## Question 1

1) (a) The forward interest rates are:

$$R_{1-2} = R_2 + \frac{(R_2 - R_1)T_1}{T_2 - T_1}$$

$$R_{3-6} = 3.2 + \frac{(3.2-3)\times 3}{6-3} = 3.4$$

$$R_{6-9} = 3.4 + \frac{(3.4-3.2)\times 6}{9-6} = 3.8$$

$$R_{9-12} = 3.5 + \frac{(3.5-3.4)\times 9}{12-9} = 3.8$$

$$R_{12-15} = 3.6 + \frac{(3.6-3.5)\times 12}{15-12} = 4.0$$

$$R_{15-18} = 3.7 + \frac{(3.7-3.6)\times 15}{18-15} = 4.2$$

II) (a) Theoretical Price:

$$P = Fe^{-r(n)n} + \sum_{t\in\{0.5,1,1.5,2\}} c\cdot e^{-r(t)t}$$

$$= 1000e^{-0.032\times 2} + 20e^{-0.02\times 0.5} + 20e^{-0.023\times 1} + 20e^{-0.027\times 1.5} + 20e^{-0.032\times 2}$$

$$= 1015.32$$

(b) bond's yield $y$:

$$1015.32 = Fe^{-yn} + \sum_{t\in\{0.5,1,1.5,2\}} c\cdot e^{-yt}$$

$$1015.32 = 20e^{-0.5y} + 20e^{-y} + 20e^{-1.5y} + 1020e^{-2y}$$

$$y = 0.0318 = 31.8\%$$

iii) (a) annual compounding:

$$\left(1+\frac{5\%}{2}\right)^2 = \left(1+\frac{r_a}{1}\right)^1$$

$$r_a = \left(1+\frac{5\%}{2}\right)^2 - 1 = 5.06\%$$

(b) Monthly Compounding:

$$\left(1+\frac{5\%}{2}\right)^2 = \left(1+\frac{r_m}{12}\right)^{12}$$

$$r_m = 12 \times \left[\sqrt[12]{\left(1+\frac{5\%}{2}\right)^2} - 1\right] = 4.95\%$$

(C) Continuous Compounding:

$$\left(1+\frac{5\%}{2}\right)^2 = e^{r_c}$$

$$r_c = 2 \times \ln\left(1+\frac{5\%}{2}\right) = 4.94\%$$

## Question 2

1) (a) Applying the dividend discount model:

$$P_0 = \sum_{t=1}^{\infty} \frac{D_t}{(1+k)^t} = D \cdot \sum_{t=1}^{\infty} \frac{1}{(1+k)^t}$$

$$= 5 \times \frac{1}{1+k} \times \frac{1}{1-\frac{1}{1+k}}$$

$$= 5 \times \frac{1}{1.125} \times \frac{1}{1-\frac{1}{1.125}} = 40$$

(b) Since : $P_0 = \dfrac{D_1 + P_1}{1+r}$

we have :

$$P_1 = P_0(1+r) - D_1$$

thus, after paying out $D_1$:

$$P_1 = 40 \times (1 + 12.5\%) - 5$$
$$= 40 = P_0$$

the value per share will not change because the $\frac{D}{P}$ ratio is equivalent to the market capitalization ratio making the growth rate zero.

(C) set a higher plowback ratio to retain more earnings for reinvestment.
we have the growth rate $g = b \times ROE$, thus we want to aviod paying out all the earnings as dividends (i.e. aviod $b=0$).

11) (a) expected yield:

$$y_e = r_p + r = 3\% + 5\% = 8\%$$

and we have:

$$y_e = 0.80 \times y_p + 0.20 \times y_d$$

where the promised yield:

$$y_p = \frac{1050}{P} - 1$$

and the default yield:

$$y_d = \frac{0}{P} - 1 = -1$$

thus:

$$0.80 \times \left(\frac{1050}{P} - 1\right) - 0.20 \times 1 = 8\%$$

$$P = 777.78 \quad (\text{price of bond})$$

and then: $y_P = \frac{1050}{777.78} - 1 = 35\%$

(b) recovery rate of a bond is the bond's market value shortly after default, as a fraction of its face value.

$\because \ \lambda = \frac{s}{1-R}$

$\therefore \ s = \lambda(1-R) = 0.20 \times (1-0.40) = 0.12$

$\because \ s = y_P - r = \frac{1050}{P} - 1 - 5\% = 0.12$

$\therefore \ P = 897.44$, the prices increases if $R=40\%$

(c) Buy. I will want to purchase for protection against losses from the default.

## Question 3

I) average hazard rate:

$$\bar{\lambda}(3) = \frac{s}{1-R} = \frac{0.0060}{1-40\%} = 1\%$$

II) (a) upper bound:

$$F = (20 + 2e^{-7\%}) \cdot e^{7\%} = 23.45$$

(b) assume we have $x$ amount of CHF at $t=0$

then we can exchange for $1.1x$ EUR at $t=0$

At $t=\frac{1}{2}$, to have no arbitrage:

$$1.1xe^{r\cdot\frac{1}{2}} = 1.0950\cdot xe^{3\%\cdot\frac{1}{2}}$$

$$\therefore r = 2\times\ln(1.095\times e^{1.5\%}) = 2.09\%$$

## Question 4

1) (a) initial value: $f_0 = S_0 - Ke^{-rT} = 0$

(b) forward price at $t=0$:

$$F_0 = S_0e^{rT} = 40\times e^{5\%\times1} = 42.05$$

(c) forward price:

$$F_{\frac{1}{2}} = S_{\frac{1}{2}}\cdot e^{r(1-\frac{1}{2})}$$

$$= 45\times e^{5\%\times\frac{1}{2}} = 46.14$$

forward value:

$$f_0 = S_{\frac{1}{2}} - Ke^{-r(1-\frac{1}{2})}$$

$$= 45 - 42.05\times e^{-5\%\times\frac{1}{2}}$$

$$= 3.99$$

(d) The value of the forward contract increased before maturity. (from 0 to 3.33)

Reflecting a shift in the underlying

spot exchange rate.
Pepaw can close out the forward contract by entering into an opposite forward contract with the same expiration data at the current market rate to take the opportunity profit available at this time.

11) let $u=1.3$ and $d=1.05$

$$f_{uu} = \max(Su^2-K,0) = \max(169-105,0) = 64$$

$$f_{ud} = \max(Sud-K,0) = \max(136.5-105,0) = 31.5$$

$$f_{dd} = \max(Sd^2-K,0) = \max(110.25-105,0) = 5.25$$

For $r = r_{uu} = 1\% + 4\% + 4\% = 9\%$:

$$f_{u_1} = e^{-r_{uu}\Delta t}(P_u \cdot f_{uu} + P_d \cdot f_{ud})$$

$$= e^{-0.09} \cdot (0.65 \times 64 + 0.35 \times 31.5)$$

$$= 48.10$$

$$f_{d_1} = e^{-r_{uu}\Delta t}(P_u f_{ud} + P_d \cdot f_{dd})$$

$$= e^{-0.09} \cdot (0.65 \times 31.5 + 0.35 \times 5.25)$$

$$= 20.39$$

For $r = r_{ud} = 1\% + 4\% + 0 = 5\%$:

$$f_{u_2} = e^{-r_{ud}\Delta t}(P_u f_{uu} + P_d f_{ud})$$

$$= e^{-0.05} \cdot (0.65 \times 64 + 0.35 \times 31.5)$$

$$= 50.06$$

$$f_{d_2} = e^{-r_{ud}\Delta t}\left( P_u f_{ud} + P_d \cdot f_{dd} \right)$$
$$= e^{-0.05} \cdot \left(0.65 \times 31.5 + 0.35 \times 5.25\right)$$
$$= 21.22$$

$$\begin{cases} f_u = 0.70 f_{u_1} + 0.30 f_{u_2} \\ \quad = 0.70 \times 48.10 + 0.30 \times 50.06 = 48.69 \\ f_d = 0.70 f_{d_1} + 0.30 f_{d_2} \end{cases}$$
$$= 0.70 \times 20.39 + 0.30 \times 21.22 = 20.64$$

For $r = r_u = 1\% + 4\% = 5\%$ :

$$f_1 = e^{-r_u \Delta t}\left( P_u f_u + P_d \cdot f_d \right)$$
$$= e^{-0.05} \cdot \left(0.65 \times 48.69 + 0.35 \times 20.64\right)$$
$$= 36.98$$

For $r = r_d = 1\% + 0 = 1\%$ :

$$f_2 = e^{-r_d \Delta t}\left( P_u f_u + P_d \cdot f_d \right)$$
$$= e^{-0.01} \cdot \left(0.65 \times 48.69 + 0.35 \times 20.64\right)$$
$$= 38.49$$

$\therefore$ Finally : $f = 0.70 f_1 + 0.30 f_2$

$$= 0.70 \times 36.98 + 0.30 \times 38.49$$
$$= 37.43$$

is the value of the call option.

# Question 5

In [135…
```python
import numpy as np
import pandas as pd
from scipy.stats import norm
```

## (a) Why might Icarus sign this reverse mortgage?

Icarus wanted to have immediate access (in terms of money) to the value of his property while still being able to stay within it for the rest of his life. At the age of 80, Icarus might not have cash incomes to support his daily life and therefore reverse mortgage is a very good choice of loan to obtain cash at hand without the need to repay the loan monthly. Additionally, Icarus might think that he will only be living in his property for a relatively short time (due to his age) which means that the interest of the loan will not likely to exceed the value of his property. Even though a reverse mortgage might not be the best loan (financially speaking), Icarus will loss his property eventually (in a relatively short time), making use of the value of the property at least makes his current life better.

## (b) With volatility of the property price prescribed as 13%, and deferment rate prescribed as 1%; calculate the No-Negative Equity Guarantee of this reverse mortgage to 1 decimal place, showing your procedure.

In [136…
```python
r = 0.03                    #risk-free interest rate
q = 0.01                    #deferment rate
sigma = 0.13                #volatility
l = 0.025                   #loan rate
K = 500000                  #loan principle
S = 500000                  #current value of the property
```

In [137…
```python
#import data from local excel file (change path before running the code)
path = '2015-vbt-mns-anb.xlsx'
data = pd.read_excel(path, sheet_name='2015 MNS ANB', header=2)
data = data[data.columns[:28]].copy()
data = data.iloc[:-11].copy()
```

In [138…
```python
def calculate_qx(data, x):
    #obtain the mortality rates (n-1|qx) before the attained age
    df_2 = data.loc[data["Iss. Age"] == x] / 1000
    df_2 = df_2.rename(columns={"Iss. Age": 0})
    df_2 = df_2.drop(["Att. Age"], axis=1)
    df_2[0] = 0
    #obtain 1-(n-1|qx) before the attained age
    df_3 = 1 - df_2
    #obtain the annual unconditional survival rates (npx) before the attained age
    npx = [1]
    for r in df_3.values[0][1:]:
        np80.append(r*npx[-1])
    #obtain the mortality rates (n-1|qx) beyond the attained age
    df_4 = data["Ult."].loc[data["Iss. Age"] > x] / 1000
    #obtain the annual unconditional survival rates (npx) beyond the attained age
    for r in df_4.values:
        npx.append((1-r)*npx[-1])
    #compute the probability that a person aged x exact will die before reaching age (x +1)
    p = npx[0] * df_2[1]
    return p.values[0]
```

In [139…
```python
max_age = max(data['Iss. Age'])

#obtain qx
qx = []
for x in range(80, max_age+1):
    p = calculate_qx(data, x)
    qx.append(p)
```

In [140…
```python
frame = pd.DataFrame(index=range(0,len(qx)), data={'age':range(80, max_age+1), 'qx':qx})

frame['px'] = 1 - frame['qx']

frame.loc[0,'npx'] = 1
for t in range(1,len(qx)):
    frame.loc[t,'npx'] = frame.loc[t,'px'] * frame.loc[t-1,'npx']

frame.loc[0,'pt'] = 0
for t in range(1,len(qx)):
    frame.loc[t,'pt'] = frame.loc[t,'qx'] * frame.loc[t-1,'npx']

frame.loc[0, 'd1'] = 0
for t in range(1,len(qx)):
    frame.loc[t, 'd1'] = (np.log(S*np.exp(-q*t)/(K*np.exp((l-r)*t))) + 0.5*t*sigma**2) / (sigma*np.sqrt(t))

for t in range(0, len(qx)):
    frame.loc[t,'d2'] = frame.loc[t,'d1'] - sigma * np.sqrt(t)

frame['N-d1'] = norm.cdf(-frame['d1'])
frame['N-d2'] = norm.cdf(-frame['d2'])

for t in range(1, len(qx)):
    frame.loc[t,'NNEG'] = frame.loc[t,'pt'] * (K*np.exp((l-r)*t)*frame.loc[t,'N-d2'] - S*np.exp(-q*t)*frame.loc

for t in range(1, len(qx)):
    frame.loc[t,'pv'] = frame.loc[t,'pt'] * K*np.exp((l-r)*t)

frame.head()
```

Out[140…

| | age | qx | px | npx | pt | d1 | d2 | N-d1 | N-d2 | NNEG | pv |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 80 | 0.00487 | 0.99513 | 1.000000 | 0.000000 | 0.000000 | 0.000000 | 0.500000 | 0.500000 | NaN | NaN |
| **1** | 81 | 0.00528 | 0.99472 | 0.994720 | 0.005280 | 0.026538 | -0.103462 | 0.489414 | 0.541202 | 142.449728 | 2626.832945 |
| **2** | 82 | 0.00581 | 0.99419 | 0.988941 | 0.005779 | 0.037531 | -0.146317 | 0.485031 | 0.558164 | 223.035489 | 2860.908987 |
| **3** | 83 | 0.00668 | 0.99332 | 0.982335 | 0.006606 | 0.045966 | -0.179201 | 0.481669 | 0.571110 | 314.365520 | 3253.885676 |
| **4** | 84 | 0.00786 | 0.99214 | 0.974613 | 0.007721 | 0.053077 | -0.206923 | 0.478835 | 0.581965 | 426.135825 | 3784.130291 |

In [141…
```python
#compute the present value of the NNEG
NNEG = sum(frame['NNEG'].iloc[1:])
print("The NNEG of the reverse mortgage is:  ", round(NNEG,1))
```

```
The NNEG of the reverse mortgage is:   33026.2
```

## (c) Thus calculate the present value of the equity release mortgage to 4 significant figures.

In [145…
```python
#compute the present value of the loan
PV_loan = sum(frame['pv'].iloc[1:])
print("Present value of the loan is: ", round(PV_loan, 1))
```

```
Present value of the loan is:  169471.6
```

In [146…
```python
#compute the present value of the equity release mortgage
PV_mortgage = PV_loan - PV_NNEG
print('Present value of the equity release mortgage is ', round(PV_mortgage, 4));
```

```
Present value of the equity release mortgage is  136445.4561
```

## Question 6

### (a) Describe and explain both what is an API and the cloud, giving an example.

An application programming interface (API) is a set of definitions and protocols for building and integrating application software. APIs enable two software components to communicate without having to know how each other is implemented. This effectively simplifies the development process and save time and money. For example, the weather app on my phone communicates with the weather bureau's software system through an API to acquire the weather information and finally displaying it on my screen (while the app itself does not have the ability to make weather forecast).

The cloud refers to servers that can be accessed online, and the software and databases that run on those servers. Users of the cloud do not have to run software applications or store data on their own machines. Instead, they can access files and applications from any devices because the computing and storage takes place on servers in a data centre but not locally. For example, users of Gmail (a cloud email provider) can log in to their accounts on any device and gain access to their emails and even the attachments (documents, pictures, etc.) that are stored on Google Drive (a cloud storage provider).

### (b) How have API's changed the way businesses interact, between both external businesses and customers. Give one example of such development.

APIs allow businesses to share data and applications. APIs make the process of sharing data and applications between different systems simple and effective. With an API, a firm can quickly and easily connect their systems to an external firm' programs without any pre-existing connections and use external functionalities without understanding how they work. Therefore, workflows can be carefully and strategically orchestrated with APIs even when firms are collaborating remotely. APIs help ensure that the connectivity and collaboration between firms remain strong even when workflows and processes change since firms can always interact with each other's system (under permission) through APIs. One example of companies interacting with each other through API is the pop-up weather snippet whenever you use Google search for weather. This is Google coordinating with other software on the web through API.

Customers are distinct and often have unique set of needs and wants that cannot all be satisfied by a single service/product provided by the businesses. However, by opening up information access via API, businesses empower their customers to tailor their own experiences which leads to seemingly endless possibilities. Additionally, while customers are enjoying their tailored service/product, companies are also getting a better understanding of customer behaviours and preferences by monitoring the API usage. These are useful insights that can be used to make improvements and bring better experience to customers. For example, online retailers can use APIs to track a customer's journey on its website and use this information make better product suggestions that the customer might like. Providing customers with recommendations based on their individual preferences will boost engagement between the company and the customers.

## (c) Describe the processes in order to generate accurate and relevant computational models, what are some of the pitfalls during modelling.

The input data is one of the key factors that determine whether a computational model is accurate and relevant. For example, to train a powerful machine learning model, the training data used must be relevant, up-to-date and adequate. Moreover, better the data at hand the more insights can be extracted which are useful information when modelling. By understanding the data well, a suitable model can be chosen. For example, tree models will perform better on non-linear data than linear models.

Leveraging domain expertise during the modelling process can also improve the performance of the model. For example, important/new features can be identified/engineered based on previous experience and domain knowledge. This customizes the model to better fit the specific business scenario that the model will be applied to.

Furthermore, testing for generalization with a well-designed evaluation strategy is a crucial step before the model can be applied. This avoids the modelling process from falling into the pitfall of overfitting. That is, designing a very complex model with great performance on the existing data but not on new data. During the modelling process, it is very common that the designers focus too much on the performance of the model and end up with a very complex model (not interpretable and time inefficient).

Another pitfall during modelling is the failure to document the progress and the results of previous versions of the model. A well-documented progress allows designers and stakeholders to see the improvements along the modelling progress. Keeping documentations can also help interpreting the model and demonstrate what makes the model powerful (the changes that brought great improvements).