Deep portfolio optimization with stocks and options

Kristoffer Andersson and Kees Oosterlee[1]

[1]Utrecht University

TopQuants, Nov. 1st, 2023

Agenda

- Stochastic control, optimal portfolio selection
- Our trading strategy with options, stocks and bonds
- Deep learning algorithm
- Numerical examples

Stochastic control problem

- Mean variance (MV) optimal asset optimization is well-known.
- We focus on realistic asset dynamics, as well as objective functions, in line with rational preferences of an investor.
- We explore how options can be used as complement to risky assets and bonds to improve the performance, for general objective functions.
- Represent the strategy with a sequence of neural networks, with as the loss function an empirical objective function.
- Optimization is performed only once for the entire problem. Time-consistent and time-inconsistent problems can be treated similarly.

Adding options and gain flexibility

- A *trader* is allowed to trade in a riskfree bond, $N^{\text{stocks}} \in \mathbb{N}$ stocks, and $N^{\text{options}} \in \mathbb{N}$ options.
- $S = (S_t)_{t\in[0,T]}$ is an $\mathbb{R}^{N^{\text{stocks}}}-$valued time-continuous Markov process on a complete probability space $(\Omega, \mathcal{F}, \mathcal{A})$.
- The bond is $B = (B_t)_{t\in[0,T]}$ and for $i \in \{1, 2, \dots, N^{\text{options}}\}$, $V^i(t, S_t\,; K)$ is an option with S as underlying (single stock or a basket of stocks), at time $t \in [0, T]$, terminating at T, with $K \in \mathbb{R}$ the strike price.
- We set the initial values to unity at $t = 0$, *i.e.*, for $j \in \{1, 2, \dots, N^{\text{stocks}}\}$ and $i \in \{1, 2, \dots, N^{\text{options}}\}$, we set $S_0^j = 1,\ V^i(0, S_0) = 1$ and $B_0 = 1$.

Stochastic control problem

- $\alpha^k = (\alpha_t^k)_{t\in[0,T]}$ is the process describing the amount in stock k and when $k = 0$, the amount in the bond.
- Total wealth of the portfolio stemming from the stock and the bond holdings,

$$x_t = \alpha_t^0 B_t + \sum_{k=1}^{N^{\text{stocks}}} \alpha_t^k S_t^k = A_t^0 + \sum_{k=1}^{N^{\text{stocks}}} A_t^k. \tag{1}$$

- Since the portfolio is self-financing,

$$\alpha_t^0 = \frac{1}{B_{\tau(t)}} \left(x_{\tau(t)} - \sum_{k=1}^{N^{\text{stocks}}} \alpha_{\tau(t)}^k S_{\tau(t)}^k \right), \tag{2}$$

where $\tau(t) = \max_s \{ s \in \mathcal{T} \,|\, s \leq t \}$, *i.e.*, the most recent trading date.
- The return on investment is then given by

$$R_{\text{SB}}(S; \alpha) = x_T - x_0. \tag{3}$$

Stochastic control problem

- Denote the amount of option i in the portfolio by β^i. So,

$$y_t = \sum_{i=1}^{N^{\text{options}}} \beta^i V^i(t, S_t; K^i),$$

- Return on the investment from the static option position:

$$R_{\text{O}}(S; \beta) = y_T - y_0. \tag{4}$$

- Summing up (3) and (4), we obtain the total return

$$R(S; \alpha, \beta) = R_{\text{SB}}(S; \alpha) + R_{\text{O}}(S; \beta). \tag{5}$$

Market frictions

- We add transaction costs as well as a non-bankruptcy constraint and for the trading strategies, given by α and β, we introduce leverage constraints.
- In discrete time, the value of the stocks and bond can then be re-written as

$$x_{t_{n+1}} = x_{t_n} + \alpha^0_{t_n}(B_{t_{n+1}} - B_{t_n}) + \sum_{k=1}^{N^{\text{stocks}}} \alpha^k_{t_n}(S^k_{t_{n+1}} - S^k_{t_n}). \tag{6}$$

- The sum of the transaction costs for stock k :

$$\mathrm{TC}^k = \sum_{n=1}^{N} C\mathrm{e}^{r(T-t_n)}(\alpha^k_{t_n} - \alpha^k_{t_{n-1}})S^k_{t_n}, \tag{7}$$

  where $100 \times C \in \mathbb{R}_+$ is a percentage of the size of the transaction.
- We do not pay transaction costs immediately, but instead at the end of the trading period, with appropriate interest rate.

Constraints:

- No-bankruptcy: When the stocks plus bond value is non-positive, the portfolio is liquidated.

$$x_{t_{n+1}} = x_{t_n} + \mathcal{I}_{\{x>0\}}(x_{t_n})\left(\alpha^0_{t_n}(B_{t_{n+1}} - B_{t_n}) + \sum_{k=1}^{N^{\text{stocks}}} \alpha^k_{t_n}(S^k_{t_{n+1}} - S^k_{t_n})\right), \quad (8)$$

  where $\mathcal{I}_{\{x>0\}}(\cdot)$ is the indicator function.
- No short-selling of stocks, for $t \in [0, T]$ and $1 \geq k \geq N^{\text{stocks}}$, $\alpha^k_t \geq 0$.
- No leverage: We cannot short sell the bond, *i.e.*, for $t \in [0, T]$, $\alpha^0_t \geq 0$.
- No bankruptcy: If $x_{t_n} \leq 0$, all positions are liquidated and for $t \geq t_n$, $x_t = x_{t_n}$.
- Positivity of the bond and the stocks part of the portfolio - $x_0 \geq 0$.

Objective function

- A good quality objective function is able to represent the investor's preferences of how much risk would be acceptable for a certain level of potential profit.
- To penalize downside risk, we maximize the average of the 10% worst outcomes; to encourage upside potential, we maximize the average of the 10% best outcomes. Expected shortfall can be defined by *Value at Risk*,

$$\mathrm{ES}_p^+(R) = \mathbb{E}[R \mid R \leq \mathrm{VaR}_p(R)], \quad \mathrm{ES}_p^-(R) = \mathbb{E}[R \mid R \geq \mathrm{VaR}_p(R)].$$

A typical objective function would then be

$$U = \mathbb{E}[R] - \lambda_1 \mathrm{Var}[R] + \lambda_2 \mathrm{ES}_{p_1}^-(R) + \lambda_3 \mathrm{ES}_{p_2}^+(R), \tag{9}$$

with $\lambda_1, \lambda_2, \lambda_3 \in \mathbb{R}_+$ describing the risk preference and $p_1, p_2 \in (0, 1)$ controlling the sizes of the left and right tails.

Stochastic control problem

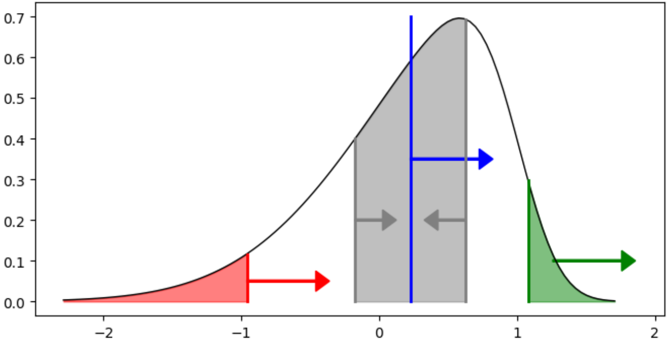


Figure: Example of probability density function for terminal wealth. Red, blue, green represent the lower expected shortfall, mean and higher expected shortfall. Gray area is the mean plus/minus the variance.

Full optimization problem

- So far, α and β are the trading strategies.
- We add the set of strike prices, $K = (K^1, K^2, \ldots, K^{\text{options}})$, as part of the trading strategy, $\pi = (\alpha, \beta, K)$.

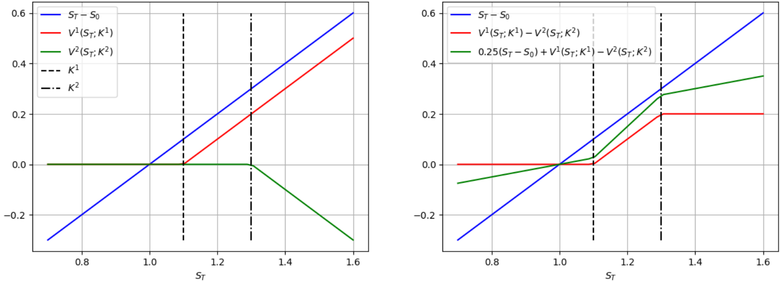


Figure: Returns against stock value at terminal time T. **Left:** Return for investing in a stock, buying one unit of option 1 and selling one unit of option 2. **Right:** Returns for three different combinations of the products;the red line is the classical bull-call spread

Stochastic control problem

- With objective function $U(\pi) = u\big(\mathcal{L}[R(S;\pi)]\big)$, initial wealth $x_0^{\mathrm{IC}} \in \mathbb{R}_+$ and Π the allowed trading strategies (taking all trading constraints into account).

$$\begin{cases} \underset{\pi \in \Pi}{\text{maximize}} = U(\pi), \quad \text{where,} \\ R(S\,;\pi) = R_{\mathrm{SB}}(S\,;\pi) + R_{\mathrm{O}}(S\,;\pi) \\ R_{\mathrm{SB}}(S\,;\pi) = x_T(S;\pi) - x_0(\pi) - \sum_{k=1}^{N^{\mathrm{st}}} \mathrm{TC}^k, \quad R_{\mathrm{O}}(S\,;\pi) = y_T(S\,;\pi) - y_0(\pi), \\ x_T(S;\pi) = x_0 + \sum_{n=0}^{N} \mathcal{I}_{\{x>0\}}\left(x_{t_n}(S;\pi)\right)\left[\alpha_{t_n}^0 (B_{t_{n+1}} - B_{t_n}) + \sum_{k=1}^{N^{\mathrm{st}}} \alpha_{t_n}^k (S_{t_{n+1}} - S_{t_n})\right], \\ x_0 = x_0^{\mathrm{IC}} - y_0(\pi), \quad y_T(S\,;\pi) = \sum_{i=1}^{N^{\mathrm{options}}} \beta^i V^i(T, S_T\,;\, K^i), \\ y_0(\pi) = \sum_{i=1}^{N^{\mathrm{options}}} \beta^i V^i(0, S_0\,;\, K^i). \end{cases} \tag{10}$$

Stochastic control problem

- Given S_0 and assuming known drift and diffusion and jump coefficients μ, σ and J, we employ,

$$\mathrm{d}S_t = \mu(t, S_t)dt + \sigma(t, S_t)dW_t + J(t, S_t)dX_t,$$

  where X_t represents a jump process.
- Let $t_N = T$ and for $0 \leq i \leq N-1$, $t_i < t_{i+1}$, generate $M \in \mathbb{N}_+$ samples of the $N^{\text{stocks}}-$dimensional asset process S. Asset k, realization m, at time $t_n \in \mathcal{T}_N$ is $S^k_{t_n}(m)$, etc.
- We use empirical distributions for $\mathcal{L}[R(S;\pi)]$ in a Monte–Carlo fashion.
- Discrete scheme is approximated by letting deep neural networks represent trading strategies and optimizing with a gradient-decent algorithm.

Neural network approximation

- The trading strategy π is represented by a sequence of neural networks.
- A neural network is a mapping, $\phi(\,\cdot\,;\theta)\colon \mathbb{R}^{\mathfrak{D}^{\text{in}}} \to \mathbb{R}^{\mathfrak{D}^{\text{out}}}$, with θ containing all trainable parameters of the network.
- Number of layers is $\mathfrak{L} \in \mathbb{N}$; for layer ℓ, the number of nodes is $\mathfrak{N}_\ell \in \mathbb{N}$.
- The weight matrix, between $\ell - 1$ and ℓ, is $w_\ell \in \mathbb{R}^{\mathfrak{N}_{\ell-1} \times \mathfrak{N}_\ell}$; the bias $b_\ell \in \mathbb{R}^{\ell}$;
- The (scalar) activation function $a_\ell\colon \mathbb{R} \to \mathbb{R}$ and the vector activation function $\boldsymbol{a}_\ell\colon \mathbb{R}^{\mathfrak{N}_\ell} \to \mathbb{R}^{\mathfrak{N}_\ell}$, which, for $x = (x_1, x_2, \ldots, x_{\mathfrak{N}_\ell})$, is defined by

$$\boldsymbol{a}_\ell(x) = \begin{pmatrix} a_\ell(x_1) \\ \vdots \\ a_\ell(x_{\mathfrak{N}_\ell}) \end{pmatrix};$$

$\Rightarrow$ The output of the network should obey the trading constraints, which are managed by choosing an appropriate activation function in the output layer.

Neural networks representing the trading strategy

- The trading strategy π consists of three parts; *i*) the static amount invested in each option β, *ii*) the static strike prices of the options K, and *iii*) the dynamic amount invested in each stock α.
- β and K are decided at $t = 0$ and with a deterministic initial wealth x_0^{IC}, we have a deterministic representation for β and K.
- α may depend on previous performance, which is affected by randomness through the stock process (a dynamic strategy).
- For the dynamic trading strategy, we use a deep neural network taking the current wealth as input and outputs the stock allocation.
- The *admissible trading strategies* are $\Pi^{\mathrm{NN}} = \{\Pi^{\beta}, \Pi^{K}, \Pi^{\alpha_0}, \Pi^{\alpha_1}, \ldots, \Pi^{\alpha_{N-1}}\}$, where $\Pi^{\alpha_1}, \ldots, \Pi^{\alpha_{N-1}}$, may depend on the stock.

Optimization problem with neural networks

$$
\begin{cases}
\underset{\theta\in\Theta^{\mathrm{NN}}}{\text{maximize}} = U^M(\theta), \quad \text{where } M \textit{ i.i.d.} \text{ random variables are distributed according to,} \\
R(S\,;\theta) = R_{\mathrm{SB}}(S\,;\theta) + R_{\mathrm{O}}(S\,;\theta) \\
R_{\mathrm{SB}}(S\,;\theta) = \hat{x}_{t_N} - \hat{x}_0 - \sum_{k=1}^{N^{\text{stocks}}} \mathrm{TC}^k, \quad R_{\mathrm{O}}(S\,;\theta) = \hat{y}_{t_N} - \hat{y}_0, \\
\hat{x}_{t_N} = \hat{x}_0 + \sum_{n=0}^{N} \mathcal{I}_{\{x>0\}}\left(\hat{x}_{t_n}\right)\left[\hat{\alpha}_n^0(B_{t_{n+1}} - B_{t_n}) + \sum_{k=1}^{N^{\text{stocks}}} \hat{\alpha}_n^k(S_{t_{n+1}} - S_{t_n})\right], \quad \hat{x}_0 = \hat{x}_0^{\mathrm{IC}} - \hat{y}_0 \\
\hat{y}_{t_N} = \sum_{i=1}^{N^{\text{options}}} \hat{\beta}^i V^i(T, S_{t_N}\,;\, \hat{K}^i), \quad \hat{y}_0 = \sum_{i=1}^{N^{\text{options}}} \hat{\beta}^i V^i(0, S_0\,;\, \hat{K}^i), \\
\hat{\alpha}_0^0 = \hat{x}_0 - \sum_{k=1}^{N^{\text{stocks}}} \hat{\alpha}_0^k, \quad (\hat{\alpha}_0^1, \ldots, \hat{\alpha}_0^{N^{\text{stocks}}})^\top = \boldsymbol{a}^{\alpha_0}(\theta^{\alpha_0}), \quad \hat{\beta} = \boldsymbol{a}^{\beta}(\theta^{\beta}), \quad \hat{K} = \boldsymbol{a}^{K}(\theta^{K}), \\
\hat{\alpha}_n^0 = \frac{1}{B_{t_n}}\left(\hat{x}_{t_n} - \sum_{k=1}^{N^{\text{stocks}}} \hat{\alpha}_n^k S_{t_n}^k\right), \quad (\hat{\alpha}_n^1, \ldots, \hat{\alpha}_n^{N^{\text{stocks}}})^\top = \phi(\hat{x}_{t_n}\,;\theta^{\alpha_n}).
\end{cases}
\tag{11}
$$

General neural network settings

- We use a sequence of neural networks, as tools to solve the problem.
- The number of training samples is $M_{\text{train}} = 2^{22}$, the batch size $M_{\text{batch}} = 2^{12}$, the number of epochs $M_{\text{epoch}} = 10$ and the number of layers $\mathfrak{L} = 4$.
- For the interior layers, *i.e.*, $\ell \in \{2, 3\}$, set the number of nodes to $\mathfrak{N}_\ell = 20$ and the activation functions $\mathbf{a}_\ell(\cdot) = \text{ReLU}(\cdot)$.

$\Rightarrow$ $\mathfrak{D}_{\text{input}} = 1$ and $\mathfrak{D}_{\text{output}}$, as well as the activation function in the output layer, depend on the trading constraints and are specified for each specific problem.

- Initial learning rate is 0.01. After two batches, it decreases by a factor $\exp(-0.5)$ for each new batch.

Classical continuous mean-variance optimization

- Classical MV problem: asset process is geometric Brownian motion.
- Trading is carried out without transaction costs, *i.e.*, setting $C = 0$.
- There are no constraints and trading in the options is not allowed.
- The objective function is given by

$$U(\theta) = E[x_T] - \lambda \mathrm{Var}[x_T].$$

  where $\lambda > 0$ controls the risk aversion.
- Closed-form expression for the optimal allocation as well as an optimal mean and variance of the terminal wealth. $T = 2, N = 20, r = 0.06, \lambda = 1.104$

$$a = \begin{pmatrix} 0.08 \\ 0.07 \\ 0.06 \\ 0.05 \\ 0.04 \end{pmatrix}, \sigma = \begin{pmatrix} 0.23 & 0.05 & -0.05 & 0.05 & 0.05 \\ 0.05 & 0.215 & 0.05 & 0.05 & 0.05 \\ -0.05 & 0.05 & 0.2 & 0.05 & 0.05 \\ 0.05 & 0.05 & 0.05 & 0.185 & 0.05 \\ 0.05 & 0.05 & 0.05 & 0.05 & 0.17 \end{pmatrix}. \tag{12}$$

Stochastic control problem

- The optimal value of the objective function is approximately 1.1637.

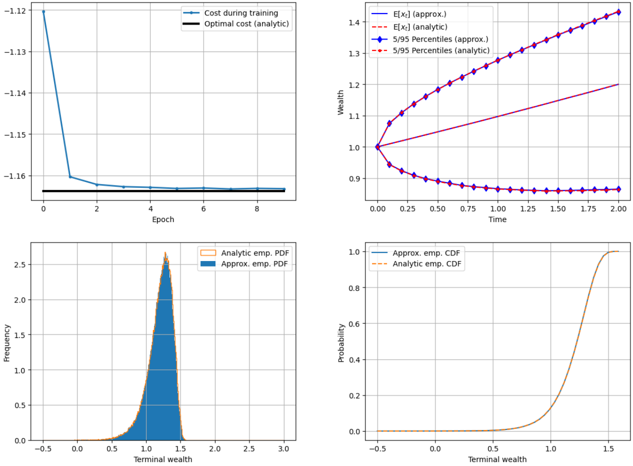


Figure: **Upper:** Convergence of the loss to the analytic counterpart with respect to the number of training epochs. Comparison with reference solution. **Lower:** Comparison o the empirical pdfs and reference. Comparison of the empirical CDFs and the reference.

Beyond MV, with market frictions and jumps

- Consider the full generality of the asset model, as well as transaction costs, no bankruptcy constraint and trading in European call and put options.
- The parameter values are reused and $\lambda_J = 0.05$, $\mu_J = (0, \ldots, 0)^\top$, $\Sigma_J = \text{diag}(0.2, \ldots, 0.2)$, $\text{NB} = 1$ and $C = 0.005$.
- An interpretation of C is as a penalizing term for too heavy reallocation (which is something that for instance pension funds want to avoid).

$$U(\theta) = \mathbb{E}[R] - \lambda_1 \text{Var}[R] + \lambda_2 \text{ES}^{-}_{p_1}(R) + \lambda_3 \text{ES}^{+}_{p_2}(R).$$

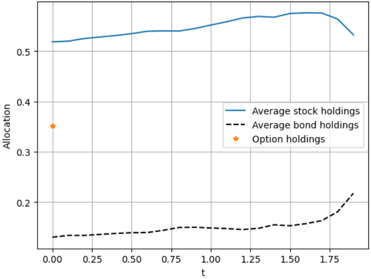
- $p_1 = 0.01$ i.e., we penalize low values of the expected return of the worst 1% performance of the portfolio. For the upper tail, we maximize the expected return of the 5% best outcomes, $p_2 = 0.95$.
- The weights are set to $\lambda_1 = 0.552$, $\lambda_2 = 0.276$ and $\lambda_3 = 0.110$.

Problem specific neural network settings

- We set $\beta_{\text{max}} = 1$ (the maximum amount of allocation into the options is 100% of the initial wealth).
- We use a slight modification of the activation function for this.
- Then, the option allocation range is $[0, \beta^{\text{max}}]$, while keeping the sum of the allocations into each option to $[0, \beta^{\text{max}}]$.
- For the strike prices, we use $K^{\text{low}} = (0.75, 0.75 \ldots, 0.75)^\top$ and $K^{\text{high}} = (1.25, 1.25 \ldots, 1.25)^\top$, *i.e.,* setting the range for strike prices between 75% to 125% of the stock price at the initial time.
- We set $\alpha_n^{\text{low}} = -2x_n$ and $\alpha_n^{\text{high}} = 2x_n$ implying that we can allocate into each stock between -200% and 200% of the total value of the stocks and bond.

Evaluation of the results

- The algorithm returns a dynamic strategy for the bond and stocks, the static strategies for the allocations into the options and a strike for each option.



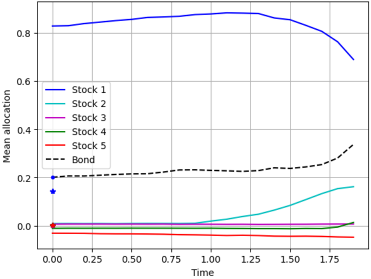


Figure: **Left:** Average allocation to stocks, bond and options over time. **Right:** Averag allocation to stocks, bond and options over time for each stock. Asterisks and bullets represent call and put option holdings, respectively.

Stochastic control problem

- The strike prices are optimized by the neural networks to 0.75 for all call options and 1.25 for all put options, *i.e.*, deep in the money.
- For the best performing outcomes, the main option contribution comes from call options; for the worst performing outcomes from put options.

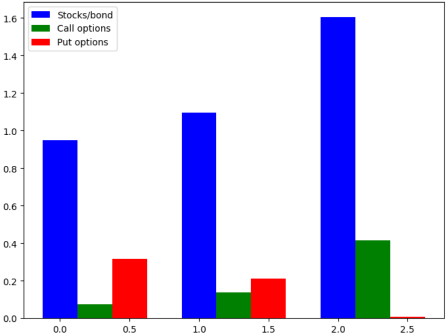


Figure: Contribution to the portfolios for terminal wealth less than 1.03 (33% of the outcomes), between 1.03 and 1.12 (41%), and above 1.12 (26%).

Stochastic control problem

- With options, we observe *i*) a thinner left tail, *ii*) a higher density around the expected terminal wealth, and *iii*) a fatter right tail.
- The first and last items are beneficial since the objective function aims to prevent large losses (by the lower expected shortfall term) and encourages large gains (by the upper expected shortfall term).
- Regarding a comparison with the MV-strategy, in contrast to our strategies, we encounter a fatter left than the right tail, which is non-desirable.

Stochastic control problem

- For all measures, but the variance, the portfolio with options performs best.
- By the strategy with options, transaction costs decrease by $> 60\%$ compared to the strategy without options and $> 90\%$ compared to the MV-strategy.
- This is beneficial since less aggressive re-allocation is desirable for a fund.

	$\mathbb{E}[R]$	$\mathrm{Var}[R]$	$\mathrm{ES}^{+}_{p_1}(R)$	$\mathrm{ES}^{-}_{p_2}(R)$	$U(\theta^*)$	Tr. cost
With options	1.146	0.081	0.971	2.18	1.61	0.386%
Without options	1.140	0.045	0.931	1.93	1.58	1.01%
MV strategy	1.146	0.077	-0.208	1.48	1.21	3.98%

Table: For the MV-strategy, λ is set to make the mean coincide with the mean obtaine from the strategy with options. The trading cost is a percentage of the initial wealth.

Stochastic control problem

- The market does *not* behave exactly as the model.
- Test the algorithm's robustness for model miss-specification, applying the strategies with higher and lower volatility of the underlying asset process.

$\Rightarrow$ In the high volatility case, we multiply σ by two and in the low volatility case we divide σ by two.

Stochastic control problem

- Most notable is the lower expected shortfall which expresses a loss of 172.8% and 98.4% and the trading costs at 10.1% and 2.74% for the higher and lower volatilities, respectively.
- Options in the portfolio are beneficial when the volatility increases and less beneficial when the volatility decreases.
- Variance is larger with options, due to the fatter right tail of the distribution.

	$\mathbb{E}[R]$	$\text{Var}[R]$	$\text{ES}^+_{p_1}(R)$	$\text{ES}^-_{p_2}(R)$	$U(\theta^*)$	Trading co
Evaluation with increased volatility for the underlying assets ($\sigma \mapsto 2 \times \sigma$).						
With options	1.318	0.660	0.763	3.268	1.540	0.838%
Without options	1.350	0.175	0.734	2.635	1.532	1.23%
MV strategy	1.081	0.674	-0.728	1.460	0.668	10.1%
Evaluation with decreased volatility for the underlying assets ($\sigma \mapsto 0.5 \times \sigma$).						
With options	1.074	0.0262	0.969	1.620	1.506	0.261%
Without options	1.143	0.0160	0.957	1.548	1.569	0.636%
MV strategy	1.163	0.0569	0.0160	1.487	1.301	2.74%

Table: Comparison of three strategies. For the MV-strategy, λ is set to make the mear coincide with the mean obtained from the strategy with options. The trading cost, as percentage of initial wealth reflects the volatility of portfolio re-allocations.

Conclusions

- The choice of objective function should reflect the true incentives of a rational trader
- Adding options makes shaping of the distribution of the terminal wealth more flexible due to the asymmetric distribution of option returns.
- Options significantly reduce re-allocation and in turn the trading cost;
- A sequence of neural networks produces high quality allocation strategies in high dimensions (many assets, options and strike prices for each option).
- Extension to trading options in a dynamic setting is straightforward if we have access to an efficient option valuation along stochastic asset trajectories.

Welcome to ICCF24, in CWI Amsterdam, 2-5 April 2024

International Conference on Computational Finance —ICCF24 — organised in CWI, Amsterdam, 2-5 April 2024

Home Conference Submission Registration Program Awards

We would like to welcome you to ICCF24, in Amsterdam, next year April!

The international Conference on Computational Finance (ICCF)
is the leading conference in the area of mathematical modeling,
numerical methods and scientific computing for financial applications.

https://www.iccf24.nl