Evaluating Effectiveness of Nonlinear Dimensionality Reduction in Hedge Funds' Returns Forecasting

Milica Zukanović
0000-0003-3650-8327
University of Belgrade – Faculty of Organizational Sciences, Jove Ilića 154, 11000 Belgrade, Serbia
Email: milica.zukanovic@fon.bg.ac.rs

Aleksa Radosavčević
0009-0006-0069-2117
University of Belgrade – Faculty of Organizational Sciences, Jove Ilića 154, 11000 Belgrade, Serbia
Email: ar20235065@student.fon.bg.ac.rs

Ana Poledica, Pavle Milošević, Ivan Luković
0000-0002-5929-6446
0000-0002-5943-6023
0000-0003-1319-488X
University of Belgrade – Faculty of Organizational Sciences, Jove Ilića 154, 11000 Belgrade, Serbia
Email: {ana.poledica; pavle.milosevic; ivan.lukovic}@fon.bg.ac.rs

Abstract—**Hedge funds (HF) are actively managed investment vehicles employing diverse and often complex strategies. Accurate returns forecasting is essential for optimizing their performance and managing risk. This paper investigates the application of nonlinear dimensionality reduction (DR) methods in forecasting HF strategy performance, building upon prior work in financial time series analysis. We evaluate the effectiveness of Kernel Principal Component Analysis (KPCA), t-Distributed Stochastic Neighbor Embedding (t-SNE), Uniform Manifold Approximation and Projection (UMAP), and autoencoders on predictive performance of machine learning models. The extracted features are fed into several forecasting models, Support Vector Machine (SVM) with linear and nonlinear kernels, Neural Network (NN), and Extreme Gradient Boosting (XGB), to predict returns of five diverse HF investment strategies: Commodity Trading Advisors, Equity Long Short, Equity Market Neutral, Fixed Income Arbitrage, and Global Macro. The results demonstrate that nonlinear DR methods, particularly autoencoders, and KPCA combined with NN, significantly outperform other techniques. Our findings highlight the value of nonlinear transformations in enhancing predictive accuracy for HF returns time series.**

Index Terms— **dimensionality reduction, hedge funds, PCA, KPCA, t-SNE, UMAP, autoencoders.**

I. Introduction

THE hedge fund industry has experienced substantial growth [1], due to investors' demand for alternative sources of return and portfolio diversification. Compared to traditional financial market forecasting, hedge funds introduce further layers of complexity, including the lack of transparency, and very diverse strategies [2]. A wide array of factors in investment models often lead to noise and overfitting causing poor decision-making. However, by effectively using DR techniques, hedge fund analysts can enhance predictive model accuracy, gain deeper insights into portfolio risk, and make smarter investment decisions.

DR techniques in machine learning (ML) are typically implemented for several key reasons: to mitigate the curse of dimensionality, prepare data for ML algorithms, boost model performance, and enhance interpretability and explainability [3]. DR algorithms are commonly distinguished based on linearity, interpretability, computational cost, data type, hyperparameters, etc. [4], while the success of DR algorithms is highly dependent on data characteristics, quality and size [5]. A recent survey [3] highlights approaches developed for time-series analysis e.g. autoencoders, Principal Component Analysis (PCA) and KPCA, while PCA and UMAP are commonly used for financial times series [6]. Popular nonlinear techniques, t-SNE and UMAP, are shown to be successful for both visualization and DR in financial time series analysis [7]. Finally, more transparent approaches, e.g. based on fuzzy sets and fuzzy rough sets, are proposed in literature [8]. Still, the authors conclude that while nonlinear DR approaches often yield superior results on artificial datasets, this performance does not necessarily translate to real-world datasets [5].

In forecasting, DR algorithms serve as inputs to predictive models to enhance forecasting performance. The widespread adoption of ML models in financial forecasting is driven by their ability to capture nonlinear relationships and adapt to rapidly changing market conditions. According to a recent review [9], the predominant ML algorithms for stock market prediction are SVM and NN, followed by deep learning models. In [10], the authors demonstrate benefits of using ML approach to forecasting stock returns; the best performing models are boosting techniques and neural networks. Finally, many approaches tend to incorporate DR in their ML framework, e.g. [11]

In this research, we examine the effect of nonlinear DR techniques, selected from various category groups on performance of hedge fund returns forecasting. In our study, PCA will serve as a linear benchmark, and KPCA, t-SNE, UMAP and autoencoders will be exploited to uncover the underlying structure of the data, modeling nonlinearity from different perspectives. As part of ML forecasting framework, predominant ML algorithms are selected based on the surveyed research papers, i.e. linear and kernel SVM are chosen to match diverse combinations of linear/nonlinear DR algorithms, NNs as most effective in hedge fund return prediction, and XGB as advanced representative of boosting algorithms. The datasets used in this research contain time series of monthly returns for the period of 1999-2023 for 5 HF investment, diverse in types of risks, financial instruments and investment philosophy [12]. The research addresses the following questions aimed at exploring the effectiveness of nonlinear DR in case of hedge funds' returns forecasting:

RQ1: How does the selection of nonlinear DR techniques affect the predictive power of ML forecasting models for hedge funds' return time series?

RQ2: Which ML forecasting models demonstrate best performance when utilizing specific DR techniques?

RQ3: How do forecasting outcomes contribute to investment decision making for hedge fund managers in terms of explainability and meaningful insights?

By evaluating the performance of nonlinear DR in forecasting HF returns, this paper aims to address the broader investment community interested in the intersection of ML and financial modeling.

The rest of this paper is organized as follows. In Section II we outline the theoretical background of DR techniques, highlighting their benefits and drawbacks. In Section III we cover the problem setup, dataset analysis, and experimental design. Main results and discussion appear in Section IV.

II. Theoretical background

Due to the increasing challenges set by enormous amounts of financial data, dimensionality reduction is one of the necessary steps during data preprocessing. When it comes to various technical and fundamental analysis indicators, information from social networks, as well as numerous historical data, this is a significant technique for improving the performance of algorithms, reducing model training, eliminating noise and irrelevant data for specific problems.

A. Principal Component Analysis

PCA is the widely used linear orthogonal DR technique with main goal to identify the principal components (PCs) that capture the maximum variance of data [13]. During transformation, the first PC represents a linear combination of the original features and explains the biggest amount of variance. Each subsequent component is performed with the next highest score of variances.

The main advantage of PCA is its speed and ability to efficiently process large datasets, as well as a clear mathematical interpretation of the variance of the data [13]. However, the principle of linearity limits its ability to capture nonlinear relationships that are present in many real-life datasets. Hence, this inability to model nonlinearity drives the need for the development of more advanced, nonlinear dimensionality reduction techniques, such as KPCA [14].

KPCA is a nonlinear approach of generalizing PCA. Using the kernel method, KPCA first maps data to a higher dimensional space [14]:

$$\mathbb{R}^d \to \mathbb{R}^D,\ where\ D \gg d \tag{1}$$

Once the kernel method has been applied to transform data into a linearly separable form, PCA is used to dimensionality reduction [5]. KPCA uses several kernel functions like RBF kernel, polynomial and sigmoid. The choice of the kernel function plays an important role, since the performance of the methods depends on the choice.

B. t-Distributed Stochastic Neighbor Embedding

t-SNE is a nonlinear technique of DR with primary purpose to visualize high dimension datasets into low dimension spaces, usually two or three-dimension spaces [15]. Furthermore, the main aim of t-SNE is to preserve complex relationships between original data.

Firstly, t-SNE applies SNE to the dataset which assigns a higher probability to similar pairs of high-dimensional objects, and a lower probability to different data points [5]:

$$p_{a|b} = \frac{exp\frac{-\|x_b - x_a\|^2}{2\sigma^2}}{\sum_{a\neq k}\frac{-\|x_k - x_a\|^2}{2\sigma^2}} \tag{2}$$

where $p_{a|b}$ is conditional probability, x_a high-dimensional data, x_b low-dimensional data and σ^2 given variance. This is achieved with a Gaussian kernel, which represents the similarity between data points as a conditional probability. Thereafter, t-SNE defined SNE in a low-dimension map using the Student's t-distribution to avoid the crowding problem.

Even though this method is widely recognized for its excellence in visualizing and preserving local structure, t-SNE produces high computational costs and time complexity, which is limited for large datasets.

C. Uniform Manifold Approximation and Projection

UMAP is one more nonlinear method for DR based on advanced mathematical concepts, manifold theory and topological data analysis. UMAP stands out in preserving local and global data structure, which is a significant advantage over t-SNE, which focuses on local structure [16]. At the same time, UMAP overcomes the limitation of t-SNE such as execution speed and feature limitation [17].

The algorithm consists of two key steps, graph construction for the high dimensional space and optimization of low dimensional graph layout. In the first step, UMAP constructs a probability distributions graph of nearest neighbors, while the optimization step implies stochastic gradient descent on individual observations [17]. Precisely, initial construction of the graph is crucial in ensuring preservation of both local and global structure. Moreover, two hyperparameters are key for embedding results and control of structure preservation, the number of nearest neighbors that the algorithm takes and the minimum distance between data points in a low-dimensional space [17].

D. Autoencoders

An autoencoder is an unsupervised NN architecture designed to compress (encode) input data into its essential attributes and then reconstruct (decode) the original input from this compressed representation [18]. Therefore, this NN consists of three components: encoder, code and decoder [19]. The encoder is defined by the function that reduces the input data into projection within latent space, while the decoder is defined by the function that decodes the projection into its reconstruction [19]. The code part incorporates the projection as the output from the encoder to the decoder. The output is the reconstruction of the input data [18].

Training of autoencoders aims to minimize the reconstruction loss which measures the difference between the decoder's reconstruction and the original input [19]. This loss function is used to optimize model weights via gradient descent during backpropagation. Furthermore, the performance of autoencoder depends on the tuning of the hyperparameters.

One of the main advantages of autoencoders is the denoising in the data as well as the detection of anomalies and adaptation to changing distributions. However, their implementation requires careful consideration of resources, given the computational complexity.

E. Comparative analysis of dimensionality reduction methods

For easier comparison of techniques, a comparative analysis is provided in Table I. The table summarizes all relevant aspects of applied DR methods, including goal, explainability, computational complexity, linearity, topology, strengths and weaknesses. This review allows a clear insight into the limitations of each method, as well as advantages depending on the specific requirements.

III. PROBLEM STATEMENT AND METHODOLOGY

A. The problem setup

Our specific case study addresses a problem of forecasting HFs' returns using complex and broad feature space. For investment managers, both predictive accuracy and understanding of key drivers are important for informed investment decision making as they enable effective risk management and portfolio optimization. Given the inherent complexity of financial data, dimensionality reduction becomes a necessary step to simplify data, remove noise and redundancy, and adapt data for building robust and scalable forecasting models for HFs investment returns.

Thus, we address the research questions Q1-Q3 (section 1) and analyze the effectiveness of advanced nonlinear DR techniques in predictive ML models' performance i.e. accuracy and interpretability. The case study includes the analysis of HFs returns for five various investment strategies that suit different investment styles and risk tolerance, as to provide credible and general insights.

B. Datasets

As a follow-up, we further analyze the five datasets from our previous research in [12].

Similarly, this case study is intended to analyze and predict monthly returns of five hedge funds strategies from Morning CISDM Database: Commodity Trading Advisors (CTA), Equity Long Short (ELS), Equity Market Neutral (EMS), Fixed Income Arbitrage (FIA), and Global Macro (GM) strategies. Each strategy suits a different investment style and risk tolerance, allowing traders to choose approaches that align with their financial goals, time horizons, and market outlooks. CTA mostly uses managed futures contracts and trend-following approaches to forecast commodity price. ELS and EMN strategies primarily focus on analyzing and trading companies' stocks listed on stock exchanges. ELS typically involves taking long positions in undervalued stocks and short positions in overvalued ones, while also utilizing options to hedge risks and leverage to enhance potential returns. EMN aims to maintain minimal correlation with the broader equity market by hedging against factors such as currency fluctuations, sector exposure, and market volatility. FIA targets undervalued debt securities of traded companies, such as bonds and other fixed income products. To mitigate risks associated with high-yield positions, FIA may also involve equity positions in the issuing firms as a hedging mechanism. Finally, the GM strategy is based on analyzing macroeconomic trends in countries or regions, using instruments like stocks, bonds, and commodities, and managing risk through derivatives.

TABLE I.
DEFINING CHARACTERISTICS OF FIVE DIMENSIONALITY REDUCTION METHODS

Method	Goal	Explainability	Computational Complexity	Linearity	Topology	Strength	Weakness
PCA	Maximize variance	Medium	$O(d^2n+n^3)$	Linear	Random projection	Computational efficiency	Linear projection
KPCA	Linearly separate data	Low	$O(n^3)$	Nonlinear	Manifold	Capturing nonlinear patterns	High training time
t-SNE	Preserve local structure	Low	$O(n^2)$	Nonlinear	Manifold	Exceptional visualization	Provide only 2 to 3 features
UMAP	Preserve local and global structure	Low	$O(n^{1.14})$	Nonlinear	Manifold	Strong mathematical foundations	Finding the spurious structure of the multiplicity
Autoencoder	Effective compression	Low	Depends on the structure	Nonlinear	Manifold	Denoising	Training data required

Following the recommendations of financial experts and literature review, for each HF strategy we include data on four groups of factors describing trends, debt, capital market conditions, and macroeconomic aspects. Several factors were selected from each group:

- **Interest rates** are represented by a 3-month treasury bill, 5-year constant maturity, and 10-year constant maturity rates, minus federal funds rates.
- **Credit rating categories** include investment grade, high yield below investment grade, and high yield below investment grade.
- **Elements of Fama-French 5-factor model** [20]: NYSE, AMEX, and NASDAQ stock exchanges minus federal funds rate, average return on the small capitalization stock portfolios minus the average return on the large stock portfolios, average return on the value portfolios minus the average return on the growth portfolios, average return on the robust operating profitability portfolios minus the aver-age return on the weak operating profitability portfolios, and average return on the conservative investment portfolios minus the average return on the aggressive investment portfolios.
- **Primitive trend following strategies** on bonds, commodities, interest rates and stocks.

The data was collected monthly over a period of 25 years, from January 1999 to December 2023. For each HF, the final dataset consists of 15 features and 300 instances in total.

C. Data preprocessing and experimental setup

As part of the data preprocessing, the five HFs returns series were divided into training set which include the first 22.5 years, and test set, i.e. the last 2.5 years. Since the data are on different scales, we applied min-max normalization as the preprocessing step. The data also shows peaks and drawdowns, such as “dotcom” stock market bubble in the 2000s, he global economic crisis from 2007 to 2009, and Covid-19 in 2019 [12]. There is also evidence of strong correlations between factors within interest rate group and credit rating categories, as well as performance of conservative against aggressive portfolios and performance of value against growth portfolios.

To address the research questions, we aim to identify a ML prediction algorithm that performs best on transformed data. The forecasting algorithms are implemented in Python, using default hyperparameter values to isolate the influence of DR techniques. The selected algorithms range from simple linear models, such as linear SVM (epsilon value 0.1), to more complex ones, NN with two hidden layers with 64 and 32 neurons and a RELU activation function, as default hyperparameters.

TABLE II.
PERFORMANCE OF PREDICTION MODELS EVALUATED USING RRMSE

		CTA		**ELS**		**EMN**		**FIA**		**GM**	
		STD	**RRMSE**	**STD**	**RRMSE**	**STD**	**RRMSE**	**STD**	**RRMSE**	**STD**	**RRMSE**
Non-reduced	SVM - linear kernel	0.0510	0.0828	0.2661	0.4504	0.0415	0.0675	0.2346	0.4513	0.0698	0.1513
	SVM - RBF kernel	0.0510	0.0828	0.1950	0.3397	0.0415	0.0675	0.0229	0.3778	0.0698	0.1513
	NN	0.0847	0.1086	0.2051	0.3261	0.0615	0.1078	0.1825	0.2813	0.1073	0.1812
	XGB	0.0744	0.1035	0.1349	0.2742	0.0351	0.0684	0.4164	0.6141	0.0654	0.1290
PCA	SVM - linear kernel	0.0510	0.0828	0.1887	0.3789	0.0415	0.0675	0.2720	0.4455	0.0698	0.1513
	SVM - RBF kernel	0.0510	0.0828	0.1176	**0.1849**	0.0415	0.0675	0.2006	0.4124	0.0698	0.1513
	NN	0.0554	0.0840	0.1933	0.3248	0.0574	0.0891	0.1637	0.3025	0.0815	0.1394
	XGB	0.0546	0.0892	0.1725	0.3361	0.0491	0.0794	0.1390	0.2489	0.0904	0.1524
KPCA	SVM - linear kernel	0.0510	0.0828	0.1856	0.3637	0.0415	0.0675	0.3028	0.4773	0.0698	0.1513
	SVM - RBF kernel	0.0510	0.0828	0.1202	0.2004	0.0415	0.0675	0.1775	0.3687	0.0698	0.1513
	NN	0.0514	0.0825	0.1730	0.2749	0.0493	0.0740	0.1504	0.2390	0.0675	0.1351
	XGB	0.0618	0.0969	0.1651	0.3031	0.0438	0.0731	0.1504	0.2609	0.0779	0.1482
t-SNE	SVM - linear kernel	0.0510	0.0828	0.1793	0.2877	0.0415	0.0675	0.1676	0.3209	0.0698	0.1513
	SVM - RBF kernel	0.0510	0.0828	0.1917	0.3543	0.0415	0.0675	0.2020	0.3986	0.0698	0.1513
	NN	0.1326	0.2006	0.6024	0.8879	0.2847	0.4307	0.7113	1.0910	0.3966	0.5741
	XGB	0.0752	0.1086	0.1844	0.2958	0.0372	0.0683	0.1496	0.2353	0.0723	0.1297
UMAP	SVM - linear kernel	0.0510	0.0828	0.1839	0.3027	0.0415	0.0675	0.3211	0.5134	0.0698	0.1513
	SVM - RBF kernel	0.0510	0.0828	0.1723	0.2920	0.0415	0.0675	0.2396	0.4991	0.0698	0.1513
	NN	0.0958	0.1981	0.3228	0.7567	0.1823	0.3648	0.3520	0.8028	0.2826	0.5249
	XGB	0.0745	0.1072	0.2449	0.3906	0.0335	0.0699	0.1471	0.2409	0.1059	0.1680
Autoencoder	SVM - linear kernel	0.0510	0.0828	0.1770	0.2933	0.0415	0.0675	0.2245	0.3694	0.0698	0.1513
	SVM - RBF kernel	0.0510	0.0828	0.1494	0.2732	0.0415	0.0675	0.2395	0.4932	0.0698	0.1513
	NN	0.0573	**0.0822**	0.1522	0.2300	0.0299	**0.0571**	0.1246	**0.2071**	0.0645	**0.1078**
	XGB	0.0635	0.0932	0.2678	0.4082	0.0472	0.0860	0.1905	0.2904	0.0998	0.1717

We also employ nonlinear SVM (RBF kernel and epsilon value 0.1) to account for the nonlinear nature of financial data and to explore how a nonlinear feature extraction technique aligns with a relatively simple nonlinear predictor. Finally, we utilize XGB (100 decision trees with a maximum depth of 6), as a representative of ensemble learning methods particularly effective in prediction tasks.

The success of the proposed methods will be measured using standard evaluation metrics for regression: mean absolute error (MAE) and relative root mean squared error (RRMSE). RRMSE is normalized error using the average of actual values and is often expressed as a percentage.

IV. Experimental results

The effectiveness of DR methods is validated by comparing predicted values with actual hedge fund returns across all five HF strategies. The performance of ML predictive models, using RRMSE and MAE, is shown in Tables II and III.

As expected, more sophisticated techniques, such as autoencoders, deliver better performance in all cases except for the ELS strategy. This exception may stem from the Fama-French asset pricing factors already being carefully selected by domain experts. From an investment perspective, the analysis of individual principal component loadings indicates that factors such as investment-grade (BAA), high-yield below investment grade (BBB), and 5-year and 10-year constant maturity rates have a significant impact on most strategies. This is also expected, as these are key drivers of performance across various industries and sectors.

Furthermore, the empirical results are used to address main research questions. The prediction results of HFs return series confirm that dimensionality reduction, both linear and nonlinear, in most cases, positively affects the predictive capacity of ML forecasting models. However, nonlinear DR and nonlinear ML predictors together potentially generate more noise in the data e.g. combination of t-SNE/UMAP with NN increases error.

In general, linear PCA provides equal or better prediction results compared to non-reduced input datasets. It was expected that adding nonlinear NN and XGB predictors to non-reduced and/or PCA-reduced data will significantly reduce error. However, that was not the case for most strategies, even though linear DR did contribute to reducing error in general. This indicates the potential presence of nonlinear dependence in the original data which required analysis of more complex DR algorithms.

Among the tested nonlinear DR methods, t-SNE and UMAP resulted in higher prediction errors across strategies

Table III.
Performance of prediction models evaluated using MAE

		CTA		ELS		EMN		FIA		GM	
		STD	**MAE**	**STD**	**MAE**	**STD**	**MAE**	**STD**	**MAE**	**STD**	**MAE**
Non-reduced	SVM - linear kernel	0.0294	0.0375	0.0434	0.0593	0.0120	0.0154	0.0296	0.0486	0.0147	0.0283
	SVM - RBF kernel	0.0294	0.0375	0.0318	0.0454	0.0120	0.0154	0.0289	0.0378	0.0147	0.0283
	NN	0.0488	0.0391	0.0335	0.0414	0.0177	0.0256	0.0230	0.0270	0.0227	0.0308
	XGB	0.0428	0.0414	0.0220	0.0390	0.0101	0.0170	0.0525	0.0569	0.0138	0.0235
PCA	SVM - linear kernel	0.0294	0.0375	0.0308	0.0537	0.0120	0.0154	0.0343	0.0445	0.0147	0.0283
	SVM - RBF kernel	0.0294	0.0375	0.0192	**0.0233**	0.0120	0.0154	0.0253	0.0454	0.0147	0.0283
	NN	0.0319	0.0363	0.0316	0.0426	0.0166	0.0197	0.0206	0.0321	0.0172	0.0239
	XGB	0.0314	0.0406	0.0282	0.0471	0.0142	0.0180	0.0175	0.0260	0.0191	0.0259
KPCA	SVM - linear kernel	0.0294	0.0375	0.0303	0.0511	0.0120	0.0154	0.0382	0.0465	0.0147	0.0283
	SVM - RBF kernel	0.0294	0.0375	0.0196	0.0262	0.0120	0.0154	0.0224	0.0407	0.0147	0.0283
	NN	0.0296	0.0371	0.0282	0.0349	0.0142	0.0159	0.0190	0.0234	0.0143	0.0247
	XGB	0.0355	0.0429	0.0270	0.0415	0.0126	0.0169	0.0190	0.0269	0.0164	0.0266
t-SNE	SVM - linear kernel	0.0294	0.0375	0.0293	0.0367	0.0120	0.0154	0.0211	0.0345	0.0147	0.0283
	SVM - RBF kernel	0.0294	0.0375	0.0313	0.0487	0.0120	0.0154	0.0255	0.0433	0.0147	0.0283
	NN	0.0763	0.0866	0.0984	0.1065	0.0822	0.0933	0.0897	0.1043	0.0837	0.0876
	XGB	0.0432	0.0451	0.0301	0.0378	0.0107	0.0165	0.0189	0.0229	0.0153	0.0227
UMAP	SVM - linear kernel	0.0294	0.0375	0.0300	0.0393	0.0120	0.0154	0.0405	0.0505	0.0147	0.0283
	SVM - RBF kernel	0.0294	0.0375	0.0281	0.0385	0.0120	0.0154	0.0302	0.0552	0.0147	0.0283
	NN	0.0551	0.0997	0.0527	0.1117	0.0526	0.0912	0.0444	0.0909	0.0597	0.0934
	XGB	0.0429	0.0443	0.0400	0.0497	0.0097	0.0177	0.0185	0.0240	0.0224	0.0275
Autoencoder	SVM - linear kernel	0.0294	0.0375	0.0289	0.0382	0.0120	0.0154	0.0283	0.0370	0.0147	0.0283
	SVM - RBF kernel	0.0294	0.0375	0.0244	0.0374	0.0120	0.0154	0.0302	0.0544	0.0147	0.0283
	NN	0.0330	**0.0339**	0.0248	0.0282	0.0086	**0.0140**	0.0157	**0.0209**	0.0136	**0.0182**
	XGB	0.0366	0.0392	0.0437	0.0503	0.0136	0.0207	0.0240	0.0276	0.0211	0.0295

(e.g., GM and FIA), indicating its limitations for this forecasting task. On the other hand, autoencoders are proven to contribute to predictive power to most ML models, and for all analyzed HF strategies. For all HF strategies, the best performance is achieved with two nonlinear DR techniques, KPCA and autoencoders, combined with NN forecasting model.

When considering algorithm characteristics, KPCA is computationally intensive and sensitive to high dimensionality, while autoencoders are better suited for processing larger datasets. From the perspective of HF managers, improved forecasting helps reduce performance uncertainty to some extent, supporting more informed decision-making when selecting among diverse hedge fund strategies. However, both approaches lack explainability, making it challenging for managers to justify their decisions.

V. Conclusion

This paper extends the research initiated in [12] on identifying and extracting key features in financial time series forecasting. In this study, the influence of various nonlinear DR methods was analyzed for HF performance forecasting. An evaluation of effectiveness of KPCA, t-SNE, UMAP and autoencoders was examined on real-world data. Reduced data serves as inputs to SVM with linear and nonlinear kernel, NN and XGB as predominant ML predictors for financial time series. To provide credible conclusions, we utilized HF returns data that represent different investment styles like CTA, ELS, EMN, FIA, and GM [12]. The four group of factors (15 features in total) covering interest rate, credit rating, trend and Fama-French model were included into analysis, over 25 years period.

The empirical findings address the primary research questions. The analysis shows that both linear and nonlinear DR contribute to ML forecasting results for diverse HF strategies. Linear PCA was not effective enough, even when combined with nonlinear ML predictors. The best performance was achieved with nonlinear DR, KPCA and autoencoders when paired with neural networks. Other critical DR characteristics were discussed for the most effective combinations. From the point of view of HF managers, the results provide better forecasting accuracy and contribute to more informed investment decision making. Besides the necessity for domain ML knowledge (e.g. hyper-parameters), the lack of explainability was identified as the main drawback, as expected.

Our future work will be oriented towards the development and application of investment-based DR methods that offer explainability as a baseline for effective decision-making (e.g. [12]). Also, investment metrics will be included to validate investment performance.

Acknowledgment

This study was supported by the University of Belgrade Faculty of Organizational Sciences.

References

[1] HFR Industry Reports. [Online]. Available: www.hfr.com/hfr-industry-reports/

[2] W. Wu, J. Chen, Z. Yang, and M. L. Tindall, “A cross-sectional machine learning approach for hedge fund return prediction and selection,” *Manag. Sci.*, vol. 67, no. 7, pp. 4577–4601, 2021. https://doi.org/10.1287/mnsc.2020.3696

[3] M. Ashraf *et al*., “A survey on dimensionality reduction techniques for time-series data,” *IEEE Access*, vol. 11, pp. 42909–42923, 2023, doi: 10.1109/ACCESS.2023.3269693.

[4] R. Zaheer, M. K. Hanif, M. U. Sarwar, and R. Talib, “Evaluating the effectiveness of dimensionality reduction on machine learning algorithms in time series forecasting,” *IEEE Access*, 2025. https://doi.org/10.1109/ACCESS.2025.3551741

[5] F. Anowar, S. Sadaoui, and B. Selim, “Conceptual and empirical comparison of dimensionality reduction algorithms (PCA, KPCA, LDA, MDS, SVD, LLE, ISOMAP, LE, ICA, t-SNE),” *Comput. Sci. Rev.*, vol. 40, 100378, 2021. https://doi.org/10.1016/j.cosrev.2021.100378

[6] IEEE Xplore. [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=10929010

[7] A. M. Lopes and J. A. T. Machado, “Dynamical analysis of the Dow Jones index using dimensionality reduction and visualization,” *Entropy*, vol. 23, no. 5, p. 600, 2021. https://doi.org/10.3390/e23050600

[8] Z. Wang, H. Chen, X. Yang, J. Wan, T. Li, and C. Luo, “Fuzzy rough dimensionality reduction: a feature set partition-based approach,” *Information Sciences*, vol. 644, Art. no. 119266, 2023.

[9] D. Kumar, P. K. Sarangi, and R. Verma, “A systematic review of stock market prediction using machine learning and statistical techniques,” *Mater. Today Proc.*, vol. 49, pp. 3187–3191, 2022. https://doi.org/10.1016/j.matpr.2020.11.399

[10] S. Gu, B. Kelly, and D. Xiu, “Empirical asset pricing via machine learning,” *Rev. Financ. Stud.*, vol. 33, no. 5, pp. 2223–2273, 2020. https://doi.org/10.1093/rfs/hhaa009

[11] C. Lin, “Key financial indicators analysis and stock trend forecasting based on a wrapper feature selection method,” in *2024 19th Conf. on Computer Science and Intelligence Systems (FedCSIS)*, Belgrade, Serbia, Sep. 2024, pp. 755–759.

[12] A. Radosavcevic, A. Poledica, and I. Antovic, “Discovery of key factors in hedge funds investment strategies using optimal IBA-based logical polynomials,” in *Proc. Int. Conf. Inf. Process. Manage. Uncertainty Knowl.-Based Syst.*, Cham, Switzerland: Springer, 2024, pp. 335–346. https://doi.org/10.1007/978-3-031-74000-8_28

[13] B. M. S. Hasan and A. M. Abdulazeez, “A review of principal component analysis algorithm for dimensionality reduction,” *J. Soft Comput. Data Mining*, vol. 2, no. 1, pp. 20–30, 2021. https://doi.org/10.30880/jscdm.2021.02.01.003

[14] L. J. Cao *et al*., “A comparison of PCA, KPCA and ICA for dimensionality reduction in support vector machine,” *Neurocomputing*, vol. 55, no. 1–2, pp. 321–336, 2003. https://doi.org/10.1016/S0925-2312(03)00433-8

[15] R. Silva and P. Melo-Pinto, “t-SNE: A study on reducing the dimensionality of hyperspectral data for the regression problem of estimating oenological parameters,” *Artif. Intell. Agric.*, vol. 7, pp. 58–68, 2023. https://doi.org/10.1016/j.aiia.2023.02.003

[16] M. W. Dorrity, L. M. Saunders, C. Queitsch, S. Fields, and C. Trapnell, “Dimensionality reduction by UMAP to visualize physical and genetic interactions,” *Nat. Commun.*, vol. 11, no. 1, p. 1537, 2020. https://doi.org/10.1038/s41467-020-15351-4

[17] Y. Wang, H. Huang, C. Rudin, and Y. Shaposhnik, “Understanding how dimension reduction tools work: An empirical approach to deciphering t-SNE, UMAP, TriMAP, and PaCMAP for data visualization,” *J. Mach. Learn. Res.*, vol. 22, no. 201, pp. 1–73, 2021.

[18] B. Ghojogh *et al*., “Feature selection and dimensionality reduction in pattern analysis: A literature review,” *arXiv preprint arXiv:1905.02845*, 2019. https://doi.org/10.48550/arXiv.1905.02845

[19] Q. Fournier and D. Aloise, “Empirical comparison between autoencoders and traditional dimensionality reduction methods,” in *Proc. 2nd Int. Conf. Artif. Intell. Knowl. Eng. (AIKE)*, pp. 211–214, IEEE, June 2019. https://doi.org/10.1109/AIKE.2019.00044

[20] E. F. Fama and K. R. French, “Common risk factors in the returns on stocks and bonds,” *J. Financ. Econ.*, vol. 33, no. 1, pp. 3–56, 1993. https://doi.org/10.1016/0304-405X(93)90023-5