

1.1

1)

spot price yields $5050 \times \exp(0.035 \times 1) = 5229$ at maturity.

Yes, there's an arbitrage opportunity. Borrow \$5050 from the money market, buy 1 ounce of gold by entering a long position of the contract at \$5200. At maturity, we will receive a profit of $5229 - 5200 = \$29$

2)

Yes, there's an arbitrage opportunity. Borrow \$5050 from the money market and buy 1 ounce of gold at spot price. Then enter a short position of the contract at \$5300. At maturity, we will receive a profit of $5300 - 5229 = \$71$

1.2

1)

$$R_2 = m_2 * [(1 + R_1/m_1)^{(m_1/m_2)} - 1]$$

If it's annual compounding, $m_1=2$, $R_1=0.0325$, $m_2=1$

$$R_2 = 0.0327$$

2)

If it's quarterly compounding, $m_1=2$, $R_1=0.0325$, $m_2=4$

$$R_2 = 0.0323$$

3)

If it's quarterly compounding, $R_c = m * \ln(1 + R_m/m)$

$$R_m = 0.0325, m = 2$$

$$R_c = 0.0322$$

```

import numpy
y=(1+0.0325/2)**(2)-1
y
0.0327640625000003

y=(1+0.0325/2)**(1/2)-1
y=y*4
y
0.03236903073119102

y=2*math.log(1+0.0325/2)
y
0.03223876375976678

```

1.3

$A=10000, n=30/365, R = 0.01715$

$A1 = A * \exp(R * n) = 10141.96$

1.4

$FV = 100$

1)

$Bond1 = FV * 0.05 / 2 * (\exp(-0.03 * 0.5) + \exp(-0.03 * 1) + \exp(-0.035 * 1.5)) + (FV + FV * 5\% / 2) * \exp(-0.035 * 2) = 102.83$

2)

$Bond2 = FV * 0.06 / 2 * (\exp(-0.03 * 0.5) + \exp(-0.03 * 1) + \exp(-0.035 * 1.5) + \exp(-0.035 * 2) + \exp(-0.0425 * 2.5) + \dots + \exp(-0.0425 * 5) + \exp(-0.045 * 5.5) + \dots + \exp(-0.045 * 9.5)) + (FV + FV * 0.06 / 2) * \exp(-0.045 * 10) = 111.8428$

3)

$FV * 0.05 / 2 * (\exp(-y1 * 0.5) + \exp(-y1 * 1) + \exp(-y1 * 1.5)) + (FV + FV * 5\% / 2) * \exp(-y1 * 2) = 102.83$

$y1 = 0.0349$

Replace all the zero rates with y_2 in the second question of the problem we'll get $y_2 = 0.0447$

```
y = 2.5 * (math.exp(-0.03*0.5)+math.exp(-0.03)+math.exp(-0.035*1.5))+102.5*math.exp(-0.07)
y
102.83139602587813
from scipy.optimize import root
f = lambda x: 2.5*(math.exp(-0.5*x)+math.exp(-x)+math.exp(-1.5*x))+102.5*math.exp(-2*x)-102.83
sol = root(f,0)
print(sol.x[0])
0.034915034740735755
/tmp/ipython-input-2396133106.py:2: DeprecationWarning: Conversion of an array with ndim > 0 to a
  f = lambda x: 2.5*(math.exp(-0.5*x)+math.exp(-x)+math.exp(-1.5*x))+102.5*math.exp(-2*x)-102.83
```

0.5	3	0.03	2.955336
1	3	0.03	2.911337
1.5	3	0.035	2.846563
2	3	0.035	2.797181
2.5	3	0.0425	2.697599
3	3	0.0425	2.64088
3.5	3	0.0425	2.585354
4	3	0.0425	2.530994
4.5	3	0.0425	2.477778
5	3	0.0425	2.425681
5.5	3	0.045	2.342251
6	3	0.045	2.290138
6.5	3	0.045	2.239186
7	3	0.045	2.189367
7.5	3	0.045	2.140656
8	3	0.045	2.093029
8.5	3	0.045	2.046462
9	3	0.045	2.00093
9.5	3	0.045	1.956412
10	103	0.045	65.6757
			111.8428
		0.04465	
		111.8428	

1.5

1)

Locked in paying $1\text{mil} \times 1.15 = 1.15\text{mil}$

Without hedging $1\text{mil} \times 1.175 = 1.175\text{mil}$

Gain $1.175 - 1.15 = 0.025\text{mil}$

2)

$$R_f = R \cdot \exp[(r_{USD} - r_{EUR}) \cdot t]$$

$$R_f = 1.15, R = 1.1, t=0.5$$

$$r_{USD} - r_{EUR} = 0.0889$$