swissQuant
Mastering Complexity

Mathematical Challenge November 2017

MILP formulation for Value at Risk constraints

References

- [1] Artzner, P., Delbaen, F., Eber, J. M., & Heath, D. (1999). Coherent measures of risk. *Mathematical finance*, *9*(3), 203-228.
- [2] Benati, S., & Rizzi, R. (2007). A mixed integer linear programming formulation of the optimal mean/value-at-risk portfolio problem. *European Journal of Operational Research*, *176*(1), 423-434.
- [3] Feng, M., Wächter, A., & Staum, J. (2015). Practical algorithms for value-at-risk portfolio optimization problems. *Quantitative Finance Letters*, *3*(1), 1-9.
- [4] Jorion, P. (1997). *Value at risk: the new benchmark for controlling market risk*. Irwin Professional Pub.
- [5] Rockafellar, R. T., & Uryasev, S. (2000). Optimization of conditional value-at-risk. *Journal of risk*, *2*, 21-42.

Description

In Markowitz's mean-variance approach, the risk exposure is measured by the variance of the selected portfolio. Markowitz himself noticed that, since this measure penalizes losses as well as profits, a risk measure accounting only for losses would better describe natural behaviors of rational investors. Furthermore, when distribution comprises asymmetries and tail dependencies, variance-based risk measures can lead to terribly deceiving conclusions.

A lot of work has been invested in order to find more reliable risk measures. Value-at-Risk (VaR) and Conditional-Value-at-Risk (CVaR) have been identified as two good candidates. VaR as introduced in [4], is defined as an upper percentile of the loss distribution. CVaR is the expected loss under the condition that it exceeds VaR, see [5].

Based on theoretical and computational reasons, CVaR could be preferred over VaR as it has been proved to be a coherent risk measure, see [1], and it has been shown to be expressed as an efficient linear program, as first described in [5]. Nevertheless, VaR played an important role in practical applications, academic research, and capital adequacy requirements. Various algorithms have been designed to solve VaR problems, both heuristically and optimally.

The scope of this challenge is to review MILP (Mixed-Integer Linear Program) formulations for modelling VaR constraints and thus presenting an optimal approach for solving VaR problems. For any given probability threshold α in $(0,1)$ and a portfolio strategy *x* in *X*, VaR_α is defined as the α-quantile of the loss distribution corresponding to x. Concretely, if our strategy *x* has losses express by the random variable $L(x)$, $VaR_\alpha(x) = \min\{\zeta \; in \; \mathbb{R} \mid P(L(x) \leq \zeta) \geq \alpha\}$. We refer to [5], for a formal description of VaR.

In this Challenge, we consider the following problem. Let S be the universe of assets and let R be the vector representing their expected returns. For a given upper bound UB, our goal is to find a solution x that maximizes the expected return $R\,x$ and for which $VaR_{\alpha}(x)$ is smaller or equal than UB, which reads as

$$\begin{aligned}\max\quad & R\,x\\ s.t.\quad & VaR_{\alpha}(x) \leq UB\\ & \sum_{s\ in\ S} x_s = 1\\ & x_s\ in\ [0,1] \qquad \forall s\ in\ S.\end{aligned}$$

In order to solve the latter problem, we need a description of $L(x)$, which is not always available or easy to handle. Instead, assume losses are linear, hence $L(x) = L\,x$ for a random variable L, and L can be described by a Monte Carlo simulation of samples L_j in $\mathbb{R}^S$, for j in 1,…,m. As suggested by [2], requiring $VaR_{\alpha}(x) \leq UB$ is equivalent to impose that at most $\lfloor (1-\alpha)m \rfloor$ many samples correspond to a loss that can exceed UB. Thus, we can rewrite the latter problem as follows.

$$\begin{aligned}\max\quad & R\,x\\ s.t.\quad & if\ b_j = 0 \rightarrow L_j\,x \leq UB\\ & \sum_{j=1}^{m} b_j \leq (1-\alpha)m\\ & \sum_{s\ in\ S} x_s = 1\\ & b_j\ in\ \{0,1\} \qquad \forall j\ in\ \{1,\dots,m\}\\ & x_s\ in\ [0,1] \qquad \forall s\ in\ S.\end{aligned}$$

In the problem above, the VaR constraint has been substituted with the set of indicator constraints $if\ b_j = 0 \rightarrow L_j\,x \leq UB$, which describes the logic that the constraint is considered only when $b_j = 0$ is satisfied. Thus, since we have at most $\lfloor (1-\alpha)m \rfloor$ binary variables b_j equal to 1, there are at least $\alpha\,m$ many constraints $L_j\,x \leq UB$ that are valid. The problem above can finally be reduce as a Mixed-Integer Linear Program (MILP) in the following way.

$$\begin{aligned}\max\quad & R\,x\\ s.t.\quad & L_j\,x \leq UB + M_j\,b_j\\ & \sum_{j=1}^{m} b_j \leq (1-\alpha)m\\ & \sum_{s\ in\ S} x_s = 1\\ & b_j\ in\ \{0,1\} \qquad \forall j\ in\ \{1,\dots,m\}\\ & x_s\ in\ [0,1] \qquad \forall s\ in\ S,\end{aligned}$$

for M_j sufficiently large enough. Concretely, chose each M_j to be equal to $\max\,\{L_j\,x - UB \mid j\ in\ \{1,\dots,m\},\ x\ in\ [0,1]^S\}$. When $b_j = 0,\ L_j\,x \leq UB$ is enforced, otherwise if $b_j = 1,\ L_j\,x \leq UB + M_j$ is not bounding $L_j\,x$. Notice that M_j enters in the model as a parameter and it is not a variable.

These parameters, whose contribution is to allow indicator constraints to be modeled as mixed integer linear constraints, are technically called big-M. Without going into details, when MILP solvers handle big-M constraints, the smaller are these big-M the more likely is the algorithm to run faster.

A better choice for our model is given by $\mathrm{M_j} = \max\{L_{s,j} \mid s \ in\ S\} - UB$ as $\max\{L_{s,j} \mid s\ in\ S\} = \max\{L_j\ x \mid x\ in\ [0,1]^S\}$ and each M_j can be adapted to each constraints $j\ in\ \{1, \dots, m\}$ separately. Another choice for M_j is presented in [3]. The authors analyze a more general framework in which VaR appears in the objective, instead of considering it as a constraint. With respect to our problem, they prove that $\mathrm{M_j}$ can be chosen to be the $(\lfloor(1-\alpha)\ m\rfloor + 1)$th smallest element in the vector $\left(\max\{L_{s,j} - L_{s,i} \mid s\ in\ S\}\right)_{i\ in\ \{1,\dots,m\}}$. Understanding the reason why this holds goes beyond the scope of this challenge, but any interested reader can find it in [3].

Questions:

- [Q1] In this challenge we presented two different good options for the selection of $\mathrm{M_j}$. Which is the best one? Can you derive even better ones?
- [Q2] Assume we want to solve our problem with the difference that we consider two different VaR constraints, say $VaR_{\alpha_1}(x) \leq UB_1$ and $VaR_{\alpha_2}(x) \leq UB_2$. An easy way to do that is to handle the way we explain above the two constraints separately. This approach brings twice as many binary variables and twice as many big-M constraints. Can you model it by using twice as many binary variables and only a constant number of additional constraints? Can you even reduce the number of binary variables needed?

We look forward to your opinions and insights.

Best Regards,

swissQuant Group Leadership Team