

Modélisation de copules dynamiques bivariées par deep learning et application au pricing d'options via Monte Carlo

Un mémoire présenté par :

Tristan Voirin et Rémi Schmitt

Encadrant :

Jérôme Mathis



Année universitaire 2024-2025

M1 EIF

Table des matières

1	Introduction	2
2	Théorie	4
2.1	Définitions	4
2.2	Copules classiques	6
2.2.1	Copule empirique	6
2.2.2	Copules elliptiques	6
2.2.3	Copules archimédiennes	8
2.3	Méthodes d'estimation des paramètres de la copule	10
2.3.1	Méthodes des moments	10
2.3.2	Méthodes basées sur le maximum de vraisemblance	11
2.3.3	Approche par deep learning	13
2.4	Simulations de Monte Carlo avec copule dynamique	13
2.4.1	Fondamentaux	13
2.4.2	Mise en place	14
3	Estimation du modèle Transformeur	16
3.1	Formatage des features	16
3.2	Hyperparamètres et entraînement	16
4	Résultats des simulations	18
4.1	Modèle à copule dynamique	18
4.2	Comparaison avec d'autres modèles	21
5	Discussion du modèle à copule dynamique	26
5.1	Avantages	26
5.2	Limites	27
5.3	Perspectives	28
6	Conclusion	29
	Appendice	31
	Bibliographie	36

1 Introduction

Depuis les années 1970, les travaux fondateurs de Black, Scholes et Merton ont permis un essor considérable de la finance quantitative, notamment pour la valorisation d'options et de produits dérivés. Toutefois, ces modèles classiques se concentrent principalement sur un actif isolé, sans prendre en compte les relations d'interdépendance entre plusieurs actifs financiers. Or, dans un monde financier de plus en plus interconnecté, cette simplification constitue une limite importante.

Parmi les premières tentatives de modélisation de la dépendance entre actifs, les coefficients de corrélation de Pearson et de Spearman sont les plus utilisés. Le coefficient de Pearson, par exemple, est défini par :

$$\rho_p = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}},$$

tandis que celui de Spearman repose sur les rangs des observations :

$$\rho_s = 1 - \frac{6 \sum_{i=1}^n d_i^2}{n(n^2 - 1)}.$$

Ces mesures, bien que simples à interpréter, présentent des limites notables : elles peinent à capturer des dépendances non linéaires, asymétriques, ou concentrées dans les queues de distribution — précisément là où se logent les événements extrêmes les plus pertinents en gestion des risques.

C'est dans ce contexte qu'interviennent les copules, introduites par Sklar en 1959¹. Une copule permet de séparer la modélisation des lois marginales de celle de la structure de dépendance, ce qui en fait un outil puissant pour étudier les co-mouvements complexes entre variables aléatoires. En finance, les copules paramétriques statiques — telles que les copules de Clayton, Gumbel ou Student — sont largement utilisées pour modéliser des formes spécifiques de dépendance. Néanmoins, leur principal défaut réside dans leur nature stationnaire : elles supposent une relation de dépendance fixe dans le temps, alors que les marchés évoluent constamment.

Face à cette rigidité, les modèles à copule dynamique proposent une alternative plus réaliste, en permettant à leurs paramètres de varier au fil du temps. Ces modèles offrent une meilleure représentation de l'évolution temporelle des relations entre actifs. Toutefois, leur estimation peut s'avérer complexe, notamment lorsque les données présentent des comportements non linéaires ou lorsque la dimension du problème augmente.

C'est ici que les méthodes d'apprentissage automatique — et en particulier le deep learning — offrent de nouvelles perspectives. Grâce à leur capacité à capturer des structures complexes dans des données temporelles, ces méthodes permettent d'estimer de manière

¹Sklar, A. (1959). Fonctions de répartition à marges données. *Université Paris*, 8(3), 229-231.

flexible et dynamique les paramètres d'une copule, même dans des contextes hautement non linéaires.

Dans ce mémoire, nous proposons une méthode de simulation du prix d'options sur panier fondée sur une copule de Student à paramètres dynamiques, estimés à l'aide d'un modèle Transformeur entraîné sur des séries temporelles. L'estimation est intégrée dans un algorithme de Monte Carlo afin de simuler les rendements conjoints des actifs sous-jacents.

Nous comparons les résultats obtenus avec ceux issus de deux méthodes classiques : l'utilisation d'un modèle de Black-Scholes multivarié et celle d'une copule statique. Cette approche vise à répondre à la question suivante :

Dans quelle mesure l'utilisation d'une copule dynamique calibrée par apprentissage profond permet-elle d'améliorer la simulation et l'évaluation d'options multi-actifs comparé au modèle de Black-Scholes et à un modèle à copule statique ?

2 Théorie

2.1 Définitions

Sklar a établi en 1959 qu'une copule bivariée est une fonction qui relie les distributions marginales de variables aléatoires pour modéliser leur dépendance.

$$F(x_1, \dots, x_d) = C(F_1(x_1), \dots, F_d(x_d))$$

Dans le cas bivarié, considérons un vecteur (X_1, X_2) de variables aléatoires dont les fonctions de répartition marginales, notées F_1 et F_2 , sont continues. On sait grâce à la transformation par probabilité intégrale² que le vecteur $(U_1, U_2) = (F_1(X_1), F_2(X_2))$ possède des marginales uniformément distribuées sur $[0, 1]$. La copule C est alors définie comme la fonction de répartition jointe de (U_1, U_2) , soit :

$$C(u_1, u_2) = P[U_1 < u_1, U_2 < u_2]$$

Elle représente la probabilité que les deux variables, une fois transformées sur une échelle commune via leurs fonctions de répartition, soient simultanément inférieures à des seuils u_1 et u_2 compris entre 0 et 1³.

Cet outil mathématique possède notamment trois caractéristiques :

- $C : [0, 1] \times [0, 1] \rightarrow [0, 1]$ (i)
- $C(u_i, 0) = 0$ et $C(u_i, 1) = u_i$ (ii)
- $C(u_1, u_2)$ est croissante en u_1 et u_2 (iii)

Ces trois caractéristiques sont assez intuitives à comprendre. La condition (i) garantit que la copule est bien une fonction de répartition valide. La condition (ii) constitue une condition de bord et stipule que si une des variables u_i vaut 0, alors la probabilité que les deux variables uniformisées soient simultanément inférieures à leurs quantiles u_i respectifs est nulle. Et inversement, si u_i vaut 1, la probabilité dépend uniquement de l'autre variable, qui reste libre de varier dans $[0, 1]$. Enfin, la condition (iii) s'interprète comme une condition de croissance monotone, où si le quantile u_i augmente pour au moins une des variables, alors la probabilité associée à la copule augmente ou reste stable.

De plus, toute copule doit respecter les inégalités de Fréchet-Hoeffding dont les bornes sont les copules usuelles maximale (dite comonotone) et minimale :

$$\max(0, u_1 + u_2 - 1) \leq C(u_1, u_2) \leq \min(u_1, u_2)$$

²Voir Feickert (2021) pour une démonstration détaillée.

³Nelsen, R. B. (2006). *An Introduction to Copulas* (2nd ed.). Springer.

Un autre concept important dans le contexte de la modélisation par copule est celui de la dépendance de queues, qui décrit comment la dépendance entre les variables aléatoires évolue spécifiquement dans les queues de la distribution. On distingue donc la dépendance de queue supérieure (λ_u) et la dépendance de queue inférieure (λ_l) :

$$\lambda_U = \lim_{q \rightarrow 1^-} P(Y > F_Y^{-1}(q) \mid X > F_X^{-1}(q))$$

$$\lambda_L = \lim_{q \rightarrow 0^+} P(Y \leq F_Y^{-1}(q) \mid X \leq F_X^{-1}(q))$$

avec $F_i^{-1}(q)$ la fonction inverse de distribution de la variable i

L'idée est de mesurer la probabilité λ_l (λ_u) que la variable aléatoire Y soit inférieure (*supérieure*) à son quantile q lorsqu'il tend vers 0 (1) sachant que l'autre variable aléatoire X est déjà inférieure à son quantile du même ordre q . Typiquement, les indices de dépendances λ_u et λ_l de deux variables aléatoires parfaitement indépendantes (*corrélées*) doivent être égaux à 0 (1). Il existe également le cas où $\lambda_u \neq \lambda_l$. Par exemple, deux variables aléatoires peuvent être parfaitement indépendantes dans la queue supérieure de leur distribution, mais corrélées dans leur queue inférieure (ou inversement).⁴

⁴Schmidt, R. (2007). *Tail dependence*. University of Cologne.

2.2 Copules classiques

Dans cette section, nous exposons les copules usuelles et présentons leurs principales caractéristiques.

2.2.1 Copule empirique

La copule empirique se déduit directement depuis l'échantillon de données observées et ne repose sur aucun modèle théorique, contrairement aux copules présentées par la suite.

Pour un échantillon de n observations de deux variables (X_1, X_2) , on la définit :

$$C(u_1, u_2) = \frac{1}{n} \sum_{i=1}^n \mathbf{1} \left(\frac{R_{i1}}{(n+1)} \leq u_1, \frac{R_{i2}}{(n+1)} \leq u_2 \right)$$

où :

- R_{ij} est le rang de l'observation X_{ij} dans l'échantillon de la variable j
- $\mathbf{1}$ est une fonction indicatrice
- $(n+1)$ permet d'éviter d'atteindre les valeurs extrêmes 0 et 1

La force de cette copule est sa capacité à modéliser la dépendance entre variables sans faire d'hypothèses paramétriques, en s'appuyant uniquement sur leurs rangs. Cependant, son principal défaut est qu'elle reste une estimation empirique, ce qui peut la rendre imprécise, en particulier pour de petits échantillons et pour des dépendances complexes et dynamiques.

2.2.2 Copules elliptiques

Comme leur nom l'indique, les copules elliptiques reposent sur des distributions elliptiques, qui sont des extensions de la loi normale. Bien que cette famille soit vaste, la copule de Gauss et la copule de Student sont celles qui trouvent le plus d'applications en pratique.

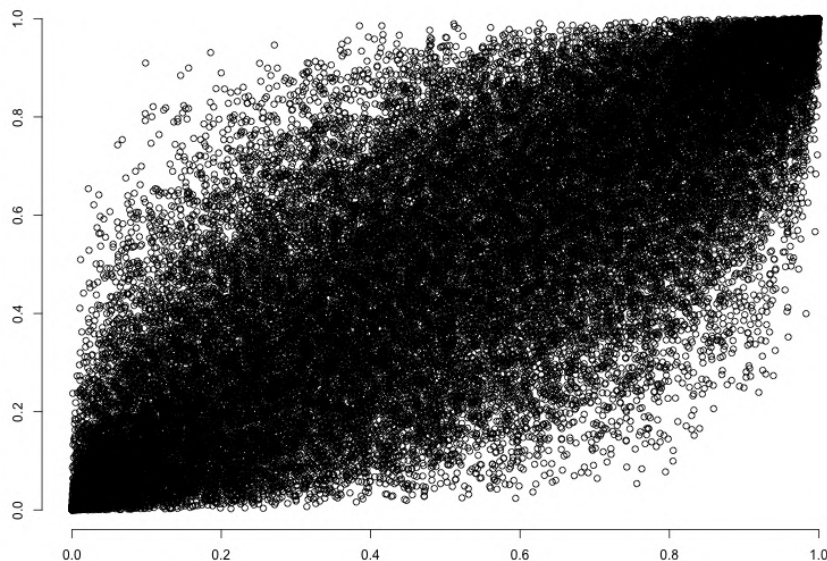
- **Copule de Gauss**

Popularisée par *David X. Li*⁵, la copule de Gauss était un outil extrêmement utilisé jusqu'à la crise de 2008 en raison de sa simplicité analytique et de la facilité avec laquelle elle peut être calibrée sur des données réelles. Elle prend en paramètre une matrice de corrélation, qui dans le cas bivarié, se réduit simplement à un coefficient de corrélation ρ .

On la définit alors :

$$C(u_1, u_2) = \Phi_\rho \left(\Phi^{-1}(u_1), \Phi^{-1}(u_2) \right)$$

Figure 1: *Copule Gaussienne* ($\rho = 0.8$)



où Φ_ρ est la fonction de répartition d'une loi normale bivariée avec corrélation ρ et Φ^{-1} l'inverse de la fonction de répartition d'une loi normale standard univariée.

Cette copule fut au cœur des polémiques après la crise de 2008, certains l'appelant même "*The formula that killed Wall Street*"⁶. La raison est qu'elle était massivement utilisée dans l'évaluation des risques et le pricing de CDO alors qu'elle ne présente pas de dépendance de queues (sauf dans le cas où $|\rho| = 1$). Cela a donc entraîné une sous-estimation des probabilités de défauts conjoints, amplifiant les pertes sur les CDOs⁷.

• Copule de Student

Cette copule est très proche de la copule de Gauss. La principale différence est qu'elle repose sur la loi de student multivariée et qu'elle intègre donc un deuxième paramètre ν gérant l'épaisseur des queues de distribution.

Elle se définit donc :

$$C(u_1, u_2) = T_\nu (\Phi^{-1}(u_1), \Phi^{-1}(u_2))$$

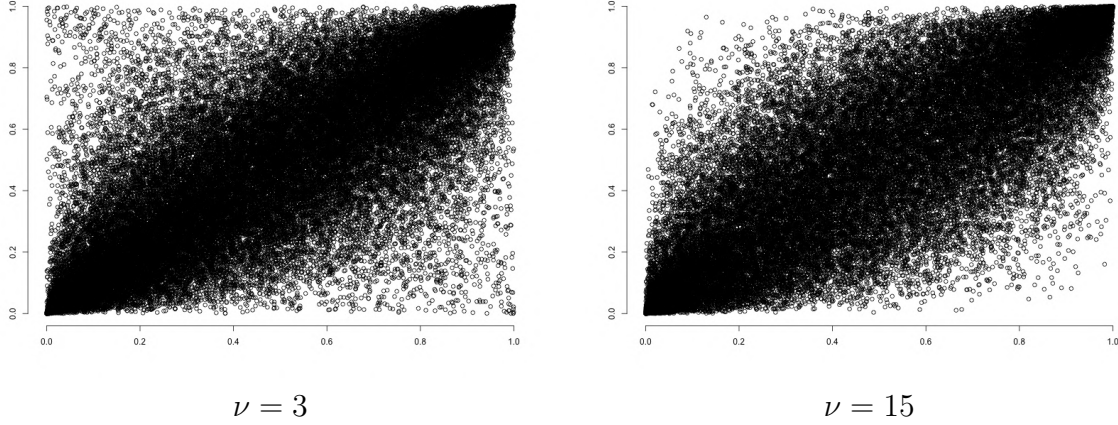
où T_ν est la fonction de répartition de la distribution de Student avec ν degrés de liberté. On peut voir cette copule comme une généralisation de la copule de Gauss car lorsque $\nu \rightarrow \infty$, la distribution de Student tend vers une distribution normale. En effet, plus

⁵Li, D. X. (2000). On Default Correlation: A Copula Function Approach. *Journal of Fixed Income*, 9(4), 43–54.

⁶Salmon, F. (2009). Recipe for Disaster: The Formula That Killed Wall Street. *WIRED*. Retrieved from <https://www.wired.com/2009/02/wp-quant/>

⁷Watts, S. (2016). The Gaussian Copula and the Financial Crisis: A Recipe for Disaster or Cooking the Books? *University of Oxford*. Retrieved from https://samueldwatts.com/wp-content/uploads/2016/08/Watts-Gaussian-Copula_Financial_Crisis.pdf

Figure 2: *Comparison de deux copules de Student avec $\rho = 0.8$*



le nombre de degrés de liberté ν est faible, plus la dépendance dans les queues de la distribution devient forte :

$$\lambda_U = \lambda_L = 2t_{\nu+1} \left(\sqrt{\frac{(\nu+1)(1-\rho)}{1+\rho}} \right), \quad (1)$$

Cette particularité a fait que cette copule a gagné en popularité après la crise des subprimes, au détriment de la copule de Gauss comme expliqué précédemment. Néanmoins aujourd'hui, elle n'est en pratique presque jamais utilisée seule, mais plutôt couplée avec d'autres outils mathématiques.

Nous avons ainsi sélectionné cette copule comme base de notre approche de pricing par copule dynamique avec apprentissage machine, en raison de sa simplicité, reposant sur seulement deux paramètres (ρ et ν), et de sa capacité à modéliser des dépendances symétriques dans les queues.

2.2.3 Copules archimédiennes

Cette section vise à évoquer les copules archimédiennes qui capturent des dépendances de queues asymétriques bien qu'elles ne seront pas utilisées par la suite.

Ces copules sont définies de la façon suivante :

$$C(u, v; \theta) = \varphi^{-1}(\varphi(u; \theta) + \varphi(v; \theta))$$

Ici, leur construction repose sur une fonction génératrice φ strictement décroissante, convexe, monotone et avec $\varphi(1) = 0$. Cette approche permet, contrairement aux copules elliptiques, de modéliser des dépendances de queues asymétriques. Le paramètre θ , appelé paramètre de dépendance, provient directement de la fonction génératrice φ qui définit la copule. Il contrôle l'intensité et parfois la direction de la dépendance entre les

variables. Chaque choix de φ — propre à une famille de copule (Clayton, Gumbel, etc.) — génère une structure de dépendance spécifique, et θ permet d'en ajuster la force. De plus, elles ont l'avantage d'avoir peu de paramètres (souvent un seul) ce qui leur offre une grande flexibilité⁸.

Il existe une multitude de copules archimédiennes, les plus populaires étant :

- la copule de Clayton :

$$C_{\text{Clayton}}(u_1, u_2) = \left(\max(u_1^{-\theta} + u_2^{-\theta} - 1, 0) \right)^{-1/\theta}, \quad \theta > 0$$

- la copule de Gumbel :

$$C_{\text{Gumbel}}(u_1, u_2) = \exp \left(- \left[(-\ln u_1)^\theta + (-\ln u_2)^\theta \right]^{1/\theta} \right), \quad \theta \geq 1$$

- la copule de Frank :

$$C_{\text{Frank}}(u_1, u_2) = -\frac{1}{\theta} \ln \left(1 + \frac{(e^{-\theta u_1} - 1)(e^{-\theta u_2} - 1)}{e^{-\theta} - 1} \right), \quad \theta \neq 0$$

Figure 3: *Comparaison des copules archimédiennes*

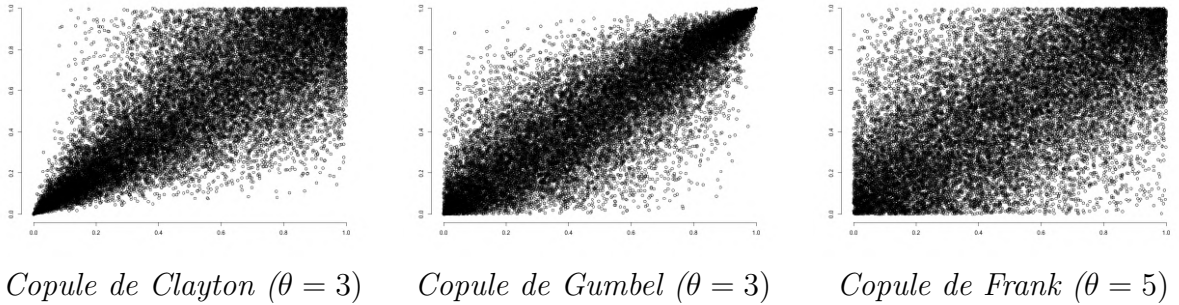


Table 1: Coefficients de dépendance de queue de copules archimédiennes

Copule	λ_L	λ_U
Clayton	$2^{-1/\theta}$	0
Gumbel	0	$2^{-1/\theta}$
Franck	$2^{-1/\theta}$	$2^{-1/\theta}$

En finance, ce type de copule est principalement utilisé dans la gestion des risques. Il se révèle moins adapté à une problématique de pricing d'options sur panier que la famille des copules elliptiques, bien qu'il fasse encore l'objet de recherches.

⁸Joe, H. (2014). *Dependence Modeling with Copulas*. CRC Press.

2.3 Méthodes d'estimation des paramètres de la copule

Dans cette section, le but est de présenter les méthodes usuelles d'estimation des paramètres statistiques. Toutefois, ne connaissant pas les lois usuelles de nos 2 variables aléatoires, la méthode utilisée par la suite est celle du maximum de vraisemblance canonique.

2.3.1 Méthodes des moments

La méthode des moments consiste à évaluer les moments théoriques de la copule aux moments empiriques observés dans les données. Cette méthode est souvent utilisée comme première approximation des paramètres en raison de sa simplicité et de l'existence d'autres méthodes plus précises.

On commence donc par estimer un moment empirique de dépendance à partir des variables uniformisées $U_1 = \frac{\text{rang}(X_1)}{n+1}$, $U_2 = \frac{\text{rang}(X_2)}{n+1}$ (comme durant l'estimation de la copule empirique). Ensuite, puisque nous travaillons avec une copule de Student, il est optimal de choisir comme moment le tau de Kendall du fait de l'existence de la relation :

$$\tau = \frac{2}{\pi} \arcsin(\rho), \quad (2)$$

où ρ est le paramètre de corrélation de la copule de Student.

Il est important de comprendre que l'estimation directe de ρ par la corrélation de Pearson est inadaptée car cette mesure est influencée par les distributions marginales des variables d'origine, ce qui peut biaiser la représentation de la dépendance pure capturée par la copule. En revanche, la corrélation de Kendall τ , calculée sur les rangs, est une mesure non paramétrique plus robuste qui reflète uniquement la structure de dépendance de la copule (à la manière de la corrélation de Spearman).

À partir de la valeur empirique $\hat{\tau}$ calculée sur les données (U_1, U_2) , on peut donc inverser cette relation pour estimer ρ . En résolvant l'équation, on obtient :

$$\hat{\rho} = \sin\left(\frac{\pi}{2} \hat{\tau}\right). \quad (3)$$

Cependant, la copule de Student dépend également d'un second paramètre, ν , le nombre de degrés de liberté, qui influence la dépendance dans les queues. Puisque τ ne dépend pas de ν , une étape supplémentaire est nécessaire pour l'estimer. L'approche la plus courante consiste à examiner la dépendance de queue empirique (λ), et à ajuster ν pour que la dépendance théorique, donnée par (1) corresponde à la valeur observée.

Cette méthode d'estimation perd de son intérêt lorsque l'on travaille avec des copules dynamiques où les paramètres changent à chaque pas de temps, car elle suppose des paramètres fixes. De plus dans le cas de la copule de Student, l'estimation de ν manque

de précision. C'est pour cela qu'on préfère en pratique utiliser d'autres méthodes.

2.3.2 Méthodes basées sur le maximum de vraisemblance

Les méthodes basées sur le maximum de vraisemblance (*MLE, Maximum Likelihood Estimation*) consistent à estimer les paramètres du modèle en maximisant la vraisemblance de l'échantillon observé, c'est-à-dire la probabilité conjointe des observations en fonction des paramètres du modèle. Dans le cadre des copules, il existe plusieurs variantes basées sur cette méthode.

Tout d'abord, la **méthode du full MLE** consiste à estimer simultanément tous les paramètres : ceux de la copule mais aussi ceux des marges. Cela passe par la maximisation de la (log) vraisemblance jointe.

Mathématiquement, pour un échantillon (X_1, X_2) avec des marges $(F_1(x_1; \theta), F_2(x_2; \theta))$ et une copule $C(u_1, u_2; \rho, \nu)$, la log-vraisemblance jointe s'écrit :

$$\mathcal{L}(\theta_1, \theta_2, \rho, \nu) = \sum_{j=1}^n \log c\left(F_1(x_{1,j}; \theta_1), F_2(x_{2,j}; \theta_2); \rho, \nu\right) + \sum_{j=1}^n \log f_1(x_{1,j}; \theta_1) + \sum_{j=1}^n \log f_2(x_{2,j}; \theta_2) \quad (4)$$

où $c(u_1, u_2; \rho, \nu)$ est la densité de la copule et f_1 et f_2 sont les densités marginales.

Cette méthode a l'avantage d'être très précise. Néanmoins, elle possède deux défauts majeurs : une complexité computationnelle très élevée ainsi qu'une nécessité de modéliser correctement les marges.

D'autres méthodes ont été développées pour contourner ces problèmes. La **méthode d'estimation par quasi-maximum de vraisemblance** (*QMLE, Quasi-Maximum Likelihood Estimation*) propose de maximiser une vraisemblance supposée, même si celle-ci ne correspond pas exactement à la vraie distribution des données :

$$\ell_{\text{QMLE}}(\rho, \nu) = \sum_{i=1}^n \log c(U_{i1}, U_{i2}; \rho, \nu)$$

où U_{i1} et U_{i2} sont des variables uniformes supposées, et c est une densité choisie. Cette approche offre une bonne robustesse face à une mauvaise spécification du modèle, mais elle peut perdre en précision si l'écart entre le modèle supposé et la réalité est trop important.

Une autre alternative est la **méthode des fonctions d'inférence pour les marginales** (*IFM*), qui adopte une stratégie en deux étapes :

1. Une estimation paramétrique des distributions marginales :

$$\hat{\theta}_1 = \arg \max_{\theta_1} \sum_{i=1}^n \log f_1(X_{i1}; \theta_1), \quad \hat{\theta}_2 = \arg \max_{\theta_2} \sum_{i=1}^n \log f_2(X_{i2}; \theta_2)$$

2. Une estimation des paramètres de la copule en utilisant les marginales préalablement estimées :

$$(\hat{\rho}, \hat{\nu}) = \arg \max_{\rho, \nu} \sum_{i=1}^n \log c(\hat{F}_1(X_{i1}), \hat{F}_2(X_{i2}); \rho, \nu)$$

où \hat{F}_1 et \hat{F}_2 sont les fonctions de répartition estimées.

Lorsque les marginales et la copule sont correctement spécifiées, les estimations obtenues par l'IFM convergent vers celles du Full MLE, tout en étant moins exigeantes en termes de calcul.

Dans notre cas, où les distributions marginales des deux variables (actifs financiers) sont inconnues, ces approches ne sont pas optimales. C'est pourquoi nous avons privilégié la **méthode d'estimation par maximum de vraisemblance canonique** (*CMLE*, *Canonical Maximum Likelihood Estimation*). La CMLE est particulièrement adaptée lorsque les marginales ne peuvent pas être spécifiées de manière paramétrique. Elle repose sur une transformation standard des données en variables uniformes (de la même manière qu'avec la copule empirique) ce qui permet de se concentrer exclusivement sur la modélisation des paramètres de la copule.

La méthode CMLE se décompose donc en deux étapes principales :

- **Transformation des données en uniformes** : Les observations brutes X_{i1} et X_{i2} sont converties en variables uniformes \hat{U}_{i1} et \hat{U}_{i2} sur l'intervalle $[0, 1]$ à l'aide de leurs rangs empiriques :

$$\hat{U}_{i1} = \frac{\text{rang}(X_{i1})}{n+1}, \quad \hat{U}_{i2} = \frac{\text{rang}(X_{i2})}{n+1}$$

- **Maximisation de la vraisemblance de la copule** : Les paramètres ρ et ν de la copule t sont estimés en maximisant la log-vraisemblance :

$$(\hat{\rho}, \hat{\nu}) = \arg \max_{\rho, \nu} \sum_{i=1}^n \log c(\hat{U}_{i1}, \hat{U}_{i2}; \rho, \nu)$$

où c est la densité de la copule de Student bivariable.

La méthode CMLE ressemble à la méthode IFM, mais une distinction clé réside dans leur traitement des marginales : l'IFM repose généralement sur une estimation paramétrique des marginales, tandis que la CMLE privilégie une approche non paramétrique via une transformation empirique.

2.3.3 Approche par deep learning

Les modèles à réseaux de neurones ont un champ d'application extrêmement large et peuvent notamment servir à estimer les paramètres d'une copule. Il ne s'agit donc pas d'une "méthode" statistique clairement identifiée mais plutôt d'une approche via du machine learning.

L'idée est de réaliser un apprentissage supervisé en entraînant un modèle sur d échantillons observés (X_{1i}, X_{2i}) , où chaque échantillon $i \in \{0, \dots, d\}$ est labelisé par une paire (ρ_i, ν_i) , estimée à l'aide d'une méthode statistique classique. L'intérêt principal de cette approche réside dans son application à une copule dynamique, où les paramètres doivent être estimés à chaque pas de temps pour chaque trajectoire lors d'une simulation Monte Carlo. En effet, recourir à une méthode traditionnelle (comme le MLE, QMLE ou IFM) engendrerait un coût computationnel démentiel, d'autant que même une approche par deep learning reste exigeante en termes de calcul. De plus, l'utilisation du deep learning présente l'avantage de capturer des dynamiques temporelles complexes et non linéaires ce qui améliore la précision des simulations Monte Carlo pour le pricing d'options sur panier.

Plusieurs facteurs doivent être considérés lors de l'estimation des paramètres d'une copule avec un modèle Transformeur :

- **Les features** : Les variables d'entrée doivent capturer les dynamiques des actifs tout en étant normalisées pour assurer la stabilité de l'entraînement.
- **Les hyperparamètres** : La longueur des séquences, le nombre de couches, les têtes d'attention, le taux d'apprentissage et les paramètres d'entraînement (nombre d'époques, taille des batches) doivent être correctement ajustés pour éviter l'overfitting et permettre au modèle de généraliser.
- **La fenêtre d'estimation initiale pour CMLE** : La taille de la fenêtre utilisée pour estimer les cibles ρ_t et ν_t et via CMLE doit équilibrer la stabilité des estimations et la réactivité aux variations temporelles des paramètres.
- **La robustesse de l'estimation initiale** : Les contraintes sur les paramètres et la gestion des erreurs lors de l'estimation via CMLE doivent assurer des cibles d'entraînement fiables pour le Transformateur.

2.4 Simulations de Monte Carlo avec copule dynamique

2.4.1 Fondamentaux

Comme mentionné dans l'introduction, la simulation Monte Carlo est une méthode adaptée pour le *pricing* d'options sur panier, car elle permet de modéliser les dépendances complexes entre actifs tout en estimant le prix via l'espérance actualisée du *payoff* sous une

mesure risque-neutre \mathbb{Q} . Plus précisément, le prix d'une option est donné par :

$$V_0 = e^{-rT} \mathbb{E}^{\mathbb{Q}} [\text{Payoff}],$$

où r est le taux sans risque et T l'échéance. Cette méthode repose sur deux hypothèses fondamentales :

- **L'absence d'arbitrage** : il est impossible de réaliser un profit sans risque ni investissement initial, soit, pour un portefeuille V , si $V_0 = 0$ et $V_T \geq 0$, alors $\mathbb{P}(V_T > 0) = 0$. Cette condition garantit alors l'existence d'une mesure risque-neutre \mathbb{Q} .
- **La mesure risque-neutre** : sous \mathbb{Q} , les prix actualisés des actifs sont des martingales, c'est-à-dire :

$$\mathbb{E}^{\mathbb{Q}} \left[\frac{S_T}{B_T} \middle| \mathcal{F}_t \right] = \frac{S_t}{B_t}, \quad \forall t \leq T,$$

où $B_t = e^{rt}$ est le numéraire et \mathcal{F}_t est l'information disponible à la date t .

La méthode des simulations de Monte Carlo permet d'estimer cette espérance en générant un grand nombre de trajectoires pour le sous-jacent S_T , puis en calculant la moyenne empirique des *payoffs*. D'après la loi des grands nombres, cette moyenne converge vers l'espérance théorique lorsque le nombre de simulations N tend vers l'infini :

$$\frac{1}{N} \sum_{i=1}^N f(S_T^{(i)}) \rightarrow \mathbb{E}^{\mathbb{Q}} [f(S_T)]$$

De plus, le théorème de la limite centrale garantit que l'erreur d'estimation suit une loi normale d'écart-type proportionnel à $1/\sqrt{N}$.

2.4.2 Mise en place

Finalement, le dernier élément nécessaire pour réaliser une simulation de Monte Carlo est la modélisation du cours du sous-jacent. Dans notre cas, on veut modéliser n trajectoires de deux sous-jacents liés par une copule de student dynamique dont les paramètres sont ajustés à chaque pas de temps. Pour mettre en place cette méthode, il faut cependant choisir deux actifs ayant une relation de dépendance cohérente avec une copule de Student (par exemple deux indices boursiers).

Voici les étapes à suivre pour simuler un pas de temps $t \in \{1, \dots, T\}$ d'une trajectoire $i \in \{1, \dots, n\}$ du panier de sous-jacents :

1. **Estimation des paramètres de la copule** : À partir de l'historique des rendements de la trajectoire i , on réalise une prédiction avec notre modèle Transformeur (qui a été entraîné au préalable) des paramètres ρ et ν .

2. **Génération de paire de variables corrélées** : Une fois $\hat{\rho}$ et $\hat{\nu}$ estimés, on simule une paire de variables uniformes (U_1, U_2) suivant la copule t bivariée. Pour cela, on génère deux variables indépendantes $Z_1, Z_2 \sim t(\hat{\nu})$, chacune suivant une distribution Student univariée avec $\hat{\nu}$ degrés de liberté. On définit ensuite la matrice de corrélation $\Sigma = \begin{bmatrix} 1 & \hat{\rho} \\ \hat{\rho} & 1 \end{bmatrix}$ à laquelle on va appliquer une factorisation de Cholesky pour avoir $\Sigma = LL^T$, par exemple $L = \begin{bmatrix} 1 & 0 \\ \hat{\rho} & \sqrt{1 - \hat{\rho}^2} \end{bmatrix}$, pour permettre d'introduire la corrélation. On obtient alors des variables corrélées $Z' = (Z'_1, Z'_2)$ par $Z' = Z \cdot L^T$, avec $Z = (Z_1, Z_2)$. Enfin, on applique la fonction de répartition t_ν de la t-Student univariée pour transformer Z'_1 et Z'_2 en uniformes : $U_1 = t_{\hat{\nu}}(Z'_1)$, $U_2 = t_{\hat{\nu}}(Z'_2)$.
3. **Modélisation des marginales des actifs** : Les uniformes (U_1, U_2) sont transformées en rendements (R_1, R_2) via une distribution normale :

$$(R_1, R_2) = (F_1^{-1}(U_1), F_2^{-1}(U_2))$$

où F_j^{-1} est la fonction quantile inverse de la loi normale centrée réduite. La raison pour laquelle nous utilisons la loi normale et non la loi de Student avec $\hat{\nu}$ degrés de liberté est que cette dernière générerait des rendements excessifs dans notre contexte de simulation. De ce fait, l'utilisation de la loi normale, tout en conservant une dépendance modélisée via une copule t, reste théoriquement valide : la spécification de la copule capture la dépendance entre actifs, tandis que les marginales peuvent être choisies indépendamment pour garantir la plausibilité économique des simulations. Pour chaque actif j , le rendement est donné par :

$$R_{j,t} = \left(\mu_j - \frac{1}{2}\sigma_j^2 \right) dt + \sigma_j \sqrt{dt} t_{\hat{\nu}}^{-1}(U_{j,t}),$$

où μ_j est la dérive, σ_j la volatilité, et $t_{\hat{\nu}}^{-1}$ la fonction quantile de la t-Student. Le prix est alors mis à jour selon :

$$S_{j,t} = S_{j,t-1} e^{R_{j,t}}$$

4. **Simulation des trajectoires de prix** : Le prix simulé $S_{j,t}$ est intégré à l'historique des rendements de l'actif j et le processus pour chaque trajectoire du pas de temps concerné, avant de passer au pas de temps suivant.

3 Estimation du modèle Transformeur

3.1 Formatage des features

Pour que le modèle puisse prédire les paramètres de la copule à chaque période, il doit recevoir en entrée les mêmes caractéristiques (features) que celles utilisées lors de la phase d’entraînement. Bien que cette affirmation puisse paraître évidente, elle impose néanmoins une contrainte importante sur le choix des features : celles-ci doivent être disponibles à la fois durant la phase d’entraînement sur les données historiques, mais aussi lors de la simulation sur des périodes futures. Concrètement, cela implique de restreindre le choix des features aux seules données simulées, soit les cours des actifs. Il est par conséquent impossible d’utiliser en entrée des variables telles que le niveau du VIX, celui d’un taux sans risque ou d’un quelconque autre indicateur (il aurait sinon fallu les modéliser également, ce qui aurait grandement complexifié l’approche et introduit de nouvelles sources d’erreur).

De par cette contrainte, nous avons choisi comme features les rendements logarithmiques des sous-jacents ainsi que leurs 3 premiers moments statistiques. Plus précisément, le calcul de la volatilité est réalisé sur une fenêtre glissante de 20 jours tandis que celui du skewness et du kurtosis est réalisé sur une fenêtre glissante de 30 jours, le tout sur l’historique des cours depuis 2005. Ces tailles de fenêtres ont été choisies pour équilibrer la stabilité des estimations et la réactivité aux variations récentes des distributions. Toutes ces données sont ensuite normalisées ($\frac{x-\mu}{\sigma}$) afin de faciliter la convergence de l’optimisation et d’éviter de potentiels biais d’échelles entre les features. Finalement, nous avons retenu une longueur de séquence de 50 pour les entrées du modèle, à la suite de plusieurs comparaisons effectuées manuellement.

Les labels, (ρ_t et ν_t), seront quant à eux estimés sur une fenêtre glissante de 100 jours avec la méthode CMLE exposée précédemment. Une fenêtre de 100 jours permet de lisser suffisamment les valeurs pour permettre au modèle de prédire correctement tout en limitant le temps de calcul ($\approx 2h20$), ce qui n’est pas le cas d’une fenêtre de 50 jours.

3.2 Hyperparamètres et entraînement

Notre modèle est de type Transformeur, ce qui impose une certaine architecture.⁹. Toutefois, plusieurs paramètres liés au processus d’apprentissage doivent être définis en amont. Contrairement aux paramètres appris automatiquement lors de l’entraînement, ceux-ci ne sont pas ajustés par le modèle, ce pourquoi on les nomme ”hyperparamètres”. Plusieurs méthodes existent pour déterminer les hyperparamètres optimaux afin d’assurer une performance maximale du modèle. Par exemple, la recherche par grille (*grid search*) testant exhaustivement toutes les combinaisons possibles; la *random search* sélectionnant

⁹Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.

aléatoirement des combinaisons dans l'espace de recherche, souvent avec de meilleurs résultats pour un coût moindre; ou encore une approche bayésienne modélisant la performance attendue selon les hyperparamètres testés. Dans notre cas, nous avons opté pour une sélection manuelle car une exploration automatique aurait été trop coûteuse en temps et peu pertinente dans notre configuration.

Pour l'architecture du modèle, nous avons opté pour une configuration avec 8 têtes d'attention, une dimension d'encodage de 64 et une couche feedforward de dimension 512. Afin de condenser les informations de la séquence en une représentation unique et exploitable, une étape de pooling par moyenne globale a été intégrée. Concernant les hyperparamètres d'apprentissage, un taux de dropout de 0,2 a été choisi pour limiter le surapprentissage, avec un entraînement sur 25 époques et une taille de batch fixée à 32. La normalisation des couches utilise une valeur epsilon de 10^{-5} pour assurer la stabilité, et l'optimiseur *Adam* a été retenu pour garantir une convergence efficace.

Figure 4: *Évolution de la perte pendant l'entraînement*

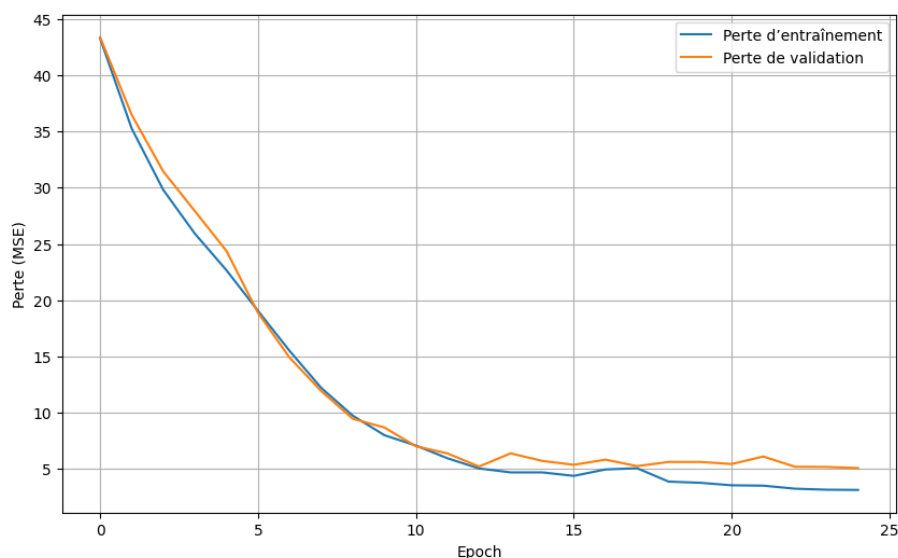


Table 2: Tableau des valeurs de $\nu\epsilon\tau\rho$

nu_true	nu_pred	rho_true	rho_pred
2.01	2.45	0.74	0.77
2.22	3.20	0.54	0.74
4.71	3.40	0.59	0.50
3.38	3.16	0.72	0.77
13.41	21.87	0.53	0.56
...
5.05	4.11	0.72	0.70
2.45	1.61	0.53	0.53
2.01	2.81	0.64	0.75
4.10	4.88	0.63	0.73
5.33	4.40	0.37	0.49

Perte moyenne au carré (MSE) :

- ρ_t : 0.0093

- ν_t : 9.65

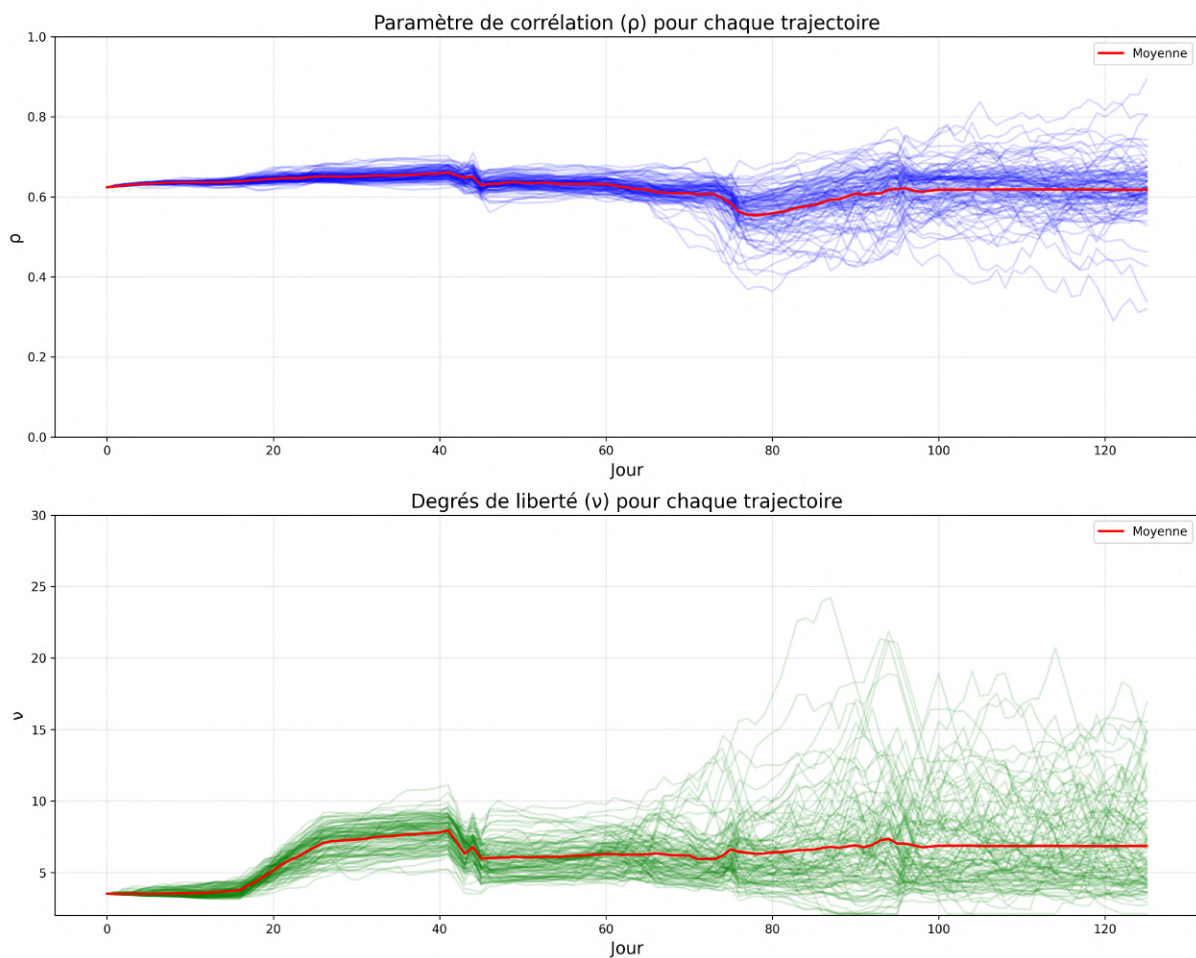
4 Résultats des simulations

4.1 Modèle à copule dynamique

Récapitulatif : Notre objectif est de pricer une option basket équipondérée sur le SP500 et le CAC40 à maturité dans 6 mois (126 jours) avec une approche par monte carlo intégrant une copule de student dynamique propre à chaque trajectoire. Les données historiques des indices sont issues des *ETFs* *SPY* et *EWQ* et la durée totale d'exécution du code a été d'environ 24 heures (depuis un macbook air M1) pour 10 000 simulations. Le taux sans risque a été établi à 3% et les paramètres de volatilité ont été établis à 20% pour le CAC40 et 30% pour le SP500.

Figure 5: *Évolution des paramètres*

Trajectoires individuelles des paramètres de la copule t-Student

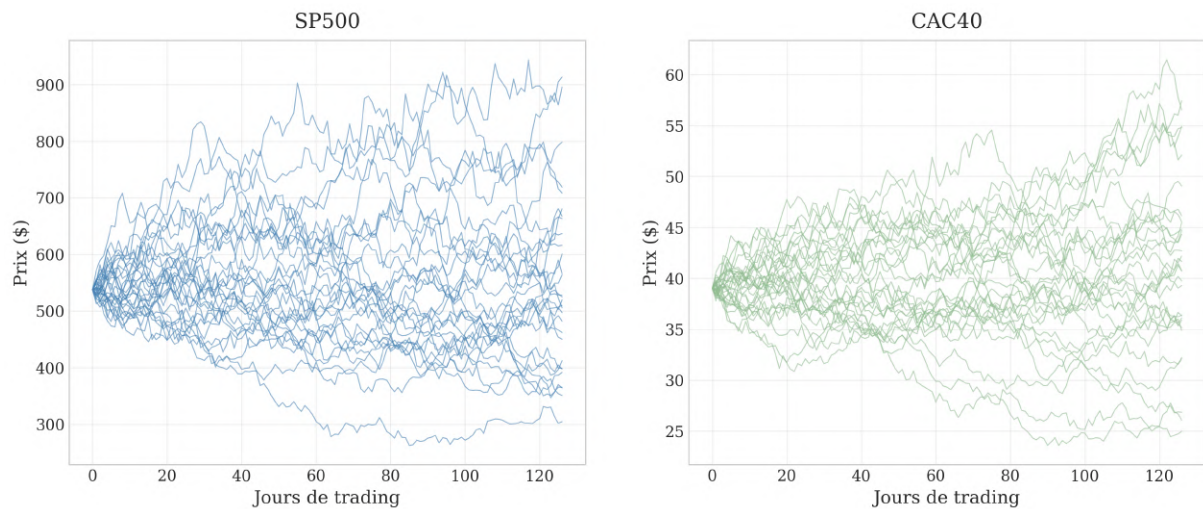


Tout d'abord, on observe que l'évolution des paramètres dynamiques ρ_t et ν_t tend à

devenir de plus en plus volatile au cours du temps. Cela s'explique par la diminution progressive de l'historique commun entre les trajectoires simulées : à chaque itération, les nouvelles valeurs générées viennent perturber l'estimation des paramètres, qui devient alors plus sensible aux spécificités de chaque trajectoire.

En moyenne, le paramètre de corrélation ρ_t s'établit autour de 0.6, traduisant une dépendance modérée entre les actifs sous-jacents. Le paramètre ν_t , quant à lui, converge en moyenne vers 8, ce qui correspond à des queues relativement épaisses dans la copule de Student, renforçant la prise en compte d'événements extrêmes dans les simulations.

Figure 6: *Trajectoires des prix*

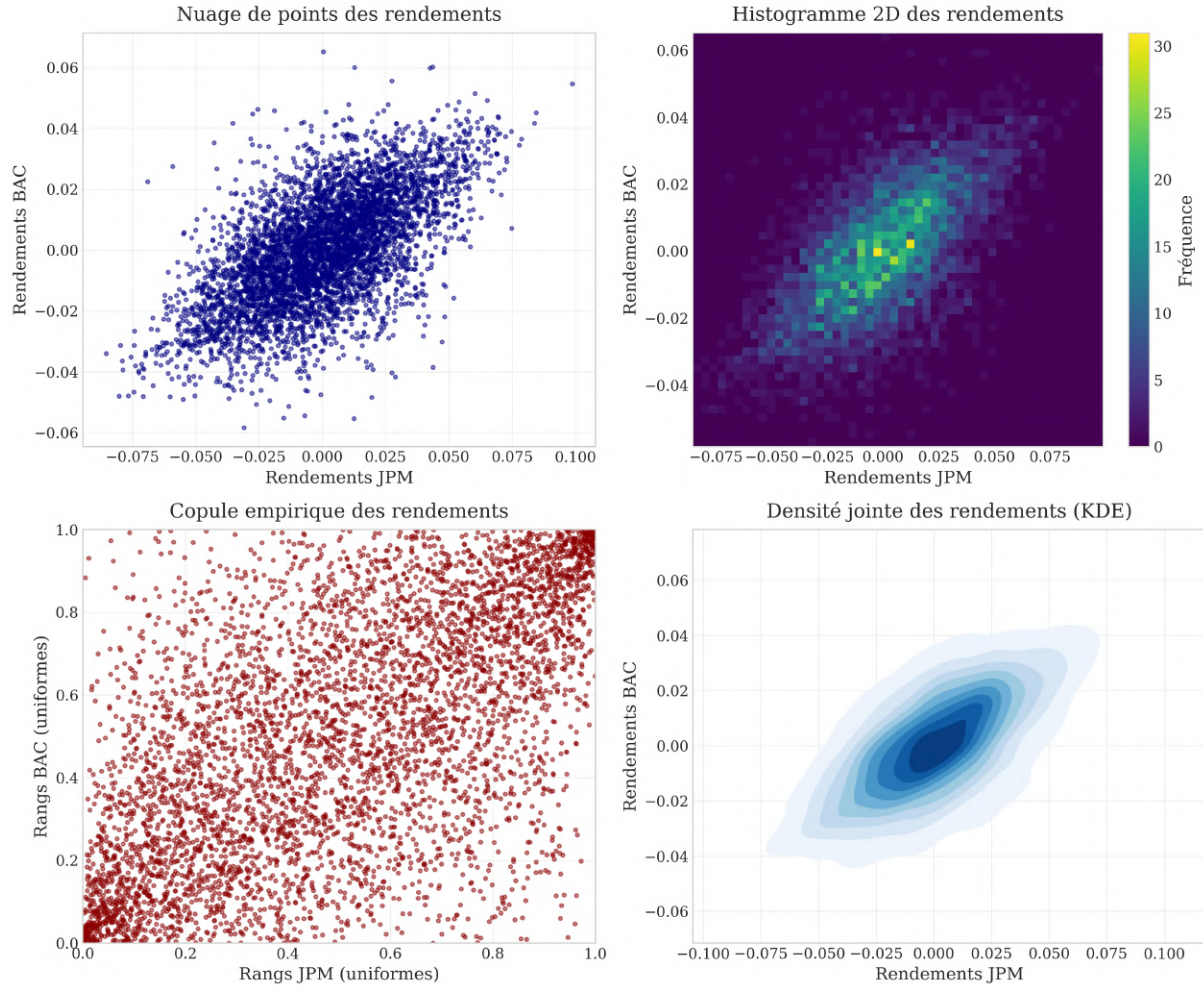


On constate via la représentation graphique des trajectoires simulées que les trajectoires du *SP500* tendent à être plus volatiles que celles du *CAC40*, ce qui est cohérent avec notre paramétrage de volatilité initial. La simulation nous donne alors un prix final moyen de 552.41 pour le *SP500* et de 40.25 pour le *CAC40* (on les considère équivalents aux ETFs *SPY* et *EWQ*). Cela représente donc une augmentation de 2.75% pour le *SP500* et de 3.2% pour le *CAC40*.

Indice	Prix initial	Prix final moyen	Variation (%)
SP500	\$537.61	\$552.41	+2.75%
CAC40	\$39.00	\$40.25	+3.21%

Finalement, on trouve un prix de 36\$ pour un call à maturité 126 jours (6 mois) et ayant un strike à 288.31\$ $((537.61 + 39)/2)$. Ce prix élevé s'explique par le fait que le prix final moyen du panier vaut 296.33\$ et que la valeur moyenne des payoffs positifs vaut 77.48\$. De plus, on trouve l'intervalle de confiance du prix au seuil d'erreur de 5% : $[35.01, 37.35]$.

Figure 7: *Analyse de la dépendance des rendements*



La figure 7 propose une exploration visuelle de la structure de dépendance entre les rendements du SP 500 et du CAC 40, à partir des données simulées sous la copule de Student. Ces représentations permettent d'évaluer la cohérence entre le comportement généré par le modèle et les propriétés théoriques attendues de la copule utilisée.

Le nuage de points (en haut à gauche) met en évidence une relation globalement croissante entre les deux séries, traduisant une corrélation positive. Cette impression est confirmée par l'histogramme 2D (en haut à droite), où l'on observe une concentration nette de la densité le long de la diagonale ascendante. Ces éléments valident la capacité de la copule à restituer la dépendance centrale des rendements.

Plus intéressant encore, la copule empirique (en bas à gauche), fondée sur les rangs uniformisés des rendements, révèle une forte concentration de points dans les coins supérieur droit et inférieur gauche de l'espace unité. Ce schéma est typique d'une dépendance dans les queues, et donc conforme à la spécificité structurelle de la copule de Student, qui est précisément conçue pour modéliser les co-mouvements extrêmes.¹⁰

¹⁰McNeil, A. J., Frey, R., & Embrechts, P. (2015). *Quantitative Risk Management: Concepts, Tech-*

La densité jointe estimée par noyau (en bas à droite) confirme ce diagnostic. La forme elliptique étirée, combinée à une densité significative en périphérie, illustre une dépendance à la fois forte au centre et substantielle dans les extrêmes.

Ainsi, l'ensemble des visualisations confirme que le modèle simulé capture fidèlement la structure de dépendance attendue d'une copule de Student, tant au niveau global qu'au niveau des queues de distribution. Ce constat renforce la pertinence du choix de cette copule dans le cadre de la valorisation de produits dérivés multi-actifs, et notamment pour la prise en compte des risques extrêmes dans le pricing d'options baskets.

4.2 Comparaison avec d'autres modèles

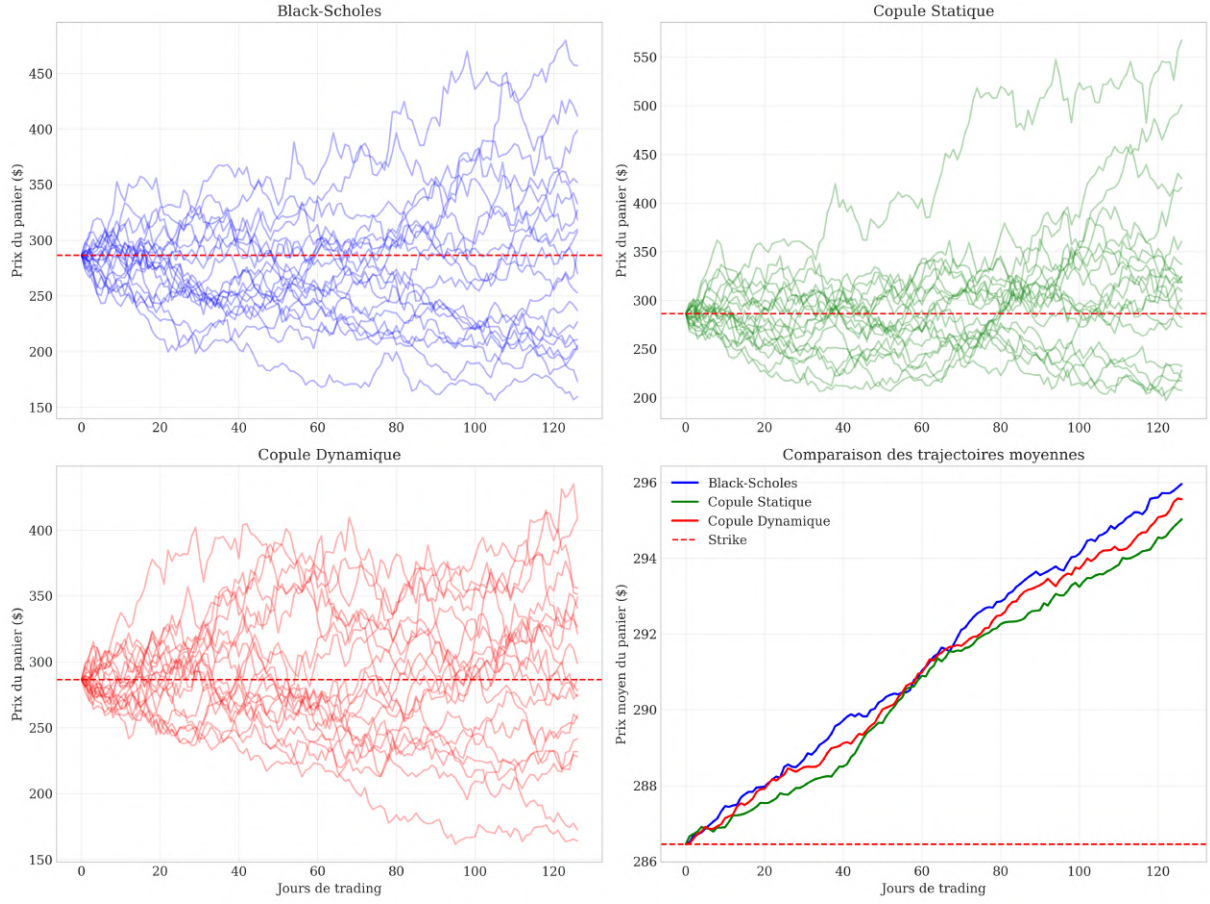
Finalement, comparons les résultats de notre méthode de simulation de Monte Carlo par copule dynamique avec ceux de méthodes plus classiques comme le modèle à copule statique et le modèle de Black Scholes.

Le modèle de Black-Scholes suppose une dynamique log-normale pour les prix des actifs S_t , avec une dérive ajustée au taux sans risque $r = 0,03$, soit :

$$dS_t = rS_t dt + \sigma S_t dW_t,$$

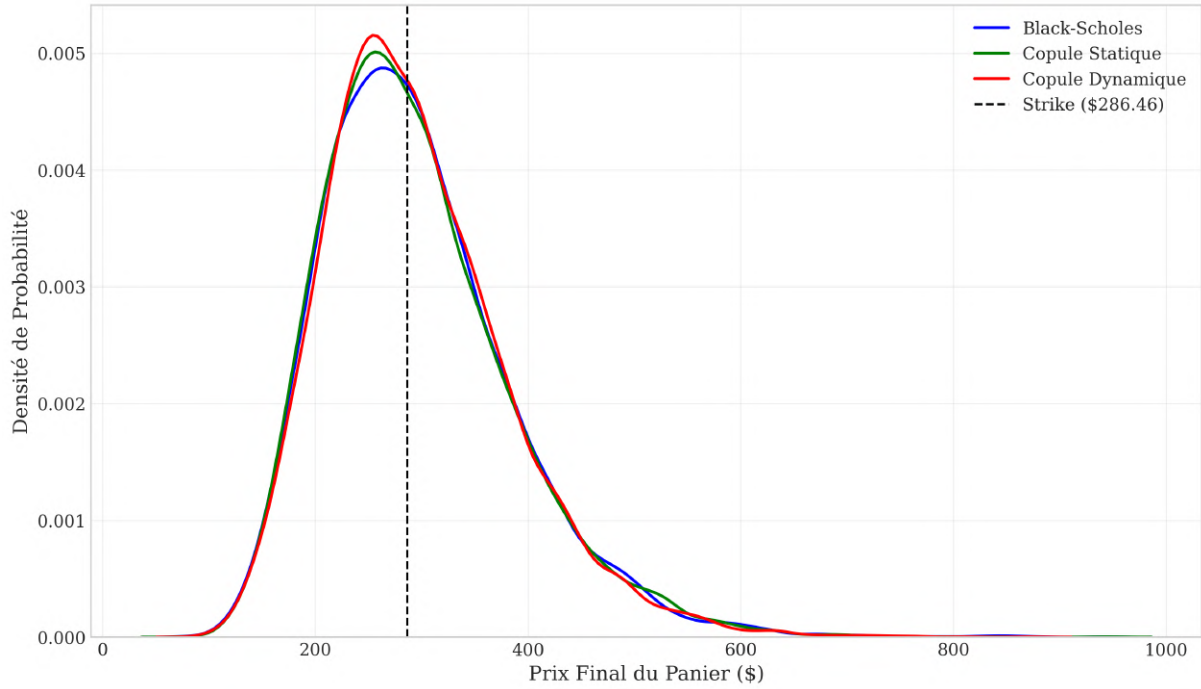
où σ est la volatilité et W_t un mouvement brownien corrélé avec la corrélation empirique ρ (≈ 0.62). Les trajectoires sont toujours simulées sur $N = 10\,000$ simulations et $T = 126$ jours de trading.) La copule statique, quant à elle, utilise une copule de Student avec des paramètres ρ et ν estimés par maximum de vraisemblance sur l'ensemble des données historiques, transformées en uniformes via les fonctions de répartition empiriques : $u_i = \text{rank}(r_i)/(n + 1)$. Ces paramètres restent fixes sur toute la période de simulation, et les rendements sont générés en transformant des uniformes corrélées en rendements normaux via $\Phi^{-1}(u_i)$, avant d'être appliqués à la dynamique des prix, soit la même méthode que celle utilisée avec la copule dynamique.

Figure 8: *Comparaison des trajectoires des modèles*



Les deux sous-graphiques du haut et celui d'en bas à gauche montrent respectivement les trajectoires de Black-Scholes, de la copule statique, et de la copule dynamique, tandis que le sous-graphique en bas à droite compare leurs trajectoires moyennes. On observe une dispersion plus marquée pour les copules statique et dynamique, avec des prix variant entre 200 \$ et 500 \$, contre 200 \$ à 450 \$ pour Black-Scholes, ce qui reflète la capacité des copules de Student à capturer les queues épaisses des distributions de rendements. La copule dynamique présente des trajectoires plus irrégulières, dues à l'ajustement temporel des paramètres ρ_t et ν_t , contrairement à la copule statique où ces paramètres sont fixes. Les trajectoires moyennes, proches du prix d'exercice (300 \$), évoluent de 288 \$ à environ 294 \$ pour Black-Scholes et 296 \$ pour les copules, confirmant une calibration cohérente sous la mesure risque-neutre, où $\mathbb{E}^{\mathbb{Q}}[S_T] \approx S_0 e^{rT}$. Ces résultats soulignent l'impact des dépendances dynamiques sur la variabilité des prix simulés, un aspect clé pour le pricing d'options sur panier.

Figure 9: *Distribution des prix finaux*



La figure 9 présente les distributions des prix finaux du panier simulées selon trois approches : le modèle Black-Scholes multivarié, la copule statique et la copule dynamique. Les densités sont relativement proches, traduisant une cohérence globale entre les méthodes. Toutefois, la copule statique présente une concentration légèrement plus marquée autour du mode, suggérant une sous-estimation de la probabilité des extrêmes par rapport aux deux autres modèles. Le seuil d'exercice (286,46\$), indiqué par la ligne verticale en pointillés, se situe dans la partie décroissante de la distribution, ce qui implique une probabilité d'exercice non négligeable mais inférieure à celle du pic de densité. Ces différences, bien que subtiles, peuvent entraîner des écarts significatifs dans la valorisation des options, en particulier pour les produits exotiques sensibles aux queues de distribution.

Table 3: Comparaison des prix d'options baskets et des mesures de risque.

Modèle	Prix Call (\$)	Prix Put (\$)	VaR 95%	ES 95%
Black-Scholes	37.88	28.66	38.84 %	45.68 %
Copule statique	37.34	29.02	38.87 %	45.32 %
Copule dynamique	36.82	28.00	38.52 %	45.34 %

On constate que le prix du call diminue légèrement lorsque l'on passe de Black-Scholes à la copule dynamique, tandis que le prix du put connaît une variation moins systématique. Les mesures de risque $\text{VaR}_{95\%}$ et $\text{ES}_{95\%}$ restent stables autour de 39 % et 45 % respectivement, ce qui traduit une sensibilité comparable aux queues de distribution dans les trois cadres considérés.

La parité call-put, dans un univers sans dividendes, s'écrit :

$$C - P = S_0 - K e^{-rT}.$$

Pour chaque modèle, nous calculons la différence empirique $C - P$, la comparaison avec $S_0 - K e^{-rT}$ et l'écart relatif. Les résultats sont présentés dans le tableau 4.

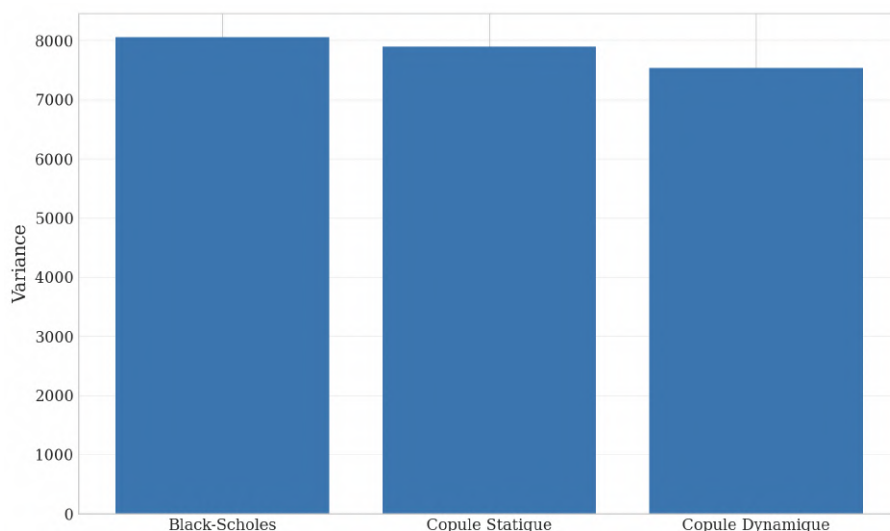
Modèle	$C - P$	$S_0 - K e^{-rT}$	Écart (\$)	Écart (%)
Black-Scholes	9.22	8.47	+0.76	+8.92 %
Copule statique	8.32	8.47	-0.15	-1.76 %
Copule dynamique	8.83	8.47	+0.36	+4.28 %

Table 4: Vérification numérique de la parité call-put pour chaque modèle.

La présence de ces écarts de quelques pourcents est dû au faible nombre de simulations réalisées (10000). En effet, en augmentant le nombre de simulations réalisées à plusieurs millions via une forte réduction de la complexité des paramètres, les écarts tendent bien vers 0.

Afin de compléter l'analyse comparative des modèles, nous examinons à présent la dispersion des prix simulés à maturité. La variance de ces prix constitue un indicateur pertinent de la stabilité des trajectoires générées, notamment en contexte de pricing d'options. La figure suivante illustre la variance des prix finaux du panier pour chacun des trois modèles étudiés :

Figure 10: *Variance des prix finaux*



Comme le montre la figure 10, la variance la plus élevée est obtenue sous le modèle de Black-Scholes, suivi de près par la copule statique. La copule dynamique, quant à elle, présente une variance plus contenue, traduisant une meilleure stabilité des prix simulés. Cette observation est cohérente avec la capacité des copules dynamiques à modéliser des

structures de dépendance évolutives, ce qui tend à limiter les co-mouvements extrêmes entre les actifs.

Le tableau suivant synthétise les statistiques de dispersion associées à chaque modèle pour les prix finaux du panier :

Modèle	Variance	Écart-type	Coefficient de variation (CV)
Black–Scholes	8058.63	89.77	0.3033
Copule statique	7902.62	88.90	0.3013
Copule dynamique	7542.90	86.85	0.2939

Table 5: Statistiques de dispersion des prix finaux simulés pour le panier.

Il ressort de cette comparaison que l'utilisation d'une structure de dépendance dynamique permet de réduire légèrement l'incertitude sur la valeur future du panier. Cette caractéristique est particulièrement souhaitable dans des applications de gestion du risque ou de valorisation sensible à la queue de distribution, telles que le calcul de la Value-at-Risk (VaR) ou de l'Expected Shortfall (ES), étudiés précédemment. Cela vient renforcer la pertinence des modèles à copule dynamique dans un cadre réaliste, où la dépendance entre actifs n'est ni constante ni symétrique au cours du temps.

Ce constat est d'autant plus soutenu par l'observation d'un *coefficient de variation* (CV) plus faible pour ce modèle. Le coefficient de variation, noté CV , est défini comme le rapport entre l'écart-type σ et la moyenne μ d'un échantillon :

$$CV = \frac{\sigma}{\mu}$$

Cette mesure permet d'évaluer la dispersion relative d'une variable aléatoire, indépendamment de son ordre de grandeur. Dans le contexte présent, un CV plus faible indique une meilleure stabilité relative du prix du panier simulé. Or, le modèle à copule dynamique présente le plus faible coefficient de variation ($CV = 0.2939$) parmi les trois approches testées, contre 0.3013 pour la copule statique et 0.3033 pour Black-Scholes.

5 Discussion du modèle à copule dynamique

5.1 Avantages

Capture des dépendances évolutives : contrairement aux modèles à copule statique ou à l’approche Black-Scholes, la copule dynamique permet de modéliser l’évolution temporelle des dépendances entre les actifs sous-jacents. Comme illustré dans la figure 5, les paramètres ρ_t et ν_t s’ajustent dynamiquement à chaque période, reflétant les variations des corrélations et des queues de distribution. Cette flexibilité est particulièrement pertinente dans des contextes de marchés financiers où les relations entre actifs, telles que celles entre le S&P 500 et le CAC 40, fluctuent en réponse à des événements économiques ou des chocs exogènes. Par exemple, la corrélation moyenne estimée de $\rho_t \approx 0.6$ dans nos simulations traduit une dépendance modérée, mais sa variabilité capture des épisodes de co-mouvements plus intenses.

Modélisation des événements extrêmes : la copule de Student est reconnue pour sa capacité à modéliser les dépendances dans les queues de distribution, ce qui est crucial pour le pricing d’options sensibles aux risques extrêmes. Nos résultats, notamment la figure 7, confirment une forte concentration de densité dans les extrêmes, cohérente avec les propriétés théoriques de la copule. La dynamique des paramètres, avec ν_t convergeant vers 8 en moyenne, renforce cette aptitude en ajustant l’épaisseur des queues au fil du temps. Comparée à Black-Scholes, qui suppose une distribution log-normale sans dépendance dans les queues, la copule dynamique offre une représentation plus réaliste des co-mouvements extrêmes, un facteur clé pour les options baskets.

Réduction de la dispersion des prix simulés : comme montré dans le tableau 5, le modèle à copule dynamique présente une variance et un coefficient de variation inférieurs à ceux des modèles Black-Scholes et à copule statique. Cette stabilité accrue des prix finaux simulés (variance de 7542.90 contre 8058.63 pour Black-Scholes) traduit une meilleure maîtrise des incertitudes, particulièrement dans les scénarios de marché volatils. Cette propriété est avantageuse pour des applications de gestion des risques, telles que le calcul de la Value-at-Risk (VaR) ou de l’Expected Shortfall (ES), où une dispersion excessive peut fausser les estimations.

Flexibilité de l’architecture Transformeur : l’utilisation d’un modèle Transformeur pour prédire les paramètres de la copule confère une grande adaptabilité. La capacité des mécanismes d’attention à capturer des relations complexes entre les rendements logarithmiques et leurs moments statistiques (volatilité, skewness, kurtosis) permet une estimation robuste des ρ_t et ν_t . La configuration retenue (8 têtes d’attention, dimension d’encodage de 64) équilibre performance et coût computationnel, comme en témoigne la perte moyenne au carré (MSE) de 0.0093 pour ρ_t et 9.65 pour ν_t . Cette approche surpasse

les méthodes traditionnelles d'estimation par fenêtres glissantes, qui peinent à intégrer des dépendances non linéaires ou des historiques longs.

5.2 Limites

Malgré ses nombreux avantages, le modèle à copule dynamique présente des limites qui méritent d'être discutées pour en apprécier pleinement la portée.

Coût computationnel élevé : l'entraînement du modèle Transformeur et la génération des simulations Monte Carlo sont particulièrement gourmands en ressources. Avec une durée d'exécution d'environ 24 heures pour 10 000 simulations sur un MacBook Air M1, cette approche peut poser des contraintes dans des contextes nécessitant des calculs en temps réel, comme le trading haute fréquence ou la gestion dynamique de portefeuilles. De plus, l'estimation des paramètres ρ_t et ν_t sur une fenêtre glissante de 100 jours, bien que nécessaire pour lisser les estimations, accroît le temps de calcul (environ 2h20 pour l'entraînement). Ce coût limite la scalabilité du modèle pour des portefeuilles comprenant un grand nombre d'actifs.

Sensibilité aux données simulées : la dépendance exclusive aux rendements logarithmiques et à leurs moments statistiques, imposée par l'absence de données exogènes (par exemple, le VIX ou les taux sans risque), constitue une contrainte majeure. Comme mentionné dans la section 3, cette restriction garantit la compatibilité entre les phases d'entraînement et de simulation, mais elle peut réduire la richesse des informations exploitables. Par exemple, l'omission d'indicateurs macroéconomiques ou de sentiment de marché pourrait limiter la capacité du modèle à anticiper des changements structurels dans les dépendances entre actifs.

Biais d'estimation des paramètres : les écarts observés entre les valeurs prédites et réelles de ν_t (MSE de 9.65) indiquent une certaine imprécision dans l'estimation des degrés de liberté, particulièrement pour des valeurs élevées (voir tableau 2). Cette variabilité peut affecter la modélisation des queues de distribution, un aspect critique pour les options sensibles aux extrêmes. De plus, la volatilité croissante des paramètres ρ_t et ν_t au fil des simulations, comme illustré dans la figure 5, reflète une sensibilité accrue aux trajectoires individuelles, ce qui peut introduire du bruit dans les estimations à long terme.

5.3 Perspectives

Optimisation computationnelle : une première perspective consisterait à optimiser le coût computationnel du modèle. Des techniques comme la parallélisation des simulations Monte Carlo sur des architectures GPU ou l'utilisation d'algorithmes d'inférence accélérée pour le Transformeur pourraient réduire significativement le temps d'exécution. Par ailleurs, l'exploration de méthodes d'estimation plus rapides pour ρ_t et ν_t , telles que des approximations bayésiennes ou des réseaux neuronaux récurrents (RNN), pourrait alléger la dépendance aux fenêtres glissantes coûteuses.

Enrichissement des features : pour surmonter la contrainte des données simulées, une approche hybride pourrait être envisagée, combinant des modèles de simulation pour des variables exogènes (par exemple, le VIX ou les taux d'intérêt) avec la copule dynamique. Cela nécessiterait de développer des modèles auxiliaires pour ces variables, mais pourrait améliorer la robustesse des prédictions en capturant des dynamiques macroéconomiques ou de marché. Une analyse de sensibilité aux features exogènes pourrait également guider leur sélection.

Amélioration de l'estimation des paramètres : pour réduire les biais dans l'estimation de ν_t , des techniques de régularisation avancées, comme des pénalités adaptatives ou des priors informatifs dans un cadre bayésien, pourraient être intégrées à l'entraînement du Transformeur. De plus, l'utilisation de modèles à copules asymétriques, comme la copule de Clayton ou de Gumbel, pourrait mieux capturer des dépendances non symétriques dans les queues, un aspect que la copule de Student ne modélise qu'imparfaitement.

Validation sur des données réelles : enfin, une validation plus poussée du modèle sur des données de marché réelles, couvrant des périodes de crises financières (par exemple, 2008 ou 2020), permettrait d'évaluer sa robustesse face à des chocs extrêmes. Une telle analyse pourrait inclure des tests de backtesting pour comparer les prix simulés aux prix observés, ainsi qu'une évaluation des performances du modèle dans des stratégies de couverture dynamique.

6 Conclusion

Ce travail a exploré une approche novatrice pour la valorisation d'options sur panier, en s'appuyant sur une copule de Student dynamique dont les paramètres ρ_t et ν_t sont estimés à l'aide d'un modèle Transformeur. En combinant des simulations Monte Carlo avec une modélisation des dépendances évolutives, cette méthode vise à capturer les dynamiques complexes des co-mouvements entre actifs financiers, ici représentés par les indices S&P 500 et CAC 40. Les résultats obtenus, tant sur le plan du pricing que de l'analyse des dépendances, confirment la pertinence de cette approche dans un cadre réaliste de finance quantitative.

L'utilisation de la copule de Student s'est révélée particulièrement adaptée pour modéliser les dépendances dans les queues de distribution, un aspect crucial pour les options sensibles aux événements extrêmes. Comme illustré dans la figure 7, la forte concentration de densité dans les extrêmes traduit une prise en compte efficace des co-mouvements extrêmes, contrairement à la copule de Gauss, critiquée pour sa sous-estimation des risques lors de la crise de 2008. L'intégration d'un modèle Transformeur a permis d'estimer dynamiquement les paramètres de la copule, capturant ainsi les variations temporelles des corrélations et des épaisseurs de queues, avec une corrélation moyenne de $\rho_t \approx 0.6$ et des degrés de liberté convergeant vers $\nu_t \approx 8$. Ces résultats, appuyés par une perte moyenne au carré (MSE) de 0.0093 pour ρ_t , témoignent de la robustesse de l'approche, bien que l'estimation de ν_t (MSE de 9.65) reste perfectible.

La comparaison avec des modèles de référence, tels que Black-Scholes et la copule statique, a mis en évidence plusieurs avantages de la copule dynamique. Comme montré dans le tableau 5, elle réduit la dispersion des prix simulés (variance de 7542.90 contre 8058.63 pour Black-Scholes), offrant une meilleure stabilité pour les applications de gestion des risques, telles que le calcul de la Value-at-Risk (VaR) et de l'Expected Shortfall (ES). Le prix obtenu pour un call sur panier (36\$ pour un strike de 288.31\$) s'inscrit dans un intervalle de confiance étroit ([35.01, 37.35]), confirmant la fiabilité des simulations malgré un nombre limité de 10 000 trajectoires. Toutefois, des écarts dans la parité call-put (4.28% pour la copule dynamique, tableau 4) soulignent la nécessité d'augmenter le nombre de simulations pour réduire les erreurs de Monte Carlo.

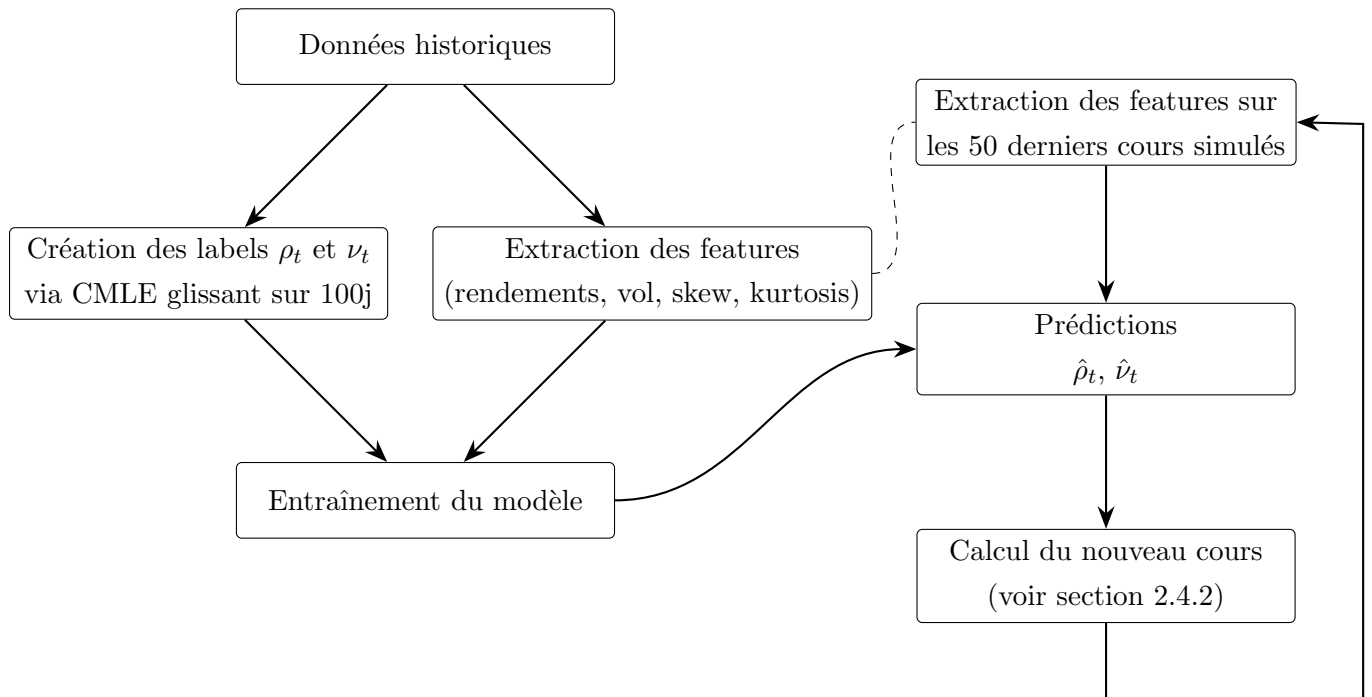
Malgré ces atouts, le modèle présente des limites, notamment un coût computationnel élevé (24 heures pour 10 000 simulations) et une dépendance aux données simulées, qui exclut l'utilisation de variables exogènes comme le VIX. Ces contraintes, discutées dans la section **Discussion**, ouvrent des perspectives prometteuses. L'optimisation computationnelle via des architectures GPU, l'enrichissement des features par des modèles de variables exogènes, ou l'exploration de copules asymétriques comme celles de Clayton ou Gumbel pourraient renforcer la précision et la flexibilité de l'approche. De plus, une

validation sur des données réelles, notamment lors de périodes de crises financières, permettrait d'évaluer la robustesse du modèle face à des chocs extrêmes.

En définitive, cette étude a permis de mettre en avant le potentiel des copules dynamiques couplées à des techniques d'apprentissage profond pour la valorisation de produits dérivés multi-actifs. Les résultats obtenus pourraient motiver une étude plus approfondie, notamment en explorant d'autres types de copules, d'architectures de réseaux ou d'applications à des produits encore plus complexes.

Appendice

Figure 11: Schéma structurant la démarche



Dans cette section, nous présentons uniquement les extraits de code essentiels à la compréhension de notre démarche. Le code complet est très long mais peut vous être envoyé sur demande.

Préparation des données d'entrée du modèle

```
data["r_spy"] = np.log(data["^GSPC"] / data["^GSPC"].shift(1))
data["r_fchi"] = np.log(data["^FCHI"] / data["^FCHI"].shift(1))
data["vol_spy"] = data["r_spy"].rolling(window=20).std()
data["vol_fchi"] = data["r_fchi"].rolling(window=20).std()
from scipy.stats import skew, kurtosis
data["skew_spy"] = data["r_spy"].rolling(window=30).apply(lambda
    x: skew(x))
data["kurt_spy"] = data["r_spy"].rolling(window=30).apply(lambda
    x: kurtosis(x))
data["skew_fchi"] = data["r_fchi"].rolling(window=30).apply(
    lambda x: skew(x))
data["kurt_fchi"] = data["r_fchi"].rolling(window=30).apply(
    lambda x: kurtosis(x))

# Transformation en uniformes
data["u_spy"] = data["r_spy"].rank() / (len(data["r_spy"]) + 1)
data["u_fchi"] = data["r_fchi"].rank() / (len(data["r_spy"]) + 1)
```



```

# Calcul de rho_t et nu_t sur des fenetres roulantes
window_size = 100
rho_targets = []
nu_targets = []
print(f"Calcul des parametres rho_t et nu_t sur fenetres de {
    window_size} jours...")
for i in tqdm(range(window_size, len(data))):
    window = data[["u_spy", "u_fchi"]].iloc[i - window_size:i].
        values
    rho_t, nu_t = estimate_rho_nu(window)
    rho_targets.append(rho_t)
    nu_targets.append(nu_t)

X = data[["r_spy", "r_fchi", "vol_spy", "vol_fchi", "skew_spy", "
    kurt_spy", "skew_fchi", "kurt_fchi"]].values[window_size:]

# Normalisation des donnees
scaler_X = StandardScaler()
X_scaled = scaler_X.fit_transform(X)

```

Construction du modèle Transformeur

```

def build_transformer(input_shape, num_heads=8, d_model=64, dff
    =512, dropout_rate=0.2):
    inputs = Input(shape=input_shape)
    x = inputs
    attn_output = MultiHeadAttention(num_heads=num_heads, key_dim
        =d_model)(x, x)
    attn_output = Dropout(dropout_rate)(attn_output)
    x = LayerNormalization(epsilon=1e-5)(x + attn_output)
    ffn = Dense(dff, activation='relu')(x)
    ffn = Dense(input_shape[1])(ffn)
    ffn = Dropout(dropout_rate)(ffn)
    x = LayerNormalization(epsilon=1e-6)(x + ffn)
    x = GlobalAveragePooling1D()(x)
    outputs = Dense(2)(x)
    model = Model(inputs=inputs, outputs=outputs)
    model.compile(optimizer='adam', loss='mse')
    return model

```

Fonction simulant les trajectoires de prix avec la copule dynamique

```
def simulate_prices_with_dynamic_copula(n_days, n_simulations,
    S0_snp, S0_cac,
    sigma_snp, sigma_cac, dt,
    transformer_model,
    hist_returns_snp,
    hist_returns_cac):

    prices_snp = np.zeros((n_simulations, n_days + 1))
    prices_cac = np.zeros((n_simulations, n_days + 1))

    prices_snp[:, 0] = S0_snp
    prices_cac[:, 0] = S0_cac

    # Creation d'un historique separe pour chaque trajectoire
    history_length = min(100, len(hist_returns_snp))

    # Verification que l'historique est suffisant
    initial_history_snp = hist_returns_snp[-history_length:].copy()
    ()
    initial_history_cac = hist_returns_cac[-history_length:].copy()
    ()

    # Creation des matrices d'historique en dupliquant les
    donnees initiales
    histories_snp = np.tile(initial_history_snp, (n_simulations,
        1))
    histories_cac = np.tile(initial_history_cac, (n_simulations,
        1))

    rho_history = []
    nu_history = []

    r = 0.03
    drift_snp = r
    drift_cac = r

    rho_history = np.zeros((n_simulations, n_days))
    nu_history = np.zeros((n_simulations, n_days))

    from tqdm import tqdm
```

```

for day in tqdm(range(1, n_days + 1)):

    for i in range(n_simulations):
        # Preparation des donnees d'entree pour le modele
        # avec l'historique specifique
        model_input = prepare_model_input(histories_snp[i],
            histories_cac[i])

        prediction = transformer_model.predict(model_input,
            verbose=0)
        nu_t = prediction[0, 0]
        rho_t = prediction[0, 1]

        # Stockage des parametres individuels
        rho_history[i, day-1] = rho_t
        nu_history[i, day-1] = nu_t

        # Generation d'une seule paire de variables uniformes
        # correlees
        corr_uniforms_i = generate_correlated_uniforms(1,
            rho_t, nu_t)

        returns_snp_i = norm.ppf(corr_uniforms_i[0, 0]) *
            sigma_snp * np.sqrt(dt) + (drift_snp - 0.5 *
            sigma_snp**2) * dt
        returns_cac_i = norm.ppf(corr_uniforms_i[0, 1]) *
            sigma_cac * np.sqrt(dt) + (drift_cac - 0.5 *
            sigma_cac**2) * dt

        prices_snp[i, day] = prices_snp[i, day-1] * np.exp(
            returns_snp_i)
        prices_cac[i, day] = prices_cac[i, day-1] * np.exp(
            returns_cac_i)

        # Mise a jour de l'historique
        if histories_snp.shape[1] >= 100:
            histories_snp[i, :-1] = histories_snp[i, 1:]
            histories_snp[i, -1] = returns_snp_i

            histories_cac[i, :-1] = histories_cac[i, 1:]
            histories_cac[i, -1] = returns_cac_i
        else:

```

```
        histories_snp[i] = np.append(histories_snp[i  
            ][1:], returns_snp_i)  
        histories_cac[i] = np.append(histories_cac[i  
            ][1:], returns_cac_i)  
  
    return prices_snp, prices_cac, rho_history, nu_history
```

Bibliographie

- [1] Sklar, A. (1959). Fonctions de répartition à marges données. *Université Paris*, 8(3), 229–231.
- [2] Feickert, M. (2021). *Probability Integral Transform*. Retrieved from <https://matthewfeickert.github.io/Statistics-Notes/notebooks/Introductory/probability-integral-transform.html>
- [3] Nelsen, R. B. (2006). *An Introduction to Copulas* (2nd ed.). Springer.
- [4] Li, D. X. (2000). On Default Correlation: A Copula Function Approach. *Journal of Fixed Income*, 9(4), 43–54.
- [5] Schmidt, R. (2007). *Tail dependence*. University of Cologne. Retrieved from https://wisostat.uni-koeln.de/fileadmin/sites/statistik/pdf_publicationen/TDCSchmidt.pdf
- [6] Salmon, F. (2009). Recipe for Disaster: The Formula That Killed Wall Street. *WIRED*. Retrieved from <https://www.wired.com/2009/02/wp-quant/>
- [7] Watts, S. (2016). The Gaussian Copula and the Financial Crisis: A Recipe for Disaster or Cooking the Books? *University of Oxford*. Retrieved from https://samueldwatts.com/wp-content/uploads/2016/08/Watts-Gaussian-Copula_Financial_Crisis.pdf
- [8] Joe, H. (2014). *Dependence Modeling with Copulas*. CRC Press.
- [9] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention is all you need. *Advances in Neural Information Processing Systems*, 30.
- [10] McNeil, A. J., Frey, R., & Embrechts, P. (2015). *Quantitative Risk Management: Concepts, Techniques and Tools*. Princeton University Press.