Intro to Financial Engineering IEOR W4700

Homework 10

Linan Qiu
lq2137

December 6, 2015

Problem 1.

$$e^{-y_2(T_2-t)} = e^{-y_1(T_1-t)}e^{-y_{1,2}(T_2-T_1)}$$

Then,

$$-y_2(T_2 - t) = -y_1(T_1 - t) - y_{1,2}(T_2 - T_1)$$

Or setting the subject to be the forward rate $y_{1,2}$,

$$y_{1,2} = \frac{-y_1(T_1 - t) + y_2(T_2 - t)}{T_2 - T_1}$$

This function in `R`:

```
forward_rate = function(T1, T2, t, y1, y2) {
  return ((-y1*(T1 - t) + y2*(T2-t))/(T2-T1))
}
```

Then, we recognize that for the problem, $t = 0$ for all calculations, while T_2 and T_1 always have a difference of $\frac{1}{4}$ since we are advancing in quarters. Then we can do a shortcut as such:

```
maturities = c(3, 6, 9, 12, 15, 18)
maturities = maturities / 12
spot_rates = c(7, 7.2, 7.3, 7.5, 7.7, 7.9)
spot_rates = spot_rates / 100

quarterly_forward_rate = function(period) {
```

```
  T2 = maturities[period]
  T1 = maturities[period - 1]

  y2 = spot_rates[period]
  y1 = spot_rates[period - 1]

  t = 0

  return(forward_rate(T1, T2, t, y1, y2))
}

forward_rates = sapply(c(2:6), quarterly_forward_rate)
```

To get the forward rates, we do

```
> forward_rates
[1] 0.074 0.075 0.081 0.085 0.089
```

Problem 2.

Value of \$1 FRA can be replicated by longing $Z(T_1)$ and shorting $(1+K(T_2-T_1))Z(T_2)$. The value of the FRA in the question is then

$$N\left[Z(T_1) - (1 + K(T_2 - T_1))Z(T_2)\right]$$

Now, using the spots given,

$$Z(T_1) = Z(12/12) = e^{-0.085*1}$$

$$Z(T_2) = Z(15/12) = e^{-0.086*1.25}$$

$$K = 0.095$$

Then, the value of the FRA is

$$2000000 * \left[e^{-0.085} - (1 + 0.095(3/12))e^{-0.086}\right] = -1787.111$$

The FRA is worth \$-1787.111.

To verify this answer, we can use the forward rates.

Using the same method as in the previous section, we find the quarterly forward rates as follows:

```
> spot_rates = c(8, 8.2, 8.4, 8.5, 8.6, 8.7)
> forward_rates = sapply(c(2:6), quarterly_forward_rate)
> forward_rates
[1] 0.084 0.088 0.088 0.090 0.092
```

Then the annual forward rate for one quarter starting in 1 year is 0.090.

We can find the expiration value of the fixed leg: 0.095/4. We can also find the expiration value of the floating leg: $e^{(0.09*0.25)} - 1$. Then, by finding the present value of that difference and multiplying it by the notional, we arrive at the same answer.

```
> fixedleg = 0.095/4
> floatleg = exp(0.09*0.25) - 1
> difference = floatleg - fixedleg
> difference_pv = difference * exp(-0.086*5/4)
> fra = difference_pv*2000000
> fra
[1] -1787.111
```

Problem 3.

```
time_month = c(0, 6, 12, 18, 24, 30)
time = time_month / 12
time_payment_month = time_month + 6
time_payment = time_payment_month / 12

six_m_libor = c(5, 5.8, 5.3, 5.5, 5.6, 6.1)
six_m_libor = six_m_libor / 100
fixed_rate = 0.1
fixed_rate = c(rep(fixed_rate, 6))

fixed_payments = (fixed_rate / 2) * 10000000
floating_payments = (six_m_libor/ 2) * 10000000
net_payments = fixed_payments - floating_payments

table = data.frame(time_payment_month, fixed_rate , six_m_libor, fixed_
    payments, floating_payments, net_payments)
colnames(table) = c('Time', 'Fixed Rate', '6 Month Libor Rate', 'Fixed
    Payments', 'Floating Payments', 'Net Payments')
library(xtable)
xtable(table)
```

Table generated is reproduced on the next page.

	Time	Fixed Rate	6 Month Libor Rate	Fixed Payments	Floating Payments	Net Payments
1	6	0.10	0.05	500000.00	250000.00	250000.00
2	12	0.10	0.06	500000.00	290000.00	210000.00
3	18	0.10	0.05	500000.00	265000.00	235000.00
4	24	0.10	0.06	500000.00	275000.00	225000.00
5	30	0.10	0.06	500000.00	280000.00	220000.00
6	36	0.10	0.06	500000.00	305000.00	195000.00

Table 1: Payment stream of IRS

Problem 4.

Problem 4(a).

Current value of the IRS is the PV of the net payments.

```
> value = sum(net_payments * discount_factors)
> value
[1] 1225252
```

Problem 4(b).

Current forward swap rate is the fixed rate that makes the swap value zero ie. the PV of the fixed leg equal to the PV of the floating leg.

```
discount_factors = c(0.9778, 0.9541, 0.9291, 0.9048, 0.8781, 0.8479)

gradient_descent = function(k) {
  fixed_rate = c(rep(k, 6))
  fixed_payments = (fixed_rate / 2)
  floating_payments = (six_m_libor/2)
  pv_fixed = fixed_payments * discount_factors
  pv_float = floating_payments * discount_factors
  return(sum(pv_fixed) - sum(pv_float))
}
```

Using this function, we can calculate the fixed rate that makes this function 0.

```
> uniroot(gradient_descent, c(0, 0.1))
$root
[1] 0.05537886

$f.root
[1] 0

$iter
[1] 1

$init.it
[1] NA

$estim.prec
[1] 0.05537886
```

$K = 0.05537886$

The closed form solution for this can be found by making K the subject of this equation, where V_{swap} is the value of the swap found earlier by using $K = 0.1$:

$$V_{\text{swap}} = 1 - \left(Z(t, T_6) + \frac{K}{2} \sum_{i=1}^{n} Z(t, T_i) \right)$$

However, it was just easier to calculate it directly (though computationally inefficient).