MAT1856/APM466 Assignment 1

Gavin Liang, Student #: 1010728492

February, 2024

Fundamental Questions - 25 points

1.

(a)

Bonds are issued to mitigate deficits in government budget and control the country's money supply, whereas printing out money means inflation and the reality that the central bank cannot really take charge of the money supply once the money is in the market, which can cause economic harms. [1]

(b)

An example could be that the macroeconomic/long-term outlook is bad for the country, meaning that the investors do not believe in the market's future growth potential, making them more likely to invest in short-term securities.

(c)

Quantitative easing is a monetary policy that aims to purchase securities from the open market so that more money can be supplied to the market, thereby reducing interest rates and encouraging economic activities, and the US Fed employed this policy by lowering the federal funds rate for overnight borrowing, purchasing Treasury securities back, and establishing fund lending programs. [2]

2.

The 10 (11) bonds that I have selected are:

CAN 2.25 Mar 24/CAN 1.5 Sep 24/CAN 1.25 Mar 25/CAN 0.5 Sep 25/CAN 0.25 Mar 26/CAN 1 Sep 26/CAN 1.25 Mar 27/CAN 2.75 Sep 27/CAN 3.5 Mar 28/CAN 3.25 Sep 28/CAN 4 Mar 29

The reason why I selected those 10 (11) bonds is that there is an exact half-year increment in their maturity dates, which makes it possible for me to calculate the yield curve based on half-year intervals and makes calculations practical. If we choose some bond maturing on another date, we need to do extra calculations to get the interpolated values. In terms of the reason why I chose the bond maturing on Mar/01/2024 as my starting point, it is because in the year of 2028, there will be only three bonds maturing in that year on Mar/01, Jun/01, Sep/01, respectively. Since we want a yield curve based on half-year intervals, we need to select the bonds maturing on Mar/01 and Sep/01.

3.

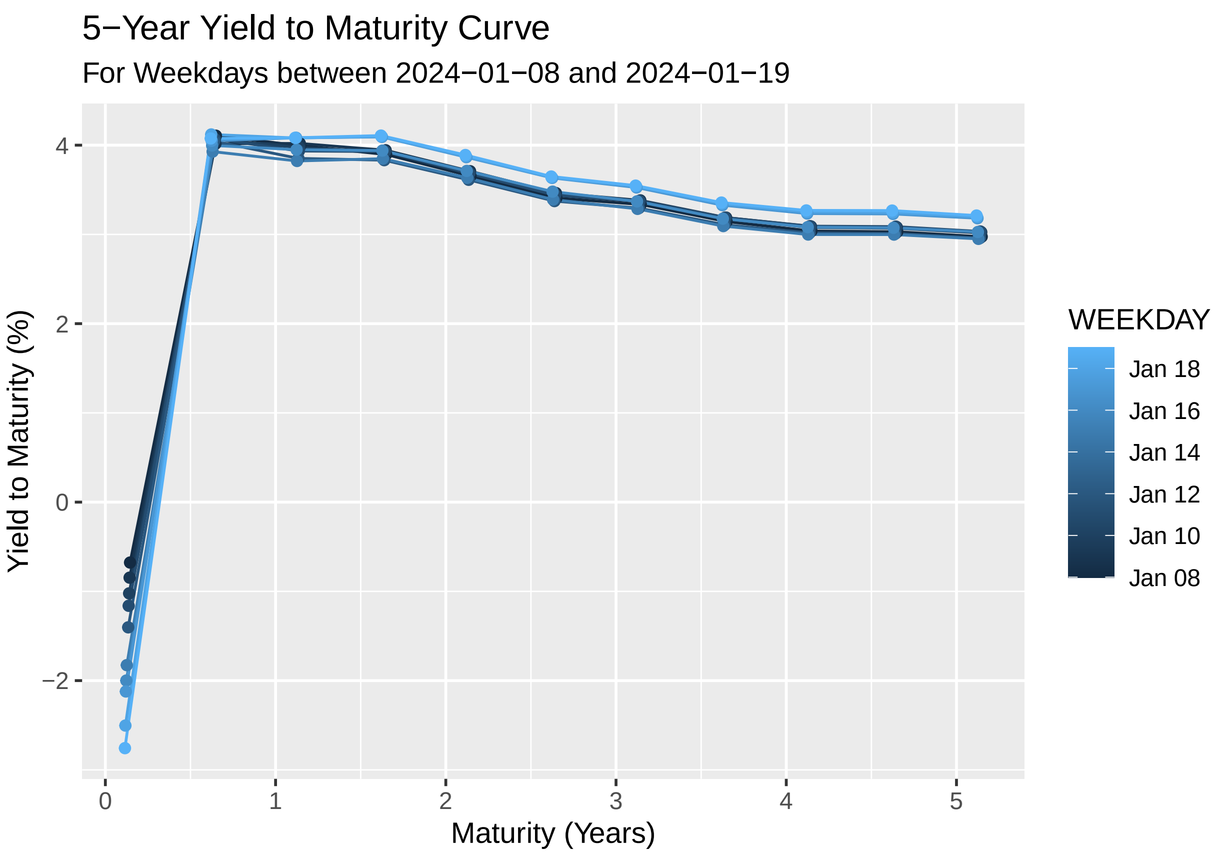
The eigenvalues associated with the covariance matrix of those stochastic processes are informative of the directions of movement of return rate curves. Moreover, the eigenvalues imply how much variance can be explained by a given principal component and therefore its associated eigenvector. The larger the eigenvalue is, the greater the variability along the corresponding eigenvector. In the context of stochastic curves, higher eigenvalues imply points of the curve where the associated stochastic processes show higher variability. As for eigenvectors, they tell us the direction in which the data varies the most. Hence, in the context of stochastic curves, they can show us the patterns of variation of the respective stochastic processes on the curve. As a result, we can use eigenvalues and eigenvectors to compare different stochastic processes. [3][4]

Empirical Questions - 75 points

4.

(a)

I used the yield to maturity formula according to Investopedia, which gives $YTM = \frac{C+\frac{FV-PV}{t}}{\frac{FV+PV}{2}}$.



(b)

a vector of 0's to store spot rates spot_rates=numeric(length(maturity_value))

for (each maturity_value in [1,2,3,4,5]):{ cash_flows=[coupon,coupon,...,100+coupon] #a number of cash flows to be discounted

time_periods=[1,2,...,maturity_value] # a number of time periods where payments are present

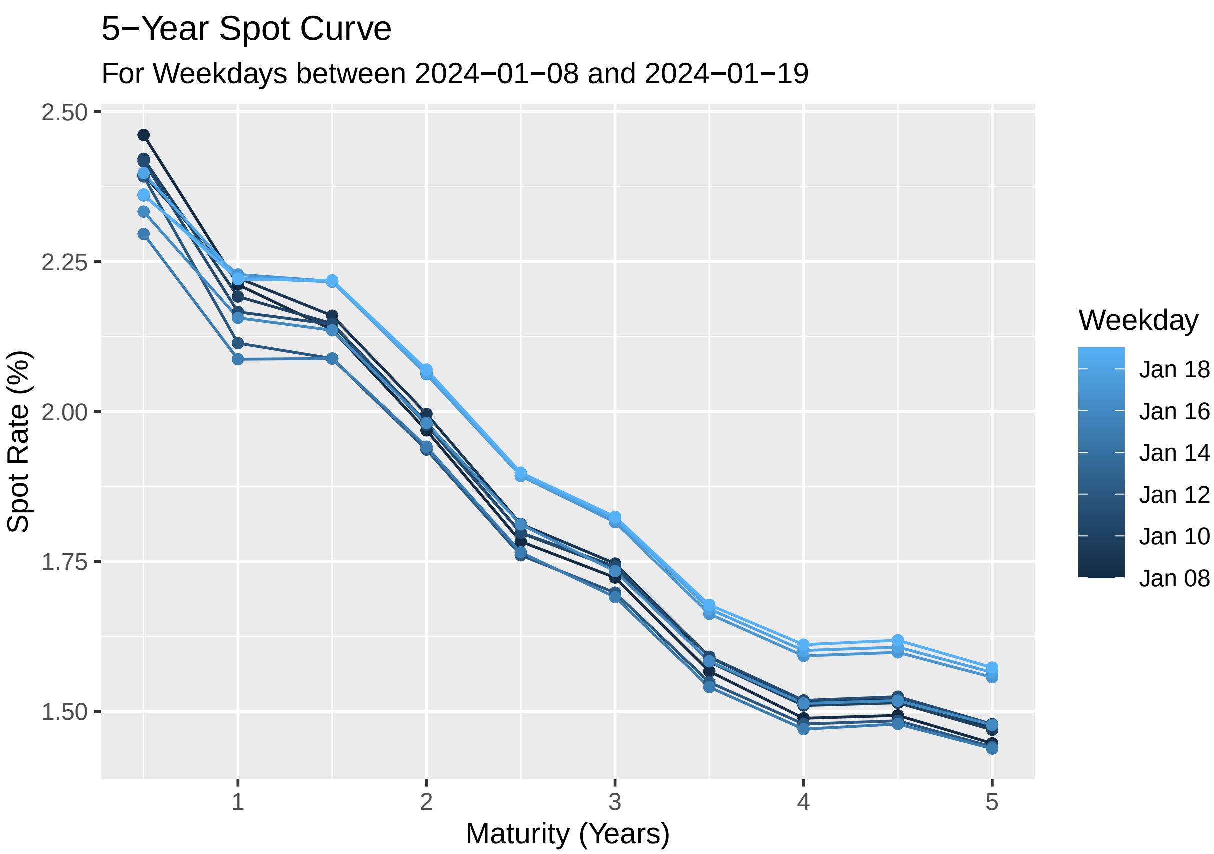
present_value=dirty price of each bond #prices looked up online

#bootstrap part to calculate spot rates for (each spot_rate_approximation in spot_rates_guess):{ discounted_cash_flows=[cf/(1+spot_rates_guess[j])**time_periods[j] for j, cf in enumerate(cash_flows)] #to discount future cashflows into present value

```
if (abs(present_value-discounted_cash_flows)<=0.0001){
#compare the different between the present value and price to see how accurate the spot rate guess is

    spot_rates[i]=each spot_rate_approximation
    #if approximation is very good, we can use this rate.
}
```

} }



(c)

#We suppose we already have a vector of spot_rate for a given day from (b), where the i-th element is the spot rate for bonds with i years to maturity. spot_rates=c(s1,. . . ,s5)

#We now define a vector of 0's so that the results can be stored. forward_rates=numeric(length(spot_rates))

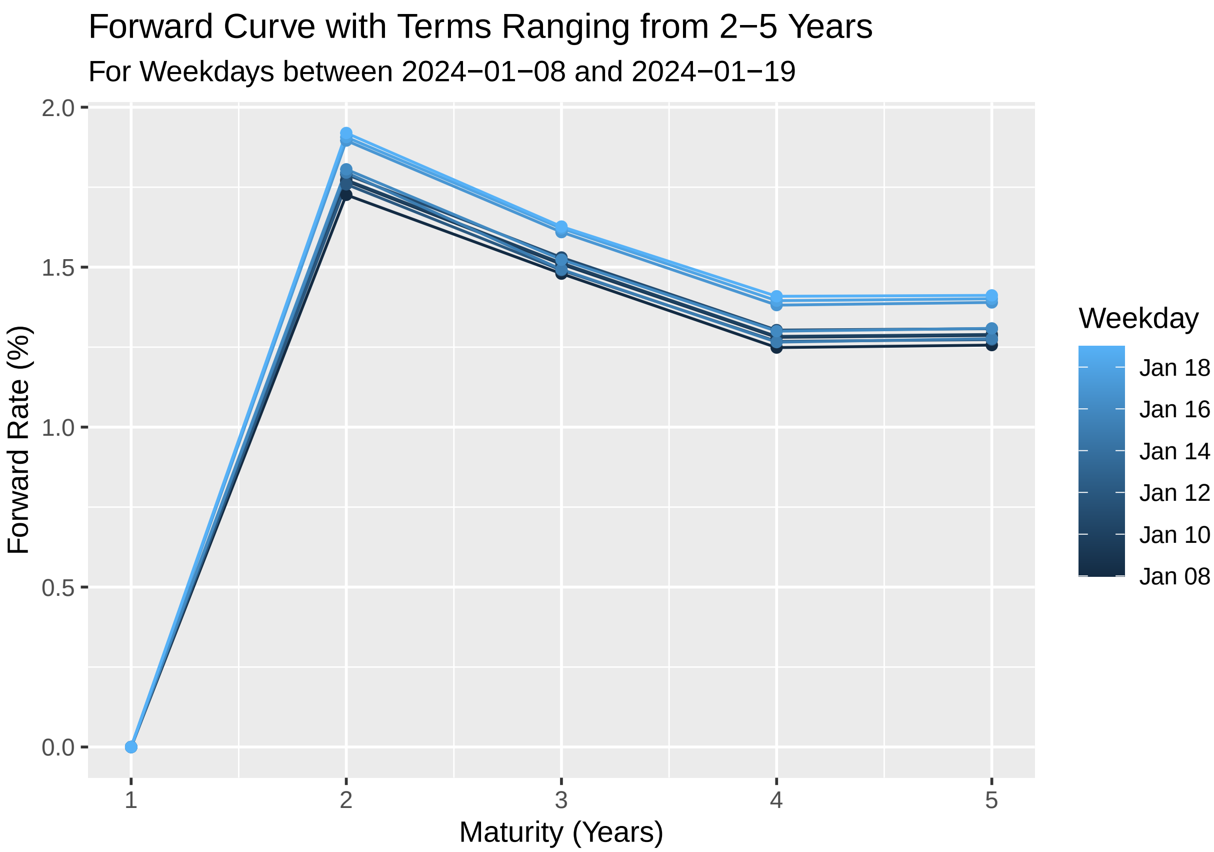
#We then loop over the spot_rate vector so that each possible forward rate can be calculated; for example, if we have 5 spot rates for 5 different years of maturity, then we can compute 1-1, 1-2, . . . , 1-5 forward rates.

for (each spot rate in spot_rates):{

#We now calculate the forward rate according to the formula. Since coupons are paid out semi-annually, we need to do semi-annual compounding. forward_rate=((1+spot_rates[each])^(2*each)/(1+spot_rates[starting_point])^2)(1/(2(
1)))-1

for example, the forward rate for 1-2 years is: forward_rate=((1+s2)^(2*2)/(1+s1)$^{2)}$(1/2)-1

#We finally store the resulting forward rate in the defined vector, which we can then bind with our existing dataset. forward_rates[i]=forward_rate }



5.

```
##              [,1]         [,2]         [,3]         [,4]         [,5]
## [1,] 0.0003614959 0.0003039153 0.0003525971 0.0003740904 0.0003677121
## [2,] 0.0003039153 0.0002952883 0.0003306420 0.0003481856 0.0003491018
## [3,] 0.0003525971 0.0003306420 0.0003851637 0.0004097288 0.0004079117
## [4,] 0.0003740904 0.0003481856 0.0004097288 0.0004389692 0.0004354884
## [5,] 0.0003677121 0.0003491018 0.0004079117 0.0004354884 0.0004336248
```

```
##              [,1]         [,2]         [,3]         [,4]
## [1,] 0.0003314661 0.0003524267 0.0003769044 0.0003771520
## [2,] 0.0003524267 0.0004645853 0.0005140739 0.0005003976
## [3,] 0.0003769044 0.0005140739 0.0005768100 0.0005568490
## [4,] 0.0003771520 0.0005003976 0.0005568490 0.0005408147
```

6.

Below is the eigenvalues and eigenvectors for the yield returns.

```
## eigen() decomposition
## $values
```

```
## [1] 1.865995e-03 3.524286e-05 1.261065e-05 6.558668e-07 3.764125e-08
##
## $vectors
##            [,1]       [,2]        [,3]        [,4]        [,5]
## [1,] -0.4224921  0.8969000 -0.07213044 -0.01430276  0.10799724
## [2,] -0.3908269 -0.1075361  0.88264768 -0.22542248 -0.07621093
## [3,] -0.4535427 -0.1582411 -0.05419797  0.78237091 -0.39270490
## [4,] -0.4827066 -0.2167721 -0.42688924 -0.57908063 -0.44992815
## [5,] -0.4796333 -0.3346281 -0.17480969  0.03926217  0.79112326
```

Below is the eigenvalues and eigenvectors for the forward rates.

```
## eigen() decomposition
## $values
## [1] 1.847519e-03 6.388020e-05 1.987150e-06 2.897625e-07
##
## $vectors
##            [,1]       [,2]        [,3]        [,4]
## [1,] -0.3884358  0.9079683 -0.14901346 -0.05006186
## [2,] -0.5003893 -0.1112840  0.76031884 -0.39892563
## [3,] -0.5542580 -0.3628751 -0.63028013 -0.40480440
## [4,] -0.5399317 -0.1775698  0.04957026  0.82127069
```

The first eigenvalue implies the magnitude of the variability along the first principal component and its associated with eigenvector implies the principal component direction, of which both eiganvalues are small.

References and GitHub Link to Code

1. Investopedia: https://www.investopedia.com/terms/y/yieldtomaturity.asp#:~:text=Yield%20to%20maturity%20is%2(

2. Investopedia: https://www.investopedia.com/terms/q/quantitative-easing.asp

3. Sebastian Raschka https://sebastianraschka.com/Articles/2015_pca_in_3_steps.html

4. OARC Stats https://stats.oarc.ucla.edu/spss/seminars/efa-spss/

5. GitHub: https://github.com/Gavinz23/mat1856

I was not able to minimize my plots. Except for the big plots, everything else should be within the limit of 3 pages.