Financial Programming in C++: Homework Assignment 7

Fall 2025,

MSQF, Fordham University

Due: October 23rd, 2025

1 HW7

- To complete problems 1 to 6, follow te the instructions in the file HW7.cpp
- To complete problem 7, follow the instructions below in this document.

You must submit your answers by returning

- a completed `HW7.cpp` and the results of running the program in a file `HW7.csv`.
- the file `solve_portfolio.cpp` and the results of running that file in a file `solve_portfolio.txt`

Problems 1-6

Problem 1

in Problem 1 you must implement the function

```
double bond_price(double notional, Date maturity,
                  double coupon, int frequency,
                  Date valuation_date, double rate);
```

pricing a fixed income bond with notional N paying an annual coupon c with frequency f.

The bond price is given in terms of the coupon payment dates by

$$\begin{aligned}\text{bond\_price} &= N\left(\text{df}(d_0) + \frac{c}{f}\,(\text{df}(d_0) + \text{df}(d_1) + \cdots + \text{df}(d_k))\right)\\ &= N\left(\text{df}(d_0) + \frac{c}{f}\sum_{i=0}^{k}\text{df}(d_i)\right)\end{aligned}$$

the *coupon payment dates* $d_0, d_1, \ldots, d_k$ are separated by $\frac{365}{f}$ days:

$$\begin{aligned}
d_0 &= \text{bond_maturity} \\
d_1 &= \text{bond_maturity} - 365\frac{1}{f} \\
d_2 &= \text{bond_maturity} - 365\frac{2}{f} \\
&\vdots \\
d_{i+1} &= \mathrm{d}_i - \frac{365}{f} \\
&\vdots \\
d_k &= \text{bond_maturity} - 365\frac{k}{f}
\end{aligned}$$

and k is the *largest* k such before the bond valuation date:

$$d_k \geq \texttt{valuation\_date}$$

The *discount factor* at date d_i is given by:

$$\mathrm{df}(d_i) = e^{-r\,t_i}$$

where t is the time (measured in years) from `valuation_date`:

$$t_i = \frac{d_i - \text{valuation\_date}}{365}$$

and r is the level of interest rates as of `valuation_date` in the bond currency,

Problems 2-6

Follow instructions on file `HW7.cpp`

Problem 7

Rename file `solve_portfolio0.cpp` to `solve_portfolio.cpp`. For this problem, do all your work in file `solve_portfolio.cpp` Consider the portfolio risk described in the lecture notes.

We have a portfolio with the following risk exposures:

Risk Metric	Value
Delta	$100,000,000
Rho	$ -50,000
Vega	$ 1,000,000

We want to hedge with three options with the following risks:

Option	Delta	Rho	Vega
Option 1	\$50,000,000	\$ -25,000	\$ 700,000
Option 2	\$20,000,000	\$ -15,000	\$ 150,000
Option 3	\$20,000,000	\$ -20,000	\$ 350,000

use **LU** decomposition to solve for the notional of Options 1,2 and 3, that replicate the portfolio risks.

Your program must output only the portfolio weights (you do not need to output intermediate computations).