Apm466 Assignment 1

tiaotiao jiang

2025-02-03

Read bond data

```
library(readr)
bond_data <- read_csv("bond_data.csv")
```

```
## Rows: 11 Columns: 15
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr  (4): name, rating, issue_time, maturity_time
## dbl (11): coupon, 2025/1/6, 2025/1/7, 2025/1/8, 2025/1/9, 2025/1/10, 2025/1/...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

Set up the data

```
trading_dates<- as.Date(colnames(bond_data)[6:15], format = "%Y/%m/%d")
bond_data$issue_time<- as.Date(bond_data$issue_time, format = "%Y/%m/%d")
bond_data$maturity_time<- as.Date(bond_data$maturity_time, format = "%Y/%m/%d")
coupon <- bond_data$coupon*0.01
```

close_matrix

```
close_matrix <- as.matrix(bond_data[1:11, 6:15])
print(close_matrix)
```

```
##       2025/1/6 2025/1/7 2025/1/8 2025/1/9 2025/1/10 2025/1/13 2025/1/14
##  [1,]    99.73    99.73    99.74    99.73     99.74     99.73     99.73
##  [2,]    98.40    98.41    98.40    98.40     98.37     98.36     98.36
##  [3,]    96.99    96.98    96.97    97.00     96.90     96.91     96.80
##  [4,]    97.01    96.99    97.00    97.03     96.86     96.77     96.71
##  [5,]    96.62    96.59    96.58    96.63     96.39     96.28     96.19
##  [6,]    99.63    99.56    99.54    99.58     99.27     99.13     99.02
##  [7,]   101.82   101.82   101.70   101.72    101.52    101.26    101.02
##  [8,]   101.10   101.14   100.99   100.94    100.79    100.49    100.19
##  [9,]   104.07   104.01   103.90   103.87    103.53    103.29    102.99
## [10,]   102.22   102.14   102.04   101.98    101.59    101.28    101.00
## [11,]    98.81    98.60    98.53    98.54     97.96     97.61     97.38
##       2025/1/15 2025/1/16 2025/1/17
##  [1,]     99.77     99.78     99.80
##  [2,]     98.40     98.47     98.47
##  [3,]     96.80     97.01     97.06
```

```
##  [4,]     96.85     97.03     97.07
##  [5,]     96.36     96.60     96.64
##  [6,]     99.20     99.50     99.53
##  [7,]    101.24    101.58    101.65
##  [8,]    100.42    100.92    101.02
##  [9,]    103.24    103.80    103.90
## [10,]    101.27    101.85    101.97
## [11,]     97.76     98.40     98.47
```

YTM

Firstly, we will calculate the YTM for each bonds mature in different date. It is important to know that the bonds paying semi-annually, so far we need consider them in each 0.5 time period.

```
# make an ytm matrix to collect the result
ytm_matrix <- matrix(NA, nrow = 11, ncol = 10)


freq <- 2
face_value <- 100

# loop
for (bond_order in 1:11){
  for ( t in 1:10) {
  bond_name <- bond_data$name[bond_order]
  maturity <- as.Date(bond_data$maturity_time[bond_order])
  coupon <- bond_data$coupon[bond_order]/2  # semi-annually payment
  close_price <- as.numeric(close_matrix[bond_order, t])
  day <- as.Date(trading_dates[t])
### According to the continuous discounting
    ytm_function <- function(x) {
      value_1 <- coupon * exp(-x * as.numeric(maturity - day) / 365) +
                face_value * exp(-x * as.numeric(maturity - day) / 365) -
                close_price

      if (bond_order >= 2) {
        intervals <- seq(-0.5, -5.0, by = -0.5)
        extra_terms <- intervals[1:(bond_order - 1)]
        value_1 <- value_1 + sum(coupon * exp(-x * (as.numeric(maturity - day) + extra_terms)))
      }
      return(value_1)
    }

    ytm_matrix[bond_order, t] <-
      uniroot(ytm_function, interval = c(-1, 1))$root
  }
}

print(ytm_matrix)
```

```
##              [,1]        [,2]        [,3]        [,4]        [,5]        [,6]
##  [1,] 0.060391472 0.061530620 0.062009824 0.063943021 0.064489694 0.069384065
##  [2,] 0.028555521 0.028519918 0.028797560 0.028920124 0.029518224 0.030061919
##  [3,] 0.027686943 0.027843172 0.028000160 0.027796308 0.028769807 0.028888032
##  [4,] 0.021394788 0.021555317 0.021528578 0.021376373 0.022480429 0.023162813
```

```
##  [5,] 0.018915013 0.019083794 0.019156473 0.018939317 0.020126338 0.020740040
##  [6,] 0.006597924 0.006860201 0.006940723 0.006801042 0.007960709 0.008515156
##  [7,] 0.002223948 0.002225995 0.002339670 0.002322492 0.002532089 0.002833714
##  [8,] 0.002496385 0.002450899 0.002615023 0.002730477 0.002920388 0.003437136
##  [9,] 0.001227754 0.001243625 0.001272906 0.001281728 0.001318402 0.001397983
## [10,] 0.001540941 0.001576173 0.001622291 0.001651676 0.001825384 0.002038629
## [11,] 0.004979532 0.005393988 0.005534564 0.005517819 0.006658190 0.007355552
##              [,7]        [,8]        [,9]       [,10]
##  [1,] 0.070892236 0.069214882 0.069956381 0.069881891
##  [2,] 0.030192650 0.029677996 0.028672750 0.028799081
##  [3,] 0.029966244 0.030039365 0.028180320 0.027788821
##  [4,] 0.023582167 0.022733083 0.021628692 0.021411152
##  [5,] 0.021206366 0.020402644 0.019256838 0.019086430
##  [6,] 0.008942769 0.008267099 0.007147127 0.007043008
##  [7,] 0.003195390 0.002858931 0.002480159 0.002409184
##  [8,] 0.004074903 0.003574257 0.002767434 0.002604068
##  [9,] 0.001549946 0.001416958 0.001306844 0.001280713
## [10,] 0.002303832 0.002046562 0.001724042 0.001664442
## [11,] 0.007813337 0.007068239 0.005815467 0.005679929
```

turn the matrix into data frame

```r
ytmdataframe <- as.data.frame(ytm_matrix)
colnames(ytmdataframe) <- format(trading_dates)
rownames(ytmdataframe) <- format(bond_data$name)
print(ytmdataframe)
```

```
##             2025-01-06  2025-01-07  2025-01-08  2025-01-09  2025-01-10
## march_2025 0.060391472 0.061530620 0.062009824 0.063943021 0.064489694
## sept_2025  0.028555521 0.028519918 0.028797560 0.028920124 0.029518224
## march_2026 0.027686943 0.027843172 0.028000160 0.027796308 0.028769807
## sept_2026  0.021394788 0.021555317 0.021528578 0.021376373 0.022480429
## march_2027 0.018915013 0.019083794 0.019156473 0.018939317 0.020126338
## sept_2027  0.006597924 0.006860201 0.006940723 0.006801042 0.007960709
## march_2028 0.002223948 0.002225995 0.002339670 0.002322492 0.002532089
## sept_2028  0.002496385 0.002450899 0.002615023 0.002730477 0.002920388
## march_2029 0.001227754 0.001243625 0.001272906 0.001281728 0.001318402
## sept_2029  0.001540941 0.001576173 0.001622291 0.001651676 0.001825384
## march_2030 0.004979532 0.005393988 0.005534564 0.005517819 0.006658190
##             2025-01-13  2025-01-14  2025-01-15  2025-01-16  2025-01-17
## march_2025 0.069384065 0.070892236 0.069214882 0.069956381 0.069881891
## sept_2025  0.030061919 0.030192650 0.029677996 0.028672750 0.028799081
## march_2026 0.028888032 0.029966244 0.030039365 0.028180320 0.027788821
## sept_2026  0.023162813 0.023582167 0.022733083 0.021628692 0.021411152
## march_2027 0.020740040 0.021206366 0.020402644 0.019256838 0.019086430
## sept_2027  0.008515156 0.008942769 0.008267099 0.007147127 0.007043008
## march_2028 0.002833714 0.003195390 0.002858931 0.002480159 0.002409184
## sept_2028  0.003437136 0.004074903 0.003574257 0.002767434 0.002604068
## march_2029 0.001397983 0.001549946 0.001416958 0.001306844 0.001280713
## sept_2029  0.002038629 0.002303832 0.002046562 0.001724042 0.001664442
## march_2030 0.007355552 0.007813337 0.007068239 0.005815467 0.005679929
```

YTM Plot

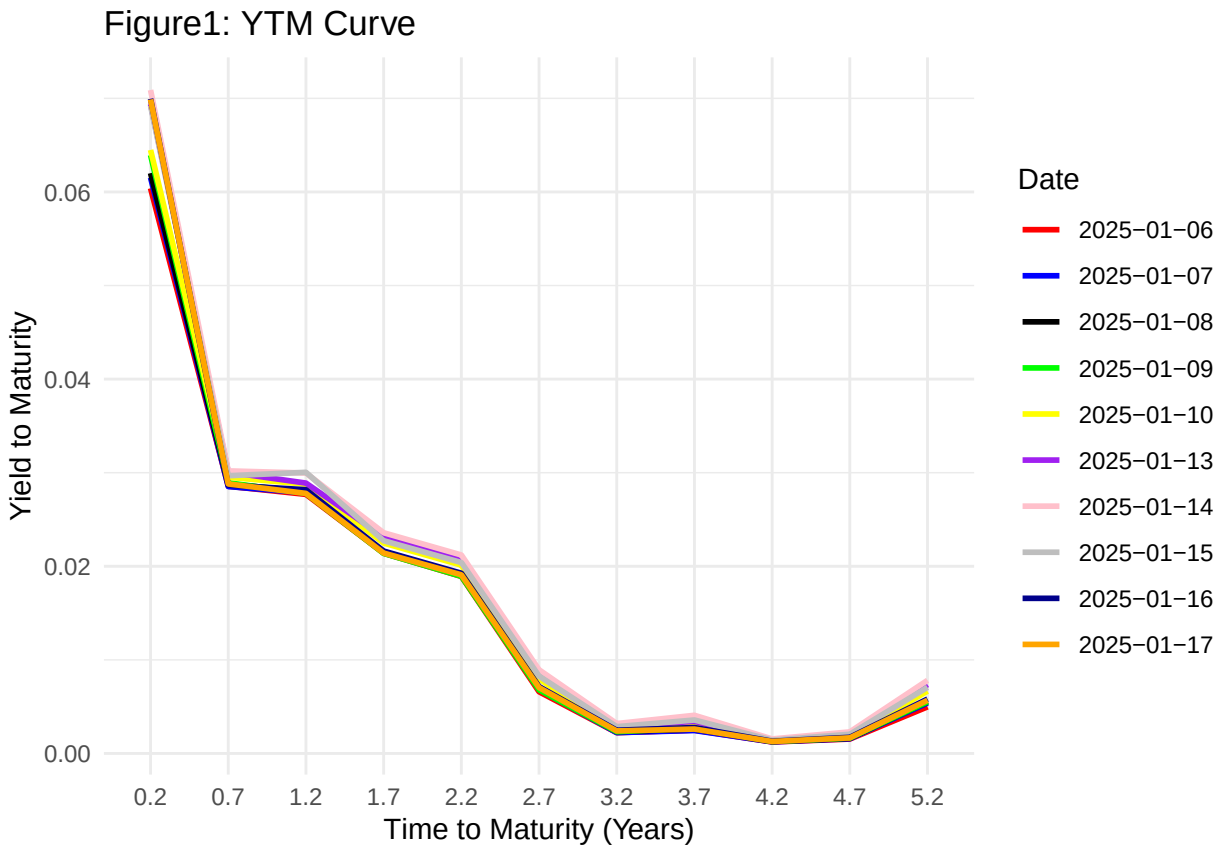
```
library(ggplot2)
library(readr)

ytmdataframe$mature <- c("0.2","0.7","1.2","1.7","2.2","2.7","3.2","3.7","4.2","4.7","5.2")
## randomly select

color_ytm = c("2025-01-06" = "red", "2025-01-07" = "blue",
              "2025-01-08" = "black", "2025-01-09" = "green",
              "2025-01-10" = "yellow", "2025-01-13" = "purple",
              "2025-01-14" = "pink", "2025-01-15" = "grey",
              "2025-01-16" = "darkblue","2025-01-17" = "orange")

ggplot(ytmdataframe, aes(x = mature)) +
    geom_line(aes(y = `2025-01-06`, color = "2025-01-06", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-07`, color = "2025-01-07", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-08`, color = "2025-01-08", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-09`, color = "2025-01-09", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-10`, color = "2025-01-10", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-13`, color = "2025-01-13", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-14`, color = "2025-01-14", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-15`, color = "2025-01-15", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-16`, color = "2025-01-16", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-17`, color = "2025-01-17", group = 1), size = 1) +
    labs(x = "Time to Maturity (Years)",
         y = "Yield to Maturity",
         title = "Figure1: YTM Curve",
         color = "Date") +
    scale_color_manual(values = color_ytm) +
    theme_minimal()
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Spot Rate

```
### we will plan to calculate different spot rate according to the different trading date.


spot_matrix <- matrix(NA, nrow = 11, ncol = 10)
face_value <- 100

# Bootstrapping
for (t in 1:10) {
  maturity <- as.Date(bond_data$maturity_time)
  day <- as.Date(trading_dates[t])
  Mt <- as.numeric(maturity - day) / 365
  price <- as.numeric(close_matrix[, t])
  coupon <- bond_data$coupon / 2
  spot_matrix[1, t] <- ((coupon[1]+face_value) / price[1]) - 1

  for (bond_order in 2:11) {

    pv <- sum(coupon[1:(bond_order - 1)] /
                ((1 + spot_matrix[1:(bond_order - 1), t])^(1:(bond_order - 1))))

    ## function define
    spot_function <- function(x) {
      pv + (face_value + coupon[bond_order]) / ((1 + x)^Mt[bond_order]) - price[bond_order]
    }
```

```
spot_matrix[bond_order, t] <- uniroot(spot_function, interval = c(0, 1))$root
  }
}


print(spot_matrix)
```

```
##             [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
##  [1,] 0.00897423 0.00897423 0.00887307 0.00897423 0.00887307 0.00897423
##  [2,] 0.03899067 0.03899473 0.03932853 0.03949813 0.04016898 0.04086832
##  [3,] 0.03602069 0.03620248 0.03638438 0.03618843 0.03722361 0.03740045
##  [4,] 0.02780187 0.02797883 0.02795983 0.02781042 0.02896940 0.02970988
##  [5,] 0.02609484 0.02627750 0.02636130 0.02614452 0.02737796 0.02803586
##  [6,] 0.01416075 0.01444770 0.01454031 0.01440060 0.01562002 0.01621354
##  [7,] 0.01005712 0.01005611 0.01045397 0.01040354 0.01102015 0.01188200
##  [8,] 0.01498715 0.01487317 0.01530631 0.01547122 0.01585883 0.01672114
##  [9,] 0.01027727 0.01043037 0.01068754 0.01077146 0.01151027 0.01207448
## [10,] 0.01694052 0.01712681 0.01733579 0.01748402 0.01831746 0.01899447
## [11,] 0.02513376 0.02560120 0.02573228 0.02571704 0.02692332 0.02765642
##             [,7]       [,8]        [,9]       [,10]
##  [1,] 0.00897423 0.00856971 0.008468631 0.008266533
##  [2,] 0.04104960 0.04055793 0.039548875 0.039728636
##  [3,] 0.03854835 0.03864866 0.036725628 0.036335740
##  [4,] 0.03015564 0.02928060 0.028144708 0.027928991
##  [5,] 0.02852531 0.02770487 0.026534205 0.026368137
##  [6,] 0.01666042 0.01597416 0.014818745 0.014718766
##  [7,] 0.01266956 0.01198041 0.010914290 0.010698403
##  [8,] 0.01757040 0.01695825 0.015582114 0.015311835
##  [9,] 0.01278750 0.01221725 0.010969499 0.010741859
## [10,] 0.01958372 0.01903449 0.017827081 0.017581830
## [11,] 0.02809525 0.02733833 0.026085250 0.025974157
```

turn the matrix into data frame

```
spotdataframe <- as.data.frame(spot_matrix)
colnames(spotdataframe) <- format(trading_dates)
rownames(spotdataframe) <- format(bond_data$name)
print(spotdataframe)
```

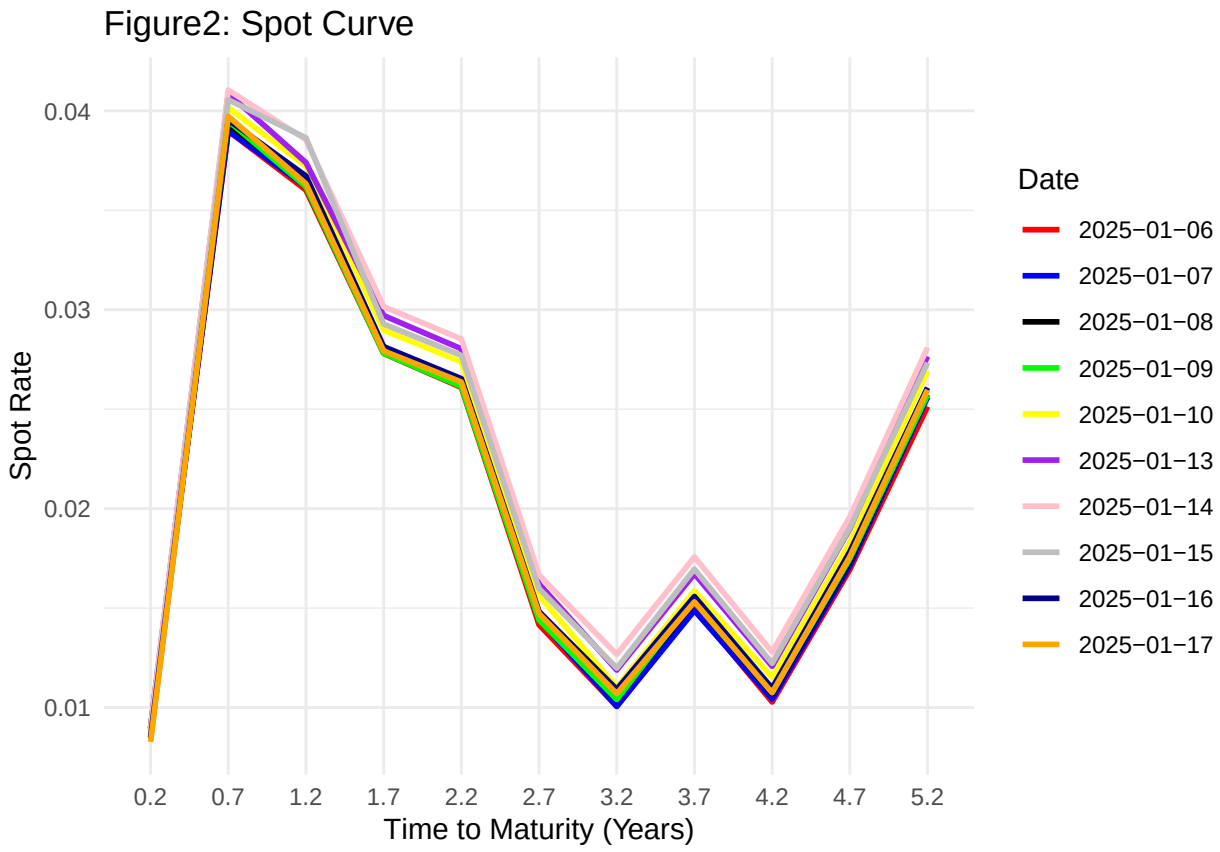
```
##            2025-01-06 2025-01-07 2025-01-08 2025-01-09 2025-01-10 2025-01-13
## march_2025 0.00897423 0.00897423 0.00887307 0.00897423 0.00887307 0.00897423
## sept_2025  0.03899067 0.03899473 0.03932853 0.03949813 0.04016898 0.04086832
## march_2026 0.03602069 0.03620248 0.03638438 0.03618843 0.03722361 0.03740045
## sept_2026  0.02780187 0.02797883 0.02795983 0.02781042 0.02896940 0.02970988
## march_2027 0.02609484 0.02627750 0.02636130 0.02614452 0.02737796 0.02803586
## sept_2027  0.01416075 0.01444770 0.01454031 0.01440060 0.01562002 0.01621354
## march_2028 0.01005712 0.01005611 0.01045397 0.01040354 0.01102015 0.01188200
## sept_2028  0.01498715 0.01487317 0.01530631 0.01547122 0.01585883 0.01672114
## march_2029 0.01027727 0.01043037 0.01068754 0.01077146 0.01151027 0.01207448
## sept_2029  0.01694052 0.01712681 0.01733579 0.01748402 0.01831746 0.01899447
## march_2030 0.02513376 0.02560120 0.02573228 0.02571704 0.02692332 0.02765642
##            2025-01-14 2025-01-15  2025-01-16  2025-01-17
## march_2025 0.00897423 0.00856971 0.008468631 0.008266533
## sept_2025  0.04104960 0.04055793 0.039548875 0.039728636
```

```
## march_2026 0.03854835 0.03864866 0.036725628 0.036335740
## sept_2026  0.03015564 0.02928060 0.028144708 0.027928991
## march_2027 0.02852531 0.02770487 0.026534205 0.026368137
## sept_2027  0.01666042 0.01597416 0.014818745 0.014718766
## march_2028 0.01266956 0.01198041 0.010914290 0.010698403
## sept_2028  0.01757040 0.01695825 0.015582114 0.015311835
## march_2029 0.01278750 0.01221725 0.010969499 0.010741859
## sept_2029  0.01958372 0.01903449 0.017827081 0.017581830
## march_2030 0.02809525 0.02733833 0.026085250 0.025974157
```

```
spotdataframe$mature <- c("0.2","0.7","1.2","1.7","2.2","2.7","3.2","3.7","4.2","4.7","5.2")
## randomly select

color_ytm = c("2025-01-06" = "red", "2025-01-07" = "blue",
              "2025-01-08" = "black", "2025-01-09" = "green",
              "2025-01-10" = "yellow", "2025-01-13" = "purple",
              "2025-01-14" = "pink", "2025-01-15" = "grey",
              "2025-01-16" = "darkblue","2025-01-17" = "orange")

ggplot(spotdataframe, aes(x = mature)) +
    geom_line(aes(y = `2025-01-06`, color = "2025-01-06", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-07`, color = "2025-01-07", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-08`, color = "2025-01-08", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-09`, color = "2025-01-09", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-10`, color = "2025-01-10", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-13`, color = "2025-01-13", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-14`, color = "2025-01-14", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-15`, color = "2025-01-15", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-16`, color = "2025-01-16", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-17`, color = "2025-01-17", group = 1), size = 1) +
    labs(x = "Time to Maturity (Years)",
         y = "Spot Rate",
         title = "Figure2: Spot Curve",
         color = "Date") +
    scale_color_manual(values = color_ytm) +
    theme_minimal()
```



Forward Rate

```
forward_matrix <- matrix(NA, nrow = 4, ncol = 10)
for (t in 1:10) {
  S1 <- spot_matrix[1, t]
  for ( a in 2:5) {
    ST <- spot_matrix[a , t]
    forward_matrix[a-1, t] <- ((1 + ST)^a / (1 + S1)^1)^(1 / (a - 1)) - 1
  }
}
forward_matrix
```

```
##            [,1]       [,2]       [,3]       [,4]       [,5]       [,6]
## [1,] 0.06990009 0.06990844 0.07070338 0.07094544 0.07243571 0.07377059
## [2,] 0.04981459 0.05009092 0.05042011 0.05006956 0.05169625 0.05191248
## [3,] 0.03415550 0.03439291 0.03440200 0.03416697 0.03575675 0.03671603
## [4,] 0.03042019 0.03064948 0.03078052 0.03048255 0.03205698 0.03285727
##            [,7]       [,8]       [,9]      [,10]
## [1,] 0.07414465 0.07356069 0.07158699 0.07217249
## [2,] 0.05365891 0.05402288 0.05114966 0.05066203
## [3,] 0.03731446 0.03627831 0.03478834 0.03456799
## [4,] 0.03347198 0.03254512 0.03110094 0.03094409
```

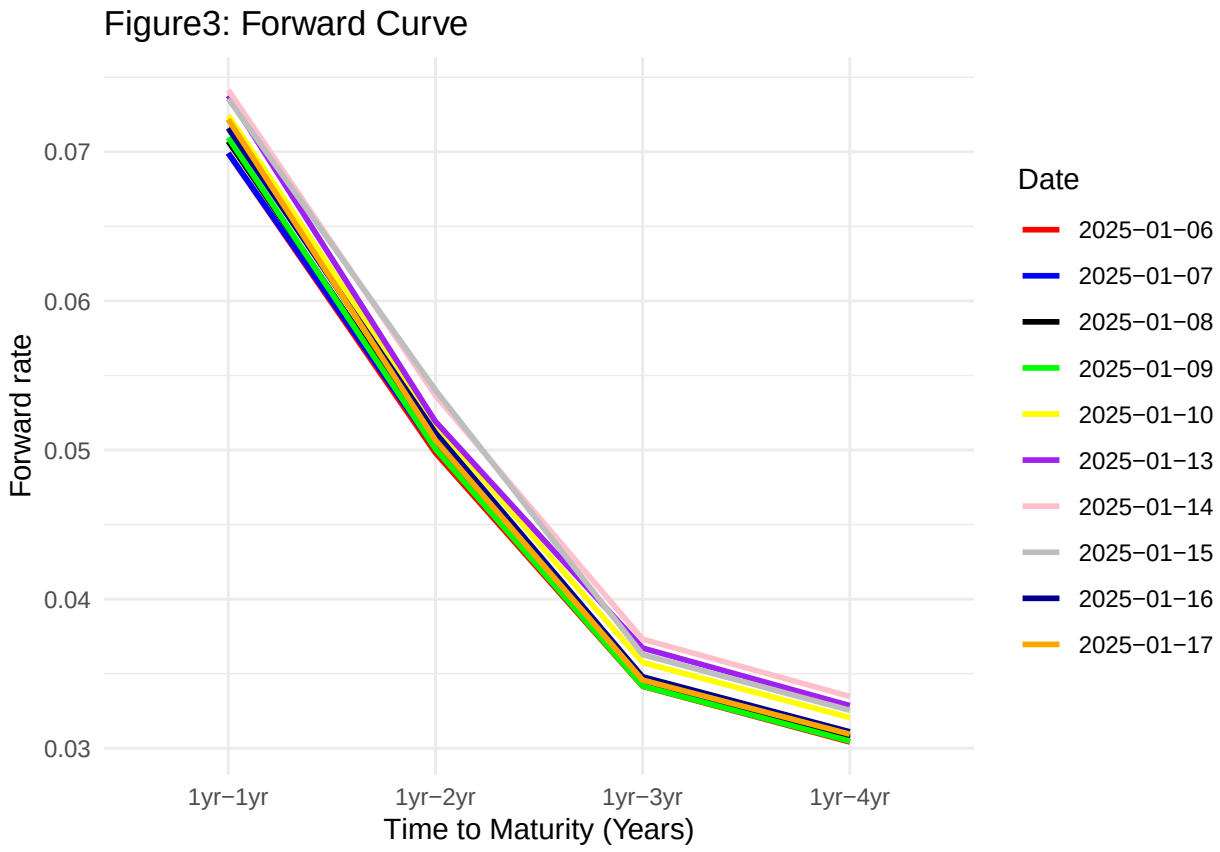
```
forwarddataframe <- as.data.frame(forward_matrix)
colnames(forwarddataframe) <- format(trading_dates)
rownames(forwarddataframe) <- format(c("1yr-1yr", "1yr-2yr", "1yr-3yr", "1yr-4yr"))
print(forwarddataframe)
```

```
##         2025-01-06 2025-01-07 2025-01-08 2025-01-09 2025-01-10 2025-01-13
## 1yr-1yr 0.06990009 0.06990844 0.07070338 0.07094544 0.07243571 0.07377059
## 1yr-2yr 0.04981459 0.05009092 0.05042011 0.05006956 0.05169625 0.05191248
## 1yr-3yr 0.03415550 0.03439291 0.03440200 0.03416697 0.03575675 0.03671603
## 1yr-4yr 0.03042019 0.03064948 0.03078052 0.03048255 0.03205698 0.03285727
##         2025-01-14 2025-01-15 2025-01-16 2025-01-17
## 1yr-1yr 0.07414465 0.07356069 0.07158699 0.07217249
## 1yr-2yr 0.05365891 0.05402288 0.05114966 0.05066203
## 1yr-3yr 0.03731446 0.03627831 0.03478834 0.03456799
## 1yr-4yr 0.03347198 0.03254512 0.03110094 0.03094409
```

```r
forwarddataframe$mature <- c("1yr-1yr", "1yr-2yr", "1yr-3yr", "1yr-4yr")

color_ytm = c("2025-01-06" = "red", "2025-01-07" = "blue",
              "2025-01-08" = "black", "2025-01-09" = "green",
              "2025-01-10" = "yellow", "2025-01-13" = "purple",
              "2025-01-14" = "pink", "2025-01-15" = "grey",
              "2025-01-16" = "darkblue","2025-01-17" = "orange")

ggplot(forwarddataframe, aes(x = mature)) +
    geom_line(aes(y = `2025-01-06`, color = "2025-01-06", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-07`, color = "2025-01-07", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-08`, color = "2025-01-08", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-09`, color = "2025-01-09", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-10`, color = "2025-01-10", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-13`, color = "2025-01-13", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-14`, color = "2025-01-14", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-15`, color = "2025-01-15", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-16`, color = "2025-01-16", group = 1), size = 1) +
    geom_line(aes(y = `2025-01-17`, color = "2025-01-17", group = 1), size = 1) +
    labs(x = "Time to Maturity (Years)",
         y = "Forward rate",
         title = "Figure3: Forward Curve",
         color = "Date") +
    scale_color_manual(values = color_ytm) +
    theme_minimal()
```



5 & 6

```
### YTM
yieldreturn <- matrix(NA,ncol = 5, nrow = 9)
for (j in 1:9) {
  for (i in 1:5) {
  yieldreturn[j,i]=log(ytm_matrix[i,j+1]/ytm_matrix[i,j])
  }
}
yieldreturn
```

```
##               [,1]         [,2]         [,3]         [,4]         [,5]
##  [1,]  0.018687041 -0.001247575  0.005626832  0.007475182  0.008883589
##  [2,]  0.007757878  0.009687918  0.005622473 -0.001241284  0.003801150
##  [3,]  0.030699564  0.004247041 -0.007307037 -0.007094984 -0.011400622
##  [4,]  0.008513036  0.020470147  0.034423279  0.050358869  0.060789289
##  [5,]  0.073151815  0.018251389  0.004100931  0.029903000  0.030036795
##  [6,]  0.021503685  0.004339279  0.036644155  0.017942659  0.022235284
##  [7,] -0.023945032 -0.017192614  0.002437113 -0.036669510 -0.038636901
##  [8,]  0.010656033 -0.034458680 -0.063884816 -0.049800494 -0.057798311
##  [9,] -0.001065382  0.004396278 -0.013990038 -0.010108876 -0.008888556
```

```
cov_yieldreturn = round(cov(yieldreturn, yieldreturn),7)
cov_yieldreturn
```

```
##           [,1]      [,2]      [,3]      [,4]      [,5]
## [1,] 0.0006949 0.0002109 0.0000652 0.0004075 0.0004014
```

```
## [2,] 0.0002109 0.0002973 0.0003812 0.0004880 0.0005593
## [3,] 0.0000652 0.0003812 0.0008678 0.0007292 0.0008630
## [4,] 0.0004075 0.0004880 0.0007292 0.0009733 0.0011107
## [5,] 0.0004014 0.0005593 0.0008630 0.0011107 0.0012773
```

```
eigen_yieldreturn <- eigen(cov_yieldreturn)
round(eigen_yieldreturn$values,7)
```

```
## [1] 0.0032532 0.0006826 0.0001282 0.0000458 0.0000008
```

```
round(eigen_yieldreturn$vectors, 7)
```

```
##            [,1]       [,2]       [,3]       [,4]       [,5]
## [1,] -0.2181038  0.8694828 -0.4319438  0.0588530  0.0799450
## [2,] -0.2794824  0.0319856  0.0687665 -0.9565237 -0.0346435
## [3,] -0.4410795 -0.4877741 -0.7510639  0.0585494  0.0005870
## [4,] -0.5423887  0.0645736  0.2922990  0.2090555 -0.7566385
## [5,] -0.6209538 -0.0297184  0.3989488  0.1856518  0.6480019
```

```
### Forward
forwardreturn <- matrix(NA, ncol = 4, nrow = 9)
for (j in 1:9) {
  for (i in 1:4) {
  forwardreturn[j,i]=log(forward_matrix[i,j+1]/forward_matrix[i,j])
  }
}
forwardreturn
```

```
##                [,1]         [,2]          [,3]         [,4]
##  [1,]  0.0001194655  0.005531878  0.0069266529  0.007509201
##  [2,]  0.0113069694  0.006550304  0.0002641858  0.004266469
##  [3,]  0.0034178312 -0.006976815 -0.0068551796 -0.009727722
##  [4,]  0.0207881795  0.031971933  0.0454795517  0.050360412
##  [5,]  0.0182607459  0.004174144  0.0264744583  0.024658135
##  [6,]  0.0050578212  0.033088256  0.0161676069  0.018535810
##  [7,] -0.0079070795  0.006760125 -0.0281611168 -0.028081437
##  [8,] -0.0271975083 -0.054651791 -0.0419375676 -0.045389312
##  [9,]  0.0081455694 -0.009579151 -0.0063543447 -0.005056024
```

```
cov_forwardreturn = round(cov(forwardreturn, forwardreturn),7)
cov_forwardreturn
```

```
##           [,1]      [,2]      [,3]      [,4]
## [1,] 0.0002107 0.0002818 0.0003450 0.0003702
## [2,] 0.0002818 0.0006687 0.0005409 0.0005964
## [3,] 0.0003450 0.0005409 0.0007127 0.0007583
## [4,] 0.0003702 0.0005964 0.0007583 0.0008124
```

```
eigen_forwardreturn <- eigen(cov_forwardreturn)
round(eigen_forwardreturn$values,7)
```

```
## [1] 0.0021773 0.0001874 0.0000379 0.0000019
```

```
round(eigen_forwardreturn$vectors, 7)
```

```
##            [,1]       [,2]       [,3]       [,4]
## [1,] -0.2824072 -0.1197813  0.9517124 -0.0119242
## [2,] -0.4921884  0.8680934 -0.0374513 -0.0525533
## [3,] -0.5605762 -0.3701889 -0.2217899 -0.7067699
```

```
## [4,] -0.6031177 -0.3082637 -0.2089264  0.7053880
```