APM466 A1: Yield Curves

Victor Su, Student Number: 1008839330

January 2025

Fundamental Questions - 25 points

1.
   (a) Printing money increases the amount of money in circulation which causes inflation. Selling bonds does not necessarily increase money in circulation. [1]

   (b) Normally long term yields are higher, to compensate for inflation and keeping money locked up for a long period. Yield curve flattening can occur when there is a greater demand for long term bonds as this lowers the construct of yield, and can be a sign for recession or expectation of dropping interest rates. [3]

   (c) Quantitative easing is a type of monetary policy that a central bank can pursue by buying securities such as government bonds in the market to decrease interest and increase money supply, thereby influencing the economy. [2]

2. The 10 bonds I chose:

   - CAN 1.25 3/1/2025
   - CAN 0.5 9/1/2025
   - CAN 0.25 3/1/2026
   - CAN 1.0 9/1/2026
   - CAN 1.25 3/1/2027
   - CAN 2.75 9/1/2027
   - CAN 3.5 3/1/2028
   - CAN 3.25 9/1/2028
   - CAN 4.0 3/1/2029
   - CAN 4.5 9/1/2029.

   This group of 10 bonds were the only 10 that were evenly spaced 6 months at a time and covered almost 5 years in maturities. This simplifies the bootstrapping process as we do need to do interpolation. These bonds also have close and roughly growing coupon rates with maturity.

3. If the stochastic curve process is given by $X = (X_1, \ldots X_n)^T$, then the eigenvalues in increasing order of size maximize $\text{var}(Xu_i)$ for unit vectors u. The eigenvectors are the direction corresponding to the eigenvalues. In other words, the eigenvector of the largest eigenvalue represents the direction that captures the most variance of X and the eigenvalue represents the proportion.

Empirical Questions - 75 points

4.

(a) I used a 365 day convention for this assignment. After adjusting all prices to dirty price, I iterated through days then bonds, and calculated the YTMS by using scipy's fsolve function which is essentially a version of gradient descent to find the roots of the equation

$$0 = -P + \sum_{i=1}^{n+1} p_i(1+r)^{-t_i}$$

where $p_i = [c] * n + [100 + c]$ is the list of cashflows, c the semicoupon, and n is the position of the bond in order of maturities.

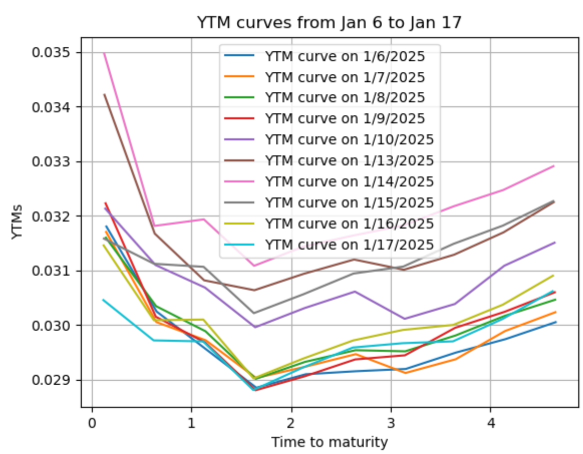

Figure 1: YTM curves

(b) Here I use yield to mean spot rate. The algorithm is as follows

i. data = []

ii. iterate through days

iii. yields = dictionary of (time to maturity, spot rate)

iv. iterate through bonds with index i

v. calculate time_delta = time to maturity as fraction of year

vi. calculate_yield(yields,semi_coupon, price, time_delta, i) with bootstrapping:

A. cashflows = [semi-coupon] * i + [100 + semi-coupon]

B. count = 0

C. add up known discounted cashflows with $sum = \sum_{k=0}^{i-1} \text{cashflows}[i]e^{-r[i]*t[i]}$ with r = yield.values() and t = yield.keys().

D. return $-\log((P - sum)/cashflows[i])/time\_delta$

vii. yields[time_delta] = calculate_yield(yields,semi_coupon, price, time_delta, i) so we add the newly calculated spot yield to the yields dictionary which will be used to calculate the yield with the next bond.

viii. data.append(yields.values()) which will be used later.

ix. return data

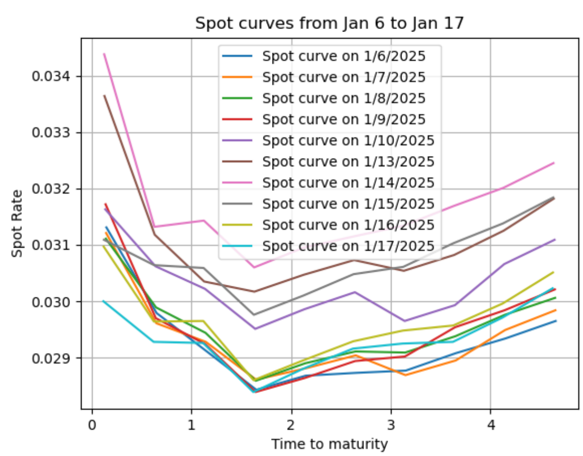

Figure 2: Spot curves

(c) The algorithm is as follows:

i. data = yields from the short curve calculation, so we have yields at each bond for every day

ii. we want the 1Y,2Y,3Y,4Y,5Y rates so we use np.interp on data at [1,2,3,4,5] giving the linear interpolations for those times.

iii. iterate through i in range(1,5)

iv. calculate the 1Y-t[i]Y using the formula

$$r(0, 1, t[i]) = \frac{r[i] * t[i] - r[1]}{t[i] - 1}.$$

Here r is the linearly interpolated yields at times t=1,2,3,4,5, and t is just times [1,2,3,4,5]

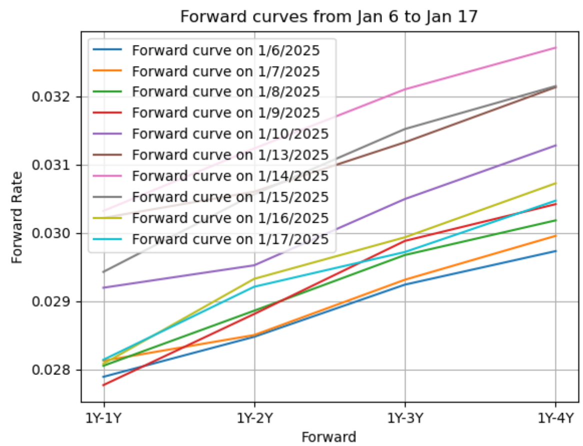

Figure 3: Forward curves

5. Here are the tables for the covariances of YTM and Forwards. The covariance for YTM is 5x5 and covariance for forward is 4x4.

	X_1	X_2	X_3	X_4	X_5
X_1	4.83e-04	5.14850e-04	4.74230e-04	4.84950e-04	4.69780e-04
X_2	5.14850e-04	6.13110e-04	5.44550e-04	5.45300e-04	5.43540e-04
X_3	4.74230e-04	5.44550e-04	5.22820e-04	5.35890e-04	5.16700e-04
X_4	4.84950e-04	5.45300e-04	5.35890e-04	5.69140e-04	5.43540e-04
X_5	4.69780e-04	5.43540e-04	5.16700e-04	5.43540e-04	5.29190e-04

Table 1: Covariance Matrix for Log-returns of YTM.

	X_1	X_2	X_3	X_4
X_1	8.7065e-04	6.3404e-04	6.0934e-04	6.1800e-04
X_2	6.3404e-04	5.7364e-04	5.7754e-04	5.3930e-04
X_3	6.0934e-04	5.7754e-04	6.2001e-04	5.7037e-04
X_4	6.1800e-04	5.3930e-04	5.7037e-04	5.4367e-04

Table 2: Covariance Matrix for Log-returns of Forwards.

6. The largest eigenvector and eigenvalue represents the curve level. This is because the eigenvector has all equal values (approximately), so is a

constant shift across the curve.

λ	2.09412080e-02	4.71217655e-04	2.53701304e-04	6.99148371e-05	7.76119514e-06
$\vec{v}$	$\begin{bmatrix}0.4147\\0.4724\\0.4437\\0.4581\\0.4452\end{bmatrix}$	$\begin{bmatrix}-0.5340\\-0.5259\\0.1573\\0.5505\\0.3322\end{bmatrix}$	$\begin{bmatrix}0.7257\\-0.6295\\-0.0789\\0.2176\\-0.1533\end{bmatrix}$	$\begin{bmatrix}-0.0892\\-0.0916\\0.8146\\-0.0679\\-0.5617\end{bmatrix}$	$\begin{bmatrix}-0.0908\\0.3091\\-0.3296\\0.6596\\-0.5937\end{bmatrix}$

Table 3: Eigenvalues and Eigenvectors of YTM Covariance.

	λ_1	λ_2	λ_3	λ_4
Eigenvalue	1.9523×10^{-2}	1.1603×10^{-3}	1.6231×10^{-4}	1.8073×10^{-5}
$\vec{v}_1$	0.5648	0.4769	0.4863	0.4659
$\vec{v}_2$	0.7961	-0.1905	-0.5332	-0.2136
$\vec{v}_3$	-0.0708	0.8056	-0.1677	-0.5638
$\vec{v}_4$	0.2053	-0.2954	0.6716	-0.6477

Table 4: Eigenvalues and Eigenvectors of Forward Covariance.

References and GitHub Link to Code

References

[1] Investopedia. "What Are Open Market Operations?" Available at: `https://www.investopedia.com/ask/answers/06/openmarketoperations.asp`. Accessed: February 3, 2025.

[2] Investopedia. "What is Quantitative Easing (QE)?" Available at: `https://www.investopedia.com/terms/q/quantitative-easing.asp`. Accessed: February 3, 2025.

[3] Millennial Revolution. "A Flattening Yield Curve—What Does It Mean?" Available at: `https://www.millennial-revolution.com/invest/a-flattening-yield-curve-what-does-it-mean/`. Accessed: February 3, 2025.

-Github: https://github.com/VictorSu33/APM466