



ATHENA - AUTOCALL PRICER

Simulation Monte Carlo | Produit Structuré |
Application Financière Interactive

Mohamed Boumezou

Université
Paris Dauphine - PSL

mohamed.boumezou@dauphine.eu

Présentation




Objectif :

Ce document présente une application Python interactive de pricing d'un produit structuré de type Autocall Athena, conçue avec Streamlit. Le but est de rendre la modélisation financière accessible, interactive, et adaptable à différents scénarios de marché.

Fonctionnement de l'autocall :

L'Autocall Athena est un produit structuré dont le rendement dépend de la performance d'un sous-jacent (souvent une action ou un indice), avec des dates d'observation régulières (généralement annuelles).

Il offre :

-  **Rappel automatique** si, à une date d'observation, le sous-jacent est supérieur ou égal à son niveau initial : **remboursement du capital + coupon**.
-  **Sinon**, le produit continue.
-  **À maturité :**
 - Si le sous-jacent \geq niveau initial \rightarrow **capital + coupon**
 - Si le sous-jacent $<$ niveau initial mais \geq barrière \rightarrow **remboursement du capital**
 - Si le sous-jacent $<$ barrière \rightarrow **perte en capital proportionnelle**

Modélisation mathématique :

Pour simuler le comportement de l'Autocall Athena, on modélise deux choses :

- Le **taux sans risque** : il évolue de manière aléatoire selon un modèle réaliste utilisé en finance (appelé modèle CIR).
- Le **sous-jacent** (action ou indice) : son prix évolue aussi de façon aléatoire, avec une **volatilité qui change dans le temps et selon le niveau du prix** (ce qu'on appelle une *volatilité locale*).

Utilisation du pricer

L'application **Athena - Autocall Pricer** permet de simuler le prix d'un produit structuré de type Autocall à partir d'une interface intuitive et interactive. L'utilisateur peut visualiser les résultats en temps réel après avoir défini ses propres hypothèses.

Voici un premier aperçu de l'interface :

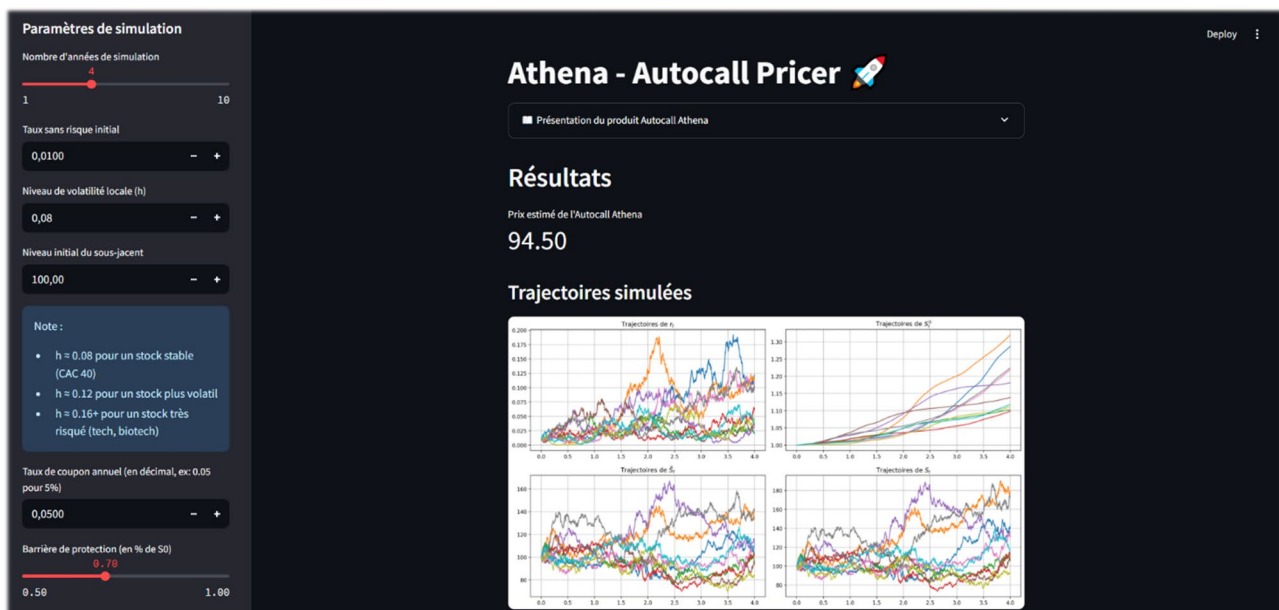


Figure 1 - Interface principale de l'application Athena - Autocall Pricer

Personnalisation des paramètres de simulation :

Chaque paramètre est **entièrement modifiable** afin de permettre à l'utilisateur d'adapter la simulation à ses hypothèses de marché ou aux caractéristiques spécifiques d'un produit. Les éléments ajustables sont :

- La **durée de vie du produit**, exprimée en années, peut être choisie entre 1 et 10 ans.
- Le **taux sans risque initial** peut être défini librement, afin d'intégrer différents scénarios de marché (modélisé ici via un processus CIR).
- Le **niveau de volatilité locale** h représente la variabilité du sous-jacent en fonction du temps et du prix :
 - $h \approx 0,08$ pour un stock stable (ex : CAC 40)
 - $h \approx 0,12$ pour un actif plus volatil
 - $h \geq 0,16$ pour un actif très risqué (ex : tech, biotech)
- Le **niveau initial du sous-jacent** est également personnalisable.
- L'utilisateur peut choisir le **taux de coupon annuel**, en décimal (ex : 0.05 pour 5 %).
- Enfin, la **barrière de protection**, exprimée en pourcentage du niveau initial, peut être ajustée pour refléter le degré de protection souhaité.

Résultats de simulation :

Une fois les paramètres définis, l'application affiche dans la section "Résultats" le **prix estimé** de l'Autocall Athena. Ce prix correspond à la **valeur actuelle attendue du produit**, calculée à partir de plusieurs milliers de trajectoires simulées, en tenant compte :

- des règles de rappel automatique,
- de la barrière de protection,
- des éventuels coupons versés,
- et des performances possibles du sous-jacent à l'échéance.

Il s'agit donc d'une **estimation par simulation de Monte Carlo**, reflétant à la fois l'aléa du marché et les caractéristiques contractuelles du produit.

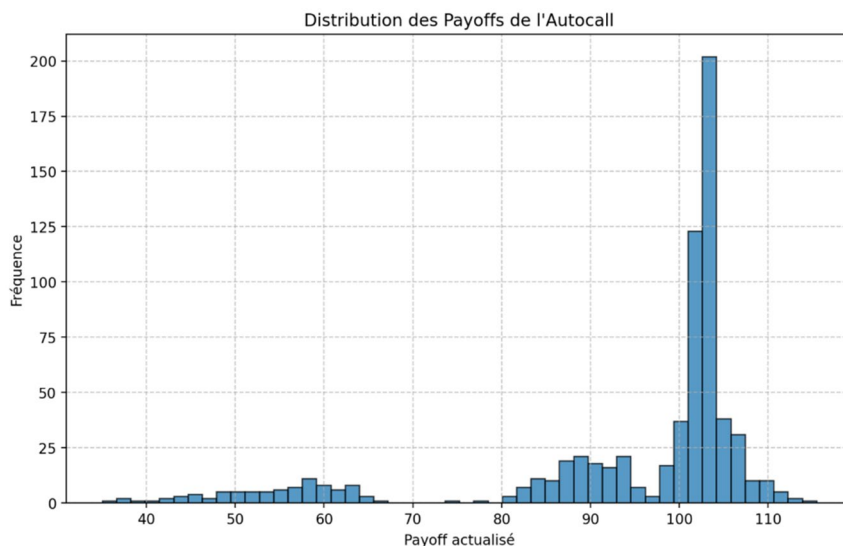
Graphiques des trajectoires simulées :

En fonction des paramètres choisis, l'application affiche plusieurs graphiques représentant les trajectoires simulées des variables principales du modèle.

On y retrouve tout d'abord le **taux d'intérêt**, qui évolue selon un processus aléatoire réaliste. Ensuite, différentes courbes montrent l'évolution du **cours initial projeté**, du **cours final simulé**, ainsi que du **prix du sous-jacent actualisé**, c'est-à-dire corrigé des effets du temps et des taux.

Ces visualisations permettent de comprendre comment les paramètres influencent les résultats, de repérer des scénarios extrêmes (hausse soudaine, chute rapide...), et surtout d'avoir une idée plus concrète des chances de **rappel anticipé** ou de **perte en capital** selon la configuration choisie.

Enfin, une **distribution des payoffs** de l'Autocall est également disponible, calculée à partir de l'ensemble des trajectoires simulées. Elle permet d'avoir une vision d'ensemble sur les scénarios possibles, les pertes potentielles et les gains attendus selon les paramètres choisis.



Fonctionnalité de backtest :

Cette fonctionnalité permet de récupérer automatiquement les données historiques d'un actif via *yfinance*, à partir de son ticker (ex. : TTE.PA) et d'une période définie. Elle simule le comportement du produit structuré Autocall Athena sur cette période, en appliquant les règles de rappel anticipé et de protection du capital. L'utilisateur peut ainsi connaître **le payoff qu'il aurait obtenu et l'année à laquelle le produit aurait été rappelé**, si les conditions avaient été remplies.

Paramètres du backtest

Ticker du sous-jacent (ex: TTE.PA)

Date de début

Date de fin

Lancer le backtest

Conclusion :

Ce projet a été conçu avec l'objectif de **rendre accessible la modélisation d'un produit structuré complexe**, en combinant rigueur financière, approche pédagogique et interactivité.

L'application permet de **simuler, visualiser et interpréter** le comportement d'un Autocall Athena dans différents contextes de marché, grâce à une interface claire et des outils personnalisables.

Elle reflète ma volonté de **comprendre en profondeur les produits financiers**, mais aussi de développer des **outils concrets** pouvant servir aussi bien à des étudiants qu'à des professionnels.