

Projet

*(A réaliser d'une manière individuelle)
(A rendre au plus tard le 13 février 2015)*

Ce projet comporte deux parties, la première concerne l'optimisation de la CVaR. La deuxième est liée au calcul de l'index fund.

I)

Considérons un ensemble de portefeuilles qui doivent vérifier les contraintes de Markowitz ($\sum x_i = 1$, $\sum \mu_i x_i \geq R$). La fonction perte est donnée par :

$$f(x,y) = (b-y)^T x$$

où x est le portefeuille, b le vecteur estimé des prix des actifs et y un vecteur de probabilités.

1) Ecrire le modèle qui permet de minimiser la CVaR pour une probabilité donnée α . (On donne le modèle général avec un nombre d'actifs n , un nombre de scénarios S , un vecteur rendement μ et un rendement minimum R).

2) Résoudre le modèle pour des données aléatoires n , S , μ , R , α et y . On donne à chaque fois la CVaR et la VaR. On fait varier les paramètres n , S , μ , R , α et y . On considère des instances de grandes tailles. On analysera la variation du temps de calcul en fonction de n , S , R et α . On donnera à chaque fois la Var.

3) Considérons le modèle de Markowitz maximisant le rendement du portefeuille et assurant une CVaR ne dépassant pas un certain seuil.

Ecrire le modèle correspondant et le résoudre pour certaines données aléatoires. On fera le même type d'analyse que dans 2).

II)

L'objet de cette partie est de mettre en œuvre la méthode dite « Relaxation Lagrangienne » vue en TD à travers l'exercice ci-dessous, pour la construction d'un index fund.

Considérons le modèle P suivant permettant de construire un index fund :

$$\begin{aligned} \text{Max } Z &= \sum_i \sum_j p_{ij} x_{ij} \\ \text{Sc} \\ \sum_j y_j &= q \\ \sum_j x_{ij} &= 1, i=1, \dots, n \\ x_{ij} &\leq y_j, i=1, \dots, n; j=1, \dots, n \\ x_{ij}, y_j &= 0 \text{ ou } 1, i=1, \dots, n; j=1, \dots, n \end{aligned}$$

Pour $u=(u_1, \dots, u_n)$, considérons le programme P_u suivant :

$$\begin{aligned} \text{Max } L(u) &= \sum_i \sum_j \rho_{ij} x_{ij} + \sum_i u_i (1 - \sum_j x_{ij}), \\ \text{Sc} \\ \sum_j y_j &= q \\ x_{ij} &\leq y_j, \quad i=1, \dots, n; \quad j=1, \dots, n \\ x_{ij}, y_j &= 0 \text{ ou } 1, \quad i=1, \dots, n; \quad j=1, \dots, n \end{aligned}$$

- 1) Soient Z^* et $L^*(u)$ les valeurs optimales des programmes P et P_u respectivement. Montrer que $Z^* \leq L^*(u)$.
- 2) Montrer que la fonction objectif du programme P_u ci-dessus peut être écrite sous la forme :

$$\text{Max } L(u) = \sum_i \sum_j (\rho_{ij} - u_i) x_{ij} + \sum_i u_i$$

Soient

$$(\rho_{ij} - u_i)^+ = \rho_{ij} - u_i \text{ si } \rho_{ij} - u_i > 0 \text{ et } (\rho_{ij} - u_i)^+ = 0 \text{ sinon et}$$

$$C_j = \sum_i (\rho_{ij} - u_i)^+$$

- 3) Montrer que le programme P_u est équivalent au programme Q_u suivant :

$$\begin{aligned} \text{Max } L(u) &= \sum_j C_j y_j + \sum_i u_i \\ \text{Sc} \\ \sum_j y_j &= q \\ y_j &= 0 \text{ ou } 1, \quad j=1, \dots, n \end{aligned}$$

- 4) Montrer que dans une solution optimale de Q_u , $y_j=1$ pour les q plus grandes valeurs de C_j et les autres y_j sont nuls.
- 5) En déduire une méthode approchée de résolution pour le problème de la construction de l'index fund.

La réponse aux questions n'est pas demandée.

Appliquer cette méthode pour des données aléatoire et des données réelles si possible (CAC 40, ...). On peut générer aléatoirement les similarités entre les actifs. Faites varier q la taille de l'index. Donner des solutions le plus proche possible de l'optimum. Analyser la variation du temps de calcul en fonction de la taille de la population et de q . On peut résoudre le problème d'une manière exacte pour des petites instances en utilisant un logiciel comme Cplex ou GLPK.