Valentin PICAVET - 73161
Elias HOSNI - 72998
Wael SFAR - 73224
Laurie SCRIBOT - 74706
Mathilde BRUNO - 72802

Advanced Fixed Income and Credit

Group Project

Please consider the Python code attached for our answers.

Question 1: Bootstrapping

Q1.1) We want to calculate the prices of discount bonds given the market prices of coupon bonds for maturities N from 0.5 years to 26 years, compounded semi-annually. We assume the face value is 100.

For N = 0.5 years,

$$Current\ price = \left(100 + \frac{Coupon\ rate}{2}\right) \times P_0^{0.5}$$

with $P_0^{0.5}$ being the price of a discount bond with maturity 0.5 years at time 0, the value we are looking for. The coupon rate is annual, this is why we divide it by the frequency.

Therefore,

$$P_0^{0.5} = Current\ price \Big/ \left(100 + \frac{Coupon\ rate}{2}\right)$$

$$P_0^{0.5} = \ 101.0544/101.5 = 0.995610$$

For N = 1 year,

$$Current\ price = \left(\frac{Coupon\ rate}{2}\right) \times P_0^{0.5} + \left(100 + \frac{Coupon\ rate}{2}\right) \times P_0^{1}$$

So,

$$P_0^1 = \left[Current\ price - \left(\frac{Coupon\ rate}{2}\right) \times P_0^{0.5}\right] \Big/ \left(100 + \frac{Coupon\ rate}{2}\right)$$

We deduce the below relationship,

$$\boldsymbol{P_0^N = \left[Current\ price - \frac{Coupon\ rate}{2} \times \sum_{k=1}^{2N} P_0^{k/2}\right] \Big/ \left(100 + \frac{Coupon\ rate}{2}\right)}$$

Then, we look for the zero-coupon yields,

$$P_0^N \ = \frac{1}{\left(1 + \frac{y_0^N}{2}\right)^{2N}}$$

Thus,

$$\boldsymbol{y_0^N} = \mathbf{2} \times \left(\frac{\mathbf{1}}{(\boldsymbol{P_0^N})^{\frac{\mathbf{1}}{\mathbf{2}\boldsymbol{N}}}} - \mathbf{1}\right)$$

We implement the formulas in bold in Python and we obtain the following discount prices and zero-coupon yields.

Coupon Rate (%)	Maturity Date	Current Price	Tenor	Discounted Price	Zero Coupon Yield (in %)
3	2004-02-15	101.0544	0.5	0.995610	0.881901
2.125	2004-08-15	100.9254	1.0	0.988176	1.192962
1.5	2005-02-15	99.8942	1.5	0.976738	1.575289
6.5	2005-08-15	109.0934	2.0	0.963406	1.872718
5.625	2006-02-15	108.4380	2.5	0.947375	2.174163
2.375	2006-08-15	99.7848	3.0	0.928970	2.471112
6.25	2007-02-15	111.7184	3.5	0.907564	2.790463
3.25	2007-08-15	101.0841	4.0	0.887418	3.008378
3	2008-02-15	99.1692	4.5	0.864791	3.254351
3.25	2008-08-15	99.2710	5.0	0.841559	3.479914
5.5	2009-02-15	109.7707	5.5	0.819380	3.654950
6	2009-08-15	112.1450	6.0	0.794000	3.881711
6.5	2010-02-15	114.9084	6.5	0.769343	4.075084
5.75	2010-08-15	110.3894	7.0	0.746507	4.220336
5	2011-02-15	105.2934	7.5	0.724062	4.351720
5	2011-08-15	104.7607	8.0	0.701204	4.486530
4.875	2012-02-15	103.4391	8.5	0.680072	4.587793
4.375	2012-08-15	99.2806	9.0	0.660381	4.663971
3.875	2013-02-15	95.0288	9.5	0.643389	4.696459
4.25	2013-08-15	97.7693	10.0	0.627754	4.710684
13.25	2014-02-15	174.3251	10.5	0.611738	4.735682

Coupon Rate (%)	Maturity Date	Current Price	Tenor	Discounted Price	Zero Coupon Yield (in %)
12.5	2014-08-15	168.9389	11.0	0.585340	4.928490
11.25	2015-02-15	157.0552	11.5	0.546185	5.328860
10.625	2015-08-15	152.4222	12.0	0.528678	5.382614
9.25	2016-02-15	140.0135	12.5	0.509846	5.462435
7.5	2016-08-15	123.3044	13.0	0.492711	5.519651
8.75	2017-02-15	136.0598	13.5	0.476050	5.574292
8.875	2017-08-15	137.5040	14.0	0.457551	5.663463
9.125	2018-02-15	140.7920	14.5	0.444315	5.673620
9	2018-08-15	139.9079	15.0	0.429353	5.716684
8.875	2019-02-15	138.7431	15.5	0.412853	5.789727
8.125	2019-08-15	130.7162	16.0	0.398744	5.829829
8.5	2020-02-15	135.2938	16.5	0.386180	5.850305
8.75	2020-08-15	138.3466	17.0	0.371999	5.902262
7.875	2021-02-15	128.4995	17.5	0.360481	5.916195
8.125	2021-08-15	131.7341	18.0	0.349287	5.929877
8	2022-02-15	130.4736	18.5	0.338053	5.949207
7.25	2022-08-15	121.5800	19.0	0.327859	5.956296
7.125	2023-02-15	120.1744	19.5	0.317791	5.966029
6.25	2023-08-15	109.4538	20.0	0.309470	5.951302
7.5	2024-02-15	125.4600	20.5	0.301222	5.939697
7.5	2024-08-15	125.4466	21.0	0.290205	5.978892
7.625	2025-02-15	127.1477	21.5	0.280453	6.001532
6.875	2025-08-15	117.5509	22.0	0.272593	5.996206
6	2026-02-15	106.3626	22.5	0.267598	5.945634
6.75	2026-08-15	116.1986	23.0	0.257756	5.982261
6.625	2027-02-15	114.7086	23.5	0.251277	5.964657
6.375	2027-08-15	111.4036	24.0	0.244246	5.960339
6.125	2028-02-15	108.0391	24.5	0.237439	5.955713

Coupon Rate (%)	Maturity Date	Current Price	Tenor	Discounted Price	Zero Coupon Yield (in %)
5.5	2028-08-15	99.6330	25.0	0.232986	5.912825
5.25	2029-02-15	96.2876	25.5	0.228238	5.878319
6.125	2029-08-15	108.4062	26.0	0.220240	5.904868

Q1.2) We want to compare the market zero-coupon yields that we have just calculated to the zero-coupon yields from Deutsche Bank's model. From Deutsche Bank's perspective, here are the rules we define:

- If the market yields are below our (DB's) model yields, which means that the price on the market seems to be higher than what it should be according to our model, then we sell the zero-coupon bond.
- Otherwise, we buy the zero-coupon bond when the market yields are above the model yields.

Maturity (years)	Model Prediction (BEY)	Zero Coupon Yield (in %)	Decision
1y	1.2443	1.192962	Sell
2y	1.8727	1.872718	Buy
3y	2.4110	2.471112	Buy
4y	2.9665	3.008378	Buy
5y	3.4454	3.479914	Buy
6y	3.8557	3.881711	Buy
7y	4.1996	4.220336	Buy
8y	4.4677	4.486530	Buy
9y	4.6528	4.663971	Buy
10y	4.7107	4.710684	Sell
15y	5.7160	5.716684	Buy
20y	5.9517	5.951302	Sell
25y	5.9315	5.912825	Sell

Q1.3) This strategy of buying some bonds and selling others is an arbitrage strategy but whether it is a risk-free strategy depends on the accuracy of the Deutsche Bank's model at predicting the price. We cannot say it is a risk-free strategy because there are risks the model might be wrong and not 100% right. In fact, it depends on the risk of the model thus it is not risk-free.

Question 2: Cubic splines

Q2.1) The cubic spline function consists of six cubic polynomials spliced together at the following set of knot points: 2, 5, 10, 15, 20.

$$y(t) = \begin{cases} a_0 t^3 + b_0 t^2 + c_0 t + d_0 & for\ 0 < t < 2 \\ a_1(t-2)^3 + b_1(t-2)^2 + c_1(t-2) + d_1 & for\ 2 < t < 5 \\ a_2(t-5)^3 + b_2(t-5)^2 + c_2(t-5) + d_2 & for\ 5 < t < 10 \\ a_3(t-10)^3 + b_3(t-10)^2 + c_3(t-10) + d_3 & for\ 10 < t < 15 \\ a_4(t-15)^3 + b_4(t-15)^2 + c_4(t-15) + d_4 & for\ 15 < t < 20 \\ a_5(t-20)^3 + b_5(t-20)^2 + c_5(t-20) + d_5 & for\ 20 < t < T \end{cases}$$

We construct a cubic spline interpolation using the 'make_interp_spline' and 'cubic_splines' on python.

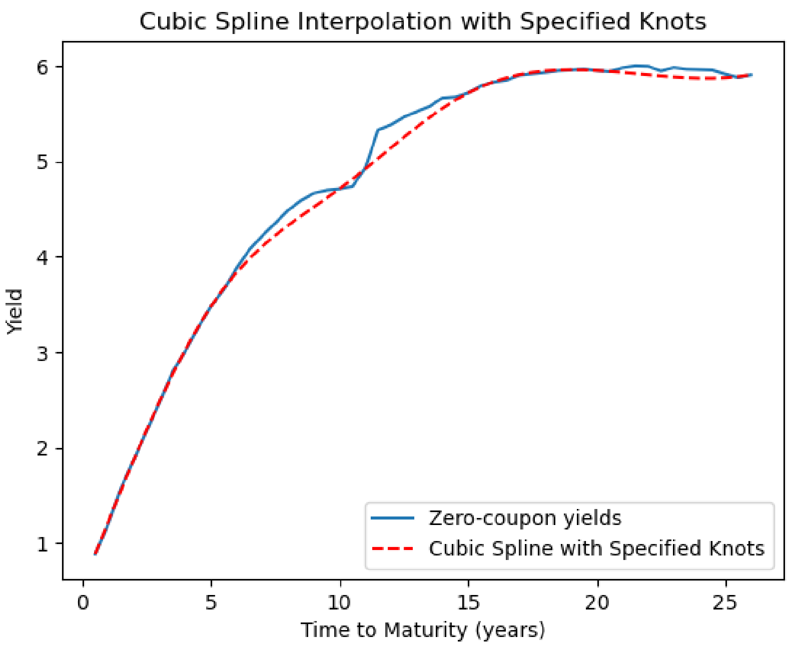
The 'make_interp_spline' function:

- It is part of the scipy.interpolate module and is used to create a spline interpolation object.
- It takes input knots and corresponding zero-coupons yields and constructs a spline interpolation function.
- It is used to create a cubic spline interpolation (k=3) based on the specified knots and corresponding zero coupon yields.

The 'cubic_splines' function:

- It is a custom function defined in the code.
- It takes the dataframe 'data' containing the zero-coupon yields and tenors and the list of knots as input.
- It preprocesses the data, extracting the 'Tenor' column, converting it to floating-point numbers, and adjusting the knots by including the first and last values of the 'Tenor' column.
- It then uses the make_interp_spline function to perform cubic spline interpolation and returns the interpolated values.

The plot resulting is the following one:



Q2.2) We compare the yields obtained with the cubic spline with the yields from Deutsche Bank's model.

We see that the cubic spline overlays the zero-coupon yields curve on some parts except between 5 and 15 years and between 20 and 25 years.

We reproduce the same reasoning as in Q1.2 replacing the previous zero-coupon yields by those calculated with the cubic spline curve. Here is the result, again from Deutsche Bank's perspective:

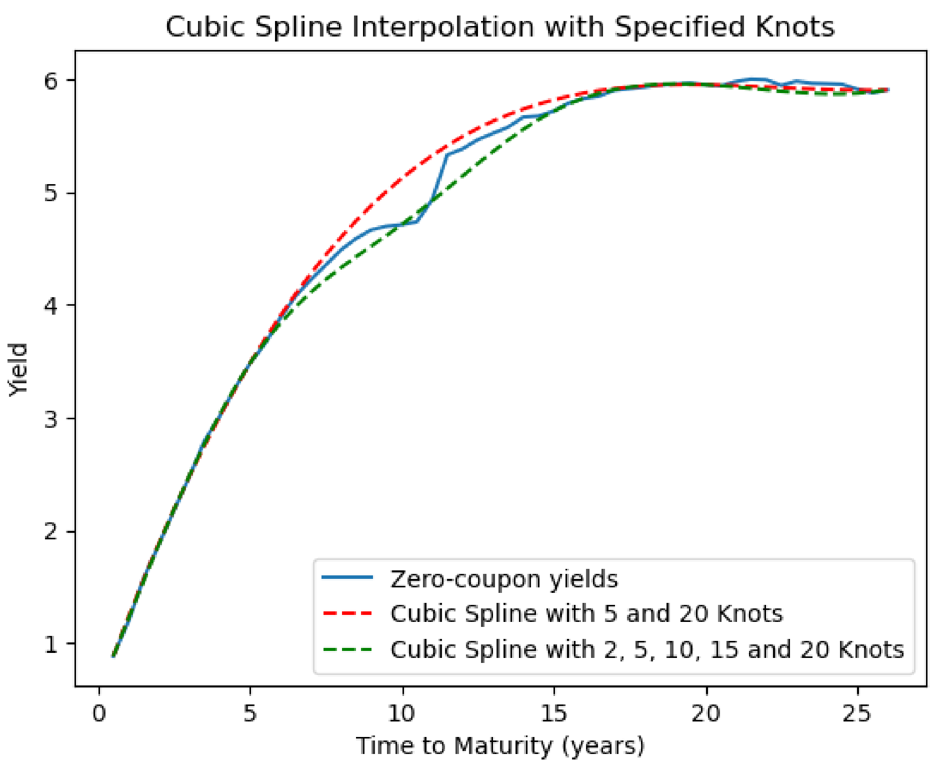
Maturity (years)	Model Prediction (BEY)	Zero Coupon Yield (in %)	Decision
1y	1.2443	1.218998	Sell
2y	1.8727	1.872718	Buy
3y	2.4110	2.481531	Buy
4y	2.9665	3.024307	Buy
5y	3.4454	3.479914	Buy
6y	3.8557	3.835220	Sell
7y	4.1996	4.109089	Sell
8y	4.4677	4.328381	Sell
9y	4.6528	4.519959	Sell
10y	4.7107	4.710684	Sell
15y	5.7160	5.716684	Buy
20y	5.9517	5.951302	Sell
25y	5.9315	5.875359	Sell

Using those new 'spline-based' zero-coupon yields affects the conclusions reached in Q1.2 for maturities between 6y and 9y as seen in the highlighted parts of the table above.

Q2.3) When using only two knots 5- and 20- years zero-coupon yields, the cubic splines' function is:

$$y(t) = \begin{cases} a_0 t^3 + b_0 t^2 + c_0 t + d_0 \quad for\ 0 < t < 5 \\ a_1(t-5)^3 + b_1(t-5)^2 + c_1(t-5) + d_1 \quad for\ 5 < t < 20 \\ a_2(t-20)^3 + b_2(t-20)^2 + c_2(t-20) + d_2 \quad for\ 20 < t < T \end{cases}$$

The plot of the cubic spline using only two knots is the below curve in red:



Changing the knots affects the estimated zero coupon yields and thus the estimated discount function. Indeed, by looking at the graph, we can see that using only 5 and 20 zero coupon yields as knots give us a curve that follows the trend of the real zero-coupon yields but it is less precise than with 2-, 5-,10-, 15- and 20-year zero coupon yields as knots because it has only 2 knots to approximate the function whereas earlier we used more knots so it was better approximated.

Question 3: Nelson Siegel model

Q3.1) Here we use the NSS model to fit a yield curve to empirical data, helping to understand and predict the term structure of interest rates. Our code essentially performs optimization to find the parameters that best fit the NSS model to the observed market data.

Let's have a deeper look at our code and the method we used.

-To estimate the discount function according to the Nelson-Siegel-Svensson model on Python, we first imported necessary libraries, particularly 'curve-fit' from 'spicy.optimize' for curve fitting.

-The nss_model function implements the Nelson-Siegel-Svensson model, taking in parameters t (time to maturity) and several beta (β) parameters along with tau (τ) parameters that define the shape of the yield curve.

-We extract the observed market data we will be using with the NSS method: the time to maturity ('x') and the zero-coupon yields ('y').

-We initialize initial values for beta and tau parameters and create an initial guess for them: as seen in class, the parameters are estimated such that they minimize the sum of squared pricing error for the coupon bonds. Thus, we start with some value for the parameters that seem relevant, we estimate the value of the target function, and we change until we cannot reduce the pricing error anymore.

-For beta0, being the long forward rate, we chose the highest possible yield in our data set (we assume here that long maturity means higher interest rate, so that the rate curve is and increasing function of time).

-For beta1, being the slope, we consider the difference between the long forward rate and the short forward rate.

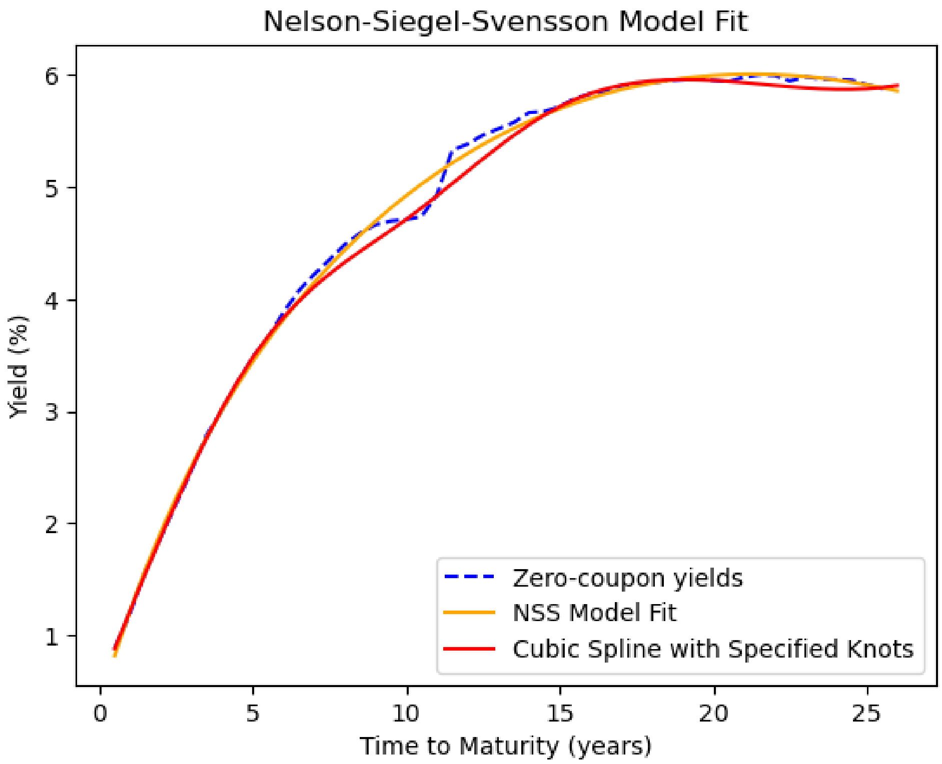
-For beta2, being the curvature parameter, it was harder to find a starting point. We tried to use an empirical approach to estimate the initial value for this parameter by creating a symmetrical initial estimate around the mean of the yield data.

-For beta3, being the second curvature parameter, we decided to use the same value as above, because to us, this value had little incidence and we didn't manage to find another way to estimate it.

-The values of tau1 and tau2 were find by realizing some tests. We tried to find the right balance realizing that smaller values of tau1 and tau2 result in steeper slopes and more pronounced curvature for shorter maturities in the yield curve whereas larger values make the decay slower, resulting in more gradual changes in the slope and curvature across different maturities.

-The 'curve_fit' fonction, as described above, helps us find best-fit parameters using least-squares fitting.

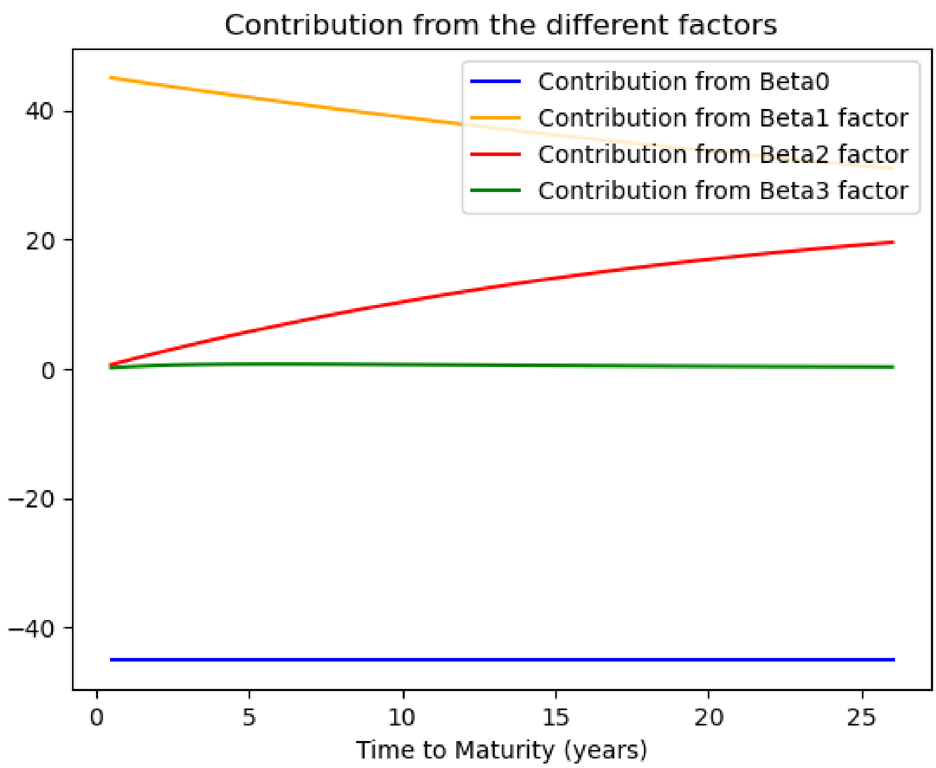
Here is the final graph we obtain if we plot our result against the curve obtained in Q2.1.



Q3.2) We can see that both methods (NSS Model Fit and Cubic Spline with Specified Knots) look accurate and quite equivalent, except for maturities around 7 to 15 years. For this specific period, what we can see is that NSS method is quite effective in finding the overall smooth shape of the curve whereas cubic spline looks more flexible and captures slightly more local behaviours in the yield curve.

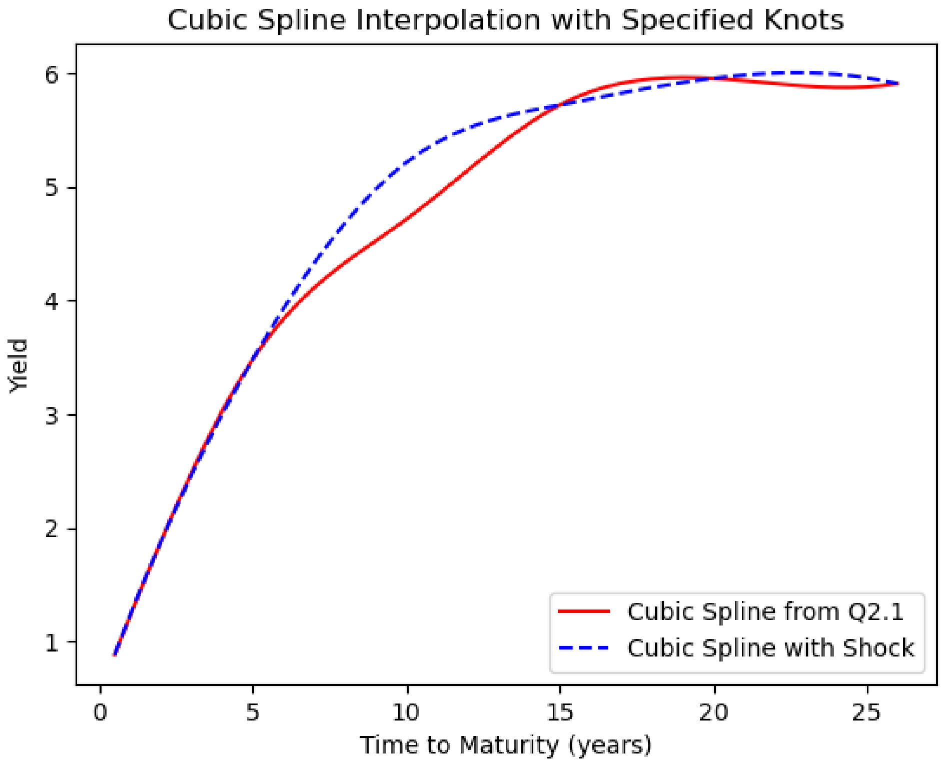
Q3.3) Below is the plot we obtain when analyzing the contributions from the different factors affecting the shape of the spot zero-coupon curve.

What we can see is that the contribution of Beta0 is negative and does not depend on time to maturity. Beta3 does not depend on the time to maturity either and has a contribution of 0, which makes sense taking into consideration our assumption that Beta2 and Beta3 are equal. However, this is not the case for Beta1 and Beta2. Beta1 has a higher contribution for short maturities whereas Beta2 has higher contribution for longer maturities.



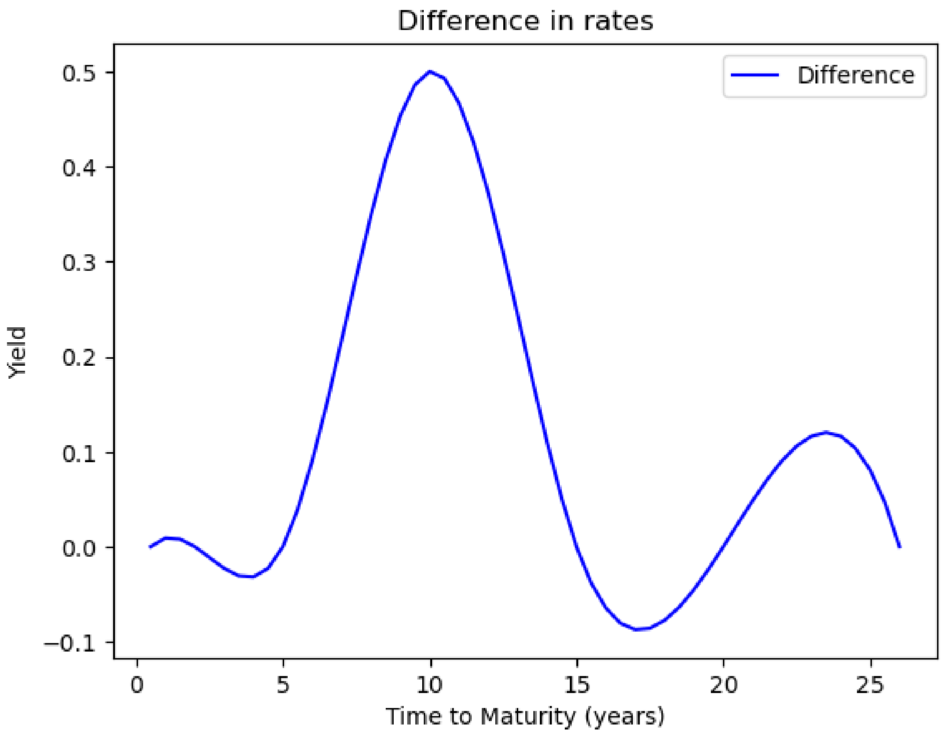
Question 4: Hedging

Q4.1) Below is the plot showing the discount function we obtain after shocking the 10-year rate by 50bps compared to the initial discount function.



At first sight, short maturities don't seem to be affected by the shock. The shock has a clearly more pronounced impact for maturities going from 5 to 15 years with still an impact afterwards but less significant.

To confirm this and better analyse the impact, we plot the below second graph, showing the difference between both discount functions (before shock and after shock). The conclusion we draw is that short maturities are nearly non affected (the change in yield is close to 0). The change starts to be more significant for maturities around 5 years (negative impact on yield at first) with a major change for maturities around 10 years (the yield change is close to 0.50% which makes sense given that this is where the shock of 50 bps occurs). Afterwards, the impact declines to reach a negative level for maturities from 15 to 20 years (15 and 20 being our knots, the change in yield is 0 at those points) and the change in yield starts to be positive again after that to reach a level close to 0 for maturities around 25 years.



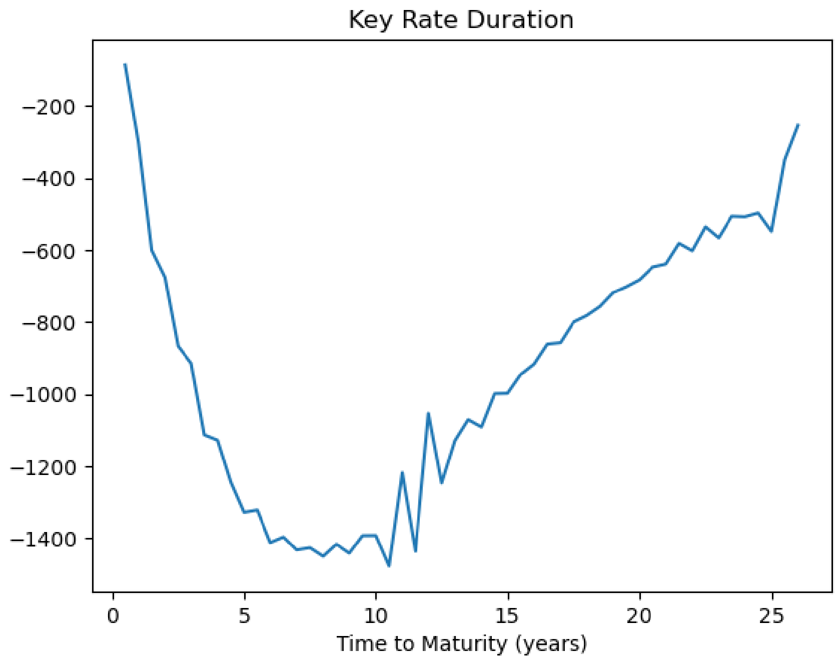
Q4.2) As the prices of the bonds are different, we have decided to define the weights as the price of the bond divided by the sum of the prices of all bonds, so it is equally invested in all bonds.

To value the KRD of our portfolio, we follow the below steps:

- We compute the portfolio present value using the reference (unperturbed) yield curve.
- We shock the 10y rate while keeping all others constant.
- We rebuild the discount curve using our previous cubic spline function to get the new (perturbed) yield curve.
- We re-estimate the PV given the new perturbated discount function.
- We compute the difference of PVs that is our KRD.

We obtain a KRD of –4,270.642452120672. This means that, for a portfolio worth 1 million dollar, if there is a shock of 50 bps (quite huge for a bond) on the 10-year bond, then the value of our portfolio fells by $4,270.

Q4.3) For this question, we shocked by 50 bps the yields of all maturities one by one. After shocking each maturity, we compute the KRD of our portfolio, so we finally obtain the below graph:

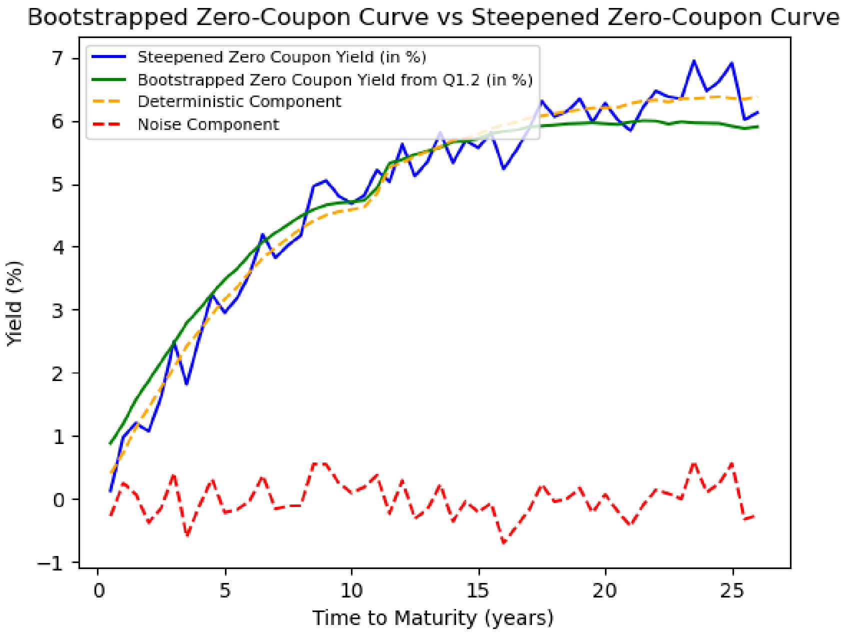


We observe that the KRDs is overall negative. However, we can see those shocking bonds of 6 to 10-year maturities has a more pronounced impact on the value of our portfolio.

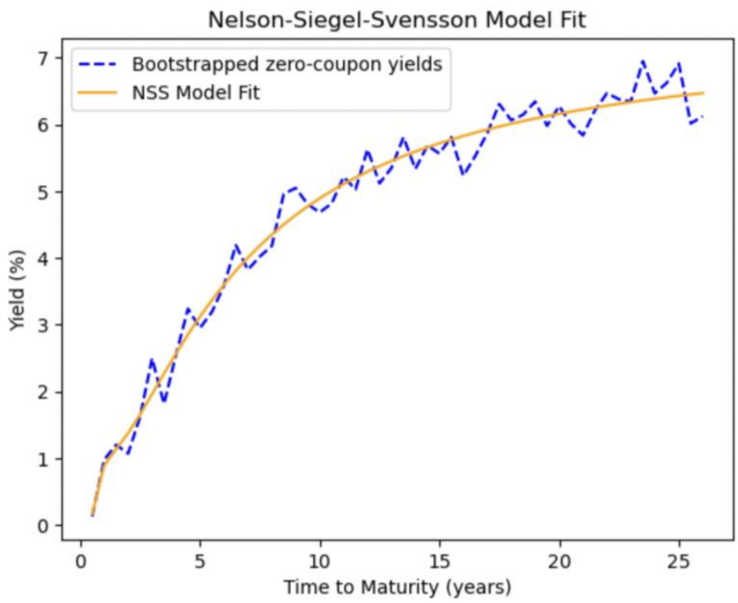
NB: To obtain this result, we assumed that a change in yield on one maturity i will affect the bonds that are close (with maturities i-1 and i+1). Therefore, we estimated the new yields by using a cubic spline functions with knots corresponding to all the maturities except the i-1 and i+1, to reflect the potential local perturbation. We tried the same method by modifying the deformation zone and therefore removing knots i-2, i-1 as well as i+1 and i+2. But the curve loses smoothness and accuracy. And that's a reason why, for the 10-year maturity, with only 4 knots we do not find the same KRD as the one we have for Q4.2). Finally, using the NSS model seems to be relevant for this work, and would allow local deformation to be reflected without neglecting the more or less significant effects on overall rates. But this requires the initial parameters to be rebalanced for each shock, which is more computationally expensive. A potentially interesting approach would be to use the parameters corresponding to the previous shock as the initial parameters for the next shock.

Question 5: Portfolio strategy

Q5.1) The new curve according to our expectation is the below one:



Q5.2) To re-estimate the discount function using the Nelson-Siegel-Svensson methodology, we first decide to plot the yield that are calculated with this formula: $F_t^{T:T+\tau} = \beta_0 + \beta_1 e^{\left(\frac{-T}{\theta}\right)} + \beta_2 \left(\frac{T}{\theta}\right) e^{\left(\frac{-T}{\theta}\right)} + \beta_3 \left(\frac{T}{v}\right) e^{\left(\frac{-T}{v}\right)}$



Now that we have the yield, we can calculate the price using the bootstrapped zero-coupon yield in the present value formula:

Coupon Rate (%)	Maturity Date	Maturity (years)	Current Price	Discounted Price	Zero Coupon Yield (in %)	Steepened Zero-coupon yields (in %)
3	2004-02-15	0.5y	101.0544	0.995610	0.881901	-0.000520
2.125	2004-08-15	1y	100.9254	0.988176	1.192962	-0.000901
1.5	2005-02-15	1.5y	99.8942	0.976738	1.575289	-0.000992
6.5	2005-08-15	2y	109.0934	0.963406	1.872718	-0.000917
5.625	2006-02-15	2.5y	108.4380	0.947375	2.174163	-0.000756
2.375	2006-08-15	3y	99.7848	0.928970	2.471112	-0.000561
6.25	2007-02-15	3.5y	111.7184	0.907564	2.790463	-0.000361
3.25	2007-08-15	4y	101.0841	0.887418	3.008378	-0.000176
3	2008-02-15	4.5y	99.1692	0.864791	3.254351	-0.000014
3.25	2008-08-15	5y	99.2710	0.841559	3.479914	0.000120
5.5	2009-02-15	5.5y	109.7707	0.819380	3.654950	0.000227
6	2009-08-15	6y	112.1450	0.794000	3.881711	0.000307
6.5	2010-02-15	6.5y	114.9084	0.769343	4.075084	0.000365
5.75	2010-08-15	7y	110.3894	0.746507	4.220336	0.000402
5	2011-02-15	7.5y	105.2934	0.724062	4.351720	0.000422
5	2011-08-15	8y	104.7607	0.701204	4.486530	0.000429

Coupon Rate (%)	Maturity Date	Maturity (years)	Current Price	Discounted Price	Zero Coupon Yield (in %)	Steepened Zero-coupon yields (in %)
4.875	2012-02-15	8.5y	103.4391	0.680072	4.587793	0.000424
4.375	2012-08-15	9y	99.2806	0.660381	4.663971	0.000410
3.875	2013-02-15	9.5y	95.0288	0.643389	4.696459	0.000390
4.25	2013-08-15	10y	97.7693	0.627754	4.710684	0.000365
13.25	2014-02-15	10.5y	174.3251	0.611738	4.735682	0.000336
12.5	2014-08-15	11y	168.9389	0.585340	4.928490	0.000304
11.25	2015-02-15	11.5y	157.0552	0.546185	5.328860	0.000271
10.625	2015-08-15	12y	152.4222	0.528678	5.382614	0.000237
9.25	2016-02-15	12.5y	140.0135	0.509846	5.462435	0.000203
7.5	2016-08-15	13y	123.3044	0.492711	5.519651	0.000169
8.75	2017-02-15	13.5y	136.0598	0.476050	5.574292	0.000136
8.875	2017-08-15	14y	137.5040	0.457551	5.663463	0.000104
9.125	2018-02-15	14.5y	140.7920	0.444315	5.673620	0.000072
9	2018-08-15	15y	139.9079	0.429353	5.716684	0.000042
8.875	2019-02-15	15.5y	138.7431	0.412853	5.789727	0.000012
8.125	2019-08-15	16y	130.7162	0.398744	5.829829	-0.000016

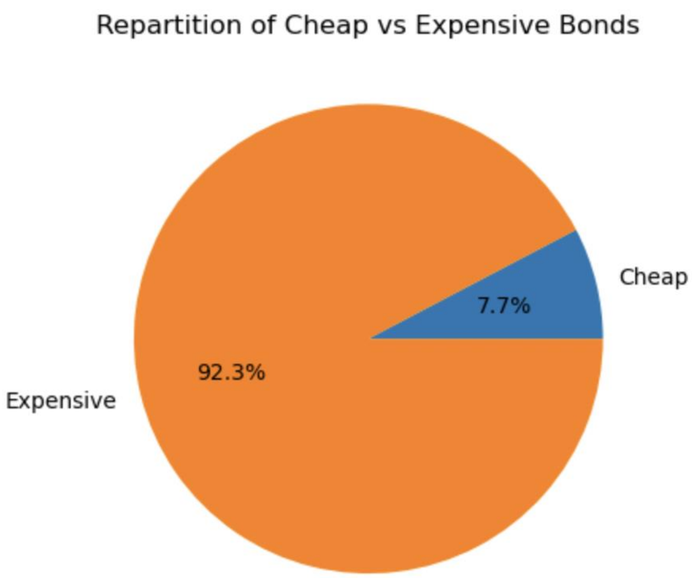
Coupon Rate (%)	Maturity Date	Maturity (years)	Current Price	Discounted Price	Zero Coupon Yield (in %)	Steepened Zero-coupon yields (in %)
8.5	2020-02-15	16.5y	135.2938	0.386180	5.850305	-0.000043
8.75	2020-08-15	17y	138.3466	0.371999	5.902262	-0.000069
7.875	2021-02-15	17.5y	128.4995	0.360481	5.916195	-0.000094
8.125	2021-08-15	18y	131.7341	0.349287	5.929877	-0.000118
8	2022-02-15	18.5y	130.4736	0.338053	5.949207	-0.000140
7.25	2022-08-15	19y	121.5800	0.327859	5.956296	-0.000162
7.125	2023-02-15	19.5y	120.1744	0.317791	5.966029	-0.000183
6.25	2023-08-15	20y	109.4538	0.309470	5.951302	-0.000203
7.5	2024-02-15	20.5y	125.4600	0.301222	5.939697	-0.000221
7.5	2024-08-15	21y	125.4466	0.290205	5.978892	-0.000240
7.625	2025-02-15	21.5y	127.1477	0.280453	6.001532	-0.000257
6.875	2025-08-15	22y	117.5509	0.272593	5.996206	-0.000273
6	2026-02-15	22.5y	106.3626	0.267598	5.945634	-0.000289
6.75	2026-08-15	23y	116.1986	0.257756	5.982261	-0.000304
6.625	2027-02-15	23.5y	114.7086	0.251277	5.964657	-0.000319
6.375	2027-08-15	24y	111.4036	0.244246	5.960339	-0.000333

Coupon Rate (%)	Maturity Date	Maturity (years)	Current Price	Discounted Price	Zero Coupon Yield (in %)	Steepened Zero-coupon yields (in %)
6.125	2028-02-15	24.5y	108.0391	0.237439	5.955713	-0.000346
5.5	2028-08-15	25y	99.6330	0.232986	5.912825	-0.000359
5.25	2029-02-15	25.5y	96.2876	0.228238	5.878319	-0.000372
6.125	2029-08-15	26y	108.4062	0.220240	5.904868	-0.000383

To decide if a bond is cheap or expensive, we compare the current price to the bootstrapped price.

- If current price is higher than bootstrapped price: the bond is expensive

- If current price is lower than bootstrapped price: the bond is cheap



From the pie chart, we observe that there is a majority of expensive bonds.

Q5.3) Creating a diversified zero-cost long-short portfolio strategy involves taking long position in the undervaluated bonds i.e. the cheap bonds. While taking short position in the overvalued bonds i.e. the expensive bonds. The aim of this strategy is to profit from the expected price convergence.

The expected return of each bond is given by the formula:

(Market_Price - Boostrapped _Price) x Investment_Amount

To calculate the expected return of the portfolio, we just need to sum the expected return of all the bonds.

With Python, we obtain:

Expected_return_portfolio = 761.29