01/03/2018

Pricing Auto-callable

Produit structuré

Professeur : Philippe DUMONT

UNIVERSITE PARIS DAUPHINE

Magistère Banque Finance Assurance

Thanh Tu NGUYEN

Nghia Quan HOANG

Yuyan GAN

Paul LEGREZ

Table des matières

I. Introduction

Depuis leur introduction en 2003, les produits structurés auto-callables ont été activement négociés en Europe et aux États-Unis depuis des années et ont maintenant acquis une place de plus en plus prédominante parmi les dérivés de flux.

Bien que leur effet théorique en Europe et aux États-Unis soit dilué en raison d'une plus grande activité des hedge funds et des fonds de couverture, le cas de l'Asie est la preuve que les produits structurés jouent un rôle essentiel dans les marchés de volatilité.

Par conséquent, notre projet se concentre sur le pricing d'un Auto-call spécifique. Bien qu'il existe toutes sortes d'auto-call, dans notre projet, nous nous concentrerons sur le produit suivant à titre d'exemple : the Phoenix income AutoCall 8.8% - uK88.

II. Pricer B&S Monte Carlo

Un produit Auto-callable est un produit structuré doté d'une fonction d'appel automatique à des dates prédéfinies. Dans le cadre de notre projet, nous analyserons donc le Phoenix income AutoCall 8.8% - uK88.

II.1 Description générale du produit :

- Type : Produit Dérivé Titrisé adossé à un indice boursier, le FTSE 100
- Date de Valorisation Initiale : 04/07/2012
- Date de Valorisation Finale : 05/07/2018
- Maturité maximale : 6 ans
- Coupon : 8.80% du capital investi à fréquence annuelle, sous réserve de certaines conditions
- Dates de Valorisation Intermédiaires : 05/07 de chaque année, ou le premier jour ouvré suivant cette date, de 2013 à 2017 incluses
- Date de paiement du coupon : 19/07 de chaque année, ou le premier jour ouvré suivant cette date, de 2013 à 2018 incluses (dans notre pricing, pour simplifier, on suppose que le coupon soit payé à la même date que la valorisation)
- Niveau Initial de l'indice boursier : cours de clôture du FTSE 100 à la Date de Valorisation Initiale.

II.2 Payoff :

- A chaque Date de paiement du coupon, dans le cas où le cours de clôture du FTSE 100 est supérieur ou égal aux 80% de son Niveau Initial, un coupon de 8.80% du capital investi est versé.

- Aucun coupon n'est versé à ces dates dans le cas contraire, mais les coupons non versés peuvent être récupérés ultérieurement comme suit.
- Le coupon est mémorable : Pour toute année durant laquelle aucun coupon n'a été versé, **le coupon non versé peut être récupéré à toute Date de Valorisation Intermédiaire ultérieure** à laquelle le FTSE 100 clôt à un cours supérieur ou égal aux 80% de son Niveau Initial.
- A compter de la seconde Date de Valorisation Intermédiaire, si le FTSE 100 clôt à un cours supérieur ou égal aux 110% de son Niveau Initial, le produit arrive à expiration, et ce irréversiblement.
- Si le produit n'a expiré à aucune des Dates de Valorisation Intermédiaire entre la 2ème et la 5ème année, le capital initial est intégralement protégé à maturité sous réserve que le cours de clôture du FTSE 100 à la Date de Valorisation Finale soit supérieur ou égal aux 60% de son Niveau Initial. Dans le cas contraire, le capital n'est pas protégé.

II.3 Implémentation dans Python

Concernant le facteur d'escompte, nous avons créé une fonction pour faciliter l'actualisation :

$$Discount\ factor\ (t1, t2) = \exp(-r * \frac{t2-t1}{365})$$

Le prix de sous-jacent suit le processus stochastique suivant :

$$S_T = S_t * \exp\left(\left(r - \frac{v^2}{2}\right) * (T-t)/365 + v * \sqrt{(T-t)/365} * N(0,1)\right)$$

A chaque date d'anniversaire, le payoff est exprimé en fonction du prix du sous-jacent à partir de la date initiale et ce jusqu'à cette date d'anniversaire.
Valeur de l'autocall à la date initiale = $\sum(payoff\ * discount\ factor)$
On met en œuvre 1 000 000 de simulations et on prend la moyenne pour avoir le prix de l'autocall.

II.4 Résultat

Avec les paramètres r= 4%, v= 20% la valeur de l'auto-call est = 1050 donc > valeur notionnelle (1000). Cela revient à dire que même si le sous-jacent finit hors de la monnaie du put, le capital investi n'est pas entièrement assuré.

III. Modèle de taux stochastique

III.1 L'approche théorique

Pour ajouter le modèle de taux stochastique, nous utilisons le modèle de Cox-Ingersoll-Ross (1985).

$$dr_t = k_r(\theta_r - r_t)dt + \sigma_r\sqrt{r_t}dZ_t$$

Le prix de sous-jacent suit toujours le processus stochastique suivant :

$$dS_t = r_t * d_t + \sigma_S * \sqrt{d_t} * N(0,1)$$

Le discount factor est comme suit :

$$df_T = exp\left(\int_0^T -r_t\, dt\right)$$

III.2 Implémentation en Python

Pour implémenter ce modèle dans notre pricing, nous prenons dt = 1/252 (= 1 working day).

On suppose ici qu'il n'y ait pas de corrélation entre le taux d'intérêt et le prix de sous-jacent (Si on veut faire un pricer avec la corrélation, on doit simplement ajouter la matrix de corrélation et utiliser la méthode de Cholesky).

Pour chaque date, nous avons :

$$S_t = S_{t-1} * \exp\left(\left(r_{t-1} - \frac{v^2}{2}\right) * dt + v * \sqrt{dt} * N(0,1)\right)$$

Concernant le taux d'intérêt, le modèle inclut la racine carrée de Rt, donc, il faut faire attention car dès lors que le taux d'intérêt est négatif, cela ne fonctionne pas. Alors, on met (Truncation) :

$$r_t = Max(r_{t-1} + k_r(\theta_r - r_{t-1})dt + \sigma_r\sqrt{r_{t-1}} * N(0,1); 0)$$

Pour ce qui est du facteur d'actualisation :

$$DF_t = DF_{t-1} * exp(-r_{t-1} * dt)$$

III.3 Résultat

Avec les paramètres de modèle de Cox-Ingersoll-Ross (1985), en mettant le taux d'intérêt court terme (r0) et le taux d'intérêt long terme (theta_r) à 4% et la vol a 20%, nous avons trouvé un prix d'autocall = 1050, ce qui n'est pas très différent par rapport prix que le modèle B&S simple nous a donné (avec les même inputs). Cela montre que le fait qu'on a ajouté le modèle de taux stochastique n'a pas beaucoup d'impact sur le prix d'autocall.

IV. Pricer EDP

IV.1 Méthodes de différence explicite

Les méthodes de différence finie sont utilisées pour pricer des produits dérivés en résolvant l'équation différentielle conjointement à la condition liée au prix initial de l'actif et aux conditions liées aux valeurs limites, i.e. les payoffs que le produit dérivé doit remplir. L'équation différentielle est convertie en un système d'équations de différence qui sont résolues de manière itérative. Prenons l'EDP de Black & Scholes :

$$\frac{\partial f}{\partial t} + (r - q) * S\ \frac{\partial f}{\partial S} + \ \frac{1}{2}\ \sigma^2 S^2 * \frac{\partial^2 f}{\partial S^2} = rf$$

Elle est sujette à la condition liée au payoff suivante à la maturité :

$$f(S_T, T) = \begin{cases} Notional + Coupon\ si\ \frac{S_T}{S_0} \geq 80\% \\ Notional\ si\ 80\% > \frac{S_T}{S_0} > 60\% \\ Notional * \ \frac{S_T}{S_0}\ si\ \ \frac{S_T}{S_0} \leq 60\% \end{cases} \qquad (4.1)$$

De manière similaire aux arbres trinomiaux, lorsque l'on implémente des méthodes de différence finie, on divise l'espace et le temps en des intervalles discrets Δt et Δx, ce qui génère une grille. On ajoute les conditions de valeurs limites à la grille, ce qui détermine le prix de l'option tel une fonction du prix de l'actif pour des valeurs élevées comme pour des valeurs faibles de telle sorte que δf/δS = 0 pour S élevé et que δf/δS = 1 pour S faible.

On simplifie l'EDP de Black & Scholes en remplaçant l'EDP par des différences finies. Ainsi, on peut discrétiser l'EDP pour donner lieu à un système de différences finies numériques. Tout d'abord, on simplifie l'EDP ; en prenant x = ln(S) de sorte que :

$$\frac{\partial f}{\partial t} + \mu \frac{\partial f}{\partial x} + \ \frac{1}{2}\ \sigma^2 \frac{\partial^2 f}{\partial S^2} = rf$$

Avec μ = r – q. Afin d'éliminer le terme rf du côté gauche, prenons une nouvelle fonction u(x, t) telle que :

$$u(x, t) = \ e^{r(T-t)} f(e^x, t)$$

Le terme u est le prix forward de l'option f et satisfait l'EDP :

$$\frac{1}{2}\ \sigma^2 \frac{\partial^2 u}{\partial x^2} + \mu \frac{\partial u}{\partial x} = -\frac{\partial \mu}{\partial t}$$

Nous allons discrétiser cet EDP en prenant la différence centrale de la variable x, et la différence forward de la variable temporelle t. On note alors :

$$u_{i,j} = u(x_j, t_i)\ et\ t_i = i * \ \Delta t$$

$$x_j = j * \ \Delta x$$

En substituant les différences finies dans l'EDP, on obtient :

$$\frac{1}{2}\ \sigma^2 \left(\frac{u_{j+1,i+1} - 2u_{j,i+1} + u_{j-1,i+1}}{\Delta x^2}\right) + \mu \left(\frac{u_{j+1,i+1} - u_{j-1,i+1}}{2\Delta x}\right) = -\left(\frac{u_{j,i+1} - u_{j,i}}{\Delta t}\right)$$

On a alors l'équation ci-dessous pour exprimer le prix de l'option :

$$u_{j,i} = \widetilde{P_u} * u_{j+1,i+1} + \widetilde{P_m} * u_{j,i+1} + \widetilde{P_d} * u_{j-1,i+1} \qquad (4.2)$$

Où :

$$\widetilde{P_u} = \frac{\sigma^2 \Delta t}{2\Delta x^2} + \frac{\mu \Delta t}{2\Delta x}$$

$$\widetilde{P_m} = 1 - \frac{\sigma^2 \Delta t}{\Delta x^2}$$

$$\widetilde{P_d} = \frac{\sigma^2 \Delta t}{2\Delta x^2} - \frac{\mu \Delta t}{2\Delta x}$$

On peut noter que : $\widetilde{P_u} + \widetilde{P_m} + \widetilde{P_d} = 1$; ce qui indique que $\alpha = \Delta t/(\Delta x)^2$ et $\beta = \mu\Delta t/\Delta x$. Nous pouvons donc réécrire l'équation (1) :

$$\widetilde{P_u} = \frac{1}{2}(\sigma^2 \alpha + \beta)$$

$$\widetilde{P_m} = 1 - \sigma^2 \alpha$$

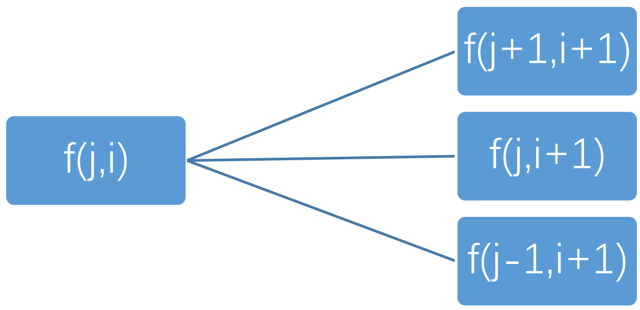
$$\widetilde{P_d} = \frac{1}{2}(\sigma^2 \alpha - \beta)$$

En substituant la valeur actuelle de l'option $f_{j,i} = e^{-r(\mathrm{T}-t_i)} u_{j,i}$ dans l'équation (4.2). Nous arrivons à la relation d'induction backward :

$$f_{j,i} = e^{-r\Delta t}\left(\widetilde{P_u} f_{j+1,i+1} + \widetilde{P_m} f_{j,i+1} + \widetilde{P_d} f_{j-1,i+1}\right) \qquad (4.3)$$

Ceci est similaire à l'induction backward de l'arbre trinomial. Par ailleurs, ceci est équivalent à l'espérance actualisée du prix de l'option à terme (sous une mesure neutre au risque). Ainsi, nous avons montré que le schéma de différences finies explicite équivaut à approximer le processus de diffusion par un processus trinomial discret.

Le graphique suivant montre une vue schématique de la discrétisation par différence finie explicite :



IV.2 Implémentation en Python

Pour simplifier le pricing, on élimine le fait que les coupons non versés peuvent être récupérables ultérieurement. C'est-à-dire qu'à chaque date d'anniversaire, si la performance du sous-jacent est inférieure à la barrière de coupon (80%), le coupon est définitivement perdu et n'est pas reporté à l'anniversaire prochain.

En implémentant notre méthode en python, on crée une matrice de valeur de l'autocall (longueur = N = nombre de working days depuis la date initiale jusqu'à la date de maturité et largeur = 2*M +1 = nombre de prix de sous-jacent, dans notre pricer on prend M= 500).
On choisit : dt = 1/252 (1 working day) et dx = vol * racine(dt) (plus que dx, dt sont petits, mieux on est).

- A la date de maturité : on utilise la fonction (4.1) pour calculer la valeur de l'option
- A chaque date entre les diverses dates d'anniversaire : on utilise l'équation (4.3) pour calculer la valeur d'option
- A chaque date d'anniversaire, on doit faire attention aux barrières :

$$\begin{cases} \dfrac{S_t}{S_0} \geq 110\%:\ V = Notional + Coupon \\ 110\% > \dfrac{S_t}{S_0} \geq 80\% :\ \ V = V_{deductive} + Coupon \\ \dfrac{S_t}{S_0} < 80\%:\ V = V_{deductive} \end{cases}$$

IV.3 Résultat

Avec les mêmes paramètres, la valeur de autocall = 1023 < la valeur de autocall qu'on a trouvé par la méthode B&S (dans la section 2). Cela semble normal puisque l'on a éliminé la caractéristique récupérable du coupon pour simplifier le pricing.

IV.4 Ajouter la possibilité de faire des clauses callables au lieu d'autocallable

On a ajouté cette possibilité en ajoutant la variable callable_clause dans notre pricer. Si on met **callable_clause = False**, on a le pricer d'un autocall normale et si on met **callable_clause = True**, on a le pricer d'un produit avec la clause callable au lieu d'autocallable.
La différence entre les deux est qu'à chaque date d'anniversaire, avec la clause callable, l'investisseur a le choix entre exécuter le produit et le garder en attendant un payoff plus grand dans la future. C'est pourquoi, à chaque date d'anniversaire : **la valeur de autocall = Max (la valeur d'autocall si elle est exercée, la valeur déductive de cette autocall).**
En ajoutant la possibilité de faire les clauses callables, on trouve le prix d'autocall = 1116 > 1023 (le prix d'autocall avec les clauses autocallables). Cela conforme à notre attente parce que naturellement la clause optionnelle rend le produit cher.