

MANSEUR Amina

ENSAE 2^{ème} année

Stage d'application
Année scolaire 2023/2024

Diffusion pour des trajectoires

Aix-Marseille Université
AMSE - UMR 7316
Marseille

Maître de stage : Badih GHATTAS
Du 17 juin 2024 au 27 septembre 2024

—
ENSAE Paris

TSA 26644

www.esnse.fr

Service des relations entreprises et des stages
5, avenue Henry Le Chatelier - 91764 PALAISEAU CEDEX - FRANCE - Tél : +33 (0)1 70 26 67 39 - Courriel : stage@ensae.fr

Remerciements

Je tiens à remercier sincèrement mon maître de stage, Monsieur Badih GHATTAS, pour m'avoir offert l'opportunité de travailler sur un sujet novateur et enrichissant alliant modélisation mathématique et intelligence artificielle. Je remercie également Monsieur Georges OPPENHEIM pour son accompagnement inestimable tout au long de ce projet. Leur encadrement à tous deux, a été d'une grande qualité, mêlant challenge et conseils judicieux, tout en me permettant d'évoluer avec une certaine autonomie.

Je souhaite également remercier toutes les personnes qui m'ont aidée sur les aspects techniques du projet, notamment Xinyu YUAN, autrice de l'article qui a inspiré mon travail, ainsi que mon amie Asmaa EL FRAIHI pour son soutien précieux.

Table des matières

1	Introduction	5
2	Analyse du problème statistique et état de l'art des modèles de diffusion	6
2.1	Modèle basé sur l'estimation du score	6
2.1.1	Processus de diffusion direct	6
2.1.2	Processus de diffusion inverse	6
2.1.3	Estimation du score	7
2.1.4	Génération de nouvelles données	7
2.2	Modèle basé sur le débruitage	7
2.2.1	Processus de diffusion direct	8
2.2.2	Processus de diffusion inverse	8
2.2.3	Estimation du modèle	8
2.2.4	Génération de nouvelles données après estimation du modèle paramétrique	10
2.3	Modèles adaptés pour les séries temporelles	10
3	Méthodologie de résolution	11
3.1	Le modèle retenu	11
3.1.1	Théorie stochastique	11
3.1.2	Algorithmique et implémentation	12
3.1.3	Paramètres de bruitage choisis en pratique	14
3.2	Les données d'entraînement	15
3.2.1	Les données du Bitcoin	16
3.2.2	Les données du S&P500	18
3.2.3	Pré-traitement des données	21
4	Expériences réalisées et résultats	22
4.0.1	Les premières métriques utilisées	22
4.1	Les méthodes et leurs limites	23
4.1.1	Méthode 1 : Comparaison simple entre signal initial et signal généré	23
4.1.2	Méthode 2 : Génération multiple et moyennage des signaux	24
4.1.3	Méthode 3 : Comparaison individuelle des signaux générés	24
4.1.4	Méthode 4 : Comparaison agrégée entre données réelles et simulées	25
4.2	Les métriques utilisées	25
4.2.1	Critères visuels	25
4.2.2	Tests statistiques	26
4.2.3	Coefficients et caractéristiques statistiques	26
4.2.4	Distances	26
4.3	Les résultats	27
4.3.1	Méthode 1 : Comparaison simple entre signal initial et signal généré	27
4.3.2	Méthode 2 : Génération multiple et moyennisation des signaux	28
4.3.3	Méthode 3 : Comparaison individuelle des signaux générés	29
4.3.4	Méthode 4 : Comparaison agrégée entre données réelles et simulées	31
5	Conclusion et discussion	34

6 Annexes	36
6.1 Formule de Tweedie	36
6.2 Problème d'optimisation	37
6.3 Preuve de l'égalité des espérances conditionnelles	37
6.4 Quelques propriétés de l'espérance conditionnelle	38
6.5 Calcul du noyau de transition pour l'ajout de bruit en une étape du modèle DDPM	38
6.6 Chaîne de Markov retournée	38
6.7 borne supérieure de la log-vraisemblance	39
6.8 Loi de $p_{t-1 t,0}(\mathbf{x}_{t-1} \mathbf{x}_t, \mathbf{x}_0)$	40
6.9 Divergence KL entre deux gaussiennes	40
6.10 Calcul de $\tilde{\mu}$	41
6.11 Mécanisme de Diffusion-TS	42
6.12 Architecture du code de Diffusion-TS	42
6.13 Optimiseur Adam	43
6.14 Précision sur les métriques utilisées pour mesurer la qualité de la génération	43
6.14.1 Le score Context-FID	43
6.14.2 Perte de corrélation croisée	43
6.14.3 ACF : fonction d'autocorrélation	44
6.14.4 Technique de réduction de la dimensionnalité	44
6.14.5 Coefficient de Hurst	45
6.14.6 Divergence KL-Fourier	45
6.14.7 Distance de Wasserstein-Fourier	45
6.14.8 Distance de Wasserstein	46
6.15 Expériences et résultats	46
6.15.1 Méthode 1	46
6.15.2 Méthode 2	52
6.15.3 Méthode 3	59
6.15.4 Méthode 4	69

1 Introduction

Le stage de deuxième année à l'ENSAE permet aux étudiants d'appliquer les concepts de mathématiques appliquées et d'économie acquis au cours de leur formation. D'une durée minimale de dix semaines, ce stage exige que l'étudiant développe une réflexion et une démarche scientifique autour d'une problématique qui peut être liée à l'analyse des phénomènes économiques ou financiers, en intégrant des techniques de traitement de l'information et de modélisation statistique.

C'est dans ce cadre que s'inscrit mon stage de quinze semaines, sur le thème de la "*Diffusion pour des trajectoires*", qui vise à générer des séries temporelles financières à l'aide de modèles génératifs de diffusion. Ce travail répond aux défis posés par la rareté et le coût élevé des données financières, en particulier celles concernant l'évolution des prix des actifs ou leurs rendements, souvent représentés comme des signaux unidimensionnels. Contrairement aux approches traditionnelles de prévision, telles que les modèles ARIMA, notre objectif est de créer de nouvelles données pour pallier ces limitations.

L'idée est donc d'entraîner un modèle à partir d'un ensemble de données réelles puis de générer de nouvelles données qui devront ressembler aux données originelles et suivre la même distribution de probabilité, sans pour autant être identiques. Ces données synthétiques viendront compléter les données réelles, augmentant ainsi le volume global disponible. Nous pourrons alors les utiliser seules ou les combiner avec les données réelles pour enrichir les ensembles d'entraînement, une méthode connue sous le nom de *data augmentation*.

Les modèles génératifs les plus courants pour les séries temporelles sont basés sur des GAN (comme TimeGAN [4]) ou des VAE (comme TimeVAE [10]). Cependant, ces approches rencontrent des problèmes de manque de diversité des données et d'instabilité de l'entraînement. Les VAE, bien qu'ils intègrent les structures temporelles comme les tendances et les saisonnalités, nécessitent une définition préalable de la distribution.

L'objectif de ce stage est d'explorer une nouvelle méthode de génération basée sur les modèles de diffusion, une approche innovante et tendance pour produire des images de qualité et de diversité prometteuses. Ces modèles offrent plusieurs avantages par rapport aux techniques classiques, en remédiant notamment aux problèmes de collapse et d'instabilité des GANs. Néanmoins, ils impliquent un temps de génération plus long à partir de la distribution apprise, et nécessite donc un compromis entre la vitesse d'échantillonnage et la qualité des résultats.

Nous viserons à générer des signaux unidimensionnels, en se basant sur la compréhension de la théorie stochastique et de l'architecture des modèles. L'évaluation de la qualité des signaux se fera selon des critères spécifiques usuels, tels que la similarité des distributions par rapport aux données d'origine, la diversité des données produites, ainsi que la simplicité et l'interprétabilité du modèle.

Par ailleurs, notons que, bien que les courbes soient plus simples à générer que les images, en raison de leur dimensionnalité inférieure, l'évaluation de leur qualité demeure complexe. Il est facile de visualiser la qualité d'une image, mais cela s'avère moins évident pour des signaux unidimensionnels.

La génération de séries temporelles financières impose donc l'introduction de nouveaux critères d'évaluation, en plus des critères habituels. Ces critères devront prendre en compte les faits stylisés, qui représentent des propriétés statistiques caractéristiques des données financières, et que nous explorerons dans le corps du rapport, en nous inspirant du travail [3].

En somme, ce projet se concentre sur l'application des modèles de diffusion pour générer des signaux unidimensionnels représentant des séries temporelles financières, tout en évaluant leur qualité selon des critères appropriés. Dans la suite de ce rapport, nous commencerons par introduire la théorie stochastique des modèles de diffusion, suivie de la présentation du modèle retenu et des algorithmes choisis. Nous détaillerons ensuite les expériences réalisées et les résultats obtenus, avant de conclure par une discussion sur les limites du stage et des résultats.

2 Analyse du problème statistique et état de l'art des modèles de diffusion

Dans le cadre de la génération de données, nous considérons un ensemble de données réelles $\{\mathbf{x}_0^i\}_{i=1}^N$ de \mathbb{R}^d , représentant des réalisations d'une distribution de probabilité inconnue p_0 . L'objectif est d'apprendre cette distribution afin d'échantillonner de nouvelles données à partir de celle-ci. Les modèles de diffusion, qui fonctionnent selon un processus itératif en temps discret¹, sont conçus pour atteindre cet objectif en utilisant deux phases principales : le bruitage et le débruitage. Dans la phase de bruitage, un signal d'origine \mathbf{x}_0 , qui est une réalisation de $\mathbf{X}_0 \sim p_0$, est progressivement dégradé par l'ajout de bruit en T étapes. À chaque étape t , on applique un noyau de transition directe $p_{t|t-1}(\mathbf{x}_t|\mathbf{x}_{t-1})$ ou une transition à une seule étape $p_{t|0}(\mathbf{x}_t|\mathbf{x}_0)$, jusqu'à obtenir un bruit final \mathbf{x}_T qui est proche d'un bruit blanc gaussien. Ce processus est souvent basé sur des hypothèses de convergence qui justifient que le signal bruité \mathbf{x}_T à l'étape T se comporte comme un bruit blanc. La phase de débruitage cherche à inverser ce processus de bruitage. En partant de \mathbf{x}_T , l'objectif est de générer un signal \mathbf{x}_0 distribué selon p_0 . Ce processus inverse, noté $p_{t-1|t}$ ou $p_{t|0}$, dépend de la méthode de bruitage utilisée, intégrant généralement une quantité inconnue comme le score ou le bruit. Pour estimer cette quantité, nous avons deux approches principales :

1. **Estimation du score** : Cette méthode consiste à estimer le score, c'est-à-dire le gradient du logarithme de la distribution p_t du signal bruité \mathbf{x}_t . L'apprentissage se fait en minimisant une fonction objectif qui évalue la différence entre le score estimé et le score réel.
2. **Estimation du bruit** : Dans cette approche, nous cherchons à estimer le bruit ajouté au signal initial \mathbf{x}_0 pour obtenir \mathbf{x}_t . Comme pour l'estimation du score, l'objectif est de minimiser une fonction qui quantifie l'écart entre le bruit estimé et le bruit réel.

Ces modèles de diffusion se révèlent être des techniques prometteuses pour générer non seulement des images de haute qualité, mais aussi des signaux temporels. Cette partie présente en détail ces modèles, en examinant les mécanismes sous-jacents ainsi que les méthodes de résolution statistiques envisageables pour surmonter les défis liés à l'apprentissage de la distribution de données.

2.1 Modèle basé sur l'estimation du score

La modélisation générative par l'estimation des gradients de la distribution des données, telle que décrite par Song et al. (2020) [9], repose sur un processus de diffusion discrétré en plusieurs étapes de bruitage, où la diffusion directe suit une distribution gaussienne, tandis que le processus inverse est caractérisé par l'estimation du score des densités de probabilité des signaux bruités.

2.1.1 Processus de diffusion direct

L'article [9] introduit le modèle génératif de diffusion basé sur le score, et considère le processus d'ajout de bruit en une étape suivant :

$$\forall t \in \{1, \dots, T\}, \mathbf{x}_t = \mathbf{x}_0 + \sigma_t z_t$$

avec z_t une réalisation de $Z_t \sim \mathcal{N}(0, \mathbf{I}_d)$ et $\sigma = (\sigma_t)_{t=1}^T$ un ensemble croissant de paramètres de contrôle du bruit. Le noyau de transition directe en une étape est dans ce cas $p_{t|0}(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0, \sigma_t^2)$.

2.1.2 Processus de diffusion inverse

Le processus inverse permet d'obtenir un signal $\hat{\mathbf{x}}_0$ à partir de n'importe quel signal bruité \mathbf{x}_t , $t \in \{1, \dots, T\}$ et se définit grâce à la formule de Tweedie [5] en faisant apparaître un terme de score :

$$\mathbb{E}[\mathbf{x}_0|\mathbf{x}_t] = \mathbf{x}_t + \sigma_t^2 \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) \quad (1)$$

1. Il existe également des modèles de diffusion selon des processus de bruitage en temps continu que nous n'allons pas évoquer ici.

où $\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t)$ est le score de Stein de la densité bruitée $p_t(\mathbf{x}_t)$. La preuve de ce résultat est donnée en annexe 6.1.

2.1.3 Estimation du score

Sachant qu'on se place dans le cas où $p_{t|0}(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\mathbf{x}_0, \sigma_t^2)$, nous pouvons réécrire le score sous la forme d'un problème d'optimisation. La preuve est donnée en annexe 6.2.

$$\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) = \operatorname{argmin}\{\mathbb{E}_{p_{0,t}} \|\mathbf{f}(\mathbf{x}_t) - \nabla_{\mathbf{x}_t} \log p_{t|0}(\mathbf{x}_t|\mathbf{x}_0)\|^2, f \in L^2(p_t)\} \quad (2)$$

où $L^2(p_t)$ est l'ensemble des fonctions de carré intégrable par rapport à la distribution p_t .

Sachant que la distribution conditionnelle $p_{t|0}$ est gaussienne, nous connaissons explicitement son score. L'équation 2 se réécrit donc comme

$$\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) = \operatorname{argmin} \left\{ \mathbb{E}_{p_{0,t}} \left\| \mathbf{f}(\mathbf{x}_t) + \frac{\mathbf{x}_t - \mathbf{x}_0}{\sigma_t^2} \right\|^2, \mathbf{f} \in L^2(p_t) \right\} \quad (3)$$

Dans [9], on estime le score par un modèle paramétrique $s_\theta(\mathbf{x}_t, t)$ que l'on apprend en minimisant la fonction objectif globale suivante :

$$\mathcal{L}(\theta, \{\sigma_t\}_{t=1}^T) = \frac{1}{T} \sum_{t=1}^T \frac{1}{2} \lambda_t \mathbb{E}_{\mathbf{x}_0 \sim p_0(\mathbf{x}_0)} \mathbb{E}_{\mathbf{x}_t \sim p_{t|0}(\mathbf{x}_t|\mathbf{x}_0)} \left[\left\| s_\theta(\mathbf{x}_t, t) + \frac{\mathbf{x}_t - \mathbf{x}_0}{\sigma_t^2} \right\|_2^2 \right] \quad (4)$$

où λ_t est un poids positif dépendant du bruit σ_t . [9] choisit $\lambda_t = \sigma_t^2$ de sorte à ce que les termes à l'intérieur de la somme de (4) ne dépendent pas de σ . $s_\theta(\mathbf{x}_t, t)$ est le modèle de paramètres θ qui permet d'estimer le score pour tout $t \in \{1, \dots, T\}$.

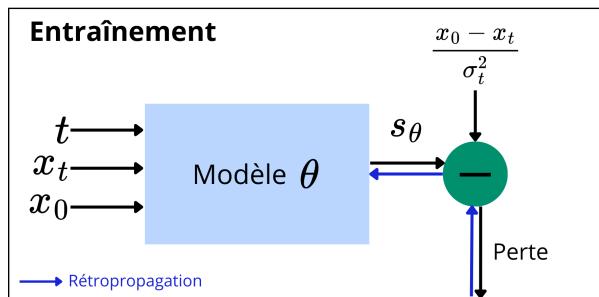


FIGURE 1 – Processus d'entraînement du modèle basé sur le score

2.1.4 Génération de nouvelles données

Une fois le score estimé par s_θ , nous pouvons générer de nouvelles données selon p_θ en reprenant l'équation 1. On a alors, pour tout $t \in \{1, \dots, T\}$:

$$\hat{\mathbf{x}}_0 = \mathbf{x}_t + \sigma_t^2 s_\theta(\mathbf{x}_t, t)$$

2.2 Modèle basé sur le débruitage

Les *Denoising Diffusion Probabilistic Models* (DDPM) de Ho et al. (2020) [6] sont des modèles avec un nombre discret d'étapes de bruitage et pour lesquels les processus de diffusion directe et inverse sont des chaînes de Markov.

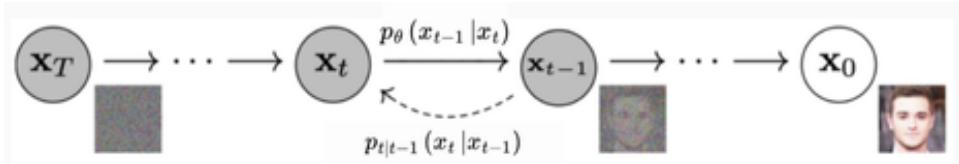


FIGURE 2 – Modèle de diffusion probabiliste basé sur le débruitage (Denoising Diffusion Probabilistic Models, Ho et al., 2020)

2.2.1 Processus de diffusion direct

Dans le cadre d'étude de [6], le processus direct du modèle de diffusion est défini par :

- Distribution des échantillons inconnue : $p_0(\mathbf{x}_0)$
- Ajout d'un bruit Gaussien à chaque étape selon le noyau de transition : $p_{t|t-1}(\mathbf{x}_t|\mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1 - \beta_t}\mathbf{x}_{t-1}, \beta_t\mathbf{I}_d)$

$$\text{pour tout } t \in \{1, \dots, T\} : \quad \mathbf{x}_t = \sqrt{1 - \beta_t}\mathbf{x}_{t-1} + \sqrt{\beta_t}\mathbf{z}_t$$

où la variance $(\beta_t)_{1 \leq t \leq T}$ qui permet de contrôler le bruit ajouté est fixée² et $\mathbf{z}_t \sim \mathcal{N}(\mathbf{0}, \mathbf{I}_d)$

- Ajout de bruit en une étape, selon le noyau de transition : $p_{t|0}(\mathbf{x}_t|\mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t}\mathbf{x}_0, (1 - \bar{\alpha}_t)\mathbf{I}_d)$, avec $\alpha_t = 1 - \beta_t$ et $\bar{\alpha}_t = \prod_{k=1}^t \alpha_k$ (voir annexe 6.5 pour plus de détails)

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t}\mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t}\varepsilon_t \quad \text{où} \quad \varepsilon_t \sim \mathcal{N}(0, \mathbf{I}_d). \quad (5)$$

2.2.2 Processus de diffusion inverse

Il est facile de comprendre que le processus de diffusion inverse $p_{t-1|t}$ est également une chaîne de Markov (voir annexe 6.6 pour la preuve de ce résultat). L'article [6] se place dans le cas où l'espérance est inconnue et la variance préservée : $p_{t-1|t} = \mathcal{N}(\mu(\mathbf{x}_t, t), \beta_t\mathbf{I}_d)$. L'idée est donc d'estimer $\mu(\mathbf{x}_t, t)$ par un modèle paramétrique $\mu_\theta(\mathbf{x}_t, t)$ pour tout $t \in \{1, \dots, T\}$. Le processus de diffusion inverse ainsi estimé, aura la forme : $p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) = \mathcal{N}(\mu_\theta(\mathbf{x}_t, t), \beta_t\mathbf{I}_d)$ et permettra de débruiter les signaux, en partant d'un signal gaussien centré réduit \mathbf{x}_T , réalisation de $\mathbf{X}_T \sim \mathcal{N}(0, \mathbf{I}_d)$ de la manière suivante :

$$\hat{\mathbf{x}}_{t-1} = \mu_\theta(\mathbf{x}_t, t) + \sqrt{\beta_t}z_t \quad \text{où} \quad z_t \quad \text{est une réalisation de} \quad Z_t \sim \mathcal{N}(0, \mathbf{I}_d) \quad (6)$$

2.2.3 Estimation du modèle

L'objectif est d'estimer le décodeur stochastique markovien $p_{t-1|t}$ selon un modèle paramétrique p_θ .

Pour cela, nous allons utiliser la méthode d'estimation du maximum de vraisemblance. Comme repris dans le cours [1], la maximisation de la vraisemblance (ou la minimisation de la log-vraisemblance) est équivalente à la minimisation de l'évidence (Evidence Lower Bound, ELBO)³, notée L et obtenue par règle de Bayes et inégalité de Jensen :

$$(-\log p_\theta(\mathbf{x}_0)) \leq \mathbb{E}_q \left[-\log \left[\frac{p_\theta(\mathbf{x}_{0:T})}{p_{1:T|0}(\mathbf{x}_{1:T} | \mathbf{x}_0)} \right] \right] := L$$

D'après la formule des probabilités totales combinée à la règle de Bayes, on a :

$$p_\theta(\mathbf{x}_{0:T}) = p(\mathbf{x}_T) \prod_{t=1}^T p_\theta(\mathbf{x}_{t-1}|\mathbf{x}_t) \quad (7)$$

2. Version simplifiée. D'autres modèles utilisent un estimateur de la variance pour chaque signal bruité , voir (Nichol et Dhariwal, 2021)

3. Voir annexe 6.7

où $p_T(\mathbf{x}_T)$ est la distribution a priori fixée et $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}(\mu_\theta(\mathbf{x}_t, t), \beta_t \mathbf{I}_d)$.

Ainsi, on a donc :

$$\begin{aligned} L &= \mathbb{E} \left[-\log p_T(\mathbf{x}_T) - \sum_{t=1}^T \log \frac{p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)}{p_{t|t-1}(\mathbf{x}_t | \mathbf{x}_{t-1})} \right] \\ &= \mathbb{E} \left[\underbrace{D_{\text{KL}}(p_{t|0}(\mathbf{x}_T | \mathbf{x}_0) \| p_T(\mathbf{x}_T))}_{L_T} + \sum_{t=2}^T \underbrace{D_{\text{KL}}(p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t))}_{L_{t-1}} - \underbrace{\log p_\theta(\mathbf{x}_0 | \mathbf{x}_1)}_{L_0} \right] \end{aligned}$$

où $D_{\text{KL}}(p, q)$ est la **divergence de Kullback-Leibler**⁴ entre les distributions p et q , définie par :

$$D_{\text{KL}}(p \| q) = \int_{\mathbb{R}^d} p(\mathbf{x}) \log \left(\frac{p(\mathbf{x})}{q(\mathbf{x})} \right) d\mathbf{x} = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x})} \left[\log \left(\frac{p(\mathbf{x})}{q(\mathbf{x})} \right) \right]$$

Par ailleurs, le calcul de $p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$ en utilisant la règle de Bayes, montre qu'il s'agit d'une distribution normale $\mathcal{N}(\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}_d)$ avec

$$\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \mathbf{x}_0 + \frac{\sqrt{\alpha_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_t \quad \text{et} \quad \tilde{\beta}_t = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t. \quad (8)$$

La preuve de ce résultat est donnée en annexe 6.8.

Ainsi, comme $p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t) \sim \mathcal{N}(\mu_\theta(\mathbf{x}_t, t), \beta_t \mathbf{I}_d)$ et $q(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) \sim \mathcal{N}(\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}_d)$, on utilise l'expression de la divergence KL entre distributions gaussiennes⁵, pour obtenir :

$$\begin{aligned} D_{\text{KL}}(p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t)) &= \frac{1}{\beta_t} \|\mu_\theta(\mathbf{x}_t, t) - \tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0)\|^2 + C \\ L_t &= \mathbb{E} [D_{\text{KL}}(p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) \| p_\theta(\mathbf{x}_{t-1} | \mathbf{x}_t))] = \frac{1}{\beta_t} \mathbb{E} [\|\mu_\theta(\mathbf{x}_t, t) - \tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0)\|^2] + C \end{aligned}$$

où C est une constante qui ne dépend pas de θ .

Comme l'objectif final est de minimiser $L = \sum_{t=0}^T L_t$, on peut essayer de minimiser chaque terme de la somme et donc, de minimiser l'écart entre $\tilde{\mu}$, dont on connaît l'expression, et μ_θ .

Ainsi, comme \mathbf{x}_t est décrit par l'équation (5), on peut montrer que⁶ :

$$\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0) = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \varepsilon_t \right)$$

où $\varepsilon_t \sim \mathcal{N}(0, \mathbf{I}_d)$.

Donc, si on paramètre μ_θ comme

$$\mu_\theta(\mathbf{x}_t, t) = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \varepsilon_\theta(\mathbf{x}_t, t) \right)$$

en dénotant par $\varepsilon_\theta(\mathbf{x}_t, t)$ l'estimateur de ε_t , le bruit ajouté à \mathbf{x}_0 pour obtenir \mathbf{x}_t , alors, la perte est simplement :

$$\begin{aligned} L_t &= \frac{1 - \alpha_t}{1 - \bar{\alpha}_t} \mathbb{E} [\|\varepsilon_\theta(\mathbf{x}_t, t) - \varepsilon_t\|_2^2] + C \\ &= \frac{1 - \alpha_t}{1 - \bar{\alpha}_t} \mathbb{E}_{\mathbf{x}_0, \varepsilon_t} [\|\varepsilon_\theta(\sqrt{\alpha_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon_t, t) - \varepsilon_t\|_2^2] + C \end{aligned}$$

4. La divergence de Kullback-Leibler mesure l'information perdue lorsque q est utilisé pour approximer p

5. Voir annexe 6.9

6. Voir annexe 6.10

On cherche donc à estimer le bruit ε_t ajouté à \mathbf{x}_0 pour obtenir \mathbf{x}_t , pour chaque étape t .

(Ho et al.,2020) propose une simplification de la fonction de perte totale (que l'on chercher à minimiser) par :

$$L_{\text{simple}} = \mathbb{E}_{t, \mathbf{x}_0, \varepsilon_t} \left[\left\| \varepsilon_\theta(\sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon_t, t) - \varepsilon_t \right\|^2 \right]$$

Dans la pratique, la perte totale est calculée en tirant aléatoirement avec remise un instant t et un bruit ε_t que l'on associe à une des séries initiales multivariées \mathbf{x}_0 du lot considéré. Le modèle calcule l'estimateur ε_θ à partir des paramètres d'entrée et la perte est alors calculée par le carré de la différence entre ε_t et son estimateur. La perte totale représente la somme des pertes sur toutes les étapes de bruitage t tirées et sur toutes les séries réelles \mathbf{x}_0 considérées.

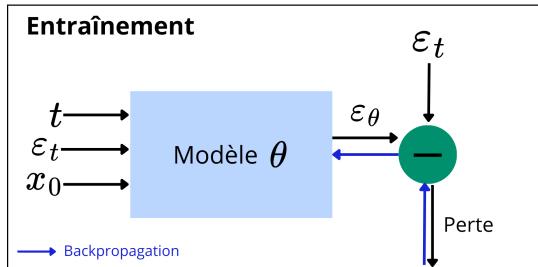


FIGURE 3 – Processus d’entraînement du modèle basé sur le débruitage

2.2.4 Génération de nouvelles données après estimation du modèle paramétrique

En reprenant 6, on peut générer des signaux selon une approximation de la loi $p_0(\mathbf{x}_0)$ en partant de \mathbf{x}_T une réalisation de $\mathbf{x}_T \sim p_T(\mathbf{x}_T)$, selon :

$$\hat{\mathbf{x}}_{t-1} = \frac{1}{\sqrt{\alpha_t}} \left(\mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \varepsilon_\theta(\mathbf{x}_t, t) \right) + \sqrt{\beta_t} z_t, \quad z_t \sim \mathcal{N}(0, \mathbf{I}_d)$$

L'estimateur de \mathbf{x}_0 obtenu par ce mécanisme vérifie :

$$\hat{\mathbf{x}}_0 = \frac{1}{\sqrt{\alpha_t}} \mathbf{x}_t - \sqrt{\frac{1}{\sqrt{\alpha_t}} - 1} \varepsilon_\theta(\mathbf{x}_t, t)$$

2.3 Modèles adaptés pour les séries temporelles

Ces deux types de modélisations ont été adaptées pour la génération de séries temporelles, avec par exemple le Time Series Generative Model (TSGM) [7] qui est basé sur le score et utilise un encodeur et un décodeur pour estimer la fonction score. Bien que le modèle génère des échantillons de haute qualité et diversifiés, le processus de génération est computationnellement coûteux et génère des séries relativement courtes. Des modèles plus récents comme TransFusion [8] et Diffusion-TS [11] utilisent des Transformers pour estimer le processus de débruitage. Les mécanismes d'attention des transformers permettent de mieux capturer les dépendances à long terme des données et donc de générer des séries plus longues. Diffusion-TS utilise une fonction de perte basée sur les transformées de Fourier pour une génération plus précise et reconstitue la série selon une tendance et une saisonnalité avec des termes d'erreur. Nous allons commencer par l'étude du modèle de Diffusion-TS qui est un modèle récent utilisant des techniques nouvelles tout en restant interprétable grâce à sa reconstitution du signal selon une tendance et une saisonnalité. L'idée sera de comprendre l'architecture du modèle et d'évaluer les caractéristiques qu'il réussit à apprendre en plus du critère donné. En effet, les séries temporelles financières ont des particularités propres (les faits stylisés), et nous cherchons à savoir si ces propriétés statistiques sont également présentes dans les données générées, bien qu'elles ne soient pas un critère pour l'entraînement du modèle.

3 Méthodologie de résolution

3.1 Le modèle retenu

Le modèle retenu est celui de Diffusion-TS [11], qui combine des techniques de décomposition de tendance et de saisonnalité avec un processus de diffusion basé sur le débruitage. Il génère des séries temporelles qualitatives et variées grâce à une fonction de perte fondée sur les transformées de Fourier et une architecture de type transformer, offrant ainsi une solution efficace et interprétable pour la génération de séries temporelles.

3.1.1 Théorie stochastique

Le modèle de Diffusion-TS se base sur le modèle de diffusion par débruitage introduit dans 2.2, mais apprend directement à estimer le signal débruité \mathbf{x}_0 plutôt que le bruit ε_t ajouté à \mathbf{x}_0 pour obtenir un signal bruité \mathbf{x}_t . Par ailleurs, les auteurs de l'article n'utilisent pas l'hypothèse de préservation de la variance, comme introduit dans [9]. En gardant les mêmes notations que précédemment, pour $t \in \llbracket 1, T \rrbracket$, on définit :

- **le processus de diffusion direct** : $p_{t|t-1}(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\sqrt{1 - \beta_t} \mathbf{x}_{t-1}, \beta_t \mathbf{I}_d)$

$$\mathbf{x}_t = \sqrt{1 - \beta_t} \mathbf{x}_{t-1} + \sqrt{\beta_t} \mathbf{z}_t \quad \text{où} \quad z_t \sim \mathcal{N}(0, \mathbf{I}_d).$$

- le processus de diffusion direct en une étape : $p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I}_d)$, avec $\alpha_t = 1 - \beta_t$ et $\bar{\alpha}_t = \prod_{k=1}^t \alpha_k$

$$\mathbf{x}_t = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon_t \quad \text{où} \quad \varepsilon_t \sim \mathcal{N}(0, \mathbf{I}_d).$$

- **le processus de diffusion inverse** : en partant d'une réalisation \mathbf{x}_T de $\mathbf{X}_T \sim p_T$, on considère : $p_{t-1|t}(\mathbf{x}_{t-1} | \mathbf{x}_t) = \mathcal{N}\left(\mu(\mathbf{x}_t, t), \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t\right)$

Comme dans 2.2, on cherche à estimer la moyenne de la loi gaussienne mais cette fois-ci, on paramétrise $\mu_\theta(\mathbf{x}_t, t)$ en faveur de $\hat{\mathbf{x}}_0$ plutôt qu'en faveur du bruit ε_θ , en utilisant l'expression de $\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0)$ dans l'équation 8.

Ainsi, pour chaque instant $t \in \llbracket 1, T \rrbracket$, on obtient l'expression de \mathbf{x}_{t-1} en fonction de \mathbf{x}_t et de $\hat{\mathbf{x}}_0$, l'estimateur de \mathbf{x}_0 :

$$\mathbf{x}_{t-1} = \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \hat{\mathbf{x}}_0(\mathbf{x}_t, t, \theta) + \frac{\sqrt{\bar{\alpha}_t}(1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_t + \sqrt{\frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t} z_t, \quad z_t \sim \mathcal{N}(0, \mathbf{I}_d) \quad (9)$$

$\hat{\mathbf{x}}_0(\mathbf{x}_t, t, \theta)$ est un estimateur de \mathbf{x}_0 obtenu par le modèle de paramètres θ qui est entraîné en minimisant la perte :

$$\mathcal{L}_\theta = \mathbb{E}_{t, \mathbf{x}_0} \left[w_t \left[\lambda_1 \|\mathbf{x}_0 - \hat{\mathbf{x}}_0(\mathbf{x}_t, t, \theta)\|^2 + \lambda_2 \|\mathcal{FFT}(\mathbf{x}_0) - \mathcal{FFT}(\hat{\mathbf{x}}_0(\mathbf{x}_t, t, \theta))\|^2 \right] \right] \quad (10)$$

avec $w_t = \frac{\lambda \alpha_t(1 - \bar{\alpha}_t)}{\beta_t^2}$, $\lambda = 0.01$ une constante, λ_1, λ_2 deux poids pour équilibrer les deux pertes et où \mathcal{FFT} dénote la transformée de Fourier.

La minimisation de cette fonction objectif dans la phase d'entraînement permet de réduire le carré de la distance entre \mathbf{x}_0 et $\hat{\mathbf{x}}_0$ ainsi qu'entre leur transformée de Fourier. La partie pratique de l'entraînement et de la génération sera développée dans la section suivante.

Par ailleurs, le signal $\hat{\mathbf{x}}_0$ est reconstitué selon une tendance, des saisonnalités et un terme d'erreur. La composante tendancielle est déterminée par des régresseurs polynomiaux et la composante saisonnière par la recomposition d'un signal à partir des K plus grandes fréquences de la transformée de Fourier du signal, où K est un entier fixé. Cette technique de reconstitution selon une tendance, des saisonnalités et un terme d'erreur se fait grâce à

une architecture de décodeur composé de plusieurs couches selon la figure 4. L'architecture complète comprend également un encodeur, une figure est présente en annexe 6.11 pour plus de détails à ce propos.

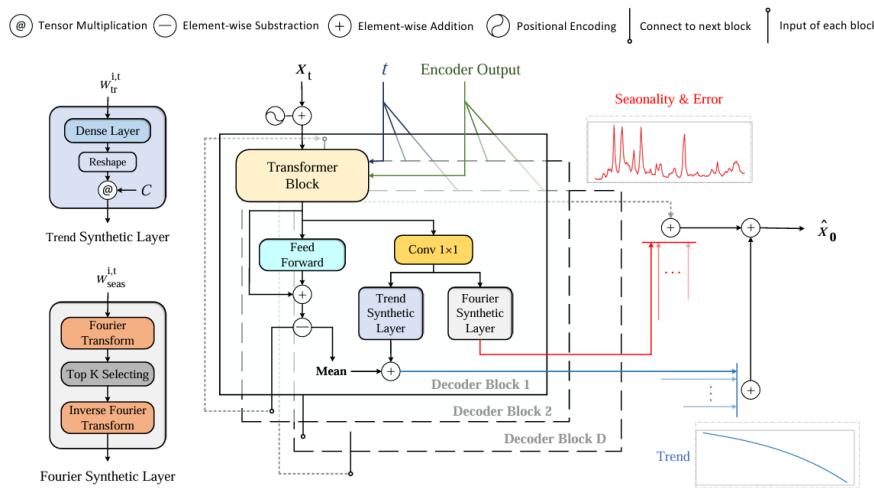


FIGURE 4 – Architecture du décodeur utilisé dans Diffusion-TS [11].

3.1.2 Algorithmique et implémentation

Dans cette section est introduit le mécanisme utilisé dans Diffusion-TS et sont expliquées l'implémentation de l'entraînement et de la génération, ainsi que les différents paramètres utilisés. L'architecture du code est donné en annexe 6.12.

Algorithm 1 Boucle d'entraînement

Input: S : Nombre de pas d'entraînement, N : Nombre de batchs avant mise à jour des paramètres, b : Taille d'un batch de données

```

1  $s \leftarrow 0$ 
2 while  $s < S$  do
3    $total\_loss \leftarrow 0$ 
   for  $n = 1$  to  $N$  do
     Charger le batch de données  $S_n : \{\mathbf{x}_0^i\}_{i=1}^b$ 
     Calculer la perte sur le batch des données  $S_n : loss/N$ 
     Calculer et accumuler les gradients
      $total\_loss \leftarrow total\_loss + loss$ 
   
```

4 Clipping des gradients
Mise à jour des paramètres θ_s avec l'optimiseur Adam
Ajuster le taux d'apprentissage η_s
 $s \leftarrow s + 1$
Mettre à jour la moyenne mobile exponentielle : $\bar{\theta}_{s+1} \leftarrow \beta \bar{\theta}_s + (1 - \beta) \theta_s$

L'annexe 6.13 précise la manière dont l'optimiseur Adam fonctionne.⁷

7. Par ailleurs, le clipping des gradients est une technique utilisée pour empêcher que les gradients deviennent trop grands pendant la rétropropagation dans l'entraînement, ce qui peut conduire à l'instabilité ou à l'explosion des gradients. Dans ce cas, la norme des gradients sera limitée à 1.0. Si la norme totale des gradients dépasse cette valeur, les gradients seront réduits proportionnellement.

A chaque passage dans la boucle d'entraînement, une perte est calculée et permet de mettre à jour les paramètres du modèle après calcul du gradient par rétropropagation. Voici un schéma de la manière dont est calculée la perte entre les signaux réels du batch de données considéré et leur estimation par le modèle de paramètres θ . On considère un ensemble de données réelles d'entraînement constitué de plusieurs séries multivariées de feature_size coordonnées, chacune de longueur seq_length.

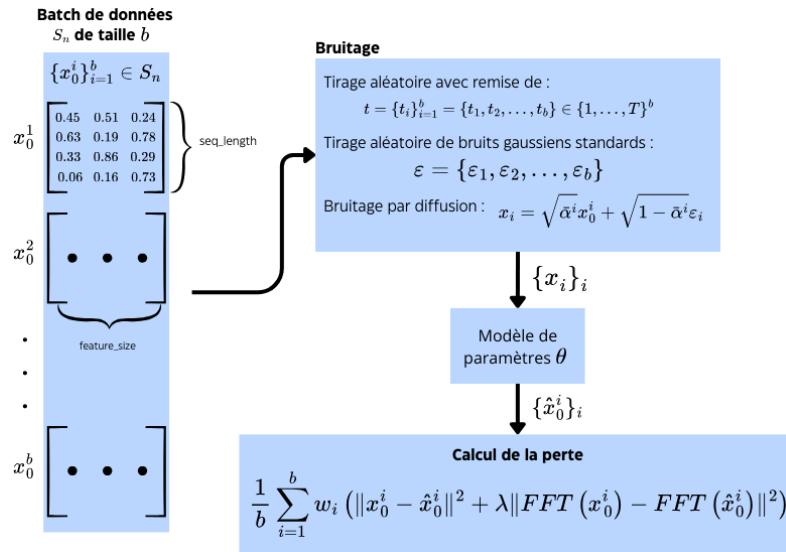


FIGURE 5 – Calcul de la perte dans un batch de données

avec $w_i = \frac{\sqrt{\alpha_{t_i}}(1-\bar{\alpha}_{t_i})}{100\beta_{t_i}}$ pour tout $i \in \{1, \dots, b\}$, b la taille d'un batch est fixée et correspond au nombre de séries temporelles dans le batch et $\lambda = \frac{\sqrt{\text{seq_length}}}{5}$.

De manière plus simple, la figure suivante décrit comment la perte est calculée, à partir d'un lot de signaux x_0 et d'une étape de bruitage aléatoire t , ainsi que le processus de génération, qui est séquentiel contrairement à l'entraînement.

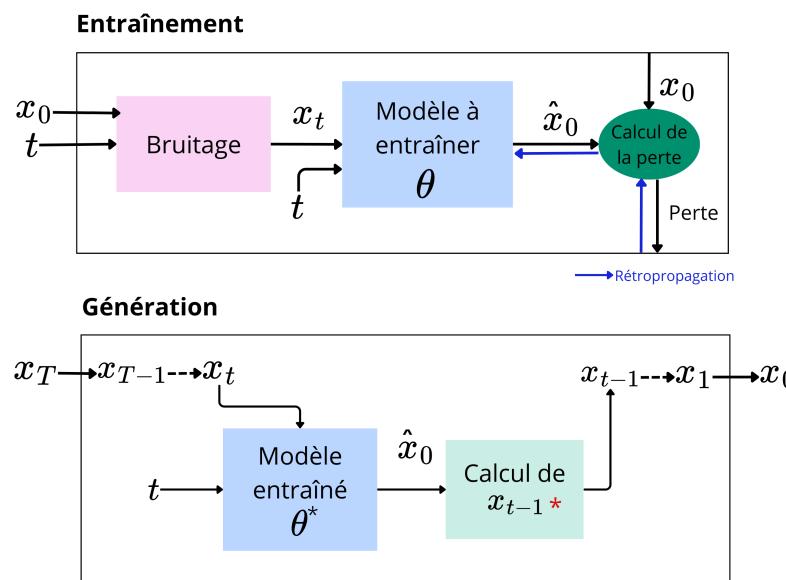


FIGURE 6 – Entrainement et génération

L'idée de l'entraînement est de partir d'un signal x_0 réel et d'une étape de bruitage choisie aléatoirement $t \in \{1, \dots, T\}$, afin d'obtenir le signal bruité x_t à l'étape considérée t . Le modèle de paramètre θ va permettre

d'obtenir une image $\hat{\mathbf{x}}_0$ de \mathbf{x}_0 à partir de \mathbf{x}_t et de t , puis la perte est calculée en comparant les deux signaux \mathbf{x}_0 et $\hat{\mathbf{x}}_0$. L'entraînement ne se fait pas de manière séquentielle, les étapes de bruitage considérées dans l'entraînement sont tirées aléatoirement avec remise, ceci afin de réduire le coût computationnel.

Le processus de génération, quant à lui, se fait de manière séquentiel, en partant d'un bruit blanc gaussien \mathbf{x}_T et en le débruitant en T étape. A chaque étape le modèle entraîné de paramètre θ^* prend en entrée l'étape de bruitage actuelle t et le signal bruité \mathbf{x}_t et reconstitue une image $\hat{\mathbf{x}}_0$. Le signal débruité \mathbf{x}_{t-1} à l'étape $t - 1$ est ensuite obtenu grâce à l'équation 9.

Dans les expériences qui suivront, nous auront besoin de fixer des valeurs pour des hyperparamètres du modèle, que nous précisons ci-dessous.

Paramètre	Définition
seq_length	Longueur de la série temporelle
feature_size	Nombre de coordonnées de la série
batch_size	Taille d'un lot de données dans le processus d'entraînement (nombre de séries temporelles dans le lot)
max_epochs	Nombre maximal d'epochs pendant lesquelles le modèle sera entraîné, c'est-à-dire le nombre maximum de passages complets à travers l'ensemble des données d'entraînement (S dans l'algorithme 1)
save_cycle	Détermine la fréquence à laquelle le modèle est sauvegardé pendant l'entraînement
timesteps	Nombre total d'étapes de bruitage dans le processus de diffusion
n_heads	Nombre de têtes d'attention dans le mécanisme d'attention multi-têtes des Transformers
n_layer_encod	Nombre de couches dans le bloc encodeur du modèle
n_layer_dec	Nombre de couches dans le bloc décodeur du modèle
patience	Nombre d'epochs à attendre sans amélioration avant d'arrêter l'entraînement (permet d'éviter l'overfitting)

TABLE 1 – Définitions des hyperparamètres du modèle

3.1.3 Paramètres de bruitage choisis en pratique

Les paramètres de bruitage $(\beta_t)_t$ choisis dans les expériences sont définis comme ce qui suit, pour un nombre d'étapes de bruitage $T = 500$ et un paramètre $s = 0.008$. La méthode utilisée pour les calculer est dite sinusoïdale.

Pour $t \in \{0, \dots, T\}$:

- $\bar{\alpha}_t = \frac{1}{a_s} \cos^2 \left(\frac{(\frac{t}{T} + s)}{1+s} \cdot \frac{\pi}{2} \right)$ où $a_s = \cos \left(\frac{s}{1+s} \frac{\pi}{2} \right)$
- $\beta_t = 1 - \frac{\bar{\alpha}_{t+1}}{\bar{\alpha}_t}$ pour $t = 0, 1, 2, \dots, T - 1$
- $\beta_t = \max(0, \min(\beta_t, 0.999))$

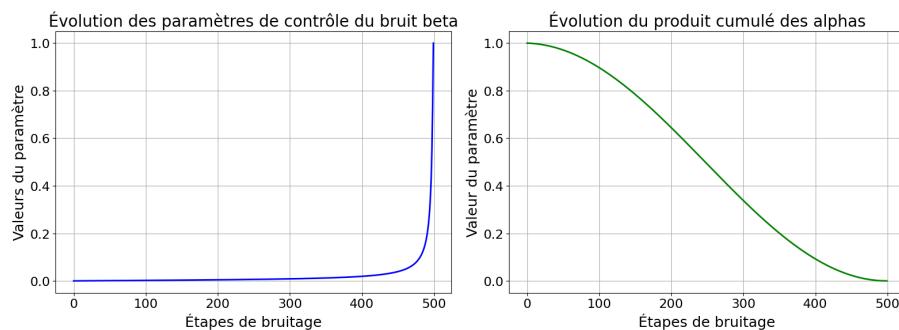


FIGURE 7 – Evolution des paramètres de contrôle de bruit

Remarque 1. Quelque soient les valeurs de T , on remarque qu'après T étapes de bruitages la valeur de β_T est très proche de 1 (on l'interdit d'être égal à 1 par définition) et que celle du produit cumulé des α est très proche de 0.

3.2 Les données d'entraînement

Pour entraîner le modèle, deux ensembles de données financières ont été utilisés. Le premier est constitué de l'évolution journalière des prix des 16 produits financiers les plus capitalisés du S&P500⁸, couvrant une période de 2400 jours, du 2 janvier 2015 au 17 juillet 2024. Ces données incluent des actions majeures telles que Apple, Microsoft, Amazon, et Google, avec les prix exprimés en dollars, récupérés via la plateforme Yahoo-Finances⁹. Le second ensemble de données est constitué de l'évolution à la minute du prix du Bitcoin, entre le 1er janvier 2017 à 00h00 et le 6 août 2021 à 06h42, totalisant 2 416 722 minutes. Ces données ont été extraites de la plateforme Binance, et les prix sont également exprimés en dollars.

Les données que nous considérons présentent des propriétés statistiques particulières qui sont communes à la plupart des actifs financiers. C'est ce que l'on appelle les **faits stylisés**. Parmi les faits stylisés, nous avons par exemple l'absence d'autocorrélations des rendements, l'existence de clusters de volatilité et la présence de queues de distributions des rendements lourdes. Nous allons nous concentrer sur ces trois faits stylisés pour la présente analyse descriptive des données.

Remarque 2. La plupart des propriétés statistiques propres aux données financières sont relatives aux rendements des actifs, plutôt qu'à leur prix. Notons p_t le prix d'un actif à l'instant t . Le **rendement** d'un actif à l'instant t , noté r_t , est défini comme la variation relative du prix de cet actif entre l'instant $t - 1$ et l'instant t . Il existe deux façons courantes de calculer ce rendement :

- Le **rendement simple** à l'instant t est donné par :

$$r_t = \frac{P_t - P_{t-1}}{P_{t-1}} = \frac{P_t}{P_{t-1}} - 1$$

où P_t est le prix de l'actif à l'instant t et P_{t-1} est le prix de l'actif à l'instant $t - 1$.

- Le **rendement logarithmique** à l'instant t est défini comme le logarithme du ratio des prix successifs :

$$r_t = \ln \left(\frac{P_t}{P_{t-1}} \right)$$

Le rendement logarithmique est souvent préféré en finance car il possède des propriétés mathématiques utiles, comme l'additivité sur des périodes successives et c'est celui que nous considérerons dans la suite.

8. Le S&P 500 (Standard & Poor's 500) est un indice boursier qui regroupe 500 des plus grandes entreprises cotées aux États-Unis. Il est souvent utilisé comme un indicateur de la performance globale du marché boursier américain. Ces 500 entreprises couvrent une large gamme de secteurs d'activités, ce qui en fait un indice représentatif de l'économie américaine. Le S&P 500 est pondéré en fonction de la capitalisation boursière, ce qui signifie que les entreprises avec une plus grande capitalisation ont un poids plus important dans l'indice.

9. Voici les 16 actifs financiers les plus capitalisés du S&P500 utilisés dans cette étude : Apple, Microsoft, Amazon, Nvidia, Google (Alphabet), PepsiCo, Costco, Adobe, Bank of America, Cisco Systems, Pfizer, Thermo Fisher Scientific, McDonald's, Accenture, Salesforce .

3.2.1 Les données du Bitcoin

	Prix (\$)	Rendements logarithmiques
Nombre	2 416 671	2 416 670
Moyenne	12 211	1.548×10^{-6}
Ecart-type	13 395	1.338×10^{-3}
Minimum	752	-9.140×10^{-2}
Quantile 25%	4 943	-4.680×10^{-4}
Quantile 50%	8 112	0.000
Quantile 75%	11 091	4.802×10^{-4}
Maximum	64 843	9.996×10^{-2}

FIGURE 8 – Résumé statistique des données

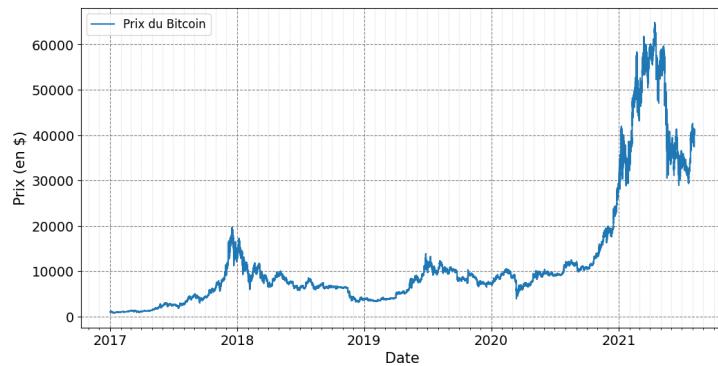


FIGURE 9 – Évolution temporelle du prix du Bitcoin

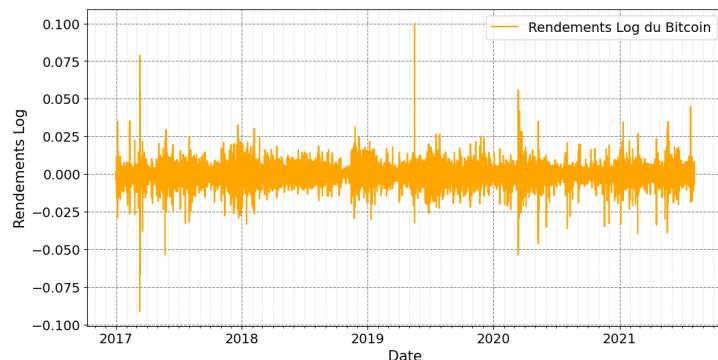


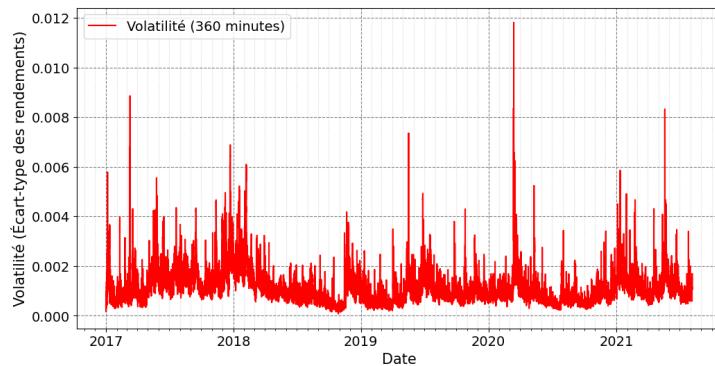
FIGURE 10 – Évolution temporelle des rendements du Bitcoin

La série de prix du Bitcoin présente des phases de tendances marquées ainsi que des pics et des chutes soudaines et plus ou moins importants, caractéristiques d'une volatilité significative.

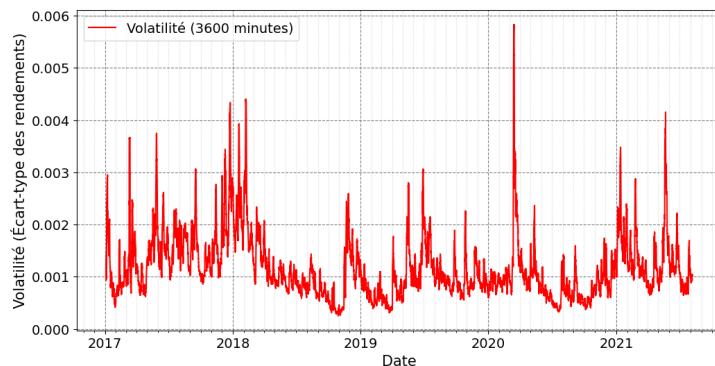
Par ailleurs, dans des données financières, on s'attend à une **absence d'autocorrélation des rendements**¹⁰. En observant le graphique de l'évolution temporelle des rendements du Bitcoin, on remarque que ceux-ci semblent évoluer de façon aléatoire autour de zéro, sans tendance marquée ni saisonnalité, ce qui suggère un manque de corrélation à court terme. Le graphique montre également des valeurs de rendements extrêmes, typiques des **queues de distribution lourdes**¹¹. Les queues épaissest se réfèrent à une distribution des rendements qui présente une fréquence plus élevée d'événements extrêmes (gains ou pertes très importants) par rapport à une distribution normale (qui suit la loi de Gauss). Cela signifie que les rendements ont plus de chances de connaître des fluctuations extrêmes que ce que l'on attendrait d'une distribution normale.

10. L'absence d'autocorrélation signifie que les rendements passés d'un actif ne prédisent pas les rendements futurs. Autrement dit, il n'y a pas de relation systématique entre les rendements d'une période à une autre. Les rendements sont donc en grande partie indépendants et suivent un mouvement aléatoire. Cette observation soutient l'hypothèse d'efficience des marchés.

11. La présence de queues épaissest a des implications importantes pour la gestion des risques, car cela signifie qu'il y a une probabilité plus élevée d'événements extrêmes, tels que des krachs boursiers ou des bulles. Cela remet en question l'utilisation de modèles basés sur l'hypothèse de normalité des rendements (comme le modèle de Black-Scholes pour les options), et incite les analystes à adopter des modèles qui prennent en compte des distributions avec des queues épaissest, comme la distribution de Lévy ou d'autres modèles de distributions robustes.



(a) Volatilité calculée sur une fenêtre glissante de 360 minutes (6 heures)



(b) Volatilité calculée sur une fenêtre glissante de 3600 minutes (60 heures ou 2,5 jours)

FIGURE 11 – Évolution temporelle de la volatilité des rendements du Bitcoin, calculée sur différentes fenêtres de temps.

Les graphiques ci-dessus représentent la volatilité des rendements, c'est-à-dire l'écart-type calculé sur une fenêtre glissante de 360 ou 3600 minutes. On peut voir des périodes de fortes volatilité suivies par des périodes de faible volatilité, bien qu'ici les périodes de haute volatilité soient assez courtes comparées à celles de faible volatilité. Les **clusters de volatilité** se réfèrent à cette tendance des données financières de présenter des périodes de grandes volatilité suivies par des périodes de faibles volatilité. Par ailleurs, ils sont davantage observables sur le deuxième graphique pour lequel la volatilité est calculée sur une fenêtre plus longue.

Ce phénomène a pour conséquence que les mouvements de prix tendent à se concentrer en vagues, alternant entre des phases de calme et des phases de forte activité. Cela est souvent observé dans des contextes de crise ou d'incertitude économique. Les investisseurs peuvent réagir de manière collective à des nouvelles ou à des événements, provoquant une volatilité accrue.

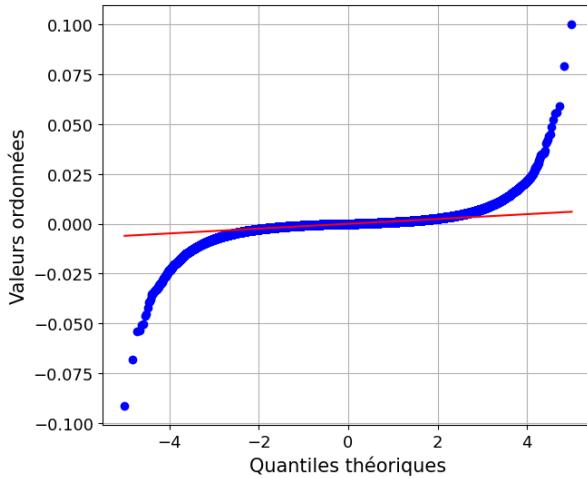


FIGURE 12 – QQ plot des rendements du Bitcoin

Dans le QQ-plot des rendements, on peut voir des points proches de la ligne centrale pour les quantiles intermédiaires (indiquant une similitude avec la distribution normale dans la partie centrale), mais s'éloignant dans les extrémités. Cela montre la présence de queues lourdes : les valeurs extrêmes de rendements sont plus fréquentes que ce qu'une distribution normale prédirait, comme expliqué précédemment.

3.2.2 Les données du S&P500

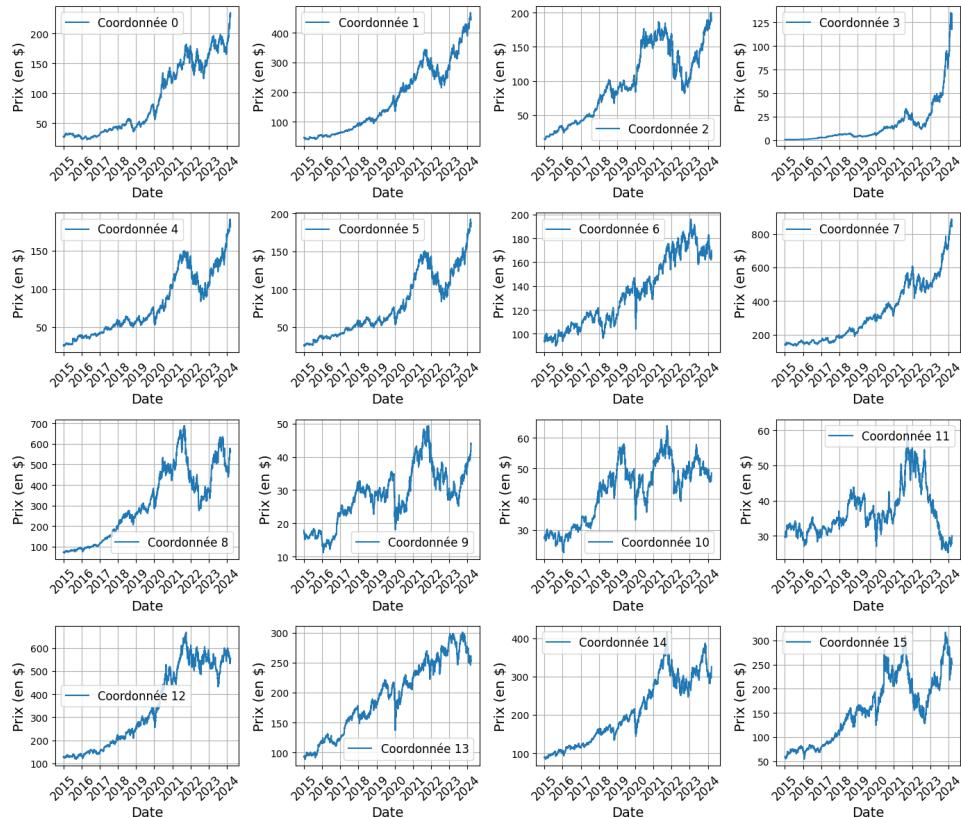


FIGURE 13 – Évolution temporelles des prix des 16 produits les plus capitalisés du S&P500

La série de prix des 16 produits les plus capitalisés du S&P500 révèle des phases de tendance marquées, avec des fluctuations significatives à court terme comme le montre la figure 13. Ces variations et changements abrupts

de direction reflètent l'impact des éléments économiques, financiers et politiques et sont typiques d'une volatilité élevée.

Les séries montrent également des corrélations significatives entre elles, avec une tendance haussière dominante jusqu'en 2020-2021, suivie de variations brusques liées aux chocs économiques, notamment la crise du Covid-19. Ces évolutions suggèrent une réponse collective des actifs aux grandes dynamiques de marché, surtout en période d'incertitude.

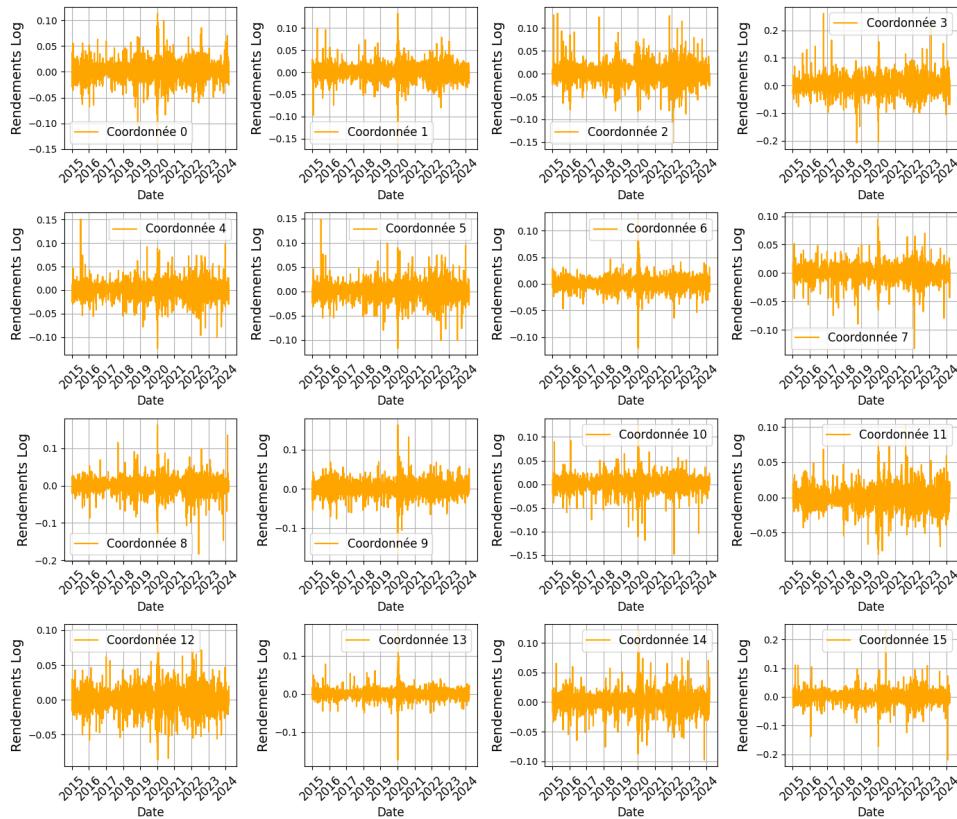


FIGURE 14 – Évolution temporelle des rendements des 16 produits les plus capitalisés du S&P500

Les rendements calculés à partir des séries de prix semblent évoluer autour de zéro de manière apparemment aléatoire, sans pattern de tendance, saisonnalité ni corrélation temporelle immédiate, avec une grande partie des valeurs comprises entre -0.05 et +0.05. Ce caractère (potentiellement) aléatoire suggère une absence d'autocorrélation à court terme, ce qui est en accord avec l'hypothèse de l'efficience des marchés, où les informations passées ne permettent pas de prévoir les rendements futurs. Par ailleurs, des valeurs de rendement extrêmes sont visibles dans les graphiques, et suggèrent des queues de distribution lourdes.

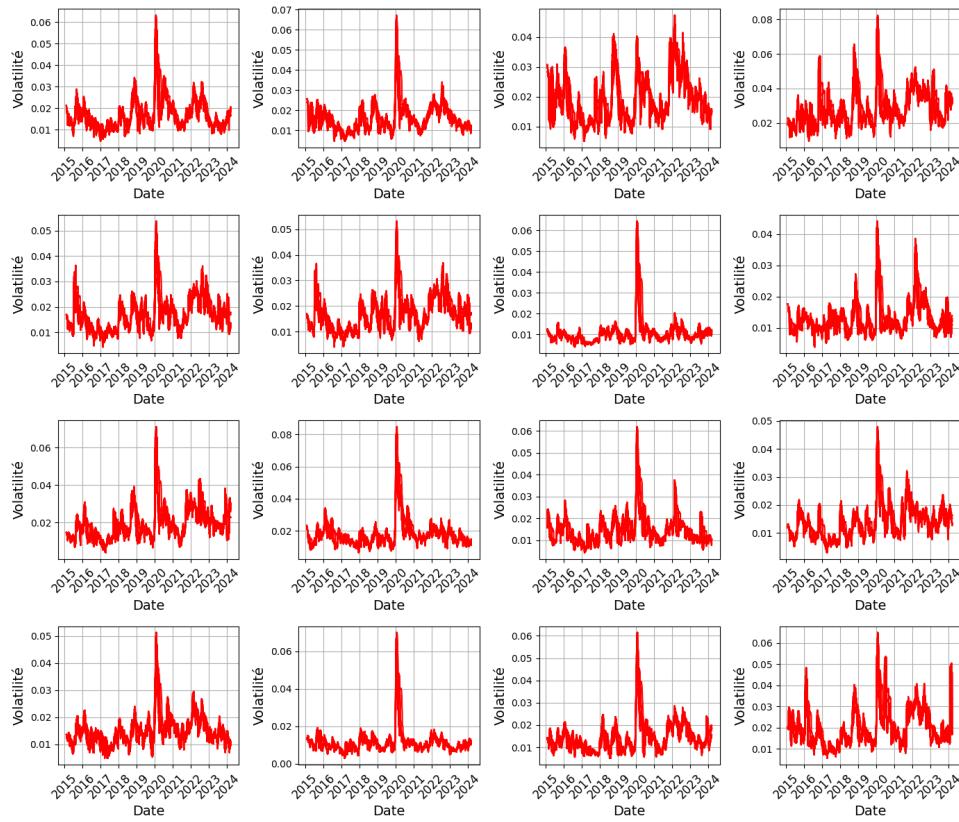


FIGURE 15 – Évolution temporelle de la volatilité des rendements des 16 produits les plus capitalisés du S&P500

Les graphiques illustrant la volatilité des rendements, calculée sur une fenêtre glissante de 25 jours (l'équivalent d'un mois boursier), mettent en lumière la présence de clusters. On observe en effet des périodes de forte volatilité suivies de phases de volatilité réduite. Les clusters de volatilité sont ici plus évidents que pour le Bitcoin. Cela est probablement dû au fait que le volume de données est mille fois inférieur à celui du Bitcoin, ce qui rend les clusters plus faciles à identifier. Ce comportement illustre la tendance des investisseurs à réagir collectivement aux nouvelles et aux événements économiques, ce qui accentue les fluctuations de prix en période d'incertitude.

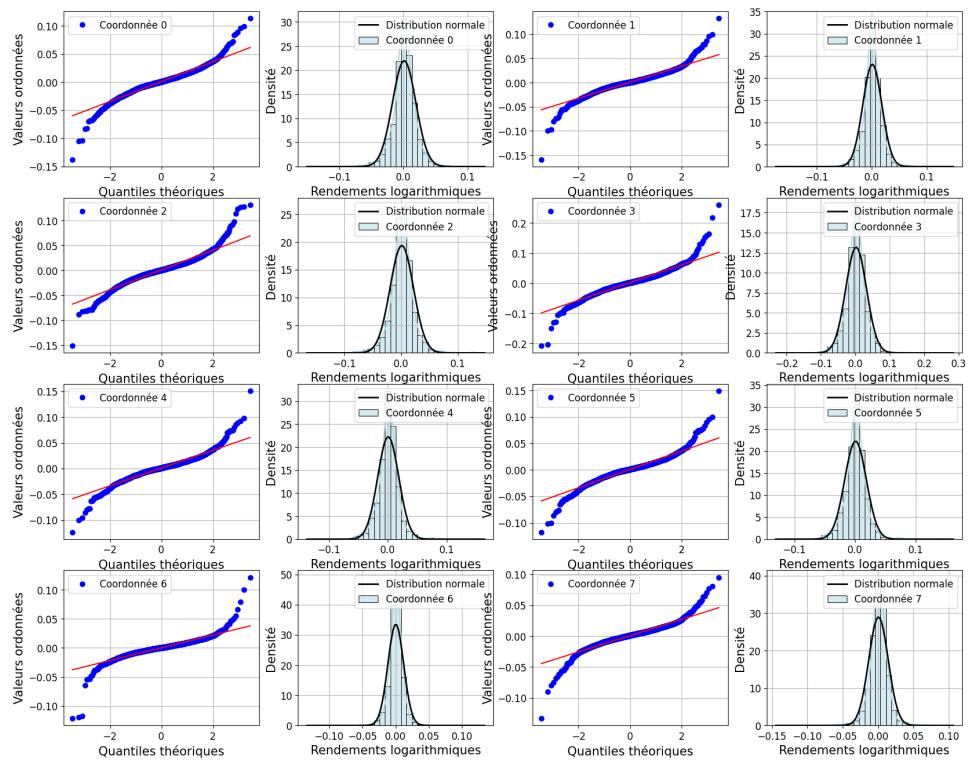


FIGURE 16 – QQ plot et histogramme des rendements des 16 produits les plus capitalisés du S&P500

Dans le QQ-plot des rendements des actifs du S&P500, les points restent proches de la ligne centrale pour les quantiles moyens, suggérant une ressemblance avec la distribution normale dans la partie centrale, confirmé par l'histogramme. Cependant, aux extrémités du QQ-plot, les points s'éloignent nettement de la ligne, confirmant la présence de queues lourdes. Cela indique que les valeurs extrêmes sont plus fréquentes dans les rendements que ce qu'une distribution normale pourrait prédire, traduisant un risque accru d'événements exceptionnels.

Remarque 3. Les séries des rendements ne présentent pas de tendances ou de saisonnalités significatives, contrairement aux séries des prix. Étant donné que le modèle de Diffusion-TS reconstitue les séries selon des tendances et des saisonnalités, l'idée est donc d'utiliser le modèle pour les séries des prix, puis de calculer les rendements à partir des prix générés afin de vérifier les faits stylisés.

3.2.3 Pré-traitement des données

Avant d'être utilisées dans le processus d'entraînement du modèle, les données subissent le pré-traitement suivant :

- Normalisation min-max sur chaque coordonnée de la série de manière indépendante, de façon à avoir des valeurs dans $[0, 1]$. La transformation s'effectue selon la formule suivante : $x' = \frac{x - \min(x)}{\max(x) - \min(x)}$ avec :
 - x est la valeur d'origine de la variable,
 - $\min(x)$ et $\max(x)$ sont respectivement la plus petite et la plus grande valeur de la variable dans l'ensemble de données,
 - x' est la valeur normalisée.
- Transformation dans $[-1, 1]$, en utilisant la formule suivante : $x'' = 2x' - 1$ avec : où :
 - x' est la valeur normalisée dans l'intervalle $[0, 1]$,

— x'' est la nouvelle valeur transformée dans l'intervalle $[-1, 1]$.

- Segmentation des données en une collection de séquences de longueur seq_length fixée, en utilisant une fenêtre glissante de longueur unitaire.

Par ailleurs, les données manquantes du Bitcoin ont été imputées par la valeur moyenne avant toute autre opération.

4 Expériences réalisées et résultats

Les expériences que nous avons d'abord réalisées ont consisté à évaluer trois métriques : la perte d'entraînement, le score context-FID et la perte de corrélation croisée, et ce pour différentes valeurs des hyperparamètres du mécanisme.

4.0.1 Les premières métriques utilisées

- **Perte d'entraînement** : La perte d'entraînement mesure l'écart entre les séries temporelles générées par le modèle et les séries temporelles réelles utilisées dans l'entraînement. Elle est ici calculée à l'aide de la fonction de perte définie dans l'équation 10, et indique dans quelle mesure le modèle apprend à reproduire les données d'entraînement. Une perte d'entraînement plus faible indique une meilleure performance du modèle.
- **Context-FID** : Le score de Context-FID correspond à la distance FID (Fréchet Inception Distance) entre les représentations de séries temporelles réelles et générées. Elle évalue la similarité des distributions en prenant en compte à la fois la différence entre les moyennes des représentations et la dissimilarité des covariances. Un score FID faible indique que les séries générées ont une distribution similaire aux séries réelles (du point de vue des moyenne et variance), ce qui signifie que les générées capturent bien les caractéristiques des données réelles (voir annexe 6.14.1 pour plus de précisions).
- **Perte de corrélation croisée** : La perte de corrélation croisée mesure la différence absolue entre les corrélations croisées des séries temporelles réelles et simulées, en comparant leurs dépendances temporelles entre différentes dimensions. Elle pénalise les écarts pour aligner les structures de corrélation des séries générées avec celles des séries réelles. Une perte de corrélation croisée plus faible indique une meilleure performance du modèle (voir annexe 6.14.2 pour plus de précisions).

Nous nous sommes ensuite servis de ces premiers résultats pour sélectionner les modèles que nous utiliserons pour évaluer la qualité des données générées par rapport à la satisfaction des faits stylisés caractéristiques des données financières.

Paramètre	Valeur
Timestep T	500
Seq_length	120, 128 ou 250
Feature_size	2, 3 ou 16
Batch_size	128
n_layer_enc	3
n_layer_dec	2 ou 3
max_epochs	10000, 30000 ou 50000
save_cycle	max_epochs / 10
patience	2000, 6000 ou 10000

FIGURE 17 – Valeurs des paramètres utilisées dans les expériences

Numéro de l'expérience	Origine des données	Longueur totale de la série initiale	Longueur de la séquence	n_layer_dec	max_epochs	patience	nombre de coordonnées	Perte d'entraînement	Context-FID	Perte de corrélation croisée
1	SNP500	2400	128	3	30000	6000	16	0.0304	0.0990 ± 0.0064	0.2407 ± 0.1230
2	SNP500	2400	128	2	10000	2000	16	0.0374	0.1061 ± 0.0119	0.1361 ± 0.0884
3	SNP500	2400	128	2	30000	6000	16	0.0309	0.1013 ± 0.0119	0.2753 ± 0.1514
4	SNP500	2400	128	3	10000	2000	16	0.0346	0.0642 ± 0.0075	0.1417 ± 0.0820
5	SNP500	2400	128	3	50000	10000	16	0.0301	0.0747 ± 0.0043	0.2103 ± 0.1504
6	SNP500	2400	250	3	30000	6000	16	0.0260	0.1287 ± 0.0059	0.2298 ± 0.1253
7	SNP500	2400	250	3	30000	6000	3	0.0414	0.1312 ± 0.0283	0.0030 ± 0.0025
8	Bitcoin	2400	120	3	30000	6000	2 (*)	0.0668	0.1392 ± 0.0204	0.0005 ± 0.0003
9	Bitcoin	2880	120	3	30000	6000	2 (*)	0.0573	0.0601 ± 0.0219	(0.0030 ± 0.0018) × 10 ⁻¹
10	SNP500	2400	120	3	30000	6000	3	0.0489	0.1083 ± 0.0139	0.0045 ± 0.0026
11	SNP500	2400	120	3	30000	6000	2	0.056216	0.1424 ± 0.0487	(0.0020 ± 0.0017) × 10 ⁻¹

FIGURE 18 – Résultats des expériences ^a

a. (*) La deuxième coordonnée de la série correspond à la première coordonnée décalée de 30 minutes.

On remarque que les résultats donnés par la valeur de la perte d'entraînement et du score context-FID sont meilleurs pour des séries avec un nombre plus important de coordonnées, alors que le score de corrélation est meilleur dans le cas contraire. En effet, il est cohérent qu'il soit plus facile d'apprendre les corrélations croisées entre les coordonnées lorsqu'elles sont moins nombreuses. Par ailleurs, les résultats relatifs à la perte d'entraînement et au score context-FID sont également meilleurs lorsque la série est plus courte, alors que le score de corrélation est meilleur lorsque la série est plus longue, car le modèle peut alors apprendre davantage les corrélations existantes à plus long terme entre les différentes coordonnées. Par souci de simplicité, nous nous intéresserons aux résultats des expériences 9 et 11, afin de n'avoir que deux coordonnées à examiner.

Pour évaluer la qualité des données générées par rapport à leur satisfaction des faits stylisés, nous avons pensé les quatre différentes méthodes qui suivent, à partir des modèles déjà entraînés 9 et 11 (que nous distinguerons par la suite en nommant les données sur lesquels ils ont été entraînés : Bitcoin ou S&P500).

Remarque 4. Nous avons d'abord envisagé de travailler avec des séries univariées afin de générer des séries similaires. Cependant, le modèle considéré n'a pas réussi à accomplir cette tâche. Pour contourner cette limitation, nous avons opté pour une série à deux coordonnées, où la deuxième coordonnée est une version décalée de 30 minutes de la première, en utilisant des données sur le Bitcoin.

Cette approche présente plusieurs avantages : elle permet d'évaluer l'impact de la duplication des données sur les résultats et d'observer si le modèle parvient à mieux apprendre les relations temporelles lorsque les données sont répétées. En effet, en apprenant deux fois la même information, le modèle pourrait améliorer sa capacité d'apprentissage. De plus, cette configuration nous offre une représentation visuelle plus claire des relations entre les coordonnées des séries générées et surtout sur la qualité d'apprentissage des corrélations entre les coordonnées.

En parallèle, nous comparons les résultats obtenus avec ceux d'une série bivariée dont les deux coordonnées sont véritablement distinctes. Cette comparaison nous permet d'évaluer si l'approche de décalage offre des avantages notables en termes de performance et de compréhension des relations entre les coordonnées.

Remarque 5. Les deux coordonnées choisies du S&P500 sont les prix des produits Apple et Microsoft.

4.1 Les méthodes et leurs limites

4.1.1 Méthode 1 : Comparaison simple entre signal initial et signal généré

L'idée de l'expérience est de partir d'un signal réel \mathbf{x}_0 fixé, de le brouter selon le processus de diffusion $p_{T|0}(\mathbf{x}_T|\mathbf{x}_0)$, afin d'obtenir un signal bruité $\mathbf{x}_T = \sqrt{\alpha_T}\mathbf{x}_0 + \sqrt{1-\alpha_T}\varepsilon_T$, où T est le nombre total d'étapes de diffusion et ε_T est une réalisation de $\mathcal{N}(0, \mathbf{I}_d)$. Nous mettons ensuite ce signal bruité en entrée du modèle entraîné

afin d'obtenir un signal \hat{x}_0 , réalisation de l'estimateur de la loi inconnue p_0 via le modèle. L'idée est ensuite de comparer le signal obtenu \hat{x}_0 avec le signal initial fixé x_0 . En effet, nous aimerais que \hat{x}_0 ressemble au signal d'entrée x_0 à l'origine du bruit d'entrée du modèle.

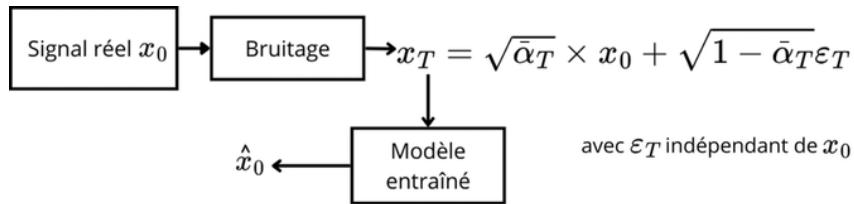


FIGURE 19 – Schéma explicatif de l'expérience 1

La principale *limite* de cette expérience repose sur le fait que le signal bruité après T étapes, noté x_T , est distribué selon une densité de probabilité gaussienne dont le bruit ajouté masque souvent le signal initial, rendant difficile la reconstruction précise par le modèle. x_T n'est donc pas associé de façon unique au signal x_0 , ce qui peut altérer la qualité de la comparaison.

4.1.2 Méthode 2 : Génération multiple et moyennage des signaux

Dans cette expérience, l'idée est de partir d'un signal x_0 fixé, comme précédemment. Cependant, ici, nous allons échantillonner 500 signaux bruités $\{x_T^i\}_{i=1}^{500}$ que nous allons mettre en entrée du modèle afin de générer 500 nouveaux signaux $\{\hat{x}_0^i\}_{i=1}^{500}$. Cela permettra de prendre en compte la variabilité existante lors de la génération de bruits à partir d'un unique x_0 . On reproduit donc l'expérience 1 de la figure 19, 500 fois en fixant un signal réel x_0 .

Nous allons ensuite moyenner les signaux générés pour chaque instant temporel, comme représenté sur la figure 20 et comparer le signal initial x_0 au signal moyen x_{moy} ainsi obtenu.

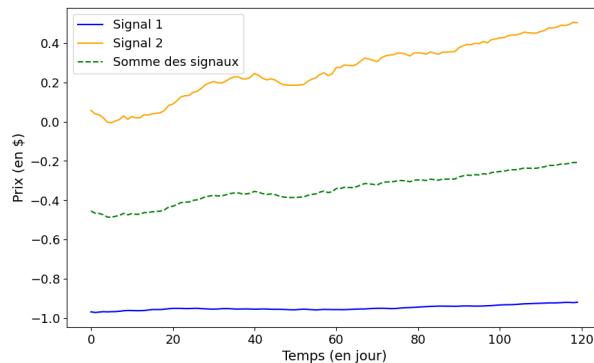


FIGURE 20 – Exemple de courbe obtenue en moyennant deux signaux générés à partir du S&P500

La principale limite de cette deuxième méthode provient du fait que la courbe moyenne obtenue est lissée et les caractéristiques importantes des séries comme les tendances, les saisonnalités et le bruit sont atténuées. Elle ne présente donc pas les caractéristiques que l'on souhaiterait faire apparaître des données.

4.1.3 Méthode 3 : Comparaison individuelle des signaux générés

Pour résoudre les problèmes liés aux effets de moyenne de l'expérience précédente, nous allons reprendre le même procédé que la partie 4.1.2, en générant plusieurs bruits $\{x_T^i\}_{i=1}^{500}$ à partir d'un unique signal x_0 et obtenir donc plusieurs images $\{\hat{x}_0^i\}_{i=1}^{500}$. Cependant, nous allons cette fois-ci comparer chacun des \hat{x}_0^i au signal réel initial x_0 , puis moyenner les résultats obtenus.

Les limites de cette expérience viennent cette fois-ci du fait que l'utilisation d'un signal \mathbf{x}_0 fixé ne permet pas de prendre en compte la diversité des signaux existants dans la base de données d'entraînement. Autre limite : le modèle ne permet pas de reconstituer un signal réel spécifique à partir d'un signal bruité, mais permet plutôt de générer un signal selon une estimation de la densité de probabilité de l'ensemble des données d'entraînement. En effet, le signal initial n'a pas une part importante dans le calcul du signal bruité comparé au bruit gaussien que l'on ajouté. Le signal est noyé dans le bruit, rendant difficile pour le modèle d'apprendre à remonter au signal d'origine.

4.1.4 Méthode 4 : Comparaison agrégée entre données réelles et simulées

La dernière expérience que nous avons réalisée consiste à comparer l'ensemble des données réelles $\{\mathbf{x}_0^i\}_{i=1}^R$ à l'ensemble des données simulées $\{\hat{\mathbf{x}}_0^i\}_{i=1}^S$ de manière agrégée, les signaux simulés étant générés à parti de bruits blancs gaussiens indépendants mis en entrée du modèle entraîné. Nous avons différentes valeurs pour les paramètres R des données issues du S&P500 et celles issues du Bitcoin, le nombre de données simulées S étant le même dans les deux cas $S = 3003$. Pour le S&P500, nous avons $R = 2281$ et pour le Bitcoin, $R = 2761$. Dans tous les cas, nous considérons des séquences bidimensionnelles de longueur 120.

L'idée est d'étudier la distribution des deux ensembles de données et de les comparer.

Les limites de cette expérience sont liées au fait que l'analyse agrégée peut masquer des variations spécifiques et des comportements particuliers des séries individuelles.

4.2 Les métriques utilisées

Pour évaluer la qualité des données générées à l'aide des modèles entraînés et selon les 4 méthodes présentées précédemment, nous allons utiliser plusieurs métriques, basées tout d'abord sur des critères qualitatifs visuels, des tests et des caractéristiques statistiques, ainsi que sur la mesure de certaines distances entre les séries originelles et les séries générées. Ces outils permettent d'analyser et de comparer la qualité des séries générées aux séries réelles selon différents aspects.

4.2.1 Critères visuels

Les critères visuels permettent d'évaluer de manière qualitative la similarité des séries temporelles réelles et générées. Voici les principaux éléments observés :

- **Évolution temporelle des prix et des rendements** : On compare directement les courbes de prix et les rendements calculés à partir des prix pour détecter visuellement des divergences ou similitudes entre les séries réelles et générées, que ce soit au niveau des tendances et des saisonsnalités ou des fluctuations.
- **ACF (Autocorrelation Function) (6.14.3)** : La fonction d'autocorrélation permet de vérifier la présence de dépendances dans le temps à différents décalages (lags) entre les valeurs d'une série temporelle, notamment pour détecter des motifs répétitifs ou des structures de tendance. On compare les ACF des séries générées avec celles des séries réelles (pour les prix et les rendements) pour s'assurer que le modèle capture correctement les dynamiques temporelles. Cela permet notamment de vérifier le fait stylisé d'absence d'autocorrelations des rendements.
- **QQ plot (Quantile-Quantile plot)** : Ce graphique a pour objectif de comparer la distribution des rendements calculés à partir des prix générés ou des rendements réels avec la distribution gaussienne. Pour ce faire, le QQ-plot représente les quantiles empiriques en ordonnées, en fonction des quantiles théoriques de la distribution normale en abscisses. Un alignement des quantiles empiriques avec la diagonale des quantiles théoriques indique que la distribution des données suit bien la distribution normale, tandis que tout écartement par rapport à cette diagonale suggère des déviations par rapport à la normalité, telles que des queues

épaisses, une asymétrie, ou d'autres anomalies dans la distribution des données. Cela permet de détecter et d'évaluer le fait stylisé de queues de distribution des rendements lourdes.

- **Volatilité calculée sur une fenêtre glissante** : En calculant la volatilité sur des fenêtres glissantes de temps définies, on visualise les périodes de haute et basse volatilité dans les rendements des séries générées, que l'on compare à celles des rendements réels afin de vérifier la présence de clusters de volatilité semblables.
- **Techniques de réduction de la dimensionnalité (6.14.4)** : Pour mieux visualiser la distribution des séries de données générées dans un espace de dimension réduite, des techniques comme l'analyse en composantes principales (PCA), le t-SNE (t-distributed Stochastic Neighbor Embedding) et la KDE (Kernel Density Estimation) sont utilisées. Ces outils permettent de représenter visuellement les séries générées dans un espace de plus faible dimension (deux dimensions, ici) tout en préservant certaines caractéristiques, facilitant ainsi une comparaison visuelle avec les séries réelles.

4.2.2 Tests statistiques

Les tests statistiques permettent d'évaluer quantitativement si les séries générées partagent certaines propriétés clés des séries temporelles réelles. Voici les principaux tests réalisés :

- **Tests de stationnarité** : La stationnarité est une propriété essentielle des séries temporelles financières. On applique le test de stationnarité de Dickey-Fuller augmenté (ADF), pour vérifier si les séries des rendements sont stationnaires, c'est-à-dire si leurs propriétés statistiques, telles que la moyenne et la variance, sont constantes dans le temps.
- **Tests de présence d'autocorrélations** : Des tests statistiques comme le test de Ljung-Box permettent de vérifier si les séries des rendements présentent des corrélations temporelles significatives à certains décalages (lags). Ces tests permettent de comparer la structure de dépendance des séries générées avec celle des séries réelles, et de vérifier la satisfaction du fait stylisé d'absence d'autocorrélation des rendements.

4.2.3 Coefficients et caractéristiques statistiques

Des coefficients et caractéristiques statistiques spécifiques sont également calculés pour comparer les propriétés des séries générées et réelles :

- **Kurtosis** : Ce coefficient mesure l'épaisseur des queues de distribution des rendements en comparant le coefficient calculé à celui de la distribution normale, qui vaut 3. La formule de la Kurtosis pour une série $X = \{X_t\}_{t=1}^N$ est $K = \frac{\frac{1}{N} \sum_{t=1}^N (X_t - \bar{X})^4}{(\frac{1}{N} \sum_{t=1}^N (X_t - \bar{X})^2)^2}$, avec $\bar{X} = \frac{1}{N} \sum_{t=1}^N X_t$ la moyenne empirique de la série.
- **Hurst (6.14.5)** : Le coefficient de Hurst est utilisé pour mesurer la tendance des séries à revenir à leur moyenne (anti-persistante) ou à poursuivre dans la même direction (persistante). En calculant ce coefficient pour les séries générées, on peut vérifier si le modèle capture correctement la persistance des séries réelles.

4.2.4 Distances

Enfin, certaines distances spécifiques sont calculées pour mesurer la divergence entre les séries temporelles générées et réelles. Ces distances permettent d'évaluer la qualité de la génération à travers des mesures quantitatives :

- **Distance L2** : Cette distance, aussi appelée norme Euclidienne, mesure la différence point par point entre deux séries temporelles. Elle est simple à calculer et offre une évaluation directe des écarts entre les séries générées et réelles pour chaque instant, mais ne prend pas compte de la dynamique temporelle des séries, comme les décalages dans le temps ou les changements d'échelle et de translation.
- **Divergence KL-Fourier (6.14.6)** : Cette mesure calcule la dissimilarité entre les distributions spectrales de puissance de deux séries temporelles. Elle se concentre sur les différences en termes de fréquence.

- **Wasserstein-Fourier (6.14.7)** : Cette distance proposée par [2] pour des séries temporelles stationnaires, est une adaptation de la distance de Wasserstein¹² aux représentations spectrales de puissance des séries temporelles. Elle permet de comparer des séries en fonction de leur contenu spectral plutôt qu'en fonction de leurs valeurs temporelles.

4.3 Les résultats

Nous allons exposer ici les résultats des quatre expériences réalisées. L'ensemble des graphiques et résultats se trouvent en annexe 6.15, mais nous allons reporter ici les principales remarques intéressantes.

4.3.1 Méthode 1 : Comparaison simple entre signal initial et signal générée

Dans cette expérience, nous avons tiré une série aléatoirement de l'ensemble d'entraînement, soit parmi les 2761 séries de Bitcoin, soit parmi les 2281 séries du S&P500. La série choisie a été bruitée via le processus de bruitage de Diffusion-TS, puis utilisée comme entrée du modèle entraîné pour générer une version estimée de la série initiale.

Cette première méthode produit des résultats nuancés, avec certains aspects stylisés financiers bien reproduits et d'autres moins bien capturés.

Dans un premier temps, le bruitage des signaux vers un bruit blanc gaussien confirme la bonne convergence théorique des signaux dans le cadre de la diffusion (voir figures 34, 44). Lors de la génération, la figure 21 montre que, pour le Bitcoin, le modèle capture efficacement le décalage temporel spécifique (30 min) entre les deux coordonnées de la série originelle en générant des signaux qui préservent cette relation temporelle ce qui démontre sa capacité à apprendre des structures spécifiques.

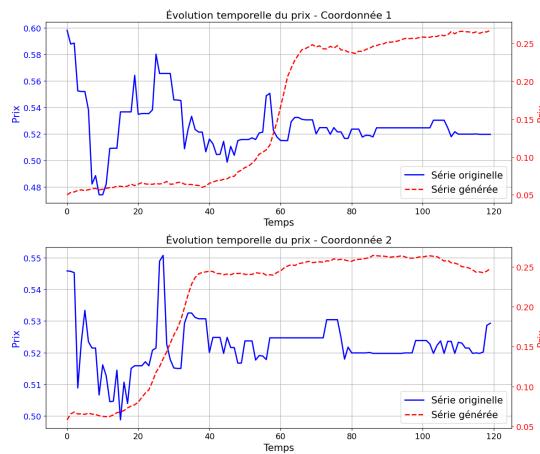


FIGURE 21 – Évolution temporelle des prix du Bitcoin

Cependant, la modélisation des autocorrélations des rendements montre des limites. Pour le Bitcoin, les séries générées présentent de nombreux lags significatifs bien que décroissants, ce qui contraste avec l'absence d'autocorrélation des rendements dans les séries financières réelles 22 (ou présence de très faibles autocorrélations.)

12. La distance de Wasserstein mesure le "coût minimum" de transport d'une distribution vers une autre.

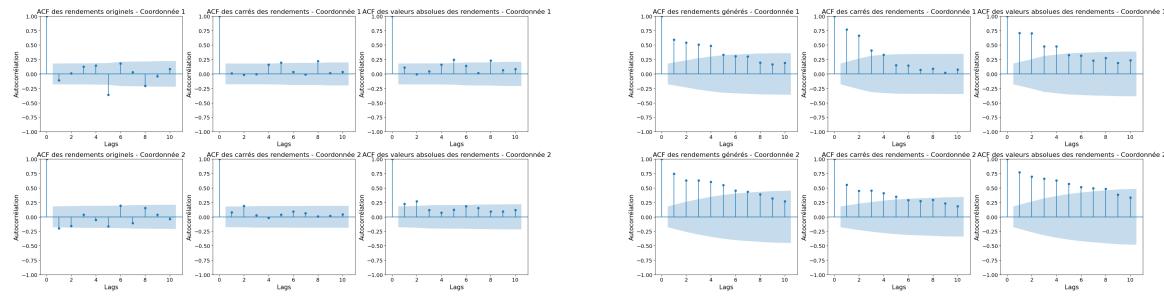


FIGURE 22 – ACF des rendements : à gauche, série originelle ; à droite, série générée

En revanche, pour le S&P500, l'ACF des rendements générés se rapproche davantage de celle des données réelles, signalant une meilleure adaptation à la dynamique temporelle de cet indice 23.

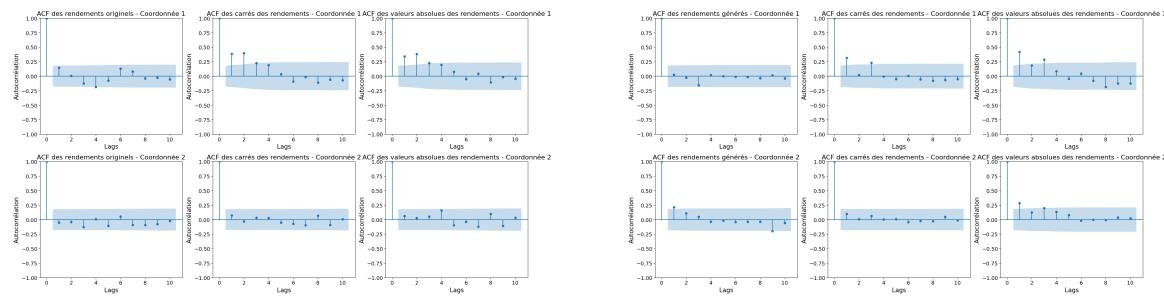


FIGURE 23 – ACF des rendements du S&P500 : à gauche, série originelle ; à droite, série générée

En termes de distribution, les rendements générés montrent des queues lourdes similaires aux séries réelles, confirmées par des kurtosis positives et des QQ-plots illustrant la non-normalité des rendements, en particulier pour le Bitcoin (39, 69). De plus, les coefficients de Hurst suggèrent une mémoire longue dans les séries générées, même plus marquée que dans les séries d'origine, traduisant une persistance accentuée dans les signaux générés.

Les clusters de volatilité, bien que présents, diffèrent légèrement en ampleur et durée par rapport aux séries originales. En revanche, les distances de Wasserstein-Fourier entre les densités spectrales sont faibles, de l'ordre de 10^{-3} , suggérant une bonne correspondance globale dans les signatures fréquentielles.

Enfin, les tests de Ljung-Box 43 montrent une autocorrélation significative pour les rendements générés du Bitcoin, en déviance avec le fait stylisé d'absence d'autocorrélation dans les rendements financiers, bien que ce fait ne soit pas universellement observé dans les séries réelles utilisées pour l'entraînement. Le S&P500 reproduit mieux ce fait, malgré des autocorrelations apparentes dans la coordonnée 1 des carrés des prix réels et la coordonnée 2 des rendements générés, marquant une approximation partielle de la dynamique réelle 52.

Ainsi, cette méthode permet de capturer certaines caractéristiques des données financières réelles tout en présentant des limites sur des aspects spécifiques, comme l'autocorrélation.

4.3.2 Méthode 2 : Génération multiple et moyennisation des signaux

Dans cette seconde méthode, nous avons généré 500 signaux à partir d'un signal réel unique. La moyenne de ces 500 signaux a ensuite été comparée au signal initial pour évaluer dans quelle mesure le modèle peut capturer les caractéristiques temporelles et distributionnelles du signal d'origine. Les résultats montrent plusieurs observations clés.

La génération de données suivie du calcul de la moyenne présente des résultats mitigés. D'un côté, pour le Bitcoin, certains aspects des propriétés stylisées des données financières sont préservés. Par exemple, les queues de distribution des rendements restent lourdes (kurtosis positive 61), ce qui est confirmé par des QQ plots similaires entre les données réelles et générées 60. De plus, la mémoire longue, vérifiée par les coefficients de Hurst, et

la présence de clusters de volatilité dans les séries générées rappellent certaines caractéristiques des données réelles, comme précisé dans les tableaux 61 et 62. Cependant, on observe un comportement inversé dans les résultats des tests statistiques liés à l'autocorrélation des coordonnées : la coordonnée 1 des séries générées est dépourvue d'autocorrélation, alors que la coordonnée 2 en présente — contrairement aux données réelles où la situation est inversée 63. De plus, la distance Wasserstein-Fourier entre les séries générées et réelles augmente considérablement (comparée aux résultats de la méthode précédente), indiquant une perte de précision dans la similarité fréquentielle par rapport aux approches précédentes.

Pour le S&P500, la moyennisation engendre une perte d'information plus marquée : la série moyennée perd toute fluctuation caractéristique de la série originelle et des données financières de manière plus générale, ne montrant plus qu'une tendance générale (résultat retrouvé aussi pour le Bitcoin, mais très marqué pour le S&P500), comme le montre la figure 24. Par ailleurs, les rendements générés deviennent presque nuls, ou tout du moins, bien inférieurs à ceux observés dans la série originelle.

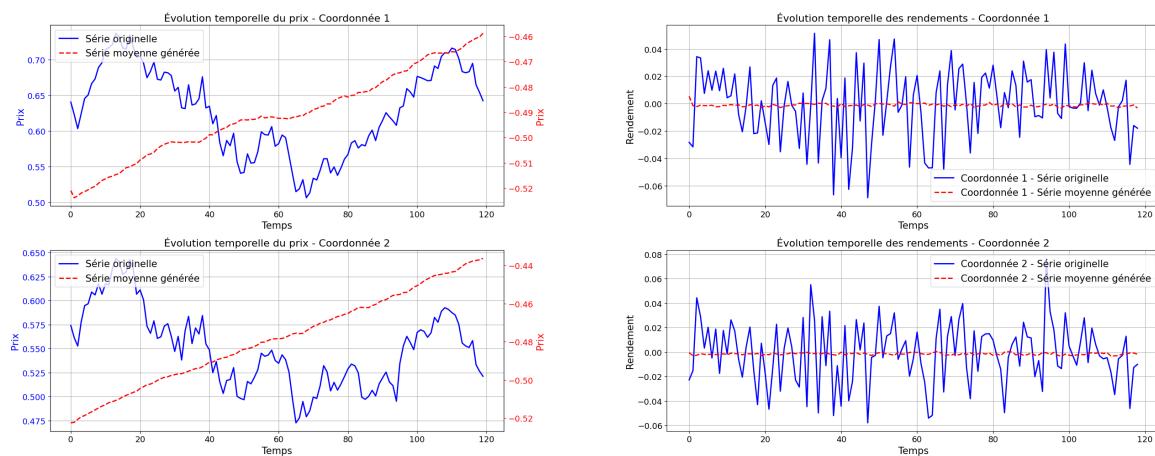


FIGURE 24 – À gauche : Évolution temporelle des prix du S&P 500 ; à droite : Évolution temporelle des rendements du S&P 500

Bien que l'autocorrélation des rendements simulés soit similaire à celle des rendements réels, comme le montre la figure 67, la faible amplitude des variations masque certains aspects structurels importants. En effet, les QQ plots 69 montrent des similarités et les queues de distribution ne sont pas lourdes d'après les valeurs des kurtosis 71. Cependant, les clusters de volatilité sont atténus et assez peu distinguables dans les données générées 72. Finalement, les distances Wasserstein-Fourier sont faibles (environ 10^{-3}), mais cette faible valeur semble davantage liée à la perte d'information dans la série moyennée qu'à une véritable similarité structurelle.

Dans l'ensemble, la moyennisation préserve certaines propriétés statistiques de surface tout en perdant en grande partie les dynamiques et fluctuations caractéristiques des séries financières réelles.

4.3.3 Méthode 3 : Comparaison individuelle des signaux générés

Dans la troisième méthode, qui consiste à comparer directement la série réelle fixée avec les 500 séries générées à partir de cette série bruitée, plusieurs observations permettent d'évaluer la performance du modèle en termes de proximité aux données originales. L'idée est de calculer la distance entre le signal réel et chacun des signaux générés et d'étudier la distribution de ces distances, ainsi que de comparer la valeur des coefficients de Hurst et Kurtosis de la série originelle à la distribution des signaux générés. Voici les tableaux récapitulatifs des résultats obtenus :

Pour le Bitcoin, la distance L2 entre les prix réels et générés est élevée, avec une majorité de valeurs entre 2 et 11, ce qui indique un écart important pouvant être expliqué par le fait que la distance L2 ne tient pas compte des dynamiques temporelles. En revanche, pour les rendements, la distance L2 est plus faible, avec des valeurs

Mesure	Série	Coordonnée	Moyenne	Ecart-type	Intervalle de confiance à 95%	Nombre d'outliers	Intervalle de confiance à 95% sans outliers
Distance L2	Prix	1	5.807	3.608	[−1.265, 12.880]	0	
		2	5.804	3.622	[−1.296, 12.904]	0	
	Rendements	1	0.297	0.401	[−0.487, 1.082]	72	[−0.487, 0.262]
		2	0.404	0.245	[−0.076, 0.883]	66	[−0.076, 0.385]
Divergence KL - Fourier	Prix	1	0.0001	0.004	[−0.008, 0.010]	79	[−0.0002, 0.0004]
		2	0.0019	0.0035	[−0.0049, 0.0088]	35	[−0.0007, 0.0034]
	Rendements	1	0.0081	0.0041	[0.0001, 0.0161]	0	
		2	0.010	0.003	[0.003, 0.016]	0	
Distance Wasserstein - Fourier	Prix	1	0.005	0.010	[−0.015, 0.025]	49	[−0.004, 0.010]
		2	0.005	0.009	[−0.013, 0.023]	46	[−0.004, 0.009]
	Rendements	1	0.004	0.023	[−0.042, 0.050]	0	
		2	0.002	0.008	[−0.013, 0.016]	0	

FIGURE 25 – Mesures - Bitcoin

Coefficient	Série	Coordonnée	Moyenne	Ecart-type	Nombre de coefficients en dessous du seuil caractéristique	Nombre de coefficients au-dessus du seuil caractéristique
Coefficient de Hurst (seuil = 0.5)	Prix	1	0.646	0.158	98	402
		2	0.628	0.176	129	371
	Rendements	1	0.702	0.214	91	409
		2	0.694	0.229	102	398
Kurtosis (seuil = 0)	Prix	1	0.127	2.887	339	161
		2	-0.103	2.022	364	136
	Rendements	1	9.105	12.939	10	442
		2	7.767	12.927	31	414

FIGURE 26 – Coefficients de Kurtosis et de Hurst - Bitcoin

comprises entre 0.05 et 0.43. La divergence KL-Fourier est faible pour les prix, se concentrant en dessous de 8×10^{-4} pour la coordonnée 1 et 4×10^{-3} pour la coordonnée 2. Pour les rendements, elle reste aussi relativement basse, entre 0.0025 et 0.0175. La distance de Wasserstein-Fourier pour les prix présente un pic autour de 0, avec des valeurs allant jusqu'à 0.014, tandis que pour les rendements, elle atteint des valeurs maximales de 0.2, bien que la plupart des valeurs soient inférieures à 0.02. Les coefficients de Hurst des prix et des rendements des séries générées incluent les valeurs observées dans la série réelle, ce qui indique une mémoire longue bien capturée. Cependant, les kurtosis des prix et des rendements des séries réelles se situent hors de la distribution de celles des séries générées, en particulier pour la coordonnée 1, comme le montrent les figures 82b et 82d données en annexe. Les coefficients de Hurst des prix et des rendements des séries générées incluent les valeurs observées dans la série réelle, ce qui indique une mémoire longue bien capturée. Cependant, les kurtosis des prix et des rendements des séries réelles se situent hors de la distribution des séries générées, surtout pour la coordonnée 1, comme le montre les annexes ?? et ??.

Pour le S&P500, la distance L2 des prix est encore plus marquée, avec des valeurs pouvant atteindre 18, tandis que pour les rendements, elle se concentre autour de 0.28 à 0.29 selon la coordonnée, illustrant un rapprochement aux rendements réels plus significatif. La divergence KL-Fourier pour les prix reste faible, entre 0 et 0.0005, mais pour les rendements, elle est plus élevée, de 0.005 à 0.02 (annexes 88a, 88b et 88c, 88d). Cette différence

Mesure	Série	Coordonnée	Moyenne	Ecart-type	Intervalle de confiance à 95%	Nombre d'outliers	Intervalle de confiance à 95% sans outliers
Distance L ₂	Prix	1	12.365	5.754	[1.087, 23.642]	0	
		2	11.500	5.152	[1.401, 21.598]	0	
	Rendements	1	0.417	0.431	[-0.426, 1.260]	57	[-0.426, 0.372]
		2	0.380	0.299	[-0.204, 0.965]	91	[-0.204, 0.306]
Divergence KL - Fourier	Prix	1	0.0018	0.0077	[-0.0132, 0.0168]	100	[-0.0006 × 10 ⁻¹ , 0.0004]
		2	0.0018	0.0067	[-0.0113, 0.0149]	93	[-0.0006 × 10 ⁻¹ , 0.0003]
	Rendements	1	0.012	0.003	[0.006, 0.017]	0	
		2	0.012	0.003	[0.007, 0.017]	0	
Distance Wasserstein - Fourier	Prix	1	0.0074	0.0036	[0.0003, 0.0145]	33	[0.0027, 0.011]
		2	0.0037	0.0032	[-0.0026, 0.0101]	131	[0.0017, 0.0032]
	Rendements	1	0.0043	0.0198	[-0.0344, 0.0431]	0	
		2	0.0023	0.0036	[-0.0049, 0.0094]	0	

FIGURE 27 – Mesures - S&P500

Coefficient	Série	Coordonnée	Moyenne	Ecart-type	Nombre de coefficients en dessous du seuil caractéristique	Nombre de coefficients au-dessus du seuil caractéristique
Coefficient de Hurst (seuil = 0.5)	Prix	1	0.706	0.209	78	414
		2	0.707	0.193	74	426
	Rendements	1	0.686	0.212	95	405
		2	0.602	0.205	167	333
Kurtosis (seuil = 0)	Prix	1	-0.887	0.613	454	46
		2	-0.871	0.577	462	38
	Rendements	1	2.970	4.576	34	400
		2	3.557	5.269	28	407

FIGURE 28 – Coefficients de Kurtosis et de Hurst - S&P500

pourrait s'expliquer par l'absence d'un critère d'apprentissage spécifique sur les rendements. Enfin, la distance de Wasserstein-Fourier des prix est modérée, entre 0.0015 et 0.008, tandis que pour les rendements, elle se concentre autour de 0, ne dépassant pas 0.02.

Toutefois, la distribution des distances montre une variabilité importante, avec des écarts-types souvent aussi élevés que les moyennes, ce qui suggère que la convergence vers le signal réel n'est pas toujours significative pour cette méthode, comme précisé dans les tableaux 25 et 27.

Ainsi, bien que certaines propriétés statistiques soient capturées (comme la mémoire longue et la faible divergence spectrale pour les prix), la méthode montre des limites dans la précision des valeurs individuelles, notamment pour les distributions de kurtosis, nécessitant possiblement des ajustements spécifiques dans la fonction de perte du modèle afin d'améliorer la qualité des séries simulées.

4.3.4 Méthode 4 : Comparaison agrégée entre données réelles et simulées

Dans cette quatrième méthode, la comparaison des distributions entre l'ensemble des séries réelles et celui des séries générées permet d'évaluer la qualité de la génération de données en termes de similarités globales des distributions.

Globalement, les rendements des séries générées sont du même ordre de grandeur que ceux des séries réelles, et la relation entre les coordonnées est bien capturée, comme en témoignent les graphiques annexes 91b et 99b

pour les rendements simulés et 90b et 98b pour les prix simulés. Toutefois, certaines séries générées montrent des autocorrelations des rendements plus marquées que celles des séries réelles, en particulier pour le Bitcoin, comme on peut le voir sur la figure 92b comparée aux ACF des rendements originels 92a.

Les analyses en PCA, t-SNE, et KDE montrent une superposition quasi parfaite entre les données d'entraînement et les séries générées, traduisant la capacité du modèle à capturer la diversité et la structure des données d'origine, comme le montrent les graphiques suivants pour le Bitcoin.

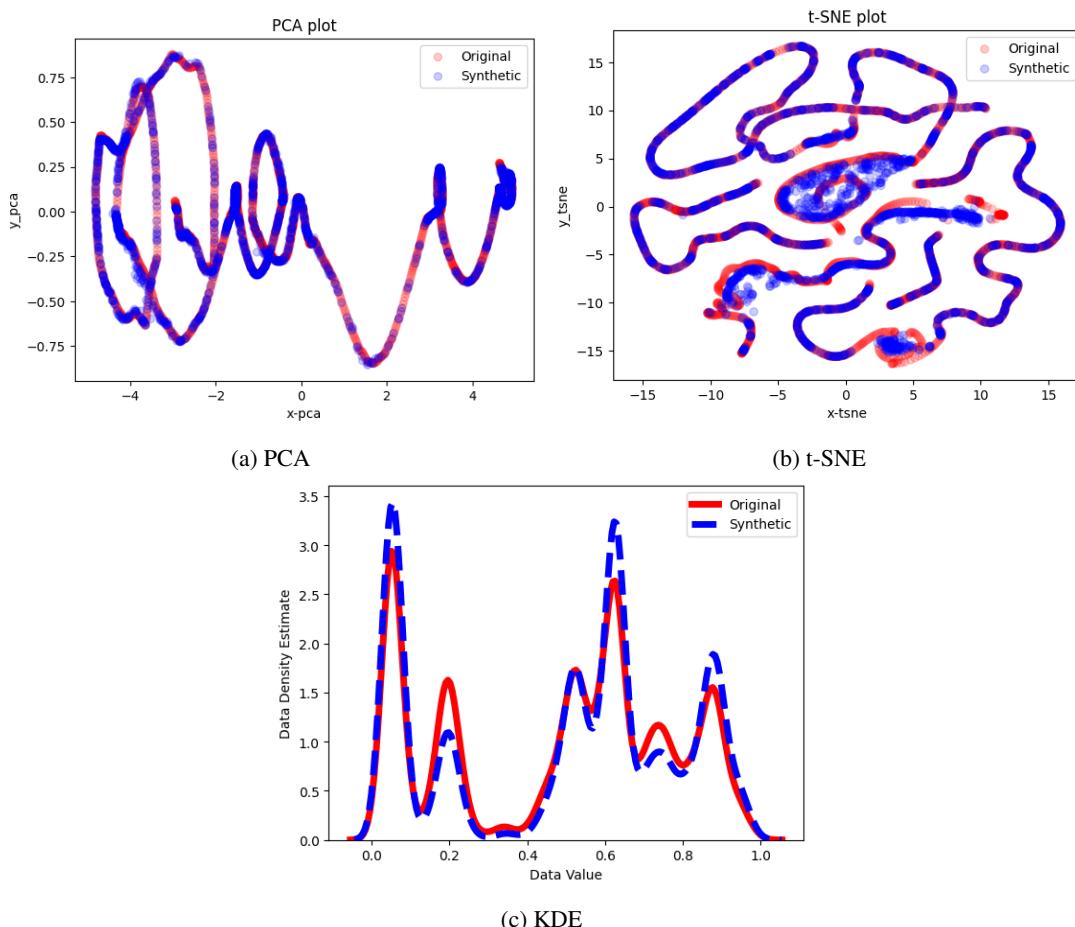


FIGURE 29 – Visualisations après réduction de dimension (Bitcoin)

Cependant, les distances entre les prix des séries sont plus faibles que celles entre leurs rendements (à l'exception de la distance L2, qui ne prend pas en compte les dynamiques temporelles et fréquentielles), comme affiché sur les tableaux annexes 95 et 102.

Les histogrammes des coefficients de Hurst et de la Kurtosis permettent d'évaluer les propriétés temporelles et la distribution des extrêmes. Pour le Bitcoin, les coefficients de Hurst des prix générés sont proches de ceux des séries réelles, variant entre 0.2 et 1, centrés autour de 0.6, bien que les données réelles montrent des valeurs élevées au-dessus de 1. Les rendements générés montrent une distribution plus large, entre 0 et 2, avec une concentration entre 0.7 et 1.7, alors que les réels se concentrent autour de 0.5. Concernant la kurtosis, les coefficients pour les prix réels sont entre 0 et 20, tandis que ceux des séries générées se concentrent principalement entre 0 et 10, avec plus de coefficients proches de 0.

Pour le S&P500, les prix montrent des pics similaires pour les coefficients de Hurst autour de 0.6 et 0.8 (réels) et 0.6 et 0.9 (générés). Cependant, les séries réelles ont plus de valeurs élevées (entre 1 et 1.2) que les simulées, qui présentent davantage de valeurs basses (sous 0.4). Pour les rendements, les réels se concentrent autour de 0.6 tandis que les générés sont plus centrés entre 0.75 et 1.25. Dans tous les cas la majeure partie des séries ont un

coefficient de Hurst supérieur à 0,5, ce qui est caractéristique d'une mémoire longue.

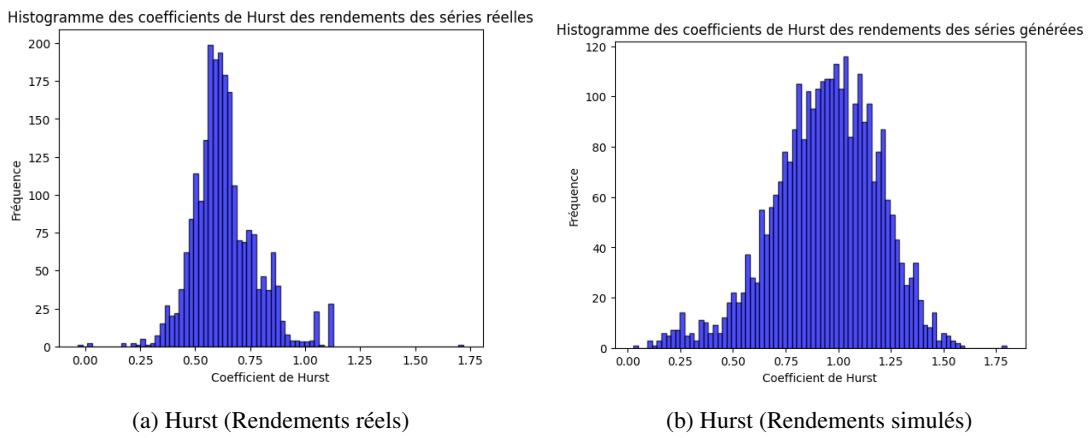
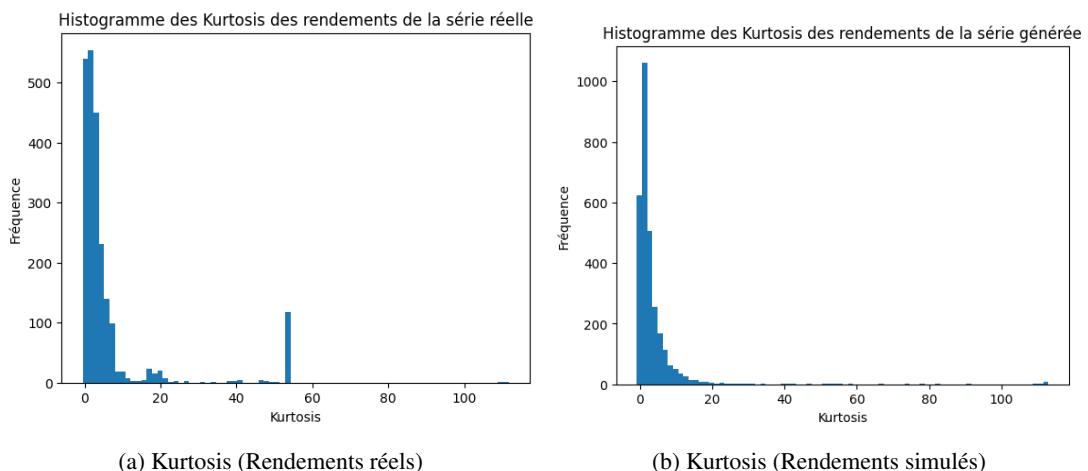


FIGURE 30 – Coefficients de Hurst

La kurtosis pour les prix réels et générés est concentrée autour de -1, bien que les réels présentent plus de valeurs extrêmes allant jusqu'à 4. Pour les rendements, un pic à 0 et une décroissance progressive sont observés pour les deux types de données, mais les réels montrent un pic supplémentaire au-delà de 50. Dans tous les cas, les kurtosis sont en très grande partie positives, ce qui est caractéristique des queues de distribution épaisses.



Ainsi, cette méthode confirme que le modèle reproduit bien certaines caractéristiques globales des séries temporelles, comme la relation entre les coordonnées et les principales propriétés de volatilité, bien qu'il peine parfois à répliquer les extrêmes observés dans les données réelles.

5 Conclusion et discussion

Ce stage a permis d'explorer l'application des modèles de diffusion pour générer des séries temporelles financières, avec un focus particulier sur la capacité de ces modèles à capter les spécificités de données financières. En raison des propriétés distinctives de ces séries, appelées faits stylisés (comme la volatilité en cluster ou les queues épaisses dans les distributions de rendements), il a été crucial de bien sélectionner les approches et métriques pour évaluer la qualité des séries simulées. Ces métriques, plus élaborées que les critères standards, permettent de mieux apprécier dans quelle mesure les séries générées reproduisent les caractéristiques essentielles des données réelles.

Les expérimentations ont montré le potentiel des modèles de diffusion, tout en soulignant plusieurs défis. Une des limites a notamment été la quantité de données traitées qui est restée faible, ce qui a pu restreindre la qualité des résultats obtenus. Le travail avec des signaux unidimensionnels a également soulevé des difficultés de modélisation, conduisant à l'utilisation de stratégies alternatives, comme le décalage temporel des données. Par ailleurs, le modèle actuel n'a pas intégré de scénarios de ruptures de marché, tels que les crises financières, qui sont pourtant des éléments structurels des séries financières. Le conditionnement des séries simulées, bien qu'il soit un axe important, n'a pu être abordé dans ce cadre.

Pour améliorer la qualité des séries générées, et en particulier pour réduire les autocorrélations dans les rendements simulés, un affinement du critère d'apprentissage pourrait s'avérer pertinent. D'autres approches, comme l'intégration de modèles continus en temps pour permettre un bruitage plus progressif, pourraient également offrir une meilleure adaptation aux dynamiques complexes des données financières. Le conditionnement, qui permettrait de contrôler les simulations en fonction de certaines caractéristiques de marché, constitue une autre perspective prometteuse. Enfin, des méthodes telles que la Transfusion pourraient être explorées pour générer simultanément des séries de prix et de rendements, renforçant ainsi la cohérence des données simulées.

Enfin, la question de la reconnaissance des données réelles et simulées reste un axe de réflexion majeur. La capacité de distinguer ces deux types de données, notamment pour évaluer leur réalisme, pourrait conduire à des ajustements des modèles, permettant ainsi d'affiner encore la pertinence des séries simulées pour des applications concrètes en finance.

Ces perspectives ouvrent des voies d'amélioration pour le développement de modèles de diffusion plus adaptés aux séries financières.

Références

- [1] V. BORTOLI. *Generative modeling*. 2023. URL : https://vdeborto.github.io/project/generative_modeling/.
- [2] Elsa CAZELLES, Arnaud ROBERT et Felipe TOBAR. “The Wasserstein-Fourier Distance for Stationary Time Series”. In : *arXiv preprint arXiv:1912.05509* (2020). URL : <https://arxiv.org/abs/1912.05509>.
- [3] C. R. da CUNHA et R. da SILVA. “Relevant Stylized Facts About Bitcoin : Fluctuations, First Return Probability, and Natural Phenomena”. In : *Physica A : Statistical Mechanics and its Applications* 550 (2020), p. 124155. DOI : [10.1016/j.physa.2020.124155](https://doi.org/10.1016/j.physa.2020.124155). eprint : [arXiv.org](https://arxiv.org). URL : <https://doi.org/10.1016/j.physa.2020.124155>.
- [4] Abhyuday DESAI et al. “TimeVAE : A Variational Auto-Encoder for Multivariate Time Series Generation”. In : *arXiv arXiv:2111.08095* (2021). arXiv.org. URL : <https://doi.org/10.48550/arXiv.2111.08095>.
- [5] Bradley EFRON. “Tweedie’s formula and selection bias”. In : *Journal of the American Statistical Association* 106.496 (2011), p. 1602-1614. URL : <https://efron.ckirby.su.domains/papers/2011TweediesFormula.pdf>.
- [6] Jonathan HO, Ajay JAIN et Pieter ABBEEL. “Denoising Diffusion Probabilistic Models”. In : *arXiv preprint arXiv:2006.11239* (2020).
- [7] Alexander NIKITIN et AL. “TSGM : A Flexible Framework for Generative Modeling of Synthetic Time Series”. In : *arXiv* (2024). eprint : [2305.11567](https://arxiv.org/abs/2305.11567). URL : <https://doi.org/10.48550/arXiv.2305.11567>.
- [8] Md Fahim SIKDER et AL. “TransFusion : Generating Long, High Fidelity Time Series using Diffusion Models with Transformers”. In : *arXiv* (2024). eprint : [2307.12667](https://arxiv.org/abs/2307.12667). URL : <https://doi.org/10.48550/arXiv.2307.12667>.
- [9] Yang SONG et Stefano ERMON. “Generative Modeling by Estimating Gradients of the Data Distribution”. In : *arXiv preprint arXiv:1907.05600* (2020). arXiv:1907.05600. URL : <https://doi.org/10.48550/arXiv.1907.05600>.
- [10] Jinsung YOON et al. “Time-series Generative Adversarial Networks”. In : *Advances in Neural Information Processing Systems*. T. 32. Neural Information Processing Systems. Curran Associates, Inc., 2019. URL : https://papers.nips.cc/paper_files/paper/2019/hash/c9efe5f26cd17ba6216bbe2a7d26d490Abstract.html.
- [11] Xinyu YUAN et Yan QIAO. “Diffusion-TS : Interpretable Diffusion for General Time Series Generation”. In : *arXiv* (2024). eprint : [2403.01742](https://arxiv.org/abs/2403.01742). URL : <https://doi.org/10.48550/arXiv.2403.01742>.

6 Annexes

Mettre faits stylisés, papiers concurrents déjà publiés, technique Montanari, critères d'évaluation, Workflow du processus (= plan de développement allant du nettoyage des données, à l'entraînement et génération... = succession des étapes...) et code avec le pré-traitement des données pour avoir les données d'entraînement (imputation pour le Bitcoin et sélection des colonnes + décalage pour le Bitcoin).

6.1 Formule de Tweedie

Soient X, Y et Z trois variables aléatoires à valeurs dans \mathbb{R}^d telles que :

$$Y = X + Z$$

où $Z \sim \mathcal{N}(0, \sigma^2)$, X et Z indépendantes et X de loi inconnue.

Propriété 1. (*Formule de Tweedie*) Soit $y \in \mathbb{R}^d$, une réalisation de Y , on a

$$\mathbb{E}[X|Y=y] = y + \sigma^2 \frac{d}{dy} \log p_Y(y)$$

où p_Y est la densité de probabilité de Y .

Démonstration. Sachant que $Y|X=x \sim \mathcal{N}(x, \sigma^2)$, pour $x \in \mathbb{R}^d$, on a pour $y \in \mathbb{R}^d$:

$$p_{Y|X}(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y-x)^2}{2\sigma^2}\right].$$

D'après la formule des probabilités totales :

$$p_Y(y) = \int_{-\infty}^{+\infty} p_{Y|X}(y|x)p_X(x)dx \quad (11)$$

D'après la formule de Bayes, la densité a posteriori est donnée par :

$$p_{X|Y}(x|y) = \frac{p_{Y|X}(y|x)p_X(x)}{p_Y(y)} = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(y-x)^2}{2\sigma^2}\right] \frac{p_X(x)}{p_Y(y)} \quad (12)$$

Par ailleurs, on a :

$$\frac{d}{dy} p_Y(y) = \int_{-\infty}^{+\infty} \frac{d}{dy} p_{Y|X}(y|x)p_X(x)dx = \int_{-\infty}^{+\infty} -\frac{(y-x)}{\sigma^2} p_{Y|X}(y|x)p_X(x)dx$$

en supposant que toutes les hypothèses du théorème de continuité sous l'intégrale soient satisfaites.

Ainsi, en appliquant les propriétés de dérivation du logarithme, on a :

$$\frac{d}{dy} \log p_Y(y) = \frac{1}{p_Y(y)} \frac{d}{dy} p_Y(y) = -\frac{y}{\sigma^2} \frac{1}{p_Y(y)} \int_{-\infty}^{+\infty} p_{Y|X}(y|x)p_X(x)dx + \frac{1}{\sigma^2} \int_{-\infty}^{+\infty} x p_{Y|X}(y|x) \frac{p_X(x)}{p_Y(y)} dx$$

En reconnaissant 11 et 12 dans l'expression précédente, on a finalement :

$$\frac{d}{dy} \log p_Y(y) = -\frac{y}{\sigma^2} + \frac{1}{\sigma^2} \int_{-\infty}^{+\infty} x p_{X|Y}(x|y) dx = \frac{1}{\sigma^2} [\mathbb{E}(X|Y=y) - y]$$

d'où :

$$\mathbb{E}(X|Y=y) = y + \sigma^2 \frac{d}{dy} \log p_Y(y) \quad (13)$$

□

En appliquant la formule de Tweedie à $X = \mathbf{x}_0 = \mathbf{x}_0$ et $Y = \mathbf{x}_t = \mathbf{x}_t$, on retrouve le résultat énoncé dans le corps du rapport à savoir :

$$\mathbb{E}[\mathbf{x}_0 | \mathbf{x}_t] = \mathbf{x}_t + \sigma_t^2 \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t).$$

6.2 Problème d'optimisation

En utilisant la formule des probabilités totales et des probabilités conditionnelles, nous avons pour tout $t \in \{1, \dots, T\}$:

$$\nabla_{\mathbf{x}_t} p_t(\mathbf{x}_t) = \int_{\mathbb{R}^d} \nabla_{\mathbf{x}_t} p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) p_0(\mathbf{x}_0) d\mathbf{x}_0$$

Par la suite, en utilisant les propriétés des logarithmes et la règle de Bayes, nous avons :

$$\begin{aligned} \nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) &= \frac{\nabla_{\mathbf{x}_t} p_t(\mathbf{x}_t)}{p_t(\mathbf{x}_t)} = \int_{\mathbb{R}^d} \nabla_{\mathbf{x}_t} p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) \frac{p_0(\mathbf{x}_0)}{p_t(\mathbf{x}_t)} d\mathbf{x}_0 \\ &= \int_{\mathbb{R}^d} \nabla_{\mathbf{x}_t} \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) \frac{p_0(\mathbf{x}_0)}{p_t(\mathbf{x}_t)} d\mathbf{x}_0 \\ &= \int_{\mathbb{R}^d} \nabla_{\mathbf{x}_t} \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) p_{0|t}(\mathbf{x}_0 | \mathbf{x}_t) \end{aligned}$$

Cette dernière équation peut être réécrite comme une espérance (voir annexe 6.3) :

$$\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) = \mathbb{E}_{\mathbf{x}_0 \sim p_{0|t}(\mathbf{x}_0 | \mathbf{x}_t)} [\nabla_{\mathbf{x}_t} \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0)] = \mathbb{E} [\nabla_{\mathbf{x}_t} \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) | \mathbf{x}_t]$$

En utilisant les propriétés de l'espérance conditionnelle introduites en annexe 6.4, nous obtenons :

$$\nabla_{\mathbf{x}_t} \log p_t(\mathbf{x}_t) = \operatorname{argmin}\{\mathbb{E}_{p_{0,t}} \|\mathbf{s}_\theta(\mathbf{x}_t, t) - \nabla_{\mathbf{x}_t} \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0)\|^2, f \in L^2(p_t)\} \quad (14)$$

où $p_{0,t}$ est la densité conjointe de \mathbf{x}_0 et \mathbf{x}_t , $t \in \{1, \dots, T\}$.

6.3 Preuve de l'égalité des espérances conditionnelles

Nous cherchons à montrer que :

$$\mathbb{E}_{\mathbf{x}_0 \sim p_{0|t}(\mathbf{x}_0 | \mathbf{x}_t)} [\nabla_{\mathbf{x}_t} \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0)] = \mathbb{E} [\nabla_{\mathbf{x}_t} \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) | \mathbf{x}_t]$$

Soient X et Y deux variables aléatoires à valeurs dans \mathbb{R}^d . Soit $x \in \mathbb{R}^d$. On a

$$\mathbb{E}[Y | X = x] = \int y p(y|x) dy$$

Si on prend, pour $t \in [0, T]$, $Y = \nabla_x \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0)$ et $X = \mathbf{x}_t$, on a alors :

$$\begin{aligned} \mathbb{E} [\nabla_x \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) | \mathbf{x}_t] &= \int \nabla_x \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) p_{0|t}(\mathbf{x}_0 | \mathbf{x}_t) d\mathbf{x}_0 \\ &= \mathbb{E}_{\mathbf{x}_0 \sim p_{0|t}} [\nabla_x \log p_{t|0}(\mathbf{x}_t | \mathbf{x}_0)] \end{aligned}$$

d'où le résultat.

6.4 Quelques propriétés de l'espérance conditionnelle

Soient X, Y et U trois variables aléatoires, \mathcal{F} une tribu. On note $\sigma(X)$ la tribu engendrée par la variable aléatoire X . On a les propriétés suivantes :

- $Y = \mathbb{E}[X|\mathcal{F}]$ si et seulement si $Y = \operatorname{argmin} \left\{ \mathbb{E} \|X - Z\|^2, Z \in L^2(\mathcal{F}) \right\}$.
- $Y \in \sigma(X)$ si et seulement si il existe $f : \mathbb{R}^d \mapsto \mathbb{R}^d$ mesurable telle que $Y = f(X)$.
- $Y = \mathbb{E}[X|U]$ si $Y = f(U)$ avec $f = \operatorname{argmin} \left\{ \mathbb{E} \|X - f(U)\|^2, f \in L^2(U) \right\}$.

6.5 Calcul du noyau de transition pour l'ajout de bruit en une étape du modèle DDPM

En partant du processus d'ajout de bruit gaussien, pour tout $t \in \llbracket 1, T \rrbracket$:

$$\mathbf{x}_t = \sqrt{\alpha_t} \mathbf{x}_{t-1} + \sqrt{1 - \alpha_t} z_t, \quad z_t \sim \mathcal{N}(0, \mathbb{I})$$

nous avons en utilisant la relation de récurrence :

$$\begin{aligned} \mathbf{x}_t &= \sqrt{\alpha_t} \mathbf{x}_{t-1} + \sqrt{1 - \alpha_t} z_t \\ &= \sqrt{\alpha_t} \left(\sqrt{\alpha_{t-1}} \mathbf{x}_{t-2} + \sqrt{1 - \alpha_{t-1}} z_{t-1} \right) + \sqrt{1 - \alpha_t} z_t \\ &= \sqrt{\alpha_t} \sqrt{\alpha_{t-1}} \mathbf{x}_{t-2} + \underbrace{\sqrt{\alpha_t} \sqrt{1 - \alpha_{t-1}} z_{t-1}}_{(1)} + \underbrace{\sqrt{1 - \alpha_t} z_t}_{(2)} \end{aligned}$$

z_t et z_{t-1} sont des bruits standards gaussiens indépendants. Le terme (1) est donc un bruit gaussien centré de variance $\alpha_t - \alpha_t \alpha_{t-1}$ et le terme (2) un bruit gaussien centré de variance $1 - \alpha_t$. On peut regrouper ces deux termes sous la forme d'un seul bruit gaussien centré de variance égale à la somme des deux variances par indépendance de z_t et z_{t-1} .

$$\begin{aligned} \mathbf{x}_t &= \sqrt{\alpha_t} \sqrt{\alpha_{t-1}} \mathbf{x}_{t-2} + \sqrt{1 - \alpha_t + \alpha_t - \alpha_t \alpha_{t-1} \varepsilon} \\ &= \sqrt{\alpha_t} \sqrt{\alpha_{t-1}} \left(\sqrt{\alpha_{t-2}} \mathbf{x}_{t-3} + \sqrt{1 - \alpha_{t-2}} z_{t-2} \right) + \sqrt{1 - \alpha_t \alpha_{t-1}} \varepsilon \\ &= \sqrt{\alpha_t \alpha_{t-1} \alpha_{t-2}} \mathbf{x}_{t-3} + \sqrt{\alpha_t \alpha_{t-1}} \sqrt{1 - \alpha_{t-2}} z_{t-2} + \sqrt{1 - \alpha_t \alpha_{t-1}} \varepsilon \\ &= \sqrt{\alpha_t \alpha_{t-1} \alpha_{t-2}} \mathbf{x}_{t-3} + \sqrt{1 - \alpha_t \alpha_{t-1} \alpha_{t-2}} \varepsilon \end{aligned}$$

avec $\varepsilon \sim \mathcal{N}(0, \mathbb{I}_d)$ en suivant le même raisonnement que précédemment.

On retrouve donc par récurrence :

$$\mathbf{x}_t = \sqrt{\alpha_t \alpha_{t-1} \alpha_{t-2} \dots \alpha_1} \mathbf{x}_0 + \sqrt{1 - \alpha_t \alpha_{t-1} \alpha_{t-2} \dots \alpha_1} \varepsilon, \quad \varepsilon \sim \mathcal{N}(0, \mathbb{I}_d)$$

Ce qui correspond au résultat souhaité.

Remarque 6. Par ailleurs, dans l'article [11] à partir duquel on travaille, on paramètre les valeurs des $(\beta_t)_t$ de sorte à ce que $\prod_{s=1}^t \alpha_s \rightarrow 0$ lorsque $t \rightarrow T$, avec $\alpha_t = 1 - \beta_t$. Ainsi, le signal bruité $\mathbf{x}_t \rightarrow \epsilon$ lorsque $t \rightarrow T$, où ϵ est une réalisation de $\mathcal{N}(0, \mathbb{I}_d)$. On retrouve donc l'hypothèse de convergence du signal bruité à l'étape T vers un bruit blanc gaussien.

6.6 Chaîne de Markov retournée

Démontrons que le processus inverse d'une chaîne de Markov est une chaîne de Markov.

Démonstration. Soit $\{X_t\}_{t \in \mathbb{Z}}$ une chaîne de Markov prenant des valeurs dans un espace d'état S , avec une distribution initiale p_0 et une matrice de transition P . La propriété de Markov implique que :

$$P(X_{t+1} = x_{t+1} \mid X_0 = x_0, X_1 = x_1, \dots, X_t = x_t) = P(X_{t+1} = x_{t+1} \mid X_t = x_t).$$

Autrement dit, la probabilité de l'état futur dépend uniquement de l'état actuel et non des états précédents.

Nous voulons montrer que le processus inverse $\{X_{T-t}\}_{t \in \mathbb{Z}}$, noté $Y_t = X_{T-t}$, est également une chaîne de Markov.

Le processus direct est défini par les probabilités de transition suivantes :

$$P(X_{t+1} = x_{t+1} \mid X_t = x_t).$$

Pour le processus inverse, les probabilités de transition s'écrivent comme suit :

$$P(Y_{t+1} = y_{t+1} \mid Y_t = y_t).$$

En remplaçant $Y_t = X_{T-t}$, cela devient :

$$P(X_{T-t-1} = y_{t+1} \mid X_{T-t} = y_t).$$

Nous cherchons donc à déterminer la transition de X_{T-t} à X_{T-t-1} , ce qui revient à la transition inverse. Pour obtenir les probabilités de transition inverses, utilisons la règle de Bayes :

$$P(X_{T-t-1} = y_{t+1} \mid X_{T-t} = y_t) = \frac{P(X_{T-t} = y_t \mid X_{T-t-1} = y_{t+1})P(X_{T-t-1} = y_{t+1})}{P(X_{T-t} = y_t)}.$$

La première partie, $P(X_{T-t} = y_t \mid X_{T-t-1} = y_{t+1})$, correspond à la probabilité directe de la chaîne de Markov, et la deuxième partie, $P(X_{T-t-1} = y_{t+1})$, est la probabilité marginale pour l'état y_{t+1} .

La chaîne de Markov inverse, décrite par les probabilités de transition :

$$P(Y_{t+1} = y_{t+1} \mid Y_t = y_t),$$

dépend uniquement de l'état actuel Y_t (qui est X_{T-t}) et non des états précédents Y_{t-1}, Y_{t-2}, \dots . Cela montre que le processus inverse vérifie également la propriété de Markov.

Ainsi, le processus inverse d'une chaîne de Markov est lui-même une chaîne de Markov, car ses probabilités de transition dépendent uniquement de l'état actuel et non des états passés, ce qui satisfait la propriété de Markov.

□

6.7 Borne supérieure de la log-vraisemblance

On cherche à montrer que l'inverse du logarithme de la vraisemblance est maximisée par l'évidence (Evidence Lower Bound, ELBO), notée L . On veut donc montrer :

$$-\log p_\theta(\mathbf{x}_0) \leq \mathbb{E}_q \left[-\log \left[\frac{p_\theta(\mathbf{x}_{0:T})}{p_{1:T|0}(\mathbf{x}_{1:T} \mid \mathbf{x}_0)} \right] \right] := L$$

Démonstration. D'après la formule des probabilités totales suivie de l'application de la règle de Bayes, nous

avons :

$$\begin{aligned}\log p_\theta(\mathbf{x}_0) &= \log \int p_\theta(\mathbf{x}_{0:T}) d\mathbf{x}_{1:T} \\ &= \log \int \frac{p_\theta(\mathbf{x}_{0:T})}{p_{1:T|0}(\mathbf{x}_{1:T}|\mathbf{x}_0)} p_{1:T|0}(\mathbf{x}_{1:T}|\mathbf{x}_0) d\mathbf{x}_{1:T} \\ &= \log \mathbb{E}_{p_{1:T|0}(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\frac{p_\theta(\mathbf{x}_{0:T})}{p_{1:T|0}(\mathbf{x}_{1:T}|\mathbf{x}_0)} \right]\end{aligned}$$

Par l'inégalité de Jensen, appliquée à la fonction logarithmique qui est concave nous avons :

$$\log p_\theta(\mathbf{x}_0) \geq \mathbb{E}_{p_{1:T|0}(\mathbf{x}_{1:T}|\mathbf{x}_0)} \left[\log \frac{p_\theta(\mathbf{x}_{0:T})}{p_{1:T|0}(\mathbf{x}_{1:T}|\mathbf{x}_0)} \right]$$

On retrouve le résultat souhaité en prenant l'inverse de l'inégalité précédente. \square

6.8 Loi de $p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$

En reprenant les mêmes notations que dans le corps du rapport, on cherche à démontrer que $p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$ est une distribution normale $\mathcal{N}(\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0), \tilde{\beta}_t \mathbf{I}_d)$ avec

$$\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \mathbf{x}_0 + \frac{\sqrt{\alpha_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_t \quad \text{et} \quad \tilde{\beta}_t = \frac{1 - \bar{\alpha}_{t-1}}{1 - \bar{\alpha}_t} \beta_t.$$

Démonstration. D'après la règle de Bayes, on a :

$$p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) = p_{t|t-1,0}(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{x}_0) \frac{p_{t|0}(\mathbf{x}_t | \mathbf{x}_0)}{p_{t-1|0}(\mathbf{x}_{t-1} | \mathbf{x}_0)}$$

Or sachant que les distributions sont markoviennes, nous avons :

$$p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0) = \frac{p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) p_{t-1|0}(\mathbf{x}_{t-1} | \mathbf{x}_0)}{p_{t|t-1}(\mathbf{x}_t | \mathbf{x}_{t-1})}.$$

D'après les hypothèses faites sur les distributions directe et indirecte, nous avons :

- $p_{t|0}(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\bar{\alpha}_t \mathbf{x}_0, (1 - \bar{\alpha}_t) I_d)$
- $p_{t|t-1}(\mathbf{x}_t | \mathbf{x}_{t-1}) = \mathcal{N}(\alpha_t \mathbf{x}_{t-1}, \beta_t I_d)$
- $p_{t-1|0}(\mathbf{x}_{t-1} | \mathbf{x}_0) = \mathcal{N}(\bar{\alpha}_{t-1} \mathbf{x}_0, (1 - \bar{\alpha}_{t-1}) I_d)$

En remarquant que $p_{t-1|t,0}(\mathbf{x}_{t-1} | \mathbf{x}_t, \mathbf{x}_0)$ s'exprime comme un produit de gaussiennes, il est possible d'exprimer cette distribution conditionnelle sous forme gaussienne également, en utilisant les formules de combinaison de distributions normales et les propriétés de conditionnement dans le cadre de modèles de diffusion gaussiens (non développé dans ce rapport). \square

6.9 Divergence KL entre deux gaussiennes

La divergence KL entre deux Gaussiennes univariées $p = \mathcal{N}(\mu_p, \sigma_p^2)$ et $q = \mathcal{N}(\mu_q, \sigma_q^2)$, s'exprime de cette manière :

$$D_{\text{KL}}(p \| q) = \frac{1}{2} \left(\frac{\sigma_p^2}{\sigma_q^2} + \frac{(\mu_q - \mu_p)^2}{\sigma_q^2} - 1 + \ln \frac{\sigma_q^2}{\sigma_p^2} \right).$$

6.10 Calcul de $\tilde{\mu}$

En repartant du processus d'ajout de bruit, on sait que, pour $t \in \llbracket 1, T \rrbracket$:

$$\mathbf{x}_t \sim q(\mathbf{x}_t | \mathbf{x}_0) = \mathcal{N}(\sqrt{\bar{\alpha}_t} \mathbf{x}_0, (1 - \bar{\alpha}_t) \mathbf{I}_d)$$

d'où

$$\mathbf{x}_t(\mathbf{x}_0, \varepsilon_t) = \sqrt{\bar{\alpha}_t} \mathbf{x}_0 + \sqrt{1 - \bar{\alpha}_t} \varepsilon_t$$

avec $\varepsilon_t \sim \mathcal{N}(0, \mathbf{I}_d)$.

On réécrit cette dernière égalité comme :

$$\mathbf{x}_0 = \frac{1}{\sqrt{\bar{\alpha}_t}} (\mathbf{x}_t - \sqrt{1 - \bar{\alpha}_t} \varepsilon_t).$$

Par ailleurs, sachant que

$$\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0) = \frac{\sqrt{\bar{\alpha}_{t-1}} \beta_t}{1 - \bar{\alpha}_t} \mathbf{x}_0 + \frac{\sqrt{\bar{\alpha}_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_t$$

on peut réécrire en fonction de \mathbf{x}_t , pour obtenir :

$$\begin{aligned} \tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0) &= \frac{\sqrt{\bar{\alpha}_{t-1}}}{1 - \bar{\alpha}_t} \frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t}} \mathbf{x}_t - \frac{\sqrt{\bar{\alpha}_{t-1}}}{1 - \bar{\alpha}_t} \frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t}} \sqrt{1 - \bar{\alpha}_t} \varepsilon_t + \frac{\sqrt{\bar{\alpha}_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \mathbf{x}_t \\ &= \left(\frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t} (1 - \bar{\alpha}_t)} + \frac{\sqrt{\bar{\alpha}_t} (1 - \bar{\alpha}_{t-1})}{1 - \bar{\alpha}_t} \right) \mathbf{x}_t - \frac{1 - \alpha_t}{1 - \bar{\alpha}_t} \sqrt{\frac{1 - \bar{\alpha}_t}{\alpha_t}} \varepsilon_t \\ &= \left(\frac{1 - \alpha_t}{\sqrt{\bar{\alpha}_t} (1 - \bar{\alpha}_t)} + \frac{\alpha_t (1 - \bar{\alpha}_{t-1})}{\sqrt{\bar{\alpha}_t} (1 - \bar{\alpha}_t)} \right) \mathbf{x}_t - \frac{1}{\sqrt{\bar{\alpha}_t}} \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \varepsilon_t \end{aligned}$$

Après simplification, on obtient bien :

$$\tilde{\mu}(\mathbf{x}_t, \mathbf{x}_0) = \frac{1}{\sqrt{\bar{\alpha}_t}} \left(\mathbf{x}_t - \frac{1 - \alpha_t}{\sqrt{1 - \bar{\alpha}_t}} \varepsilon_t \right)$$

avec $\varepsilon_t \sim \mathcal{N}(0, \mathbf{I}_d)$.

6.11 Mécanisme de Diffusion-TS

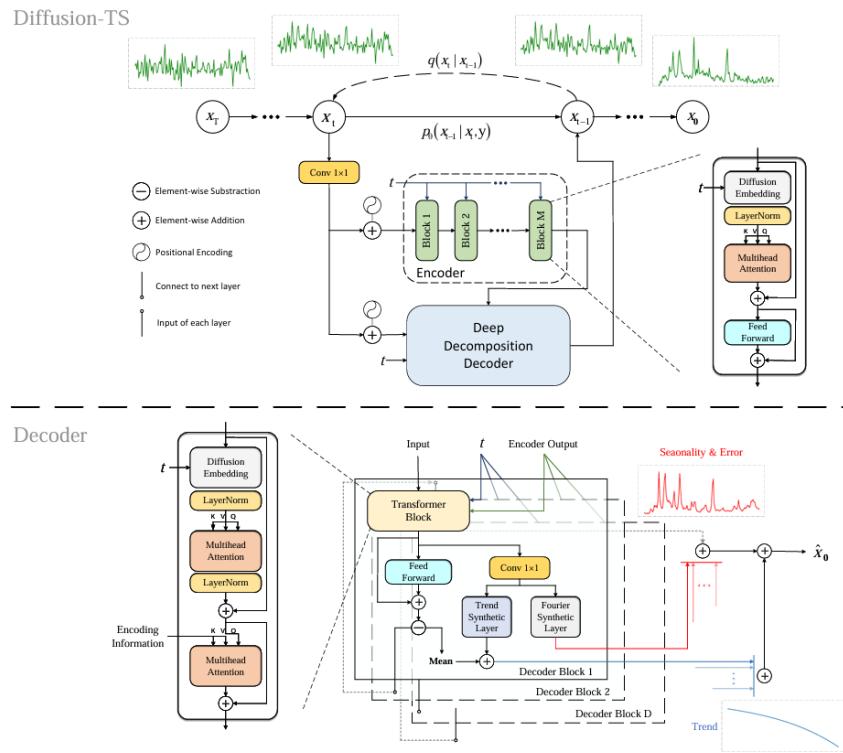


FIGURE 32 – Architecture du modèle de Diffusion-TS

6.12 Architecture du code de Diffusion-TS

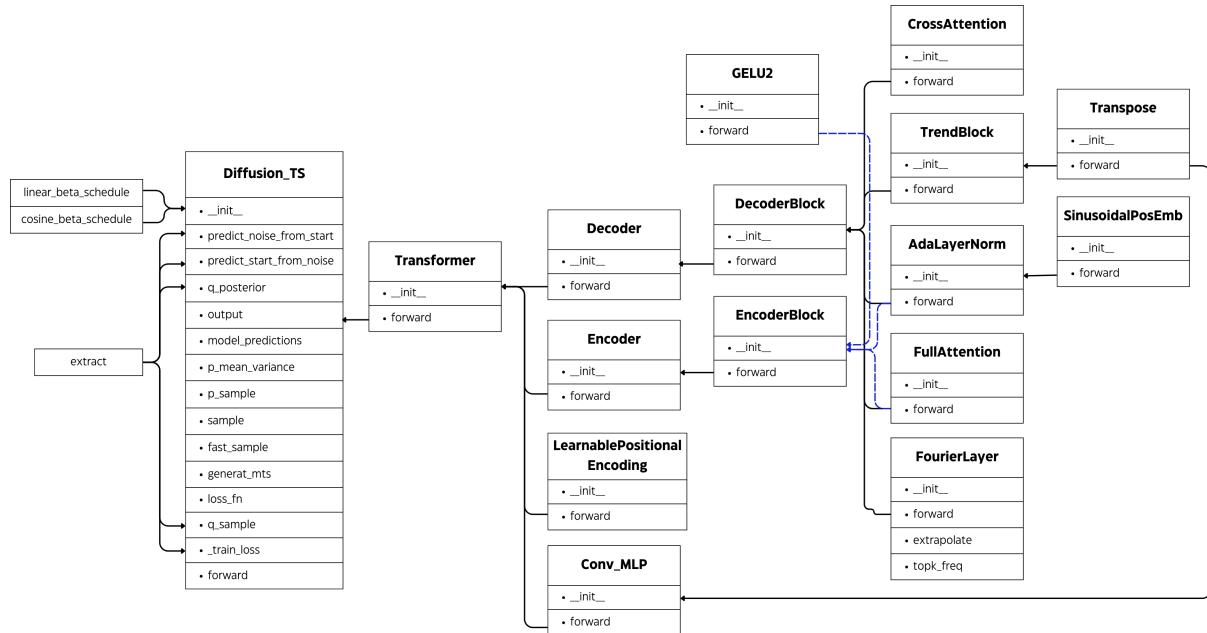


FIGURE 33 – Architecture du code de Diffusion-TS

6.13 Optimiseur Adam

Adam (Adaptive Moment Estimation) est un optimiseur largement utilisé dans le domaine de l'apprentissage automatique, en particulier pour l'entraînement de modèles de deep learning.

Les équations de mise à jour d'Adam sont :

— Calcul des moments

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) \nabla_\theta \mathcal{L}_t(\theta)$$

$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) (\nabla_\theta \mathcal{L}_t(\theta))^2$$

— Pour éviter le biais initial causé par la faible valeur des moments, on utilise des versions corrigées :

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$$

$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

— Mise à jour des poids

$$\theta \leftarrow \theta - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \epsilon}$$

Où m_t et v_t sont les estimations des deux premiers moments du gradient, β_1 et β_2 sont des hyperparamètres typiquement fixés à 0.9 et 0.96, η le taux d'apprentissage et ϵ est un terme de stabilité numérique pour éviter la division par zéro.

Les avantages de cet optimiseur sont : Convergence rapide et adaptative aux différentes dimensions des gradients. Résistant aux gradients rares ou à haute variance.

6.14 Précision sur les métriques utilisées pour mesurer la qualité de la génération

6.14.1 Le score Context-FID

1. Calcul des moyennes et des covariances des deux séries $(x_i^1)_i$ et $(x_i^2)_i$:

$$\begin{aligned} \mu_1 &= \frac{1}{n} \sum_{i=1}^n X_{1,i}, & \Sigma_1 &= \frac{1}{n} \sum_{i=1}^n (X_{1,i} - \mu_1)(X_{1,i} - \mu_1)^T \\ \mu_2 &= \frac{1}{n} \sum_{i=1}^n X_{2,i}, & \Sigma_2 &= \frac{1}{n} \sum_{i=1}^n (X_{2,i} - \mu_2)(X_{2,i} - \mu_2)^T \end{aligned}$$

2. Calcul de la différence entre les moyennes :

$$\text{SSDiff} = \|\mu_1 - \mu_2\|^2$$

3. Calcul de la racine carrée du produit des matrices de covariance :

$$\text{CovMean} = \sqrt{\Sigma_1 \cdot \Sigma_2}$$

4. Calcul du score FID :

$$\text{FID} = \text{SSDiff} + \text{Tr}(\Sigma_1 + \Sigma_2 - 2 \cdot \text{CovMean})$$

6.14.2 Perte de corrélation croisée

Voici les étapes suivies pour calculer la perte de corrélation croisée entre la série réelle et la série générée.

1. Normalisation : chacune des séries est normalisée de la manière suivante :

$$x_{\text{norm}} = \frac{x - \mu_x}{\sigma_x}$$

2. Calcul de la corrélation croisée pour un décalage k :

$$\text{CCF}_{ij}(k) = \frac{1}{T-k} \sum_{t=0}^{T-k-1} x_i(t+k)x_j(t)$$

3. Calcul de la corrélation croisée totale d'une série :

$$\text{CCF}_{\text{total}} = \frac{1}{|L|} \sum_{(i,j) \in L} \sum_{k=0}^{\text{max_lag}} \text{CCF}_{ij}(k)$$

4. Calcul de la perte (CrossCorrelLoss) :

$$\text{Loss} = \frac{1}{10} \sum_{(i,j)} |\text{CCF}_{ij}^{\text{fake}} - \text{CCF}_{ij}^{\text{real}}|$$

6.14.3 ACF : fonction d'autocorrélation

La fonction d'autocorrélation (ACF) d'une série temporelle $\{X_t\}_{t=1}^N$ à un lag k est définie par :

$$\text{ACF}(k) = \frac{\sum_{t=1}^{N-k} (X_t - \bar{X})(X_{t+k} - \bar{X})}{\sum_{t=1}^N (X_t - \bar{X})^2}$$

où $\bar{X} = \frac{1}{N} \sum_{t=1}^N X_t$.

L'ACF prend des valeurs entre -1 et +1. Elle est positive lorsqu'il existe une corrélation positive entre les valeurs à des moments espacés par k , négative lorsqu'il y a une corrélation inverse et nulle lorsqu'il n'y a pas de corrélation à ce lag.

6.14.4 Technique de réduction de la dimensionnalité

L'objectif est de réduire la dimension des données en gardant certaines de leurs caractéristiques de sorte qu'une série temporelle soit représentée par un unique point en 2D.

Avant utilisation des techniques de PCA, t-SNE ou KDE, il y a une étape de traitement des données qui permet de réduire la dimensionnalité en une première étape.

L'idée de cette **première étape** est la suivante :

Soient M séries multivariées $\{X_t\}_{t=1}^N$ dans $\mathbb{R}^{l \times f}$ avec l le nombre de lignes (pas de temps) et f le nombre de colonnes (coordonnées).

- Chacune des N séries est transformée en un vecteur de longueur l en moyennant les valeurs pour chaque instant temporel.
- On concatène les N vecteurs obtenus dans une matrice A de taille $N \times l$, chaque ligne provenant d'une unique série. Par la suite, on note $A = (a_i)_{i=1}^N$ avec pour tout i , $a_i \in \mathbb{R}^l$.

Pour le **PCA**, voici les étapes suivies afin d'obtenir une visualisation des séries en deux dimensions :

- Calcul de la matrice de covariance Σ des données centrées $\Sigma = \frac{1}{N} \sum_{i=1}^N (a_i - \bar{a})(a_i - \bar{a})^T$
- Calcul des vecteurs propres et des valeurs propres associées : $\Sigma \nu = \lambda \nu$ avec λ une valeur propre et ν le vecteur propre correspondant.

- Projection des données : on sélectionne les k vecteurs propres associés aux k plus grandes valeurs propres pour obtenir un sous-espace V_k de dimension réduite k . La projection de a_i sur cet espace est : $y_i = V_k^T a_i$. Dans notre cas, on a choisi $k = 2$.
- avec \bar{a} la moyenne empirique des $(a_i)_i$ et Σ une matrice de taille $l \times l$.

Pour le **t-SNE**, voici les étapes suivies :

- Définition de la probabilité de similarité dans l'espace d'origine entre deux points a_i et a_j : $p_{j|i} = \frac{\exp\left(-\frac{\|x_i - x_j\|^2}{2\sigma_i^2}\right)}{\sum_{k \neq i} \exp\left(-\frac{\|x_i - x_k\|^2}{2\sigma_i^2}\right)}$
Que l'on symétrise par : $p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$
- Définition de la probabilité de similarité dans l'espace réduit : $q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{k \neq l} (1 + \|y_k - y_l\|^2)^{-1}}$
- Minimisation de la divergence de Kullback-Leibler entre les distributions de similarité dans les espaces d'origine et réduit : $\text{KL}(P\|Q) = \sum_{i \neq j} p_{ij} \log \frac{p_{ij}}{q_{ij}}$ Cette minimisation est réalisée par descente de gradient pour trouver les positions des points $(y_i)_{i=1}^N$ dans l'espace réduit.

Pour la **KDE**, voici les étapes suivies :

- Obtention des représentations de chaque séries dans le plan en repartant d'une des techniques précédentes (PCA ou t-SNE)
- Estimation de la densité grâce à la technique des noyaux : $\hat{f}(y) = \frac{1}{Nh} \sum_{i=1}^N K\left(\frac{y-y_i}{h}\right)$, où h est la largeur de bande et $K(\cdot)$ est un noyau, souvent choisi comme gaussien $K(y) = \frac{1}{\sqrt{2\pi}} \exp -\frac{y^2}{2}$.

6.14.5 Coefficient de Hurst

- **H < 0.5** : La série a tendance à revenir à sa moyenne après une augmentation ou une diminution (comportement anti-persistant).
- **H = 0.5** : La série se comporte comme un processus aléatoire, sans tendance particulière et n'a pas de mémoire. Les mouvements passés n'affectent pas les mouvements futurs.
- **H > 0.5** : La série présente une tendance persistante, ce qui signifie que les mouvements passés influencent les mouvements futurs et qu'elle a tendance à continuer dans la même direction.

6.14.6 Divergence KL-Fourier

1. Calcul de la densité spectrale de puissance (DSP) :

$$\text{DSP}(f) = \frac{1}{T} \left| \sum_{t=1}^T x(t) e^{-i2\pi f t} \right|^2$$

Puis, la divergence de Kullback-Leibler est calculée entre les densités spectrales des séries temporelles réelles et générées. Cette divergence mesure la perte d'information lorsqu'une distribution (la densité spectrale générée) est utilisée pour approximer une autre (la densité spectrale réelle) :

2. Divergence KL :

$$D_{\text{KL}}(\text{DSP}_{\text{real}} \parallel \text{DSP}_{\text{fake}})$$

6.14.7 Distance de Wasserstein-Fourier

1. Calcul de la densité spectrale de puissance (DSP) :

$$\text{DSP}(f) = \frac{1}{T} \left| \sum_{t=1}^T x(t) e^{-i2\pi f t} \right|^2$$

Ensuite, la distance de Wasserstein est calculée entre les densités spectrales des séries temporelles réelles et générées. Cette distance évalue la quantité de "transport" nécessaire pour transformer une densité spectrale en une autre :

2. Distance de Wasserstein :

$$W(\text{DSP}_{\text{real}}, \text{DSP}_{\text{fake}})$$

6.14.8 Distance de Wasserstein

La distance de Wasserstein, également appelée "Earth Mover's Distance" (EMD), mesure le coût minimal pour transformer une distribution en une autre.

Définition 1. Considérons deux distributions de probabilité discrètes, notées p et q , avec des points supports $\{\mathbf{x}_i\}_{i=1}^n$ pour p et $\{y_j\}_{j=1}^m$ pour q . Les probabilités associées à chaque point support sont respectivement p_i et q_j .

Distance de Wasserstein 1D

Si les distributions p et q sont définies sur une seule dimension (par exemple, une distribution de valeurs scalaires), la distance de Wasserstein de premier ordre, également appelée W_1 -distance, est donnée par :

$$W_1(p, q) = \inf_{\gamma \in \Gamma(p, q)} \int_{\mathbb{R} \times \mathbb{R}} |x - y| d\gamma(x, y)$$

Où :

- $\Gamma(p, q)$ est l'ensemble des plans de transport, c'est-à-dire des distributions jointes γ qui ont p et q comme marges. Autrement dit, $\gamma(x, y)$ décrit comment "transporter" la masse de p à q .
- $|x - y|$ est la distance entre les points x et y .

Le but est de minimiser cette distance en trouvant le plan de transport optimal γ .

Dans le cas discret où p et q sont des vecteurs de probabilité définis sur les mêmes points support, la distance de Wasserstein devient :

$$W_1(p, q) = \sum_{i=1}^n |F_p(\mathbf{x}_i) - F_q(\mathbf{x}_i)|$$

Où F_p et F_q sont les fonctions de distribution cumulées des distributions p et q .

Ne prend pas en compte la dynamique temporelle présente dans le cas de séries temporelles.

L'idée est donc de calculer la distance de Wasserstein entre les densités spectrales de puissance de séries stationnaires. (voir The Wasserstein-Fourier Distance for Stationary Time Series [2])

6.15 Expériences et résultats

6.15.1 Méthode 1

Pour rappel, dans cette expérience, on a tiré aléatoirement une série parmi les 2761 de l'ensemble d'entraînement pour le Bitcoin ou les 2281 des données du S&P500. Après avoir bruité le signal via le processus utilisé dans Diffusion-TS, on le met en entrée du modèle entraîné pour générer une image du signal initial.

A suivre, les résultats obtenus en comparant le signal initial et le signal généré.

Pour le bitcoin :

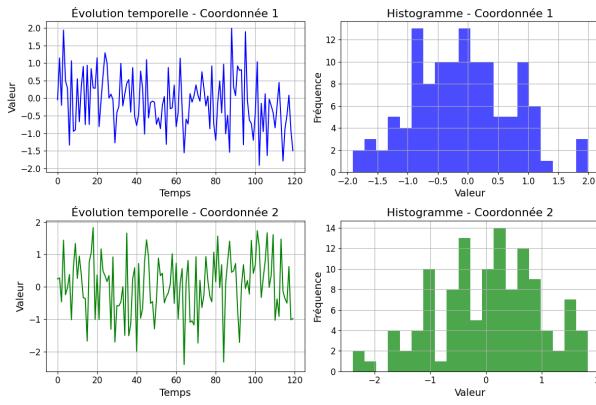


FIGURE 34 – Signal obtenu après bruitage

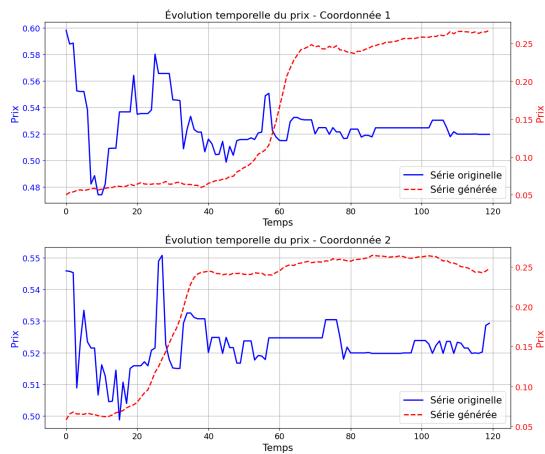


FIGURE 35 – Évolution temporelle des prix du Bitcoin

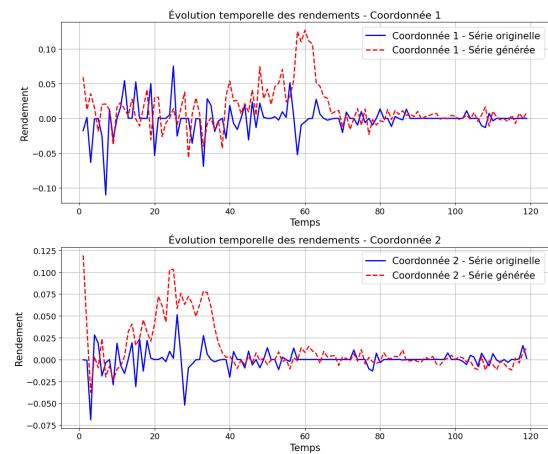


FIGURE 36 – Évolution temporelle des rendements du Bitcoin

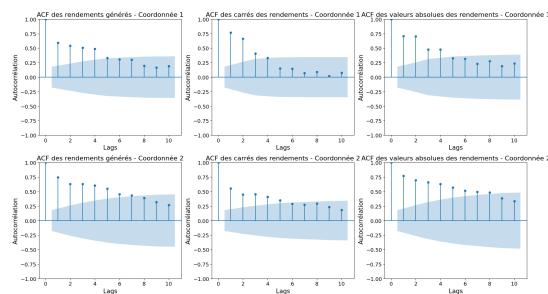
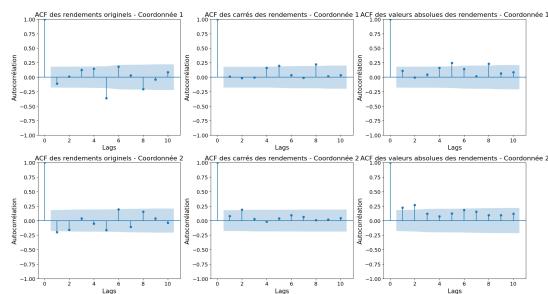


FIGURE 37 – ACF des rendements : à gauche, série originelle ; à droite, série générée

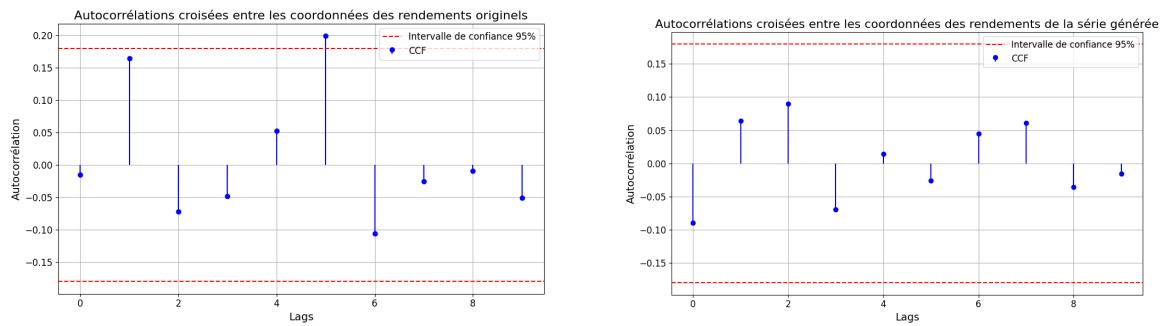


FIGURE 38 – CCF entre les coordonnées des rendements : à gauche, rendements originels ; à droite, rendements générés

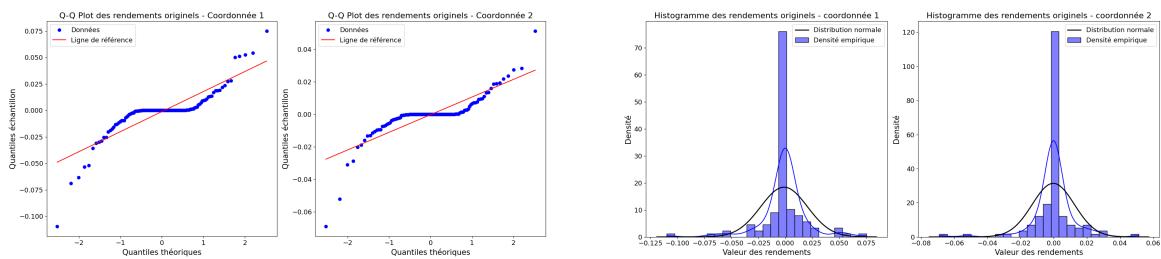


FIGURE 39 – À gauche, QQ plot des rendements originels ; à droite, histogramme des rendements originels

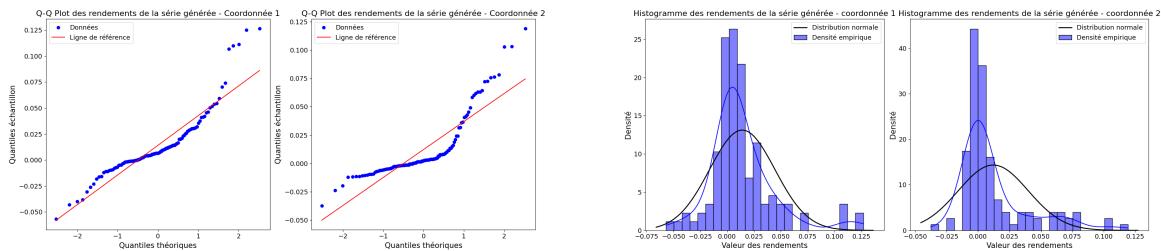


FIGURE 40 – À gauche, QQ plot des rendements générés ; à droite, histogramme des rendements générés

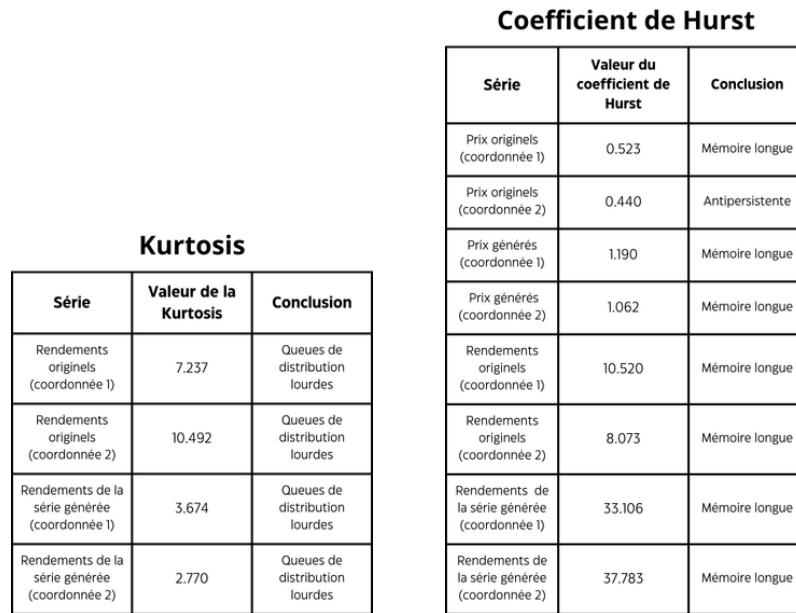


FIGURE 41 – À gauche, Kurtosis des rendements ; à droite, Hurst exponent des rendements

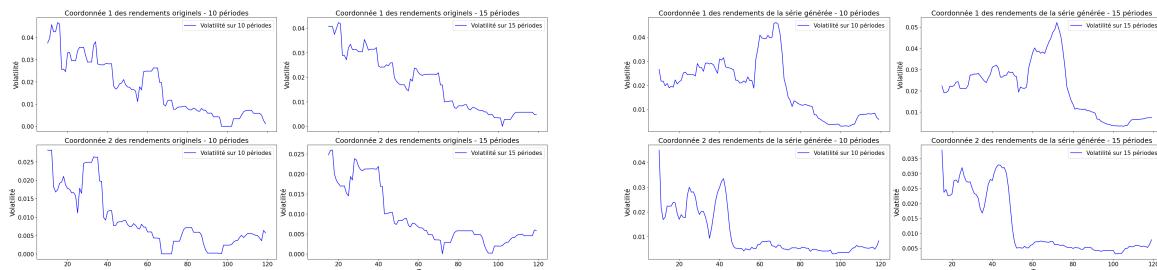


FIGURE 42 – À gauche, volatilité des rendements originels ; à droite, volatilité des rendements générés

Test de stationnarité ADF

Série	p-valeur	Conclusion
Rendements originels (coordonnée 1)	1.337×10^{-6}	Stationnaire
Rendements originels (coordonnée 2)	9.367×10^{-5}	Stationnaire
Carrés des rendements originels (coordonnée 1)	0.119	Non stationnaire
Carrés des rendements originels (coordonnée 2)	0.161	Non stationnaire
Rendements de la série générée (coordonnée 1)	0.058	Non stationnaire (au seuil de 5%)
Rendements de la série générée (coordonnée 2)	0.207	Non stationnaire
Carrés des rendements de la série générée (coordonnée 1)	0.006	Stationnaire
Carrés des rendements de la série générée (coordonnée 2)	0.162	Non stationnaire

Test d'absence d'autocorrélations (Ljung-Box)

Série	p-valeur	Conclusion
Rendements originels (coordonnée 1)	0.002×10^{-1}	Présence d'autocorrélations significatives
Rendements originels (coordonnée 2)	0.019	Présence
Carrés des rendements originels (coordonnée 1)	0.143	Absence
Carrés des rendements originels (coordonnée 2)	0.688	Absence
Rendements de la série générée (coordonnée 1)	0.000	Présence
Rendements de la série générée (coordonnée 2)	0.000	Présence
Carrés des rendements de la série générée (coordonnée 1)	0.000	Présence
Carrés des rendements de la série générée (coordonnée 2)	0.000	Présence

FIGURE 43 – Résultats des tests statistiques sur les rendements

Distance de Wasserstein-Fourier entre les séries originelle et générée :

- Pour la coordonnée 1 : 0.0028
- Pour la coordonnée 2 : 0.0006

Pour le S&P500 :

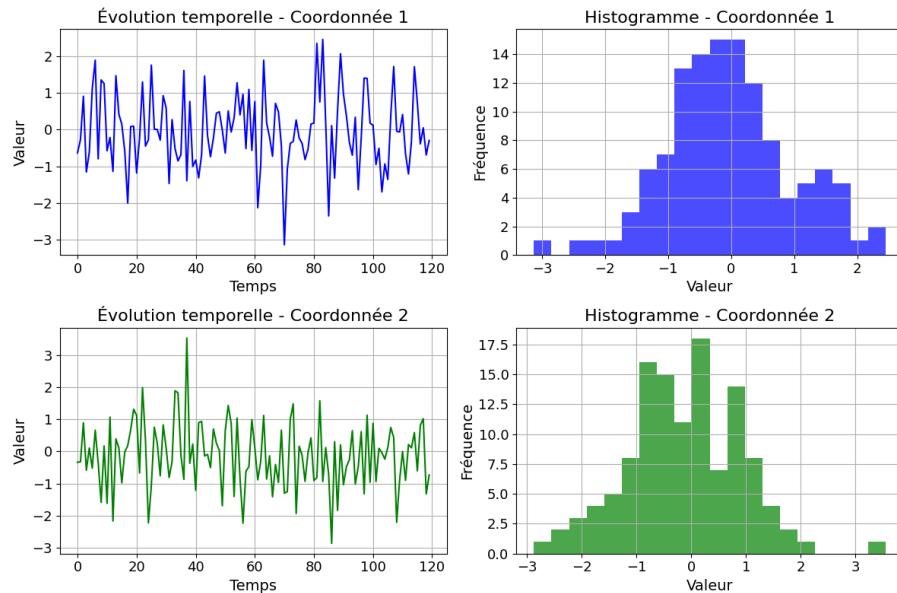


FIGURE 44 – Evolution temporelle et distribution du signal obtenu après bruitage

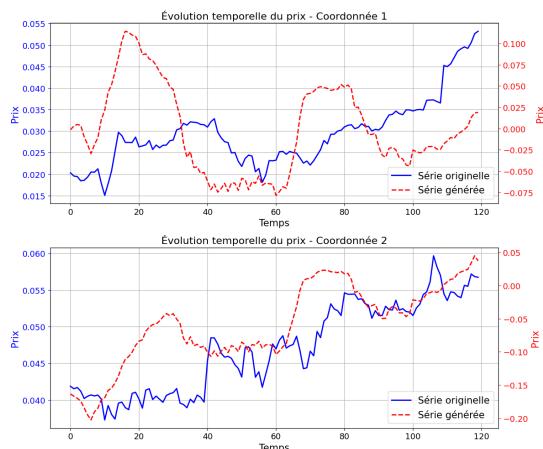


FIGURE 45 – Évolution temporelle des prix du S&P500

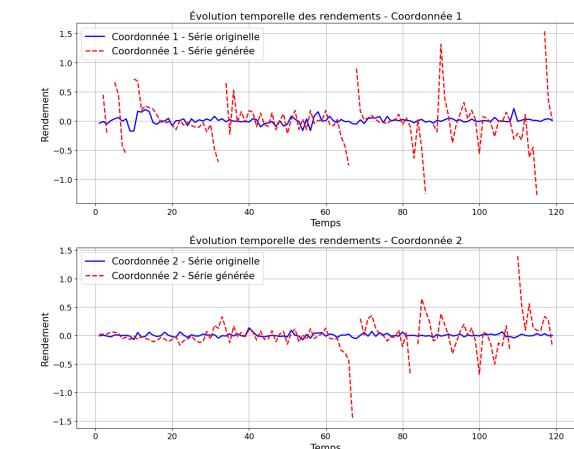


FIGURE 46 – Évolution temporelle des rendements du S&P500

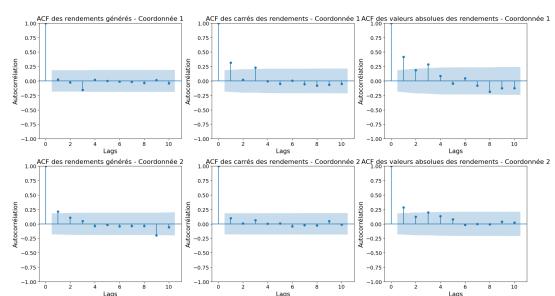
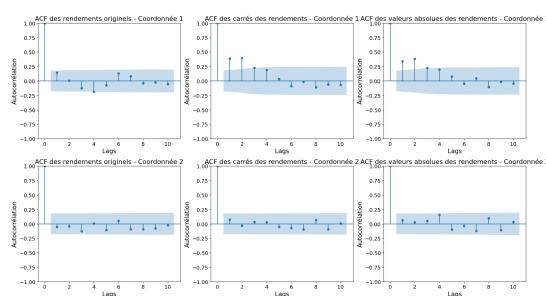


FIGURE 47 – ACF des rendements du S&P500 : à gauche, série originelle ; à droite, série générée

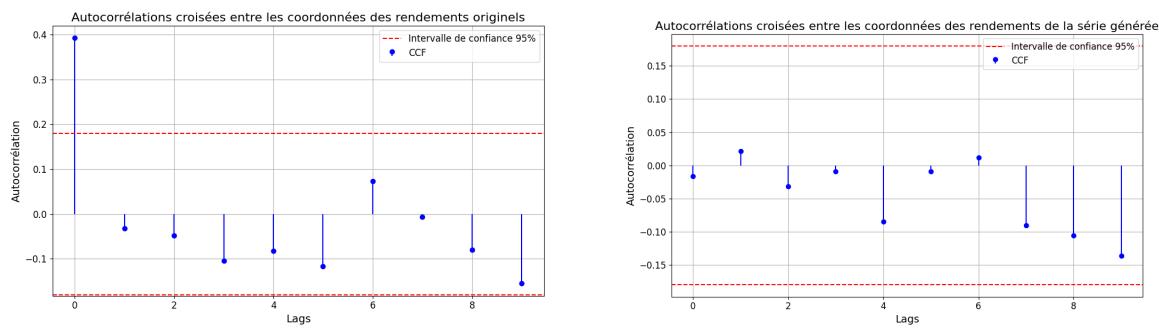


FIGURE 48 – CCF entre les coordonnées : à gauche, rendements originels ; à droite, rendements synthétiques

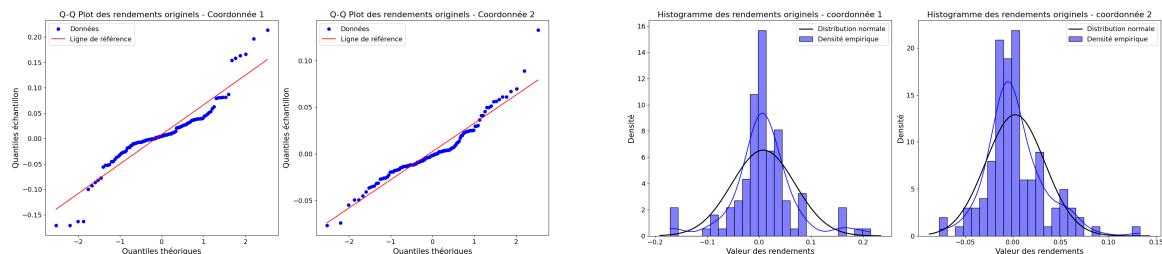


FIGURE 49 – À gauche : QQ-plot des rendements originels ; à droite : Histogramme des rendements originels

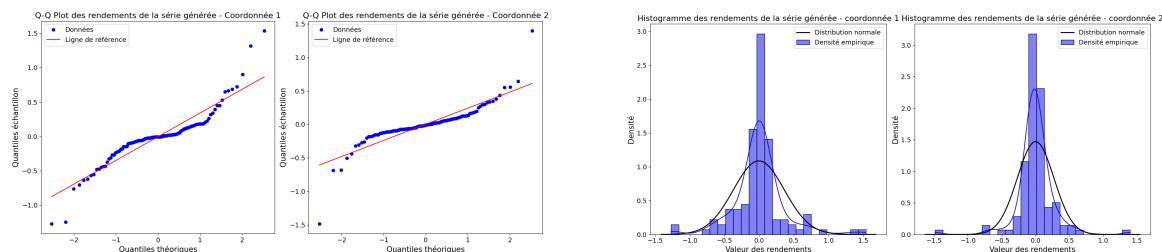


FIGURE 50 – À gauche : QQ-plot des rendements synthétiques ; à droite : Histogramme des rendements synthétiques

Kurtosis		
Série	Valeur de la Kurtosis	Conclusion
Rendements originels (coordonnée 1)	2.986	Queue de distribution lourdes
Rendements originels (coordonnée 2)	2.379	Queue de distribution lourdes
Rendements de la série générée (coordonnée 1)	4.828	Queue de distribution lourdes
Rendements de la série générée (coordonnée 2)	12.332	Queue de distribution lourdes

FIGURE 51 – Coefficient de Kurtosis

Test de stationnarité ADF

Série	p-valeur	Conclusion
Rendements originels (coordonnée 1)	9.529×10^{-16}	Stationnaire
Rendements originels (coordonnée 2)	6.643×10^{-21}	Stationnaire
Carrés des rendements originels (coordonnée 1)	3.159×10^{-4}	Stationnaire
Carrés des rendements originels (coordonnée 2)	1.692×10^{-17}	Stationnaire
Rendements de la série générée (coordonnée 1)	2.506×10^{-12}	Stationnaire
Rendements de la série générée (coordonnée 2)	2.682×10^{-12}	Stationnaire
Carrés des rendements de la série générée (coordonnée 1)	3.888×10^{-7}	Stationnaire
Carrés des rendements de la série générée (coordonnée 2)	7.513×10^{-18}	Stationnaire

Test d'absence d'autocorrélations (Ljung-Box)

Série	p-valeur	Conclusion
Rendements originels (coordonnée 1)	0.195	Absence d'autocorrélations significatives
Rendements originels (coordonnée 2)	0.665	Absence
Carrés des rendements originels (coordonnée 1)	0.000	Présence
Carrés des rendements originels (coordonnée 2)	0.895	Absence
Rendements de la série générée (coordonnée 1)	0.226	Absence
Rendements de la série générée (coordonnée 2)	0.029	Présence
Carrés des rendements de la série générée (coordonnée 1)	0.159	Absence
Carrés des rendements de la série générée (coordonnée 2)	0.997	Absence

FIGURE 52 – Résultats des tests statistiques réalisés

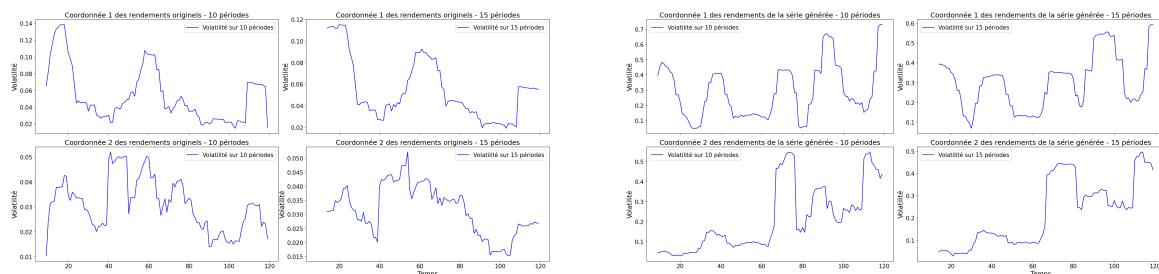


FIGURE 53 – Volatilité : à gauche, série originelle ; à droite, série générée

Distance de Wasserstein-Fourier entre les séries originelle et générée :

- Pour la coordonnée 1 : 0.0028
- Pour la coordonnée 2 : 0.0006

6.15.2 Méthode 2

Pour le bitcoin : Élément tiré = 312

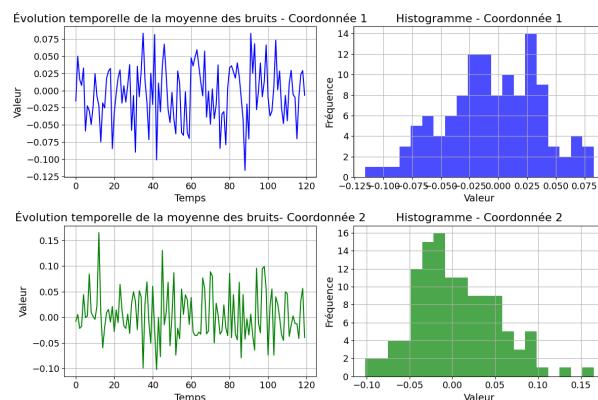


FIGURE 55 – Bruit pour le Bitcoin

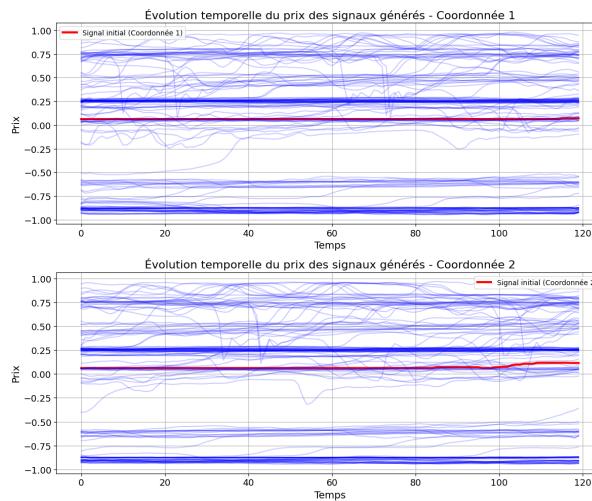


FIGURE 54 – 150 signaux générés sur les 500 ainsi que le signal original

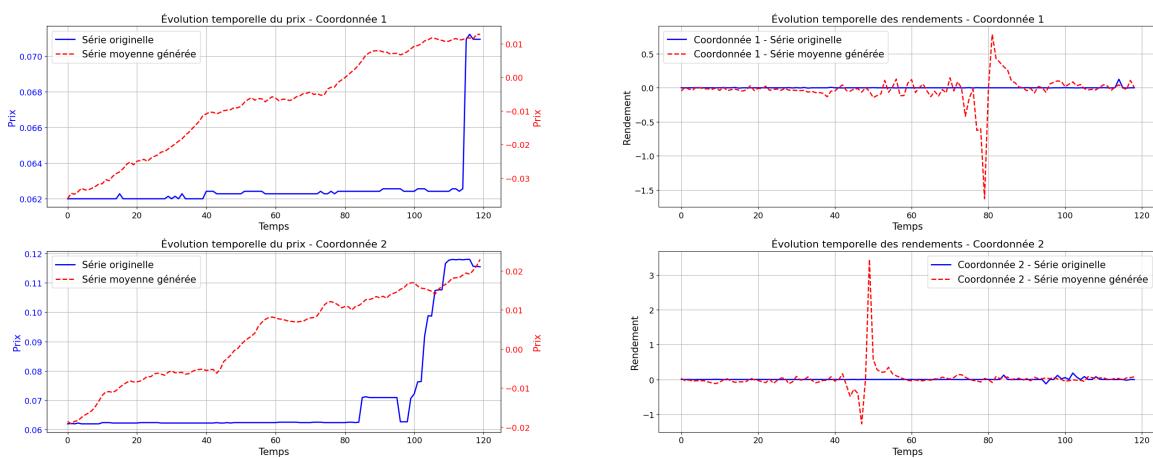


FIGURE 56 – À gauche : Évolution temporelle des prix du Bitcoin ; à droite : Évolution temporelle des rendements du Bitcoin

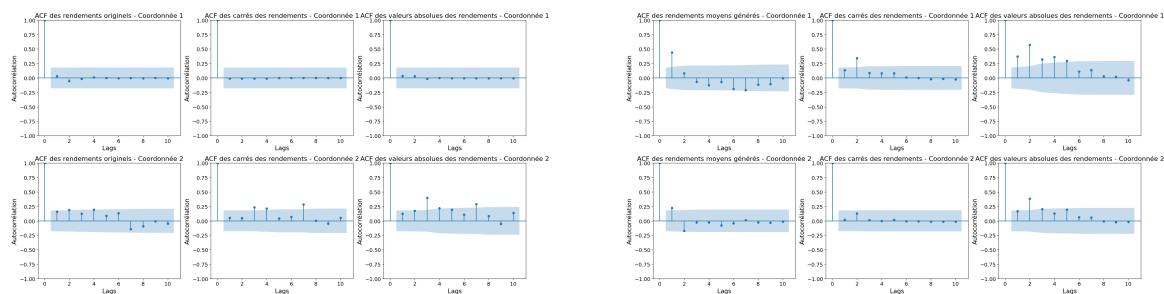


FIGURE 57 – ACF des rendements : à gauche, série originelle ; à droite, série générée

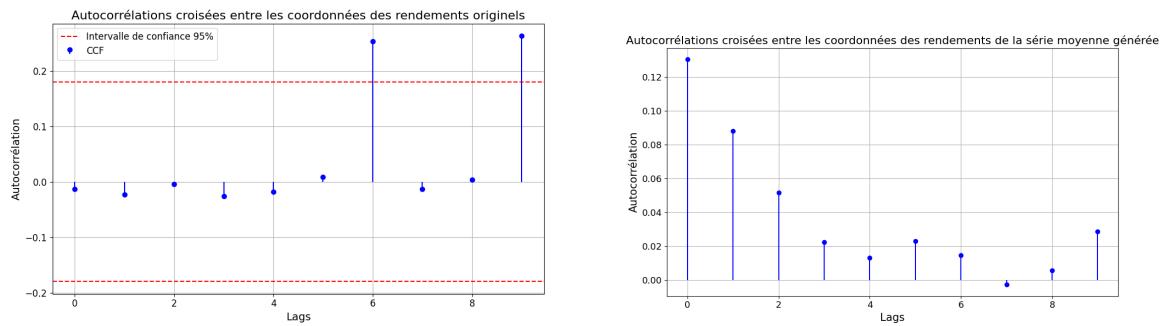


FIGURE 58 – CCF entre les coordonnées : à gauche, rendements originels ; à droite, rendements synthétiques

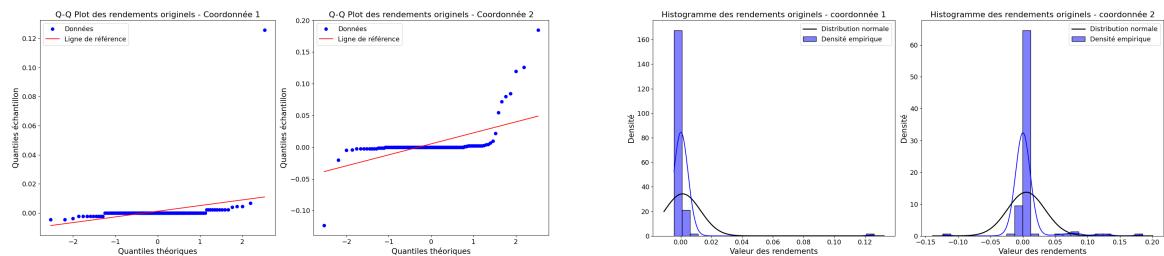


FIGURE 59 – À gauche : QQ-plot des rendements originels ; à droite : Histogramme des rendements originels

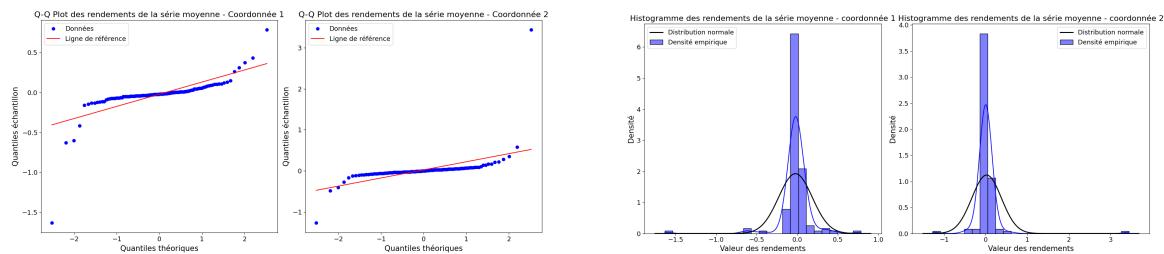


FIGURE 60 – À gauche : QQ-plot des rendements synthétiques ; à droite : Histogramme des rendements synthétiques

Coefficient de Hurst

Série	Valeur du coefficient de Hurst	Conclusion
Prix originels (coordonnée 1)	23.832	Mémoire longue
Prix originels (coordonnée 2)	41.262	Mémoire longue
Prix moyens générés (coordonnée 1)	49.754	Mémoire longue
Prix moyens générés (coordonnée 2)	53.821	Mémoire longue
Rendements originels (coordonnée 1)	11.114	Mémoire longue
Rendements originels (coordonnée 2)	19.625	Mémoire longue
Rendements de la série moyenne (coordonnée 1)	18.324	Mémoire longue
Rendements de la série moyenne (coordonnée 2)	14.708	Mémoire longue

Kurtosis

Série	Valeur de la Kurtosis	Conclusion
Rendements originels (coordonnée 1)	110.335	Queues de distribution lourdes
Rendements originels (coordonnée 2)	18.736	Queues de distribution lourdes
Rendements de la série moyenne (coordonnée 1)	32.308	Queues de distribution lourdes
Rendements de la série moyenne (coordonnée 2)	72.627	Queues de distribution lourdes

FIGURE 61 – À gauche : Coefficient de Kurtosis ; à droite : Coefficient de Hurst

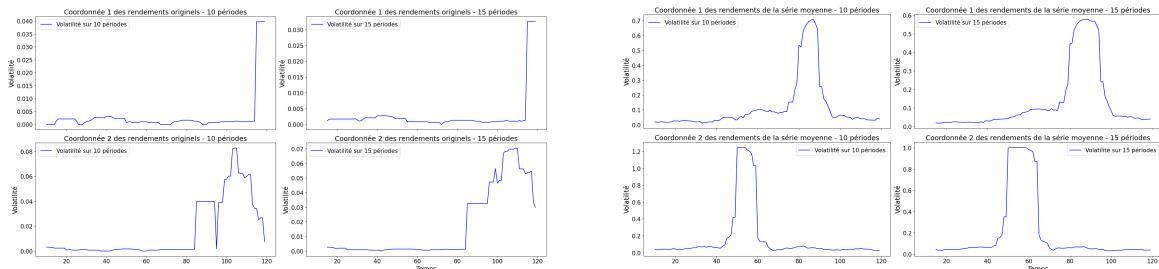


FIGURE 62 – Volatilité : à gauche, série originelle ; à droite, série générée

Test de stationnarité ADF

Série	p-valeur	Conclusion
Rendements originels (coordonnée 1)	1.6789×10^{-18}	Stationnaire
Rendements originels (coordonnée 2)	3.936×10^{-7}	Stationnaire
Carrés des rendements originels (coordonnée 1)	1.524×10^{-19}	Stationnaire
Carrés des rendements originels (coordonnée 2)	0.011	Stationnaire
Rendements de la série moyenne (coordonnée 1)	9.860×10^{-9}	Stationnaire
Rendements de la série moyenne (coordonnée 2)	1.483×10^{-13}	Stationnaire
Carrés des rendements de la série moyenne (coordonnée 1)	2.153×10^{-5}	Stationnaire
Carrés des rendements de la série moyenne (coordonnée 2)	5.164×10^{-19}	Stationnaire

Test d'absence d'autocorrélations (Ljung-Box)

Série	p-valeur	Conclusion
Rendements originels (coordonnée 1)	1.000	Absence d'autocorrelations significatives
Rendements originels (coordonnée 2)	0.018	Présence
Carrés des rendements originels (coordonnée 1)	1.000	Absence
Carrés des rendements originels (coordonnée 2)	0.005	Présence
Rendements de la série moyenne (coordonnée 1)	0.000	Présence
Rendements de la série moyenne (coordonnée 2)	0.343	Absence
Carrés des rendements de la série moyenne (coordonnée 1)	0.035	Présence
Carrés des rendements de la série moyenne (coordonnée 2)	0.994	Absence

FIGURE 63 – Tests statistiques pour le Bitcoin

La distance de Wasserstein-Fourier entre les séries originelle et générée est :

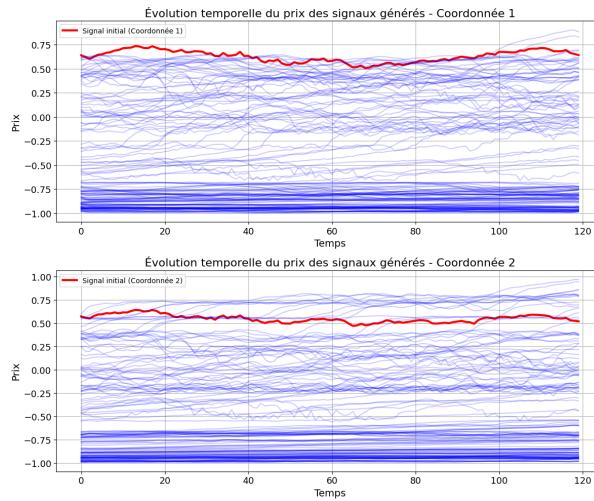


FIGURE 64 – 150 signaux générés parmi les 500 ainsi que le signal originel

- Pour la coordonnée 1 : 0.119
- Pour la coordonnée 2 : 0.567

Pour le S&P500 :

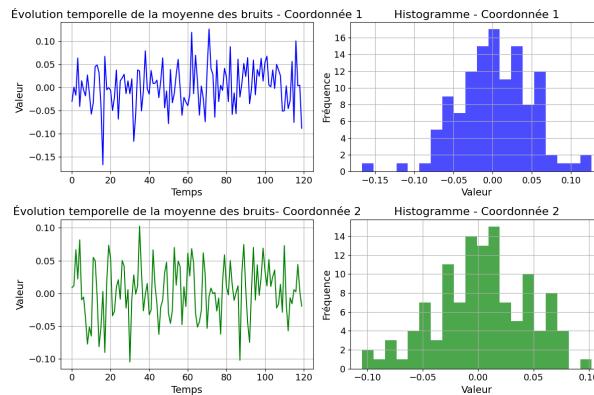


FIGURE 65 – Bruit pour le S&P 500

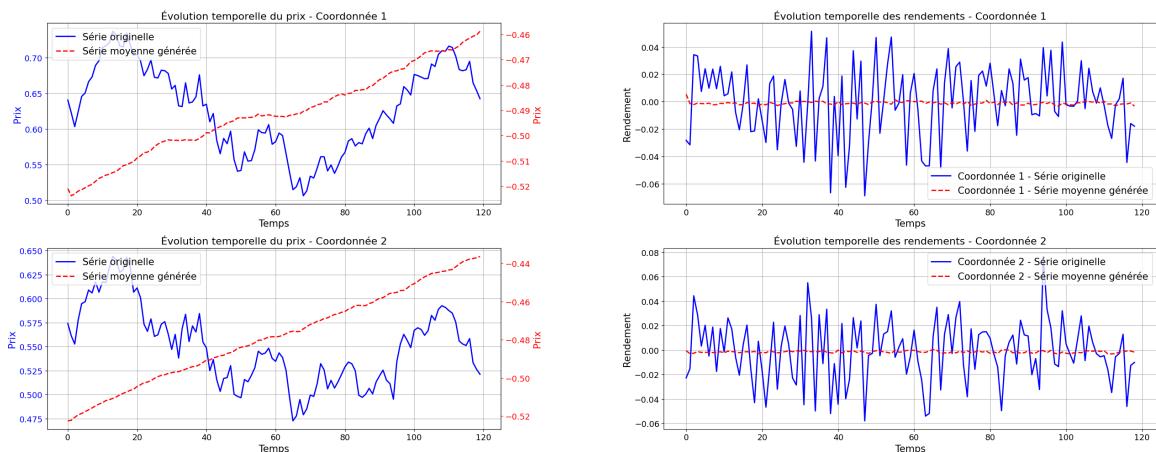


FIGURE 66 – À gauche : Évolution temporelle des prix du S&P 500 ; à droite : Évolution temporelle des rendements du S&P 500

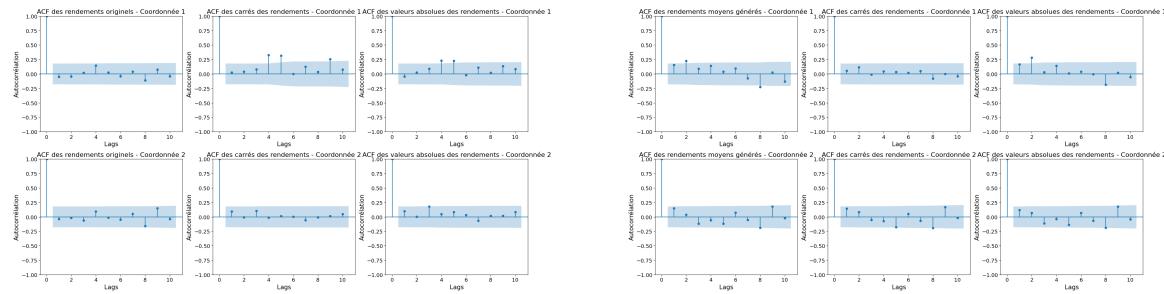


FIGURE 67 – ACF des rendements : à gauche, série originelle ; à droite, série générée

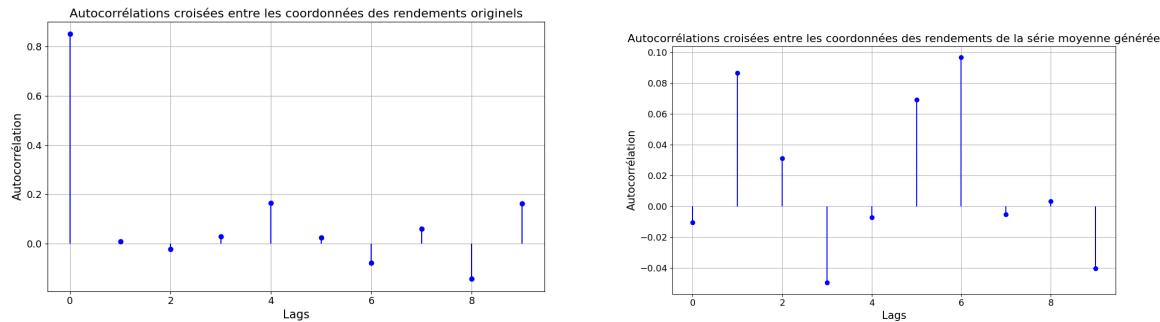


FIGURE 68 – CCF entre les coordonnées : à gauche, rendements originels ; à droite, rendements synthétiques

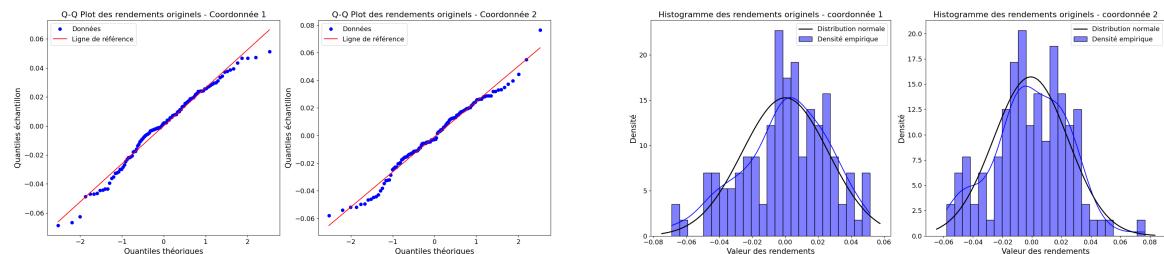


FIGURE 69 – À gauche : QQ-plot des rendements originels ; à droite : Histogramme des rendements originels

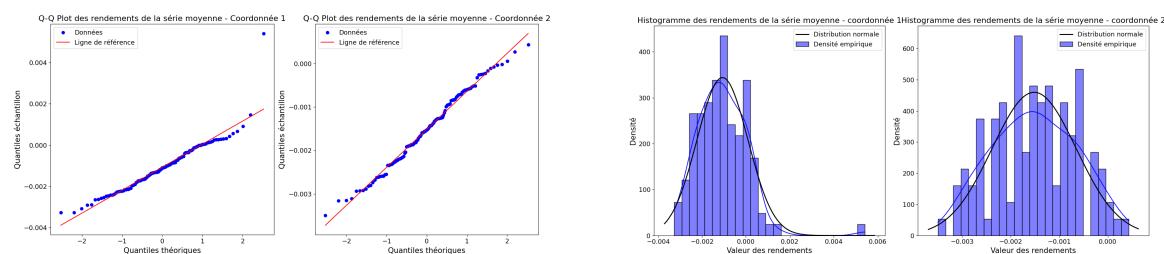


FIGURE 70 – À gauche : QQ-plot des rendements synthétiques ; à droite : Histogramme des rendements synthétiques

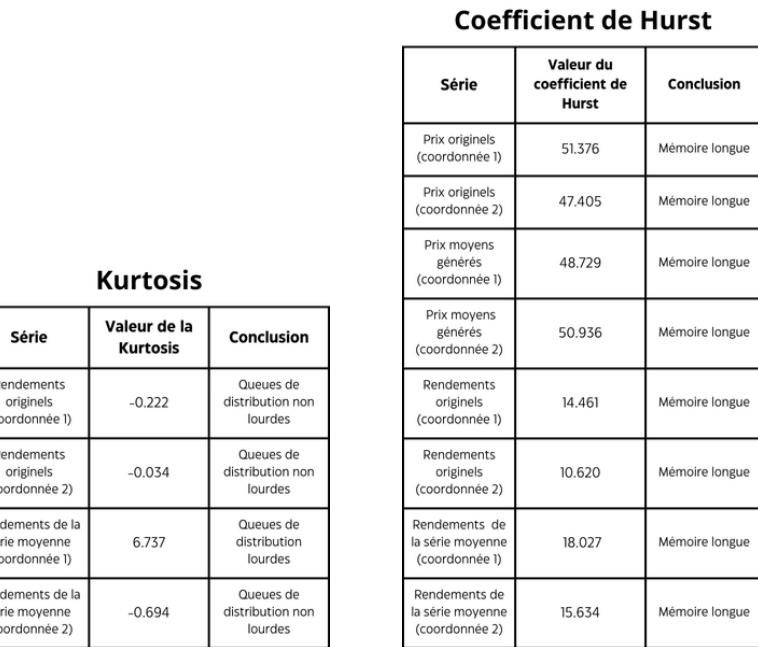


FIGURE 71 – À gauche : Coefficient de Kurtosis ; à droite : Coefficient de Hurst

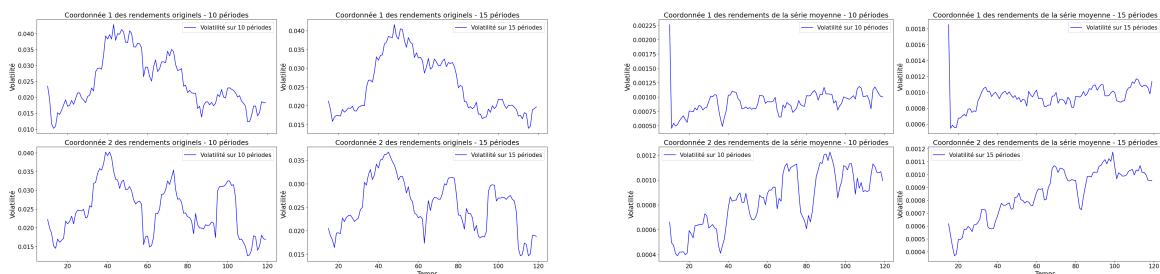


FIGURE 72 – Volatilité : à gauche, série originelle ; à droite, série générée

Test de stationnarité ADF			Test d'absence d'autocorrélations (Ljung-Box)		
Série	p-valeur	Conclusion	Série	p-valeur	Conclusion
Rendements originels (coordonnée 1)	9.155×10^{-21}	Stationnaire	Rendements originels (coordonnée 1)	0.794	Absence d'autocorrelations significatives
Rendements originels (coordonnée 2)	2.256×10^{-20}	Stationnaire	Rendements originels (coordonnée 2)	0.563	Absence
Carrés des rendements originels (coordonnée 1)	0.191	Non stationnaire	Carrés des rendements originels (coordonnée 1)	0.000	Présence
Carrés des rendements originels (coordonnée 2)	6.979×10^{-17}	Stationnaire	Carrés des rendements originels (coordonnée 2)	0.976	Absence
Rendements de la série moyenne (coordonnée 1)	0.013	Stationnaire	Rendements de la série moyenne (coordonnée 1)	0.008	Présence
Rendements de la série moyenne (coordonnée 2)	0.016	Stationnaire	Rendements de la série moyenne (coordonnée 2)	0.082	Absence
Carrés des rendements de la série moyenne (coordonnée 1)	0.019	Stationnaire	Carrés des rendements de la série moyenne (coordonnée 1)	0.954	Absence
Carrés des rendements de la série moyenne (coordonnée 2)	0.354	Non stationnaire	Carrés des rendements de la série moyenne (coordonnée 2)	0.057	Absence

FIGURE 73 – Tests statistiques pour le S&P 500

La distance de Wasserstein-Fourier entre les séries originelle et générée est :

- Pour la coordonnée 1 : 0.0018
- Pour la coordonnée 2 : 0.0014

6.15.3 Méthode 3

Pour le bitcoin :

Mesure	Série	Coordonnée	Moyenne	Ecart-type	Intervalle de confiance à 95%	Nombre d'outliers	Intervalle de confiance à 95% sans outliers
Distance L2	Prix	1	5.807	3.608	[−1.265, 12.880]	0	
		2	5.804	3.622	[−1.296, 12.904]	0	
	Rendements	1	0.297	0.401	[−0.487, 1.082]	72	[−0.487, 0.262]
		2	0.404	0.245	[−0.076, 0.883]	66	[−0.076, 0.385]
Divergence KL - Fourier	Prix	1	0.0001	0.004	[−0.008, 0.010]	79	[−0.0002, 0.0004]
		2	0.0019	0.0035	[−0.0049, 0.0088]	35	[−0.0007, 0.0034]
	Rendements	1	0.0081	0.0041	[0.0001, 0.0161]	0	
		2	0.010	0.003	[0.003, 0.016]	0	
Distance Wasserstein - Fourier	Prix	1	0.005	0.010	[−0.015, 0.025]	49	[−0.004, 0.010]
		2	0.005	0.009	[−0.013, 0.023]	46	[−0.004, 0.009]
	Rendements	1	0.004	0.023	[−0.042, 0.050]	0	
		2	0.002	0.008	[−0.013, 0.016]	0	

FIGURE 74 – Mesures

Coefficient	Série	Coordonnée	Moyenne	Ecart-type	Nombre de coefficients en dessous du seuil caractéristique	Nombre de coefficients au-dessus du seuil caractéristique
Coefficient de Hurst (seuil = 0.5)	Prix	1	0.646	0.158	98	402
		2	0.628	0.176	129	371
	Rendements	1	0.702	0.214	91	409
		2	0.694	0.229	102	398
Kurtosis (seuil = 0)	Prix	1	0.127	2.887	339	161
		2	-0.103	2.022	364	136
	Rendements	1	9.105	12.939	10	442
		2	7.767	12.927	31	414

FIGURE 75 – Coefficients de Kurtosis et de Hurst

Corrélation de Pearson		
Série des prix		
Relation	Nombre	Pourcentage significatif (%)
Faible négative	174	77.59
Faible positive	146	59.59
Modérée négative	0	0
Modérée positive	145	100.00
Forte négative	0	0
Forte positive	28	100.00
Très forte négative	0	0
Très forte positive	7	100.00

Corrélation de Spearman		
Série des prix		
Relation	Nombre	Pourcentage significatif (%)
Faible négative	174	90.23
Faible positive	39	56.41
Modérée négative	0	0
Modérée positive	43	100.00
Forte négative	0	0
Forte positive	74	100.00
Très forte négative	0	0
Très forte positive	170	100.00

Série des rendements		
Relation	Nombre	Pourcentage significatif (%)
Faible négative	268	2.99
Faible positive	226	3.10
Modérée négative	0	0
Modérée positive	4	100.00
Forte négative	0	0
Forte positive	2	100.00
Très forte négative	0	0
Très forte positive	0	0

Série des rendements		
Relation	Nombre	Pourcentage significatif (%)
Faible négative	254	1.57
Faible positive	246	2.03
Modérée négative	0	0
Modérée positive	0	0
Forte négative	0	0
Forte positive	0	0
Très forte négative	0	0
Très forte positive	0	0

FIGURE 76 – Corrélation

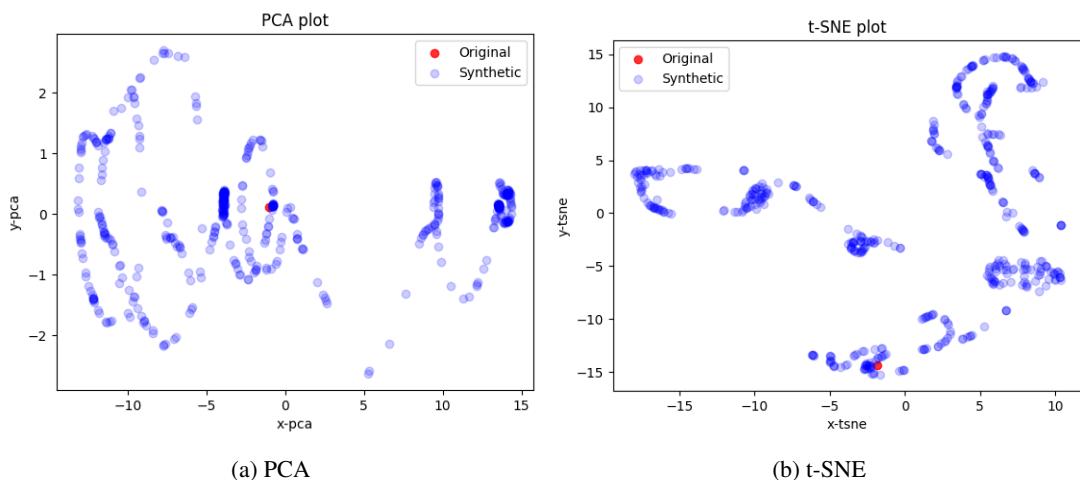


FIGURE 77 – Corrélation et visualisations en dimension réduite (PCA, t-SNE)

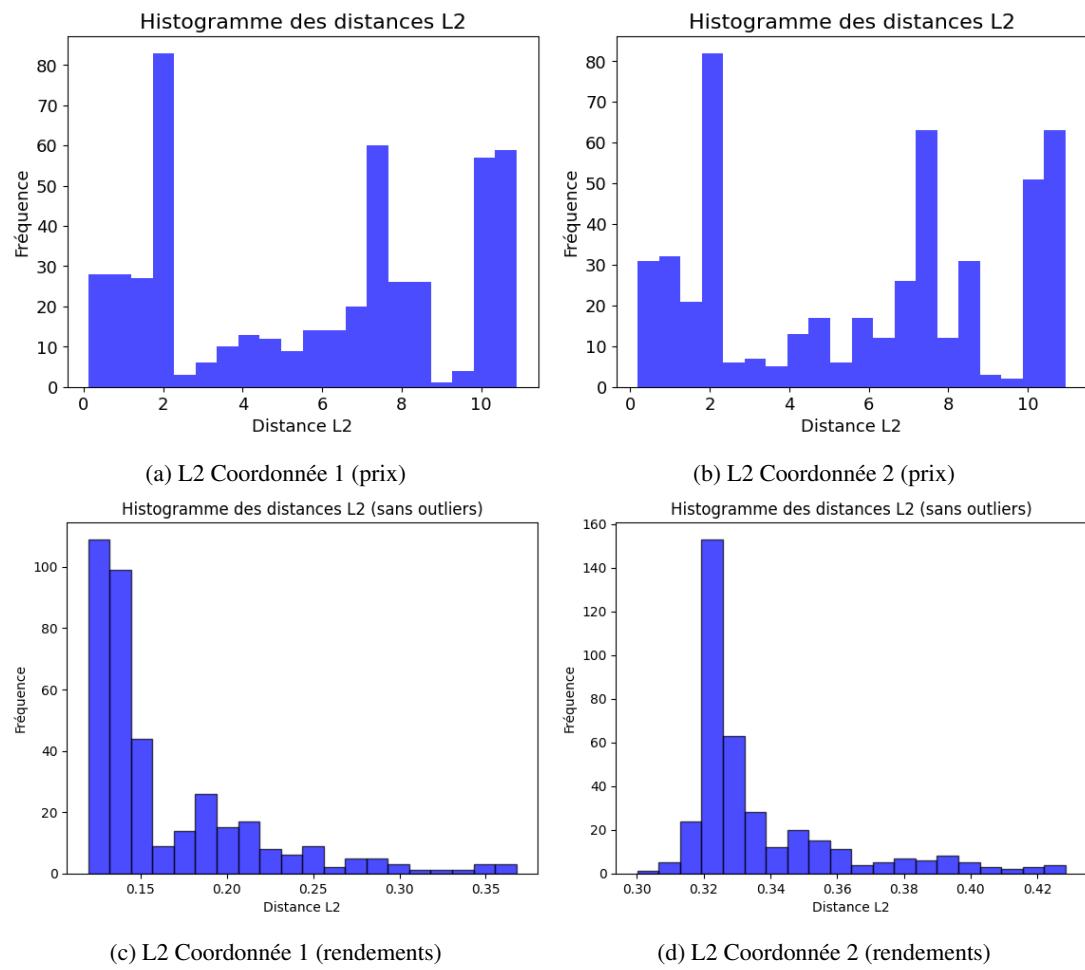


FIGURE 78 – Histogrammes de la distance L2 pour chaque coordonnée (prix et rendement)

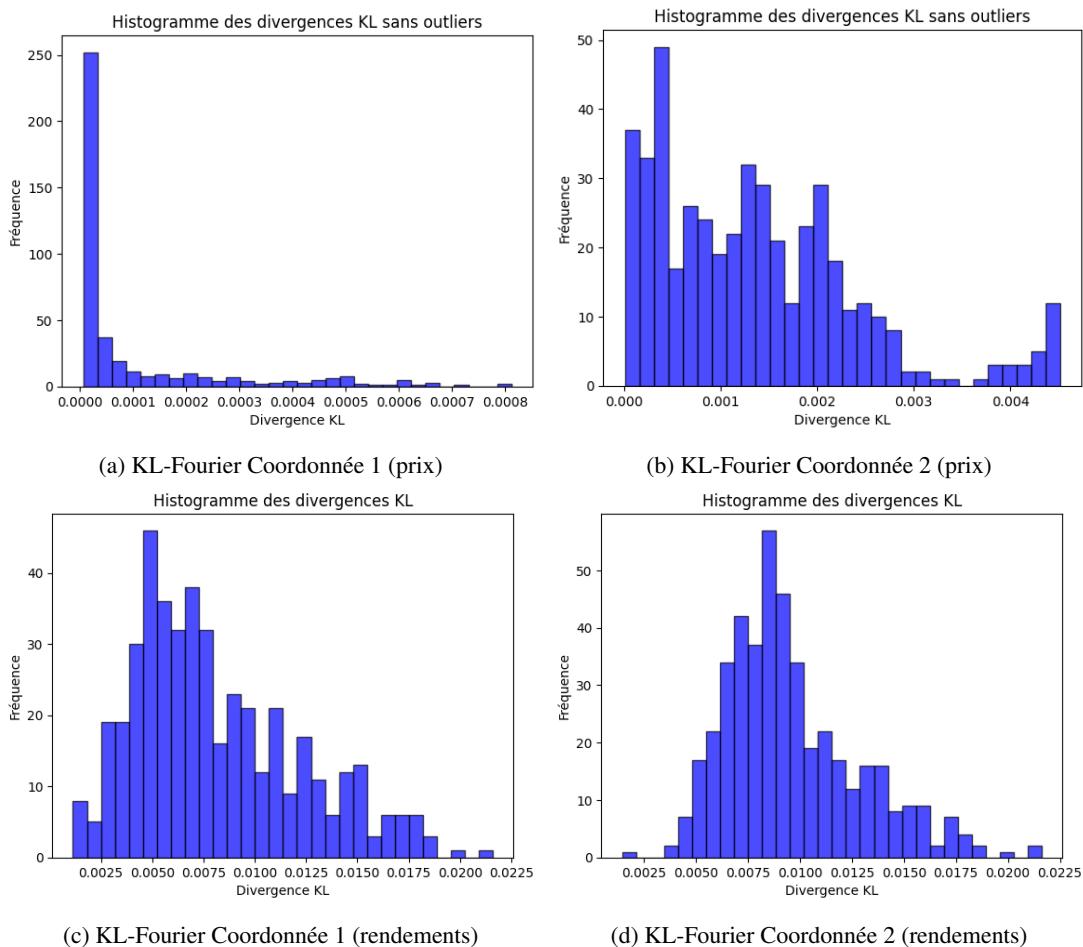


FIGURE 79 – Divergence KL-Fourier pour chaque coordonnée (prix et rendements)

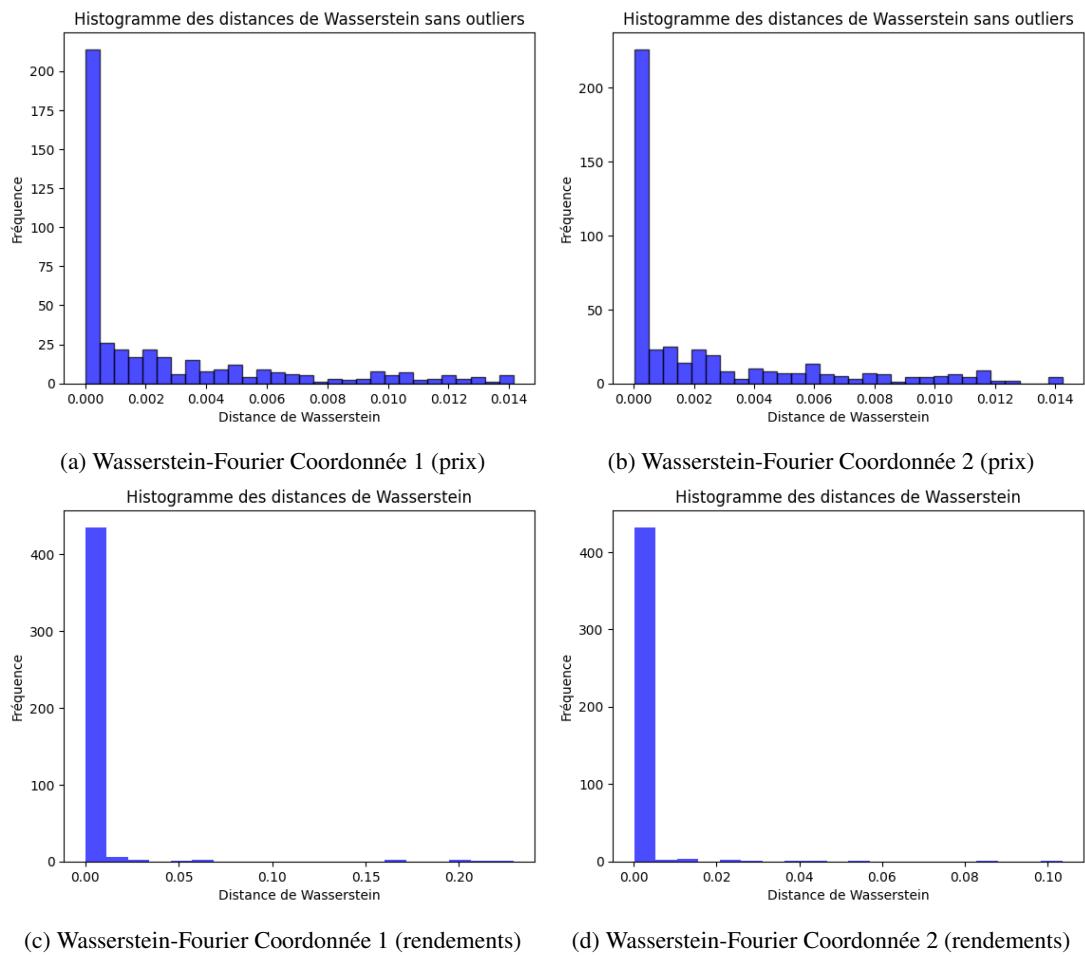


FIGURE 80 – Distance de Wasserstein-Fourier pour chaque coordonnée (prix et rendements)

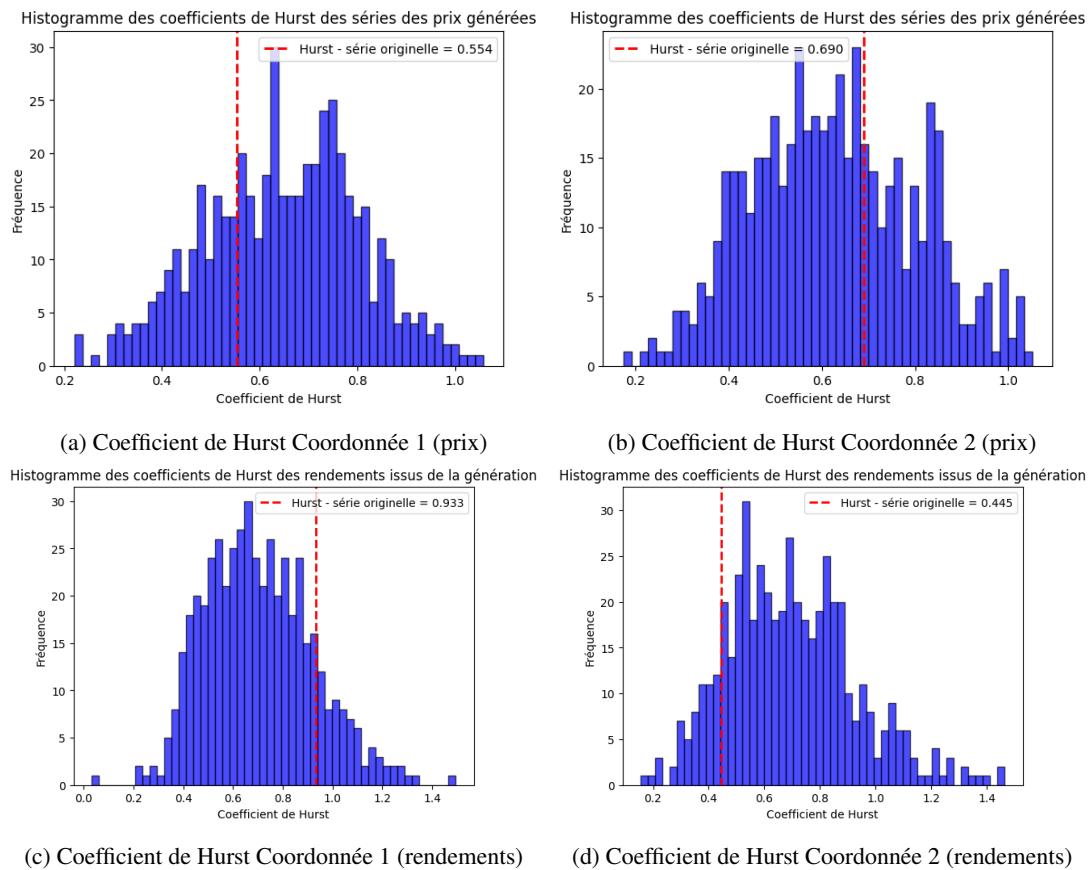


FIGURE 81 – Coefficient de Hurst pour les prix et rendements

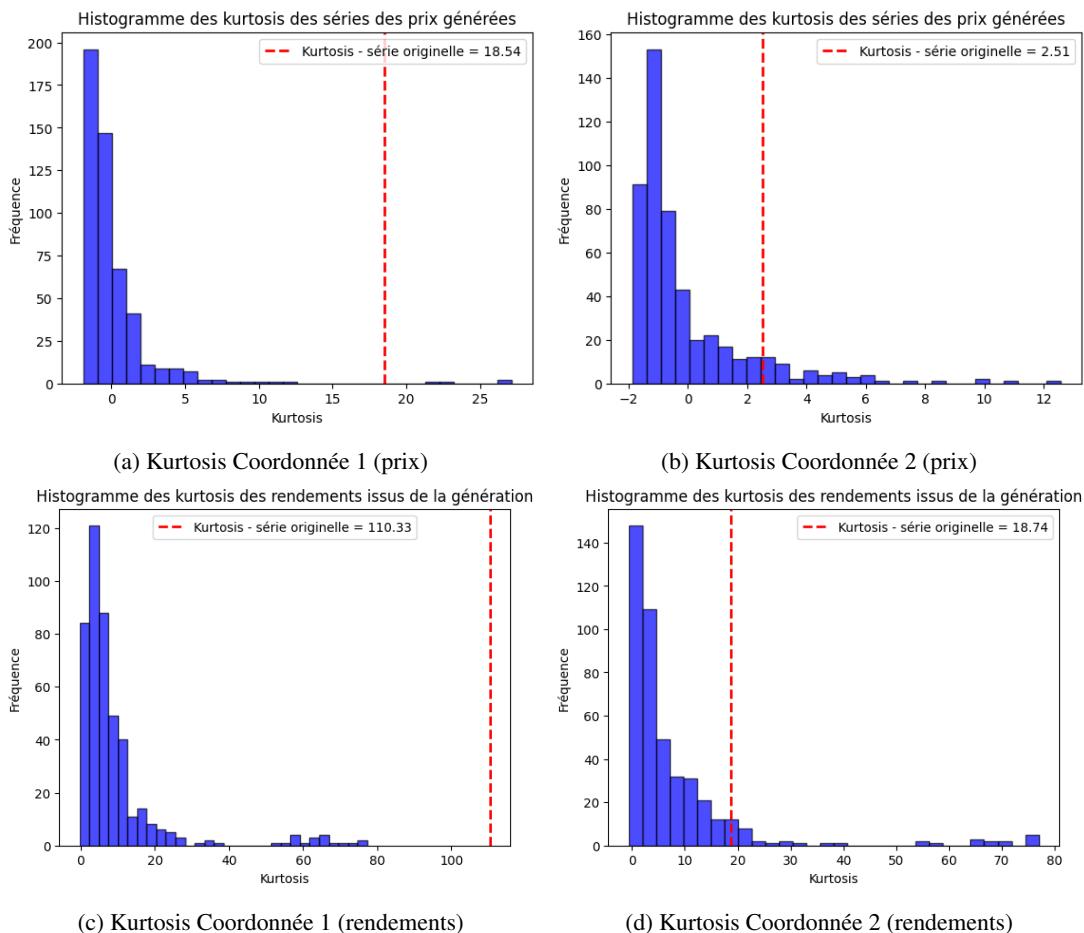


FIGURE 82 – Coefficient de Kurtosis pour les prix et rendements

Pour le S&P500 :

Mesure	Série	Coordonnée	Moyenne	Ecart-type	Intervalle de confiance à 95%	Nombre d'outliers	Intervalle de confiance à 95% sans outliers
Distance L2	Prix	1	12.365	5.754	[1.087, 23.642]	0	
		2	11.500	5.152	[1.401, 21.598]	0	
	Rendements	1	0.417	0.431	[-0.426, 1.260]	57	[-0.426, 0.372]
		2	0.380	0.299	[-0.204, 0.965]	91	[-0.204, 0.306]
Divergence KL - Fourier	Prix	1	0.0018	0.0077	[-0.0132, 0.0168]	100	[-0.0006 × 10 ⁻¹ , 0.0004]
		2	0.0018	0.0067	[-0.0113, 0.0149]	93	[-0.0006 × 10 ⁻¹ , 0.0003]
	Rendements	1	0.012	0.003	[0.006, 0.017]	0	
		2	0.012	0.003	[0.007, 0.017]	0	
Distance Wasserstein - Fourier	Prix	1	0.0074	0.0036	[0.0003, 0.0145]	33	[0.0027, 0.011]
		2	0.0037	0.0032	[-0.0026, 0.0101]	131	[0.0017, 0.0032]
	Rendements	1	0.0043	0.0198	[-0.0344, 0.0431]	0	
		2	0.0023	0.0036	[-0.0049, 0.0094]	0	

FIGURE 83 – Mesures

Coefficient	Série	Coordonnée	Moyenne	Ecart-type	Nombre de coefficients en dessous du seuil caractéristique	Nombre de coefficients au-dessus du seuil caractéristique
Coefficient de Hurst (seuil = 0.5)	Prix	1	0.706	0.209	78	414
		2	0.707	0.193	74	426
	Rendements	1	0.686	0.212	95	405
		2	0.602	0.205	167	333
Kurtosis (seuil = 0)	Prix	1	-0.887	0.613	454	46
		2	-0.871	0.577	462	38
	Rendements	1	2.970	4.576	34	400
		2	3.557	5.269	28	407

FIGURE 84 – Coefficients de Kurtosis et de Hurst

Corrélation de Pearson

Série des prix

Relation	Nombre	Pourcentage significatif (%)
Faible négative	275	68.36
Faible positive	95	27.37
Modérée négative	0	0
Modérée positive	53	100.00
Forte négative	0	0
Forte positive	28	100.00
Très forte négative	0	0
Très forte positive	49	100.00

Corrélation de Spearman

Série des prix

Relation	Nombre	Pourcentage significatif (%)
Faible négative	285	63.51
Faible positive	91	31.87
Modérée négative	0	0
Modérée positive	42	100.00
Forte négative	0	0
Forte positive	34	100.00
Très forte négative	0	0
Très forte positive	48	100.00

Série des rendements

Relation	Nombre	Pourcentage significatif (%)
Faible négative	291	11.34
Faible positive	209	4.78
Modérée négative	0	0
Modérée positive	0	0
Forte négative	0	0
Forte positive	0	0
Très forte négative	0	0
Très forte positive	0	0

Série des rendements

Relation	Nombre	Pourcentage significatif (%)
Faible négative	292	6.16
Faible positive	208	3.85
Modérée négative	0	0
Modérée positive	0	0
Forte négative	0	0
Forte positive	0	0
Très forte négative	0	0
Très forte positive	0	0

FIGURE 85 – Corrélations

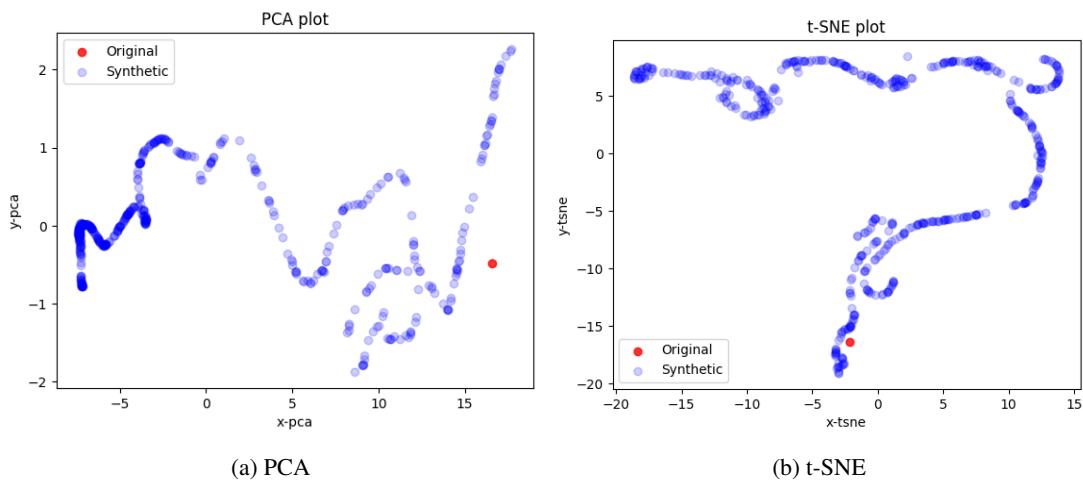


FIGURE 86 – Corrélation et visualisations en dimension réduite (PCA, t-SNE)

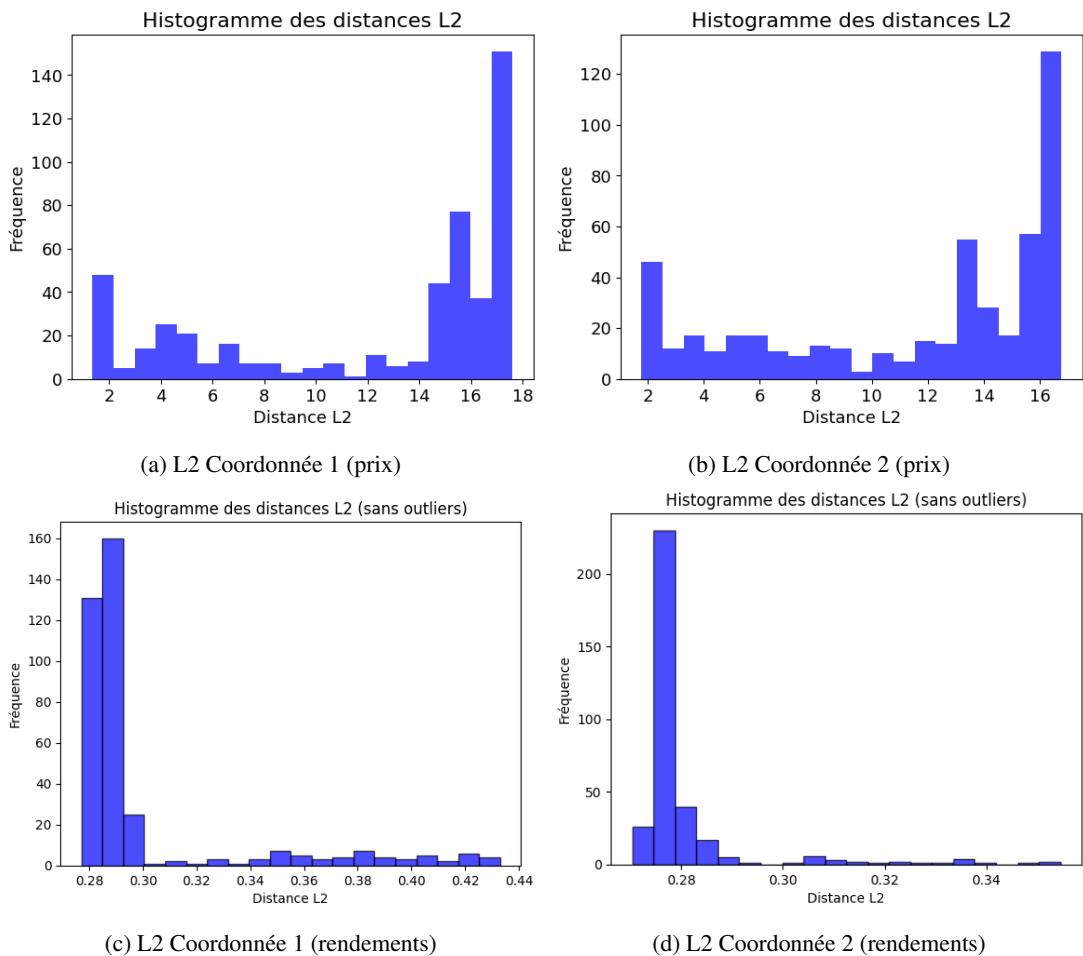


FIGURE 87 – Histogrammes de la distance L2 pour chaque coordonnée (prix et rendement)

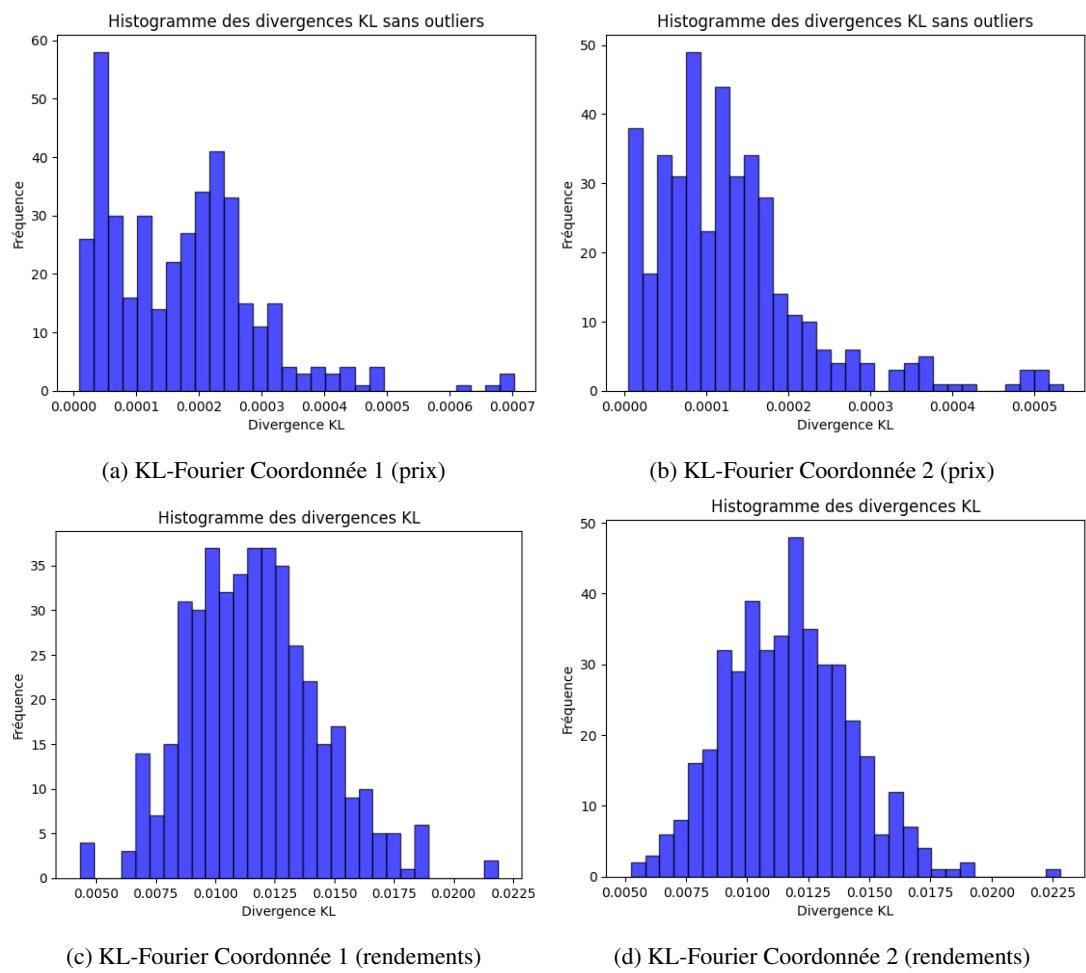


FIGURE 88 – Divergence KL-Fourier pour chaque coordonnée (prix et rendements)

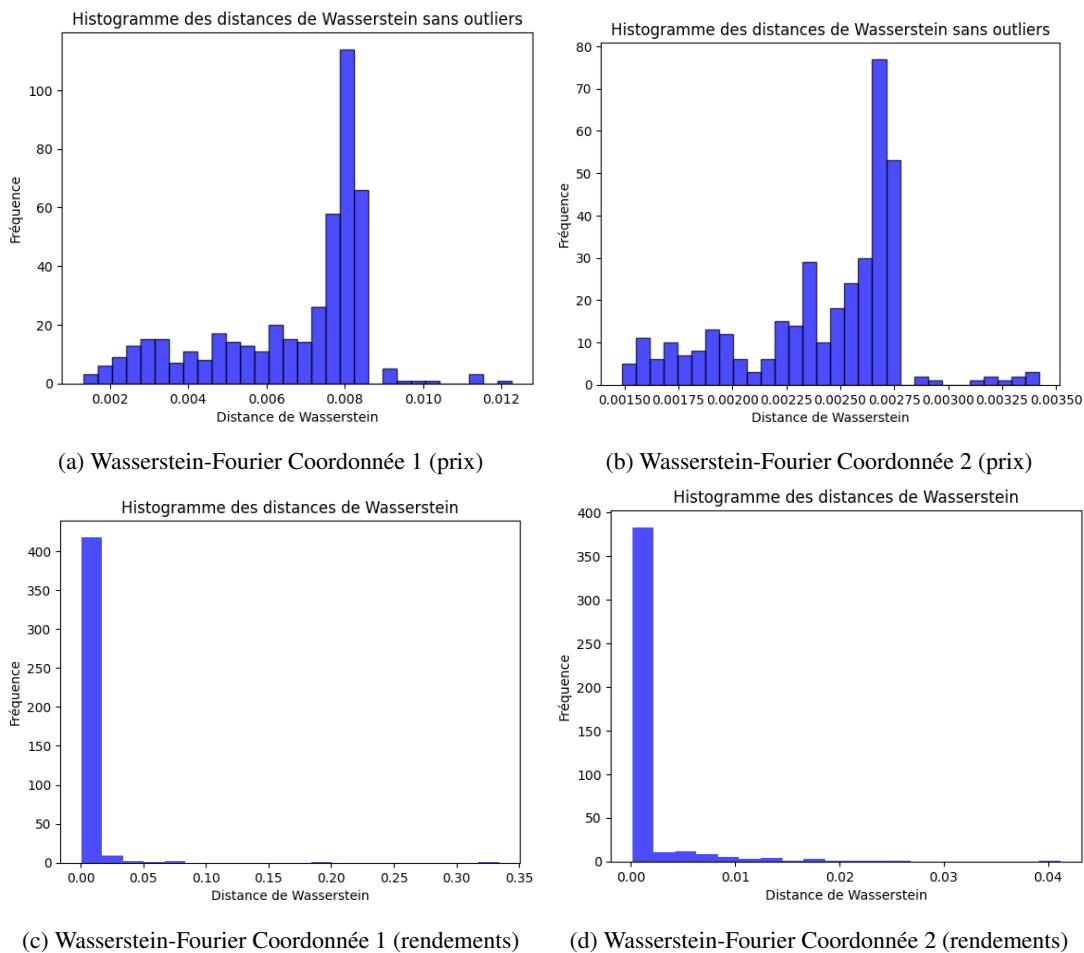


FIGURE 89 – Distance de Wasserstein-Fourier pour chaque coordonnée (prix et rendements)

Remarque 7. Pour le calcul du coefficient de Hurst :

- Pour les prix, on normalise de $[-1, 1]$ à $[0, 1]$ les valeurs.
- Pour les rendements, on impute les NaN par la moyenne des séries, puis normalisation dans $[0, 1]$ (pas plus de 4 NaN par série).

Remarque 8. Les valeurs des seuils pour établir les forces des corrélations sont : 0.3, 0.5, 0.7, 1

Remarque 9. Les éléments considérés comme des outliers sont ceux qui sont en dehors de la borne de confiance $[\min, \max]$ calculée de la manière suivante :

$$\text{— } \min = q_{25} - 1.5 \times (q_{75} - q_{25})$$

$$\text{— } \max = q_{75} + 1.5 \times (q_{75} - q_{25})$$

où q_i est le quantile d'ordre i calculé empiriquement pour les données considérées. Cette méthode est couramment utilisée pour détecter les valeurs aberrantes dans les distributions asymétriques, car elle repose sur l'écart interquartile, qui est moins sensible aux valeurs extrêmes que la moyenne ou l'écart-type.

6.15.4 Méthode 4

Pour le bitcoin :



FIGURE 90 – Comparaison des prix réels et simulés

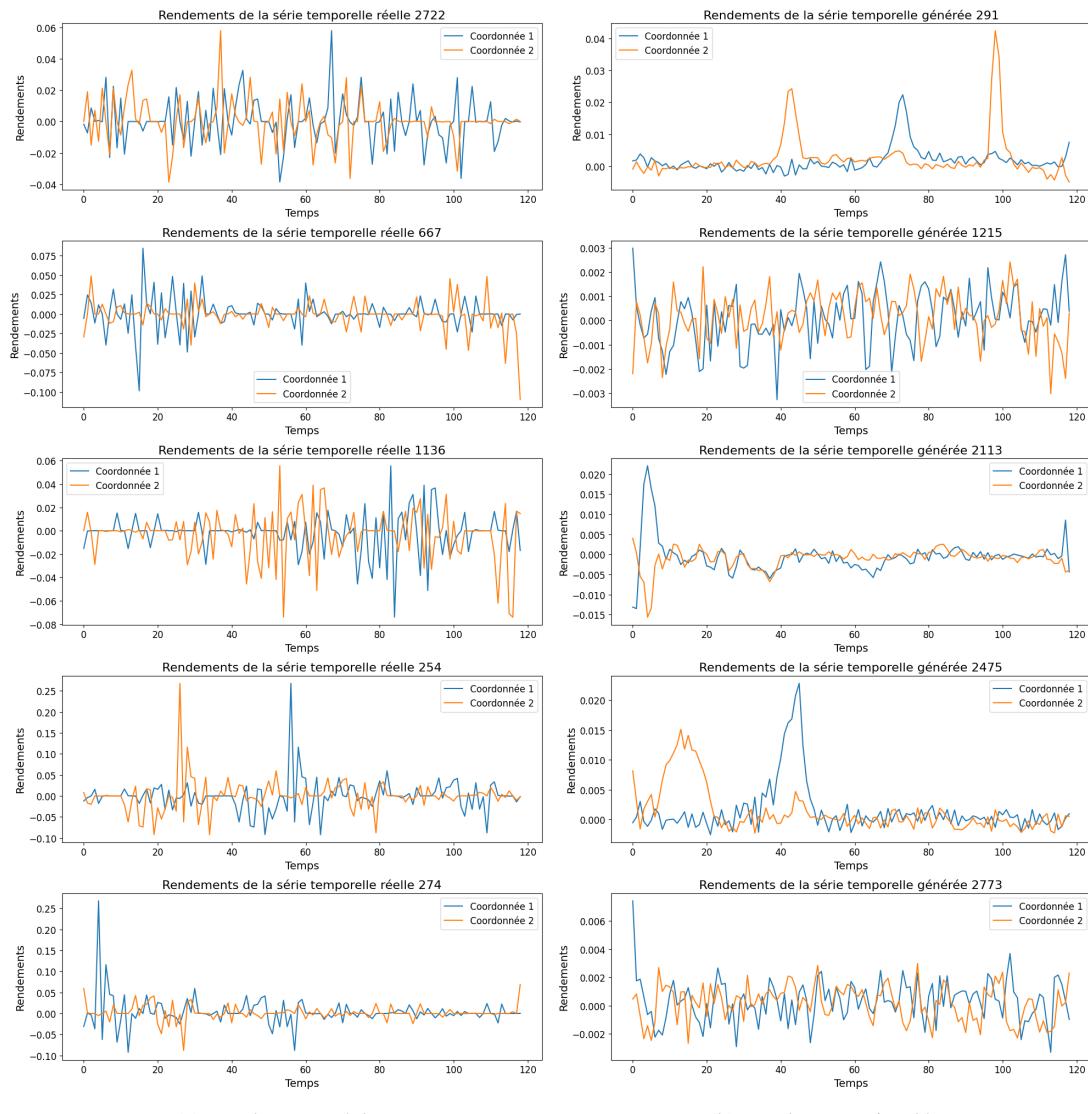


FIGURE 91 – Comparaison des rendements réels et simulés

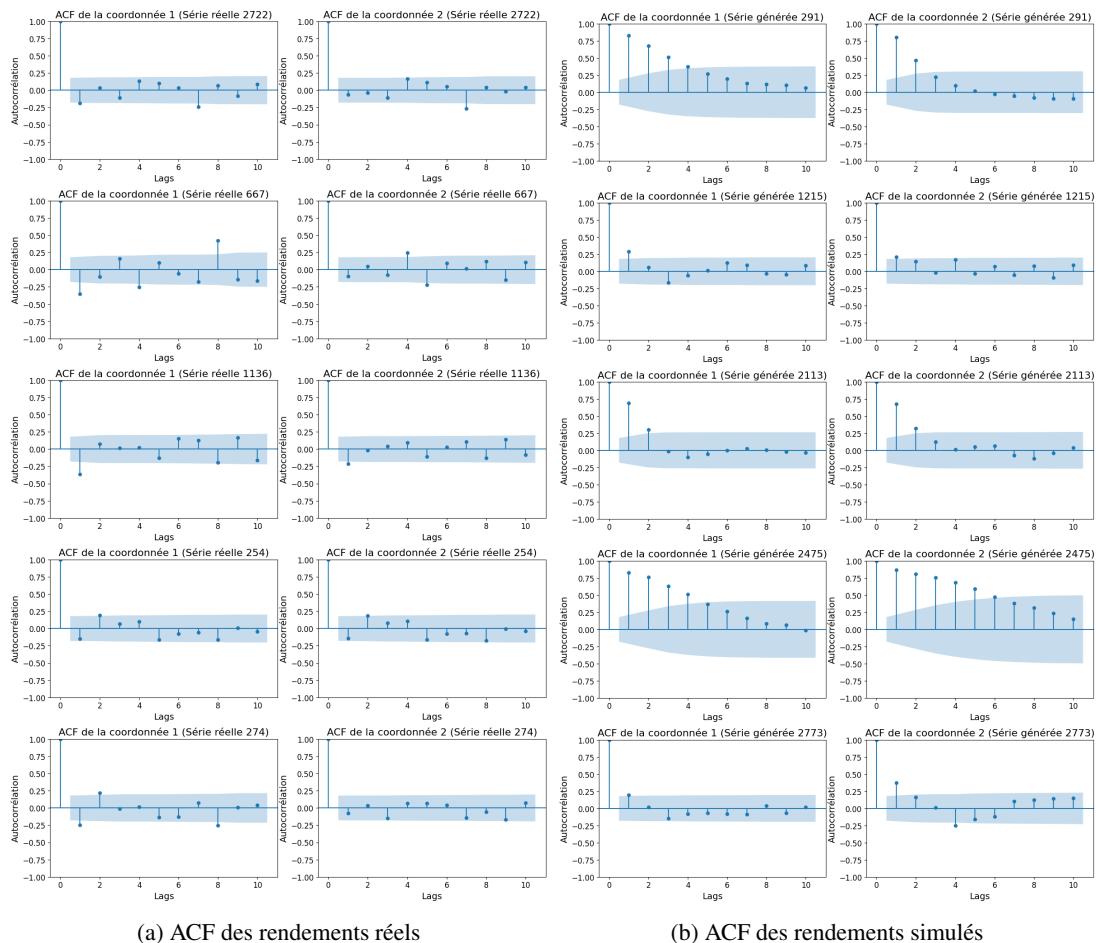


FIGURE 92 – Fonction d'autocorrélation

Traitement des données avant technique de visualisation

$$\begin{array}{l}
 f = 3 \\
 \left(\begin{array}{cccc} 0.3 & 0.7 & -0.4 \\ -0.1 & 0.2 & -0.7 \\ 0.5 & 0.9 & 0.6 \\ -0.9 & -0.2 & 0.7 \end{array} \right) \rightarrow \left(\begin{array}{c} 0.2 \\ -0.2 \\ 0.7 \\ -0.1 \end{array} \right) \rightarrow (0.2 \quad -0.2 \quad 0.7 \quad -0.1) \\
 n=1
 \end{array}$$

$$\begin{array}{l}
 l = 4 \\
 \cdot \\
 \cdot \\
 \cdot \\
 \cdot
 \end{array}$$

$$\left\{ \begin{array}{c} (0.2 \quad -0.2 \quad 0.7 \quad -0.1) \\ \cdots \quad \cdots \quad \cdots \quad \cdots \\ (0.3 \quad -0.1 \quad -0.1 \quad 0.2) \end{array} \right\} \quad N \text{ lignes}$$

$$\begin{array}{l}
 n=N \\
 \left(\begin{array}{cccc} 0.2 & 0.4 & 0.4 \\ 0.5 & -0.2 & -0.5 \\ -0.5 & 0.7 & -0.4 \\ 0.8 & 0.1 & -0.2 \end{array} \right) \rightarrow \left(\begin{array}{c} 0.3 \\ -0.1 \\ -0.1 \\ 0.2 \end{array} \right) \rightarrow (0.3 \quad -0.1 \quad -0.1 \quad 0.2) \\
 I = 4
 \end{array}$$

$$\underbrace{\left(\begin{array}{c} 0.2 \\ \cdots \\ 0.3 \end{array} \right)}_{I \text{ colonnes}} \quad N \text{ lignes}$$

FIGURE 93 – Traitement des données avant technique de visualisation

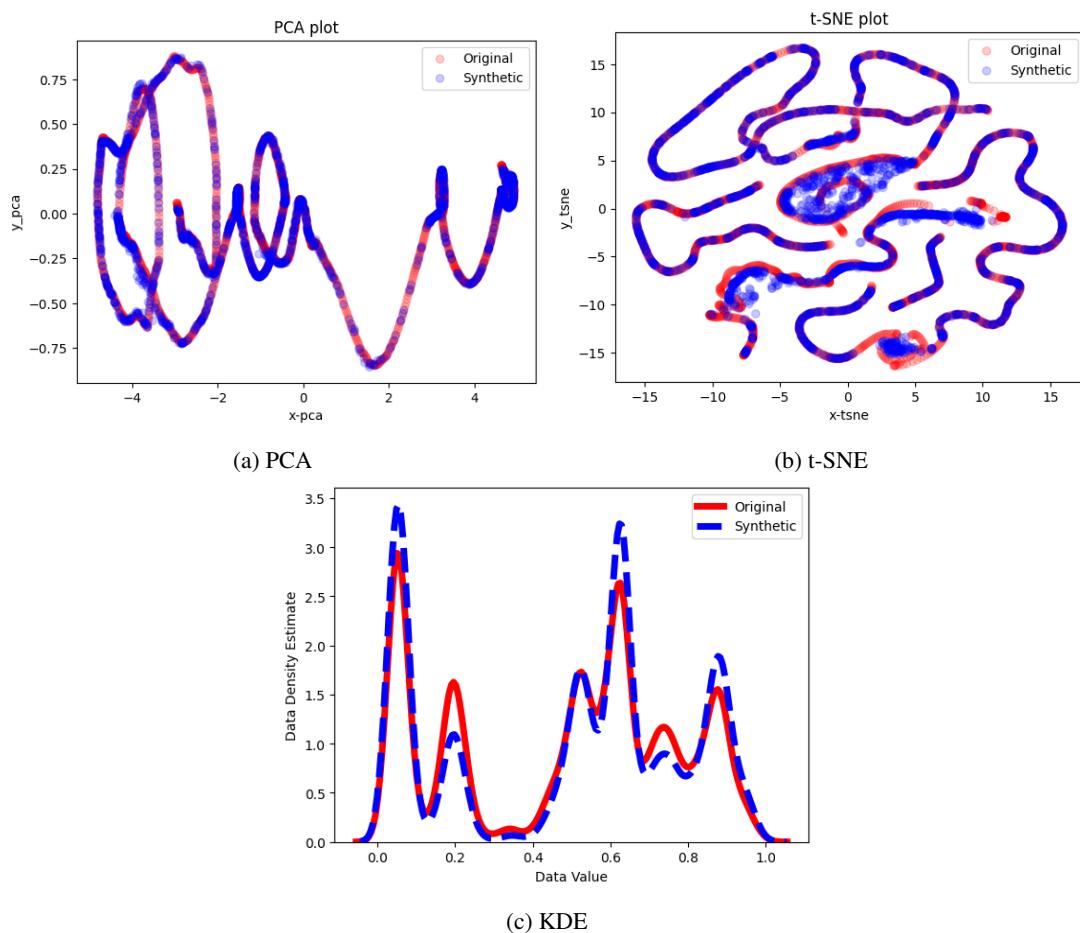


FIGURE 94 – Visualisations après réduction de dimension

Bitcoin					
Mesure	Série	Coordonnée	Moyenne	Ecart-type	Intervalle de confiance à 95%
Distance L2 entre les ACF	Rendements	1	0.718	0.261	
		2	0.745	0.294	
Divergence KL - Fourier	Prix	1	0.000458	0.00103	[0.000418, 0.000499]
		2	0.000443	0.00100	[0.000403, 0.000482]
Distance Wasserstein - Fourier	Rendements	1	0.0135	0.0060	[0.01323, 0.01370]
		2	0.0146	0.0066	[0.01431, 0.01482]
	Prix	1	0.00259	0.00352	[0.00245, 0.00275]
		2	0.00264	0.00361	[0.00250, 0.00278]
	Rendements	1	0.639	4.057	[0.48003, 0.79811]
		2	0.717	4.259	[0.54978, 0.88372]

FIGURE 95 – Mesures statistiques

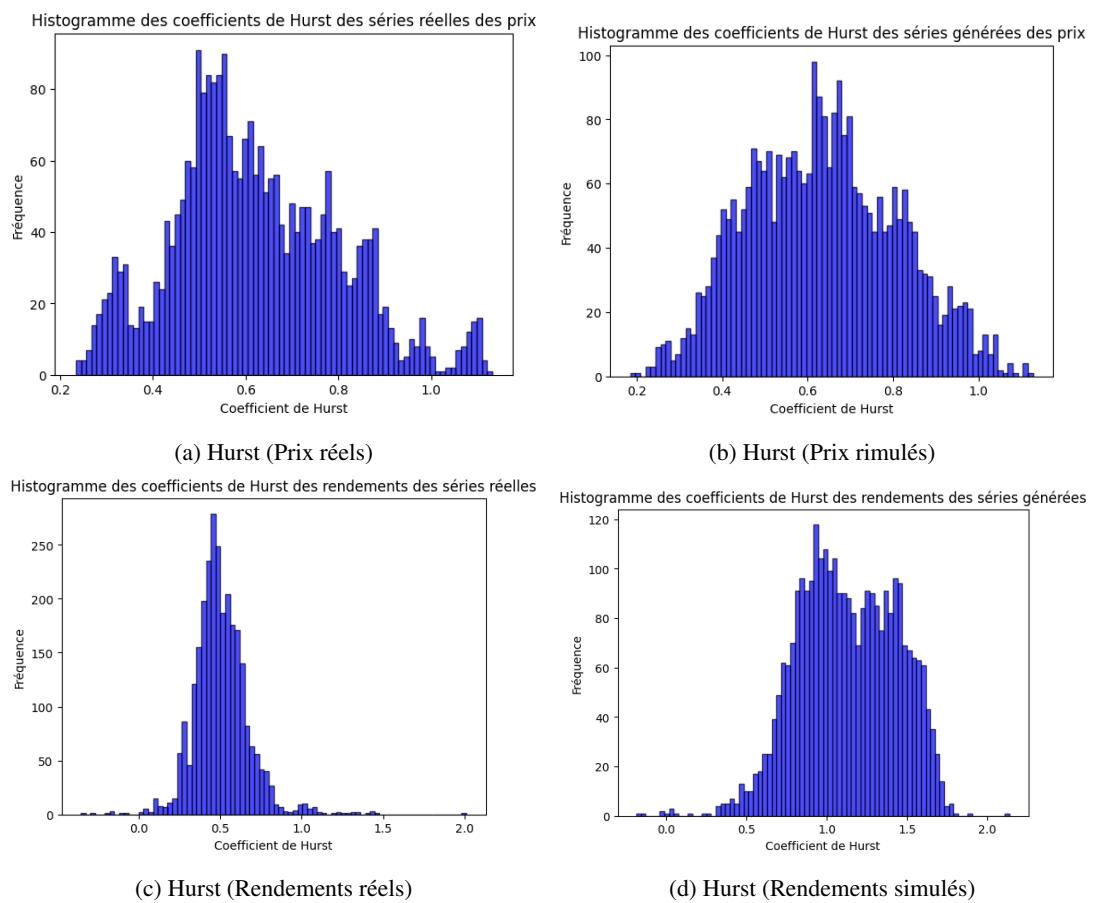


FIGURE 96 – Coefficients de Hurst

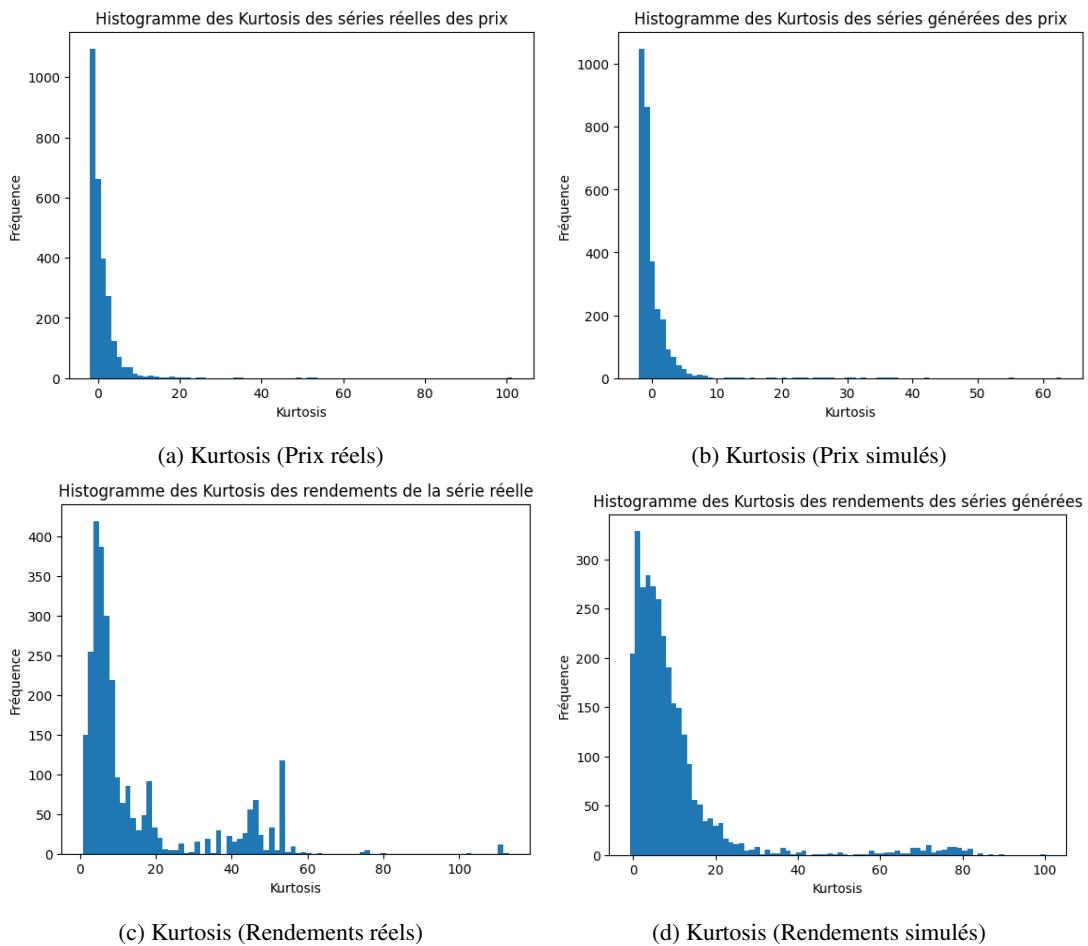


FIGURE 97 – Coefficients de Kurtosis

Pour le S&P500 :

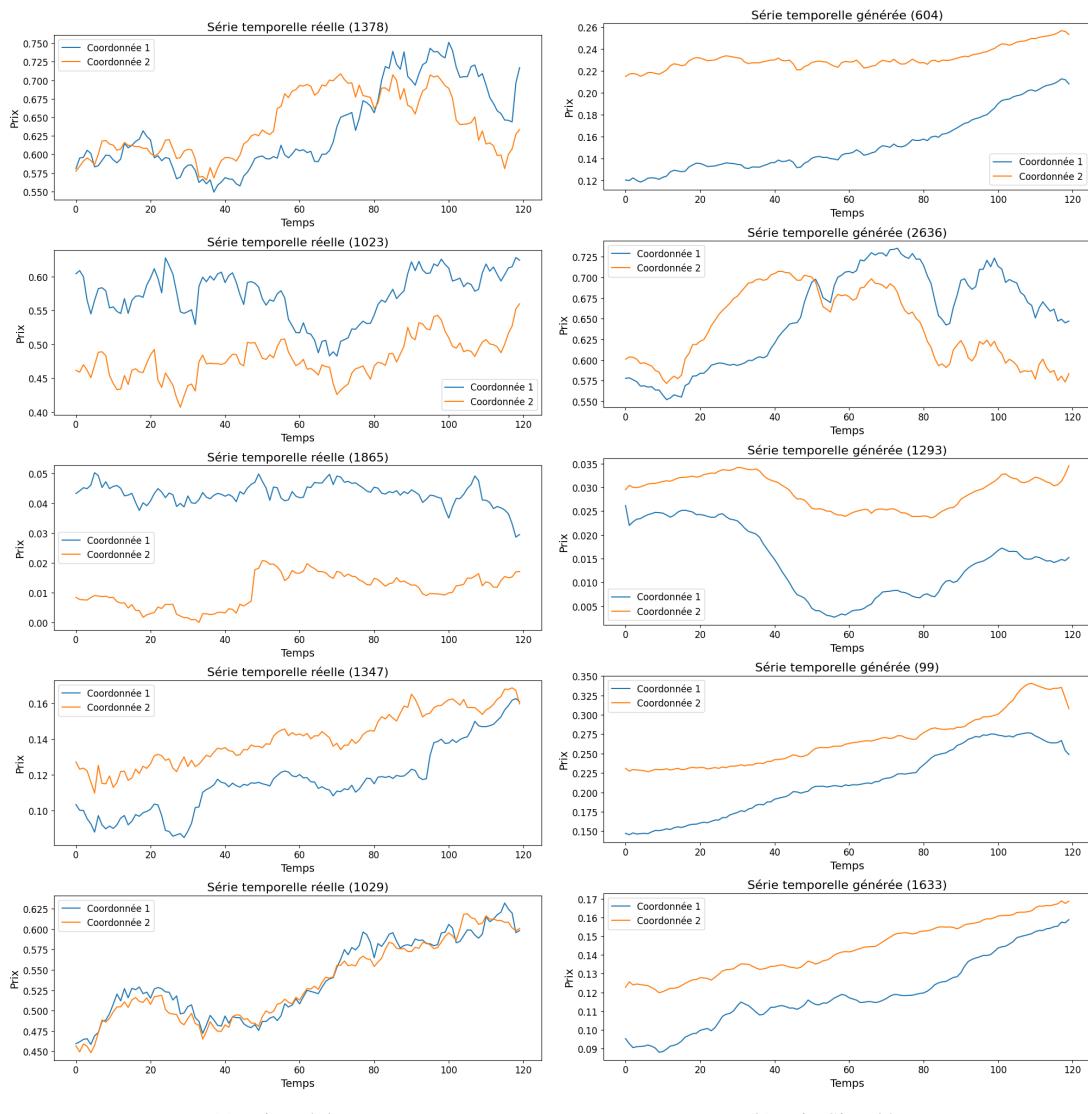


FIGURE 98 – Comparaison des prix réels et simulés

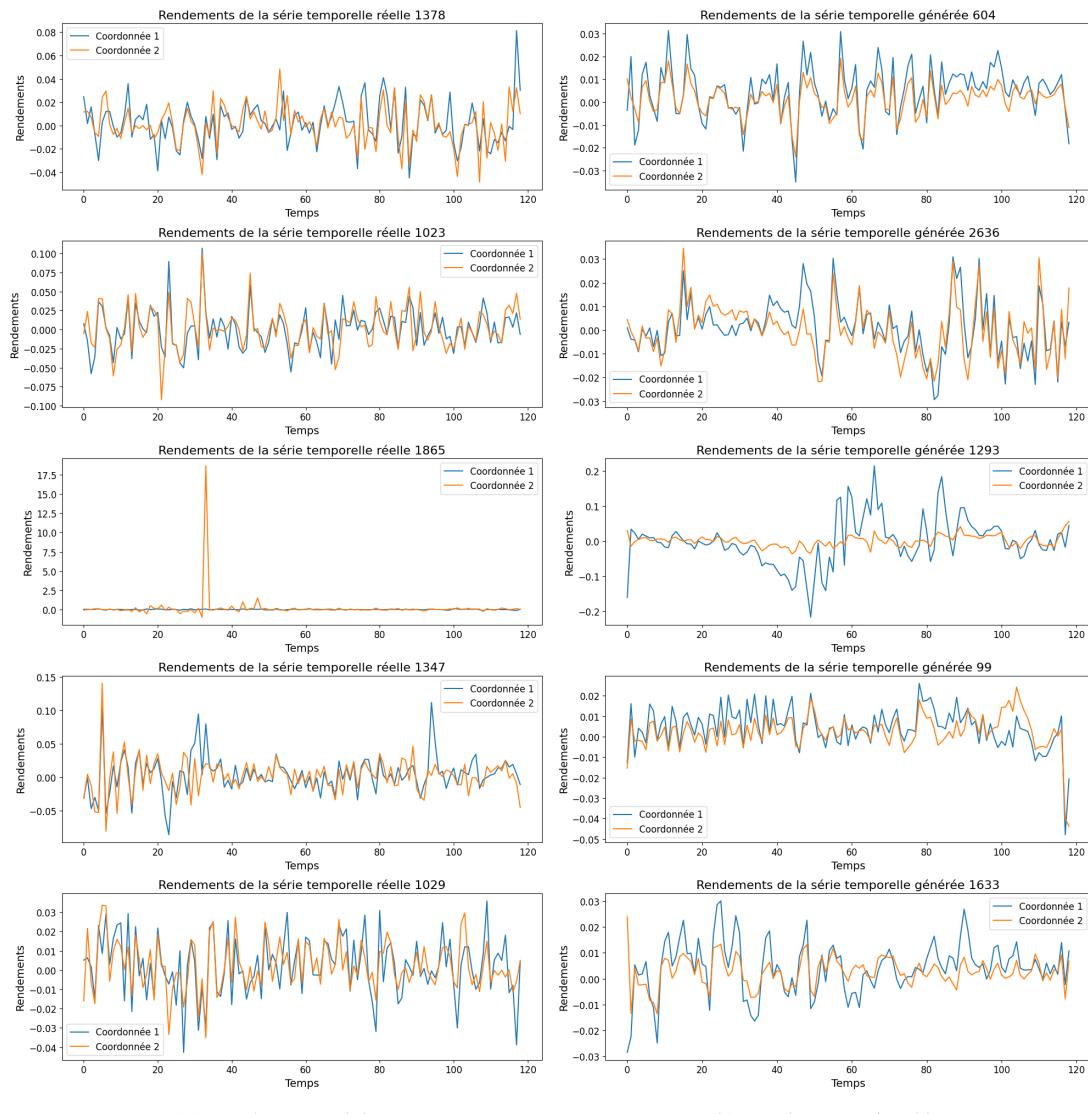
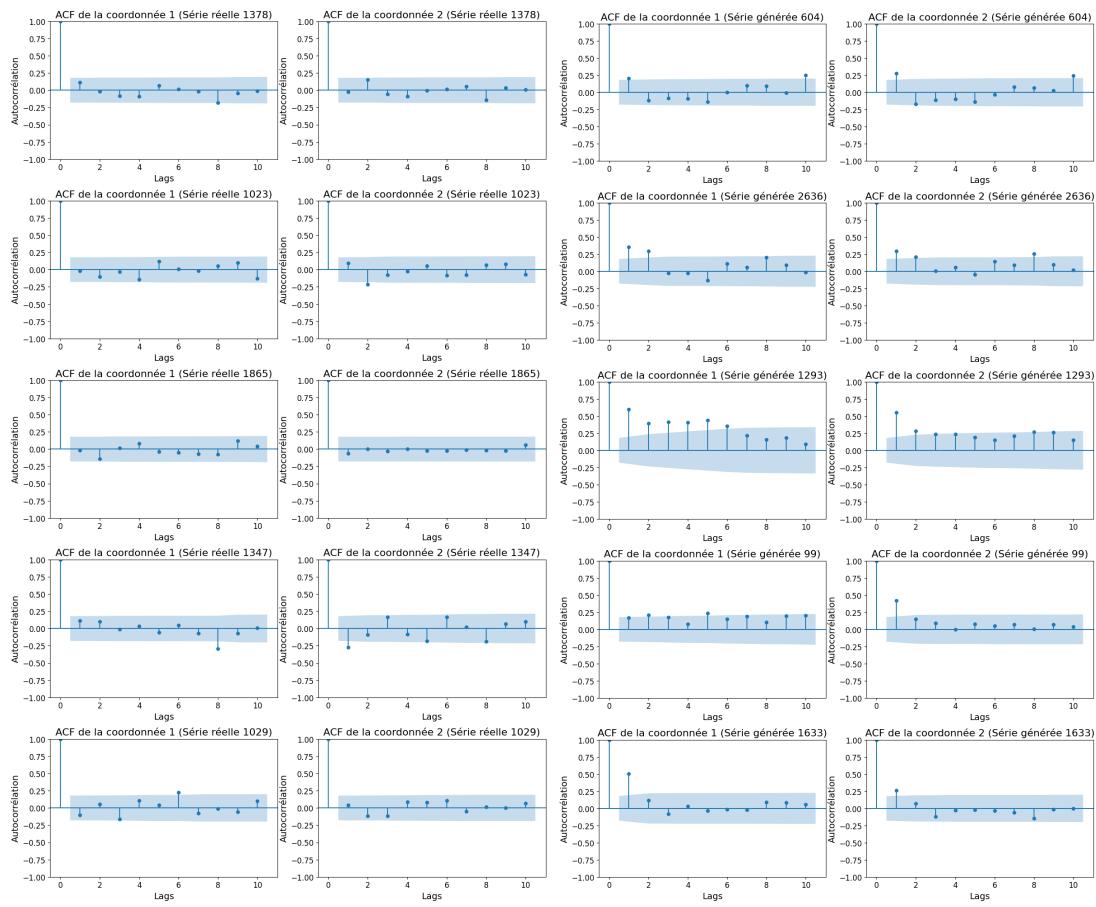


FIGURE 99 – Comparaison des rendements réels et simulés



(a) ACF des rendements réels

(b) ACF des rendements simulés

FIGURE 100 – Fonction d'autocorrélation

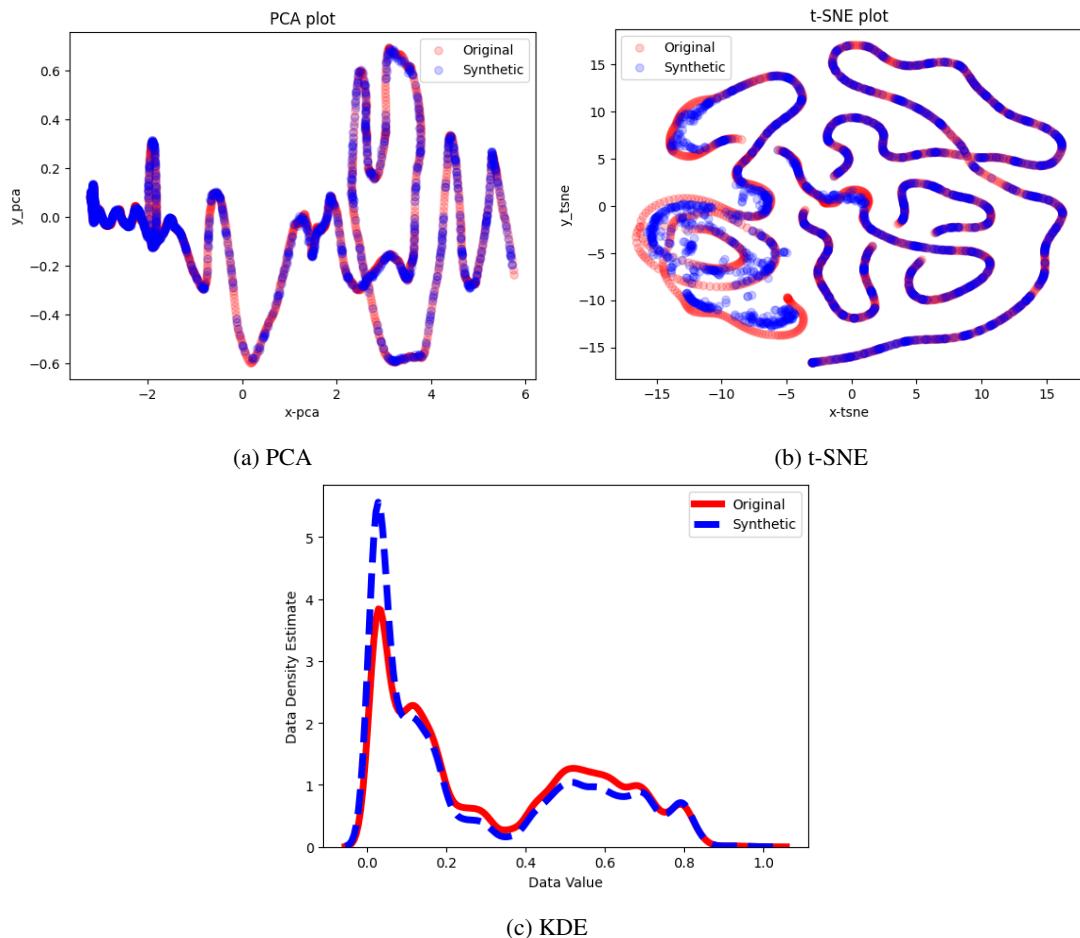


FIGURE 101 – Visualisations après réduction de dimension

SNP500					
Mesure	Série	Coordonnée	Moyenne	Ecart-type	Intervalle de confiance à 95%
Distance L2 entre les ACF	Rendements	1	0.841	0.382	
		2	0.727	0.296	
Divergence KL - Fourier	Prix	1	0.00088	0.00158	[0.00082, 0.00094]
		2	0.00048	0.00128	[0.00043, 0.00053]
Distance Wasserstein - Fourier	Rendements	1	0.01096	0.00320	[0.01083, 0.01108]
		2	0.01098	0.00329	[0.01086, 0.01111]
	Prix	1	0.00220	0.00199	[0.00212, 0.00228]
		2	0.00141	0.00130	[0.00136, 0.00147]
	Rendements	1	0.714	3.852	[0.563, 0.865]
		2	0.335	1.668	[0.270, 0.400]

FIGURE 102 – Mesures statistiques

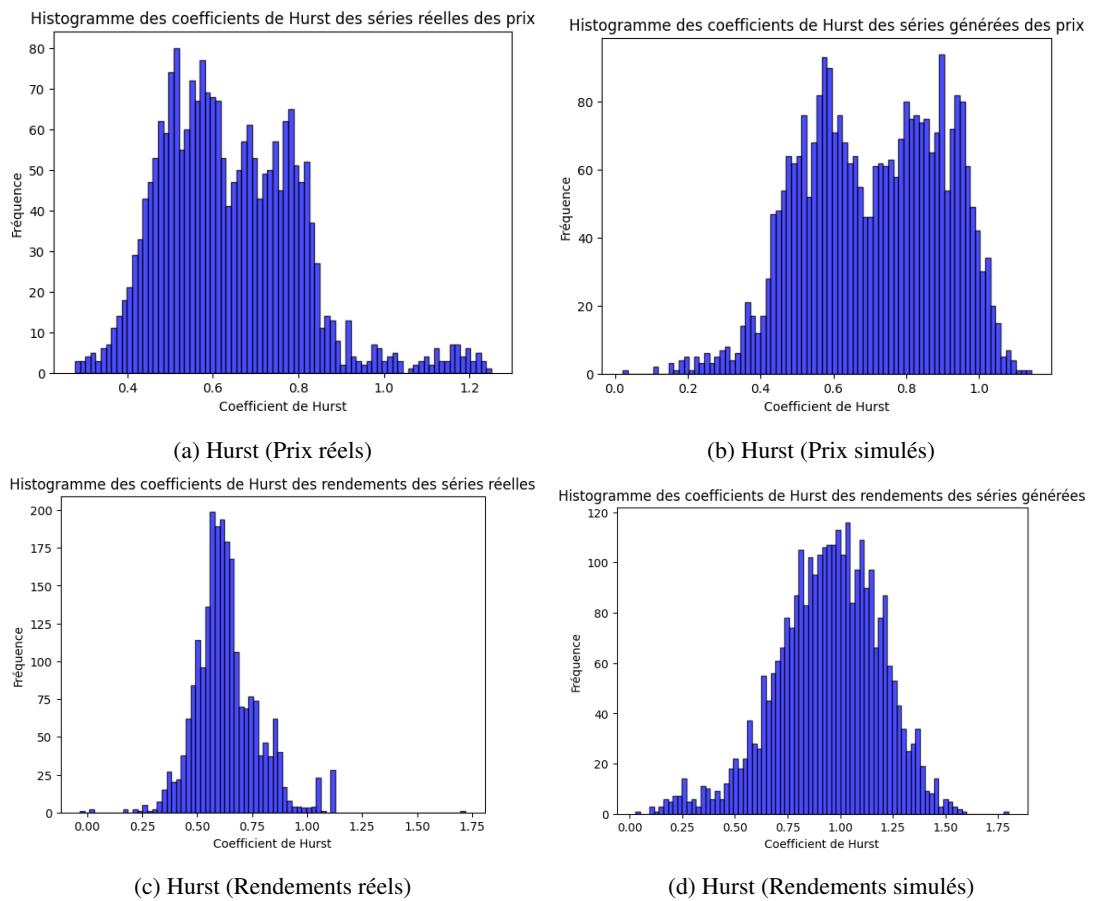


FIGURE 103 – Coefficients de Hurst

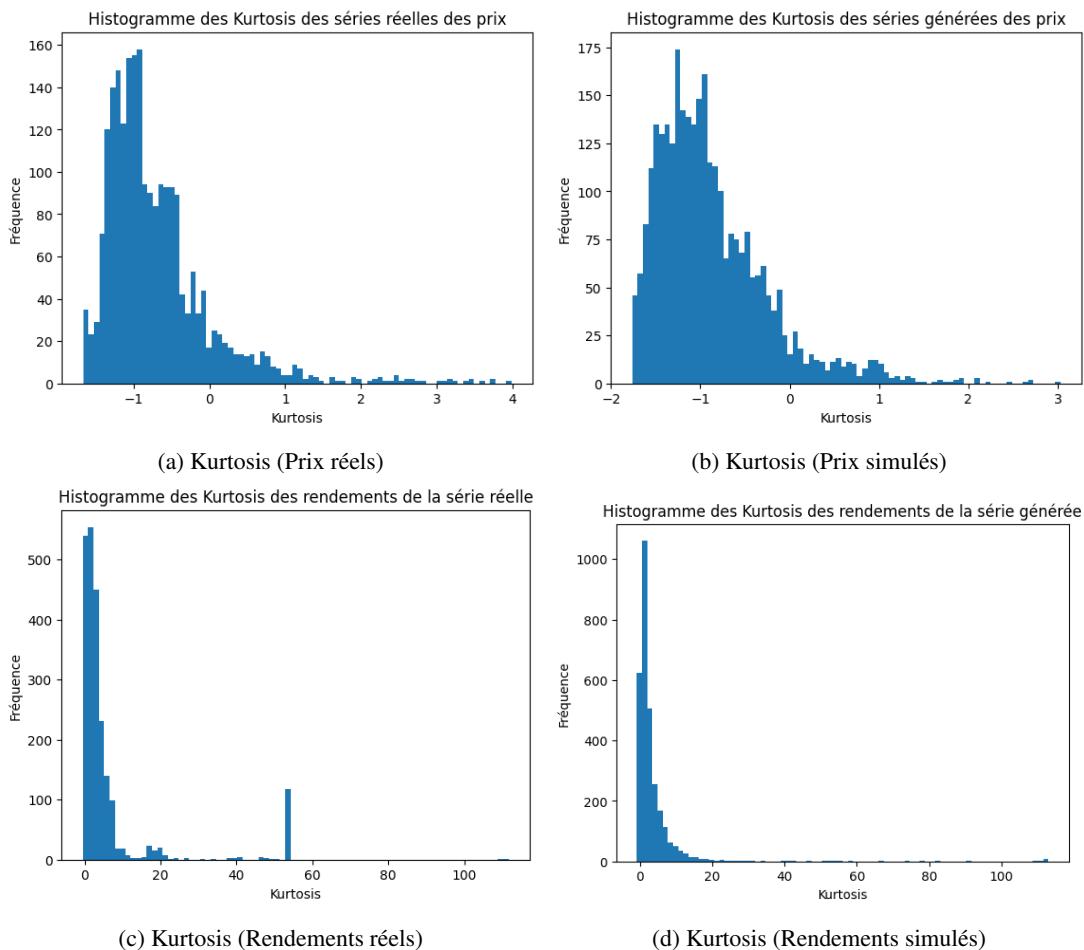


FIGURE 104 – Coefficients de Kurtosis

Remarque 10. Pour les distances de Wasserstein-Fourier et les divergences KL-Fourier, nous avons comparé 2500 séries choisies de manière aléatoire dans l'ensemble des séries originelles et tout autant parmi les séries générées (même résultats avec 5000 séries tirées).

Remarque 11. Les rendements calculés à partir des séries des prix générés présentaient quelques valeurs manquantes ou des *inf* et *sup* qui ont été respectivement imputées par la moyenne, le minimum et le maximum de la série considérée.

Remarque 12. Pour le calcul des coefficients de Hurst, 63 séries réelles des prix ont été ignorées pour le S&P500, ainsi que 120 séries des prix réels du Bitcoin.