

Pairs Trading with Wavelet Transform

Réplication de Eroglu, Yener & Yigit (2023)

Quantitative Finance, 23(7–8), 1129–1154

Yassine Mannai Antonin Bezard Issam Fradi

Université Paris-Dauphine — Master 272
Arbitrage Statistique

5 juin 2026

Plan

- 1 Le papier
- 2 Les ondelettes
- 3 Stratégie
- 4 Pipeline de réplication
- 5 Résultats principaux
- 6 Figures reproduites

Eroglu, Yener & Yigit (2023) — « Pairs trading with wavelet transform », *Quantitative Finance*.

- Le pairs trading classique repose sur un spread de prix qui mélange *tendance* (marche aléatoire) et *bruit haute fréquence*.
- Ce bruit gonfle la variance du spread, brouille les signaux d'entrée/sortie et déclenche des sorties prématurées.
- Idée : remplacer le prix brut par sa **composante lisse d'ondelettes** (MODWT niveau 1, symlet d'ordre 22 — sym22) avant de former le spread. La partie lisse conserve la dynamique lente et écarte la contamination haute fréquence.
- Résultats : 11,82 % (distance) et 9,66 % (cointégration) de rendement moyen par période contre $-0,55$ % et $-1,81$ % pour la variante standard. Sharpe : 3,69 / 2,82 contre $-0,21$ / $-0,40$.
- Univers : S&P 500, point-in-time, 05/03/2010–15/03/2018 (7 périodes annuelles, cf. Table 1 du papier), 415 tickers.

Qu'est-ce qu'une ondelette ?

- Une **ondelette** ψ est une petite onde *localisée* (moyenne nulle, énergie finie) que l'on dilate (échelle a) et translate (position b) pour sonder un signal :

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right).$$

- **Mécanisme** : corrélér le signal avec $\psi_{a,b}$ mesure son contenu autour de l'instant b à l'échelle a . Petite échelle $\Rightarrow$ détails rapides (haute fréquence) ; grande échelle $\Rightarrow$ tendance lente (basse fréquence).
- **Vs Fourier** : les sinus de Fourier sont globaux (aucune localisation temporelle). Les ondelettes sont localisées en *temps ET en fréquence* — idéales pour des séries financières *non stationnaires*.
- **Intérêt en finance** : séparer la tendance (marche aléatoire du prix) du bruit micro-structure haute fréquence ; débruitage, analyse multi-résolution, sans imposer de modèle paramétrique.

Familles d'ondelettes

Haar	La plus simple (support 2), discontinue — mauvaise pour des prix lisses.
Daubechies (dbN)	Support compact $2N$, N moments nuls, mais fortement <i>asymétriques</i> (déphasage).
Symlets (symN)	Daubechies « presque symétriques » : déphasage minimal $\Rightarrow$ le prix filtré reste aligné.
Coiflets	Moments nuls aussi sur la fonction d'échelle ; support plus long.
Meyer / Morlet	Très lisses mais à support <i>infini</i> (Morlet est continue).

- Compromis : longueur du support L vs régularité vs symétrie.
- Le papier choisit un **symlet** : quasi-symétrie $\Rightarrow$ le prix lissé n'est pas décalé dans le temps par rapport au prix brut (essentiel pour des signaux d'entrée/sortie datés au jour près).

DWT vs MODWT : pourquoi discret et non-décimé

Pourquoi *discret* (DWT/MODWT) plutôt que continu (CWT)

La CWT balaie un continuum d'échelles : redondante et coûteuse. Les prix sont échantillonnés quotidiennement $\Rightarrow$ une transformée à échelles dyadiques discrètes (2^j) suffit, est rapide et inversible.

Pourquoi *MODWT* plutôt que DWT

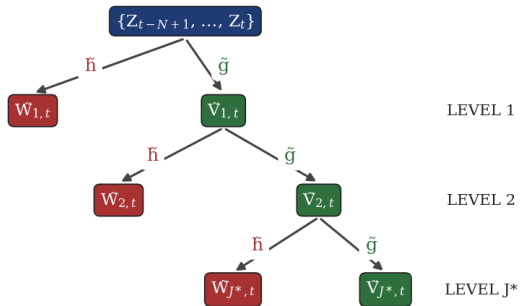
- **DWT** : décimée (sous-échantillonnage $\div 2$ par niveau) $\Rightarrow$ la longueur est divisée par 2, exige $T = 2^j$, et *n'est pas* invariante par translation (le résultat dépend du point de départ).
- **MODWT** : *non décimée* $\Rightarrow$ garde la longueur T à chaque niveau, fonctionne pour tout T , est invariante par translation et **aligne exactement** la série lissée sur les jours de trading.

Pourquoi le symlet d'ordre 22 (sym22)

- Le papier retient le **symlet d'ordre 22** : filtre à $L = 44$ coefficients.
- **Support long** $\Rightarrow$ passe-bas très régulier : capte finement la tendance lente et rejette fortement le bruit.
- **Quasi-symétrie** $\Rightarrow$ distorsion de phase minimale : le prix filtré n'est ni avancé ni retardé par rapport au prix brut.
- **Robustesse** : la Table 16 balaie sym16/sym18/sym20/sym22/db8 — les résultats varient peu, le choix précis de l'ondelette n'est pas critique.
- *Note de réplcation* : PyWavelets ne fournit pas les coefficients exacts du sym22 ; notre code retombe sur **sym20** ($L = 40$), un voisin immédiat couvert par la Table 16.

Figure 1 — Algorithme pyramidal de Mallat

Figure 1. Pyramid algorithm of Mallat (3-level MODWT decomposition).



Cascade de filtres : à chaque niveau, le passe-bas $\tilde{g}$ extrait la lissée $\tilde{V}_j$ (tendance) et le passe-haut $\tilde{h}$ les détails $\tilde{W}_j$. On ne garde que $\tilde{V}_1$.

- L'historique est découpé en blocs consécutifs de 252 jours ouvrés.
- Pour chaque période n :
 - formation = bloc k (in-sample),
 - trading = bloc $k + 1$ (out-of-sample).
- La sélection des paires se fait **uniquement** sur le bloc de formation, à partir de l'univers survivant à cette date (aucun look-ahead bias).

Distance	20 paires minimisant $\sum_t (\tilde{P}_i - \tilde{P}_j)^2$ sur les prix normalisés.
Cointégration	Test de <i>Johansen</i> (trace, 5 %) sur $(\tilde{P}_i, \tilde{P}_j)$, après pré-filtre par distance.

Sélection des paires : distance & cointégration

Prix normalisés sur le bloc de formation : $\tilde{P}_{k,t} = P_{k,t}/P_{k,0}$.

Méthode 1 — Distance minimale (Gatev et al. 2006)

Pour chaque paire (i, j) , somme des écarts quadratiques :

$$d_{ij} = \frac{1}{T} \sum_{t=1}^T (\tilde{P}_{i,t} - \tilde{P}_{j,t})^2.$$

On retient les $N = 20$ paires de plus faible d_{ij} (doublons de cotation $d_{ij} \approx 0$ exclus).

Méthode 2 — Cointégration (test de Johansen)

On teste l'existence d'une relation de long terme via la statistique de **trace** de Johansen sur le vecteur $(\tilde{P}_{i,t}, \tilde{P}_{j,t})$:

$$\text{LR}_{\text{trace}}(r=0) > 15,4943 \text{ (valeur critique à 5\%)} \Rightarrow \text{paire cointégrée.}$$

Pré-filtrage par distance (les 2000 paires les plus proches) pour limiter le coût de calcul.

L'étape ondelettes

- Chaque série de prix est rééchelonnée, $\tilde{P}_t = P_t/P_0$.
- MODWT niveau 1 avec le **symlet d'ordre 22 (sym22)**, condition aux bords périodique sur le bloc de trading (fidèle au papier, section 4.4). *Notre réplication retombe sur sym20* (coefficients sym22 non fournis par PyWavelets).
- On conserve les **coefficients lisses** (passe-bas), on jette les détails (passe-haut), puis on reconstitue le prix synthétique.
- Le spread de la paire filtrée par ondelettes est alors

$$s_t = \tilde{P}_i^{\text{wav}} - \alpha - \beta \tilde{P}_j^{\text{wav}},$$

avec (α, β) estimés par MCO sur le bloc de formation.

- La variante standard utilise la même formule *sans* l'étape MODWT — c'est la seule différence entre les courbes de chaque figure.

Définition mathématique du filtre MODWT

La **MODWT** (*Maximal Overlap Discrete Wavelet Transform*) au niveau $j = 1$ décompose une série $Z = (Z_0, \dots, Z_{T-1})$ en :

Détails	$\widetilde{W}_{1,t} = \sum_{\ell=0}^{L_1-1} \tilde{h}_\ell Z_{t-\ell \bmod T},$
Lissée	$\widetilde{V}_{1,t} = \sum_{\ell=0}^{L_1-1} \tilde{g}_\ell Z_{t-\ell \bmod T},$

où $\tilde{h}_\ell = h_\ell/\sqrt{2}$ (passe-haut) et $\tilde{g}_\ell = g_\ell/\sqrt{2}$ (passe-bas) sont les filtres orthogonaux de la famille **symlet 22** ($L_1 = 44$ coefficients), normalisés pour conserver l'énergie ($L_1 = 40$ pour le sym20 de notre réplique). Bord traité en périodisant Z .

Propriétés clés pour le pairs trading :

- $\widetilde{V}_{1,t}$ retient les dynamiques *lentes* (≥ 2 jours ouvrés) ; $\widetilde{W}_{1,t}$ (haute fréquence) est écartée.
- **Non-décimé** : série lissée de même longueur T — pas de bord mort, alignement temporel exact.
- Reconstruction parfaite : $Z_t = \widetilde{V}_{1,t} + \widetilde{W}_{1,t}$.
- Invariance par translation — pas de dépendance au point de départ (contrairement à la DWT classique).

Algorithme du pairs trading par ondelettes

Bloc de formation (T_{form} jours)

1. Normaliser les log-prix : $\tilde{P}_t^{(k)} = P_t^{(k)} / P_0^{(k)}$.
2. **Filtrer** : $\tilde{V}_{1,t}^{(k)} \leftarrow \text{MODWT-smooth}(\tilde{P}^{(k)}, \text{sym22})$ pour chaque actif k .
3. Sélectionner les $N = 20$ paires (i, j) minimisant $d_{ij} = T^{-1} \sum_t (\tilde{V}_{1,t}^{(i)} - \tilde{V}_{1,t}^{(j)})^2$.
4. Estimer par MCO : $(\hat{\alpha}_{ij}, \hat{\beta}_{ij}) = \arg \min \sum_t (\tilde{V}_{1,t}^{(i)} - \alpha - \beta \tilde{V}_{1,t}^{(j)})^2$.
5. Spread de formation : $s_t = \tilde{V}_{1,t}^{(i)} - \hat{\alpha}_{ij} - \hat{\beta}_{ij} \tilde{V}_{1,t}^{(j)} \Rightarrow$ calculer μ_s, σ_s .

Bloc de trading (T_{trade} jours)

6. **Re-filtrer** (rolling) : $\tilde{V}_{1,t}^{(k)}$ sur les T_{form} dernières observations.
7. z-score courant : $z_t = (s_t - \mu_s) / \sigma_s$.
8. **Entrée** si $|z_t| > 2$: vendre le jambage haut, acheter le jambage bas, en proportion 1 : $\hat{\beta}_{ij}$.
9. **Sortie** si z_t recroise 0 ou fin de bloc.

Le filtre $\tilde{V}_{1,t}$ est la **seule** modification par rapport au pairs trading classique.

Règle de trading

- Calculer σ du spread sur la fenêtre de formation.
- Entrer dès que $|s_t| > 2\sigma$ (vente du jambage haut, achat du jambage bas).
- Sortir quand s_t recroise zéro pour la première fois.
- Clôture forcée en fin de bloc de trading ; la variante **forced-close-only** de la Table 22 n'utilise aucune autre sortie.
- Coûts de transaction : coût par action appliqué à l'entrée et à la sortie. Nous balayons $tc \in \{0 ; 0,001 ; 0,005\}$ pour les Tables 18–21.

Architecture de la réplication

<code>research/main.py</code>	CLI unifiée. <code>-full</code> = run complet du papier, <code>-tc-sweep</code> = Tables 18–21.
<code>paper_replication/core/</code>	Moteur par période : periods, selection, spread, trading, wavelet, metrics.
<code>paper_replication/analytics/</code>	Régressions d'asset pricing (Fama-French, q-factor, ICAPM de Petkova).
<code>paper_replication/outputs/</code>	22 tables numérotées + 12 figures + résumé de validation.
<code>paper_replication/bootstrap/</code>	Récupération Bloomberg et rafraîchissement des facteurs CSV.

- Polars + numpy + statsmodels (HAC Newey–West), export PNG via matplotlib.
- Sélection des paires et trading : Python pur, déterministe, sans MATLAB.
- 41 tests unitaires (`py -m pytest -q`) couvrant pipeline, tables et seuils de validation.

Chiffres-clés (Tables 4 & 5)

Méthode	Rendement moyen (%)		Sharpe	
	Standard	Ondelettes	Standard	Ondelettes
Distance	−0,55	11,82	−0,21	3,69
Cointégration	−1,81	9,66	−0,40	2,82

Les tolérances de réplcation sont dans `paper_validation_summary.csv` : PASS si $|\Delta\text{mean}| \leq 1,0$ pp et $|\Delta\text{Sharpe}| \leq 0,30$.

Figure 4 — Rendements cumulés en excès

Figure 4 - Cumulative returns (distance)

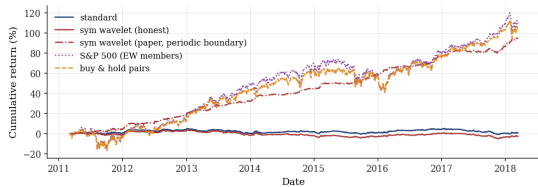
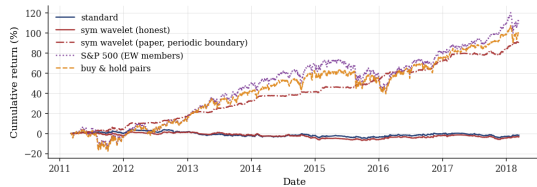


Figure 4 - Cumulative returns (cointegration)



Gauche : méthode des distances. Droite : méthode de cointégration.

Figure 5 — Ratios de Sharpe quotidiens

Figure 5 - Daily Sharpe-ratio evolution (distance)



Figure 5 - Daily Sharpe-ratio evolution (cointegration)

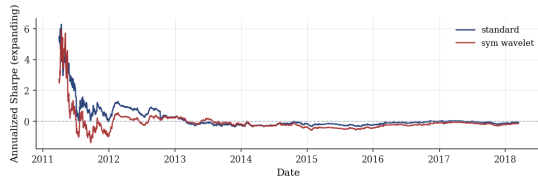


Figure 8 — Corrélation du bruit filtré

Figure 8 - Filtered-noise variance vs unfiltered return (distance); $\text{corr}=0.19$

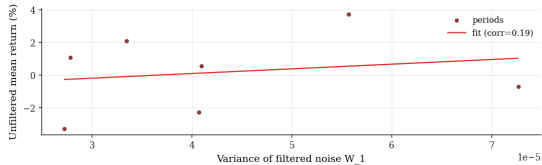


Figure 8 - Filtered-noise variance vs unfiltered return (cointegration); $\text{corr}=-0.32$

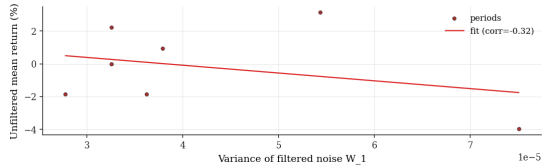


Figure 9 — Rendements anormaux annuels

Figure 9 - Yearly market-model alpha (distance); green = significant at 5%

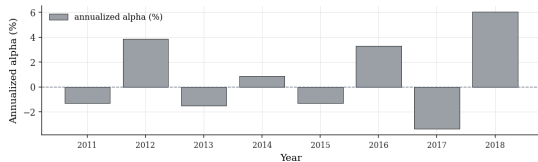
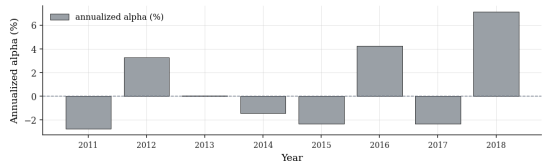


Figure 9 - Yearly market-model alpha (cointegration); green = significant at 5%



Merci de votre attention

Code de réplication, slides et sorties :
sous-arbre `research/` du backtesting framework.