Group 4 - APAC: Group Project FIN30160

Executive Summary

This report's objective is to create a fixed-income portfolio that was impervious to Delta and Gamma risks. The APAC group received an $8 billion initial investment from the UCD Investment bank's assets. We tested the portfolio in a simulation, which lasted 36 months, with hypothetical term structure changes. To achieve the portfolio's immunity, we re-immunized it every month and focused on the bonds' duration and convexity. The goal was to construct a portfolio that performed well in various scenarios and reduced the recorded tracking error of the simulation.

Our analysis involved several sections. In the first, we calculated the present value of our liabilities to plan our investment strategy accordingly. We aimed to immunize the portfolio by reducing interest rate risk and maintaining a significant amount of cash on hand to minimize risk. In the second section, we developed a strategy for simulating credit risk spreads and choosing a start date. We selected bonds with varying yields and maturities, adjusting suggested fictitious bonds with the real selected bonds in the third section. The fourth section involved managing cash inflows and outflows and developing strategies for selling bonds and updating portfolio composition.

In the fifth section, we changed the portfolio's composition and re-immunized it every month, focusing on duration calculation, estimating the present value of our liabilities, including convexity in immunisation, and selecting an optimal immunized portfolio. Finally, in the sixth section, we updated the portfolio's value, adjusted for real selected bonds, and improved the estimation of the present value of our liabilities.

Introduction and Background

As the APAC group, our goal is to understand and apply the concepts of fixed-income and interest rate term structure to design a portfolio of fixed-income instruments from the Asia-Pacific (APAC) region that are immune to Delta and Gamma risks. Our portfolio with $8bn million of the fictional UCD Investment bank's assets will be tested in a simulation with hypothetical changes to the term structure across time. The simulation will test the performance of our portfolio in different scenarios over a period of 36 months. Our goal was to improve the performance of the portfolio simulation provided to us, primarily by changing the composition of the portfolio by re-immunising the portfolio at the beginning of each month, focusing on the duration and convexity of the bonds. We will create a strategy for simulating credit risk spreads and improve the process for how cash inflows and cash outflows are managed. The Nelson Siegel model will be used to extract the yield curve.

Overall, our goal is to construct a portfolio of fixed-income instruments that can perform well in different scenarios, and to gain a deeper understanding of the fixed-income and interest rate term structure concepts, ultimately trying to reduce the recorded tracking error of the simulation.

Methodology

First Section

In section 1 of our analysis, we first calculated the present value (PV) of the liability. This represents the total amount of 36 payments of 0.01 times the assets under management (AUM) that we must pay over the course of the simulation. This calculation was crucial as it allowed us to understand our financial obligations and plan our investment strategy accordingly.

Our primary goal in this simulation was to immunize the portfolio by reducing interest rate risk. Therefore, we were willing to maintain a significant amount of cash on hand, as our objective was not to maximize returns but rather to minimize risk.

To determine the percentage of our funds to be invested in cash, we used the formula 1 - (PV_Liability/AUM). This calculation enabled us to invest only the required amount in bonds and maintain the rest as cash to meet our financial obligations.

Initially, we set the weights of the portfolio as evenly weighted, and this will change when we immunise the portfolio at the beginning of the second month.

Second Section

Deciding which periods are run by our simulation, and we have to ”create” a strategy for including default risk.

(a) For our start date, we chose a date according to the most comparable macroeconomic climate to today’s in order to make our simulation as realistic as possible. In order to match today’s economic situation to the most similar one during the period between 2005 and 2011, we subtracted the yield of US 10-year bonds from 2-year bonds, with the resulting figure being the slope of the yield curve slope for the date in question. When performing our analysis, based on our ‘today’s date’ of April 7th 2023, our yield curve was -0.58%, with a 2-year yield of 3.97% and a 10-year yield of 3.39%. The date that was closest to this was Nov 15th 2006, when the slope was -0.19 with the 2 and 10-year yields being 4.8% and 4.61% respectively. Both of these dates exhibit ‘inverted’ yield curves, seen as indicators of upcoming recessions as investors become fearful about the near term economic outlook, sending rates on earlier-maturity bonds higher than longer-dated peers. In November 2006, years of economic expansion led to fears that the upward cycle was about to end and of higher inflation. This is much akin today as recessionary fears are commonplace, as central banks across the world have raised interest rates in order to cool potentially overheating economies. Therefore, we believe our chosen start date of 15th November 2006 is an appropriate one to start our simulation on.

(b) In this section we are simulating credit spreads by calculating the 10-year interest rate for a given day and the interest rate for the same day one month prior. We begin by using the "getZeroRates" function to retrieve the 10-year interest rate for the current day, and store this value in the "R10Y_Note" variable.

Next, we want to simulate credit spreads, so we need to calculate the interest rate for the same day one month prior. To do this, we first convert the start date to a datetime object and subtract one month from it. We then calculate the ‘day difference' between the initial day and the one-month-previous day, which is more effective than simply subtracting 30 days, since it takes into account leap years and months with different numbers of days.

We then use the 'day difference' to find the index in the "RateSpecG2" array for the one-month-previous day, and use the "getZeroRates" function again to retrieve the 10-year interest rate for that day. This value is stored in the "previous_rate_10Y_note" variable.

<u>Third Section</u>:

Now that we have a strategy on what portion of our initial $8bn that we are to invest in bonds and we have a start date selected, we can adjust the suggested "fictitious" bonds with the "real" selected bonds. We selected 13 total bonds, 4 government bonds and 9 corporate bonds with varying yields and maturities. We utilized Bloomberg to identify a range of suitable fixed-income instruments from the Asia-Pacific (APAC) region. The objective was to create a diversified portfolio with a mix of government and corporate bonds, ensuring a range of maturities and yields for optimal risk management. By selecting a diverse range of bonds with different yields and maturities, our model has a greater pool of options to explore. This allows the model to incorporate them as needed, where we will create a list of all the possible combinations of bonds to immunise the portfolio. Our objective for the fund is not to have a high return on investment, but to immunize. For this reason we will likely be expecting lower yield and mid to long term bonds.

Bond	Class	Country	Mty	Days Invested
Australia 30y	**Gov**	Australia	30	43
Australia 10y			10	0
Japan 10y		Japan	10	437
South Korea 30yr		S. Korea	30	701
AIA Group	**Corp**	S. Korea	10	0
Korea Electric Power			5	43
Samsung Electronics			4	0
Standard Chd. Korea			12	307
CIMB Bank		Malaysia	5	0
Softbank		Japan	5	0
Panasonic			3	701
Adani Green Energy		India	3	0
Westpac		Australia	8	0

Above are the bonds we selected from the APAC Region. Referring to the 'Days Invested' column, these figures show how long in our simulation we were invested in each security (our methodology for finding this is to be discussed later). The Japanese 10 year, South Korean 30 year, Standard Chartered Bank Korea and Panasonic (Japan) bonds were the most selected bonds in our immunised portfolio across the span of our simulation. It is notable that seven bonds were not selected at all.

<u>Fourth Section</u>

In this section, we are tasked with handling cash inflows and outflows that occur at the beginning of each month. These cash flows are related to the fund retirements and deposits. Additionally, we need to take into account changes in the 10-year interest rate during the month and how it affects our portfolio.

In order to simulate credit spreads, we recognized that relying solely on the 10-year interest rate would not be enough. To address this, we calculated a weighted average of the credit spread in our portfolio and added it to the 10-year rate to simulate default risk. This allowed us to simulate default risk and get a more accurate representation of the cash inflows/outflows.

If there is an increase of more than 10bp in the 10-year interest rate during the month, we need to sell bonds to get cash for paying the retirements of the fund. Specifically, we should sell 100K per every 10bp increase. If we have enough cash on hand to cover the amount of cash we need to extract, we can simply subtract the cash from our cash balance. However, if we do not have enough cash on hand, we need to sell bonds to obtain the necessary funds.

To implement this in the code, we can add an "if" statement to check whether the increase in the 10-year interest rate is greater than 10bp. If it is, we can add another "if" statement to check whether we have enough cash on hand to cover the amount we need to extract. If we do, we can simply subtract the cash from our cash balance. Otherwise, we need to sell bonds to obtain the necessary funds. We do this by first finding the amount of cash needed in excess to our current cash on hand in order to pay the liability. Since we are assuming the portfolio is already immunized, we want each bond to maintain the same portfolio weighting after selling them. To do this, we found out what percentage (x%) of the portfolio needed to be sold based on how much cash we had, how much cash we needed from bond selling, and the portfolio value. With this percentage, we can sell x% of each of the bonds. This ensures the portfolio weights stay the same after the selling of bonds.

If there is a decrease of more than 10bp in the 10-year interest rate during the month, we do not need to sell bonds, but we might want to update the portfolio composition. Specifically, we should get 100K extra investment per every 10bp decrease.

Fifth Section

(a) In this section we need to adjust the suggested "fictitious" bonds with "real" selected bonds in the simulation. The code begins by setting the "Settle" variable to the current date. This variable will be used later when calculating bond prices. The code then enters a loop that will iterate over each bond in the dataset. For each bond, the code retrieves several pieces of information, including the coupon rate, maturity date, face value, coupons per year, and bond basis.

Next, the code calculates a set of zero dates for the bond using the current date as the starting point, with intervals of 365 days between each zero date and the maturity date. The zero rates are then retrieved using the "getZeroRates" function from the "RateSpecG2" object for the current day.

Using the retrieved zero rates, along with the bond's maturity date, coupon rate, face value, coupons per year, and bond basis, the code calculates the bond's price using the price by zero rates "prbyzero" function. This bond price is stored in the "Price_t" variable for that bond.

Finally, the code calculates the individual bond’s utility and stores it in the "r" variable.

(b) The duration of a portfolio is the weighted sum of the duration of the individual bond, with weighting coefficients proportional to individual bond price. If there are N bonds in a portfolio with price durations of P and D, with a common yield, the portfolio price (P) and Duration (D) are given by:

$$P = P_1 + P_2 + \cdots + P_N$$

$$D = \omega_1 D_1 + \omega_2 D_2 + \cdots + \omega_N D_N$$

$$\omega_i = P_i / P, \quad i = 1, 2, \cdots, N$$

Given a sequence of cash flows, the duration of the bond is:

$$D_i = -\frac{\frac{dP}{dy}}{P} = \frac{\sum_{k=1}^{n} \frac{dPV_k}{dy}}{P} = \sum_{k=1}^{n} \frac{c_k e^{-y(0,k)t_k}}{\sum_{k=1}^{n} c_k e^{-y(0,k)t_k}} \cdot t_k$$

Where y(0, k) is the yield to maturity k at time 0, PV_k is the present value of cash flow c_k at t_k, and c_n = coupon payment + principal payment at t_n. In this simulation each of the bonds has different yields and so the calculation for Duration becomes:

$$D_i = \frac{1}{S}\left(1 \cdot c_1 \cdot e^{-y(0,1)} + \cdots + T \cdot c_T \cdot e^{-y(0,T)T} + T \cdot Principal \cdot e^{-y(0,T)T}\right)$$

where,

$$S = t_1 \cdot c_1 \cdot e^{-y(0,t_1)t_1} + \cdots + T \cdot c_T \cdot e^{-y(0,T)T} + T \cdot Principal \cdot e^{-y(0,T)T}.$$

In order to enhance the accuracy of our duration calculations, we have made several modifications. Firstly, we have incorporated additional parameters into the 'bnddury' function. Additionally, we have replaced the value of r(i) used in the original file, which was based solely on the 10-year treasury rate, with a more comprehensive approach that takes into account credit spreads. To achieve this, we utilized the 'bndyield' function, which considers a variety of factors such as coupon rate, time to maturity, basis, period, face value and price. Moreover, we have updated the price variable with a new parameter, 'Price_T,' that is sensitive to fluctuations in treasury rates. By doing so, we ensure that the value of r(i) is not static, but instead changes dynamically as the simulation progresses. Overall, these adjustments have significantly improved the precision and reliability of our bond pricing calculations.

(c) In order to improve the estimation of the PV of liability, we adjusted the r value for each iteration of the fixed payments we were obliged to make. The first step was to create a vector named "dif" that calculates the difference between the current date and the payment dates stored in the "last_bus_date" vector. Every month we created a vector 'dif' which calculates the number of days between the current day and the payment dates. We then create a vector rate which gets the treasury rate from RateSpecG2 for each period of time in 'dif' e.g. 1 month interest rate, 2 month interest rate, etc. We calculated the present value of every fixed payment with the above data and sum them together to calculate the 'PV_liability'. Once the above is done, once a month we are dropping the payment date from 'last_bus_date' and then the next payment date. This process is repeated in the for loop.

(d) Convexity is a measure of the curvature of the relationship between bond price and yield, and is an important factor in immunizing a bond portfolio against interest rate changes. We computed the convexity of each bond in the portfolio using the 'bndconvy' function in MATLAB. The function takes as input the bond's interest rate, coupon rate, settlement date, maturity date, number of coupon payments per year, day-count basis, and face value. It calculates the convexity of the bond using numerical differentiation, which approximates the second derivative of the bond's price with respect to changes in interest rates. The output of the function is the bond's convexity and its per-period convexity. These values are then used in the immunisation process to ensure that the portfolio is protected against changes in interest rates. The convexity calculations are also integrated into the later sections of code, as can be seen from the code for the improvement of the estimation of the Present value of our Liabilities, in the Matric for solutions.

(e) In this section we were required to select the immunised portfolio. In order to decide which strategy to select, we focused primarily on minimizing the tracking error. This is the difference between the bond portfolio value and the "PV_liability." To do this, we simply used the line of code;

[~, immunisation_strategy_selected] = min(abs(TE(t,:)));

The min function returns two outputs: the minimum value found, and the index of the minimum value in the array. However, in this line of code, we are only interested in the index of the minimum value, which corresponds to the immunisation strategy that has the smallest tracking error. The tilde (~) is used to discard the first output (the minimum value) and only store the index of the minimum value in the variable immunisation_strategy_selected.

Therefore, we select the immunisation strategy that has the smallest tracking error, which indicates that this strategy is expected to provide the closest match between the bond portfolio value and the PV_liability. By minimizing the tracking error, the immunised portfolio can be constructed to match the liabilities as closely as possible, which is important for managing financial risks and ensuring that the portfolio can meet its obligations over time.

Sixth Section

Similarly to the above sections, we improve the calculation of the Present Value of the Liability and the fictitious bonds are replaced with the real bonds. This is executed in the code the same way as the above sections.

Nelson-Siegel Model - Recent Research

We have used the Nelson-Siegel Model in order to extract a yield curve from a set of bond prices. It is based on the idea that the yield curve can be decomposed into three factors: a long-term trend, a short-term deviation from that trend, and a curvature term that captures the shape of the curve. By estimating the parameters of this model, analysts can derive a curve that is consistent with market prices.

Outside of using the Nelson-Siegel Model for extracting a yield curve, it can also be useful in forecasting interest rates, analysing term structure dynamics, pricing bonds and derivatives, and estimating risk premia. The following 2 research articles discuss the uses or alternative methodology to the above version of the Nelson-Siegel Model.

(i) In the research article 'Kumar, S. and Virmani, V. (2021) "Term structure estimation with liquidity-adjusted affine Nelson Siegel model: A nonlinear state space approach applied to the Indian Bond Market,"' the authors discuss the challenges of term structure estimation in emerging markets, which are characterized not only by a lack of overall liquidity, but also by the concentration of liquidity in a few securities. The authors propose an extension of the popular Nelson-Siegel model called the Arbitrage-free Affine Nelson-Siegel (AFNS) model, which explicitly incorporates the phenomenon of liquidity concentration using a proxy for liquidity based on observable data in the bond pricing function.

The authors note that the attractiveness of the Nelson-Siegel model lies in its simplicity and ease of interpretation in terms of the level, slope, and curvature of the yield curve. However, the Dynamic Nelson-Siegel model has taken over from the original Nelson-Siegel model in applied fixed income research, and the AFNS framework is considered the state of the art for term structure modeling.

The AFNS framework retains the empirical power and intuitiveness of the classical Nelson-Siegel model while being arbitrage-free in its dynamics. However, most of the applications of AFNS so far have been to the case of liquid bond markets in developed economies. The authors implement the AFNS framework to model term structure dynamics for an emerging economy illiquid bond market, explicitly accounting for liquidity in estimation.

(ii) The study "Forecasting U.S. Yield Curve Using the Dynamic Nelson–Siegel Model with Random Level Shift Parameters" proposes a new model based on the dynamic Nelson-Siegel (DNS) model, which includes random level shift (RLS) parameters to capture cyclical fluctuations in interest rates and structural breaks induced by exogenous events like financial crises and major monetary policy interventions. The model incorporates macroeconomic and financial conditions to reduce persistence in the DNS residuals and improve forecasting accuracy of daily U.S. Treasury yields. The empirical results indicate that the model outperforms the benchmark DNS model and other popular models. The model performs particularly well for short-term and long-term bonds, and its performance improves as the forecasting horizon increases. The RLS process not only provides a better in-sample fit by incorporating infrequent breaks associated with major macro and financial events, but also generates possible out-of-sample breaks that are crucial to forecasting under an unstable environment. This study suggests that the DNS model with RLS parameters can be used for forecasting the term structure of interest rates, bond portfolio management, derivatives pricing, and risk management.

Results

The primary objective of this project was to optimize the performance of the code provided to us, through the methodology described above. The results of which we will discuss below.

Through the method described under the methodology section, the resulting starting cash amount out of the $8 billion was $5.3 billion which represents 66.25% of the portfolio. Although high, the main goal of the simulation was to immunise the portfolio from both Delta and Gamma risks.

The starting day we decided upon was 15th November 2006. This makes intuitive sense as discussed above, as by using the 10-2yr yield curve as a proxy for broader macroeconomic conditions we find the date within the 2005 to 2011 period that best aligns with today.

We successfully replaced the fictitious bonds with real bonds. The real bonds selected by the model for use tended to be low yield as generally, bonds with higher yields have higher interest rate risk because they are more sensitive to changes in interest rates. Bonds with high coupon rates also tend to be preferable as they provide a higher level of income, which can offset any potential losses if interest rates rise. For the above reasons our model did not invest in all of the available bonds as certain bonds would lead to a portfolio with less interest rate risk.

We successfully immunised the portfolio at the beginning of each month, by matching the duration, convexity and maturity of the bonds against each other and solving for the optimal portfolio combination. Our code created a matrix of solutions from which the best solution could be selected. We also improved the methodology for the calculation of present value.

The primary metric for which we used to assess the improvements to the model was that of the tracking error of the simulation. After optimising the model, we successfully reduced the tracking error down to $7.65m average tracking error. The average daily tracking error divided by the AUM of that day was 0.0011. If we instead run an equally weighted portfolio, we get an average tracking error of $78.8m and an average tracking error divided by AUM of 0.0124. Our strategy clearly improves the tracking error of the portfolio.

Summary, Conclusions and Recommendations

In conclusion, we completed the objective of the project which was to immunize the portfolio against Delta and Gamma risks. We reduced the tracking error and have created a Matlab file which clearly outlines the improvements made upon the original code. However, there were many limitations on this project, and there are many things we could have done to further improve the accuracy of our simulation.

The yield curve provided under the variable name “RateSpecG2” contains the interest rate term structure of US treasuries securities. Our designated region was APAC, so interest rate data from APAC may have led to more accurate results if we wanted to draw more realistic conclusions outside of a simulated bank setting.

The immunisation method we used to find the optimal portfolio bonds involved selecting a combination of bonds to match the duration and convexity of the liability. The current approach tests all combinations of bonds and computes the weights using a linear optimisation approach. A non-linear approach could yield better and more computationally efficient results. One possible approach is to use a quadratic programming (QP) algorithm to determine the optimal bond allocation. The QP algorithm can minimize the tracking error while incorporating constraints on the portfolio's risk, incorporating duration and convexity.

One potential improvement to the immunisation strategy for managing bond portfolios is to include Monte Carlo simulations of possible future scenarios. The interest rate term structure is dynamic, and current bond prices only provide a precise scenario of it. By incorporating Monte Carlo simulations, the immunisation code can take into account a range of possible future interest rate scenarios, which may better prepare the portfolio for unexpected changes. This optional section would enhance the existing immunisation code. By doing so, we could better mitigate the risk of interest rate changes and ensure that the portfolio's value is better protected over time.

Personal Contributions:

We split the work as evenly as possible, all working together to solve each section of code and contributing to the report, complying with the rule of 100%/n (+/-) 10% of the project.

Group Member	**Student Number**
Sam Byrne	19334421
Rajat Nepal	22210602
Hugo Brady	19412776
Fergal McCluskey	19366041

References

Kumar, S. and Virmani, V. (2021) “Term structure estimation with liquidity-adjusted affine Nelson Siegel model: A nonlinear state space approach applied to the Indian Bond Market,” Applied Economics, 54(6), pp. 648–669. Available at: https://doi.org/10.1080/00036846.2021.1967866.

Luo, D., Pang, T. and Xu, J. (2021) “Forecasting U.S. yield curve using the dynamic Nelson–Siegel model with random level shift parameters,” Economic Modelling, 94, pp. 340–350. Available at: https://doi.org/10.1016/j.econmod.2020.10.015.