

# **Pricing d'un Phoenix sous le modèle de Black-Scholes : mise en œuvre par simulation Monte Carlo et analyse de sensibilité**

Mémoire présenté par

Eole DORON et Maxime GAGLIARDI

Encadrant :

Prof. Mario HERNANDEZ TINOCO (Edhec Business School)

Juin 2025

# Sommaire

Table des graphiques .....	3
Résumé .....	4
Chapitre 1 : Les autocallables.....	5
Définition .....	5
A quoi servent-ils ? .....	5
Quels sont les deux principaux types d'autocall ? .....	6
Que représentent les trois barrières de ces autocalls ?.....	6
Payoff .....	7
Chapitre 2 : Sous-jacent, Monte Carlo, Grecques .....	9
2.1 Choix du sous-jacent et modalités du contrat .....	9
2.2 Méthode de Monte Carlo .....	10
2.3 Définition des grecques.....	12
Delta .....	13
Gamma .....	13
Vega.....	14
Theta .....	14
Rho.....	14
Epsilon .....	15
Chapitre 3 : Simulation et analyse des grecques .....	17
Données utilisées pour l'application numérique .....	17
3.1 Mise en œuvre du pricing Monte Carlo.....	18
Rétrospection sur notre autocall .....	19
3.2 Résultats numériques des grecques et interprétations .....	21
<i>Analyse des grecques</i> .....	21
Delta .....	21
Gamma .....	22
Vega.....	23
Theta .....	25
Rho.....	26
Epsilon .....	27

<i>Analyse des barrières</i> .....	29
Conclusion .....	30
Annexe - Code .....	31
Sources .....	42

## Table des graphiques

Figure 1 – Prix spots de NVDA et barrières .....	9
Figure 2 – Prix de l'autocall en fonction du prix du sous-jacent.....	19
Figure 3 – Delta de l'autocall en fonction de $S_0$ .....	21
Figure 4 – Gamma de l'autocall en fonction de $S_0$ .....	22
Figure 5 – Vega de l'autocall en fonction de $S_0$ .....	23
Figure 6 – Theta de l'autocall en fonction de $S_0$ .....	25
Figure 7 – Rho de l'autocall en fonction de $S_0$ .....	26
Figure 8 - Epsilon de l'autocall en fonction de $S_0$ .....	27
Figure 9 - Effet de la variation des barrières sur le prix de l'autocall .....	29

## Résumé

Les produits autocallables jouent un rôle central sur les marchés financiers, en particulier dans le domaine des produits structurés. Ces instruments offrent des rendements potentiels conditionnés par la performance de l'actif sous-jacent. La tarification des autocalls est une entreprise complexe en raison de leur nature, impliquant des composantes telles que les obligations zéro-coupon, les options digitales et les options de vente avec barrières. Les solutions analytiques pour tarifier ces produits ne sont pas facilement disponibles, ce qui nécessite l'utilisation de méthodes numériques. Cette thèse se concentre sur la tarification des produits autocallables en utilisant les simulations Monte Carlo et le modèle de Black-Scholes. Nous abordons en détail les produits autocallables, puis nous analyserons à partir d'un exemple les différentes sensibilités du prix du produit aux paramètres qui le composent afin d'interpréter les résultats.

L'étude débute par une introduction aux produits autocallables, incluant leur structure, leurs composantes, et l'importance des barrières et des conditions de coupon. Nous explorons ensuite la méthodologie de simulation Monte Carlo sous Black-Scholes pour le pricing, suivie d'une analyse des différentes « grecques » (Delta, Gamma, Vega, Theta, Rho, Epsilon).

# Chapitre 1 : Les autocallables

## Définition

Le terme 'autocall' vient de l'anglais 'Automatically Callable', qui signifie automatiquement remboursable. Ces produits sont normalement appelés produits autocallables, mais on utilise plus couramment le terme 'autocall'. Un produit dit 'autocall' est un produit structuré qui permet aux investisseurs de générer des rendements basés sur la performance d'un ou plusieurs actifs sous-jacents tels que des actions, des indices boursiers ou des matières premières.

Un autocall est principalement caractérisé par trois barrières : une barrière de coupon, une barrière d'autocall et une barrière de protection du capital, qui sont déterminées à la date de lancement du produit, en même temps que la date où les échéances des coupons sont fixées ainsi que leur fréquence.

## A quoi servent-ils ?

Les autocalls permettent aux investisseurs de s'exposer à un actif sous-jacent, qui peut être composé d'une ou plusieurs actions, d'un indice boursier, des taux d'intérêt, des devises, des matières premières, ou autres actifs. Ils présentent des rendements attractifs mais comportent une part de risque (« no free lunch » et efficacité présumée des marchés, rendement directement lié aux risques), avec une protection partielle ou totale du capital investi. Le prix à payer pour l'investisseur en échange de cette protection prend plusieurs formes : une date de fin incertaine (sur un produit non-liquide), un rendement maximal capé et un risque de faillite du vendeur de l'autocall.

Le fonctionnement d'un autocall repose sur un scénario préétabli. Il est généralement lié à une barrière prédéfinie liée à la valeur de l'actif sous-jacent. Si cette barrière est atteinte ou dépassée à une date d'observation prédéterminée, le produit se clôture et rembourse automatiquement les investisseurs du capital investi ainsi que du rendement accumulé jusqu'à cette date.

Si la barrière n'est pas atteinte ou dépassée à la date d'observation, l'autocall se poursuit pour une autre période, accumulant ainsi du rendement sur une autre période. Ce processus se répète jusqu'à ce que la barrière soit atteinte ou dépassée, ou jusqu'à l'échéance finale du produit.

L'avantage principal d'un produit autocallable pour les investisseurs est la possibilité de générer des rendements attrayants même dans des conditions de marché modérément baissières.

## Quels sont les deux principaux types d'autocall ?

Les autocalls Athena : Pas de coupons intermédiaires versés, toute la valeur de l'autocall est perçue à échéance (investissement et rendement), lorsque le produit est rappelé ou qu'il arrive à terme sans avoir atteint la barrière de rappel.

Les autocalls Phoenix : Des coupons conditionnels sont reçus aux dates d'observation, sinon c'est le même fonctionnement que pour un autocall de type Athena.

Dans la suite nous allons nous intéresser à un autocall Phoenix.

## Que représentent les trois barrières de ces autocalls ?

La barrière de coupon ( $B_c$ ) : Aux dates d'observation précédemment définies, la valeur de l'actif sous-jacent est relevée, cette valeur est ensuite comparée à  $B_c$ .

Notons  $t_1, \dots, t_n$  les différentes dates à laquelle le niveau du sous-jacent est relevé.

Si pour  $1 \leq i \leq n$ , on a  $S_{t_i} \geq B_c$ , l'investisseur reçoit le coupon attendu (défini dans le contrat du produit au début, soit directement, soit à l'échéance, selon qu'il s'agisse d'un Athena ou d'un Phoenix).

La barrière d'autocall ou barrière de rappel ( $B_A$ ) : A une date d'observation si la valeur du sous-jacent a atteint ou dépassé cette barrière, le produit est rappelé (ou autocalled en anglais). Le produit est alors automatiquement et intégralement remboursé à sa valeur initiale à laquelle sont ajoutés les coupons touchés lors des précédentes dates d'observation pour un Athena ou un dernier coupon pour un Phoenix, et le contrat prend fin immédiatement.

La barrière de protection du capital ( $B_p$ ) : A la dernière date d'observation, si le produit n'a pas déjà été rappelé, le montant initial investi est remboursé partiellement ou dans son entièreté selon la valeur du sous-jacent.

Si  $S_T \geq B_p$ , le montant initial de l'investissement est remboursé à sa valeur nominale, mais le détenteur peut ne pas recevoir de coupon selon le niveau de la barrière de coupon.

Si  $S_T < B_p$ , le remboursement du capital initial s'effectue partiellement, aucun coupon final n'est versé. La valeur du remboursement correspond à l'évolution de l'actif sous-jacent entre le début et la date de fin du contrat : le capital est remboursé en fonction de la performance du titre sur la durée du produit. L'investisseur effectue une perte sur son investissement, à cause de la mauvaise performance du sous-jacent (matière première, obligation, volatilité, action, panier d'action, taux, ou autre).

Dans le contrat, le vendeur et l'acheteur de l'autocall doivent définir et inscrire : le type et la valeur initiale du sous-jacent ( $S_0$ ), les dates d'observation (ou la fréquence des observations

et la date de fin du contrat, c'est équivalent), les trois barrières (en proportion de  $S_0$ ):  $B_C$ ,  $B_A$  et  $B_P$ , ainsi que les coupons.

## Payoff

Maintenant que nous avons compris le fonctionnement des produits autocallables, nous pouvons exprimer  $P$ , le payoff d'un autocall en fonction de  $N$ , le nombre de coupons perçus avant que l'autocall ne soit rappelé. On rappelle que  $t_1, t_2, \dots, t_m$  correspondent aux dates d'observation du produit.

Pour calculer le nombre de coupon  $N$ , on vérifie à chaque date d'observation  $t_i$  :

- $S_{t_i} \geq B_C$  (barrière de coupon atteinte)
- $S_{t_1} < B_A, S_{t_2} < B_A, \dots, S_{t_{i-1}} < B_A$  (barrière de rappel jamais atteinte précédemment)

On obtient la formule suivante :

$$N = \sum_{t=t_1}^{t_m} 1_{(S_t \geq B_C)} \cdot \prod_{k=1}^{t-1} 1_{(S_k < B_A)}$$

On rappelle que  $1_{(A)}$  est la fonction indicatrice qui vaut 1 lorsque l'événement  $A$  est réalisé, et 0 sinon.

Pour exprimer  $P$ , on distingue deux possibilités :

Première possibilité : L'actif sous-jacent est supérieur ou égal à la barrière de protection du capital  $B_P$  à la fin de sa vie.

Cela signifie que l'événement  $\{S_T \geq B_P\}$  ou  $\{\max(S_{t_i}) \geq B_A, 1 \leq i \leq m\}$  s'est produit.

Dans ce cas, le payoff est :

$$\text{Autocall}_{\text{payoff}} = K + N \times C$$

où  $K$  représente le capital initial et  $C$  le coupon versé à chaque date d'observation.

Deuxième possibilité : L'actif sous-jacent est strictement inférieur à la barrière de protection du capital  $B_P$  à l'échéance.

C'est-à-dire que l'événement  $\{S_T \leq B_P\}$  s'est produit.

Le payoff est alors :

$$\text{Autocall}_{\text{payoff}} = K \times \frac{S_T}{S_0} + N \times C$$

En combinant les deux possibilités, on obtient la formule complète :

$$\text{Autocall}_{\text{payoff}} = \mathbb{1}_{(S_T \geq B_P \vee \max_{1 \leq i \leq m} S_{t_i} \geq B_A)} \times (K + N \times C) + \mathbb{1}_{(S_T < B_P)} \times \left( K \times \frac{S_T}{S_0} + N \times C \right)$$



## Chapitre 2 : Sous-jacent, Monte Carlo, Grecques

### 2.1 Choix du sous-jacent et modalités du contrat

Pour notre contrat, nous avons choisi l'action Nvidia (NVDA) enregistrée au NASDAQ, à partir de laquelle nous allons constituer un Phoenix. On décide de le faire débuter au 17 décembre 2022 (pour pouvoir ensuite faire une rétroanalyse) et une date de fin de vie fixée le 17 juin 2025 (2,5 ans plus tard soit 30 mois). A date de création le prix de l'action Nvidia est de 16,25\$. La barrière d'autocall est fixée à 100 % de la valeur initiale de Nvidia ( $B_A = 16.25$ ). La barrière de protection du capital ainsi que la barrière de coupon sont fixées à 70 % de la valeur de départ de Nvidia ( $B_P = B_C = 11.375$ ).

On fixe un investissement de départ de  $I_0 = 1\,000\,000\$$  et des coupons semestriels de 2,50% (soit 5% par an), avec une fréquence d'observation semestrielle, donc tous les 6 mois après la date de lancement du produit.

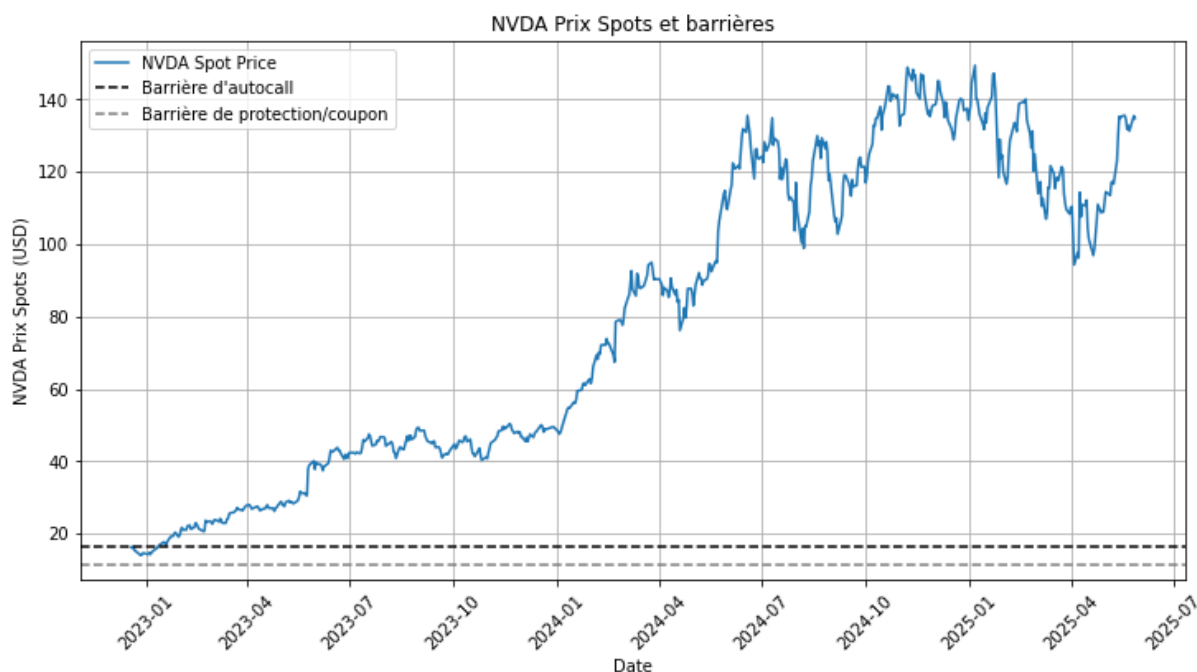


Figure 1 – Prix spots de NVDA et barrières

## 2.2 Méthode de Monte Carlo

Nous voulons déterminer le prix de notre autocall. Pour cela nous allons utiliser la méthode de Monte Carlo avec le modèle de Black-Scholes. Le modèle de Black-Scholes va nous permettre de tracer une des trajectoires possibles de l'évolution du prix de l'action Nvidia, et grâce à la méthode de Monte Carlo, nous allons répéter ce processus un grand nombre de fois et faire la moyenne de ce que nous rapportera l'autocall dans chacune de ces situations, ce qui sera notre prix (car nous ne prenons pas en compte les frais bancaires liés à leur marge et/ou à leurs coûts de fonctionnement).

On fait l'hypothèse que le prix de l'action Nvidia  $S$  suit un mouvement brownien géométrique qui nous donne la formule suivante :

$$dS = \mu S dt + \sigma S dz$$

Où  $\mu = r - q$  est le rendement espéré (ou drift net) et  $\sigma$  est la volatilité du prix.

Afin d'assurer un cadre d'évaluation sans arbitrage, toutes nos simulations de pricing sont menées sous la mesure dite « risque-neutre », c'est-à-dire avec un drift égal à  $r - q$  (taux sans risque moins rendement du dividende). Cette convention garantit que le prix actualisé de l'actif est une martingale, condition nécessaire à la cohérence de la tarification. À l'inverse, le drift historique  $\mu$  (rendement moyen observé) reste utile pour le back-testing : il permet de comparer nos trajectoires simulées aux cours réels de NVDA et d'étudier la performance ex post de notre modèle.

$S$  est toujours positif (le prix de l'action ne peut pas être négatif) et la courbe de  $S$  est continue. On peut donc lui appliquer le lemme d'Itô.

On lui applique le lemme d'Itô pour dériver le processus suivi par  $G = \ln(S)$  quand  $S$  suit une loi brownienne géométrique.

On a :

$$\frac{\partial G}{\partial t} = 0; \quad \frac{\partial G}{\partial S} = \frac{1}{S}; \quad \frac{\partial^2 G}{\partial S^2} = -\frac{1}{S^2}$$

Et donc,

$$d\ln(S) = \left( \mu - \frac{\sigma^2}{2} \right) dt + \sigma dz$$

L'expression précédente possède une part déterminée (premier facteur) et une part stochastique (deuxième facteur, dépendant de la volatilité et d'une variable aléatoire).

Or, on a :

$$d\ln(S) = \ln(S_{t+\delta t}) - \ln(S_t) = \ln(S_{t+\delta t}/S_t)$$

Donc on remplace dans l'équation précédente :

$$\ln\left(\frac{S_{t+\delta t}}{S_t}\right) = \left(\mu - \frac{\sigma^2}{2}\right)dt + \sigma dz$$

On passe les deux cotés à l'exponentielle :

$$\frac{S_{t+\delta t}}{S_t} = e^{\left(\mu - \frac{\sigma^2}{2}\right)dt + \sigma dz}$$

Et donc on peut prédire  $S_{t+1}$  en fonction de  $S_t$  :

$$S_{t+\delta t} = S_t e^{\left(\mu - \frac{\sigma^2}{2}\right)\delta t + \sigma \varepsilon \sqrt{\delta t}}$$

Où  $\varepsilon$  est un tirage suivant la loi normale  $N(0,1)$ .

Donc maintenant qu'on a une méthode pour créer une des trajectoires potentielles, nous allons trouver le prix de l'autocall avec la méthode de Monte Carlo :

Pour chacune des trajectoires de prix  $i$  de Nvidia, on calcule la valeur présente de la valeur finale de l'autocall  $f(S_i)$ .

Pour les 10 000 générations de trajectoires, l'estimation Monte Carlo donne un prix avec :

$$\hat{f}_{10000} = \frac{1}{10000} \sum_{i=1}^{10000} f(\widetilde{S}_i)$$

La précision dépend du nombre de simulations, pour  $n$  essais on obtient un intervalle de confiance de 95% sur le prix donné par :

$$\hat{f}_n - \frac{1,96\hat{\sigma}}{\sqrt{n}} < f < \hat{f}_n + \frac{1,96\hat{\sigma}}{\sqrt{n}}$$

Où  $\hat{\sigma}$  est la volatilité empirique.

## 2.3 Définition des grecques

Dans le cadre du modèle de Black-Scholes et afin de pouvoir analyser les différentes sensibilités de l'autocall aux paramètres de son prix, il est indispensable de définir ce que sont les différentes « grecques » que nous allons utiliser par la suite.

Le modèle de Black-Scholes permet de donner un prix aux options call et put européennes en se basant sur les hypothèses suivantes :

Le prix d'une option d'achat  $C$  dépend des paramètres suivants :

- Le prix au comptant du sous-jacent  $S$  à la date de valorisation.
- La volatilité  $\sigma$  des rendements du sous-jacent.
- Le taux d'intérêt  $r$  et le taux de dividende  $q$ .
- Le prix d'exercice  $K$  et le temps restant jusqu'à l'échéance  $T$ .

La formule du prix du call européen donnée par le modèle de Black-Scholes-Merton est alors :

$$C = Se^{-qT}N(d_1) - Ke^{-rT}N(d_2)$$

$$\text{Avec } d_1 = \frac{\ln\left(\frac{S}{K}\right) + \left(r - q + \frac{1}{2}\sigma^2\right)T}{\sigma\sqrt{T}}, \quad d_2 = d_1 - \sigma\sqrt{T}$$

Et  $N$  est la fonction de répartition d'une loi normale centrée réduite,

$$N(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{t^2}{2}} dt$$

On rappelle également la formule du prix d'un contrat forward  $F$  associé qui nous servira lors de nos analyses.

$$F = Se^{(r-q)T}$$

D'où l'on peut réécrire la formule du prix du call de la manière suivante :

$$C = e^{-rt}[FN(d_1) - KN(d_2)]$$

Pour une option de vente européenne, le prix du put noté  $P$  s'exprime de la manière suivante :

$$P = Ke^{-rT}N(-d_2) - Se^{-qT}N(-d_1)$$

Une fois cela défini, nous pouvons aborder les principales grecques que nous utiliserons.

L'achat ou la vente d'un dérivé génère divers risques, certains indésirables, que la couverture vise à réduire par des transactions compensatoires. Une position non couverte est très risquée. Lorsqu'une banque vend un dérivé ou une structure telle qu'un autocall, elle doit comprendre et couvrir les risques avant d'ajouter le produit à son portefeuille d'options, où le risque sera géré. Afin de mettre en place une couverture adéquate, le trader utilise les hedge ratios, ou les grecques, pour mesurer les sensibilités du prix du produit aux paramètres influençant sa valeur.

## Delta

Le Delta indique le montant approximatif par lequel le prix de l'option varie pour un changement marginal du prix du sous-jacent. Pour une option telle qu'un call ou un put, il s'agit de la dérivée première du prix de l'option par rapport au sous-jacent. Les formules du Delta données pour ces options dans le cadre du modèle de Black et Scholes sont :

$$\Delta_{call} = \frac{\partial C}{\partial S} = e^{-qT} N(d_1)$$

$$\Delta_{put} = \frac{\partial P}{\partial S} = e^{-qT} [N(d_1) - 1]$$

Le Delta d'un call est borné entre 0 et 1, se rapproche de 0 à mesure que le sous-jacent s'éloigne du prix d'exercice en diminuant, et se rapproche de 1 lorsque S s'éloigne du prix d'exercice en augmentant. Il est souvent considéré comme une approximation de la probabilité que l'option finisse « dans la monnaie ».

Pour un put, le raisonnement adéquat est l'opposé du précédent. Le Delta du put est borné entre -1 et 0, se rapproche de 0 à mesure que le sous-jacent s'éloigne du prix d'exercice en diminuant, et se rapproche de -1 lorsque S s'éloigne du prix d'exercice en augmentant.

## Gamma

Le Gamma est une autre grecque qui est cette fois-ci une approximation de comment varie le Delta lorsque le sous-jacent varie. Il correspond simplement à la pente du Delta en un point. Il est aussi un indicateur de la convexité du prix de l'option, en effet le Gamma sera d'autant plus élevé que le prix de l'option affichera une courbure prononcée à certains endroits. Sous le modèle de Black-Scholes, la formule du Gamma est la suivante :

$$\Gamma = \frac{\partial^2 C}{\partial S^2} = \frac{\partial \Delta}{\partial S} = \frac{N'(d_1)e^{-qT}}{S\sigma\sqrt{T}}$$

Etant donné qu'il s'agit de la dérivée première du Delta par rapport au sous-jacent S, on comprend que le Gamma a la même expression qu'il s'agisse d'une option de vente ou d'une option d'achat, à la différence du Delta.

## Vega

Le Vega est un indicateur essentiel de la sensibilité du prix d'une option ou d'un produit à la volatilité. Il permet d'exprimer de combien varie le prix du dérivé si la volatilité augmente d'une unité. En pratique, il est très improbable d'obtenir une variation aussi brutale de la volatilité, c'est pourquoi il faut être prudent lors des interprétations du Vega, car souvent on parlera de variation de l'ordre de quelques dixièmes tout au plus. Sous le modèle de Black-Scholes, la formule du Vega pour les options de vente et les options d'achat est donnée par

$$\mathcal{V} = S e^{-qT} N'(d_1) \sqrt{T}$$

Avec  $N'(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{x^2}{2}}$

Les options call et put sont sensibles à la volatilité, on peut même dire qu'elles sont « Vega positives ». Cela signifie que du fait de l'asymétrie de leur payoff, une hausse de la volatilité entraînera une hausse du prix de ces options. En effet, lorsque la volatilité augmente, le payoff potentiel ou gain à l'échéance augmente, ce qui augmente mécaniquement le prix de l'option.

## Theta

Le prix d'une option est composé de sa valeur intrinsèque, qui correspond pour un call à  $\max(0; S - K)$  et pour un put à  $\max(0; K - S)$ . A cette valeur intrinsèque s'ajoute une valeur extrinsèque qui correspond à la valeur-temps de l'option, qui diminue progressivement au fil du temps, un phénomène mesuré par le Theta de l'option. Cette valeur-temps est une composante venant augmenter le prix de l'option, qui témoigne de l'incertitude de l'issue obtenue à l'échéance. Plus le temps avant maturité est important, plus il y a d'incertitude concernant le payoff à l'échéance, et plus la valeur-temps de l'option est importante, car cela permet au produit d'augmenter sa probabilité de finir dans la monnaie. La valeur extrinsèque dépend fortement de la volatilité implicite du sous-jacent. La volatilité implicite joue également un rôle clé : une forte volatilité accroît la probabilité que l'option franchisse son prix d'exercice avant l'échéance, ce qui augmente la valeur extrinsèque.

## Rho

Le Rho est un indicateur clé de la sensibilité du prix d'une option ou d'un produit dérivé aux taux d'intérêt. Il permet d'exprimer de combien varie le prix de l'option lorsque le taux sans risque augmente d'une unité. En pratique, les variations des taux d'intérêt sont généralement plus progressives, ce qui signifie que l'impact du Rho est souvent plus subtil que d'autres sensibilités comme le Vega ou le Delta. Toutefois, il reste particulièrement

important pour les options de longue maturité, car le coût d'opportunité lié au taux sans risque joue un rôle plus significatif dans la valorisation du produit.

Sous le modèle de Black-Scholes, les formules du Rho pour un call et un put sont données par

$$Rho_{call} = KTe^{-rT}N(d_2)$$

Et

$$Rho_{put} = -KTe^{-rT}N(d_2)$$

On notera pour nos analyses que les prix des options call et put sont presque parfaitement linéaires par rapport aux taux d'intérêt.

## Epsilon

L'Epsilon est un indicateur secondaire qui mesure la sensibilité du prix d'une option au taux de dividende de l'actif sous-jacent. Il permet d'estimer de combien varie la valeur de l'option lorsqu'on modifie le taux de dividende anticipé d'une unité. Bien que souvent négligé dans les analyses classiques, l'Epsilon devient pertinent lorsque l'on traite d'options sur actions versant des dividendes.

En pratique, une hausse du dividende attendu tend à réduire la valeur des options d'achat (call), car l'actif sous-jacent perdra de sa valeur à la date de détachement. À l'inverse, cette variation peut augmenter la valeur des options de vente (put). Cette sensibilité est d'autant plus marquée que la maturité de l'option est longue et que le montant du dividende anticipé est important.

Comme pour le Rho, la relation entre le prix des options et le taux de dividende est généralement quasi linéaire dans la majorité des cas, ce qui fait de l'Epsilon un outil utile dans l'évaluation de scénarios sur titres à rendement.

Pour des portefeuilles d'options ou des structures complexes comme les autocalls, la sensibilité du prix aux différents paramètres l'influençant n'est pas aussi simple que pour des options vanilles. Pour déterminer les grecques globales d'un tel produit ou portefeuille, il faut en réalité agréger l'ensemble des grecques individuelles. C'est-à-dire que par exemple pour un portefeuille composé de  $n$  options, le Delta global du portefeuille est noté :

$$\Delta_{\text{Portefeuille}} = \sum_{i=1}^n \frac{\partial P_{\text{option}_i}}{\partial S} = \sum_{i=1}^n \Delta_{\text{option}_i}$$

Cela est dû à la propriété de linéarité de la dérivée. Etant donné que les grecques sont toutes déterminées par un calcul de dérivée, ce cas particulier s'applique aux autres grecques définies précédemment.

Ainsi pour le cas précis des autocalls, les grecques de ces structures sont assez complexes à déterminer avec cette méthode car pour ce faire, il est nécessaire de déterminer les grecques de chaque composante de l'autocall, puis de les additionner. Cela peut être laborieux dans le cas général notamment lorsque la maturité de l'autocall est importante. C'est pourquoi la méthode que nous utiliserons afin de calculer les grecques de l'autocall sera la méthode des différences finies.

Dans cette approche, on évalue d'abord le prix de l'autocall  $P$  pour un paramètre auquel on ajoute  $\epsilon$  (appelé « bump ») puis pour un paramètre auquel on ajoute  $-\epsilon$ , et on en déduit la dérivée par une formule dont le cas général donne :

$$f'(x) \approx \frac{f(x + \epsilon) - f(x - \epsilon)}{2\epsilon}$$

Les autocalls sont des produits que l'on appelle *path-dependent* (on doit vérifier la réalisation de la barrière à plusieurs dates). Les formules analytiques pour dériver les payoffs se compliquent rapidement lorsque la structure comporte plusieurs dates de knock-out ou de coupon. Avec les différences finies, on ne dépend plus d'une formule fermée : on réutilise simplement l'algorithme de pricing Monte Carlo.

Lorsqu'on évalue le prix de l'autocall par simulation Monte Carlo, la méthode des différences finies se conjugue naturellement avec cette approche. En effet, pour chaque bump, on lance une nouvelle série de simulations avec le paramètre modifié, puis on compare la moyenne des payoffs escomptés. En utilisant des différences finies, il est possible de tester rapidement l'impact de la taille du bump ( $\epsilon$ ) sur la stabilité numérique. Dans les analyses qui vont suivre dans le chapitre 3, pour chaque grecque, le bump est adapté de façon à obtenir le meilleur compromis entre erreur de troncature et bruit de Monte Carlo.



## Chapitre 3 : Simulation et analyse des grecques

Dans ce chapitre, nous présentons l'implémentation Python utilisée pour simuler les trajectoires du sous-jacent selon le modèle de Black-Scholes, évaluer le prix du Phoenix et calculer ses différentes sensibilités par différences finies. À l'aide d'une génération de mouvements browniens géométriques sur cinq dates d'observation semestrielles, nous estimons d'abord la valeur actuelle du produit, puis nous explorons les variations de ce prix en fonction du cours initial du sous-jacent. Les figures qui suivront illustreront la forme de la fonction de prix, ainsi que le Delta et le Gamma calculés par différences finies à partir d'une plage de valeurs du sous-jacent.

Ensuite, nous étendons cette méthode à l'ensemble des grecques : le Delta et le Gamma sont obtenus grâce au gradient d'un vecteur de prix obtenu par simulation Monte Carlo, le Vega est obtenu en bumpant la volatilité, le Theta en modifiant la maturité, le Rho en faisant varier le taux sans risque et l'Epsilon en ajustant le dividende du sous-jacent. Pour chaque paramètre, nous lançons deux sessions de Monte Carlo (un « bump » positif et un « bump » négatif) et appliquons la formule centrale de manière à tracer l'évolution de chaque sensibilité en fonction de  $S$ . Les graphiques présentés mettent en évidence les zones où l'autocall est particulièrement sensible aux changements de marché, offrant ainsi un diagnostic complet sur les risques et les leviers de couverture.

### Données utilisées pour l'application numérique

Pour notre analyse, nous avons extrait les données historiques de l'action NVDA\*, couvrant la période du 17 décembre 2022 au 17 juin 2025.

#### Résumé des composantes du Phoenix NVDA :

Date de lancement	17/12/2022
Valeur initiale du sous-jacent (VIS)	\$16.25
Valeur de référence	\$16.25 (soit 100% de $S_0$ )
Barrière de Rappel	\$11,375 (soit 70% de $S_0$ )
Barrière de Coupon et de Protection du Capital	\$11,375 (soit 70% de $S_0$ )
Nombre de dates d'observation	5
Fréquence d'observation	Semestrielle
Montant investi	\$1 000 000
Coupon	\$25 000 ( $c = 2,5\%$ )
Dividende de l'action	\$3 ( $q = 2\%$ )
Volatilité de l'action	$\sigma = 20\%$

### 3.1 Mise en œuvre du pricing Monte Carlo

L'autocall que nous avons modélisé débute avec une valeur initiale du sous-jacent (VIS) égale à 16,25\$, une maturité de 2 ans et 6 mois, cinq dates d'observation semestrielles ( $m = 5$ ), un taux sans risque de 4 % et un rendement de dividende de 2 %. La barrière d'autocall est fixée à 100 % de la VIS (16,25\$), tandis que la barrière de coupon et celle de protection de capital se situent à 60 % de la VIS (11,38\$). À chaque date d'observation, si le cours simulé dépasse la barrière d'autocall, l'investisseur perçoit un coupon de 2,5 % de l'investissement initial (0,41\$ par unité de 16,25\$) et le produit est remboursé immédiatement. Sinon, tant que le cours est au moins égal à 11,38\$, l'investisseur reçoit le même coupon semestriel, et en cas de chute finale sous cette barrière jusqu'à l'échéance, il récupère seulement une partie du capital proportionnelle au ratio  $S_T/\text{VIS}$ . En lançant 10 000 trajectoires et en actualisant chaque scénario au taux de 4 %, la moyenne des valeurs actualisées des flux futurs s'élève à 15,21\$ (prix arrondi obtenu par le script Python en annexe). Ce résultat correspond au prix « juste » de l'autocall, exprimé en unité de la valeur initiale du sous-jacent.

Un prix de 15,21\$ pour un investissement de 16,25\$ signifie que l'autocall est vendu avec une décote d'environ 6,4 % par rapport au capital initial. Cette décote reflète principalement la probabilité que le cours de Nvidia ne dépasse pas la barrière d'autocall à chaque date d'observation, ainsi que le fait qu'en cas de coupon non payé, le capital restitué à l'échéance peut être partiel si le sous-jacent clôture sous 11,38\$. Bien que la barrière d'autocall soit positionnée au pair (ATM) dès le lancement, la volatilité de 20 % et le rendement de dividende de 2 % entravent la capacité du cours à revenir à ce niveau lors des étapes intermédiaires. Sans compter qu'un coupon semestriel de 2,5 % n'est pas extrêmement généreux : l'espérance de recevoir plusieurs coupons avant un knock-out doit compenser l'absence de rendement fixe et l'éventuelle perte de capital.

La simulation confirme que, dans ces conditions, la valeur actualisée moyenne penche légèrement en faveur de l'émetteur si celui-ci vend le produit à 100% du nominal (16,25\$). Si la volatilité était plus faible, ou si le coupon semestriel était plus élevé, le prix se rapprocherait davantage de la valeur nominale de 16,25\$. Inversement, en cas de volatilité plus élevée ou de barrière de protection plus élevée (par exemple 70 % de l'VIS, le risque de remboursement partiel à l'échéance augmenterait et le prix chuterait d'autant. Le résultat chiffré à 15,21\$ permet donc d'illustrer la sensibilité du produit aux paramètres de marché et aux caractéristiques contractuelles (barrières, coupon).

## Rétrospection sur notre autocall

Comme on a choisi un début en 2022 et une maturité de 2,5 ans on peut voir comment se serait réellement comporté l'autocall.

Dates d'observation	Prix de Nvidia	Coupon
17/12/2022	\$16.25	/
17/06/2023	\$42.69	Oui
17/12/2023	\$50.08	Oui
17/06/2024	\$130.98	Oui
17/12/2024	\$130.39	Oui
17/06/2025	\$140.41	Oui

Mais dans les faits étant donné que nous n'avions pas décidé d'une période de trêve, le produit aurait été rappelé à la première date d'observation, soit le 17/06/2023, on aurait récupéré notre investissement initial augmenté de 2,50%, donc 1 025 000\$.

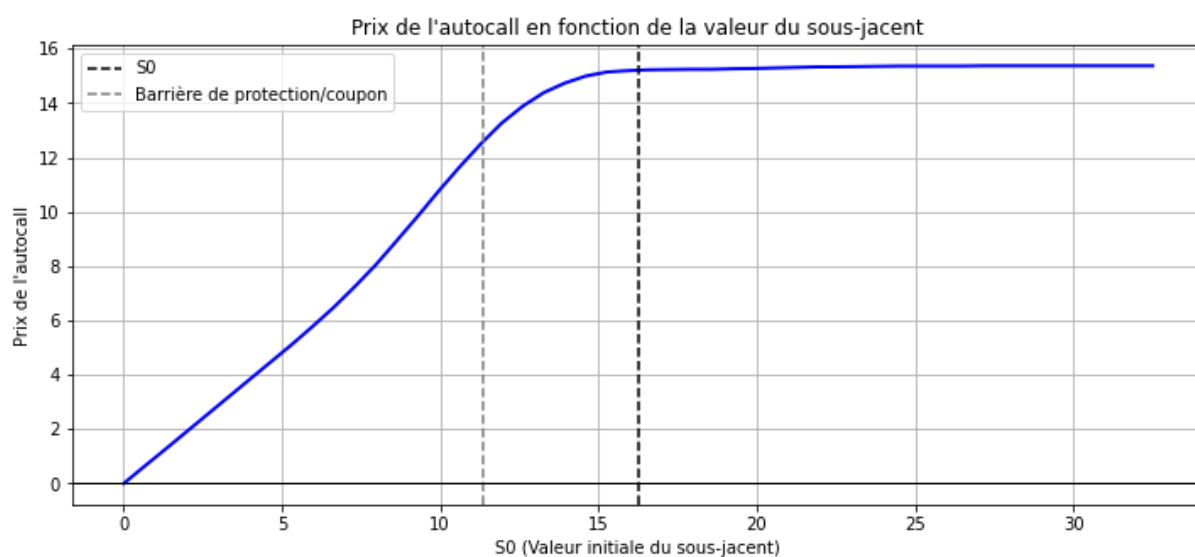


Figure 2 – Prix de l'autocall en fonction du prix du sous-jacent

Le graphique ci-dessus trace en bleu la valeur simulée de l'autocall en fonction du prix initial  $S_0$  du sous-jacent. Deux lignes en pointillés verticales marquent respectivement la VIS à 16,25\$ (barrière d'autocall) et la barrière de protection/coupon à 11,375\$. Pour  $S_0$  très faible – en dessous de la barrière de protection ( $\approx 11,38$ \$) – l'autocall ne verse jamais de coupon car le sous-jacent est trop bas à chaque date semestrielle. Dans ce cas, la seule dimension du payoff consiste à rembourser un montant proportionnel à  $S_T/VIS$  à maturité, ce qui se traduit numériquement par une valeur actualisée quasiment linéaire en  $S_0$ . Concrètement, si

$S_0$  est deux fois plus faible, l'espérance de remboursement l'est également, et donc la courbe bleue démarre au point (0, 0) et monte de façon presque linéaire quasiment jusqu'à la barrière de protection.

Dès que  $S_0$  franchit la barrière de protection, le produit devient éligible au versement semestriel d'un coupon tant qu'il ne déclenche pas l'autocall. Pour  $S_0$  intermédiaires (entre 11,38\$ et 16,25\$), la probabilité de recevoir un ou plusieurs coupons avant maturité augmente rapidement, ce qui se traduit dans la courbe par une pente forte : plus  $S_0$  se rapproche de 16,25\$, plus le scénario d'autocall anticipé devient probable dès la première date, générant ainsi la somme des coupons escomptés. À  $S_0 = \text{VIS}$  (ligne noire), l'espérance de knock-out dès la première date est maximale, d'où un pic temporaire du prix simulé, on observe en effet la valeur la plus élevée de l'autocall juste à la hauteur de cette barrière.  $S_0 = \text{VIS}$

Enfin, pour des valeurs de  $S_0$  supérieures à la barrière d'autocall, la probabilité de franchir la barrière d'autocall au premier semestre reste très forte, donc le prix tend autour d'une même valeur correspondant au paiement du premier coupon plus le remboursement du capital. Ce profil reflète parfaitement le caractère *path-dependent* du produit et l'effet barrière d'autocall : lorsque  $S_0$  est trop bas, seul un remboursement partiel à maturité existe ; près de la barrière, l'allure du prix est très sensible car chaque dixième d'unité supplémentaire augmente significativement la probabilité d'autocall ; au-delà, le produit est déjà quasiment certain d'être rappelé, de sorte que la valeur reste stable quelle que soit la hausse ultérieure de  $S_0$ .

## 3.2 Résultats numériques des grecques et interprétations

### Analyse des grecques

#### Delta

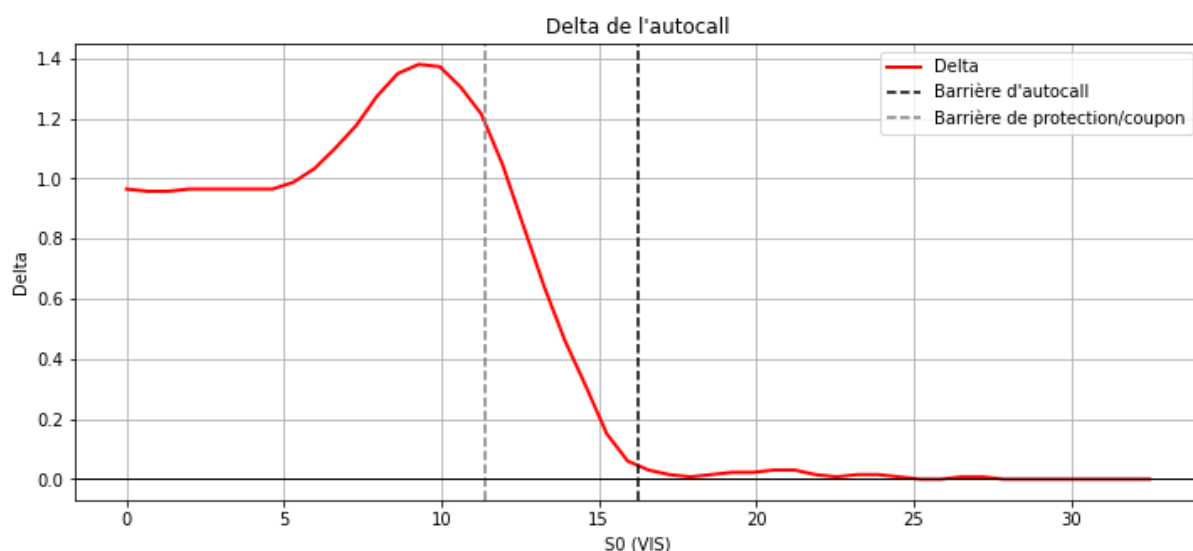


Figure 3 – Delta de l'autocall en fonction de  $S_0$

La première remarque que nous pouvons faire sur la courbe du graphique ci-dessus est que le Delta est toujours supérieur ou égal à 0. Cela signifie que, globalement, une augmentation de la valeur du sous-jacent entraîne une augmentation du prix de l'autocall. Cela fait sens pour plusieurs raisons. Premièrement, une augmentation du sous-jacent accroît la probabilité que le produit soit rappelé à la prochaine date d'observation, ce qui est positif pour le détenteur de l'autocall puisque celui-ci serait remboursé de son investissement et percevrait un coupon supplémentaire, d'où le fait que le prix de l'autocall s'accroît.

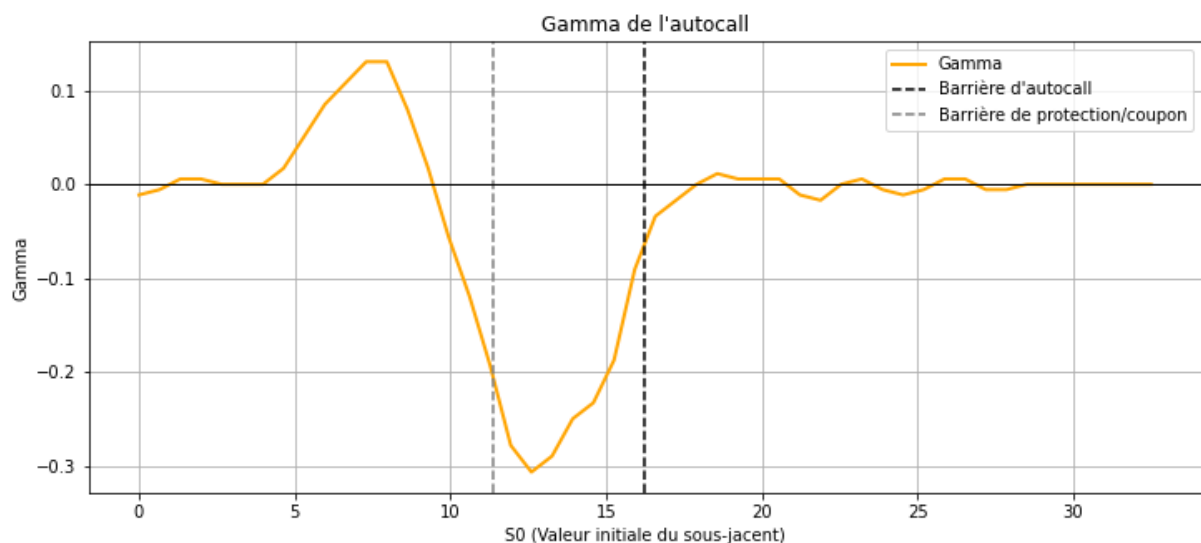
Deuxièmement, cette observation peut être comprise à partir de la décomposition de la structure autocallable. En effet, comme nous l'avons expliqué, l'autocall est constitué d'un put down-and-in et de digitales. Or, le Delta d'une option de vente (cas général) est toujours inférieur ou égal à 0, tandis que celui des digitales est toujours positif. Du point de vue de l'investisseur, la position courte sur l'option de vente down-and-in, combinée aux positions longues sur les digitales, résulte en un Delta global positif pour la structure.

Maintenant il faut souligner la forte asymétrie que présente le Delta de l'autocall. En effet, celui-ci prend des valeurs élevées lorsque  $S \leq B_p$ , avec un sommet atteint légèrement avant le franchissement de cette barrière aux alentours de 1,4. Cela indique que le prix de l'autocall est particulièrement sensible aux variations du sous-jacent dans cette zone, d'autant plus que l'on s'approche de la barrière de protection et de coupon. En effet, le franchissement de cette barrière devient de plus en plus probable, et un tel événement entraînerait un

changement radical dans le scénario de remboursement, d'où la sensibilité accrue du Delta à proximité de cette barrière. Lorsque  $B_A \geq S \geq B_P$ , le delta diminue, car le rappel du produit dès la prochaine date d'observation devient de plus en plus probable, l'effet sur le prix de l'autocall diminue progressivement jusqu'à se stabiliser autour de 0 une fois que le sous-jacent dépasse la barrière d'autocall. A ce niveau, il est quasiment certain que l'autocall sera rappelé prochainement, ce qui correspond au meilleur scénario que peut espérer l'investisseur. Dans ce cas, une augmentation additionnelle du sous-jacent n'a plus d'effet sur la valeur de l'autocall, ce qui explique pourquoi le Delta s'annule.

Avant de passer au Gamma, il est intéressant de noter que cette forte asymétrie du Delta met en lumière la non-linéarité du comportement de l'autocall. En effet, le pic du Delta avant la barrière de protection préfigure une zone où le Gamma – mesurant la vitesse d'évolution du Delta – devrait être particulièrement élevé. Cela indique que la sensibilité du Delta aux mouvements du sous-jacent change rapidement près des barrières, rendant la gestion du risque via une couverture dynamique plus délicate et coûteuse. Cette observation servira de transition naturelle vers l'analyse du Gamma, qui approfondira la compréhension de cet effet de levier sur le risque de l'autocall.

## Gamma



*Figure 4 – Gamma de l'autocall en fonction de  $S_0$*

La courbe du Gamma de l'autocall ci-dessus prend à la fois des valeurs positives et négatives, et présente des pics marqués aux abords des barrières critiques de protection et d'autocall. Lorsque le sous-jacent se trouve bien loin de ces niveaux, le Gamma est relativement proche de 0, ce qui signifie que la variation du Delta est assez faible et que le comportement de l'autocall est quasi linéaire par rapport au sous-jacent. Cependant, dès que le sous-jacent commence à s'approcher de l'une ou l'autre de ces barrières, le Gamma augmente fortement en valeur absolue. Ce phénomène traduit une importante convexité du prix de l'autocall,

signifiant qu'une infime variation du sous-jacent dans ces zones entraîne un changement rapide du Delta.

Ce pic du Gamma est particulièrement prononcé juste avant le franchissement de la barrière de protection, indiquant que le delta de l'autocall devient très sensible dans cette zone critique. Autrement dit, à proximité de cette barrière, même de petits mouvements du sous-jacent peuvent engendrer une modification significative du Delta, reflétant l'incertitude quant au rappel du produit. Cette sensibilité accrue met en évidence la difficulté de couvrir précisément ce risque, car le Delta doit être ajusté de manière très dynamique face aux variations locales du sous-jacent. De plus, l'asymétrie observée dans la courbe du Gamma est similaire à celle du Delta. Lorsque le sous-jacent dépasse la barrière d'autocall, le Gamma retrouve rapidement des valeurs faibles. Cela traduit le fait que, dans le scénario optimal (rappel imminent), le produit devient moins réactif aux mouvements supplémentaires du sous-jacent.

Ainsi, le graphique du Gamma met en lumière la forte non-linéarité associée aux barrières de l'autocall. Les pics de Gamma aux abords des barrières indiquent que la sensibilité du Delta évolue rapidement dans ces zones, rendant la couverture dynamique particulièrement délicate et soulignant l'importance de gérer avec attention le risque de convexité.

## Vega

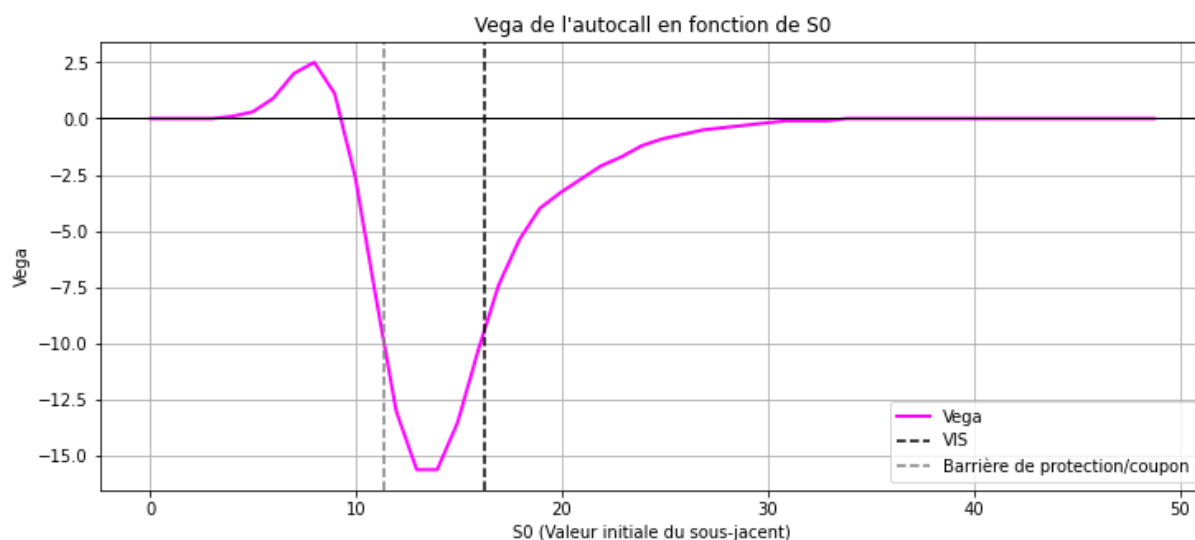


Figure 5 – Vega de l'autocall en fonction de  $S_0$

Le Vega de l'autocall prend des valeurs négatives pour la majeure partie des niveaux de  $S_0$ . Autrement dit, une hausse de la volatilité entraîne, généralement, une baisse du prix de l'autocall, sauf en cas de faibles valeurs du sous-jacent.

Ceci s'explique par le fait que l'autocall se compose, d'un côté, d'une position courte sur un put down-and-in dont la valeur augmente quand la volatilité monte et d'une position longue

sur des digitales dont le Vega varie de positif à négatif selon le niveau du forward. En fait, l'effet de la volatilité sur les digitales donne en partie sa forme au Vega de l'autocall, le Vega d'une digitale est positif lorsque le forward est inférieur au prix d'exercice (ici la barrière de coupon), car une hausse de la volatilité augmente la probabilité de recevoir le coupon, ce qui augmente le prix. Lorsque le forward est supérieur au niveau du coupon, le Vega devient négatif, car une hausse de la volatilité entraîne, à l'inverse, une plus grande probabilité que le sous-jacent termine en dessous du niveau de coupon à maturité. L'agrégation de ces deux effets se traduit par un Vega négatif, hormis pour de faibles valeurs du sous-jacent, correspondant aux niveaux où l'effet de la volatilité sur les digitales l'emporte sur celui de l'option de vente.

Pour de faibles niveaux du sous-jacent, lorsque celui-ci reste largement en-dessous de la barrière de protection, il est clair qu'une hausse de la volatilité ne peut avoir qu'un effet positif sur la valeur de l'autocall. Cela accroît la probabilité de scénario favorable à l'investisseur, avec un franchissement positif de la barrière de protection, permettant l'obtention d'un coupon.

On relève un pic négatif sur le Vega lorsque  $S_0$  se situe autour de 13\$, à équidistance entre la barrière de protection et la barrière d'autocall. Cette zone s'explique par la configuration suivante : à ces cours intermédiaires, le sous-jacent est encore susceptible de clôturer sous la barrière de protection, ce qui laisse aux digitales une probabilité non négligeable de ne pas être activées. Dans cette situation, une légère augmentation de la volatilité accroît la dispersion des trajectoires, ce qui augmente la probabilité que le sous-jacent diminue en-dessous de la barrière de protection, et donc que l'autocall perde de la valeur.

Enfin, pour  $S_0$  largement supérieurs à la barrière d'autocall, la courbe du Vega remonte progressivement vers zéro. En effet, lorsque le sous-jacent excède nettement 16,25\$, il est presque certain que l'autocall sera rappelé dès la première date, si bien qu'une variation de volatilité n'a plus d'impact significatif sur cette probabilité quasi-certaine. Au-delà de 25\$–30\$, la sensibilité à la volatilité devient pratiquement nulle, car le produit est déjà configuré pour un remboursement quasi automatique, indépendamment de la dispersion potentielle des trajectoires.



## Theta

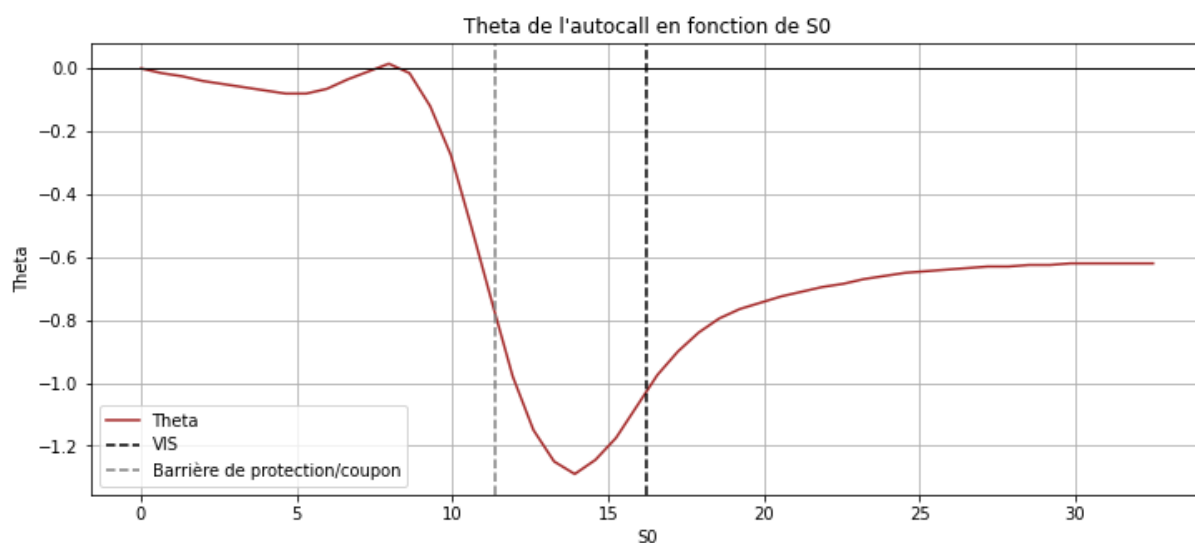


Figure 6 – Theta de l'autocall en fonction de  $S_0$

La courbe du Theta est clairement presque toujours négative, ce qui signifie qu'en l'absence de mouvement du sous-jacent, le passage du temps fait inévitablement baisser la valeur de l'autocall. Cette dépréciation temporelle correspond à l'érosion de la valeur-temps : plus l'échéance approche, moins de scénarios restent possibles pour le versement de coupons futurs ou pour le remboursement anticipé, si bien que la valeur actuelle du produit décroît.

On note toutefois un bref léger passage au-dessus de zéro pour des valeurs de  $S_0$  situées juste en dessous de la barrière de protection. Dans cette zone, diminuer d'un jour la maturité peut parfois augmenter marginalement le prix de l'autocall de manière contre-intuitive. Cet effet peut s'expliquer de la manière suivante. Tant que la maturité est longue, il existe une probabilité non négligeable que la trajectoire du sous-jacent chute davantage sous la barrière à au moins une des prochaines dates, ce qui exclurait tout versement de coupon futur et pourrait aboutir à un remboursement partiel. En raccourcissant légèrement la maturité, on réduit le nombre d'opportunités pour le sous-jacent de retomber en-dessous de 11,38\$. Pour ces niveaux du sous-jacent, cet effet l'emporte temporairement sur l'effet de la perte de valeur-temps intrinsèque liée à la disparition d'un jour de maturité, donnant un Theta faiblement positif.

A mesure que le sous-jacent s'approche de la barrière d'autocall, le Theta plonge brutalement pour atteindre son minimum à équidistance entre la barrière de protection et celle d'autocall. Lorsque le sous-jacent se situe à égale distance entre la barrière de protection et d'autocall, l'incertitude sur l'évolution du produit est la plus marquée. De ce fait c'est aussi à ce niveau que la valeur-temps de l'autocall est la plus élevée. Dans cette région, un léger mouvement du sous-jacent peut faire basculer le produit d'un état dans lequel l'autocall est peu probable à un état où il est très probable. Cette même incertitude implique

qu'à chaque instant passé, la probabilité de bénéficier d'un mouvement favorable ultérieur diminue, réduisant la valeur-temps de l'autocall, ce qui conduit à un Theta minimal.

## Rho

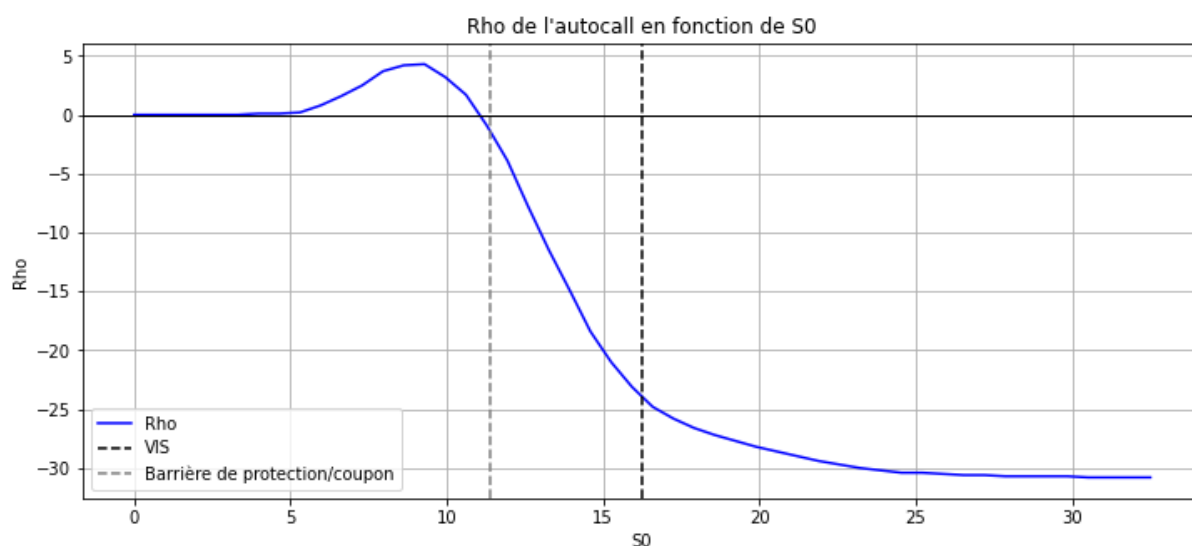


Figure 7 – Rho de l'autocall en fonction de  $S_0$

Le graphique du Rho montre comment la valeur de l'autocall réagit, pour chaque  $S_0$ , à une variation du taux sans risque. Deux effets principaux du taux sans risque sur le prix de l'autocall permettent d'expliquer la forme de cette courbe. Premièrement, l'effet le plus évident repose sur la valeur du forward. Pour rappel, le forward a comme formule  $F = S_0 e^{(r-q)T}$ . Donc lorsque les taux d'intérêt augmentent, le forward augmente, et donc la valeur de l'autocall également. Le second effet repose sur l'actualisation des flux futurs générés par l'autocall. Chaque versement futur est actualisé par le facteur  $e^{-rT}$ , cela signifie que lorsque les taux d'intérêt augmentent, la valeur actualisée des flux futurs diminue, ce qui vient réduire la valeur de l'autocall. La courbe obtenue dans le graphique est le résultat de l'agrégation de ces deux effets. On observe que globalement, l'effet de l'actualisation l'emporte sur l'effet sur le forward dans notre cas.

On observe d'abord que, pour des valeurs de  $S_0$  très faibles, le Rho est proche de zéro. Dans cette zone, l'autocall ne prévoit de verser aucun coupon et, au pire, prévoit de rembourser une somme proportionnelle à  $S_T$ . Les flux futurs étant très faibles et incertains, une variation du taux d'actualisation n'affecte presque pas la valeur actualisée attendue, l'effet de l'actualisation compense alors l'effet sur le forward, d'où un Rho négligeable.

Plus le sous-jacent s'approche du niveau de la barrière, plus l'effet de l'augmentation du taux sur le forward prend de l'importance puis atteint son maximum car une telle augmentation permettrait de franchir la barrière de protection et d'obtenir potentiellement le versement d'un coupon à la prochaine date d'observation. C'est la raison pour laquelle on peut observer

un sommet positif du Rho de la structure aux alentours de  $S_0 = 9\$$ . Ensuite, l'effet du taux d'intérêt sur le forward diminue progressivement, alors que l'effet du taux sur l'actualisation augmente une fois la barrière de coupon franchie, au vu de l'augmentation des revenus espérés par l'investisseur. Graphiquement, on observe en conséquence une chute assez rapide du Rho global qui prend alors des valeurs négatives, traduisant la perte de valeur induite par une hausse des taux à travers la hausse du taux d'actualisation des flux futurs.

Pour de grandes valeurs du sous-jacent, le Rho global se stabilise. Cela s'explique par le fait qu'au-delà de la barrière d'autocall, le produit sera quasiment certain d'être rappelé à la prochaine date, générant un flux fixe (coupon + remboursement) qui est actualisé (jusqu'à la maturité finale de 2,5 ans). Dans ce cas, toute hausse de taux réduit la valeur présente de l'autocall.

## Epsilon

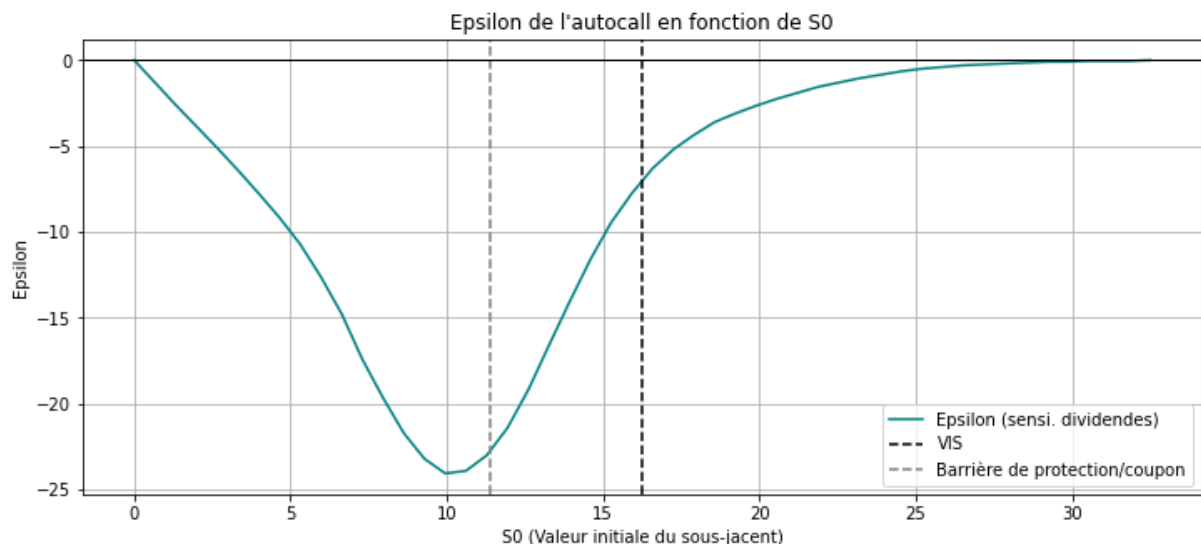


Figure 8 - Epsilon de l'autocall en fonction de  $S_0$

La courbe d'Epsilon, qui mesure la sensibilité du prix de l'autocall au rendement du dividende, présente un profil marqué en trois phases. Pour des  $S_0$  très faibles (en-dessous de quelques dollars), la valeur de l'autocall est proche de zéro, et la moindre variation du dividende n'affecte pas la probabilité de versement futur. En s'approchant de la barrière de protection, l'Epsilon devient rapidement très négatif, atteignant son minimum légèrement avant la barrière de protection. Dans cette zone, le sous-jacent est suffisamment élevé pour espérer toucher au moins un coupon, mais encore trop bas pour déclencher l'autocall dès la prochaine date. Une hausse du dividende réduit le drift net ( $r - q$ ) et donc diminue la probabilité que le cours grimpe jusqu'à la barrière de protection ou à la barrière d'autocall dans les semestres suivants ; en conséquence, les scénarios où l'investisseur perçoit plusieurs coupons ou est rappelé prématurément s'érodent. Cet effet se traduit par une

chute marquée de la valeur de l'autocall quand le taux de dividende augmente, d'où cet Epsilon très négatif à ce niveau.

Lorsque  $S_0$  franchit la barrière de protection, l'Epsilon reste encore fortement négatif mais s'atténue légèrement. À ce stade, le sous-jacent est dans l'intervalle critique entre la barrière de protection et la barrière d'autocall. Chaque augmentation supplémentaire du sous-jacent accroît encore la probabilité d'être rappelé avant maturité, mais l'impact d'un dividende plus élevé se fait toujours sentir car il peut faire basculer certains scénarios de knock-out prématuré en scénarios sans coupon. À l'approche de la barrière d'autocall, l'Epsilon remonte progressivement vers 0. Cela reflète le fait que, plus le sous-jacent est proche de la barrière d'autocall, moins le dividende importe : même avec un léger accroissement du taux de dividende, la probabilité de franchir 16,25 \$ dès la prochaine date de fixation reste élevée.

Enfin, pour  $S_0$  bien au-delà de 16,25 \$, l'Epsilon poursuit sa remontée vers zéro et s'y stabilise. Dans cette zone, l'autocall est presque certain d'être rappelé prochainement indépendamment du rendement du dividende, si bien qu'une variation du taux de dividende n'a plus qu'un effet négligeable sur la valeur actuelle du produit. L'Epsilon tend donc vers zéro pour les  $S_0$  les plus élevés, confirmant que, une fois le scénario de knock-out quasi-certain, la sensibilité aux dividendes disparaît.

## Analyse des barrières

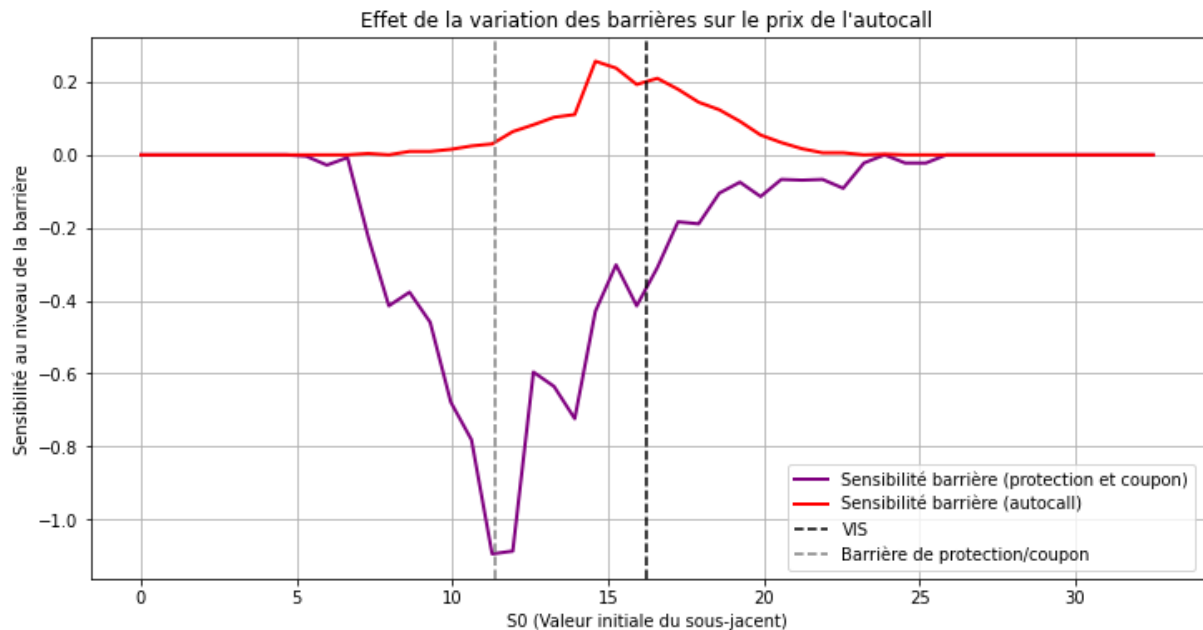


Figure 9 - Effet de la variation des barrières sur le prix de l'autocall

L'essentiel à retenir de ces deux courbes de sensibilité est que la barrière de protection a un impact nettement plus fort sur le prix de l'autocall que la barrière d'autocall. Dans la zone où  $S_0$  se situe autour de la barrière de protection et de coupon, même un petit déplacement de ce seuil fait chuter la valeur du produit, car ce changement élimine brutalement de nombreux scénarios où plusieurs coupons auraient pu être versés avant un éventuel knockout. À l'inverse, la sensibilité à la barrière d'autocall reste positive mais beaucoup plus modérée : on observe un pic d'environ +0,25 juste avant 16,25\$, où relever cette barrière rend légèrement plus incertain le knock-out au prochain semestre, mais l'effet reste loin d'égaliser l'impact de la barrière de protection. De plus, dès que  $S_0$  s'éloigne de ces seuils, les deux sensibilités se rapprochent rapidement de zéro. En-dessous, modifier un seuil n'influence que très peu la probabilité de versement de coupon ou de remboursement partiel, et au-delà, l'autocall est quasi-certain d'être rappelé prochainement. Finalement, cette analyse montre que le calibrage de la barrière de protection peut être ici un levier déterminant pour le structurer, alors que déplacer la barrière d'autocall n'affecte que marginalement la valeur.

## Conclusion

Au terme de ce travail, nous avons d'abord posé les fondations de l'étude en décrivant la structure et le fonctionnement d'un Phoenix, en détaillant le rôle fondamental des trois barrières et des mécanismes de versement de coupons conditionnels. Nous avons ensuite montré comment, sous les hypothèses du modèle de Black-Scholes, la méthode de Monte Carlo permet d'estimer de manière flexible et robuste le prix d'un tel produit, en actualisant la moyenne de milliers de trajectoires. Le calibrage des paramètres (maturité, volatilité, taux sans risque, dividende, niveaux de barrières, fréquence des observations) a conduit à un prix de 15,21\$ pour notre autocall NVDA, confirmant l'existence d'une décote dans la tarification contractuelle.

L'analyse approfondie des grecques par différences finies a ensuite mis en lumière les zones de sensibilité extrême du produit : le Delta et le Gamma subissent des variations importantes aux abords des barrières, le Vega reste globalement négatif avec un léger passage en positif en dessous de la barrière de protection, le Theta dévoile un creux marqué en zone intermédiaire et un plateau lié au mécanisme d'actualisation, tandis que le Rho et l'Epsilon illustrent la dépendance respectivement aux taux d'intérêt et aux dividendes avec des amplitudes variables selon la proximité des seuils critiques. L'étude de la sensibilité des prix aux ajustements des barrières a révélé quant à elle que la barrière de protection pèse jusqu'à cinq fois plus lourdement sur la valeur que la barrière d'autocall, ce qui représente un point de levier essentiel pour les structuriers et les traders.

Ces résultats n'ont pas uniquement une valeur académique : ils constituent un guide opérationnel pour la couverture dynamique des autocalls du même type. En identifiant précisément les intervalles de  $S_0$  où le Delta et le Gamma sont maximaux, le trader peut adapter la fréquence et la taille de ses ajustements de couverture delta-neutre. Le profil du Vega alerte sur la nécessité d'ajouter ou de retirer des positions en volatilité lorsque le sous-jacent se situe à mi-distance entre les barrières, et les courbes de Theta insistent sur l'importance de la gestion du risque temporel en fin de vie. Enfin, les sensibilités Rho et Epsilon rappellent que les évolutions des taux et des dividendes doivent être intégrées au programme de hedging pour éviter des décalages de valorisation. Bien que les travaux présentés portent sur un exemple typique d'autocall, il existe une multitude de configurations différentes qui entraîneraient des résultats parfois différents que ceux obtenus dans le cadre de ce mémoire. Néanmoins, la méthode et les diagnostics numériques présentés ici peuvent directement alimenter les pratiques de risk management au sein des desks de trading, offrant un outil de prédiction et de calibration des couvertures face aux Phoenix autocallables.

## Annexe - Code

Les scripts ci-dessous implémentent les méthodes de simulation utilisées dans ce mémoire pour le pricing d'un Phoenix autocallable sous le modèle de Black-Scholes, ainsi que pour l'analyse de sensibilité. Le code permet de générer des trajectoires du sous-jacent via un mouvement brownien géométrique sous mesure risque-neutre, de calculer la valeur actualisée du produit, puis d'en estimer les sensibilités (ou grecques) par la méthode des différences finies. De nombreux commentaires ont été ajoutés de manière à faciliter la lecture et la compréhension des résultats présentés dans les chapitres du mémoire.

```
import numpy as np

import pandas as pd

import matplotlib.pyplot as plt


r0 = 0.04

q0 = 0.02 #Rendement du dividende

sigma0 = 0.2

T = (pd.to_datetime('2025-06-17')- pd.to_datetime('2023-06-17')).days / 365.25

m = 5 #Nombre de dates d'observation semi-annuelles restantes

N = 10000 #Nombre de simulations


VIS = 16.25

barriere_autocall = 16.25 #100%*VIS

barriere_coupon = 11.375 #Barrière de coupon / protection du capital (70%)

I0 = 16.25 # Montant investi initialement

C = 0.025 * I0 # Montant du coupon


# La fonction de simulation GBM

def simulation_traj(S0, r, q, sigma, T, m, N):

    dt = T / m
```

```

S = np.zeros((m + 1, N))
S[0] = S0
for t in range(1, m + 1):
    Z = np.random.standard_normal(N)
    S[t] = S[t - 1] * np.exp((r - q - 0.5 * sigma**2) * dt + sigma * np.sqrt(dt) * Z)
return S

```

# Fonction de pricing de l'autocall en fonction d'un S0 initial variant

# Ici, même si S0 change, les paramètres d'émission restent fixes.

```
def autocall_price_for_S0_fixed(S0, r=r0, q=q0, sigma=sigma0, T=T, m=m, N=N):
```

```
    # Paramètres fixes d'émission (non scalés avec S0)
```

```
    VIS_local = VIS
```

```
    barriere_autocall_local = barriere_autocall
```

```
    barriere_coupon_local = barriere_coupon
```

```
    I0_local = I0
```

```
    C_local = C
```

```
    np.random.seed(42) # Pour la reproductibilité
```

```
    # Ici, on démarre la simulation avec S0 variable
```

```
    traj = simulation_traj(S0, r, q, sigma, T, m, N)
```

```
    coupons_payes = np.zeros(N)
```

```
    for i in range(N):
```

```
        for t in range(1, m + 1):
```

```
            if traj[t, i] >= barriere_autocall_local:
```

```
                coupons_payes[i] += C_local
```

```
                break # Dès l'autocall, on arrête d'évaluer la trajectoire
```

```
            elif traj[t, i] >= barriere_coupon_local:
```

```
                coupons_payes[i] += C_local
```

```
    # Calcul du remboursement final avec protection du capital
```



```

IF = np.where(traj[-1] < barriere_coupon_local, I0_local * (traj[-1] / VIS_local), I0_local)

val_totale = IF + coupons_payes

valeur_actuelle = np.exp(-r * T) * val_totale

return round(np.mean(valeur_actuelle), 2)

print("prix de l'autocall : ", autocall_price_for_S0_fixed(VIS))

```

---

```

# Analyse de sensibilité par rapport à S0 (delta)

# On fait varier la valeur initiale du sous-jacent sur une grille

S0_range = np.linspace(0, 2*16.25, 50)

prices = np.array([autocall_price_for_S0_fixed(s0) for s0 in S0_range])

# Calcul du Delta à l'aide du gradient

delta = np.gradient(prices, S0_range)

plt.figure(figsize=(12, 5))

# Graphique du prix de l'autocall en fonction de S

plt.plot(S0_range, prices, marker='', linestyle='-', color='blue', lw=2)

plt.axvline(x=VIS, color='black', linestyle='--', label='S0')

plt.axvline(x=barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')

plt.axhline(y=0, color='k', linestyle='-', lw=1)

plt.xlabel('S0 (Valeur initiale du sous-jacent)')

plt.ylabel('Prix de l\'autocall')

plt.title('Prix de l\'autocall en fonction de la valeur du sous-jacent')

plt.legend()

plt.grid(True)

```

```

plt.figure(figsize=(12, 5))

# Graphique du Delta en fonction de S
plt.plot(S0_range, delta, marker='', linestyle='-', color='red', label='Delta', lw=2)

plt.axvline(x=VIS, color='black', linestyle='--', label='Barrière d\'autocall')

plt.axvline(x=barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')

plt.axhline(y=0, color='k', linestyle='-', lw=1)

plt.xlabel('S0 (VIS)')

plt.ylabel('Delta')

plt.title('Delta de l\'autocall')

plt.legend()

plt.grid(True)

```

```
plt.show()
```

---

```
# Calcul du Gamma
```

```
gamma = np.gradient(delta, S0_range)
```

```
# Affichage du graphique du Gamma en fonction de S0
```

```

plt.figure(figsize=(12, 5))

plt.plot(S0_range, gamma, marker='', linestyle='-', color='orange', label='Gamma', lw=2)

plt.axvline(x=VIS, color='black', linestyle='--', label='Barrière d\'autocall')

plt.axvline(x=barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')

plt.axhline(y=0, color='k', linestyle='-', lw=1)

plt.xlabel('S0 (Valeur initiale du sous-jacent)')

plt.ylabel('Gamma')

plt.title('Gamma de l\'autocall')

plt.legend()

```

```
plt.grid(True)
```

```
plt.show()
```

---

```
# Calcul du Vega en fonction de S0
```

```
def autocall_vega(S0, sigma_ref, dsigma=0.05, r=r0, q=q0, T=T, m=m, N=N):
```

```
    price_plus = autocall_price_for_S0_fixed(S0, sigma = sigma0 + dsigma, r=r0, q=q0, T=T,
m=m, N=N)
```

```
    price_minus = autocall_price_for_S0_fixed(S0, sigma = sigma0 - dsigma, r=r0, q=q0, T=T,
m=m, N=N)
```

```
    return (price_plus - price_minus) / (2 * dsigma)
```

```
# Utilisation de la fonction ci-dessus pour calculer le vega en fonction de S0
```

```
# On définit une grille de valeurs pour S0
```

```
S0_range_vega = np.linspace(0, 3*16.25, 50)
```

```
# ici la volatilité de référence est 0.2
```

```
# Calcul du vega pour chaque valeur de S0 de la grille
```

```
vega_S0 = np.array([autocall_vega(s0, sigma_ref=sigma0, dsigma=0.05) for s0 in
S0_range_vega])
```

```
plt.figure(figsize=(12, 5))
```

```
plt.plot(S0_range_vega, vega_S0, marker='', linestyle='-', color='magenta', label='Vega', lw=2)
```

```
plt.axvline(x=VIS, color='black', linestyle='--', label='VIS')
```

```
plt.axvline(x=barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')
```

```
plt.axhline(y=0, color='k', linestyle='-', lw=1)
```

```
plt.xlabel('S0 (Valeur initiale du sous-jacent)')
```

```

plt.ylabel('Vega')
plt.title("Vega de l'autocall en fonction de S0")
plt.legend()
plt.grid(True)
plt.show()

```

---

```

#Theta

# Fonction pour calculer le theta de l'autocall (sensibilité du prix par rapport à T)
def autocall_theta(S0, T_ref, dT=1, r=r0, q=q0, sigma=sigma0, m=m, N=N):
    price_plus = autocall_price_for_S0_fixed(S0, r=r0, q=q0, sigma=sigma0, T=T + dT, m=m,
N=N)
    price_minus = autocall_price_for_S0_fixed(S0, r=r0, q=q0, sigma=sigma0, T=T - dT, m=m,
N=N)
    theta_val = (price_plus - price_minus) / (2 * dT)
    return theta_val

theta_S0 = np.array([autocall_theta(s, T_ref=T) for s in S0_range])

plt.figure(figsize=(12, 5))
plt.plot(S0_range, theta_S0, marker='', linestyle='-', color='brown', label='Theta')
plt.axvline(x=VIS, color='black', linestyle='--', label='VIS')
plt.axvline(x=barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')
plt.axhline(y=0, color='k', linestyle='-', lw=1)
plt.xlabel('S0')
plt.ylabel('Theta')
plt.title("Theta de l'autocall en fonction de S0")
plt.legend()
plt.grid(True)

```

```
plt.show()
```

---

```
# Fonction pour calculer le Rho de l'autocall (sensibilité du prix par rapport à r)
```

```
def autocall_rho(S0, r_ref, dr=0.05, q=q0, sigma=sigma0, T=T, m=m, N=N):
```

```
    price_plus = autocall_price_for_S0_fixed(S0, r=r0 + dr, q=q0, sigma=sigma0, T=T, m=m,
    N=N)
```

```
    price_minus = autocall_price_for_S0_fixed(S0, r=r0 - dr, q=q0, sigma=sigma0, T=T, m=m,
    N=N)
```

```
    return (price_plus - price_minus) / (2 * dr)
```

```
rho_S0 = np.array([autocall_rho(s0, r_ref=r0) for s0 in S0_range])
```

```
plt.figure(figsize=(12, 5))
```

```
plt.plot(S0_range, rho_S0, marker='', linestyle='-', color='blue', label='Rho')
```

```
plt.axvline(x=VIS, color='black', linestyle='--', label='VIS')
```

```
plt.axvline(x=barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')
```

```
plt.axhline(y=0, color='k', linestyle='-', lw=1)
```

```
plt.xlabel('S0')
```

```
plt.ylabel('Rho')
```

```
plt.title("Rho de l'autocall en fonction de S0")
```

```
plt.legend()
```

```
plt.grid(True)
```

```
plt.show()
```

---

```
# Fonction pour calculer l'epsilon de l'autocall
```

```
def autocall_epsilon(S0, q_ref, dq=0.1, r=r0, sigma=sigma0, T=T, m=m, N=N):
```

```
    price_plus = autocall_price_for_S0_fixed(S0, r=r0, q=q0+dq, sigma=sigma0, T=T, m=m,
    N=N)
```

```
    price_minus = autocall_price_for_S0_fixed(S0, r=r0, q=q0-dq, sigma=sigma0, T=T, m=m,
    N=N)
```

```

    return (price_plus - price_minus) / (2 * dq)

epsilon_S0 = np.array([autocall_epsilon(s, q_ref=q0) for s in S0_range])

plt.figure(figsize=(12, 5))

plt.plot(S0_range, epsilon_S0, marker='', linestyle='-', color='teal', label='Epsilon (sensi.
dividendes)')

plt.axvline(x=VIS, color='black', linestyle='--', label='VIS')

plt.axvline(x=barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')

plt.axhline(y=0, color='k', linestyle='-', lw=1)

plt.xlabel('S0 (Valeur initiale du sous-jacent)')

plt.ylabel('Epsilon')

plt.title("Epsilon de l'autocall en fonction de S0")

plt.legend()

plt.grid(True)

plt.show()

```

---

```

# Afficher le graphique des données historiques de prix de Nvidia

df = pd.read_csv("data.csv")

df["Date"] = pd.to_datetime(df["Date"])

plt.figure(figsize=(12, 6))

plt.plot(df["Date"], df["Price"], marker='', linestyle='-', label='NVDA Spot Price')

plt.xlabel("Date")

plt.ylabel("NVDA Prix Spots (USD)")

plt.title("NVDA Prix Spots et barrières")

plt.axhline(y = VIS, color='black', linestyle='--', label='Barrière d\'autocall')

```

```
plt.axhline(y = barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')

plt.xticks(rotation=45)

plt.grid(True)

plt.legend()

plt.show()
```

---

#Calcul de sensibilité aux différentes barrières

```
def autocall_price(S0, barriere_coupon, barriere_autocall, r=r0, q=q0, sigma=sigma0, T=T,
m=m, N=N):

    VIS_local = VIS

    IO_local = IO

    Qj_local = C

    np.random.seed(42)

    traj = simulation_traj(S0, r, q, sigma, T, m, N)

    coupons_payes = np.zeros(N)

    for i in range(N):

        for t in range(1, m+1):

            if traj[t, i] >= barriere_autocall:

                coupons_payes[i] += Qj_local

                break # Dès l'autocall, on arrête l'évaluation pour cette trajectoire

            elif traj[t, i] >= barriere_coupon:

                coupons_payes[i] += Qj_local

    IF = np.where(traj[-1] < barriere_coupon,

                  IO_local * (traj[-1]/VIS_local),

                  IO_local)

    val_totale = IF + coupons_payes

    valeur_actuelle = np.exp(-r * T) * val_totale

    return np.mean(valeur_actuelle)
```

```

def sensi_coupon(S0, barrier_ref, d_barrier=0.01, **kwargs):
    price_plus = autocall_price(S0, barriere_coupon=barrier_ref + d_barrier,
                                barriere_autocall=barriere_autocall, **kwargs)
    price_minus = autocall_price(S0, barriere_coupon=barrier_ref - d_barrier,
                                barriere_autocall=barriere_autocall, **kwargs)
    return (price_plus - price_minus) / (2 * d_barrier)

# Fonction de sensibilité par différences finies pour la barrière d'autocall.
# Ici, on fait varier barriere_autocall, en gardant la barrière de coupon fixe.
def sensi_autocall(S0, barrier_ref, d_barrier=0.01, **kwargs):
    price_plus = autocall_price(S0, barriere_coupon=barriere_coupon,
                                barriere_autocall=barrier_ref + d_barrier, **kwargs)
    price_minus = autocall_price(S0, barriere_coupon=barriere_coupon,
                                barriere_autocall=barrier_ref - d_barrier, **kwargs)
    return (price_plus - price_minus) / (2 * d_barrier)

S0_range = np.linspace(0, 2*16.25, 50)

# Calcul de la sensibilité pour chaque S0 pour la barrière de coupon, puis celle d'autocall
coupon_sensi = np.array([sensi_coupon(s0, barriere_coupon, d_barrier=0.01, r=r0, q=q0,
sigma=sigma0, T=T, m=m, N=N)
                           for s0 in S0_range])

autocall_sensi = np.array([sensi_autocall(s0, barriere_autocall, d_barrier=0.01, r=r0, q=q0,
sigma=sigma0, T=T, m=m, N=N)
                           for s0 in S0_range])

# Tracé des deux courbes sur le même graphique

```



```

plt.figure(figsize=(12, 6))

plt.plot(S0_range, coupon_sensi, linestyle='-', color='purple', lw=2, label="Sensibilité barrière
(protection et coupon)")

plt.plot(S0_range, autocall_sensi, linestyle='-', color='red', lw=2, label="Sensibilité barrière
(autocall)")

plt.axvline(x=VIS, color='black', linestyle='--', label='VIS')

plt.axvline(x=barriere_coupon, color='grey', linestyle='--', label='Barrière de
protection/coupon')

plt.xlabel("S0 (Valeur initiale du sous-jacent)")

plt.ylabel("Sensibilité au niveau de la barrière")

plt.title("Effet de la variation des barrières sur le prix de l'autocall")

plt.legend()

plt.grid(True)

plt.show()

```

## Sources

1. **EDHEC Business School.** (2024-2025). *Introduction to Derivatives* [Cours, par Laurent Deville].
2. **Reitano, R. R.** (2010). *Introduction to Quantitative Finance: A Math Tool Kit*. Massachusetts Institute of Technology.
3. **Hull, J.** (2021). *Options, Futures, and Other Derivatives* (Global Edition). Pearson.
4. **Quantlab.** (2018). *AutoCallable Product – Valuation and Risk* [Vidéo]. YouTube.
5. **Bouzoubaa, A.** (2010). *Exotic Options and Hybrids: A Guide to Structuring, Pricing and Trading*.
6. **de Weert, F.** (2008). *Exotic Options Trading*.