

Attention Factors for Statistical Arbitrage

Elliot L. Epstein
 Stanford University
 Stanford, United States
 epsteine@stanford.edu

Jaewon Choi
 Hanwha Life
 Seoul, South Korea
 jaewonch@hanwha.com

Rose Wang
 Stanford University
 Stanford, United States
 rosew47@stanford.edu

Markus Pelger
 Stanford University
 Stanford, United States
 mpelger@stanford.edu

Abstract

Statistical arbitrage exploits temporal price differences between similar assets. We develop a framework to *jointly* identify similar assets through factors, identify mispricing and form a trading policy that maximizes risk-adjusted performance after trading costs. Our *Attention Factors* are conditional latent factors that are the most useful for arbitrage trading. They are learned from firm characteristic embeddings that allow for complex interactions. We identify time-series signals from the residual portfolios of our factors with a general sequence model. Estimating factors and the arbitrage trading strategy jointly is crucial to maximize profitability after trading costs. In a comprehensive empirical study we show that our Attention Factor model achieves an out-of-sample Sharpe ratio above 4 on the largest U.S. equities over a 24-year period. Our one-step solution yields an unprecedented Sharpe ratio of 2.3 net of transaction costs. We show that weak factors are important for arbitrage trading.

Keywords

Deep learning, attention, statistical arbitrage, latent factor model, sequence models, equities, investment

ACM Reference Format:

Elliot L. Epstein, Rose Wang, Jaewon Choi, and Markus Pelger. 2025. Attention Factors for Statistical Arbitrage. In *6th ACM International Conference on AI in Finance (ICAIF '25), November 15–18, 2025, Singapore, Singapore*. ACM, New York, NY, USA, 9 pages. <https://doi.org/10.1145/3768292.3770398>

1 Introduction

Statistical arbitrage exploits temporal price differences between similar assets using statistical methods. Conceptually, these methods are based on relative trades between a stock and a mimicking portfolio. The mimicking portfolio is constructed to be “similar” to the target stock, usually based on historical co-movements in the price time-series. When the spread between the prices of the

two comparison assets widens, the arbitrageur sells the winner and buys the loser. If their prices move back together, the arbitrageur will profit. Statistical arbitrage trading has to solve the following three key problems: Given a large universe of assets, what are long-short portfolios of similar assets? Given these portfolios, what are time-series signals that indicate the presence of temporary price deviations? Lastly, given these signals, how should an arbitrageur trade them to maximize risk-adjusted performance after trading costs? Each of these three questions poses substantial challenges, that prior work has only partly addressed.

Previous approaches have studied this problem as a two-step approach, where the first step identifies similar assets separately from the trading objective. Similarity between assets can be captured by similar exposure to risk factors. Arbitrage portfolios are trades relative to mimicking stock portfolios with the same exposure to risk factors. A common approach is to use principal component analysis (PCA) factors, which maximize the explained correlation in a panel and where mimicking portfolios are assets with high correlation with the target stocks. The second step in arbitrage trading is to identify time-series signals from the residuals of a candidate factor model and form a trading policy. The leading approach is [19], which outperforms the benchmarks in this literature. It uses PCA-type factors in the first step and a general sequence model in the second step. It achieves high Sharpe ratios before trading costs, but degrading Sharpe ratios after trading costs. The key issue in a two-step approach is that the factors cannot adjust to reduce trading costs for arbitrage strategies. For example, PCA factors have high turnover and large short positions, which diminish net performance. We provide a solution with our one-step approach.

In this paper, we propose the *Attention Factor Model*, a framework that *jointly* learns tradable arbitrage factors and arbitrage portfolio allocations in a computationally efficient manner. Our Attention Factors are conditional latent factors. The estimation objective is not to explain variation, but to construct profitable arbitrage strategies after trading costs. The attention mechanism learns embeddings of firm characteristics and allows to capture general dependencies of the factors on firm characteristics with complex interactions. A general sequence model learns time-series signals from the residuals of our attention factors with the joint objective of maximizing the net Sharpe ratio and explained variance. Figure 1 illustrates the conceptual structure.

We perform a comprehensive empirical out-of-sample analysis on 24 years of daily returns of the 500 largest and most liquid U.S. equities using an extensive set of firm characteristics. The Attention

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

ICAIF '25, Singapore, Singapore

© 2025 Copyright held by the owner/author(s). Publication rights licensed to ACM. ACM ISBN 979-8-4007-2220-2/2025/11 <https://doi.org/10.1145/3768292.3770398>

Factor model achieves an annualized Sharpe ratio above 4 without trading frictions and 2.3 with trading frictions, significantly outperforming prior work with an 84% increase in net Sharpe ratio over the current state-of-the-art model in [19]. Our arbitrage strategy yields an annual return of 16% while being uncorrelated to market risk. The Attention Factors have an interpretable structure, where the loadings are closely related to industry sectors. Our study provides evidence for weak factors that explain less variation but are important for identifying temporal mispricing.

2 Related Work

Statistical Arbitrage. Our paper builds on the classical statistical arbitrage literature, in which the three main problems of residual portfolio construction, time-series pattern extraction, and allocation decision have traditionally been considered independently. Classical statistical methods of generating arbitrage portfolios use parametric methods and have mostly focused on obtaining multiple pairs or small portfolios of assets, using techniques like the distance method of [16], the cointegration approach of [37], or copulas as in [33]. In contrast, more general methods that exploit large panels of stock returns include the use of PCA factor models, as in [1] and its extension in [38], and the maximization of mean-reversion and sparsity statistics as in [8]. Alternative parametric models include [6, 25, 29]. The most closely related paper to our work is [19], which uses transformer models to extract general time-series patterns from residuals. The residuals are obtained from PCA and IPCA factors to optimize explained variation. Our approach unifies factor extraction and residual trading within a single learning objective.

Machine Learning in Asset Pricing. Our paper is complementary to the fast growing literature that uses machine learning methods for asset pricing. While the asset pricing literature aims to explain the risk premia of assets, our focus is on the residual component which is not explained by the asset pricing models. [5, 7, 21, 23] estimate the stochastic discount factor (SDF), which explains the risk premia of assets, with deep neural networks, decision trees, elastic net regularization and attention methods. [3, 4, 9, 14, 18, 20] predict asset returns with machine learning methods.

Statistical Factor Modeling. The workhorse models in equity asset pricing are based on linear factor models exemplified by [11, 12]. Recently, new methods have been developed to extract statistical factors from large panels with various versions of PCA that explain the systematic comovement between assets [2, 13]. Motivated by Arbitrage Pricing Theory (APT), systematic risk factors are expected to explain the cross-section of expected returns. Extensions of PCA include RP-PCA [24] to account for pricing errors, state-dependent factors in [32], interpretable PCA [31], high-frequency PCA [30], and conditional factor models in Instrumented PCA (IPCA) [22] linking latent loadings to observable characteristics. Statistical factors that explain the variation in panels are complementary to our work as they have a different objective. Our method estimates factors that are the most useful for arbitrage trading.

Machine learning for Time-Series. Our paper builds on the literature for time-series modeling with sequence models, which typically solve a time-series prediction problem. We estimate a Long-Conv [15] model jointly with our Attention Factor model [36] with

a trading objective. Popular sequence models to learn general time-series patterns are Transformer [36] models and S4 [17] models. The Set-Sequence Model [10] captures joint dependencies for arbitrary sequence models. Low-rank Gaussian copula processes model joint distributions with tractable structure [34], while global-local networks exploit parameter sharing with series-specific conditioning [35]. Transformer-based models dominate recent benchmarks: Crossformer introduces cross-dimension attention for multivariate dependencies [39], iTransformer inverts tokenization to attend over variates and scales to long horizons [26], and S4-based models use structured state spaces to efficiently handle long sequences [17].

3 Method

Notation. We consider N assets with returns $R_t \in \mathbb{R}^N$ with M time-varying characteristics $X_t \in \mathbb{R}^{N \times M}$ for the times $t = 1, \dots, T$.

Problem. The fundamental problem of statistical arbitrage consists of three elements: (1) Identification of similar assets to generate arbitrage portfolios, (2) extraction of time-series signals for temporary deviations of similarity between assets and (3) a trading policy in the arbitrage portfolios based on the time-series signals. We provide a general end-to-end solution for each element.

3.1 Factor Model

Conditional Factor Model. Factor models explain the returns of a cross-section of assets in terms of their exposure to factors $F_t = (F_{1,t}, \dots, F_{K,t})$. We use factors to identify similar assets, where similarity is defined as the same exposure to factors. We assume that asset returns can be modeled by a conditional factor model:

$$R_{i,t} = \beta_{i,t-1}^T F_t + \epsilon_{i,t}, \quad t = 1, \dots, T \text{ and } i = 1, \dots, N.$$

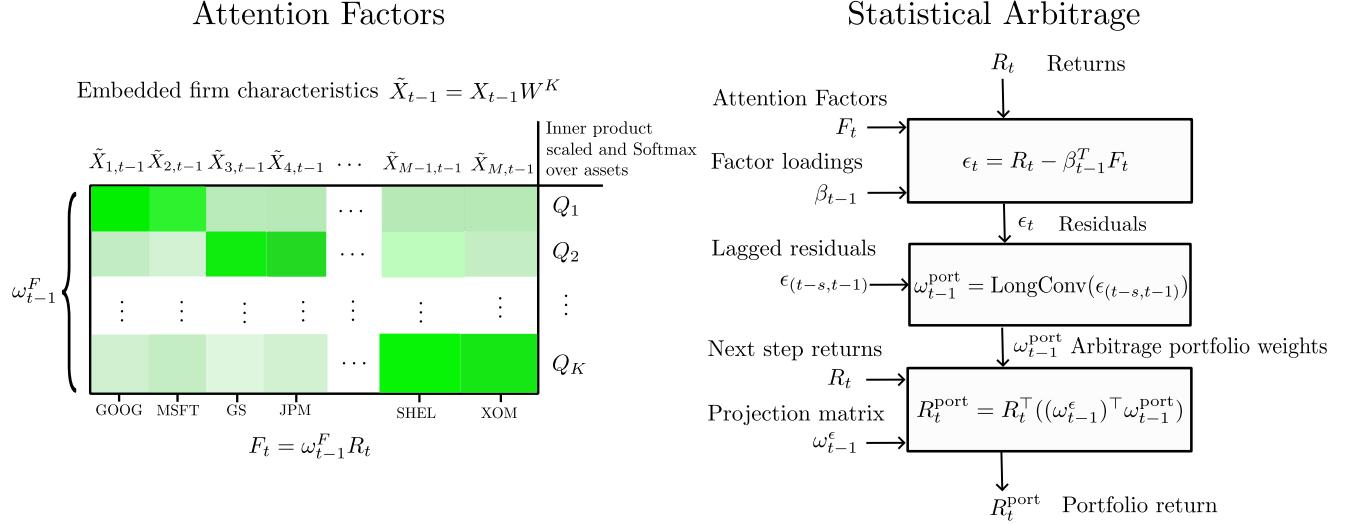
The K tradable factors $F_t \in \mathbb{R}^K$ capture systematic risk, while the loadings $\beta_{i,t-1}$ are time-varying and based on information up to time $t-1$. This general formulation includes the empirically most successful factor models.

Through the lens of Arbitrage Pricing Theory (APT) [24], if the factors capture all relevant sources of systematic risk, then the factor portfolio $\beta_{i,t-1}^T F_t$ is the “fair price”, and the residual portfolio, $\epsilon_{i,t}$, given by

$$\epsilon_{i,t} = R_{i,t} - \beta_{i,t-1}^T F_t,$$

identifies mispricing. Arbitrage trading aims to exploit temporal patterns in the residuals.

Candidate Factors. Empirically successful factor models include observed fundamental factors and statistical factors. Examples of fundamental factors are the market factor in the CAPM model, and the Fama-French 3- and 5-factor models [11] that include a market, size, value, respectively, investment and profitability factor. The Fama-French factors are tradeable portfolios, $F_t = \omega_{t-1}^{\text{FF}} R_t$, where portfolio weights ω_{t-1}^{FF} depend on past firm characteristics like size or book-to-market ratios. Statistical factor models encompass unconditional and conditional factor models. The most widely used unconditional factor models are based on versions of PCA; see [24] for an overview. PCA estimates latent factors that explain the most cross-sectional variation. These latent factors are typically extracted as $F_t = \omega^{\text{PCA}} R_t$, where ω^{PCA} are the eigenvectors of the top K eigenvalues of the return covariance matrix. Conditional

Figure 1: Conceptual Attention Factor Model

The figure illustrates the conceptual structure of the Attention Factor model. Left: Attention factors are constructed by computing scaled inner products between embedded characteristics for each asset and the K query vectors Q_k . Right: The statistical arbitrage methodology. First, for each asset, a replicating portfolio based on the attention factors is created, giving a residual mispricing. Second, a series of lagged residuals are used to construct the portfolio weights in the residual space, using a Long Convolution model for sequence modeling. Finally, the portfolio weights are mapped back to the asset space via a composition matrix, giving the next-period portfolio return.

statistical factors model the factor portfolio weights and loadings as functions of characteristics. A prominent example is IPCA in [22] with $F_t = \omega_{t-1}^{\text{IPCA}} R_t$, where the weights $\omega_{t-1}^{\text{IPCA}} = X_{t-1}^T B$ are a linear function of firm characteristics. This conditional structure allows exposures to evolve with firm attributes, linking the latent factors to observable fundamental information. However, the factors are still estimated to maximize explained variation in the cross-section.

Challenges for Arbitrage Trading. The above factor models are not constructed with the objective to create profitable arbitrage portfolios. These models impose restrictive ad-hoc assumptions on the functional form of loadings and portfolio weights, and do not target factors based on a trading objective. Importantly, the resulting factor portfolios might have high trading costs in terms of turnover and shorting positions. We provide a solution that addresses all these challenges.

3.2 Attention Arbitrage Factors

Residual Portfolios. We estimate a conditional factor model that is optimal for arbitrage trading. As in factor models from the previous section, our factors are tradable portfolios

$$F_t = \omega_{t-1}^F R_t,$$

for a factor portfolio weight matrix $\omega_{t-1}^F \in \mathbb{R}^{K \times N}$. This implies that the residuals are traded portfolios as well

$$\epsilon_t = R_t - \beta_{t-1}^T F_t = R_t - \beta_{t-1}^T \omega_{t-1}^F R_t = \omega_{t-1}^\epsilon R_t,$$

for the implied projection matrix $\omega_{t-1}^\epsilon = I_N - \beta_{t-1}^T \omega_{t-1}^F$.

Attention Factors. Our attention factors allow for a general functional form for the weights and loadings, that captures complex

dependencies between characteristics. In our approach, for each point in time, the firm characteristics X_t are first embedded as

$$\tilde{X}_t = X_t W^K, \quad W^K \in \mathbb{R}^{M \times d}.$$

Each asset is attended with a dot product to the query vector $Q_k \in \mathbb{R}^d$ for each factor, with query matrix $Q = (Q_1, \dots, Q_K)^T$ giving a factor weight matrix ω_{t-1}^F as

$$\omega_{t-1}^F = \text{Softmax}(Q \tilde{X}_{t-1}^T / \sqrt{d}), \quad (1)$$

where Softmax is applied along each row to ensure that the resulting factors are normalized. The name attention factor is due to the similarity of Equation 1 with the multi-head attention mechanism used in Transformer [36] models, but instead of standard attention that compares each token in a sequence (across time) to each other token, we compare each asset (in the cross section) with each factor. Simpler models, for example IPCA, are a special case of this formulation. Factor loadings β_{t-1} and weights ω_{t-1}^F are mechanically related, as up to a rotation the loadings represent factor portfolio weights (see, for example, [28]). We obtain the factor loadings as

$$\beta_{t-1}^T = \omega_{t-1}^F {}^T \left(\omega_{t-1}^F (\omega_{t-1}^F)^T + \lambda_{\text{ridge}} I_K \right)^{-1},$$

where we add a ridge penalty λ_{ridge} for stability. Hence, the estimation of ω_{t-1}^F directly implies β_{t-1} and ω_{t-1}^ϵ .

3.3 Arbitrage Trading

Arbitrage Portfolio. The key idea of statistical arbitrage is to exploit predictable patterns in the time-series of residual portfolios. Traditionally, statistical arbitrage focuses on parametric mean-reversion patterns. We detect time-series patterns in the residual

Table 1: Firm Characteristics by Category

Past Returns		Value	Investment	
r2_1	Short-term momentum	A2ME	Assets / market cap	Investment
r12_2	Momentum	BEME	Book-to-market ratio	NOA
r12_7	Intermediate momentum	C	Cash + ST inv. / assets	DPI2A
r36_13	Long-term momentum	CF	Free cash-flow / book value	Net operating assets
ST_Rev	Short-term reversal	CF2P	Cash-flow / price	Change in PP&E
Ret_D1	Daily return	Q	Tobin's Q	
Ret_W1	Weekly return	Lev	Leverage	
STD_W1	Weekly volatility	E2P	Earnings / Price	
Trading Frictions		Profitability	Intangibles	
AT	Total assets	PROF	Profitability	OA
LME	Size	CTO	Capital turnover	OL
LTurnover	Turnover	FC2Y	Fixed costs / sales	PCM
Rel2High	52-week-high closeness	OP	Operating profitability	Operating accruals
Resid_Var	Residual variance	PM	Profit margin	Operating leverage
Spread	Bid-ask spread	RNA	Return on NOA	Price-to-cost margin
SUV	Standard unexplained volume	D2A	Capital intensity	
Variance	Variance			
Vol	Weekly trading volume			
Beta	Beta with market			

The table shows the 39 firm-specific characteristics (six categories) used as features to construct the attention factors. Construction details are in the Internet Appendix of [7].

portfolios with a flexible data-driven filter based on convolutional networks using a trading objective. The arbitrage portfolio weight function depends on the time-series signals that we extract with LongConv [15] from the past s residuals $\epsilon_{i,(t-s,t-1)}$ as

$$\omega_{i,t-1}^{\text{port}} = \text{LongConv}_{\theta}(\epsilon_{i,(t-s,t-1)}),$$

where θ denotes the learnable parameters of the LongConv model. LongConv can capture complex time-series patterns. We chose it because of its linear scaling in the sequence length and simplicity. The choice of sequence model is flexible, and we expect alternative sequence models such as Transformers to perform similarly. Each convolution captures distinct time-series patterns, and our optimally tuned model has 32 different convolutions.

Arbitrage Trading. The arbitrage portfolio return, R_t^{port} , is

$$R_t^{\text{port}} = \epsilon_t^{\top} \omega_{t-1}^{\text{port}} = R_t^{\top} ((\omega_{t-1}^{\epsilon})^{\top} \omega_{t-1}^{\text{port}}) = R_t^{\top} \omega_{t-1},$$

with portfolio weights $\omega_{t-1} = (\omega_{t-1}^{\epsilon})^{\top} \omega_{t-1}^{\text{port}}$ in the asset space. Note that ω_{t-1}^{ϵ} is only a function of lagged firm characteristics X_{t-1} , while $\omega_{t-1}^{\text{port}}$ is only a function of the time-series patterns in residuals.

Arbitrage Trading Objective. We estimate the arbitrage portfolio weights to maximize the Sharpe ratio after transaction costs. We measure transaction costs as in [19], which is common in this literature:

$$\text{cost}(\omega_t, \omega_{t-1}) = 0.0005 \times \|\omega_t - \omega_{t-1}\|_1 + 0.0001 \times \|\max(-\omega_t, 0)\|_1,$$

The first penalty represents a transaction cost of 5 basis points per transaction, whereas the second one is a shorting cost of 1 basis point. The portfolio net return is then calculated as:

$$R_{t,\text{net}}^{\text{port}} = R_t^{\text{port}} - \text{cost}(\omega_t, \omega_{t-1}).$$

Our objective function maximizes the net Sharpe ratio of the arbitrage portfolio and the explained variance of the factors. The

tradeoff between these two objectives is selected optimally on the validation data. Including the explained variance is necessary for identification, and empirically improves the performance. This framework nests conditional latent factors that maximize explained variance as a special case.

$$\max_{\omega^F, \omega^{\text{port}}} \frac{\bar{R}_{\text{net}}^{\text{port}} - R_f}{\sqrt{\frac{1}{T} \sum_{t=1}^T (R_{t,\text{net}}^{\text{port}} - \bar{R}_{\text{net}}^{\text{port}})^2}} + \lambda_{\text{Var}} \cdot \underbrace{\frac{1}{N} \sum_{i=1}^N \left(1 - \frac{\text{Var}(e^i)}{\text{Var}(R^i)}\right)}_{\text{explained variance}}$$

subject to $\|\omega_t\|_1 = 1$ and where $\bar{R}_{\text{net}}^{\text{port}} = \frac{1}{T} \sum_{t=1}^T R_{t,\text{net}}^{\text{port}}$, and R_f is the risk-free rate. The learned parameters that determine ω^F and ω^{port} are the query matrix Q , the embedding matrix W^K , and the LongConv model parameters θ .

4 Empirical Analysis

4.1 Data

Data Sets. We collect daily equity return data for the securities on CRSP from January 1990 through December 2021. Our analysis uses only the most liquid stocks. More specifically, we consider only the 500 largest stocks based on the previous month market capitalization. We complement the stock returns with 39 firm-specific characteristics from [7], which are listed in Table 1. All these variables are constructed either from accounting variables from the CRSP/Compustat database or from past returns from CRSP. The full details on the construction of these variables are in the Internet Appendix of [7]. Firm characteristics are normalized to rank quantiles as it is standard in this literature. In addition to the most important characteristics from [7] we also include the previous day and week return and volatility. In order to keep the level information, we also include the cross-sectional median of the characteristics and the

risk free rate, which results in the dimension of X_{t-1} of 79. Missing values in the characteristics are imputed with last observed values if available and by the cross-sectional median otherwise.

4.2 Estimation

Models. We estimate our models on a rolling window of 8 years, where we retrain the models every year and evaluate them out-of-sample from January 1998 to December 2021. We compare our Attention Factor model to two natural benchmark models, which both estimate residuals to maximize explained variation. The parametric benchmark from the seminal work of [1] corresponds to classical mean-reversion trading. The second benchmark estimates factors with PCA, but uses the same flexible convolution model for trading the residuals. It allows us to understand the importance of the one-step optimization and general functional form. Both sets of benchmark models have been found to perform strongly empirically.

In summary, the three classes of models are

- (1) **Attention Factors:** The factors and arbitrage trading policy are learned in one-step with a trading objective including transaction costs. We consider 1, 3, 5, 8, 10, 15, 30, and 100 latent factors.
- (2) **PCA Factors:** The latent factors are estimated with PCA using the past 252 trading days. The residual portfolios weights $\omega_{\text{port}}^{\text{port}}$ are estimated with our LongConv model and the Sharpe ratio objective with trading costs. Hence, we allow the same flexibility for $\omega_{t-1}^{\text{port}}$ as in our attention model. This is essentially a benchmark in the spirit of [19].
- (3) **PCA+OU Thresh:** We use PCA factors to estimate residuals and use a parametric portfolio weight based on an Ornstein-Uhlenbeck model with thresholding rule proposed in [1]. We use the same implementation as in [19].

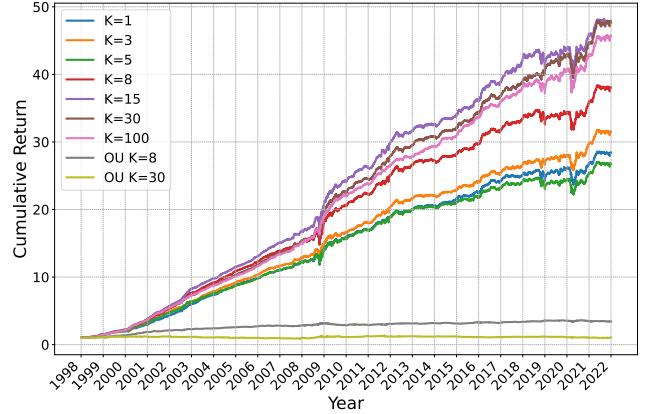
We report the out-of-sample annualized Sharpe ratio, average return, volatility and net results after subtracting transaction costs.

Implementation. Adam [27] was used as our optimizer. We use the last two years of the first training data to select tuning parameters and find that our results are robust to the tuning parameter selection. We follow the architectural choices of prior work, and test our model for a wide range of parameter choices. Our optimal Attention Factor model has a hidden dimension of $d_x = 32$. The optimal LongConv sequence model has 1 layer with hidden dimension 32. The optimal weight on the variance is $\lambda_{\text{Var}} = 100$. All sequence models use a look-back window of the past 30 daily residual returns.

4.3 Results

Performance. Table 2 shows the main results for our attention arbitrage model and the benchmarks for different number of factors. First, our Attention Factor model achieves an excellent performance as demonstrated by the annual Sharpe ratio of around 4 with 30 attention factors. It substantially outperforms the parametric benchmark, which can achieve a solid Sharpe ratio of 1.2. A two-step approach with PCA factors and the convolutional time-series filter achieves out-of-sample Sharpe ratios close to 2.8 with 30 PCA factors, which illustrates the importance of a flexible time-series

Figure 2: Cumulative Returns of Arbitrage Portfolios



The figure shows the out-of-sample cumulative returns for different arbitrage portfolios. We consider the Attention Factor model with different number of factors, K, and the parametric Ornstein-Uhlenbeck model with thresholding trading policy (OU) on the out-of-sample period from Jan 1998 to Dec 2021.

filter. All models are essentially uncorrelated with a market factor. We conclude that our attention factors identify more profitable arbitrage opportunities.

Net Results. Second, our attention factors represent the best performing model after transaction costs. We compare the Sharpe ratios and average returns after transaction costs. The strategy of the parametric model drops to negative net Sharpe ratios and mean returns due to its excessive turnover. Similarly, a simple PCA factor model cannot adjust the factor construction to trading frictions, resulting in a deterioration of the net performance to a Sharpe ratio of around 1.5. In contrast, our Attention Factor model provides unprecedented performance of 2.3 after trading costs. This is due to the end-to-end optimization and including the transaction costs in the objective function. The model learns to identify arbitrage strategy that is profitable after taking transaction costs into account. This makes our model one of the best performing models in the literature under realistic frictions.

Number of Factors. Third, we demonstrate the importance of weak factors for arbitrage trading. Our attention arbitrage model achieves a substantial performance for 8 attention factors. However, we observe an out-of-sample improvement for including 30 factors. These higher order factors capture weak signals and local dependency patterns. This is in line with [24], who show that weaker factors that capture local dependency patterns are important for trading. A model with 100 attention factors leads to further minor improvements. This means that our model does not overfit, but discovers further weak signals, as increasing K expands the model's capacity to optimize the trading objective without requiring factor independence.

Performance over Time. Figure 2 shows cumulative out-of-sample returns for the different arbitrage strategies. Our attention factors exhibit a strong performance throughout the full sample - even during the later part of the sample where arbitrage trading is more

Table 2: Out-of-sample Annualized Performance

Model	K	SR	μ	σ	SR_{net}	μ_{net}	σ_{net}	Beta
Attention Factors	1	3.05	14.45	4.74	1.68	7.94	4.72	0.05
	3	3.05	14.91	4.89	1.69	8.25	4.87	0.06
	5	2.92	14.21	4.87	1.58	7.66	4.85	0.07
	8	3.35	15.70	4.68	1.94	9.05	4.66	0.07
	15	3.81	16.66	4.37	2.25	9.78	4.35	0.06
	30	3.97	16.66	4.20	2.28	9.52	4.18	0.05
	100	4.52	16.45	3.64	2.19	7.93	3.62	0.05
Parametric Benchmark PCA + OU Thresh	1	0.40	1.85	4.57	-2.54	-11.62	4.57	0.02
	3	1.26	4.18	3.33	-2.72	-9.04	3.33	0.01
	5	0.99	2.91	2.93	-3.44	-10.11	2.93	0.00
	8	0.78	2.04	2.61	-4.15	-10.83	2.61	0.00
	10	0.80	1.99	2.50	-4.32	-10.80	2.50	0.00
	15	0.51	1.12	2.20	-5.24	-11.53	2.20	0.00
	30	0.18	0.40	2.24	-6.45	-14.74	2.29	-0.00
	100	-0.35	-0.66	1.87	-7.05	-13.23	1.88	-0.00
PCA Factors (Two-Step Approach)	1	2.26	13.10	5.79	1.19	6.98	5.78	0.10
	3	2.76	14.61	5.30	1.57	8.29	5.28	0.07
	5	2.41	14.10	5.86	1.30	7.62	5.84	0.10
	8	2.64	14.88	5.63	1.50	8.42	5.61	0.09
	10	2.66	14.94	5.61	1.52	8.48	5.59	0.09
	15	2.56	14.74	5.75	1.41	8.08	5.73	0.09
	30	2.79	15.15	5.42	1.57	8.47	5.40	0.09
	100	2.66	14.36	5.40	1.44	7.75	5.38	0.09
Market	-	0.42	8.61	20.37	0.42	8.61	20.37	1.00

The table shows the out-of-sample arbitrage trading performance using different models (Jan. 1998– Dec. 2021). We report the results for our Attention Factor model, and for a two-step approach where the residual obtained from PCA factors. The portfolio weight functions is estimated with LongConv from the residuals of each factor model. For each model, K denotes the number of factors. Parametric benchmark PCA+Thresh is the parametric Ornstein-Uhlenbeck model with thresholding trading policy based on [1] and implemented as in [19]. We use a lookback window of 30 days of residual returns, and PCA factors are estimated on rolling window of 252 days. K denotes the number of factors. The Sharpe Ratio (SR), mean return (μ in %), and standard deviation (σ) are annualized. "Net" metrics account for transaction and shorting costs. Beta denotes the market beta. The equally weighted market portfolio is provided for reference.

challenging. The large-volatility period in early 2020 due to COVID-19 led to temporary deviations in market dynamics, reflecting a short-lived distribution shift relative to the preceding training windows. While a shorter training horizon may have been able to adapt more rapidly to these changing conditions, we maintain a fixed 8-year rolling window to ensure a consistent empirical design.

4.4 Interpretation

Drivers of Performance. Table 3 reports the effect on out-of-sample performance after removing a group of characteristics. Removing past return information in the attention factors substantially reduces the performance and the net Sharpe ratios drop to 0.59. In contrast, removing any other characteristic group has a negligible effect. This indicates that price based patterns and not “classical” firm characteristics are driving arbitrage trading. The values are averaged across multiple starting seeds for the neural network estimation, and the value in parentheses for the Sharpe Ratio is the standard deviation across seeds. We conclude that the results are robust to the implementation.

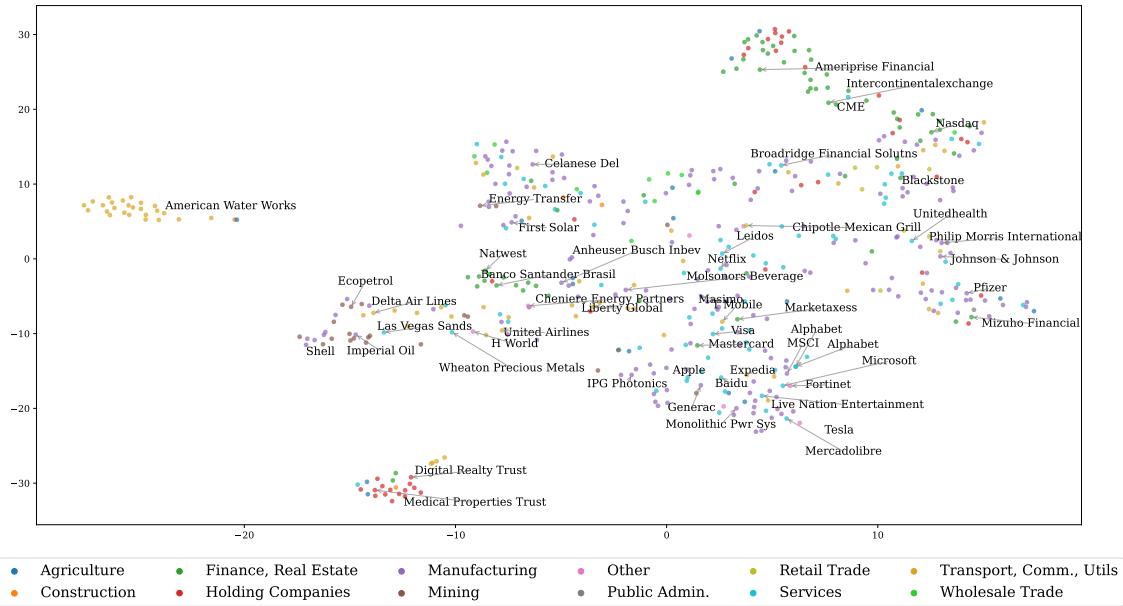
Factor Structure. Our attention factor structure has a clear economic interpretation. We focus on the attention 8-factor model, which already achieves a substantial performance. Figure 3 represents the similarity of firms captured by the loadings, that is, firms with similar loadings are considered closer. We use t-SNE to represent closeness of firms in the loading space. Here we evaluate our model in a specific date, but similar results hold throughout our sample. Firms in similar industries are grouped together, that is, our model learns specific industries. For example, the upper right cluster represents banks and financial, the lower right petroleum and energy companies, lower middle real estate companies, lower left utility companies, middle left hotels and middle-lower left technology companies. Note, we do not provide industry classification, but this similarity is learned from price data and firm fundamentals.

Factors Weights. The factor portfolio weights have a clear interpretation in terms of industry sectors. Figure 4 shows the top 10 companies that are used to construct the first six factors on a representative date. In each case, the first top 10 companies account for a large portion of the total weight in the companies (between 10%-23%). We see clear industry relationships. Factor 1 represents

Table 3: Characteristic Importance for Model Performance

Dropped Feature	SR	μ	σ	SR_{net}	μ_{net}	Beta
baseline (none excluded)	3.97 (0.13)	16.66	4.20	2.28	9.52	0.05
past returns	1.50 (0.07)	7.82	5.23	0.59	3.09	0.08
investment	3.88 (0.17)	17.93	4.63	2.19	10.06	0.06
profitability	3.94 (0.15)	18.39	4.67	2.26	10.48	0.05
intangibles	3.91 (0.15)	18.18	4.65	2.24	10.34	0.06
value	4.08 (0.12)	18.45	4.53	2.32	10.44	0.04
trading frictions	2.90 (0.14)	13.36	4.61	1.34	6.14	0.06

The table shows the out-of-sample model performance when dropping characteristic groups in the estimation and evaluation. The Sharpe Ratio (SR), mean return (μ in %), and standard deviation (σ) are annualized. "Net" metrics account for transaction and shorting costs. Reported Sharpe Ratio, mean return, and return standard deviation are annualized. Net Sharpe Ratio is calculated after accounting for transaction costs and shorting costs. Beta is relative to the market. All values are averaged across multiple starting seeds for model estimation of the neural networks, and the value in parentheses for the Sharpe Ratio is the standard deviation across seeds. Each row drops a characteristic group to assess its feature group importance. The number of attention factors are 30. The out-of-sample evaluation is from Jan. 1998– Dec. 2021.

Figure 3: Interpretation of Attention Factor Betas

The figure illustrates the attention factor loading composition by showing the t-SNE (t-distributed stochastic neighbor embedding) projection of estimated betas for the first 8 attention factors estimated on the 500 equities with the largest market cap on the first trading day of 2021. The training period is January 2013 - December 2020. The dots are colored based on industry classification. The Attention Factor model betas capture meaningful dependencies between firms. The clusters represent different industry sectors: the upper right cluster represents banks and financial firms, lower right petroleum and energy companies, lower middle real estate companies, lower left utility companies, middle left hotels and middle-left technology.

technology, factor 2 natural resources, factor 3 the financial industry, factor 4 holding companies, factor 5 consumer manufacturing and factor 6 captures energy companies.

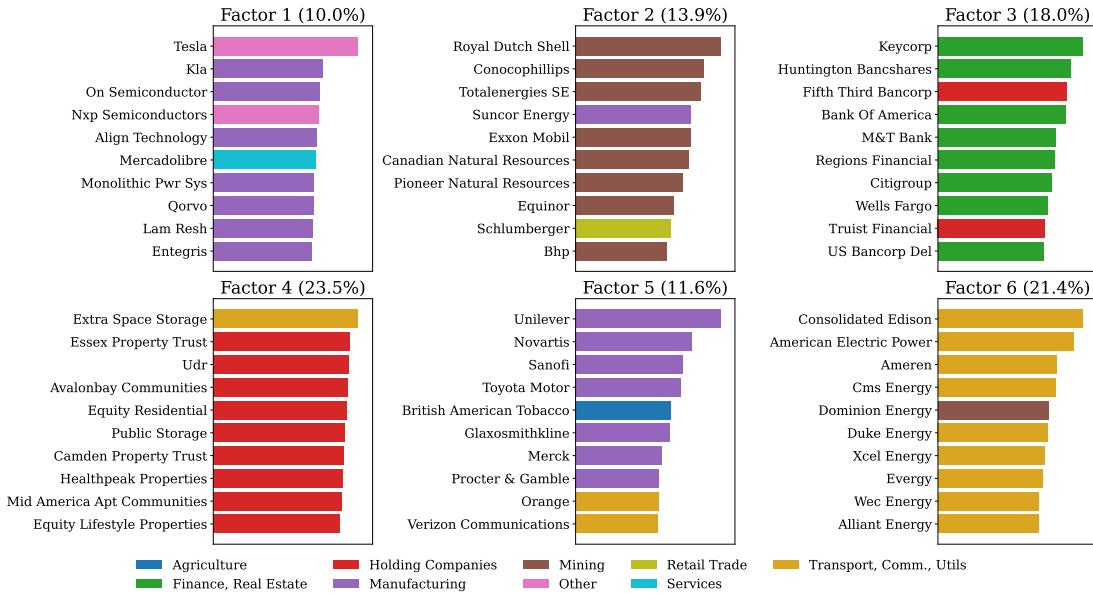
5 Conclusion

This paper develops an Attention Factor model for statistical arbitrage. We provide a one-step estimation framework for latent factors to identify similar assets and an arbitrage portfolio allocation based on time-series patterns. The two key innovations are the

conditional latent attention factors to capture complex dependencies in firm characteristics and the one-step model estimation that maximizes portfolio performance after trading frictions. In extensive empirical analysis, we demonstrate that our arbitrage model sets the new standard in this literature with the best performance under realistic trading frictions.

References

- [1] Marco Avellaneda and Jeong-Hyun Lee. 2010. Statistical arbitrage in the US equities market. *Quantitative Finance* 10, 7 (2010), 761–782. <https://doi.org/10.1080/14697680903082371>.

Figure 4: Attention Weights for Factor Portfolio Weights

This figure shows attention weights for constructing the factor portfolio weights. We display the top 10 company weights for the first 6 factors (out of total 8) for the Attention Factor model. We also report the industry for each company, along with the percentage weight (out of the full 500 assets) that the top 10 companies represent. The evaluation is on the first trading day of 2021 for the model trained from January 2013 to December 2020.

- 1080/14697680903124632
- [2] Jushan Bai and Serena Ng. 2002. Determining the Number of Factors in Approximate Factor Models. *Econometrica* 70, 1 (2002), 191–221. <https://doi.org/10.1111/1468-0262.00273>
 - [3] Turan G. Bali, Amit Goyal, Dashan Huang, Fuwei Jiang, and Quan Wen. 2022. Predicting Corporate Bond Returns: Merton Meets Machine Learning. *Working paper* (2022). <https://doi.org/10.2139/ssrn.3686164>
 - [4] D. Bianchi, M. Büchner, and A. Tamoni. 2021. Bond risk premia with machine learning. *Review of Financial Studies* 34, 2 (2021), 1046–1089. <https://doi.org/10.1093/rfs/hhaa062>
 - [5] Svetlana Bryzgalova, Markus Pelger, and Jason Zhu. 2023. Forest through the Trees: Building Cross-Sections of Stock Returns. *Journal of Finance, forthcoming* (2023). <https://dx.doi.org/10.2139/ssrn.3493458>
 - [6] Álvaro Cartea and Sebastian Jaimungal. 2016. Algorithmic trading of co-integrated assets. *International Journal of Theoretical and Applied Finance* 19, 6 (2016), 165038. <http://dx.doi.org/10.2139/ssrn.2637883>
 - [7] Luyang Chen, Markus Pelger, and Jason Zhu. 2024. Deep Learning in Asset Pricing. *Management Science* 70, 2 (2024), 714–750. <https://dx.doi.org/10.2139/ssrn.3350138>
 - [8] Alexandre d'Aspremont. 2011. Identifying small mean-reverting portfolios. *Quantitative Finance* 11, 3 (2011), 351–364. <https://doi.org/10.1080/14697688.2010.481634>
 - [9] Victor DeMiguel, J. Gil-Bazo, F.J. Nogales, and A.A.P. Santos. 2023. Machine Learning and Fund Characteristics Help to Select Mutual Funds with Positive Alpha. *Journal of Financial Economics* 150, 3 (2023), 1–22. <https://doi.org/10.1016/j.jfineco.2023.103737>
 - [10] Elliot L. Epstein, Apoorv Sadhwani, and Kay Giesecke. 2025. A Set-Sequence Model for Time Series. <https://doi.org/10.48550/arXiv.2505.11243>
 - [11] Eugene F. Fama and Kenneth R. French. 1993. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics* 33, 1 (1993), 3–56. [https://doi.org/10.1016/0304-405X\(93\)90023-5](https://doi.org/10.1016/0304-405X(93)90023-5)
 - [12] Eugene F. Fama and Kenneth R. French. 2015. A five-factor asset pricing model. *Journal of Financial Economics* 116, 1 (2015), 1–22. <https://doi.org/10.1016/j.jfineco.2014.10.010>
 - [13] Jianqing Fan, Yuan Liao, and Martina Mincheva. 2013. Large covariance estimation by thresholding principal orthogonal complements. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 75, 4 (2013), 603–680. <https://doi.org/10.1111/rssb.12016>
 - [14] Joachim Freyberger, Andreas Neuhierl, and Michael Weber. 2020. Dissecting characteristics nonparametrically. *Review of Financial Studies, forthcoming* 33, 5 (2020), 2326–2377. <https://doi.org/10.1093/rfs/hhz123>
 - [15] Daniel Y. Fu, Elliot L. Epstein, Eric Nguyen, Armin W. Thomas, Michael Zhang, Tri Dao, Atri Rudra, and Christopher Ré. 2023. Simple Hardware-Efficient Long Convolutions for Sequence Modeling. In *Proceedings of the 40th International Conference on Machine Learning*. <https://doi.org/10.48550/arXiv.2302.06646>
 - [16] Evan Gatev, William N. Goetzmann, and K. Geert Rouwenhorst. 2006. Pairs Trading: Performance of a Relative-Value Arbitrage Rule. *The Review of Financial Studies* 19, 3 (02 2006), 797–827. <https://doi.org/10.1093/rfs/hhj020>
 - [17] Albert Gu, Karan Goel, and Christopher Re. 2022. Efficiently Modeling Long Sequences with Structured State Spaces. In *International Conference on Learning Representations*. <https://doi.org/10.48550/arXiv.2111.00396>
 - [18] Shihao Gu, Bryan T. Kelly, and Dacheng Xiu. 2020. Empirical Asset Pricing Via Machine Learning. *Review of Financial Studies* 33, 5 (2020), 2223–2273. <https://doi.org/10.1093/rfs/hhaa009>
 - [19] Jorge Guijarro-Ordonez, Markus Pelger, and Greg Zanotti. 2025. Deep Learning Statistical Arbitrage. *Management Science (accepted)* (2025). <https://dx.doi.org/10.2139/ssrn.3862004>
 - [20] Ron Kaniel, Zihan Lin, Markus Pelger, and Stijn Van Nieuwerburgh. 2023. Machine-learning the skill of mutual fund managers. *Journal of Financial Economics* 150, 1 (2023), 94–138. <https://doi.org/10.1016/j.jfineco.2023.07.004>
 - [21] Bryan T. Kelly, Boris Kuznetsov, Semyon Malamud, and Teng Andrea Xu. 2025. *Artificial Intelligence Asset Pricing Models*. NBER Working Paper 33351. National Bureau of Economic Research. <http://www.nber.org/papers/w33351>
 - [22] Bryan T. Kelly, Seth Pruitt, and Yinan Su. 2019. Characteristics are covariances: A unified model of risk and return. *Journal of Financial Economics* 134, 3 (2019), 501–524. <https://doi.org/10.1016/j.jfineco.2019.05.001>
 - [23] Serhiy Kozak, Stefan Nagel, and Shrihari Santosh. 2020. Shrinking the cross-section. *Journal of Financial Economics* 135, 2 (Feb. 2020), 271–292. <https://doi.org/10.1016/j.jfineco.2019.06.008>
 - [24] Martin Lettau and Markus Pelger. 2020. Factors That Fit the Time Series and Cross-Section of Stock Returns. *The Review of Financial Studies* 33, 5 (03 2020), 2274–2325. <https://doi.org/10.1093/rfs/hhaa020>
 - [25] Paul Sopher Lintilhac and Agnes Tourin. 2016. Model-based pairs trading in the bitcoin markets. *Quantitative Finance* 17, 5 (2016), 703–716. <https://doi.org/10.1080/14697688.2016.1231928>
 - [26] Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. 2024. iTransformer: Inverted Transformers Are Effective for Time Series Forecasting. <https://doi.org/10.48550/arXiv.2310.06625>
 - [27] Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations (ICLR)*. <https://doi.org/10.48550/arXiv.1711.09205>

- 10.48550/arXiv.1711.05101
- [28] Hao Ma. 2021. *Conditional Latent Factor Models Via Econometrics-Based Neural Networks*. Technical Report. Queen Mary University. <https://dx.doi.org/10.2139/ssrn.3946388>
 - [29] Supakorn Mudchanatongsuk, James A Prinbs, and Wilfred Wong. 2008. Optimal pairs trading: A stochastic control approach. In *2008 American Control Conference*. IEEE, 1035–1039. <http://10.1109/ACC.2008.4586628>
 - [30] Markus Pelger. 2020. Understanding Systematic Risk: A High-Frequency Approach. *The Journal of Finance* 75, 4 (2020), 2179–2220. <https://doi.org/10.1111/jofi.12898>
 - [31] Markus Pelger and Ruoxuan Xiong. 2022. Interpretable Sparse Proximate Factors for Large Dimensions. *Journal of Business & Economic Statistics* 40, 4 (2022), 1642–1664. <https://doi.org/10.1080/07350015.2021.1961786>
 - [32] Markus Pelger and Ruoxuan Xiong. 2022. State-Varying Factor Models of Large Dimensions. *Journal of Business & Economic Statistics* 40, 3 (2022), 1315–1333. <https://doi.org/10.1080/07350015.2021.1927744>
 - [33] Hossein Rad, Rand Kwong Yew Low, and Robert Faff. 2016. The profitability of pairs trading strategies: distance, cointegration and copula methods. *Quantitative Finance* 16, 10 (2016), 1541–1558. <https://doi.org/10.1080/14697688.2016.1164337>
 - [34] David Salinas, Michael Bohlke-Schneider, Laurent Callot, Roberto Medico, and Jan Gasthaus. 2019. High-dimensional multivariate forecasting with low-rank Gaussian Copula Processes. In *Advances in Neural Information Processing Systems*, Vol. 32. <https://doi.org/10.48550/arXiv.1910.03002>
 - [35] Rajat Sen, Hsiang-Fu Yu, and Inderjit S Dhillon. 2019. Think Globally, Act Locally: A Deep Neural Network Approach to High-Dimensional Time Series Forecasting. In *Advances in Neural Information Processing Systems*, Vol. 32. <https://doi.org/10.48550/arXiv.1905.03806>
 - [36] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is All you Need. In *Advances in Neural Information Processing Systems*, Vol. 30. <https://doi.org/10.48550/arXiv.1706.03762>
 - [37] G. Vidyamurthy. 2004. *Pairs Trading: Quantitative Methods and Analysis*. John Wiley & Sons.
 - [38] Joongyeub Yeo and George Papanicolaou. 2017. Risk control of mean-reversion time in statistical arbitrage. *Risk and Decision Analysis* 6, 4 (2017), 263–290. <http://dx.doi.org/10.2139/ssrn.2868525>
 - [39] Yunhao Zhang and Junchi Yan. 2023. Crossformer: Transformer Utilizing Cross-Dimension Dependency for Multivariate Time Series Forecasting. In *The Eleventh International Conference on Learning Representations*. <https://openreview.net/forum?id=vSVLM2j9eie>

A Sequence Model Details

We use LongConv [15] as the sequence model on the residual time series. Given input to the LongConv layer $u \in \mathbb{R}^{N \times d \times T}$ with N assets, hidden dimension d , and sequence length T , a learnable long convolution kernel $\mathcal{K} \in \mathbb{R}^{d \times T}$ and skip parameter $D \in \mathbb{R}^d$, the LongConv computes

$$y = \mathcal{K} * u + D \odot u,$$

where $*$ denotes a convolution along the temporal dimension, and \odot denotes element-wise multiplication. The convolution is defined as

$$(\mathcal{K} * u)[i] = \sum_j u[j] \mathcal{K}[i - j],$$

which has a direct computational complexity of $O(T^2)$ for a sequence of length T . In practice, we compute it efficiently using the

FFT convolution theorem:

$$\mathcal{K} * u = \mathcal{F}^{-1}(\mathcal{F}u \odot \mathcal{F}\mathcal{K}),$$

which reduces the complexity to $O(T \log T)$. Here, \mathcal{F} and \mathcal{F}^{-1} denote the discrete Fourier transform and its inverse.

Kernel regularization. Following [15], we apply the element-wise *Squash* operator to \mathcal{K} as a simple regularizer in the model forward pass:

$$\tilde{\mathcal{K}} = \text{sign}(\mathcal{K}) \odot \max(|\mathcal{K}| - \lambda_{\text{squash}}, 0),$$

with regularization strength λ_{squash} . This acts as a proximal step for an ℓ_1 penalty and regularizes the kernel by shrinking all weights and setting small weights to zero, making the kernel more sparse.

Initialization. The kernel is initialized with a geometric decay across both the sequence and hidden dimensions following [15]:

$$\mathcal{K}_t^{(h)} = x \exp\left(-\frac{t}{T} \left(\frac{d}{2}\right)^{\frac{h}{d}}\right), \quad x \sim \mathcal{N}(0, 1),$$

for $1 \leq t \leq T$ and $1 \leq h \leq d$, which gives convolution filters that act on both short and long time-scales.

B Training Details

Model parameters. We use the last two years of the first training window to select tuning parameters, and find that our results are robust to the tuning parameter selection. The selected optimal parameters are shown in Table 4.

Table 4: Selected tuning parameters

Parameter	Value	Description
Hidden dim (d)	32	Model hidden dimension
Dropout	0.1	Model dropout
d_x	32	Model attention dim
Epochs	30	Number of passes over the data
Nr layers	1	Number of layers for sequence model
λ_{VAR}	100	Weight on variance term in loss
LR	0.003	Learning rate
Weight decay	0.05	Adam weight decay in LongConv model
LongConv init	Geom Decay	Kernel initialization of the LongConv model
λ_{squash}	0.001	LongConv Squash operator strength

Computational Setup. All the results in the paper are obtained on a Linux cluster with 5 NVIDIA RTX A6000 GPUs, each with 49140 MB memory, running on CUDA Version 12.5. The cluster is equipped with two AMD EPYC 7763 64-Core Processors (128 physical cores, 256 threads total) and 1 TB of RAM.