Problem Set 3

Ryan Shea

10/20/2022

I pledge my honor that I have abided by the Stevens Honor System.

1: Interest Rate Derivatives

1.1

(a)

```r
ask <- 91 + (12.5/32)
coup_semi <- 0.02875 / 2
yield <- 0.03963
int_r_6mo <- yield / 2

previous_coup <- as.Date('2022-05-15')
today <- as.Date('2022-10-13')
next_coup <- as.Date('2022-11-15')
maturity <- as.Date('2032-05-15')

N <- round(2 * (length(integer(maturity - next_coup)) / 365))

pv_coup <- 100 * (coup_semi / int_r_6mo) * (1 - (1 / (1 + int_r_6mo)^N))

pv_ttm <- 100 / (1 + int_r_6mo)^N

ex_coup_cash <- pv_coup + pv_ttm

coup_cash <- ex_coup_cash + coup_semi


days_curr_per <- length(integer(next_coup - today))
period <- length(integer(next_coup - previous_coup))

cash_price <- coup_cash / (1 + int_r_6mo)^(days_curr_per/period)
cash_price
```

```
## [1] 91.1494
```

(b)

```r
cash_price_corp <- coup_cash / (1 + int_r_6mo)^((days_curr_per+1)/period)
cash_price_corp
```

```
## [1] 91.13968
```

1.2

```r
px <- 4.4
basis <- 90/365


cash_px <- 100 - basis * px

(365 / 90) * log(100 / cash_px)
```

```
## [1] 0.04424043
```

1.3

Coupon	4.5
Maturity Date	2038-05-15
CUSIP	912810PX0
Delivery Month 1	December 2022
Conversion Factor 1	0.8514
Delivery Month 2	March 2023
Conversion Factor 2	0.8530

```r
coupon <- 0.045
coup_semi <- coupon / 2
yield <- 0.06
yield_semi <- yield / 2
n <- 30

face <- (coup_semi * 100 / yield_semi) * (1 - (1 / (1 + yield_semi)^n)) + 100 / (1+yield_semi)^n

face / 100 # conversion factor
```

```
## [1] 0.8529967
```

```r
face_other <- (face + (coup_semi * 100)) / (1+yield_semi)^0.5
(face_other - (coup_semi * 100 / 2)) / 100 # Divide semi by 2 for quarterly
```

```
## [1] 0.8514025
```

1.4

Bond	Quoted_Price	Conversion_Factor
1	91.59375	0.8913
2	97.87500	0.9535
3	107.31250	1.0441
4	116.56250	1.1349

```
# df is the given table above
F_0 <- 102 + (11/32)
df$Cheapest_to_Deliver <- df$Quoted_Price - (F_0 * df$Conversion_Factor)
kable(df)
```

Bond	Quoted_Price	Conversion_Factor	Cheapest_to_Deliver
1	91.59375	0.8913	0.3747656
2	97.87500	0.9535	0.2902344
3	107.31250	1.0441	0.4553906
4	116.56250	1.1349	0.4125781

The cheapest to deliver is Bond 2.

1.5

```r
today <- as.Date('2022-10-14')
delivery <- as.Date('2023-04-30')

prev_coup <- as.Date('2022-09-01')
next_coup <- as.Date('2023-03-01')

days_until_coup <- round(length(integer(next_coup - today))) # 138

days_until_delivery <- round(length(integer(delivery - today))) # 198

previous_coup_days <- round(length(integer(today - prev_coup))) # 43

days_between_coup <- round(length(integer(next_coup - prev_coup))) # 181

coup <- 0.03
coup_semi <- coup / 2
r <- 0.04

S_0 <- 85.31 + (previous_coup_days / days_between_coup) * 100 * coup_semi

S_0
```

```
## [1] 85.66635
```

```r
# Only one coupon received during time
pv_coup <- (100 * coup_semi) * exp(-r * (days_until_coup/365))
pv_coup
```

```
## [1] 1.477486
```

```r
F_0 <- (S_0 - pv_coup) * exp(r * (days_until_delivery / 365))
F_0
```

```
## [1] 86.03561
```

```r
coup_future <- as.Date('2023-09-01')

sep_to_mar <- round(length(integer(coup_future - next_coup))) # 184

accr_int <- round(length(integer(delivery - next_coup))) # 60

quoted_F <- F_0 - (100 * coup_semi) * (accr_int / sep_to_mar)

# Divide by conversion factor
quoted_F / 0.6403
```

```
## [1] 133.6038
```

1.6

(a)

$$S_0 * e^{0.05*5} - S_0 * e^{(0.05-0.0001)*5} = 0.1$$

$$\implies S_0 = 155.80$$

$$155.80 * e^{0.05*5} \approx 200.05$$

(b)

$$155.80 * e^{0.0499*5} - S_0 * e^{0.0499*10} = 0.1$$
$$\implies S_0 \approx 121.34$$

(c)

It would most likely match **Plot 3** because the portfolio is the most optimal when as close to 5% as possible. It will begin to decrease slightly when the rates diverge away from this optimal number.

1.7

(a)

```
port_0 <- 38000000
port_duration <- 11.2 # In 4 months

F_0 <- 113
contract <- 100000
F_duration <- 17.5

N_star <- (port_0 * port_duration) / ((F_0 * contract / 100) * F_duration)

round(N_star)
```

```
## [1] 215
```

The optimal number of positions is to be **short 215 positions**.

(b)

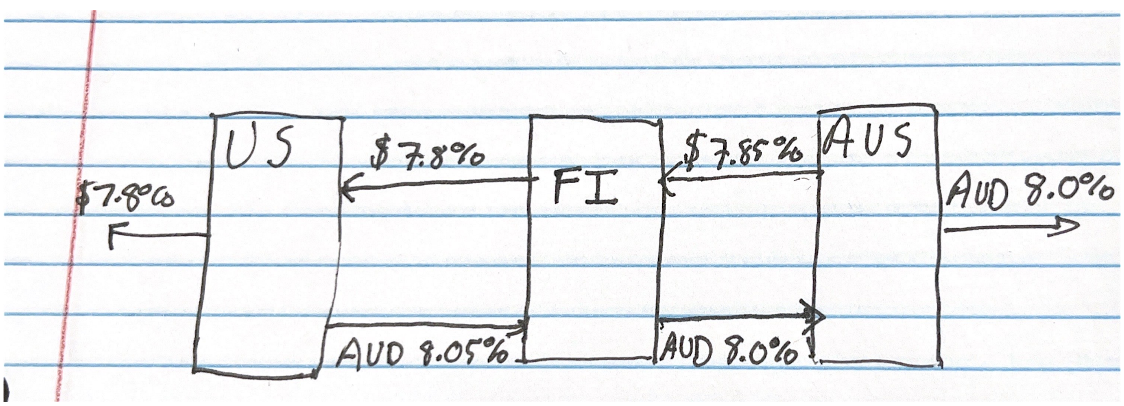
If all rates increase over the three months, but longer-term rates increase less than shorter-term rates, then there will be a gain on the short position but a loss on the actual portfolio. This is because the duration for the cheapest to deliver bonds is higher the gain on the bond portfolio.

2: Swaps

2.1

	US_Dollars	AUD
US_Company	0.078	0.084
Australian_Company	0.082	0.080

Solution on next page


US
$7.8%
FI
$7.85%
AUS
$7.8%
AUD 8.0%
AUD 8.05%
AUD 8.0%

2.2

```r
# pay <- 3mo LIBOR
receive <- 0.07
principal <- 50000000
# Payments every 3 months
life <- 11/12

libor_3mo <- 0.066
libor_3mo_back2mo <- 0.068
ois_r <- 0.06

val_swap <-
  principal * (receive - libor_3mo_back2mo) * 0.25 * exp(-ois_r * (1/12)) +
  principal * (receive - libor_3mo) * 0.25 * exp(-ois_r * (4/12)) +
  principal * (receive - libor_3mo) * 0.25 * exp(-ois_r * (7/12)) +
  principal * (receive - libor_3mo) * 0.25 * exp(-ois_r * (10/12))

val_swap
```

```
## [1] 169727
```

2.3

$$0 = 0.001e^{-0.03*1} + 0.001e^{-0.031*2} + (0.041 - r)e^{-0.032*3}$$
$$\implies r \approx 0.0431$$

2.4

```r
fixed <- 0.052
principal <- 20000000
# 6 Cash flows
swap_rate <- 0.05
ois <- 0.04

val <- 0.5 * (fixed - swap_rate) * principal
total <- 0
for (i in seq(0.5, 3, 0.5)){
  total <- total + val * exp(-ois * i)
}

total
```

```
## [1] 111952.5
```

2.5

```r
r_usd <- 0.051
r_gbp <- 0.048
er <- 1.12 # Exchange rate

s_usd <- 0.06 # Pays
s_gbp <- 0.06 # Receives

pr_usd <- 12000000
pr_gbp <- 10000000

V_usd <- pr_usd * s_usd * exp(-r_usd * 1) + pr_usd * (1 + s_usd) * exp(-r_usd * 2)
V_gbp <- pr_gbp * s_gbp * exp(-r_gbp * 1) + pr_gbp * (1 + s_gbp) * exp(-r_gbp * 2)

V_usd - V_gbp * 1.12
```

```
## [1] 744945.8
```