# Monte Carlo - Final project
# Squaring venture capital valuations with reality

Yuan-Chun Lin

April 29, 2022

**Abstract**

My goal is to implement the Monte Carlo method for the valuation of start-up firm. The reference paper is Gornall and Strebulaev (2020). In this report I would summarize the model and reproduce the result of Table 1, in particular, to consider Baseline scenario and Junior seniority.

## 1 Model Setting

The model being discussed in this project is a Valuation model of a VC-backed company. It uses the price of a VC-style financing round to find the fair value of a company at the time of current round. Consider a company that raises a financing round of amount $I$ at time 0. The company exits at value $X(T)$ at some time $T$ in an IPO, M&A, or a liquidation. All shareholders are paid out at exit, with the investor's payoff being a function of the exit amount, $f(X(T))$.

### 1.1 Baseline Scenario

Take a prototypical unicorn as an example, it is raising $100 million of new VC investment at $1 per share in a Series B round with a post–money valuation of $1 billion using standard preferred shares with a conversion option, automatic conversion in IPOs, a guaranteed return of initial investment in M&A exits and liquidation events, and no additional provisions.

In the past, this company raised $50 million of VC investment in a Series A round with a post–money valuation of $450 million using the preferred shares with the same rights and terms as, and pari passu seniority with, the newly issued shares.

Using subscripts to denote the different rounds, $P_A = 450$, $P_B = 1,000$, $I_A = 50$, and $I_B = 100$ (all values in millions). After the current round, if all shares convert, the new investor owns 10% of the total shares, the old investor owns 10%, and the current common shareholders own the remaining 80%.

For these capital structure inputs, the payout to the new investor in an IPO is the converted payoff and in an M&A exit or liquidation is the following function of the exit value X:

$$f_B^{M\&A}(X) = \max\{\min\{\frac{I_B}{I_A+I_B}X, I_B\}, X \times \frac{I_B}{P_B}\}$$

This equation is the standard formula for the valuation of convertible preferred security at maturity, for an all-equity firm.

### 1.2 Junior Seniority Scenario

Many unicorns make their most recent investors senior to all other shareholders, so that their liquidation preference must be satisfied before other investors receive anything, making an investor class senior increases their payouts in low M&A exits. In this project, we focus on the junior seniority case. In theory, the new investor could also be junior to an existing investor:

$$f_B^{M\&A}(X) = \max\{\min\{X - I_A, I_B\}, X \times \frac{I_B}{P_B}\}$$

This is extremely uncommon in practice, but, even in this case, significant overvaluation still exists because even junior preferred equity is senior to common equity.

# 2 Result

Those tables show the fair valuation of the Company and Common Share as implied by the model. We observe the post–money valuation of the new round (PMV) in millions of dollars, the fair value of the company (or common share) that makes that round fairly priced (FV) in millions of dollars, and the percentage by which the post–money valuation overstates the fair value ($\Delta V or \Delta C$)

## 2.1 Result for Baseline Scenario

This project reproduced the result for the baseline scenario based on section 2 in the reference paper. First, in our baseline case, we have $P_A = 450$, $P_B = 1,000$, $I_A = 50$, and $I_B = 100$ (all values in millions), and we assume that the asset value follows geometric Brownian motion with interest rate $r = 0.025$ and $\sigma = 0.9$. We also consider the factor of Option pool, assume unissued stock options are 5% of the total post–money valuation.

Then, through inverse method to generate the exit time T from the exponential distribution with exit rate $\lambda = 0.25$ and generate a geometric Brownian motion that representing the exit payoff.

Next, I implement the probability of an IPO exit for a given exit value according to section 2.4 Eq. 17 of the reference paper. Conditional IPO probability is described in Figure 1.

$$p^{IPO}(X) = \begin{cases} 0 & \text{for } X \leq \$32m \\ 0.65 \times \frac{\log(X)-\log(\$32m)}{\log(\$1b)-\log(\$32m)} & \text{for } \$32m \leq X \leq \$1b \\ 0.65 + 0.2 \times \frac{\log(X)-\log(\$1b)}{\log(\$100b)-\log(\$1b)} & \text{for } \$1b \leq X \leq \$100b \\ 1 & \text{for } \$100b \leq X. \end{cases}$$

Figure 1: $IPOProbability$

Lastly, I model automatic conversion terms by writing the exit payoff, $f\left(X\right)$, as the sum of the payoff in an IPO, $f^{IPO}\left(X\right)$, like:

$$f_B^{IPO}\left(X\right) = X \times \frac{I_B}{P_B}$$

and the payoff in M&A or liquidations that cannot trigger automatic conversion, $f^{M\&A}\left(X\right)$, like:

$$f_B^{M\&A}\left(X\right) = \max\left\{\min\left\{X - I_A, I_B\right\}, X \times \frac{I_B}{P_B}\right\}$$

weighted by the probability of each outcome conditional on the exit value, as the sum of the payoff in an IPO, $f^{IPO}\left(X\right)$, and the payoff in M&A or liquidations that cannot trigger automatic conversion, $f^{M\&A}\left(X\right)$, weighted by the probability of each outcome conditional on the exit value, $p^{IPO}$ and $1 - p^{IPO}$.

$$f_B\left(X\right) = p^{IPO}\left(X\right) f_B^{IPO}\left(X\right) + \left(1 - p^{IPO}\left(X\right)\right) f_B^{M\&A}\left(X\right)$$

Using the following constrain, we can get the fair value of company.

$$I_B = E\left[e^{-rT} f_B\left(X(0)e^{(r-\sigma^2/2)T+\sigma\sqrt{T}Z}\right)\right]$$

For the common share, we can get total shares after the current round would be 1,000 shares (by $P_B/\textit{stock price}$, where *stock price* equals to \$1 per share). Then, after the current round, we know the current common shareholders own the 80% of total shares, equal to 800 shares. Further, we can calculate the fair value of common share from the fair value of company. In total fair value of company, 80% of company belongs to common shareholders. Thus, we use 80% of total company's fair value divided by the total common shares to calculate the fair value of common share. We don't need to consider option pool when calculating total shares, because the fair value of company have considered the option pool factor.

Table 1 indicates the Monte Carlo simulation result for Baseline Scenario. The fair value of company in the baseline scenario is \$771 million dollars (corresponding to "X0" in my python code), which exactly equals to the result of table 1 in reference paper, with a overvaluation approximately 29.7% (Details showed in the output of my python code).

However, the simulation result of Common Share displayed slightly different result from the reference paper. In my table 1 has 0.81 as the common share fair value, compared to the 0.78 in reference paper. With a overvaluation approximately 23.2% , which is also different from original paper's 28%.

I think this is because of the option pool factor. In this paper, we didn't have the data of Series A funding. Thus, we can not calculate explicitly the fair value for each class of shares. And even though we have that number, we still don't know how option pool effects for the share or price. So, I think that is the main reason why we got the slightly different result in fair value of common share.

| Baseline Scenario | PMV($m) | FV($m) | Overvaluation |
|---|---|---|---|
| Company | 1,000 | 771 | 29.7% |
| Common Share | 1 | 0.81 | 23.2% |

Table 1: Baseline Scenario

## 2.2 Result for Junior Seniority Scenario

In the Junior Seniority Scenario, we assume that the new investors are junior to other existing investors. So, the payoff of M&A would be changed to:

$$f_B^{M\&A}(X) = \max\{\min\{X - I_A, I_B\}, X \times \frac{I_B}{P_B}\}$$

Table 2 indicates the Monte Carlo simulation result for Junior Seniority Scenario. The fair value of company in the baseline scenario is \$811 million dollars (corresponding to "X0" in my python code), which exactly equals to the result of table 1 in reference paper, with a overvaluation approximately 23.3% (Details showed in the output of my python code).

However, there are still some differences in Common Share. the simulation result of Common Share displayed slightly different result from the reference paper. In my table 2 has 0.85 as the common share fair value, compared to the 0.82 in reference paper. With a overvaluation approximately 17.1% , which is also different from original paper's 22%.

I think the reason why we got the different number is the same as the baseline scenario.

| Junior Seniority | PMV($m) | FV($m) | Overvaluation |
|---|---|---|---|
| Company | 1,000 | 811 | 23% |
| Common Share | 1 | 0.82 | 22% |

Table 2: Junior Seniority Scenario

# 3 Python Code:

In [7]:

```python
import numpy as np
import statsmodels.api as sm
import matplotlib.pyplot as plt
from scipy.stats import norm
from scipy.stats import uniform
import tqdm
import math
```

In [8]:

```python
# CI function
def CI(data,alpha):
    sample_mean=np.mean(data) # data is a list!
    sample_sigma=np.std(data)
    critical_point = norm.ppf(1-alpha/2)
    LB=sample_mean-critical_point*sample_sigma/np.sqrt(len(data))
    UB=sample_mean+critical_point*sample_sigma/np.sqrt(len(data))
    return LB,UB
```

# Baseline :

In [9]:

```python
'''
Company Fair Value and Common share fair value
'''
# parameters
share_new,share_old,share_current=0.1,0.1,0.8
r,sigma=0.025,0.9
size=10**8
lambda_param=0.25
optionpool=0.05
Pa,Pb,Ia,Ib=450*(1-optionpool),1000*(1-optionpool),50,100


X0=771 ##


U=np.random.uniform(0,1,size)
T=-np.log(1-U)/lambda_param
# T=np.random.exponential(1/lambda_param,size)

Z=np.random.normal(0,1,size)
X=X0*np.exp((r-0.5*(sigma**2))*T+sigma*np.sqrt(T)*Z)

P=np.ones(size)
P[X<=np.ones(size)*100000]=0.65+0.2*(np.log(X[X<=np.ones(size)*100000])-np.log(1000))/(np.log(100000)-np.log(1000))
P[X<=np.ones(size)*1000]=0.65*(np.log(X[X<=np.ones(size)*1000])-np.log(32))/(np.log(1000)-np.log(32))
P[X<=np.ones(size)*32]=0

Merge=np.maximum(np.minimum(Ib*X/(Ia+Ib),np.ones(size)*Ib),X*Ib/Pb)*np.exp(-r*T)
IPO = (X*Ib/Pb)*np.exp(-r*T)
payoff=IPO*P+Merge*(1-P)

price=1
common_stock_shares=(Pb/price)*share_current
common_stock_FVprice=X0*share_current/common_stock_shares

# C=np.zeros(size)
# C[X*Ia/Pa>np.minimum(Ia,X)]=1
# C[X*Ib/Pb>np.minimum(Ib,X)]=2
# total_commonshare=1000*P+(800+C*100)*(1-P) #
# FV_common=Xb/total_commonshare
# common_stock_FVprice=np.mean(FV_common)

print('\nExpectation of payoff is {}; \nThe confidence interval is {}'.format(np.mean(payoff),CI(payoff,0.05)))
print('\nOvervaluation(%): {}'.format(100*(1000-X0)/X0))

print('\nThe fair value of common share price is {}'.format(common_stock_FVprice))
print('\nOvervaluation(%): {}'.format(100*(1-common_stock_FVprice)/common_stock_FVprice))
```

```
Expectation of payoff is 100.14161266079128;
The confidence interval is (99.11860356484958, 101.16462175673298)

Overvaluation(%): 29.701686121919586

The fair value of common share price is 0.8115789473684212

Overvaluation(%): 23.216601815823587
```

# Seniority :

In [10]:

```python
'''
Company Fair Value
'''
# parameters
share_new,share_old,share_current=0.1,0.1,0.8
r,sigma=0.025,0.9
size=10**7
lambda_param=0.25
optionpool=0.05
Pa,Pb,Ia,Ib=450*(1-optionpool),1000*(1-optionpool),50,100

X0=811 ##
```

```python
U=np.random.uniform(0,1,size)
T=-np.log(1-U)/lambda_param
# T=np.random.exponential(1/lambda_param,size)

Z=np.random.normal(0,1,size)
X=X0*np.exp((r-0.5*(sigma**2))*T+sigma*np.sqrt(T)*Z)

P=np.ones(size)
P[X<=np.ones(size)*100000]=0.65+0.2*(np.log(X[X<=np.ones(size)*100000])-np.log(1000))/(np.log(100000)-np.log(1000))
P[X<=np.ones(size)*1000]=0.65*(np.log(X[X<=np.ones(size)*1000])-np.log(32))/(np.log(1000)-np.log(32))
P[X<=np.ones(size)*32]=0

Merge=np.maximum(np.minimum(X-Ia,np.ones(size)*Ib),X*Ib/Pb)*np.exp(-r*T)
IPO = (X*Ib/Pb)*np.exp(-r*T)
payoff=IPO*P+Merge*(1-P)

price=1
common_stock_shares=(Pb/price)*share_current
common_stock_FVprice=X0*share_current/common_stock_shares

print('\nExpectation of payoff is {}; \nThe confidence interval is {}'.format(np.mean(payoff),CI(payoff,0.05)))
print('\nOvervaluation(%): {}'.format(100*(1000-X0)/X0))

print('\nThe fair value of common share price is {}'.format(common_stock_FVprice))
print('\nOvervaluation(%): {}'.format(100*(1-common_stock_FVprice)/common_stock_FVprice))
```

```
Expectation of payoff is 99.99330091776632;
The confidence interval is (98.20804807454253, 101.7785537609901)

Overvaluation(%): 23.304562268803945

The fair value of common share price is 0.8536842105263159

Overvaluation(%): 17.13933415536373
```