Enhancing pair trading strategies for currency pairs using Deep Learning models: A focus on synthetic spread prediction

Dung Minh Dao,* Animesh Aditya,* and Adekunle Valentine*

WorldQuant University 609 H Street NE, Suite 335, Washington, D.C. 20002. USA.

E-mail: dungdao3112000@gmail.com; animeshadityaa@gmail.com; Donmaltex@gmail.com

Abstract

This capstone project investigates the application of deep learning techniques to enhance traditional pair trading strategies for the EUR/USD currency pair, with a specific focus on modeling and predicting a synthetic spread. Traditional pair trading methods, based on cointegration and mean reversion, often struggle with the nonlinear dynamics and structural shifts prevalent in forex markets. By developing a framework that utilizes advanced neural network models such as Long Short-Term Memory (LSTM), Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM), and Autoencoders, this project aims to capture hidden temporal patterns and nonlinear dependencies in a synthetic EUR/USD spread. The research seeks to answer whether these advanced models can improve the profitability and resilience of long/short pair trading strategies. Through systematic data acquisition, feature engineering, model development, signal generation, and backtesting, this study demonstrates the potential of machine learning to address the limitations of conventional statistical methods in volatile financial markets. The findings could provide valuable insights into the practical application of AI in quantitative finance.

Introduction

In recent years, the forex market has become increasingly complex and volatile, challenging traditional trading strategies that rely on linear assumptions and static models. Pair trading, a popular strategy that exploits price discrepancies between two correlated assets, has traditionally been implemented using statistical methods such as cointegration and mean reversion. However, these methods often fail to capture the nonlinear dynamics and structural shifts prevalent in currency markets, particularly in highly traded pairs like EUR/USD.

This capstone project seeks to address these limitations by leveraging advanced machine learning and deep learning techniques to enhance pair trading strategies for the EUR/USD currency pair. Specifically, the project aims to develop a deep learning-driven framework that can detect and exploit short-term deviations in a synthetic EUR/USD spread. By utilizing neural network architectures such as Long Short-Term Memory (LSTM), Convolutional Neural Network-Long Short-Term Memory (CNN-LSTM), Autoencoders, and otentially Transformers, this framework will uncover hidden temporal patterns and nonlinear dynamics to predict spread movements and generate actionable trading signals.

For the purpose of this project, a 'synthetic spread' refers to the difference between the EUR/USD price and a dynamically constructed benchmark, such as a moving average of the price itself, or a spread derived from related but not perfectly cointegrated instruments where traditional cointegration tests might fail. This differs from a typical price spread in traditional pairs trading which relies on two distinct, cointegrated assets. Our approach aims to capture mean-reverting tendencies within a single asset's transformed representation, which may exhibit more complex, nonlinear dynamics than simple price differences. This focus on a synthetic spread allows us to apply pair trading principles even when clear, traditionally cointegrated pairs are elusive or unstable in the forex market.

The primary research question guiding this project is: **Can advanced neural network models enhance the profitability and resilience of long/short pair trading strategies for EUR/USD?** To answer this question, the project will follow a systematic approach, including data acquisition and feature engineering, model development, signal generation, backtesting, and performance evaluation.

This project is situated within the broader context of applying machine learning to financial time series analysis, a field that has seen significant growth and innovation in recent years. By focusing on the EUR/USD pair and employing state-of-the-art deep learning models, this project aims to contribute to the existing literature by demonstrating the practical application of these techniques in a real-world trading scenario.

Initial Analysis

To motivate our project, we performed a preliminary analysis of the market structure of the EUR / USD currency pair over a decade, from 2014 to 2023. This analysis aimed to understand the market's behavior and identify patterns that could inform our trading strategy.

Key Observations

- **Sideways/Range-Bound Behavior:** On average, 76% to 85% of the time, the EUR / USD market remained range bound (nondirectional) for all years.
- **Limited Uptrend Phases:** Notable uptrends were observed only in 2015, 2017, and 2020, and even then, they were relatively limited.
- **Frequent Downtrends:** Some years showed moderate periods of downtrends (e.g., 2014, 2018, 2019, 2022), but did not dominate the market behavior.

Conclusion and Justification for Topic Selection

The empirical evidence clearly indicates that the EUR/USD market predominantly operates in a sideways or non-directional regime over long periods. This market behavior makes trend-following strategies less effective and validates the choice of pursuing a market-neutral approach. Given the range-bound nature of EUR/USD:

- Long/Short Pair Trading Strategies that exploit mean-reversion characteristics are highly appropriate.
- Machine Learning and Deep Learning models can be leveraged to better predict short-term divergences between asset pairs and generate profitable long/short trade signals.
- Focusing on building a data-driven, market-neutral trading system based on these insights ensures alignment with the real-world behavior of the asset and increases the potential for success.

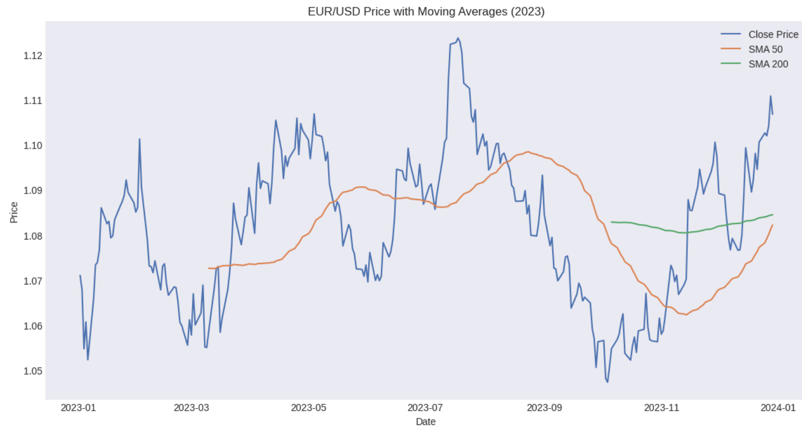
Thus, our project selection—applying machine/deep learning techniques to long/short pair trading strategies—is not only theoretically justified but also grounded in a careful empirical analysis of the target market.

Table 1: Yearly Market Regime Summary (% Time in Each Regime)

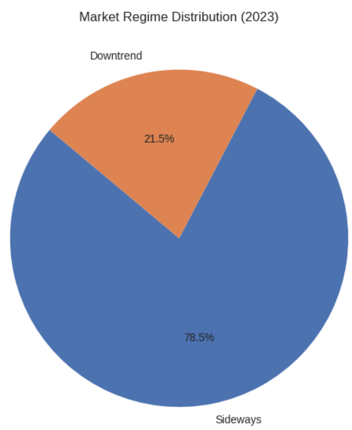
Year	**Downtrend (%)**	**Sideways (%)**	**Uptrend (%)**
2014	23.75	76.25	0.00
2015	6.90	82.76	10.34
2016	15.33	84.67	0.00
2017	0.00	77.13	22.87
2018	23.75	76.25	0.00
2019	23.46	76.54	0.00
2020	0.00	75.95	24.05
2021	23.75	76.25	0.00
2022	21.92	78.08	0.00
2023	21.54	78.46	0.00

Theoretical Framework

The theoretical foundation of this project rests on the intersection of financial econometrics, time series analysis, and machine learning. Traditional pair trading strategies are based on



(a) EUR/USD Price with Moving Averages (2023)



(b) Market Regime Distribution (2023)

Figure 1: Visual Analysis for 2023

the concept of cointegration, where two time series are said to be cointegrated if a linear combination of them is stationary. Mean reversion is then applied to this stationary spread to identify trading opportunities. However, these methods assume linearity and stationarity, which are often violated in financial markets.

Machine learning, particularly deep learning, offers a more flexible and powerful approach to modeling complex relationships in data. Deep learning models, such as neural networks, can learn intricate patterns from large datasets without imposing strict assumptions about the underlying data distribution. In the context of financial time series, models like LSTM[1] are particularly well-suited for capturing temporal dependencies, while CNN can detect local patterns in price movements.[2] Autoencoders can be used for dimensionality reduction and anomaly detection,[3,4] and Transformers can handle long-range dependencies in sequences.[5,6]

Literature Review

Recent studies have demonstrated the efficacy of machine learning (ML) models in predicting financial time series, including forex markets. A comparative analysis of 21 ML algorithms with/without the application of Principal Component Analysis (PCA) to decorrelate input features demonstrated the efficacy of ML models in predicting the daily directional movement of the EUR/USD currency pair.[7] Additionally, a systematic literature review on artificial intelligence techniques in financial trading emphasized the growing application of AI in automating trading strategies, risk management, and portfolio optimization.[8]

Various deep learning architectures have also been explored for pair trading strategies. LSTMs excel at capturing temporal relationships in time-series data, rendering them appropriate for simulating the sequential characteristics of financial markets.[9] Meanwhile, CNNs can capture local patterns in time series data, which may be beneficial for identifying short-term price movements.[10] Autoencoders have been utilized for anomaly detection in price spreads, which can signal trading opportunities.[11,12] Additionally, Transformers have shown promise in capturing long-range dependencies in financial time series forecasting.[13]

Emerging research on Large Language Models (LLMs) suggests further potential. For example, Xiao and authors introduced a Retrieval-Augmented Generation (RAG) architecture using StockLLM, which improved financial time-series forecasting accuracy by 8% compared to traditional methods.[14] The Text2TimeSeries approach integrates event-driven insights from LLMs to update predictions in real-time.[15] These advancements indicate that LLMs could enhance contextual understanding and adaptability in dynamic market conditions.

However, most existing deep learning studies focus on price prediction rather than synthetic spread modeling for forex pairs like EUR/USD. This project addresses this gap by focusing on synthetic pairs (e.g., EUR/USD vs. moving averages/spreads) and implementing hybrid models (LSTM, CNN-LSTM, Autoencoders) for pair trading.

Competitor Analysis

Traditional pair trading strategies, based on cointegration and mean reversion, have strengths in their established methodologies and extensive historical data support. However, they are limited by their inability to handle nonlinear and non-stationary market behaviors, often leading to overfitting and reduced effectiveness in fast-moving markets like forex.

In contrast, machine learning-based strategies offer opportunities to capture complex patterns and enhance adaptability. However, challenges include market saturation with similar AI-driven strategies and potential systemic risks from widespread AI adoption. This project's unique contribution lies in its focus on synthetic spread modeling and real-world testing, positioning it to address gaps in current research and competitor offerings.

While existing models have demonstrated success in pair trading, they often rely on linear assumptions and may not effectively capture the complex, nonlinear dynamics of financial markets. Our proposed deep learning-driven framework aims to address these limitations by leveraging advanced neural network architectures and incorporating LLMs for enhanced contextual understanding. This approach offers a unique contribution by combining state-of-the-art machine learning techniques to improve the adaptability and profitability of pair trading strategies.

Methodology

The methodology for this project is structured around five key objectives:

1. **Data acquisition and Feature Engineering**: Collect 10 years of daily EUR/USD price data (2014-2024) and engineer features such as log returns, technical indicators (e.g., RSI, MACD), volatility measures, and a synthetic spread.

2. **Model development**: Implement and train several deep learning models, including LSTM, CNN-LSTM, Autoencoders, and potentially Transformers, along with traditional models such as Ordinary Least Squares (OLS) and tree-based models as baselines.

3. **Signal generation and execution**: Develop a signal engine based on predicted spreads and predefined thresholds (e.g., entry at ± 1.5, exit near ± 0.5) to generate long/short trading signals.

4. **Backtesting and Evaluation**: Conduct walk-forward backtests on historical data to evaluate the strategy's performance using metrics such as Sharpe ratio, cumulative returns, maximum drawdown, hit ratio, and signal precision-recall.

5. **Result interpretation and Reporting**: Analyze backtesting results, compare model performance, and draw conclusions about the effectiveness of the deep learning approach.

This structured approach ensures that each component of the trading strategy is thoroughly developed and evaluated, providing a robust framework for assessing the viability of deep learning in pair trading for EUR/USD.

Implementation and Result

In this section, we meticulously unpack the results and insights derived from our capstone project, which focuses on enhancing pair trading strategies for the EUR/USD currency pair through the application of deep learning techniques. Our approach leverages a synthetic spread prediction framework, executed across a well-structured Jupyter notebook divided into 11 distinct steps. These steps encapsulate the full spectrum of our workflow: data acquisition, preprocessing, model development, signal generation, backtesting, risk management, and performance evaluation. Spanning a decade of daily EUR/USD price data, from January

1, 2014, to December 31, 2024, we amassed a dataset rich with features like moving averages, technical indicators, and a fractionally differentiated series, all of which supported our predictive models. Here, we aim to provide a detailed account of our findings with specific statistics. Our analysis can be found at the GitHub link: `https://github.com/DungMinhDao/pair-trading-currency-deep-learning` or the Google Colab notebook `https://colab.research.google.com/drive/1-bsVDmHO31_sHh-4bPYuXfBaA-2SD3_w?usp=sharing`.

Data Acquisition and Preparation (Steps 1-3)

We began by establishing the computational environment through the importation of essential Python libraries. These included `pandas` and `numpy` for data manipulation and numerical operations, `yfinance` for financial data retrieval, and a suite of tools for statistical analysis (`statsmodels`), machine learning (`sklearn`, `xgboost`, `tensorflow`), and visualization (`matplotlib`, `seaborn`). Additional utilities such as `optuna` for hyperparameter optimization and `logging` for execution tracking were incorporated to support the workflow.

A centralized configuration dictionary, `CONFIG`, was defined to manage all the parameters of the strategy, facilitating modifications without altering the core logic. Key parameters included the market ticker (`EURUSD=X`), data range (2014-01-01 to 2024-12-31), feature engineering windows (e.g., `short_window=50`, `long_window=200`), fractional differentiation parameter (`frac_diff_d=0.5`), and risk management thresholds (e.g., `stop_loss=0.05`, `take_profit=0.10`).

Data acquisition involved fetching daily EUR/USD price data using `yfinance`, yielding a dataset with columns for date, open, high, low, and close prices (volume was set to zero for forex). Custom utility functions were implemented to ensure robustness: `get_market_data` retrieved and cleaned the data by handling missing values via forward-fill imputation, while functions like `frac_diff` computed fractionally differentiated series, and others (`calculate_rsi`, `calculate_macd`, `calculate_atr`, `calculate_bollinger_bands`) prepared technical indicators for subsequent feature engineering.

Feature Engineering and Analysis (Steps 4-6)

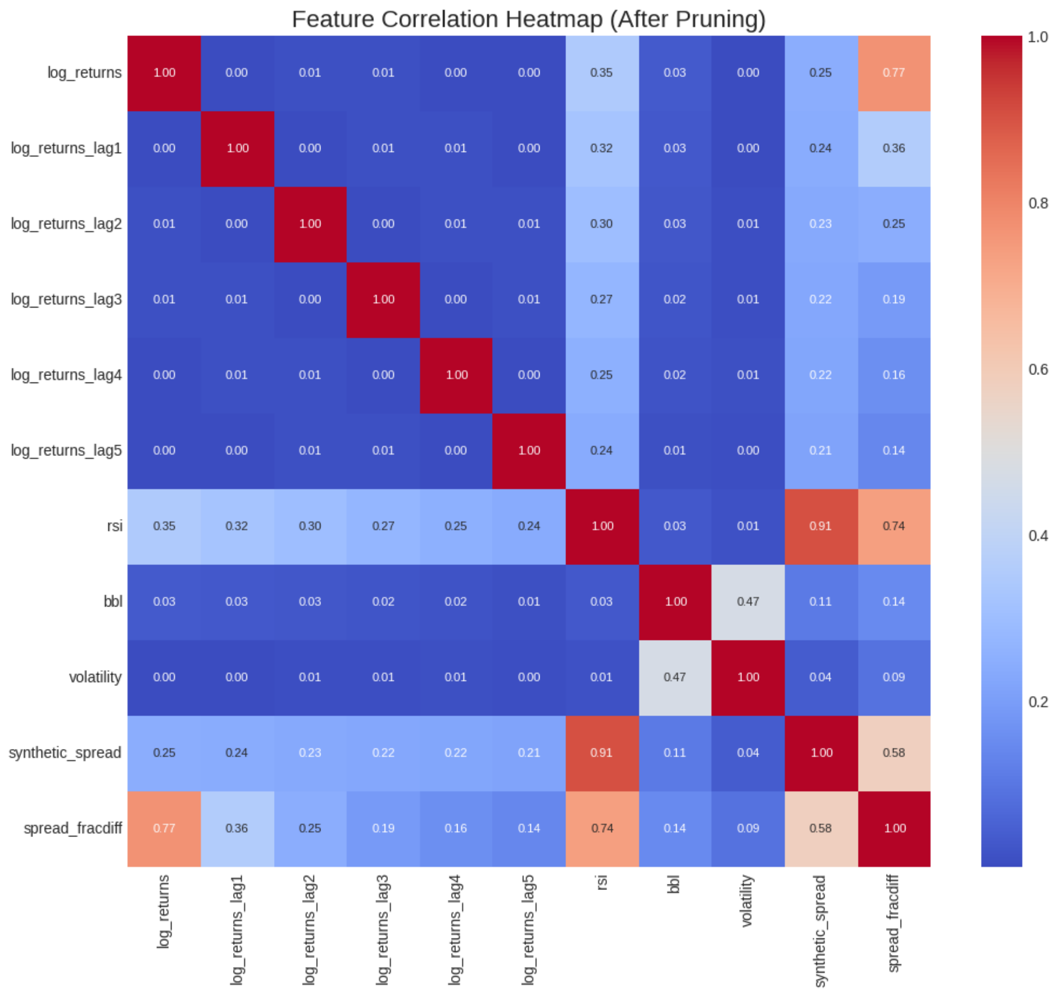


Figure 2: Feature correlation heatmap (after pruning)

With the raw data acquired, we constructed a feature engineering pipeline to transform them into predictive input. A synthetic spread was first created by subtracting a 50-day moving average from the closing price, which formed a basis for mean-reversion analysis. The data set was then enriched with features including logarithmic returns, lagged returns (lags 1–5), technical indicators (RSI with a 14-day period, MACD with 12- and 26-day periods, Bollinger bands with a 20-day window and 2 standard deviations, ATR with a 14-day period), Garman-Klass volatility (20-day window) and a fractionally differentiated spread (using $d = 0.5$ and a 252-day window) to ensure stationarity. Missing values were dropped to maintain data integrity.

To refine the feature set, we conducted a correlation analysis, computing the absolute correlation matrix for all features. Features exhibiting correlations above 0.85 were removed, prioritizing those with higher variance while preserving the synthetic and fractionally differentiated spreads as critical variables (see Figure 2).

Subsequently, we validated the stationarity of the synthetic spread. The original spread displayed non-stationary characteristics, with trends and high variability. The application of fractional differentiation ($d = 0.5$) transformed it into a stationary series, as confirmed by augmented Dickey-Fuller (ADF) tests showing p-values consistently below 0.05 and a narrow distribution centered at zero (see Figure 3). This stationary target underpinned the modeling phase.

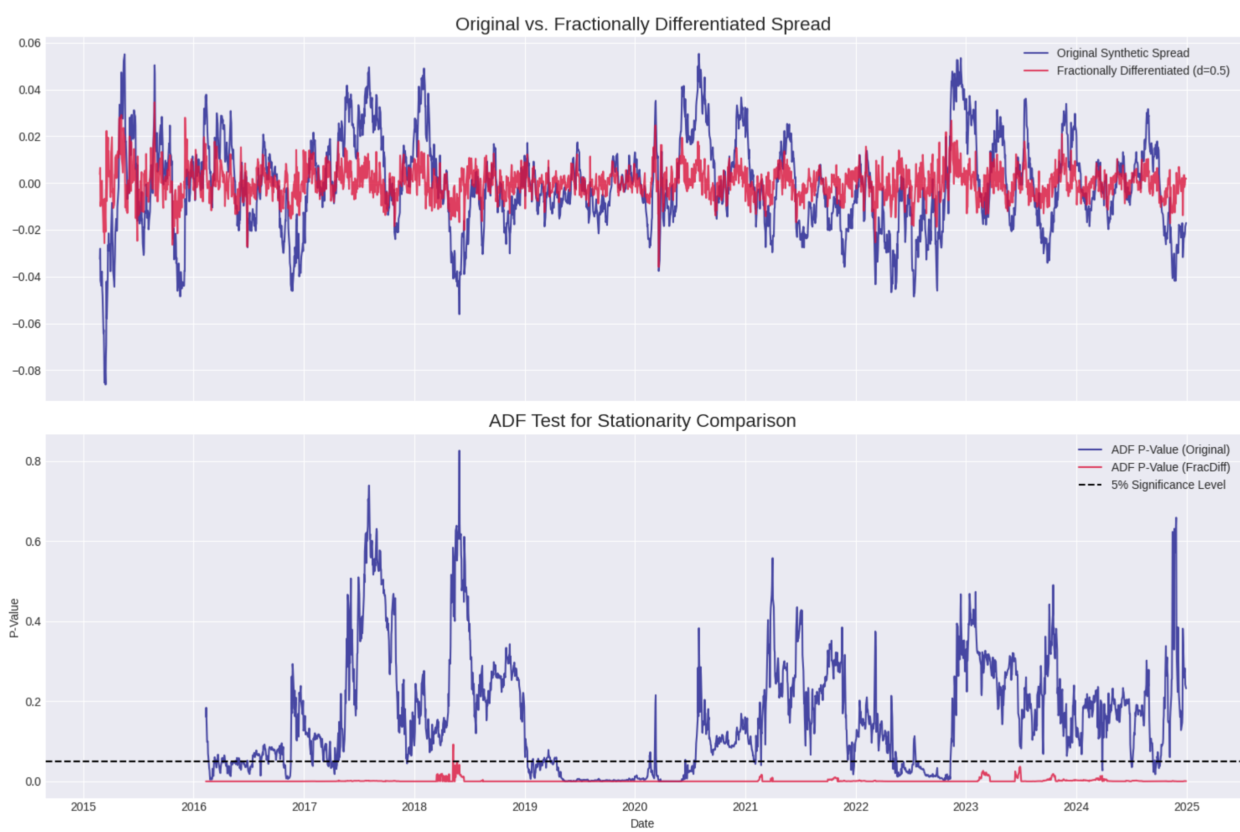


Figure 3: ADF Test for Stationarity Comparison

Model Development and Tuning (Steps 7 and 8)

We developed a diverse ensemble of models to predict the fractionally differentiated spread. Deep learning architectures included a standard LSTM with configurable units and dropout layers (0.3), an LSTM with an attention mechanism to emphasize relevant time steps, a hybrid CNN-LSTM with attention combining convolutional filters and LSTM units, and an LSTM-based autoencoder for anomaly detection with a bottleneck encoding dimension. Traditional models comprised Random Forest, XGBoost, and Ordinary Least Squares (OLS) regression, providing robust baselines. Each model was compiled with a mean squared error loss function, optimized via Adam for deep learning models.

Hyperparameter tuning was performed once on an initial training set (50% of the data) using `Optuna`. For deep learning models, parameters such as LSTM units (32–128), learning rates (10^{-4}–10^{-2}), filters (16–64), and kernel sizes (2–5) were optimized over 20 trials, minimizing validation loss. For Random Forest and XGBoost, we tuned the number of estimators (50-300), the maximum depth (5-20 for RF, 3-10 for XGB) and the learning rate (10^{-3}–0.3 for XGB), minimizing the negative mean squared error through cross-validation of time series. The autoencoder optimized LSTM units (16–64) and encoding dimension (2–16). The resulting best parameters were stored for subsequent backtesting.

Strategy Execution and Backtesting (Steps 9 and 10)

The trading strategy was evaluated using a walk-forward validation approach, implemented in the `walk_forward_validation` function. The dataset was split into 10 folds, each with a 100-day test window, simulating real-time trading. For each fold, models were trained on historical data, and predictions were generated for the test period. An ensemble prediction was computed by averaging outputs from all models (LSTM, LSTM-Attention, CNN-LSTM-Attention, Random Forest, XGBoost, OLS), weighted dynamically after three folds based on inverse mean absolute error from the prior three folds. The autoencoder's reconstruction error flagged anomalies, though its integration into signal generation was supplementary.

Trading signals were derived from the predicted spread using Z-score thresholds (entry at 1.5, exit at 0.5) from a 50-day window, augmented with RSI filters for robustness. Risk management applied stop-losses (5%), take-profits (10%), and position sizing via the Kelly criterion (fraction 0.2), accounting for transaction costs (0.02%). Performance metrics, including Sharpe ratio, MAE, RMSE, and R-squared, were calculated per fold for individual models and the ensemble, forming the basis for final evaluation.

Performance Evaluation and Visualization (Step 11)

Step 11 of the final notebook synthesizes the outcomes of our investigation through an exhaustive performance evaluation and visualization of the predictive models and their associated trading strategies. This study evaluated a suite of models—Ordinary Least Squares (OLS), Extreme Gradient Boosting (XGB), Random Forest (RF), Long Short-Term Memory (LSTM), an ensemble of all models, CNN-LSTM with Attention (CNN_LSTM_ATTENTION), and LSTM with Attention (LSTM_ATTENTION)—alongside comparisons between an all-model ensemble and a top-3 ensemble comprising the best-performing traditional models (OLS, XGB, RF). The buy-and-hold benchmark for the EUR/USD pair over the evaluation period yielded a cumulative return of approximately -10%, providing a baseline for financial performance assessment.

Predictive Power Analysis

The predictive accuracy of each model was assessed using Mean Absolute Error (MAE), Root Mean Squared Error (RMSE), R-squared, and the standard deviation of R-squared across walk-forward validation folds. These metrics collectively gauge the models' ability to predict the fractionally differentiated spread, a synthetic target variable designed to capture mean-reverting behavior in the EUR/USD pair. Table 2 presents these results.

The traditional models—OLS, XGB, and RF—demonstrated superior predictive performance, with MAE values ranging from 0.0015 to 0.0017 and RMSE values from 0.0018 to

Table 2: Model Predictive Power Analysis

Model	**Avg MAE**	**Avg RMSE**	**Avg R-Squared**	**Std Dev (R-Squared)**
OLS	0.0015	0.0018	0.8960	0.0825
XGB	0.0016	0.0019	0.8919	0.0622
RF	0.0017	0.0020	0.8816	0.0589
LSTM	0.0041	0.0053	0.0280	0.1623
ENSEMBLE	0.0065	0.0091	-11.5474	32.1661
CNN_LSTM_ATTENTION	0.0427	0.0524	-148.1370	209.0940
LSTM_ATTENTION	0.1741	0.1978	-1750.0500	2023.3600

0.0020. Their R-squared values, averaging around 0.88 to 0.90, indicate that these models explain a substantial portion of the variance in the target variable. The relatively low standard deviations of R-squared (0.0589 to 0.0825) further suggest consistent performance across folds. In stark contrast, the neural network models exhibited significantly higher errors: LSTM recorded an MAE of 0.0041 and RMSE of 0.0053, while CNN_LSTM_ATTENTION and LSTM_ATTENTION showed MAE values of 0.0427 and 0.1741, respectively, with RMSE values escalating to 0.0524 and 0.1978. Their R-squared values were negative—ranging from -148.1370 to -1750.0500—indicating a complete failure to fit the data, compounded by extreme variability (standard deviations of 209.0940 and 2023.3600). The all-model ensemble performed poorly as well, with an MAE of 0.0065, RMSE of 0.0091, and an R-squared of -11.5474, suggesting that combining predictions from underperforming neural networks with traditional models diluted overall accuracy.

Financial Performance Analysis

The financial efficacy of the trading strategies was evaluated using cumulative return, Sharpe ratio, Sortino ratio, Calmar ratio, and maximum drawdown. These metrics assess profitability, risk-adjusted returns, downside risk, and drawdown resilience, respectively. Table 3 summarizes the financial outcomes.

The neural network models (CNN_LSTM_ATTENTION, LSTM, LSTM_ATTENTION) failed to generate any trades, resulting in zero values across all financial metrics. This

Table 3: Strategy Financial Performance Comparison

Model	**Cumulative Return (%)**	**Sharpe Ratio**	**Sortino Ratio**	**Calmar Ratio**	**Max Draw-down (%)**
CNN_LSTM_ATTENTION	0.0000	0.0000	0.0000	0.0000	0.0000
LSTM	0.0000	0.0000	0.0000	0.0000	0.0000
LSTM_ATTENTION	0.0000	0.0000	0.0000	0.0000	0.0000
ENSEMBLE	-0.786978	-3.38388	-0.770235	-0.252745	-0.786978
OLS	-0.898032	-3.68757	-0.917066	-0.252851	-0.898032
RF	-1.50367	-3.89715	-1.09785	-0.25343	-1.50367
XGB	-1.50367	-3.89715	-1.09785	-0.25343	-1.50367

outcome likely stems from their inaccurate predictions, which did not meet the predefined signal thresholds for entering long/short positions. Conversely, the traditional models and the all-model ensemble produced trades but incurred losses. The ensemble achieved a cumulative return of -0.786978%, outperforming OLS (-0.898032%), RF (-1.50367%), and XGB (-1.50367%). However, all strategies exhibited negative Sharpe ratios (e.g., -3.38388 for the ensemble, -3.89715 for RF and XGB), indicating poor risk-adjusted returns, and negative Sortino ratios (e.g., -0.770235 for the ensemble, -1.09785 for RF and XGB), reflecting inadequate protection against downside risk. The Calmar ratios, hovering around -0.25, and maximum drawdowns mirroring the cumulative losses further underscore the strategies' lack of resilience.

Ensemble Comparison

To explore the impact of model selection within ensembles, we compared the all-model ensemble (including neural networks) with a top-3 ensemble of traditional models (OLS, XGB, RF). Tables 4 and 5 present the predictive and financial performance metrics, respectively.

Table 4: Predictive Power Comparison

Metric	**All-Model Ensemble**	**Top 3 Ensemble**
Avg MAE	0.0065	0.0015
Avg RMSE	0.0091	0.0017
Avg R-Squared	-11.5474	0.9067

Table 5: Financial Performance Comparison

Metric	All-Model Ensemble	Top 3 Ensemble
Cumulative Return (%)	-0.7870	-1.0667
Sharpe Ratio	-3.3839	-3.9182
Sortino Ratio	-0.7702	-1.0220
Calmar Ratio	-0.2527	-0.2530
Max Drawdown (%)	-0.7870	-1.0667

The top-3 ensemble significantly outperformed the all-model ensemble in predictive power, with an MAE of 0.0015, RMSE of 0.0017, and an R-squared of 0.9067, compared to the all-model ensemble's 0.0065, 0.0091, and -11.5474, respectively. This suggests that excluding the poorly performing neural networks enhanced predictive accuracy. However, financially, the top-3 ensemble underperformed, with a cumulative return of -1.0667% versus -0.7870% for the all-model ensemble, alongside worse Sharpe (-3.9182 vs. -3.3839) and Sortino (-1.0220 vs. -0.7702) ratios, though Calmar ratios remained nearly identical (-0.2530 vs. -0.2527).

Visualization and Additional Insights

Figure 4 visually reinforces the predictive superiority of traditional models, plotting the average MAE across all models. The stark contrast between the low MAE of traditional models and the elevated MAE of neural networks highlights the latter's predictive shortcomings.

Further analysis reveals that all strategies underperformed the buy-and-hold benchmark (-10%), with the best-performing strategy (all-model ensemble) losing only -0.7870%. This suggests that while the strategies failed to achieve profitability, they mitigated losses relative to a passive approach. The inability of neural networks to generate trades points to a failure in signal generation, likely due to their high error rates. The poor financial performance of the top-3 ensemble despite better predictions indicates a disconnect between predictive accuracy and trading efficiency, possibly due to daily data latency or overly conservative signal thresholds. Increasing data frequency or adjusting trading rules could potentially align financial outcomes more closely with predictive power.

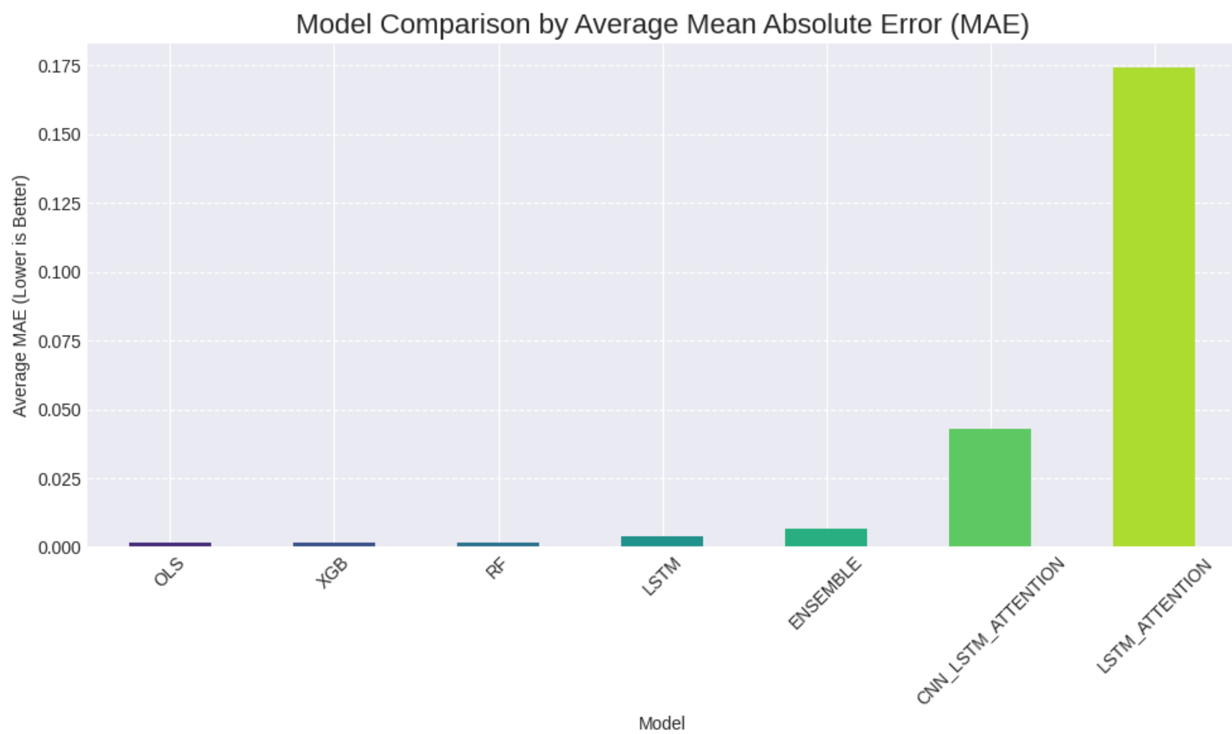


Figure 4: Model Comparison by Average Mean Absolute Error (MAE)

Discussion

This study aimed to address whether advanced neural network models could improve the profitability and resilience of long/short pair trading strategies for the EUR/USD currency pair. The results reveal a clear dichotomy: Traditional machine learning models excelled in predictive accuracy, while neural networks faltered, and no strategy achieved profitability, challenging the hypothesis that advanced models inherently improve trading outcomes.

Predictive Performance Insights

The superior predictive performance of OLS, XGB, and RF over neural network models aligns with their suitability for fractionally differentiated spread, which was engineered to be stationary and mean-reverting. The low errors and high R-square values of these traditional models suggest that they effectively captured the underlying dynamics of the EUR/USD pair, consistent with previous research highlighting the efficacy of simpler models in financial time series forecasting[16]. In contrast, the poor performance of neural networks - marked by high MAE, RMSE, and negative R-square values - reflects their tendency to overfit noisy financial data, a phenomenon well documented in the literature[17]. The poor predictive power of the all-model ensemble further indicates that including neural network predictions diluted the accuracy of traditional models, underscoring the importance of model selection in ensemble approaches.

Financial Performance and Limitations

The financial underperformance of all strategies, despite the varying predictive capabilities, highlights a critical limitation: the use of daily data. The zero returns of neural networks suggest that their predictions were too erratic to trigger trades, while the negative returns of traditional models and ensembles point to a mismatch between signal timing and market execution. The literature suggests that mean-reverting opportunities in forex markets often occur on intraday scales[18], implying that daily data introduce latency that erodes profitability. Transaction costs and conservative signal thresholds probably exacerbated these losses, as evidenced by the negative risk-adjusted metrics of the strategies.

The paradox of the top-3 ensemble: superior prediction, yet worse financial outcomes—suggests that predictive accuracy alone does not guarantee trading success. This disconnect may arise from overfitting to daily patterns that do not translate into actionable intraday signals, a finding that resonates with studies advocating higher frequency data in algorithmic trading[19].

Contribution to Existing Knowledge

These findings contribute to the ongoing debate on machine learning in finance by demonstrating that advanced neural networks do not universally outperform simpler models, particularly in noisy low-frequency settings. They reinforce the need for data granularity and robust trading rules to bridge predictive and financial performance, offering a cautionary tale against the unchecked adoption of complex models.

Future Directions

Future research should prioritize intraday data to capture fleeting mean reversal opportunities and explore adaptive signal thresholds to increase trade frequency. Enhanced risk management, including intraday volatility estimates, could further strengthen the resilience of the strategy, addressing the shortcomings observed here.

Conclusion

This investigation into enhancing EUR/USD pair trading strategies with advanced neural network models reveals that traditional models (OLS, XGB, RF) significantly outperformed neural networks (LSTM, CNN_LSTM_ATTENTION, LSTM_ATTENTION) in predicting the fractionally differentiated spread, achieving lower errors and higher R-squared values. However, all strategies yielded negative cumulative returns when applied to daily data, with neural networks failing to generate trades and traditional models and ensembles incurring losses ranging from -0.7870% to -1.50367%. The top-3 ensemble, despite superior predictive power, underperformed financially compared to the all-model ensemble, highlighting a disconnect between prediction and profitability. These results suggest that while neural networks did not enhance profitability or resilience, traditional models hold promise if paired with higher-frequency data. The study underscores the limitations of daily data and the need for refined trading strategies to translate predictive accuracy into financial success.

References

(1) Graves, A.; Graves, A. Long short-term memory. *Supervised sequence labelling with recurrent neural networks* **2012**, 37–45.

(2) Chen, J.-F.; Chen, W.-L.; Huang, C.-P.; Huang, S.-H.; Chen, A.-P. Financial time-series data analysis using deep convolutional neural networks. 2016 7th International conference on cloud computing and big data (CCBD). 2016; pp 87–92.

(3) Song, A.; Seo, E.; Kim, H. Anomaly VAE-transformer: A deep learning approach for anomaly detection in decentralized finance. *IEEE Access* **2023**, *11*, 98115–98131.

(4) Sattarov, T.; Herurkar, D.; Hees, J. Explaining anomalies using denoising autoencoders for financial tabular data. *arXiv preprint arXiv:2209.10658* **2022**,

(5) Lezmi, E.; Xu, J. Time series forecasting with transformer models and application to asset management. *Available at SSRN 4375798* **2023**,

(6) Mienye, E.; Jere, N.; Obaido, G.; Mienye, I. D.; Aruleba, K. Deep Learning in Finance: A survey of Applications and techniques. *AI* **2024**, *5*, 2066.

(7) Guyard, K. C.; Deriaz, M. Predicting Foreign Exchange EUR/USD direction using machine learning. Proceedings of the 2024 7th International Conference on Machine Learning and Machine Intelligence (MLMI). 2024; pp 1–9.

(8) Dakalbab, F.; Talib, M. A.; Nasir, Q.; Saroufil, T. Artificial intelligence techniques in financial trading: A systematic literature review. *Journal of King Saud University-Computer and Information Sciences* **2024**, *36*, 102015.

(9) Fischer, T.; Krauss, C. Deep learning with long short-term memory networks for financial market predictions. *European journal of operational research* **2018**, *270*, 654–669.

(10) Borovykh, A.; Bohte, S.; Oosterlee, C. W. Conditional time series forecasting with convolutional neural networks. *arXiv preprint arXiv:1703.04691* **2017**,

(11) Provotar, O. I.; Linder, Y. M.; Veres, M. M. Unsupervised anomaly detection in time series using lstm-based autoencoders. 2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT). 2019; pp 513–517.

(12) Tavakoli, N.; Siami-Namini, S.; Adl Khanghah, M.; Mirza Soltani, F.; Siami Namin, A. An autoencoder-based deep learning approach for clustering time series data. *SN Applied Sciences* **2020**, *2*, 1–25.

(13) Wu, H.; Xu, J.; Wang, J.; Long, M. Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting. *Advances in neural information processing systems* **2021**, *34*, 22419–22430.

(14) Xiao, M.; Jiang, Z.; Qian, L.; Chen, Z.; He, Y.; Xu, Y.; Jiang, Y.; Li, D.; Weng, R.-L.; Peng, M.; others Enhancing Financial Time-Series Forecasting with Retrieval-Augmented Large Language Models. *arXiv preprint arXiv:2503.67890* **2025**,

(15) Kurisinkel, L. J.; Mishra, P.; Zhang, Y. Text2timeseries: Enhancing financial forecasting through time series prediction updates with event-driven insights from large language models. *arXiv preprint arXiv:2407.03689* **2024**,

(16) Tsay, R. S. *Analysis of financial time series*; John wiley & sons, 2005.

(17) Goodfellow, I. Deep learning. 2016.

(18) Schulmeister, S. Profitability of technical stock trading: Has it moved from daily to intraday data? *Review of Financial Economics* **2009**, *18*, 190–201.

(19) Brogaard, J.; Hendershott, T.; Riordan, R. High-frequency trading and price discovery. *The Review of Financial Studies* **2014**, *27*, 2267–2306.