is

$$g(1) = 0 = \begin{cases} g(0) \left(\frac{\eta - 1}{\eta} \right)^2 & \text{for } 0 < x < \eta \\ 0 & \text{for } \eta < x < 1 \end{cases}. \tag{3.2.43}$$

Therefore (3.2.42) and (3.2.43) will be equal on boundary D when $\omega = 0$. Which implies that as $\omega \to 0$ the formula in group (iv) reduce to that in group (iii). Indeed, from (3.2.40) we have

$$\lim_{\omega\to 0} \omega(g(0) + g(1)) = -g(0)g(1) \tag{3.2.44}$$

$$\Rightarrow g(0) = 0. \tag{3.2.45}$$

So our new $g(x)$ ensures that the formula for group (iv) reduces to the formula for group (iii) in D as required.

At C, equation (3.2.39) is,

$$g(0) = 0 = \begin{cases} 0 & \text{for } 0 < x < \eta \\ \omega + (g(1) - \omega)\left(-\frac{\eta}{1-\eta}\right)^2 & \text{for } \eta < x < 1 \end{cases}, \tag{3.2.46}$$

whereas, if we take $\omega = 0$ in (3.2.46), we obtain

$$g(0) = 0 = \begin{cases} 0 & \text{for } 0 < x < \eta \\ g(1)\left(-\frac{\eta}{1-\eta}\right)^2 & \text{for } \eta < x < 1 \end{cases}. \tag{3.2.47}$$

Therefore, (3.2.46) and (3.2.47) will be equal at C when $\omega = 0$. Which implies that as $\omega \to 0$ the formula in group (iv) reduces to that in group (ii), hence our new $g(x)$ ensures that the formula for group (iv) reduces to the formula for group (ii) on boundary C as required.

Let us check if $\int_0^1 g(x)dx = 0$ is satisfied by our new $g(x)$.

$$\int_0^1 g(x)dx = \int_0^\eta \omega dx + \int_0^\eta (g(0) - \omega)\left(\frac{\eta - x}{\eta}\right)^2 dx + \int_\eta^1 \omega dx +$$
$$\int_\eta^1 (g(1) - \omega)\left(\frac{x - \eta}{1 - \eta}\right) dx$$
$$= \frac{2}{3}\omega + \frac{\eta}{3}g(0) + \frac{1-\eta}{3}g(1)$$
$$= -\frac{2}{3}\frac{g(0)g(1)}{g(1) + g(1)} + \frac{1}{3}\frac{g(0)g(1)}{g(1) + g(0)} + \frac{1}{3}\frac{g(1)g(0)}{g(1) + g(0)} = 0.$$

Therefore $g(x)$ satisfies our requirement.

3.3 Positivity analysis of the monotone convex method

If we require $f(t)$ to be positive then by (3.1.9) we should ensure that

$$f(t) = g(x) + f_i^d \geq 0, \tag{3.3.1}$$

for all $x \in [0, 1]$, which means

$$g(x) \geq -f_i^d \tag{3.3.2}$$

Let us start by analysing g at the end points. Since f_{i-1} and f_i are positive rates, it follows by (3.2.2), (3.2.3) and (3.3.2) that

$$g(0) = f_{i-1} - f_i^d \geq -f_i^d, \quad (3.3.3)$$
$$g(1) = f_i - f_i^d \geq -f_i^d. \quad (3.3.4)$$

Thus, the inequality (3.3.2) is satisfied at the endpoints. Recall that in Section 3.2 we ensured the monotonicity in (i), (ii) and (iii), this means that there are no sign changes in these regions. Furthermore, since we have shown that positivity is met at the endpoints $g(0)$ and $g(1)$, it is trivial to infer that positivity is met throughout groups (i), (ii) and (iii).

The only regions we need to examine now are the ones in group (iv). From (3.2.40) and (3.2.41) we know that the interpolation function $g(x)$ has a minimum value at the x-value $\eta = \frac{g(1)}{g(1)+g(0)}$ which occurs at $\omega = -\frac{g(0)g(1)}{g(0)+g(1)}$. If we can ensure that this minimum is positive we would have ensured the positivity for the whole of of group (iv). This means that it is sufficient to ensure that (3.3.5) holds.

$$\frac{g(0)g(1)}{g(0)+g(1)} \leq f_i^d. \quad (3.3.5)$$

We claim that (3.3.5) is equivalent to requiring that $f_{i-1} \leq 3f_i^d$ and $f_i \leq 3f_i^d$. To prove this we first note that if $0 < g(0) \leq 2f_i^d$ and $0 < g(1) \leq 2f_i^d$, then

$$\frac{1}{f_i^d} = \left(\frac{1}{2f_i^d} + \frac{1}{2f_i^d}\right) \leq \frac{1}{g(0)} + \frac{1}{g(1)} = \frac{g(1)+g(0)}{g(1)g(0)} \quad (3.3.6)$$
$$\Rightarrow \frac{g(1)g(0)}{g(0)+g(1)} \leq f_i^d. \quad (3.3.7)$$

Hence, we have proved our assertion since $0 < g(0) \leq 2f_i^d$ and $0 < g(1) \leq 2f_i^d$ means that $f_{i-1} \leq 3f_i^d$ and $f_i \leq 3f_i^d$. Therefore the instantaneous forward rates will be positive as long as $f_{i-1} \leq 3f_i^d$ and $f_i \leq 3f_i^d$ for $i = 1, 2, ..., n-1$.

3.4 Experimenting with Octave

Up to this point we have given only theoretical analysis of the monotone convex method. In this section we intend to give a practical test of this method. We will take the market data as inputs and apply monotone convex interpolation on it. We wish to inspect the resulting curves and see if our method achieve required result. We will take as the basis of our judgement the requirements we have stated above.

For the start let us consider the problem almost similar to the one we solved in Section 2.7 above.

Suppose we have three interest rates for three risk-free securities in the money market. These instruments they mature in 1, 6, and 12 months, respectively. Their rates are given as:

$r_{0.08} = 13.95\%$; $r_{0.5} = 14.48\%$; $r_1 = 14.88\%$

Suppose we want to determine the zero-coupon rates for four coupon-bearing bonds that are trading in the market. We assume these bonds have a par value of 1 unit and their interest rates are compounded semi annually. Furthermore we assume they are priced at present and the following information is known about them:

$\mathbb{A}_1 = 0.9751097$ $p = 0.075$

$\mathbb{A}_2 = 0.9845960$ $p = 0.08$

$\mathbb{A}_3 = 0.8766290$ $p = 0.07$

$\mathbb{A}_4 = 0.8080316$ $p = 0.065$

The zero-coupon rates four these four coupon-bearing bonds are not known, so they are guess as below:

$r_3 = 15.3\%$; $r_5 = 15.6\%$; $r_8 = 15.9\%$; $r_{10} = 16.1\%$;

Bootstrapping and applying the monotone convex method into the above rates give the zero-coupon yield curve shown in figure 3.2.

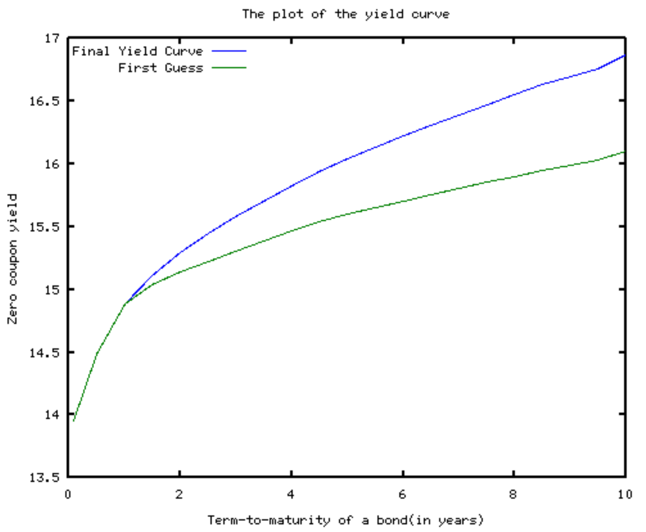

Figure 3.2: Result of the bootstrapping using Monotone convex method

In figure 3.2 we observe that the bootstrapping method combined with the monotone convex method was able to successfully modify the curve we initially guessed to give the correct zero coupon curve, which is what we desired from an interpolation method. Our curve is upward sloping which is very reasonable. For example, from an economic point of view it can be interpreted as saying that short term maturities give small yields than the long term maturities because there is a big risk in the long-term investments.

4. Conclusion

In this dissertation we presented a comprehensive review of the monotone convex method of interpolating yield curves. From the outset we stated how the interpolation problem arises in finance. We said that it is strongly linked to the bootstrapping method which extract zero-coupon rates from coupon-bearing instruments. We then stated the conditions that must be met by any interpolation method in order to give rise to the reasonable yield curve. We noted that many interpolation methods are not suitable for building yield curves. We asserted that the monotone convex method is the most promising method of interpolation. We undertook some theoretical analysis to justify our assertion. The basis of our analysis involved ensuring the continuity and positivity of forward rates and ensuring the monotone interpolating function. The results showed that the monotone convex method meets these conditions and is thus an ideal interpolation method for yield curves. For practical purposes we implemented the monotone convex method using Octave. Our Octave program can be of great aid in furthering our research in future.

Appendix A. Octave programmes

A.1 Forward rates function

The forward rate function was created to calculate discrete forward rates and instantaneous forward rates given the input set of rates and maturity dates.

```
function [fd,f]=ForwardCalc(Times, Rates)

% Inputs
%  Times= The vector corresponding to the input set of maturities
%  Rates= The vector corresponding to the input set of zero-coupon rates

% Output
% Returns fd the vector of forward rates and f the vector of instantaneous forward rates

  n=length(Rates);
  fd=zeros(1,n);
  f=zeros(1,n);

  fd(1)=Rates(1);
  for i=2:n
     fd(i)=(Rates(i)*Times(i)-Rates(i-1)*Times(i-1))/(Times(i)-Times(i-1));
  endfor

  for i=2:n-1
     f(i)=fd(i+1)*(Times(i)-Times(i-1))/(Times(i+1)-Times(i-1))+fd(i)*(Times(i+1)
     -Times(i))/(Times(i+1)-Times(i-1));
     if f(i)>3*min(fd(i),fd(i+1))
         f(i)=2*min(fd(i),fd(i+1));
     endif
  endfor
  f(1)=fd(2)*Times(1)/Times(2)+fd(1)*(Times(2)-Times(1))/Times(2);
  f(n)=fd(n)-0.5*(f(n-1)-fd(n));
endfunction
```

A.2 Monotone convex method functions

Two functions for monotone convex interpolation were created, they return the intermediate value of two rates in the input set of zero-coupon rates.

```
function LastIndex=FindLastIndex(Times,t)

% Input
%Times= The vector of input set of maturities
```

```
%t= An value which correspond to any time, it does not exceed the greates value in Times

      for i=1:length(Times)-1
          if t<=Times(i+1) & t>=Times(i)
              LastIndex=i;
          endif
      endfor
      if t<Times(1)
          i=1;
      endif
      if t>Times(length(Times))
          i=length(Times)-1;
          LastIndex=i;
      endif
endfunction

% Output
% This function return an index i which corresponds to t
```

```
function mono=Monotone_Convex(Times,Rates,dvalue)
% Input

% Times= The vector of input dates
% Rates= The vector of input rates
% dvalue=  Value corresponding to the time that the user wishes to obtain the rate for.
% dvalue cannot be greater than than the largest element in Times


% Output
% The ouput is the rate which corresponds to the dvalue.

    n=length(Rates);
    j=FindLastIndex(Times,dvalue);
    l=Times(j+1)-Times(j);
    x=(dvalue-Times(j))/l;
    [fd,f] = ForwardCalc(Times,Rates);

    g0=f(j)-fd(j+1);
    g1=f(j+1)-fd(j+1);
    f0=fd(1)-0.5*(f(1)-fd(1));

    if (dvalue<Times(1))
       mono=f0;
    elseif (dvalue>Times(n))
         mono=(1/dvalue)*(Rates(n)*Times(n)+f(n)*(dvalue-Times(n)));
    else
       if (x==0)|(x==1)
          G=0;
```

```
        elseif (g0<0 & -0.5*g0<=g1 & g1<=-2*g0)|(g0>0 & -0.5*g0>=g1 & g1>=-2*g0)
            G=l*(g0*(x-2*x^2+x^3)+g1*(-x^2+x^3));
        elseif (g0<0 & g1>-2*g0)| (g0>0 & g1<-2*g0)
            eta=(g1+2*g0)/(g1-g0);
            if (x<=eta)
                G=g0*(dvalue-Times(j));
            else
                G=g0*(dvalue-Times(j))+(g1-g0)*(x-eta)^3/(1-eta)^2/3*l;
            endif
        elseif (g0>0 & 0>g1 & g1>-0.5*g0)|(g0<0 & 0<g1 & g1<-0.5*g0)
            eta=3*g1/(g1-g0);
            if (x<eta)
               G=l*(g1*x-1/3*(g0-g1)*((eta-x)^3/eta^2-eta));
            else
               G=l*(2/3*g1+1/3*g0)*eta+g1*(x-eta)*l;
            endif
        elseif (g0=0)&(g1=0)
            G=0;
        else
           eta=g1/(g1+g0);
           a=-g0*g1/(g0+g1);
           if (x<=eta)
              G=l*(a*x-1/3*(g0-a)*((eta-x)^3/eta^2-eta));
           else
              G=l*(2/3*a+1/3*g0)*eta+l*(a*(x-eta)+(g1-a)/3*(x-eta)^3/(1-eta)^2)
           endif
        endif
        mono=1/dvalue*(Times(j)*Rates(j)+fd(j+1)*(dvalue-Times(j))+G);
    endif
endfunction
```

A.3 Bond bootstrapping function

The function for bootstrapping coupon bonds was created, it uses monotone convex function to build zero coupon yield curve. When it is called with a suitable data it would return the plot depicting the yield curve. Note that this function is not general, it depends on the user's requirements. One can alter it or write a different function to satisfy his/her requirements.

```
function Bond_Bootstrap(mat_times,inp_rates,Req_Times, m,n,Comp_Dates,Coupons,Bon_Prices)

%   Inputs variables

% inp_rates=vector representing the yields on the input set of bonds
% mat_times=vector representing the set of maturity dates
% m= an integer value representing the number of money markets
% Req_Times= The set of times that are not money markets
```

```
% Bon_Prices=The prices of the input set of bonds
% Coupns= Coupon rates corresponding to input set of bonds
%Comp_Dates= the set of all compounding dates
% n=number of compounding dates per year




test_value=10;            % The value that tests for convergence
while (test_value>0.0000000001)  % while loop that tests for convergence
   conveg1=sum(inp_rates);       % conveg1 stores the sum of the initial rates
   countv=1;                     % count1 stores the number of iterations


% These statements find all the rates for Comp_Date
for i=1:length(Comp_Dates)
   All_Rates(i)=Monotone_Convex(mat_times,inp_rates,Comp_Dates(i));
   All_Rates(i)=All_Rates(i)*0.01;
endfor

% These commands calculates the new inp_rates so that they get
  updated in each iteration
Sums=0;
Count=0;
for j=1:length(Req_Times)
  Sums=0;
  for k=1:n*Req_Times(j)-1
      Sums=Sums+exp(-All_Rates(k)*k/n);
  endfor
  Count=Count+1;
  CoupnList(Count)=Sums;
endfor

for o=1:length(Bon_Prices)
  inp_rates(m+o)=(-1/mat_times(m+o))*log((Bon_Prices(o)
   -Coupons(o)*CoupnList(o))/(1+Coupons(o)))*100;
endfor


conveg2=sum(inp_rates);        % conveg1 stores the sum of the new updated inp_rates
test_value=abs(conveg1-conveg2);  % test_value is updated to the new value to test
                                  convergence
countv=countv+1;                  % updates the number of iterations

% The following statements throw an error in case convergence is not reached
if countv>50
   disp("bootstrap is failing to converge");
```

```
endif
endwhile

%  The following statements interpolate the rates which would
 results from the above algorithm, as we can see it is outside the loop.

for p=1:length(Comp_Dates)
   FinalRates(p)=Monotone_Convex(mat_times,inp_rates,Comp_Dates(p));
endfor
[FinalRates];

%  plots the final rates to get the yield curve.
%  plots the final rates to get the yield curve.
plot(Comp_Dates,FinalRates)
title('The plot of the yield curve')
xlabel('Term-to-maturity of a bond(in years)')
ylabel('Zero coupon yield')
endfunction

% Output

% This function will plot the yield curve given the bond data
```

A.4 The flow of execution

In running these functions you must ensure that all the files are in the same directory. You must also specify the correct inputs to the program. The Monotone convex function calls the FindLast index function and the forward function. It is in turn called by the bond_boostrap function. But it is not necessary to run the functions when you are calling them. If they are in the same directory with the main script then Octave will automatically call them into your script.

Acknowledgements

I am very thankful to my supervisor Emeritus Professor Ronnie Becker for his assistance and guidance during my essay. I truly admire his approach of paying attention to every thing and ensuring precision. I also thank AIMS for granting me the opportunity to shape my future. To all my colleagues, you are amazing guys, I can now say I have friends from all over Africa. Lastly, I thank my family for their support and for giving me freedom to pursue my dream.

References

M. Choudhry. *The repo handbook*. Butterworth-Heinemann, 2002.

J. C.Hull. *Options, futures and other Derivatives*. Prentice Hall, 5th edition, 2002.

A. Colin. *Fixed income attribution*. John Wiley & Sons, 2005.

P. F. Du Preez. *An investigation into popular methods for constructing yield curves*. PhD thesis, University of Pretoria, 2011.

F. J. Fabozzi, S. V. Mann, et al. *Introduction to fixed income analytics*, volume 83. John Wiley & Sons, 2001.

E. Fama and Bliss. The information in long-maturity forward rates. *The American Economic Review.*, 7:680–692, 1987.

P. S. Hagan and G. West. Interpolation methods for curve construction. *Applied Mathematical Finance.*, 13:89–129, 2006.

J. Hull. *Options, futures and other Derivatives*. Prentice Hall, 6th edition, 2006.

C.-F. Lee. C and A. C. Lee. A. *Encyclopedia of finance*. Springer, 2006.

J. H. McCulloch J. The tax-adjusted yield curve. *The Journal of Finance*, 30(3):811–830, 1975.

R. C. Nakisa. *A Financial Bestiary*. Chesham Bois Publishing, 2010.

S. K. Nawalkha, G. M. Soto, and N. K. Beliaeva. *Interest rate risk modeling: The fixed income valuation course*, volume 178. John Wiley & Sons, 2005.

C. S. Nelson. Modelling of yield curves. *Journal of Business.*, 60:173–489, 1987.

L. Smith.L. *An Analysis of the Term Strucure of Interest Rates and Bond Options in the South African Capital Market*. PhD thesis, University of Pretoria, 2000.

Svensson. Estimating and interpolating forward interest rates. *Sweden. International Monetary Fund.*, 4:144–145, 1992.

B. Tuckman and A. Serrat. *Fixed income securities: tools for today's markets*, volume 621. John Wiley & Sons, 2011.

P. Veronesi. *Fixed income securities: Valuation, risk, and risk management*. John Wiley & Sons, 2010.

G. West. The monotone convex method of interpolation. *Financial Modelling Agency.*, 16:379–423, 623–656, 2011.