MIE 1622H Assignment 3 - Credit Risk Modeling and Simulation

1. (60 %) Implement portfolio credit risk simulation model in Python

Structural Model for Portfolio Credit Risk

N_s systemic scenarios, N_i idiosyncratic scenario for each systemic, J=100 counterparties, k=50 credit drivers

8 credit-state with 7 boundaries: $Default$ | CCC | B | BB | BBB | A | AA | AAA

8 migration exposures, consider recovery for default

Generate correlated random variable $y_{j(k)} \sim N(0,1)$, total N_s times
Generate independent random variable $z_j \sim N(0,1)$, N_i times for each systemic scenario

Compute creditworthiness index for counterparty j: $w_j = \beta_j y_{j(k)} + \sigma_j z_j$

Infer future credit state and losses:

$$w_j \in (boundary\ i, boundary\ j), where\ 1 \le i, j \le 7$$

$$\Rightarrow j^{th}\ credit\ state,\ j^{th}\ exposure$$

Total number of scenarios: $N = N_s \times N_i$

Loss matrix: $N \times J$

Scenario sets

1. Monte Carlo approximation 1: 5000 in-sample scenarios ($N_s = 1000, N_i = 5$)
2. Monte Carlo approximation 2: 5000 in-sample scenarios ($N_s = 5000, N_i = 1$)
3. True distribution: 100000 out-of-sample scenarios ($N_s = 100000, N_i = 5$)

Portfolio of corporate bonds

1. One unit invested in each of 100 bonds
2. equal value (dollar amount) is invested in each of 100 bonds

Assumption of losses distribution

1. non-Normal
2. Normal

Evaluate VaR and CVaR at quantile levels 99% and 99.9% for the two portfolios from 3 scenario sets

For each portfolio/ scenario/ quantile level α, compute 1-year loss and sort: $\ell_{(1)} \le \ell_{(2)} \le \cdots \le \ell_{(N)}$

1. non-Normal: $VaR_{\alpha,N} = \ell_{(\lceil N\alpha \rceil)}$
   $CVaR_{\alpha,N} = \frac{1}{N(1-\alpha)}\left[(\lceil N\alpha \rceil - N\alpha)\ell_{(\lceil N\alpha \rceil)} + \sum_{k=\lceil N\alpha \rceil+1}^{N} \ell_{(k)}\right]$
2. Normal: $VaR_{\alpha}^{N} = \mu_{\mathcal{L}} + \Phi^{-1}(\alpha) \cdot \sigma_{\mathcal{L}}$
   $CVaR_{\alpha}^{N} = \mu_{\mathcal{L}} + \frac{\phi(\Phi^{-1}(\alpha))}{1-\alpha} \cdot \sigma_{\mathcal{L}}$

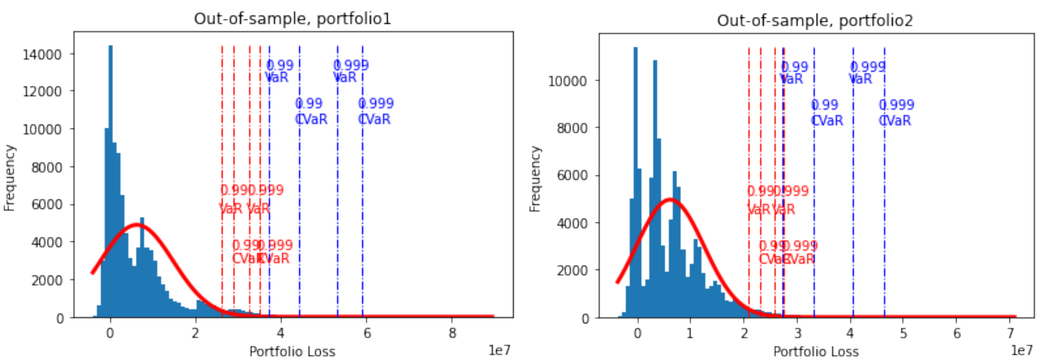
2. (25 %) Analyze your results

Produce the output from Python code

```
Portfolio 1:

Out-of-sample: VaR 99.0% = $37183974.34, CVaR 99.0% = $44195846.32
In-sample MC1: VaR 99.0% = $24834982.76, CVaR 99.0% = $32888829.99
In-sample MC2: VaR 99.0% = $37036928.81, CVaR 99.0% = $44424903.72
In-sample No: VaR 99.0% = $26180113.92, CVaR 99.0% = $29066235.87
In-sample N1: VaR 99.0% = $11920145.57, CVaR 99.0% = $13471002.09
In-sample N2: VaR 99.0% = $26142983.23, CVaR 99.0% = $29025572.88

Out-of-sample: VaR 99.9% = $53243955.32, CVaR 99.9% = $59113193.99
In-sample MC1: VaR 99.9% = $41813501.11, CVaR 99.9% = $49391724.49
In-sample MC2: VaR 99.9% = $53127487.03, CVaR 99.9% = $60962100.32
In-sample No: VaR 99.9% = $32686111.50, CVaR 99.9% = $35044106.92
In-sample N1: VaR 99.9% = $15416140.86, CVaR 99.9% = $16683208.79
In-sample N2: VaR 99.9% = $32641018.18, CVaR 99.9% = $34996127.66


Portfolio 2:

Out-of-sample: VaR 99.0% = $27356170.24, CVaR 99.0% = $33157462.59
In-sample MC1: VaR 99.0% = $18381865.29, CVaR 99.0% = $24048237.17
In-sample MC2: VaR 99.0% = $27252251.25, CVaR 99.0% = $33358194.22
In-sample No: VaR 99.0% = $21013165.53, CVaR 99.0% = $23168845.33
In-sample N1: VaR 99.0% = $10085179.15, CVaR 99.0% = $11372245.14
In-sample N2: VaR 99.0% = $21070086.56, CVaR 99.0% = $23232082.90

Out-of-sample: VaR 99.9% = $40638318.89, CVaR 99.9% = $46346896.06
In-sample MC1: VaR 99.9% = $30903305.73, CVaR 99.9% = $37206867.03
In-sample MC2: VaR 99.9% = $40821098.41, CVaR 99.9% = $47848105.13
In-sample No: VaR 99.9% = $25872574.72, CVaR 99.9% = $27633790.34
In-sample N1: VaR 99.9% = $12986528.56, CVaR 99.9% = $14038076.54
In-sample N2: VaR 99.9% = $25943734.73, CVaR 99.9% = $27710111.04
```

Plot loss distributions in Python that illustrate both out-of-sample and in-sample results.

1. True distribution (out-of-sample scenarios)

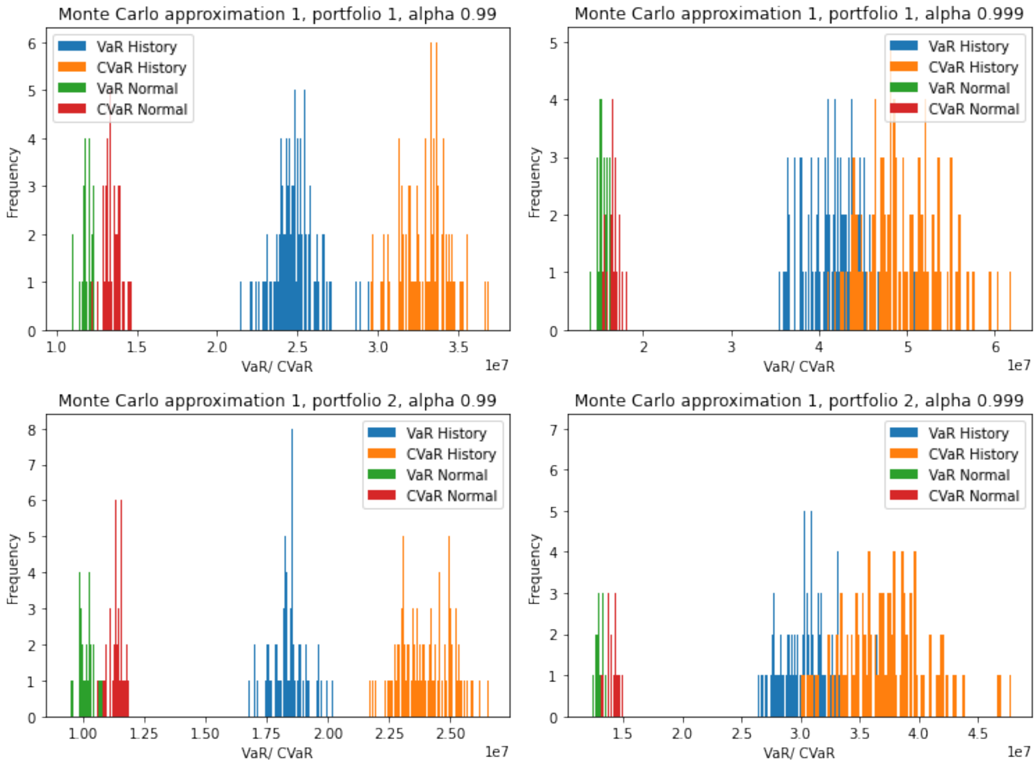


True distribution		99%		99.9%	
		VaR	CVaR	VaR	CVaR
Portfolio 1	non-Normal	37183974.34	44195846.32	53243955.32	59113193.99
	Normal	26180113.92	29066235.87	32686111.50	35044106.92
Portfolio 2	non-Normal	27356170.24	33157462.59	40638318.89	46346896.06
	Normal	21013165.53	23168845.33	25872574.72	27633790.34

portfolio 1 mean loss = 6366627.71, standard deviation = 8516991.99
portfolio 2 mean loss = 6214230.29, standard deviation = 6361445.51

To better evaluate sampling and model errors, generate in-sample datasets 100 times. Instead of showing 100 plots of loss distribution, plot VaR and CVaR distribution of 100 trails. Compute averages of VaR and CVaR of the results for 100 trials.

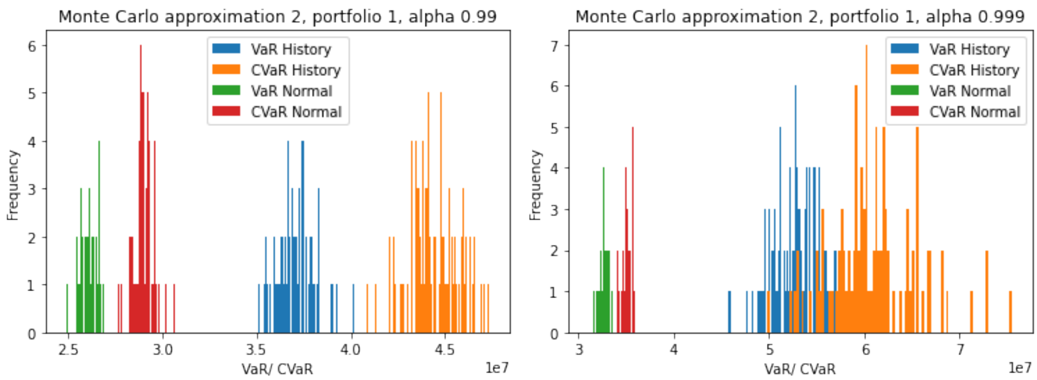
2. Monte Carlo approximation 1 (5000 in-sample scenarios)

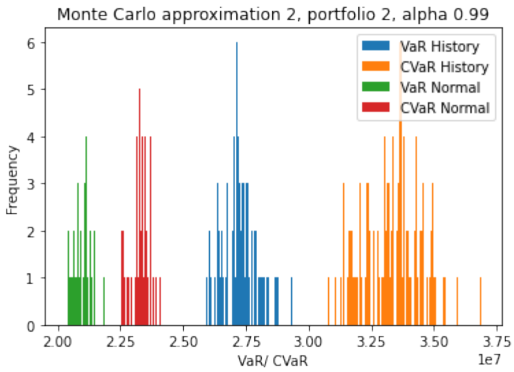


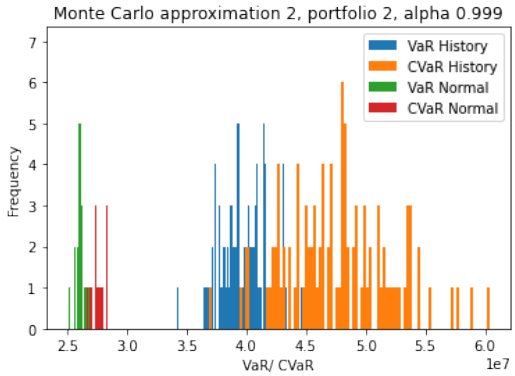
MC 1		99%		99.9%	
		VaR	CVaR	VaR	CVaR
Portfolio 1	non-Normal	24834982.76	32888829.99	41813501.11	49391724.49
	Normal	11920145.57	13471002.09	15416140.86	16683208.79
Portfolio 2	non-Normal	18381865.29	24048237.17	30903305.73	37206867.03
	Normal	10085179.15	11372245.14	12986528.56	14038076.54

portfolio 1 mean loss = 1273376.41, standard deviation = 4581040.51
portfolio 2 mean loss = 1249355.64, standard deviation = 3800070.27

3. Monte Carlo approximation 2 (1000 * 5 in-sample scenarios)







MC 2		99%		99.9%	
		VaR	CVaR	VaR	CVaR
Portfolio 1	non-Normal	37036928.81	44424903.72	53127487.03	60962100.32
	Normal	26142983.23	29025572.88	32641018.18	34996127.66
Portfolio 2	non-Normal	27252251.25	33358194.22	40821098.41	47848105.13
	Normal	21070086.56	23232082.90	25943734.73	27710111.04

portfolio 1 mean loss = 6353746.59, standard deviation = 8507905.23
portfolio 2 mean loss = 6227787.68, standard deviation = 6381009.93

Analyze sampling error when comparing non-Normal approximations to the true (out-of-sample) loss distribution.

Under the assumption that loss follows non-normal distribution, compare two in-sample datasets to the true distribution of portfolio losses (out-of-sample scenarios). ***Monte Carlo approximation 1*** covers 1000 systemic scenarios and the computational cost is 5 times faster, but it didn't generalize the true distribution well, all the values are underestimate. ***Monte Carlo approximation 2*** not saving any computational time, but it performs well, the computed VaR and CVaR are close to the true value. The performance of sampling strategy might relate to the number of systemic scenarios, 1000 systemic scenarios might not contain enough information like black swan event, the VaR would be underestimate in this case. On the other hand, there's a higher chance for 5000 systemic scenarios to include bad economic events.

non-Normal		99%		99.9%	
		VaR	CVaR	VaR	CVaR
Portfolio 1	True	37183974.34	44195846.32	53243955.32	59113193.99
	MC 1	24834982.76	32888829.99	41813501.11	49391724.49
	MC 2	37036928.81	44424903.72	53127487.03	60962100.32
Portfolio 2	True	27356170.24	33157462.59	40638318.89	46346896.06
	MC 1	18381865.29	24048237.17	30903305.73	37206867.03
	MC 2	27252251.25	33358194.22	40821098.41	47848105.13

Analyze model error when comparing Normal approximations to the true (out-of-sample) loss distribution.

Wrongly assumed that counterparty losses follow Normal distribution is dangerous, VaR and CVaR would be underestimated. Following table contains the computed mean loss and standard deviation of losses for each corporate bond from normal model, which is also underestimated.

Portfolio 1		99%		99.9%		mean loss	standard deviation
		VaR	CVaR	VaR	CVaR		
True	non-Normal	37183974.34	44195846.32	53243955.32	59113193.99		
	Normal	26180113.92	29066235.87	32686111.50	35044106.92	6366627.71	8516991.99
MC 1	non-Normal	24834982.76	32888829.99	41813501.11	49391724.49		
	Normal	11920145.57	13471002.09	15416140.86	16683208.79	1273376.41	4581040.51
MC 2	non-Normal	37036928.81	44424903.72	53127487.03	60962100.32		
	Normal	26142983.23	29025572.88	32641018.18	34996127.66	6353746.59	8507905.23

Portfolio 2		99%		99.9%		mean loss	standard deviation
		VaR	CVaR	VaR	CVaR		
True	non-Normal	27356170.24	33157462.59	40638318.89	46346896.06		
	Normal	21013165.53	23168845.33	25872574.72	27633790.34	6214230.29	6361445.51
MC 1	non-Normal	18381865.29	24048237.17	30903305.73	37206867.03		
	Normal	10085179.15	11372245.14	12986528.56	14038076.54	1249355.64	3800070.27

MC 2	non-Normal	27252251.25	33358194.22	40821098.41	47848105.13		
	Normal	21070086.56	23232082.90	25943734.73	27710111.04	6227787.68	6381009.93

3. (15 %) Discuss possible strategies for minimizing impacts of sampling and model errors

If you report the in-sample VaR and CVaR to decision-makers in your bank, what consequences for the bank capital requirements it may have?

For portfolio one under the true circumstance over one-year period, there's 1% chance the loss exceeds 37183974.34 and 0.1% chance the loss exceeds 53243955.32. The expected loss of the worst 1% scenarios is 44195846.32 and expected loss of the worst 0.1% scenarios is 59113193.99. Both in-sample VaR and CVaR are lower than the true values for both portfolios, the decisions made base on in-sample VaR and CVaR are dangerous since the risk are underestimate (MC1 will bring more risk to bank than MC2). The required capital of banks is based on VaR, the larger the VaR, the larger the amount of capital required. When the VaR is underestimate, the capital won't be sufficient to cover the loss during bad economic situation.

Can you suggest techniques for minimizing impacts of sampling and model errors?

Accurate computation of VaR and CVaR, require enough history information to include not only normal systemic scenarios. Increase the systemic scenarios to 10000 and idiosyncratic scenarios to 1000. The computational time would be 1000 time faster, compared to generate 10000000 systemic scenarios. Always assume the counterparty losses follow non-Normal distribution. The impacts of sampling and model errors would be smaller is this way.