Finance Quantitative

Méthode de Monte-Carlo

Patrick Hénaff

Version: 07 févr. 2024

```
library(anytime)
library(xts)
library(xtable)
library(timeDate)
library(fOptions)
library(fExoticOptions)
library(ggplot2)
library(lubridate)
library(NFCP)
#library(LSMRealOptions)
```

L'objectif de ce TP est de valoriser par simulation des options qui permettent plusieurs exercices durant la vie de l'option. Quelques exemples de ce type d'option:

- Le tarif "Tempo" d'EDF, qui permet d'appliquer un tarif élevé 22 jours de l'année, en période de forte demande, en échange d'un tarif avantageux le reste du temps.
- Les options "swing" ou "take-or-pay" utilisées dans les marchés de matières premières. Dans ce cas, l'option donne le droit de moduler la quantité de matières premières achetée, en fonction des conditions du moment. Dans le cas du gaz naturel, par exemple, les parties s'étant accordés sur un volume de gaz à livrer chaque jour, contracte de plus une "swing" option qui donne le droit à l'acheteur de demander livraison, à un prix déterminé à l'avance, d'une quantité supplémentaire de gaz. Sur une période d'un mois, l'acheteur peut disposer d'une dizaine de droits quotidiens de cette nature.

On s'intéresse ici aux options de type "swing". @Jaillet2004 propose un algorithme de valorisation dans le cadre d'un modèle à un facteur. Un tel modèle capture la dynamique du sous-jascent d'une manière imparfaite, mais la solution proposée ne se généralise pas à des modèles multifactoriels. Dans ce TP, nous explorons la possibilité de valoriser la même option par simulation à l'aide de l'algorithme de Longstaff-Schwartz [@].

Simulation

Pour les besoins de ce TP, on adopte un algorithme de simulation du package NFCP. On simulera le processus avec un pas d'une semaine sur une période d'un an.

Cette fonction génère un scenario comprenant:

- la simulation des deux facteurs du modèle de Schwartz
- la simulation de contrats à terme de différentes maturités
- la simulation du prix spot, qui est celui qui nous intéresse.

L'algorithme de Longstaff-Schwartz

Dans un premier temps, on se propose de valoriser une option Américaine de maturité 1 an à l'aide de l'algorithme de Longstaff-Schwartz. On utilisera un polynome d'ordre 2 du spot lui-même comme variables indépendantes dans l'équation de régression. Une version plus sophistiquée du modèle utiliserait aussi les contrats à terme et les facteurs.

On suggère d'organiser les calculs de la façon suivante, en suivant pas à pas la démarche de l'article de référence [@L]:

- une fonction calculant la valeur d'exercice, pour un vecteur de spot
- une fonction calculant les variables indépendantes de la regression, pour un vecteur de spot
- une fonction calculant la "continuation value" au temps t pour tous les scénarios dans l'argent, étant donné:
  - la "continuation value" actualisée de l'étape précédente
  - les variables d'état: dans notre cas, il s'agit seulement du vecteur de spot
  - la fonction calculant la valeur d'exercice

Avec ces fonctions, on calcule la récursion inverse en enregistrant dans un vecteur de dimension N la date d'exercice éventuel le long de ce scénario.

Dans la version élémentaire de l'algorithme, la valeur de l'option Américaine est calculée comme l'espérance de valeur actualisée des exercices.

Cependant, dans une mise en oeuvre plus rigoureuse, la récursion inverse ne sert uniquement qu'à déterminer la règle d'exercice optimale. La valeur de l'option est calculée en faisant une nouvelle simulation et en appliquant la règle d'exercice optimale à cette simulation. Mettre en oeuvre cette amélioration.

Inspirez-vous de toutes les ressources disponibles en "open source" pour mettre en oeuvre cet algorithme, mais écrivez votre propre code, afin de vous préparer à la question suivante.

Vérifiez votre algorithme en comparant votre évaluation d'options Américaines sous un processus brownien géométrique aux résultats obtenus avec un arbre binomial dans le package fOptions.

Option "swing" avec N exercices

On considère ici une option "swing" qui donne N droits d'acheter une unité de sous-jacent au prix K durant la vie de l'option. Les droits non utilisés à maturité sont perdus. On ne peut exercer qu'un droit par période.

Bornes inférieure et supérieure

On commence par calculer des bornes sup et inf du prix de l'option.

Une borne supérieure est calculée en supposant que l'on a une connaissance parfaite de l'avenir. Les scénarios sont aléatoires, mais on sait sur quel scénario on se trouve, et on connaît la trajectoire complète du scénario. Il suffit donc de déterminer pour chaque scénario les N exercices les plus profitables.

Une borne inférieure est obtenue par une heuristique, en testant des règles de décision définies *a priori*. Se rappeler bien sûr que ces règles ne doivent utiliser que des informations passées.

Valorisation

Pour valoriser une option "swing" avec N exercices, on introduit une seconde variable d'état qui est le nombre d'exercices restants, qui prend les valeurs entières $0, \ldots, N$. A chaque étape de temps dans la récursion inverse, le calcul de la "continuation value" doit donc se faire pour chaque valeur de la variable d'état "nombre d'exercices restants".

La valeur de continuation au temps t, sachant qu'il reste k options à exercer est donc calculée en comparant:

- l'exercice immédiat d'un droit + la valeur actualisée de continuation en $t+1$ avec $k-1$ droits restants
- la valeur actualisée de continuation en $t+1$ avec k droits restants.

Comme dans la partie précédente, on commence par générer M scenarios du processus spot.

Une manière de mettre en oeuvre l'algorithme peut se résumer ainsi:

Soit N le nombre de droits, T le nombre de pas de temps et M le nombre de scenarios. On va construire un tableau OD de dimension $N \times T \times M$ avec $OD(i,t,k) = 1$ s'il est optimal d'exercer un droit au temps t, sachant que l'on dispose de i droits et que l'on est dans le scenario k, et $OD(i,t,k) = 0$ sinon.

On utilisera aussi un tableau OV de dimension $N \times M$ avec $OV(i,k)$ la valeur de l'option "swing" sachant qu'il reste i droits et que l'on est dans le scenario k. Ce tableau sera mis à jour à chaque itération de la récursion inverse.

La récursion inverse suit ensuite la même logique que précédemment:

1. Calculer la valeur du tableau OV à maturité de l'option.

2. Pour chaque date d'exercice $(T-1), \dots, 1$:

2.1 Calculer la valeur de continuation actualisée par régression

2.2 Déterminer la décision optimale à cette étape de temps, pour chaque niveau de droits restants

2.3 Mettre à jour le tableau OV

2.4 Enregistrer la décision optimale dans le tableau OD.

L'étape finale consiste à utiliser le tableau OD pour calculer le cash flow associé à chaque scenario, et en déduire l'espérance et l'écart type de la valeur de l'option.

Pour vérifier votre algorithme, considérez deux cas particuliers:

- avec $N = 1$, l'option "swing" est une option Américaine
- si N est égal aux nombre de périodes, l'option "swing" est un portefeuille d'options Européennes.