Improving Real Estate Investment Trusts (REITs) time-series prediction accuracy using machine learning and technical analysis indicators

Fatim Z. Habbab[1] · Michael Kampouridis[1] · Tasos Papastylianou[2]

Abstract

The primary goal of investors who include Real Estate Investment Trusts (REITs) in their portfolios is to achieve better returns while reducing the overall risk of their investments. REITs are entities responsible for owning and managing real estate properties. To achieve greater returns while reducing risk, it is essential to accurately predict future REIT prices. This study explores the predictive capability of five different machine learning algorithms used to predict REIT prices. These algorithms include Ordinary Least Squares Linear Regression, Support Vector Regression, k-Nearest Neighbours Regression, Extreme Gradient Boosting, and Long/Short-Term Memory Neural Networks. Additionally, historical REIT prices are supplemented with Technical Analysis indicators (TAIs) to aid in price predictions. While TA indicators are commonly used in stock market forecasting, their application in the context of REITs has remained relatively unexplored. The study applied these algorithms to predict future prices for 30 REITs from the United States, United Kingdom, and Australia, along with 30 stocks and 30 bonds. After obtaining our price predictions, we employ a Genetic Algorithm (GA) to optimise weights of a diversified portfolio. Our results reveal several key findings: (i) all machine learning algorithms demonstrated low average and standard deviation values in the error rate distributions, outperforming commonly used statistical benchmarks such as Holt's Linear Trend Method (HLTM), Trigonometric Box-Cox Autoregressive Time Series (TBATS), and Autoregressive Integrated Moving Average (ARIMA); (ii) incorporating Technical Analysis indicators in the ML algorithms resulted in a significant reduction in prediction errors, up to 60% in some cases; and (iii) a multi-asset portfolio constructed using predictions that incorporated Technical Analysis indicators outperformed a portfolio based solely on predictions derived from past prices. Furthermore, this study employed Shapley Value-based techniques, specifically SHAP and SAGE, to analyse the importance of the features used in the analysis. These techniques provided additional evidence of the value added by Technical Analysis indicators in this context.

Keywords Machine learning · REITs · Financial time-series · Technical analysis

1 Introduction

The optimisation of a portfolio that includes real estate is an important area of research in finance (Thakkar and Chaudhari 2021). Real estate has gained significant interest from investors globally due to its potential to enhance returns and reduce risks in mixed-asset portfolios (Habbab et al. 2022). Empirical evidence suggests that including real estate assets in a portfolio can improve risk-adjusted returns. For example, research studies have demonstrated that real estate investments tend to have low correlation with other asset classes such as stocks and bonds, providing diversification benefits (Gatzlaff and Geltner 1991). Furthermore, real estate investments have been shown to be an effective hedge against inflation, as they tend to maintain their value or even appreciate during inflationary periods (Miles 2004).

Previous studies have recommended allocating a portion of the portfolio to real estate between 10% and 15%, with longer holding periods potentially increasing this allocation (Delfim and Hoesli 2019, 2020). However, in order to produce the requisite models for estimating asset returns and volatility in real estate investments, such studies have traditionally depended on appropriate historical data being available; this creates a challenge, as the nature of real estate markets, which are less liquid and exhibit much less frequent transactions when compared to other asset types, makes it challenging to obtain accurate and up-to-date data for performing such optimisation tasks. Additionally, real estate valuations can be subjective and influenced by factors such as market sentiment and local economic conditions, further complicating the optimisation process (Habbab and Kampouridis 2022).

To address this limitation, a two-step approach has recently been proposed for optimising mixed-asset portfolios that include real estate (Habbab and Kampouridis 2022). First, a model is constructed to predict future security prices using historical data. Then, the portfolio is optimised using these predicted prices. This methodology has been successfully applied to stock portfolios (Ma et al. 2021; Chen et al. 2021; Ma et al. 2020) and recently extended to real estate portfolios (Habbab et al. 2022; Habbab and Kampouridis 2024), which showed that incorporating price predictions into the optimisation process can lead to significant improvements in portfolio performance.

REITs, or Real Estate Investment Trusts, can be viewed as a distinct form of real estate investment that offers the opportunity of investing in real estate without the need to possess the actual properties in question. Accurate prediction of Real Estate Investment Trust (REIT) prices is crucial for optimising a mixed-asset portfolio that includes real estate (Habbab and Kampouridis 2022, 2023). In addition to utilising past REIT prices, incorporating Technical Analysis (TA) indicators (TAIs) as features in the prediction process can further improve prediction accuracy. While such indicators have been very popular in tasks like algorithmic trading (Brabazon et al. 2020), their application to predicting REIT prices is limited. An example of a study that used TA for REITs is Habbab et al. (2023), which used TAIs as part of the feature set of five different machine learning algorithms, and demonstrated that the introduction of these indicators led to a reduction in regression error of up to 50%.

Previous works adopted TAIs to predict financial instrument data, including (Agrawal et al. 2019; Oriani and Coelho 2016). However, they focused on stock market data. In this work, we aim to incorporate TA to predict REIT prices as well, to explore the potential improvement that can be made in the case of real estate investments. Moreover, previous studies that optimised investment portfolios including real estate as well as other asset

classes (Jones and Trevillion 2022; Geiger et al. 2016) relied on historical data. In contrast, we use predictions of future prices to optimise a multi-asset portfolio. Another limitation of the current literature is that most studies tend to analyse the predictive performance of at most one or two (mainly deep learning) algorithms by averaging the considered metric (such as RMSE) over a single testing period in the context of REITs data (Li et al. 2022; Axelsson and Song 2023). By contrast, this study examines the predictive performance of five algorithms over different testing periods (i.e., 30-, 60-, 90-, 120-, and 150-day), adopts two prediction methods (i.e., out-of-sample one-day-ahead and N-day ahead) and considers the volatility of results in addition to the mean.

The primary novelty of this paper lies in its comprehensive and empirical approach to predicting REITs time-series data. It involves the inclusion of Technical Analysis Indicators (TAIs) in the set of features used for prediction. This approach builds upon Habbab et al. (2023). Given how underused TAIs have been in this domain, it is important to demonstrate the advantages that TA can bring in price predictions, and subsequently, to portfolios that use REITs as one of their asset classes. The algorithms used to achieve this purpose are Ordinary Least Squares Linear Regression (LR), Support Vector Regression, K-Nearest Neighbours, eXtreme Gradient Boosting (XGBoost), and Long Short Term Memory (LSTM) Neural Networks, and extend our previous work in six key ways: (i) we incorporate a larger number of datasets (i.e., assets), expanding from 27 to 90; (ii) we add two more benchmarks, namely Holt's Linear Trend Method (HLTM) and Trigonometric Box-Cox Autoregressive Time Series (TBATS) bringing the total number to three; (iii) we consider five (instead of one) prediction periods, namely, 30-, 60-, 90-, 120-, and 150-days; (iv) we analyse two prediction methods, i.e., out-of-sample period-ahead prediction, and one-day-ahead prediction; (v) we use the price predictions as input into a portfolio that contains three asset classes, namely stocks, bonds, and REITs, and discuss in detail the positive effects that the use of the TAIs brings in the context of portfolio optimisation when using a Genetic Algorithm, and subsequently compare the portfolio's results against four benchmarks; and (vi) we conduct an in-depth analysis using Shapley Value-based metrics, namely SHAP (Lundberg and Lee 2017), and SAGE (Covert et al. 2020), providing further insight into the nature of the contribution of TAs with respect to individual predictions, and model quality more generally.

The remainder of this paper is structured as follows: Sect. 2 offers a concise overview of REITs, the Modern Portfolio Theory (MPT), and discusses related research in this area; Sect. 3 outlines the methodology used in this study; Sect. 4 presents the details of our experimental setup; Sect. 5 provides a comprehensive analysis of the experimental results achieved by applying ML techniques and the proposed benchmarks to our dataset; lastly, Sect. 6 provides a summary of the key findings and brings the paper to a conclusion.

2 Background

This section provides background information on the key topics of this study. We will first introduce real estate investments, and then present the Modern Portfolio Theory (MPT), on which our portfolio optimisation strategy is based. Lastly, we will discuss Technical Analysis Indicators (TAIs), which will form part of the machine learning algorithms' feature set.

2.1 Real estate investments

Real estate investments refer to the acquisition, ownership, management, and sale of real property with the primary goal of generating income and/or capital appreciation (Liow 2016). Real property includes land, buildings, and any improvements made to them. Real estate investments can take various forms, such as residential properties, commercial buildings, industrial facilities, and undeveloped land.

Investing in real estate markets confers great advantages. First, when managing portfolio risk, real estate investments are an excellent choice of asset to include in a diversification strategy making use of asset allocation. Historically, real estate has demonstrated relative stability and lower volatility compared to other investment classes, such as stocks or bonds (Liow 2016). The tangible nature of real estate assets and the underlying demand for housing and commercial spaces contribute to this stability.

Furthermore, one of the main attractions of real estate investments is the potential for property value appreciation over time. Studies have shown that, on average, real estate tends to appreciate in value over the long term (Gatzlaff and Sirmans 1991). Factors such as location, economic growth, supply and demand dynamics, and property improvements can influence the rate of appreciation.

A further benefit of investing in real estate is that it can generate regular cash flow through rental income. Rental properties, such as residential apartments or commercial spaces, can provide a steady stream of income that can be used for expenses, debt service, or reinvestment (Geltner et al. 2016). The positive cash flows that can result from real estate investments in this manner can contribute greatly to achieving financial goals and the building of long-term wealth.

An investor can gain exposure to real estate markets in two main ways: direct and indirect investments. Direct real estate investments involve acquiring and owning physical properties directly. In this case, investors directly own the properties and have control over their management, operations, and decision-making. Direct investments offer the potential for higher returns, direct cash flow from rental income, tax benefits, and more control over the investment. However, they also involve more hands-on management, higher transaction costs, and risk exposure concentrated on individual properties (Geltner et al. 2016).

Indirect real estate investments involve investing in real estate through financial instruments or intermediaries, such as Real Estate Investment Trusts (REITs), real estate mutual funds, real estate exchange-traded funds (ETFs), or real estate limited partnerships. Investors own shares or units of these investment vehicles rather than owning physical properties directly. Indirect investments are managed by professional fund managers who oversee the portfolio, property acquisition, and management activities. In this way, indirect investments provide access to real estate markets and opportunities that might otherwise have been difficult for individual investors to access directly (Liow 2016).

The remainder of this section is structured as follows. Section 2.1.1 presents an overview of the current sectors existing in the real estate market, while Sect. 2.1.2 describes a specific kind of publicly traded real estate investment entities known as Real Estate Investment Trusts (REITs), since they represent the primary focus of this article.

2.1.1 Real estate markets

Real estate markets are a vital component of the global economy, and they include various sectors that serve diverse purposes. The main real estate sectors include the following: (a) residential real estate; (b) commercial real estate; (c) industrial real estate; and (d) raw land.

The residential real estate sector involves properties used for residential purposes, such as single-family homes, condominiums, apartments, and townhouses. Residential real estate is usually characterised by factors such as location, size, and style, which reflect the needs of homeowners, renters, and real estate investors. Factors like population growth, employment rates, and mortgage interest rates significantly impact the demand for residential real estate.

Commercial real estate includes properties used for business activities, such as offices, retail spaces, hotels, and warehouses. It is divided into sub-sectors like office, retail, industrial, and hospitality, each with its own unique characteristics and challenges. The performance of the commercial real estate sector is closely tied to economic indicators, including consumer spending, corporate expansion, and business sentiment.

Industrial real estate is dedicated to facilities used for manufacturing, warehousing, and distribution of goods. It includes factories, distribution centres, and logistics hubs. The growth of e-commerce and changes in supply chain dynamics have significantly influenced the industrial real estate sector, leading to increased demand for modern distribution centres and last-mile delivery facilities.

Finally, raw land is undeveloped, vacant land that has not been improved or built upon. It represents a potential opportunity for future development. Investors, developers, and speculators often buy raw land with the intention of holding it until market conditions are favourable for development, or they may develop it themselves. The value of raw land can vary significantly based on its location, zoning regulations, and its potential for development.

2.1.2 Real Estate Investment Trusts

Real Estate Investment Trusts (REITs) are entities that manage, fund, or possess income-producing real estate assets. Well-known REITs include Realty Income Corporation (O), Digital Realty Trust, Inc (DLR), and Simon Property Group, Inc (SPG), among others. Through investing in REITs, regular investors can participate in real estate investments and exploit the benefits of competitive returns and dividend-based income without the large capital expenditure that direct real estate investment requires (Block 2011).

Investing in REITs is similar to investing in other financial markets, and there are various ways investors can do so. Some options include purchasing individual company stocks, mutual funds, or exchange-traded funds (ETFs). To identify suitable REIT investments, investors may consult with a broker, financial adviser, or planner to establish their financial objectives. A 2020 study conducted in the US by Chatham Partners[1] showed that approximately 80% of financial advisers recommend REITs to their clients. Additionally, investors can consider investing in private REITs or public non-listed REITs.

The ownership of some properties is transferred to investors who hold shares in REITs, allowing them to earn a share of the income generated without needing to purchase, manage, or finance the property. Optimal portfolio allocation for REITs has been studied extensively, with studies such as those conducted by Hocht et al. (2008), Bhuyan et al. (2014), and Jalil

---

[1] https://www.reit.com/investing/why-invest-reits.

et al. (2015) suggesting that REIT investment should typically make up between 5% and 15% of an investment portfolio. This weighting may vary depending on the investment horizon, with research by Stephen and Simon (2005) and Rehring (2012) highlighting that the diversification potential of REITs increases over longer holding periods.

REITs typically invest in a variety of real estate properties, including but not limited to, offices, apartments, warehouses, retail centres, medical facilities, data centres, cell towers, infrastructures, and hotels. While some REITs focus on a particular type of property, others may have portfolios that comprise multiple property types.

REITs primarily generate income by leasing properties and receiving rent payments, which are then distributed to shareholders in the form of dividends. In the US, REITs are required to pay at least 90% of their taxable income to shareholders, who are then responsible for paying taxes on those dividends.

Investors find REITs an appealing investment choice because of their competitive returns, which come from a mix of steady income and long-term capital appreciation, as well as their low correlation with other asset classes. This attribute provides an opportunity for portfolio diversification, making portfolios that include REITs less risky than those without, as illustrated in Sect. 5.

There are several REIT types, including Equity REITs (e-REITs), Mortgage REITs (m-REITs), Public Non-Listed REITs, and Private REITs. The most common type of REITs on the market are Equity REITs, which own or operate income-producing real estate. Mortgage REITs (mREITs) finance income-producing real estate by purchasing or creating mortgages and mortgage-backed securities, earning interest-based income from these investments. For example, in the US, public non-listed REITs are registered with the U.S. Securities and Exchange Commission (SEC) but do not trade on national exchanges, whereas private REITs are not traded on national exchanges and are exempt from SEC registration.

Like other financial markets, REIT share prices fluctuate throughout the trading day. The value of REIT shares is influenced by various factors such as expected earnings growth, expected total returns, dividend yields compared to other yield-oriented investments like bonds or utility stocks, dividend payout ratios, management quality, corporate structure, and the underlying asset values of the real estate and mortgages. REIT market values are represented by different indices, including the FTSE EPRA/Nareit US Real Estate Index, which contains specific REIT companies operating in the US. This study focuses on publicly listed equity REITs (e-REITs) like American Tower Corporation (AMT), Prologis (PLD), Crown Castle (CCI), Public Storage (PSA), and Welltower (WELL) that hold various types of real estate properties such as infrastructure, offices, shopping malls, and others.

2.2 Modern portfolio theory

Modern portfolio theory (MPT) is a framework for constructing and managing investment portfolios, based on the idea that investors can minimise risk for a given level of expected return through 'asset allocation' i.e., the act of diversifying their investments across a range of asset classes (Markowitz 1952). The theory suggests that an investor can minimise risk by spreading their investments across different asset classes, such as stocks, bonds, and real estate, rather than investing in a single asset class. MPT uses mean-variance analysis to measure risk and return (Elton et al. 2009).

MPT assumes that investors are rational and risk-averse, meaning that they prefer less risk for a given level of return (Sharpe 1964). The theory has been widely used in the investment industry for portfolio construction and management, but it has also been criticised for its assumptions, such as the assumption that returns follow a normal distribution and that investors are rational and risk-averse (Fama and French 1992).

Despite its criticisms, MPT has had a significant impact on the investment industry and remains a widely used framework for portfolio management (Michaud 1989).

In MPT, the two key factors used for making investment decisions are the expected portfolio return and the expected portfolio risk. The expected return of an asset is the average return that an investor expects to receive from that asset over a specific period of time. The expected return of a portfolio is calculated by weighing the potential return of each asset in the portfolio by the percentage of the portfolio invested in each asset (Markowitz 1952).

The formula for calculating the expected return of a portfolio can be represented as:

$$E[R_p] = \sum_{i=1}^{n} w_i E[R_i] \tag{1}$$

where: $E[R_p]$ is the expected return of the portfolio; w_i = the weight of the ith asset in the portfolio; and $E[R_i]$ = the expected return of the ith asset (out of a total of n assets).

Risk in MPT is typically measured using standard deviation, which is a statistical measure that indicates the degree of variation of returns around the expected return (Markowitz 1952). The expected risk of a portfolio is calculated as follows:

$$\sigma_p = \sqrt{\sum_{i=1}^{n} \sum_{j=1}^{n} w_i w_j \sigma_i \sigma_j \rho_{i,j}} \tag{2}$$

where: σ_p is the expected risk (standard deviation) of the entire portfolio; w_i is the weight of the ith asset in the portfolio; σ_i is the standard deviation of the ith asset and $\rho_{i,j}$ is the Pearson correlation coefficient between two assets in the portfolio.

In MPT, correlations between assets are important in determining the optimal portfolio for an investor. Correlation is a measure of the strength and direction of the linear relationship between two variables, and in MPT, it is used to measure the degree to which the returns of two assets move together (Markowitz 1952).

The formula for calculating the correlation between two assets can be represented as:

$$\rho_{i,j} = \frac{cov(R_i, R_j)}{\sigma_i \sigma_j} \tag{3}$$

where: $\rho_{i,j}$ is the correlation between assets i and j; $cov(R_i, R_j)$ is the covariance between the returns of assets i and j; σ_i is the standard deviation of the returns of asset i; and σ_j is the standard deviation of the returns of asset j.

A high correlation between two assets indicates that their returns tend to move in the same direction, while a low correlation indicates that their returns tend to move independently of each other.

In MPT, diversification is used to reduce risk by investing in assets that are not perfectly correlated with each other. By investing in a diverse portfolio of assets with low correlations, investors can reduce the overall risk of their portfolio.

In conclusion, MPT assumes that an optimal portfolio can be built using information gleaned from the expected return, risk, and correlations between assets. An investor is primarily concerned about maximising the expected return for a given level of risk, or minimising the expected risk for a given level of return. The overall level of correlation between assets included in a portfolio determines the level of diversification, and thus the level of risk of an investment portfolio.

2.3 Technical analysis

Technical analysis (TA) is a method used in the financial markets to evaluate and forecast the future price movements of various assets, such as stocks, currencies, and commodities. Traders and investors rely on this methodology to gain insights for informed decision-making regarding the buying, selling, or holding of various financial assets, including stocks, currencies, and commodities (Murphy 1999).

(TA) is a commonly employed method that entails the analysis of past price and volume information in financial markets to forecast future price fluctuations.

One of the key principles of TA is the belief that market prices follow trends and patterns, and that these trends can be identified and utilised for predictive purposes. Technical analysts utilise a wide range of tools and techniques to analyse market data, including chart patterns, statistical models, and technical analysis indicators—the latter of which constitutes the focus for this work.

Chart patterns are visual representations of historical price movements that can provide insights into future price direction. Examples of commonly used chart patterns include the *Head and Shoulder*, *Double Top*, and *Triangle* patterns (Bulkowski 2012). These patterns are often believed to indicate potential reversals or continuations in price trends.

Statistical models can be used to forecast future price movements. These models often involve the use of regression analysis, time-series analysis, and other statistical techniques to identify relationships and trends in the data.

Technical analysis indicators (TAIs) are mathematical calculations based on historical price and volume data. They are used to generate trading signals and identify potential buying or selling opportunities. Some of the most popular TAIs, including Moving Averages, Moving Average Convergence Divergence (MACD), Bollinger Bands (BB), and Momentum (Zhu and Zhou 2009; Aguirre et al. 2020; Lento et al. 2007; Rosillo et al. 2013), are used in this study.

While technical analysis is widely employed in financial markets, it is not without its critics. Some argue that it is based on subjective interpretations and lacks a solid theoretical foundation (Levy 1966). Others contend that it is a self-fulfilling prophecy, as the actions of market participants following TA patterns can create the predicted price movements. Nevertheless, technical analysis continues to be popular among traders and investors, and numerous studies have explored its effectiveness; e.g. Christodoulaki et al. (2023) used TAIs in combination with sentiment analysis to produce effective trading strategies; Long et al. (2023) used technical analysis alongside indicators derived from an event-based system, in the context of a multi-objective optimisation approach; Santos and Torrent (2022)

incorporated technical analysis into a portfolio strategy where optimal weights were directly parameterised as a function of multiple trend-following signals.

In summary, technical analysis is a well-established approach in financial markets, which implies examining historical price and volume data to anticipate future price changes. It utilises chart patterns, statistical models, and technical analysis indicators to detect trends and patterns within the data. Despite criticism, research has demonstrated its potential effectiveness under specific market conditions. Notably, prior to this study, there has been no investigation into the application of Technical Analysis Indicators (TAIs) for predicting REIT prices. Thus, this research seeks to assess their potential utility in enhancing the precision of price predictions in this particular domain.

3 Methodology

The methodology of this study relies on two main stages: (i) price prediction, where we use different machine learning algorithms that include Technical Analysis Indicators (TAIs) in their feature set; and (ii) portfolio optimisation, where the predicted prices from the above step are used as input to a portfolio, whose weights are optimised by means of a Genetic Algorithm.

This section will provide a comprehensive explanation of the two steps mentioned above. Section 3.1 describes the nature of the data in general terms; Sect. 3.2 discusses the pre-processing steps that were necessary for deriving the feature set; Sect. 3.3 presents the features used in our experiments; Sect. 3.4 presents the machine learning algorithms used in our experiments; Sect. 3.5 discusses the loss function chosen; and lastly Sect. 3.6 explains the setup of the Genetic Algorithm, which was used for the portfolio optimisation task. Sections 3.2–3.5 correspond to the first step (price prediction) of our methodology, whereas Sect. 3.6 corresponds to the second step of our methodology (portfolio optimisation via a Genetic Algorithm).

3.1 Data

In this study, we consider a number of datasets[2] from financial instruments in relation to three asset classes — namely: stocks, bonds, and REITs; and three different markets—namely: United States (US), United Kingdom (UK), and Australia (AU). To avoid currency risk, all data is obtained as US dollars (USD). For more details regarding the exact number and specifics of the actual data used in our experimental setting, see Sect. 4.1 later on.

Each dataset is then further subdivided into three subsets, contiguous in time: a training set, which serves as the portion of the data that will be used to train the machine learning model; a validation set, which is used to select optimal hyperparameters for the model; and a testing set, which serves as the unseen part of data that is used for the final evaluation step, after the model has tuned and trained.

---

[2] In the context of this study, the word 'dataset' is used to refer to a single time-series of daily prices for a given asset.

Table 1 Example of time-series differencing and scaling

t	P_t	P_{t-1}	D_t	N_t	N_{t-1}	N_{t-2}
t1	3.77	–	–	–	–	–
t2	3.69	3.77	− 0.08	0.30	–	–
t3	3.7	3.69	0.01	0.70	0.30	–
t4	3.6	3.7	− 0.1	0.22	0.70	0.30
t5	3.68	3.6	0.08	1	0.22	0.70
t6	3.53	3.68	− 0.15	0	1	0.22
t7	3.54	3.53	0.01	0.70	0	1

Legend: t represents the time steps

P_t represents the security's price at time t

P_{t-1} represents the one-lag value of P_t

D_t represents the differenced value at time t

N_t represents the value of D_t following standardisation, N_{t-1} the value of D_{t-1} following standardisation, etc

3.2 Data preprocessing

Data coming from an asset's daily-price time-series cannot be plugged directly into the algorithms (see under ARIMA in Sect. 4.3.1.1 for an explanation as to why this is the case). Therefore, we perform a process of differencing and scaling on the time-series data associated with each asset. Differencing, a significant technique in time-series analysis, involves calculating the difference between successive observations within a time-series. This step proves valuable in eliminating trend and seasonality components inherent in time-series data, which can complicate modelling and analysis. First-order differencing involves subtracting the value of the previous timepoint from the current timepoint; this is represented mathematically as:

$$D_t = P_t - P_{t-1} \tag{4}$$

where P_t is the value of the time-series at time t, and D_t is the differenced time-series at time t. In cases where trend and seasonality components remain after initial differencing, it is possible to employ higher-order differencing. The selection of the differencing order is related to the unique attributes of the analysed time-series. For the scope of this paper, we exclusively focus on first-order differencing.

After obtaining D_t, the values are further standardised to the range [0, 1], by using the following scaling transformation:

$$N_t = \frac{(D_t - D_{min})}{(D_{max} - D_{min})} \tag{5}$$

where N_t is the standardised value of each variable (in this case the differenced price D_t), and D_{min} and D_{max} are the minimum and maximum values respectively, that result from the differencing of the relevant asset's time-series. We note that under this transformation, price analysis is equivalent to a holding-period-returns analysis, since the latter time-series is simply a linear transformation of the former, and thus differencing and then normalising either yields the same dataset of normalised values.

Table 2 Example of feature selection (lagged observations)

t	N_t	N_{t-1}	N_{t-2}	N_{t-3}	N_{t-4}	N_{t-5}
t2	0.30	–	–	–	–	–
t3	0.70	0.30	–	–	–	–
t4	0.22	0.70	0.30	–	–	–
t5	1	0.22	0.70	0.30	–	–
t6	0	1	0.22	0.70	0.30	–
t7	0.70	0	1	0.22	0.70	0.30

In Table 1, we present an illustration of the differencing and scaling processes using sample data for the SPG time-series spanning from January 1, 2021, to January 15, 2021.

3.3 Features

The two features that we use to tackle our regression problem are: (i) historical observations (i.e., 'lags') of the time-series variable N_t; and (ii) Technical Analysis Indicators (TAIs). Thus, the feature vector (i.e. inputs) for all ML models consists of the TAIs values obtained from the original time series, concatenated with the historical values.

3.3.1 Past observations (lags)

For the first type of features, we incorporate n past observations of N_t, i.e., N_{t-1}, N_{t-2}, N_{t-3},..., N_{t-n}, where the number of lags n is determined using the Akaike Information Criterion (AIC). The optimal value for n corresponds to the optimal parameter p in the ARIMA model, while AIC is commonly employed for model selection (Vrieze 2012; Yamaoka et al. 1978; Khairi et al. 2019). In other words, the number of lags n is obtained as a result of the hyperparameter optimisation process for the ARIMA model, which is then applied identically to all algorithms in our study; this varies for each dataset, and thus determines the overall number of features in each case. This process is further explained in Sect. 4.3.1.1. Table 2 provides an illustration of lagged observations for a selected number of lags ($n = 5$).

3.3.2 Technical analysis indicators (TAIs)

In addition to historical data points, we incorporate five Technical Analysis Indicators (TAIs) at each timepoint—Simple Moving Average (SMA), Exponential Moving Average (EMA), Moving Average Convergence/Divergence (MACD), Bollinger Bands, and Momentum—as recommended in previous studies such as Oncharoen and Vateekul (2018); Agrawal et al. (2019); Khairi et al. (2019). These indicators play a crucial role in identifying both short-term and long-term trends within a time-series, making them valuable tools for the prediction of prices.

3.3.2.1 Simple moving average The Simple Moving Average (SMA) is commonly employed to predict future data points by providing an estimate of a time-series' level

(Dinesh et al. 2021). Mathematically, the SMA can be expressed as the weighted average of the past T prices, and it is represented as:

$$\text{SMA}(t) = \frac{\sum_{i=t-(T-1)}^{t} \left[N_i \right]}{T},$$

$$(6)$$

where N_t is the normalised price at time i, and T is the number of timepoints considered. In Python, we calculate the SMA using the `rolling` method.[3] It is important to note that the period of interest T used for window-averaging is independent of the number of lags n, which determines the number of historical timepoints used for training purposes.

3.3.2.2 Exponential moving average The Exponential Moving Average (EMA) is a similar technique to the SMA, but with the key difference being that it considers all past observations, with weights that decay exponentially as a function of the distance in time between each observation and the current timepoint. More recent observations are given greater weight than older observations. The EMA is typically expressed through the following difference equation:

$$\text{EMA}(t) = \alpha N_t \; + \; (1 - \alpha) \, \text{EMA}(t - 1),$$

$$(7)$$

where α is a parameter representing the amount of weight decay applied at each timestep. α is calculated as $\alpha = 2/(T + 1)$, where T is the period of interest. It can take any real value between 0 and 1, with lower values assigning more importance to past information, and higher values indicating less importance given to past prices. In Python, we calculate the EMA using the `ewm` method.[4]

3.3.2.3 Moving average convergence/divergence The Moving Average Convergence/ Divergence (MACD) indicator is a measure of the difference between a short-term and a long-term Exponential Moving Average (EMA). It is useful for identifying *bullish* moments (i.e. periods characterised by notable market price increase relative to historically lower or more stable prices), or *bearish* moments (i.e. periods characterised by notable market price decrease compared to historically higher or more stable prices). To calculate the MACD, we select an *H-day* denoting the start of a longer, 'historical' period (lasting until the present day), and an *R-day* (closer in time to the present day compared to the H-day), denoting the start of a shorter, more 'recent' period. The 'recent' period typically represents a period of interest, whose trend one wishes to compare against the longer, 'historical' period, in order to identify a change in market trend as compared to historical levels. This is done by first

---

[3] https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.rolling.html Last accessed: June 2023.

[4] https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.ewm.html Last accessed: June 2023.

Table 3 Example of feature selection (TAIs)

t	SMA	EMA	MACD	Upper band	Lower band	Momentum
t2	–	0.15	–	–	–	–
t3	–	0.43	–	–	–	–
t4	–	0.32	0.19	0.53	0.28	–
t5	0.41	0.66	0.15	0.80	0.48	0.70
t6	0.64	0.33	0.18	0.59	0.23	0.70
t7	0.41	0.52	0	0.78	0.35	0.48

obtaining EMAs for both periods; the MACD is then obtained as the difference between the 'recent' EMA compared to the 'historical' one (Martins 2017):

$$\text{MACD}(t) = \text{EMA}_R(t) - \text{EMA}_H(t) \tag{8}$$

3.3.2.4 Bollinger bands Bollinger Bands (BB) are defined as a price range around the Simple Moving Average (SMA) price at time t, obtained as follows: first, we compute the standard deviation of all observations (i.e. with respect to the SMA), within a period of interest T, where T is typically the same period used to calculate the SMA. This is then multiplied by a modifier D, which determines the number of standard deviations away from the mean we want to set our range to. This is represented mathematically as follows:

$$\text{BB}(t) = \text{SMA}(t) \pm D\sqrt{\left(\frac{1}{T}\right)\sum_{i=t-(T-1)}^{t}\left[N_i - \text{SMA}(t)\right]^2} \tag{9}$$

Bollinger Bands are a useful tool for assessing whether the current price of a security shows a substantial deviation, indicated by a measurement of D standard deviations, from its mean. Additionally, they can assist in recognising whether the price is likely to either increase or return to its average level.

3.3.2.5 Momentum The Momentum indicator (Rosillo et al. 2013) is determined by taking the difference between the price at time t and the price from T periods ago, as illustrated below.

$$\text{Momentum} = N_t - N_{t-T} \tag{10}$$

This metric serves as a reliable indicator of the strength of a price trend, enabling the estimation of the future direction of a time-series.

Table 3 shows the TAIs computed for the preprocessed data described in Table 1. Specifically, we compute the 3-day Simple Moving Average (SMA), the Exponential Moving Average (EMA) with $\alpha = 0.5$, the Moving Average Convergence/Divergence (MACD) as the difference between the 3-day EMA and the 6-day EMA, the upper and lower Bollinger Bands using the 3-day SMA and the standard deviation of the 3-day SMA multiplied by 0.5,

and the Momentum as the difference between the current price N_t and the price N_{t-T} that was observed $T = 5$ timepoints before t.

In total, we use these six TA-based features together with the lag-based features, resulting in $n + 6$ features for our regression task.

3.4 Machine learning algorithms

Once the relevant features have been extracted from all datasets, we feed them to our 'bag' of machine learning models[5] in order to solve our price prediction problem. For each model, we obtain two variants: one that incorporates TAIs in its feature set and one that does not. This allows us to compare the performance between the two variants for each dataset and assess the importance of including TAIs in the feature set.

Our 'bag' of machine learning models comprises a diverse set of regression algorithms selected from the Machine Learning (ML) literature, including: Ordinary Least Squares Linear Regression (LR) (Kavitha et al. 2016), Support Vector Regression (SVR) (Sharifzadeh et al. 2019), eXtreme Gradient Boosting (XGBoost) (Shehadeh et al. 2021), Long/Short-Term Memory Neural Networks (LSTM) (Mussumeci and Coelho 2020), and k-Nearest Neighbours Regression (KNN) (Kohli et al. 2020). The following python libraries/functions were used for this purpose:

- `sklearn.linear_model.LinearRegression`
- `sklearn.svm.SVR`
- `xgboost.XGBRegressor`
- `keras.models.Sequential`
- `sklearn.neighbors.KNeighborsRegressor`In all cases, optimal model hyperparameters are determined through 'grid search' (see Sect. 4.2 for details). Once optimal hyperparameters are established, a model is trained one last time on the expanded set of training + validation data combined and then used to make predictions on the test set.

3.5 Evaluation metrics

All of the above algorithms use the *root mean square error* (RMSE) as the loss function, defined as follows:

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{|j|}(P_i - \hat{P}_i)^2}{|j|}}, \tag{11}$$

where P_t refers to the actual price value, $\hat{P}_t$ is its predicted value, and $|j|$ denotes the number of observations in each dataset j (i.e., in other words, we obtain *one* RMSE value per dataset). Note that the RMSE here expresses the prediction error in terms of *US dollars*, and thus needs to be calculated on the basis of the original price data (i.e., P_t), rather than

---

[5] For the purposes of this study, our operational definition of a 'machine-learning' (ML) model is any optimisation algorithm from the machine-learning or statistics literature, that can accept arbitrary features (such as TAIs) for training, as opposed to statistical methods that operate on lagged values exclusively.

the scaled data (i.e., N_t); therefore, it was necessary to reverse the scaled values to their original price values to compute the RMSE in a meaningful manner (cf. Section 3.2). We primarily express RMSE in terms of US dollars here to make the numbers involved more intuitive; however, we note that there is a linear relationship between this RMSE and one calculated with respect to scaled values, and therefore the latter evaluation would have been equivalent.

We evaluate all algorithms using two out-of-sample prediction methods-one relying on long-term prediction on the basis of fixed information and intermediate predictions, and one relying on consecutive short-term predictions on the basis of continuously updated information. Both methods are evaluated over the same range of time periods, namely 30, 60, 90, 120, and 150 days.

3.5.1 Long-term out-of-sample prediction

In this method, the known closing prices from all historical time points up until our starting point of interest, t_0 (with closing price N_0 respectively), are used to train a model, which is then used to predict the closing price for the next day (i.e., price $\hat{N}_1$, corresponding to time point t_1). Once this is obtained, the model is retrained, with $\hat{N}_1$ incorporated into the training dataset, as if it were the 'known' price at time t_1; this model is then used to predict the price for the next time point (i.e., price $\hat{N}_2$ corresponding to time point t_2). $\hat{N}_2$ is then used to predict $\hat{N}_3$ in the same manner, and so forth, until the final time point in the evaluation period of interest is reached. We will refer to this evaluation method simply as *out-of-sample* prediction henceforth in the text.

3.5.2 Consecutive one-day-ahead predictions

In this case, when it comes to predicting the closing price $\hat{N}_1$ corresponding to time point t_1, we use the known closing prices from all historical time points up until t_0, just as we did before. However, when it then comes to predicting the *next* item (i.e., the closing price $\hat{N}_2$ corresponding to time point t_2), instead of incorporating the *predicted* price, $\hat{N}_1$ to our training set at position t_1, we simply incorporate the true, *known* closing price, N_1, to the training dataset at that position instead; the updated model is then used to predict the price for the next time point (i.e., $\hat{N}_2$ for position t_2), much like before. The known N_2 is then used to predict $\hat{N}_3$ in the same manner, and so forth, until the final time point in the evaluation period of interest is reached. We will refer to this evaluation method simply as *one-day-ahead* prediction henceforth in the text.

Our expectation is to achieve higher prediction accuracy by adopting the second technique. However, it is a suitable technique to evaluate the performance, which is meaningful in the context of portfolios that follow a short-term trading strategy that needs to be adjusted according to the current market conditions. Conversely, the first approach is more suitable as an evaluation strategy in the context of investors with long investment horizons, who might thus only rebalance their portfolios periodically or adjust their investment strategies based on evolving market conditions.

3.6 Portfolio optimisation using a genetic algorithm

Once we have solved our price prediction problem, the subsequent stage involves incorporating the obtained prices into a portfolio. As previously indicated, our portfolio includes three different asset classes, namely REITs, stocks, and bonds. Our primary goal is to demonstrate that, in the case of portfolio optimisation, the use of predictive models that consider both historical prices *and* TAIs as part of their feature set yields portfolios with better performance, compared to when using models that rely on past prices only.

For the optimisation of each asset's weight within the portfolio, we employ a Genetic Algorithm (GA). GAs, a subset of Evolutionary Algorithms, draw inspiration from the field of Genetics and the principle of *natural selection*. They are designed to emulate the way living organisms evolve and adapt over time, mirroring the processes that enable genes to evolve across generations, resulting in the development of increasingly 'fit' organisms. When provided with appropriate definitions that extend the concepts of 'genes' and 'fitness' to a specific problem domain, GAs prove effective in tackling challenging optimisation tasks (Goldberg 1989). One of the prominent advantages of Genetic Algorithms is their capacity to handle complex, high-dimensional optimisation problems that may pose challenges for traditional optimisation techniques (Whitley 1994). They have been successfully applied in diverse domains, including algorithmic trading (Adegboye et al. 2023), engineering design (Deb 2011), financial portfolio optimisation (Li et al. 2015), and image recognition (Liu et al. 2002). In the following discussion, we explore the standard components of a Genetic Algorithm and how they were adapted and applied to our particular problem domain.

3.6.1 Representation

The first step in our GA implementation is to create an initial set of solutions tailored to the specific problem, which is known as 'initialisation'. During this stage, we create an initial population, and each individual in this population is represented as a 'chromosome'. In the case of portfolio optimisation, each chromosome is composed of N 'genes' which correspond to N weights given to the assets that form our portfolio. Each of these weights is represented as a real number within the range of 0 to 1 and must collectively sum to 1, reflecting the entire 100% of the total capital that an investor intends to invest in the complete portfolio. For instance, let's consider a chromosome with $N = 4$ and genes [0.5, 0.2, 0.1, 0.2]. This configuration implies a portfolio composed of four assets, with the first asset receiving 50% of the total capital, the second asset allocated 20%, the third assigned 10%, and the final asset given the remaining 20%. In the initial configuration, each gene is assigned an equal weight (i.e., $W_i = 1/N$ for each asset i). Subsequently, such weights are adjusted using specific operators during the evolutionary process.

3.6.2 Operators

In our method, we make use of three well-established GA operators to produce fresh individuals as part of the evolutionary process: *elitism*, *one-point crossover*, and *one-point mutation*. Elitism is employed to safeguard the top-performing individuals, while crossover and mutations serve to introduce new genetic material. By combining these operators, we create diverse populations that may contain optimal solutions. Following the application

of crossover and mutation operators, we perform a renormalisation of each GA individual. This step ensures that the sum of the weights assigned to assets within each individual continues to equal 1, reflecting the total capital to be invested in the portfolio.

3.6.3 Fitness function

In the training phase, we evaluate the performance of GA individuals using a fitness function. In the current literature, different metrics have been utilised as fitness functions to address portfolio optimisation problems. In our research, we utilise the Sharpe Ratio as our fitness function. The Sharpe Ratio is determined as the ratio of the difference between the mean return and the risk-free rate to the standard deviation of returns. Specifically, the Sharpe Ratio is calculated using the following formula:

$$S = \frac{r - r_f}{\sigma_r}, \tag{12}$$

where: r represents the average return on the investment. r_f is the risk-free rate, which denotes the minimum return expected from an investment with zero risk of default, such as government bonds. σ_r signifies the standard deviation of returns. The average return for each asset is computed as the simple average of its returns over time, using the following formula:

$$r = \frac{\sum_i^N r_i}{N}. \tag{13}$$

here r_i represents the return observed at each time point i, and N is the total number of observations during the training period. Additionally, the standard deviation of returns is determined as the square root of the average of the squared differences between the average return and each observed return, as shown in the following formula:

$$\sigma_r = \sqrt{\frac{\sum_i^N (r - r_i)^2}{N}}. \tag{14}$$

We defer the discussion about which algorithms and benchmarks were considered and compared during the optimisation process, the specific time periods, and the range of metrics used for evaluating and comparing their performance to Sects. 4.3.2 and 5.2.

4 Experimental setup

The primary objective of this study is to showcase the advantages of incorporating TAIs into the feature set of ML algorithms used for predicting REIT prices. To achieve this, we have broken down the above goal into two sub-goals: (i) to demonstrate that the use of TAIs leads to a significant reduction in the regression error, and (ii) to illustrate that the incorporation

	US	UK	Australia
Table 4 Eikon Refinitiv tickers used			
Stocks	AAPL, AMZN, BRKb, GOOGL, JNJ, META, MSFT, NVDA, TSLA, UNH	AZN, BATS, BP, DGE, GLEN, GSK, HSBA, RIO, SHEL, ULVR	ANZ, BHP, CBA, CSL, FMG, MQG, NAB, WBC, WES, WOW
Bonds	AFIF, HOLD, IBMN, IUWAA, JNK, KORP, LQD, LQDI, NFLT, RIGS	AGPH, CCBO, DTLE, EMDD, EMES, ERNA, ERNS, FLOS, IHYG, SDHY	CRED, HBRD, IAF, QPON, RCB, RINCINAV, VACF, VAF, VBND, VGB
Real estate	AMT, AVB, CCI, DLR, EQIX, PLD, PSA, SBAC, SPG, WELL	AEWU, AGRP, BLND, BYG, CAL, CREI, CSH, CTPT, DLN, EPICE	BWP, CHC, DXS, GMG, GOZ, GPT, MGR, SCG, SGP, VCX

of TAIs results in a notable enhancement in the financial performance of an investment portfolio that includes REITs.

The following sections outline our experiments in detail: Sect. 4.1 presents the data used in our study; Sect. 4.2 discusses how hyperparameter tuning was performed in the ML algorithms; and Sect. 4.3 provides a comprehensive analysis of the benchmark models used in our experiments.

4.1 Data

We gathered the daily closing price data for our experiments from the *Eikon Refinitiv* database,[6] corresponding to financial instruments across three countries (US, UK, and Australia), and three asset classes (stocks, bonds, and real estate), spanning the period from January 2019 to July 2021. The selection of the US, UK, and Australia was justified by their well-developed and highly liquid financial markets. Additionally, these countries represent different geographical regions and economic environments, allowing for a comprehensive examination of the predictive capabilities of our models across varied market conditions. The diversity of those markets also allows for lower correlation among the selected markets, representing an advantage for investors seeking to diversify their portfolios and reduce risk.

The consideration of three asset classes (stocks, bonds, and real estate) is justified by their different characteristics and roles within an investment portfolio. Stocks are typically associated with higher returns but also higher volatility, while bonds provide more stable returns with lower risk. Real estate, through Real Estate Investment Trusts (REITs), often offers the benefit of diversification due to its historically lower correlation to stocks and bonds. This combination allows for a more comprehensive analysis of the predictive performance of the machine learning algorithms and an enhanced performance of the portfolio built from those assets.

For each of the resulting nine 'country/asset-class' pairs above, we obtained asset-price data from 10 different assets within that category (in other words: 10 stocks, 10 bonds, and 10 REITs from each country), resulting in a dataset pool consisting of a total of 90 datasets (refer to Table 4). The selection of the ten assets in each category was based on market

---

[6] https://eikon.refinitiv.com—Last access: July 2023.

capitalisation (as assessed on 1st January 2019), to ensure that the chosen assets are among the largest and most liquid (i.e. in terms of trading volume) in their respective markets and provide a representative sample for our analysis. The reason for considering only ten assets within each category in the first place relates to the limited availability of suitable data for study: while data for stocks were available for the 2017-2021 period, there was a more limited number of datasets available for the same time frame regarding REITs and bonds. Therefore, the fact that in some markets (especially the UK and Australia), there were few companies with enough data for our period of study (i.e. from 2019 to 2021) naturally forced a constraint on the number of datasets to ten for each market and each asset class.

We remind the reader that in the context of this study the word 'dataset' is used to refer to a single time-series of daily prices for a given asset. To mitigate currency risk, we obtained all data denominated in USD. The currency conversion for each dataset was based on daily exchange rates provided by Eikon Refinitiv for the considered period (2017-2021).

It is worth recognising that numerous price series datasets can exhibit substantial fluctuations, especially in the case of stocks and REITs. As an example, refer to Fig. 1, illustrating the time-series US REIT closing prices for the period spanning from 1st January 2021 to 1st July 2021. This figure clearly illustrates significant downward variations in the trend. These fluctuations have the potential to affect the performance of certain algorithms, particularly ARIMA (one of our benchmark models), which heavily relies on assumptions of stationarity.

Table 5 shows summary statistics referring to the daily return distributions categorised by each of the nine asset classes we have taken into account. The term 'return' in this context is used specifically to refer to the quantity $(N_t - N_{t-1})/N_{t-1}$, i.e. the relative rate of returns, which is the difference between an asset's normalised price difference on a particular day compared to the day before, expressed as a percentage of the latter. For each asset class, we computed the mean, median, standard deviation, interquartile range, and maximum-minimum range to summarise the return distributions. Each asset within an asset class was given an equal weight, and the summary statistics were calculated based on the training period.

The first column shows the average daily return for each asset class. Australian stocks present the highest daily average return at 2.00×10^{-3}, followed by US stocks at 1.10×10^{-3}
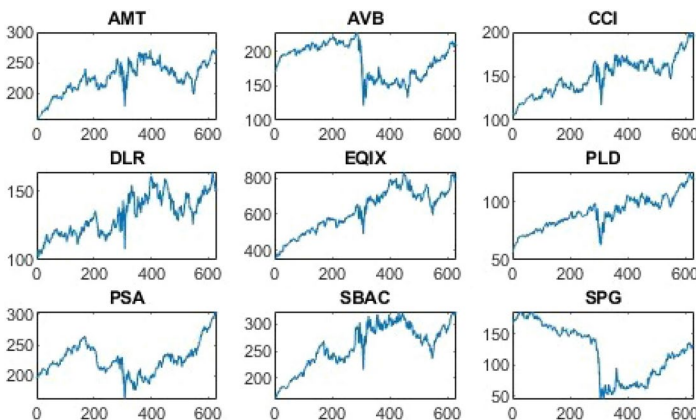

Fig. 1 US REIT time-series. The x-axis represents time in days; the y-axis refers to the price value in USD

Table 5 Summary statistics for different asset classes

	Average	Median	Std Dev	IQR	Max–Min
AU bonds	1.97×10^{-4}	3.15×10^{-4}	$\mathbf{5.70 \times 10^{-3}}$	$\mathbf{3.00 \times 10^{-3}}$	$\mathbf{9.54 \times 10^{-2}}$
AU REITs	7.35×10^{-4}	1.20×10^{-3}	2.44×10^{-2}	1.87×10^{-2}	2.95×10^{-1}
AU stocks	$\mathbf{2.00 \times 10^{-3}}$	$\mathbf{1.80 \times 10^{-3}}$	2.44×10^{-2}	2.14×10^{-2}	2.59×10^{-1}
UK bonds	2.38×10^{-4}	3.86×10^{-4}	7.90×10^{-3}	5.70×10^{-3}	1.12×10^{-1}
UK REITs	7.11×10^{-5}	4.35×10^{-4}	2.56×10^{-2}	2.14×10^{-2}	3.51×10^{-1}
UK stocks	1.88×10^{-4}	3.83×10^{-5}	2.14×10^{-2}	1.93×10^{-2}	2.61×10^{-1}
US bonds	3.11×10^{-4}	2.74×10^{-4}	8.50×10^{-3}	7.70×10^{-3}	1.07×10^{-1}
US REITs	6.99×10^{-4}	7.25×10^{-4}	2.59×10^{-2}	1.95×10^{-2}	3.49×10^{-1}
US stocks	1.10×10^{-3}	1.20×10^{-3}	2.25×10^{-2}	1.86×10^{-2}	2.40×10^{-1}

Values in bold denote the best values for each column

, and Australian REITs at 7.35×10^{-4}. The highest median value is observed for Australian stocks at 1.80×10^{-3}, followed by Australian REITs and US stocks at 1.20×10^{-3}, and US REITs at 7.25×10^{-4}. Stocks tend to have higher rates of return compared to other asset classes such as REITs and bonds.

As for the standard deviation of returns, Australian bonds exhibit the lowest volatility value at 5.70×10^{-3}, followed by UK bonds at 7.90×10^{-3}, and US bonds at 8.50×10^{-3}. Similarly, the lowest interquartile range is observed for Australian bonds at 3.00×10^{-3}, followed by UK bonds at 5.70×10^{-3}, and US bonds at 7.70×10^{-3}. The maximum-minimum ranges show the lowest value for Australian bonds at 9.54×10^{-2}, followed by US bonds at 1.07×10^{-1}, and UK bonds at 1.12×10^{-1}. This is expected since bond rates of return tend to be less volatile than those of other asset classes.

In summary, our findings indicate that bond rates of return exhibit lower volatility and lower average values compared to other asset classes. In contrast, stock markets tend to be more volatile but also offer higher potential profitability than other asset classes. Real estate returns fall in between in terms of expected return and volatility. This observation explains why portfolios that include real estate tend to demonstrate a balance of higher returns and lower risks when compared to portfolios composed uniquely of stocks and bonds (Habbab et al. 2022).

Furthermore, it is important to highlight that the correlation between real estate asset classes and the other asset classes tends to be low, particularly when investing internationally, which provides diversification benefits and consequently reduces the overall risk level of a mixed-asset portfolio (refer to Fig. 2). For example, the correlation between UK REITs and Australian stocks is -0.23, the correlation between UK REITs and US bonds is 6.66×10^{-4}, and the correlation between US REITs and Australian stocks is 0.12. In contrast, the correlation between US stocks and Australian bonds is 0.89, the correlation between UK stocks and UK bonds is 0.81, and the correlation between Australian stocks and US stocks is 0.78. These values illustrate why adding international REIT investments to a portfolio can help to mitigate risk, as per the MPT.

4.2 Experimental tuning of hyperparameters

Before employing our machine learning algorithms to make predictions on the selected datasets, we conducted a process of hyperparameter tuning for each machine learning algorithm. Such tuning was carried out to ensure that each dataset was equipped with a tailored set of hyperparameters. To find the optimal hyperparameters, we adopted the 'Grid Search' method, a recognised technique in the field of machine learning. The selection of hyperparameter value ranges was guided by the specific attributes and requirements of the datasets under consideration. It is worth noting that hyperparameter tuning was not performed for the LR model, as it lacks hyperparameters that require tuning.

The dataset has been split in the following way: the training period is between January 2019 and July 2020, i.e. for 19 months (corresponding to 218 data points); the validation period is between July 2020 and January 2021, i.e. for 5 months (resulting in 110 data points); and the testing period is between January 2021 and July 2021, i.e. for 7 months (resulting in 154 data points). While the number of data points resulting from the above timeline may seem somewhat limited, this choice was motivated by the need to ensure consistency across the asset classes (REITs, stocks, and bonds), given the limited availability
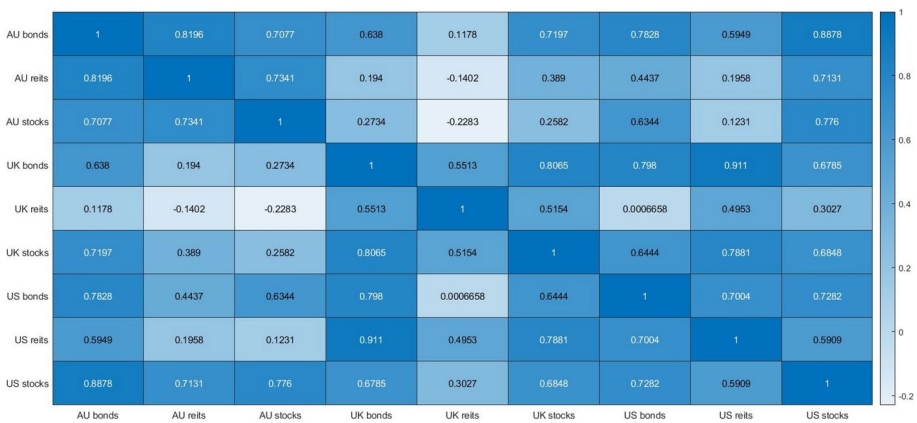
	AU bonds	AU reits	AU stocks	UK bonds	UK reits	UK stocks	US bonds	US reits	US stocks
AU bonds	1	0.8196	0.7077	0.638	0.1178	0.7197	0.7828	0.5949	0.8878
AU reits	0.8196	1	0.7341	0.194	-0.1402	0.389	0.4437	0.1958	0.7131
AU stocks	0.7077	0.7341	1	0.2734	-0.2283	0.2582	0.6344	0.1231	0.776
UK bonds	0.638	0.194	0.2734	1	0.5513	0.8065	0.798	0.911	0.6785
UK reits	0.1178	-0.1402	-0.2283	0.5513	1	0.5154	0.0006658	0.4953	0.3027
UK stocks	0.7197	0.389	0.2582	0.8065	0.5154	1	0.6444	0.7881	0.6848
US bonds	0.7828	0.4437	0.6344	0.798	0.0006658	0.6444	1	0.7004	0.7282
US reits	0.5949	0.1958	0.1231	0.911	0.4953	0.7881	0.7004	1	0.5909
US stocks	0.8878	0.7131	0.776	0.6785	0.3027	0.6848	0.7282	0.5909	1

Fig. 2 Correlation matrix between asset classes

Table 6 TA hyperparameters

Parameter	Indicator	Values
α	EMA	0.01, 0.05, 0.1
Short-day	MACD	20
Long-day	MACD	50
D	Bollinger bands	2

of data simultaneously present across all three of them. It is worth noting that some REITs, such as AEWU and AGRP in the UK and CHC in Australia, entered the market relatively recently (2019), which constrained the available data to that year onwards. Similarly, some bonds ETFs also began trading recently, including AFIF, HOLD, KORP, and NFLT in the US; AGPH and DTLE in the UK; and HBRD and VBND in Australia. Given that price predictions are subsequently used for portfolio optimisation, it is crucial that all datasets cover the same time period. Since some REITs only began trading in 2019, we needed all asset classes' data to start from that year.

We also conducted a 'Grid Search' tuning procedure for each dataset to optimise hyperparameters related to the TAIs discussed in Sect. 3.3. In particular, we determined the best value for α in the Exponential Moving Average (EMA) calculation from the set 0.01, 0.05, 0.1, as suggested in previous research (Maricar 2019). For the remaining hyperparameter values of the TAIs, we referred to established practices outlined in prior works (Hung 2016; Bollinger 1992). The candidate values for α and the selected values for the remaining TAI hyperparameters are shown in Table 6.

Hyperparameter values for the GA were fine-tuned using the identical validation dataset. The optimised hyperparameter values obtained from this tuning process are detailed in Table 7.

4.3 Benchmarks

As mentioned at the beginning of Sect. 4, our two sub-goals are to demonstrate the effectiveness of the use of TAIs in the price prediction task, and in the portfolio optimisation task. In order to investigate the benefits of using TAIs in the feature set, we employ and

Table 7 GA hyperparameters

Parameter	Values
Population size	500
Tournament size	3
Mutation rate	0.1
Number of generations	25

compare against several benchmarks, in accordance with the above two sub-goals. Section 4.3.1.1 presents the benchmarks chosen in relation to the regression task (four in total), and Sect. 4.3.2 presents the benchmarks chosen for the portfolio optimisation task (four in total).

4.3.1 Regression task benchmarks

4.3.1.1 Autoregression with ML In Sect. 3.3, we outlined the various features adopted to solve our regression problem. To evaluate the potential enhancement in the predictive power of the adopted ML algorithms from incorporating TAIs together with lagged values for predicting our time-series data, we conducted a comparative analysis. In particular, we assessed the performance of five ML algorithms that incorporated both lagged prices and TAIs (that is, our proposed approach). Such an assessment was done in comparison to the performance of five ML algorithms that relied uniquely on lagged prices (i.e., excluding TAIs), which is a common practice in the REIT literature. In our analysis, the dependent variable under consideration is denoted as N_t, while the independent variables are drawn from past observations, specifically $N_{t-1}, N_{t-2}, ..., N_{t-T}$, *excluding* TAIs.

4.3.1.2 Holt's linear trend method *Holt's Linear Trend Method* (HLTM; also known as 'Double-Exponential Smoothing' due to the involvement of two exponentially weighted moving average processes in its formulation) is a forecasting method that makes a prediction on the basis of a predicted baseline at the last known data point, and a linear trend extending from that point into the future. It is an extension of Simple Exponential Smoothing that adds a trend component to the model, and where that trend itself is also the result of a Simple Exponential Smoothing process over past trends.

HLTM has two smoothing parameters, α and β, which control the weight given to the most recent observation and the trend, respectively. The forecast equation for HLTM is as follows:

$$\hat{N}_{t+h|t} = l_t + hb_t, \tag{15}$$

where ℓ_t is the level estimate at time t, b_t is the trend estimate at time t, and h is the number of periods ahead to forecast. The level and trend estimates are updated at each time step as follows:

$$l_t = \alpha N_t + (1 - \alpha)(l_{t-1} + b_{t-1}) \tag{16}$$

$$b_t = \beta(l_t - l_{t-1}) + (1 - \beta)b_{t-1}, \tag{17}$$

where N_t is the observed value at time t, and $0 < \alpha < 1$ and $0 < \beta < 1$.

The HLTM method has been widely used in forecasting and has been shown to perform well in many different applications (Hyndman and Athanasopoulos 2018). Given that it uses a weighted average of past observations to make its predictions, it is not able to also use TAIs in its feature set. Nevertheless, it forms a valuable benchmark, as it allows us to compare the performance of our proposed approach with a well-known time-series prediction benchmark.

4.3.1.3 TBATS *Trigonometric Box-Cox Autoregressive Time Series* (TBATS) is a state-of-the-art forecasting model that extends the traditional exponential smoothing framework to handle complex time-series with multiple seasonal patterns and non-linear trends. TBATS was proposed by De Livera et al. (2011).

The TBATS model involves the decomposition of a time-series into multiple components: a non-seasonal component, seasonal components, and an autoregressive component. The non-seasonal component captures the overall trend of the time-series and is modelled using a Box-Cox transformation and an exponential smoothing model. The seasonal components capture the periodic patterns in the time-series and are modelled using a set of trigonometric functions. Finally, the autoregressive component captures the temporal dependencies in the time-series and is modelled using an Autoregressive Moving Average (ARMA) model.

The TBATS model can be written as:

$$N_t = \mu_t + \sum_{j=1}^{J} \gamma_j s_{t,j} + \sum_{i=1}^{p} \phi_i N_{t-i} + \sum_{i=1}^{q} \theta_i e_{t-i} + e_t \tag{18}$$

where N_t is the observed value of the time-series at time t, μ_t is the non-seasonal component at time t, $s_{t,j}$ is the seasonal component for season j at time t, γ_j is the coefficient for season j, p and q are the orders of the autoregressive and moving average components, respectively, ϕ_i and θ_i are the corresponding coefficients, e_t is the error term at time t, and J is the number of seasonal patterns in the data.

TBATS has been shown to outperform traditional forecasting models such as ARIMA and exponential smoothing on time-series with multiple seasonal patterns and non-linear trends (Hyndman and Athanasopoulos 2018). Similarly to HLTM, TBATS is not able to use TAIs in its feature set; however, it also serves as a valuable benchmark, as yet another well-known and widely-used time-series prediction benchmark.

4.3.1.4 ARIMA *Autoregressive Integrated Moving Average* (ARIMA) is a commonly used time-series model for forecasting. It is a statistical model that uses past values and errors to make predictions. ARIMA models can capture both trend and seasonality in the data and are widely used in many fields, including economics, finance, and engineering.

The ARIMA model is denoted by ARIMA(p, d, q), where p is the order of the autoregressive term, d is the degree of differencing required to make the series stationary, and q is the order of the moving average term. The model assumes that the time-series is stationary, which means that its mean and variance are constant over time.

The ARIMA model can be represented mathematically as:

$$N_t = c + \sum_{i=1}^{p} \phi_i N_{t-i} + \epsilon_t + \sum_{i=0}^{q} \theta_i \epsilon_{t-1} \tag{19}$$

where ϕ represents the autoregression coefficient, θ corresponds to the moving average coefficient, and ϵ indicates the error term of the autoregression model at each time point. The selection process of the appropriate ARIMA model for each training dataset was the *Akaike Information Criterion* (AIC), which helps determine the most suitable values for the model's parameters, denoted as p, d, and q.

It is worth noting that ARIMA models are designed for stationary time-series data, where statistical properties like mean and variance remain constant over time. However, many financial time-series do not exhibit this stationary behaviour. To make these datasets suitable for ARIMA modelling, various transformations, such as differencing, logarithmic transformation, and Box-Cox transformation, are often applied. These transformations help make the data more amenable to the assumptions of the ARIMA model, improving the model's effectiveness in predicting future values.

ARIMA has been widely applied in various fields. For example, it has been used to forecast stock prices (Jiang et al. 2019), electricity demand (Dutta and Ramanathan 2019), and weather variables (Lawal et al. 2018). As with HLTM and TBATS, it is not able to also use TAIs in its feature set, but again enjoys wide use in the financial forecasting literature, and therefore forms a valuable benchmark.

4.3.2 Portfolio optimisation benchmarks

Portfolio optimisation involves running a Genetic Algorithm on the price data predicted by our TAI-enhanced ML algorithms, in order to obtain appropriate weights for the different asset classes for each of the 90 assets that make up a portfolio. The quality of the resulting portfolios is then assessed on the basis of financial metrics calculated from the observed prices for that period. Furthermore, in order to assess the usefulness of TAIs in producing better portfolios, we compare the performance of the above with portfolios that have been optimised with respect to price predictions obtained from the non-TAI-enhanced ML algorithm variants, as in Sect. 4.3.1.1.

For completeness, we also benchmark our proposed approach against portfolios obtained on the basis of predictions made using the HLTM, TBATS, and ARIMA algorithms respectively, as these are well-known prediction algorithms that are widely used in the financial literature. In all cases, we evaluate the results in the test set using three financial metrics: expected return, expected risk, and the Sharpe Ratio.

5 Results

In Sect. 5.1, we assess and compare the performance of the five ML algorithms mentioned in Sect. 3.4 when making use of TAIs in their feature set, against a) the same set of ML algorithms when using only lagged values but no TAIs as features, and b) the three conventional techniques outlined in Sect. 4.3.1.1 (i.e. HLTM, TBATS, and ARIMA), which also rely on lagged values exclusively for their function. In Sect. 5.2, we examine the implications of using TAIs in this manner, in the context of using the obtained algorithmic predictions to

perform optimisation of a multi-asset portfolio using a Genetic Algorithm approach, and the extent to which this affects expected return, risk, and Sharpe Ratio values in the resulting portfolios. In Sect. 5.3, we further analyse the importance of each feature in two distinct ways, by using the SHAP and SAGE algorithms, which are metrics of feature quality that build on the concept of Shapley values (Roth 1988). Finally, Sect. 5.4 examines the computational times involved for the algorithms used, and Sect. 5.5 offers a short discussion on the insights gained from the experimental results.

5.1 Performance

We evaluate and compare the performance of the proposed approaches and benchmarks, by reporting the RMSE mean and standard deviation per asset class for each algorithm across all markets, where the RMSE for each dataset is obtained as per Sect. 3.5. It is worth noting that although ML predictions are made on the price returns, RMSE data is expressed in terms of US dollars to reflect the actual deviation of predicted prices from the actual prices.

Table 8 shows RMSE descriptive statistics for REITs in the case of out-of-sample and one-day-ahead prediction over a 30-, 60-, 90-, 120-, and 150-day period. We note that, in the case of *out-of-sample* prediction, the average RMSE is consistently lower for algorithms that use TAIs when compared to the algorithms that use lagged prices only. This is the case across all periods (30, 60, 90, 120, and 150 days). It is also worth noting that the improvements in RMSE means tend to be large. For example, in the 30-day period, we note a reduction from an 'RMSE means' average of 5.6 (i.e. when averaging the individual RMSE means of each non-TAI model) to an average of 4.0 when TAIs are added into the feature set. We also note even larger improvements in other isolated instances; for example, the 90-day LR features a reduction from 9.70 to 5.94 (i.e. an error reduction of $\approx 39\%$), and the 120-day LSTM features a reduction from 10.99 to 5.87 (i.e. an error reduction of $\approx 46\%$). Furthermore, the ML algorithms using TAIs in their feature set also experience lower average standard deviations when compared to the ML algorithms that do not use TA. In addition, it is worth noting that the performance of the conventional time-series benchmarks (HLTM, TBATS, ARIMA)[7] is generally poor by comparison and consistently outperformed by the machine learning algorithms, regardless of whether TAIs are included in the feature set or not.

A similar picture can be observed with the *one-day-ahead* prediction results. The performance of the algorithms that use TAIs tends to be better than the ones without TAIs, with the only exception being the 30-day SVR and XGBoost, and the 120-day XGBoost entries. However, it is worth noting that, while for the out-of-sample results, the introduction of TAIs led to large reductions in error, this reduction is not as impressive in the case of one-day-ahead predictions. This is to be expected, since this method predicts the next day's value using only real —rather than predicted— values in the test period, and therefore the errors are always going to be much smaller. In fact, this is the case regardless of whether we use TAIs or not. As a result, the margin for improvements is also small. Nevertheless, the fact remains that when using TAIs, we still observe consistent average RMSE improvements.

---

[7]As mentioned in Sect. 3, due to the autoregressive elements of HLTM, TBATS, and ARIMA, they cannot use TAIs in their feature set. Hence the relevant rows under the 'With TA' headings in Table 8, and across all remaining tables in this paper, are empty.

Table 8 RMSE summary statistics for REITs

30 Days	Out-of-sample				One-day-ahead			
	Without TA		With TA		Without TA		With TA	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
LR	5.60	12.49	**4.55**	**7.55**	**1.04**	2.10	**1.04**	**1.38**
SVR	5.59	12.45	**4.13**	**7.28**	**1.02**	2.01	1.06	1.41
KNN	5.61	12.53	**4.06**	**7.36**	1.03	2.04	**1.02**	**1.41**
XGBOOST	5.60	12.49	**3.83**	**6.53**	**1.02**	2.00	1.04	**1.62**
LSTM	5.60	12.57	**3.54**	**5.44**	1.08	2.16	**1.01**	**1.24**
HLTM	21.77	40.15			6.47	14.23		
TBATS	21.77	40.15			6.47	14.23		
ARIMA	21.47	38.98			6.69	14.68		
60 days								
LR	7.47	14.79	**6.97**	**14.09**	2.40	5.76	**2.22**	**4.00**
SVR	7.46	14.75	**5.66**	**10.05**	2.40	5.76	**2.33**	**4.04**
KNN	7.48	14.82	**5.73**	**11.05**	2.39	5.75	**2.29**	**4.58**
XGBOOST	7.49	14.87	**5.36**	**8.00**	2.39	5.75	**2.28**	**3.71**
LSTM	7.56	14.50	**5.31**	**7.92**	2.40	5.75	**2.27**	**2.58**
HLTM	16.87	35.63			10.28	24.67		
TBATS	16.87	35.63			10.28	24.67		
ARIMA	17.08	35.82			10.60	25.29		
90 days								
LR	9.70	19.79	**5.94**	**10.75**	1.15	2.18	**0.99**	**0.94**
SVR	9.69	19.73	**6.67**	**12.19**	1.13	2.12	**0.98**	**0.95**
KNN	9.70	19.74	**6.64**	**12.14**	1.13	2.12	**0.99**	**1.06**
XGBOOST	9.70	19.78	**5.81**	**9.72**	1.13	2.13	**0.97**	**0.91**
LSTM	9.72	19.86	**6.08**	**8.58**	1.14	2.16	**0.92**	**0.88**
HLTM	20.82	35.66			9.30	17.45		
TBATS	21.28	36.75			9.30	17.45		
ARIMA	20.81	35.67			9.47	17.78		
120 days								
LR	10.96	16.75	**7.19**	**11.43**	1.16	2.22	**1.14**	**1.23**
SVR	10.95	16.73	**6.63**	**10.10**	1.14	2.16	**1.11**	**1.22**
KNN	10.97	16.79	**6.55**	**8.55**	**1.14**	2.16	**1.14**	**1.23**
XGBOOST	10.95	16.75	**6.30**	**8.92**	**1.14**	2.16	1.15	**1.32**
LSTM	10.99	16.81	**5.87**	**7.39**	1.17	2.23	**1.15**	**1.25**
HLTM	22.91	35.97			9.83	15.19		
TBATS	22.91	35.97			9.83	15.19		
ARIMA	22.88	35.95			10.01	15.51		
150 days								
LR	8.00	12.91	**5.90**	**10.27**	1.16	2.19	**1.11**	**1.96**
SVR	8.00	12.91	**5.14**	**6.90**	1.15	2.16	**1.05**	**1.10**
KNN	8.02	12.96	**5.11**	**6.59**	1.15	2.16	**1.06**	**1.21**
XGBOOST	8.00	12.91	**4.97**	**7.46**	1.15	2.17	**1.04**	**1.11**
LSTM	8.00	12.92	**4.51**	**7.52**	1.16	2.18	**1.04**	**1.04**
HLTM	17.32	27.37			7.91	12.70		
TBATS	17.32	27.37			7.91	12.70		
ARIMA	16.91	26.80			8.07	13.00		

Values in bold denote the best values for each row

We can observe a similar pattern in the results for stocks (Table 9). Machine learning algorithms that incorporate TAIs in their feature set consistently show better results in terms of mean RMSE and standard deviation for both out-of-sample and one-day-ahead predictions, across all periods (30, 60, 90, 120, and 150 days). It is also worth noting that both the means and standard deviations of the RMSE values here tend to be higher than the respective values in Table 8; this can be explained by the more volatile nature of stock data, which makes it much harder to predict accurately.

Finally, Table 10 shows the RMSE distribution statistics for bonds. One noticeable difference here compared to the previous two tables is the very low mean and standard deviation values observed across all algorithms and methodologies. This is due to the nature of bonds, which have very low volatility and are thus much easier to predict. With regard to the comparison of results when using TAIs, we again observe that the introduction of TAIs leads to consistent improvements in the out-of-sample results, whereas in the one-day-ahead case, due to the very low error values involved, the results are more mixed, with TAI algorithms occasionally being marginally outperformed by their respective non-TAI counterparts.

To compare the distributions of RMSE scores resulting from the use of TAIs versus non-TAI machine learning algorithms, we conducted Kolmogorov-Smirnov (KS) tests at a 5% significance level for all asset classes. The null hypothesis being tested is that the RMSE distributions being compared originate from the same continuous distribution. Given that we are conducting five separate comparisons, one for each considered period (i.e., 30-, 60-, 90-, 120-, and 150-day periods), we adjusted the significance level (α) using the Bonferroni correction, resulting in $\alpha = 0.01$.

For out-of-sample predictions, the KS test p-values were consistently less than 0.001 in all cases, significantly lower than the adjusted α threshold of 0.01. Specifically, for each of the 30-, 60-, 90-, 120-, and 150-day periods, the obtained p-values were approximately 1.02×10^{-9}, 7.14×10^{-7}, 5.66×10^{-8}, 4.60×10^{-8}, and 2.61×10^{-7}, respectively. These results strongly indicate that the inclusion of TAIs leads to a marked reduction in RMSE.

Conversely, for one-day-ahead predictions, the KS test p-values were not significant, with values like 1.04×10^{-1}, 7.37×10^{-1}, 6.17×10^{-1}, 3.07×10^{-1}, and 1.48×10^{-1} for the respective periods. This suggests that there is no substantial difference in the RMSE distributions between the TAI and non-TAI models. However, it is important to note that in the case of one-day-ahead predictions, RMSE values tend to be very small, making it challenging to achieve statistically significant results, despite the observed slight reduction in RMSE scores.

In summary, our analysis reveals that the RMSE distributions exhibit lower average values and less variability for machine learning algorithms incorporating TAIs compared to benchmark algorithms, which is more evident in the case of out-of-sample predictions. In general, we observed that the reduction in RMSE mean values due to the inclusion of TAIs can be substantial, with reductions of up to 45%. Furthermore, we noted that the lowest average RMSE values were consistently observed in bond predictions, followed by REITs, and then stocks. This ranking aligns with the relative volatility of these asset classes, as discussed in Sect. 4.1. Bonds, known for their lower price volatility, exhibit the lowest average RMSE values. REITs, which typically fall between bonds and stocks in terms of risk and return characteristics, show higher average RMSE values than bonds but lower than stocks. Our findings, supported by the Kolmogorov-Smirnov (KS) test results, emphasise that the

Table 9 RMSE summary statistics for stocks

	Out-of-sample				One-day-ahead			
	Without TA		With TA		Without TA		With TA	
30 Days	Mean	SD	Mean	SD	Mean	SD	Mean	SD
LR	9.19	20.62	**7.61**	**16.62**	2.28	4.38	**2.18**	**2.34**
SVR	9.16	20.51	**6.84**	**12.92**	2.29	4.44	**2.16**	**2.60**
KNN	9.21	20.66	**7.69**	**14.72**	2.30	4.45	**2.15**	**2.18**
XGBOOST	9.19	20.58	**6.25**	**10.31**	2.30	4.48	**2.17**	**2.29**
LSTM	9.11	20.21	**6.38**	**9.47**	2.31	4.39	**2.13**	**2.27**
HLTM	41.21	100.20			11.36	26.54		
TBATS	41.21	100.20			11.36	26.54		
ARIMA	41.47	100.15			11.72	27.15		
60 days								
LR	12.34	23.85	**11.00**	**21.10**	2.77	5.64	**2.72**	**3.05**
SVR	12.32	23.75	**10.48**	**21.89**	2.77	5.64	**2.65**	**3.10**
KNN	12.30	23.80	**11.31**	**21.99**	2.76	5.63	**2.54**	**2.76**
XGBOOST	12.31	23.75	**9.35**	**16.18**	2.77	5.63	**2.70**	**3.03**
LSTM	12.29	23.67	**10.03**	**13.88**	2.77	5.63	**2.55**	**2.23**
HLTM	30.29	71.90			12.38	24.16		
TBATS	30.29	71.90			12.38	24.16		
ARIMA	31.30	75.72			12.73	24.77		
90 days								
LR	19.45	43.66	**15.37**	**30.84**	**3.25**	6.97	3.29	**2.78**
SVR	19.45	43.63	**13.20**	**24.21**	3.35	7.31	**3.33**	**2.77**
KNN	19.39	43.57	**14.11**	**27.87**	**3.24**	6.96	3.29	**2.79**
XGBOOST	19.44	43.61	**11.89**	**18.21**	**3.25**	7.00	3.27	**2.63**
LSTM	19.44	43.53	**11.73**	**18.59**	3.25	6.97	**3.24**	**2.64**
HLTM	42.37	98.42			18.72	44.32		
TBATS	42.85	100.01			18.72	44.32		
ARIMA	42.37	98.45			19.08	44.93		
120 days								
LR	28.90	85.13	**21.89**	**62.52**	3.39	7.47	**3.38**	**2.41**
SVR	28.87	84.97	**18.29**	**52.29**	3.45	7.70	**3.14**	**2.57**
KNN	28.82	84.91	**17.19**	**29.41**	3.36	7.43	**3.35**	**2.51**
XGBOOST	28.88	85.06	**16.79**	**22.44**	3.39	7.49	**3.21**	**2.33**
LSTM	28.89	85.01	**16.56**	**16.86**	3.36	7.39	**3.18**	**2.27**
HLTM	62.94	192.98			28.82	81.52		
TBATS	62.94	192.98			28.82	81.52		
ARIMA	62.76	193.89			29.20	82.25		
150 days								
LR	28.62	75.28	**22.93**	**52.34**	3.28	7.15	**3.21**	**1.96**
SVR	28.55	75.08	**20.01**	**34.49**	3.28	7.18	**3.14**	**2.08**
KNN	28.58	75.07	**20.02**	**38.75**	3.27	7.13	**3.07**	**2.02**
XGBOOST	28.62	75.24	**19.09**	**24.23**	3.27	7.15	**3.16**	**3.13**
LSTM	28.46	74.86	**19.03**	**23.88**	3.27	7.13	**3.12**	**2.20**
HLTM	71.50	190.19			29.09	75.40		
TBATS	71.50	190.19			29.09	75.40		
ARIMA	71.46	190.44			28.78	75.06		

Values in bold represent the best results for each row

Table 10 RMSE summary statistics for bonds

| 30 Days | Out-of-sample | | | | One-day-ahead | | | |
| | Without TA | | With TA | | Without TA | | With TA | |
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
LR	0.51	0.56	**0.49**	**0.55**	**0.17**	**0.18**	0.21	0.20
SVR	0.51	**0.56**	**0.45**	0.60	**0.17**	0.18	0.19	**0.17**
KNN	0.51	**0.56**	**0.44**	0.62	**0.17**	0.18	**0.17**	**0.17**
XGBOOST	0.51	0.56	**0.43**	**0.41**	**0.17**	**0.18**	0.18	0.21
LSTM	0.52	0.56	**0.45**	**0.37**	0.18	0.18	**0.17**	**0.16**
HLTM	1.22	1.48			0.48	0.57		
TBATS	1.16	1.37			0.48	0.57		
ARIMA	1.22	1.48			0.51	0.60		
60 days								
LR	**0.58**	0.73	**0.58**	0.55	0.17	0.17	**0.16**	**0.15**
SVR	0.58	0.73	**0.55**	0.58	0.17	0.17	**0.17**	**0.15**
KNN	0.58	0.73	**0.52**	0.56	0.17	0.17	0.18	0.17
XGBOOST	0.58	0.73	**0.53**	0.64	0.17	0.17	**0.17**	**0.13**
LSTM	0.59	0.74	**0.56**	0.31	0.18	0.18	0.18	0.13
HLTM	0.93	1.24			0.60	0.68		
TBATS	0.93	1.24			0.60	0.68		
ARIMA	0.96	1.29			0.62	0.69		
90 days								
LR	0.87	0.89	**0.61**	**0.52**	**0.20**	**0.20**	0.22	0.24
SVR	0.87	0.89	**0.79**	**0.68**	**0.20**	0.20	0.20	**0.15**
KNN	0.87	0.89	**0.74**	**0.59**	0.20	0.19	**0.18**	**0.17**
XGBOOST	0.87	0.90	**0.63**	**0.51**	**0.20**	**0.20**	0.21	0.22
LSTM	0.88	0.90	**0.66**	**0.40**	**0.20**	0.20	**0.20**	**0.17**
HLTM	1.74	2.05			0.85	0.86		
TBATS	1.74	2.05			0.85	0.86		
ARIMA	1.72	2.02			0.87	0.88		
120 days								
LR	0.94	1.12	**0.91**	**0.79**	**0.19**	0.19	**0.19**	**0.18**
SVR	0.93	1.10	**0.81**	**0.66**	**0.19**	**0.19**	0.20	0.24
KNN	0.93	1.12	**0.79**	**0.65**	**0.19**	0.18	0.20	**0.15**
XGBOOST	0.94	1.12	**0.78**	**0.75**	0.20	0.20	**0.19**	**0.16**
LSTM	0.94	1.12	**0.77**	**0.49**	0.20	**0.19**	**0.18**	0.20
HLTM	2.05	2.48			0.99	1.19		
TBATS	2.05	2.48			0.99	1.19		
ARIMA	2.07	2.51			1.01	1.20		
150 days								
LR	1.03	1.26	**0.94**	**1.10**	0.20	0.19	**0.19**	**0.16**
SVR	1.03	1.26	**0.94**	**0.47**	**0.19**	**0.19**	0.20	**0.19**
KNN	1.04	1.26	**0.93**	**0.94**	0.20	0.19	0.21	**0.16**
XGBOOST	1.03	1.26	**0.95**	**0.89**	0.20	0.19	**0.18**	**0.14**
LSTM	1.04	1.25	**0.94**	**0.90**	0.20	0.19	**0.18**	**0.16**
HLTM	1.79	2.37			1.03	1.28		
TBATS	1.79	2.37			1.03	1.28		
ARIMA	1.83	2.41			1.05	1.29		

Values in bold represent the best results for each row

adoption of an out-of-sample methodology significantly reduces RMSE mean values, suggesting that the use of TAIs in machine learning models contributes to enhanced predictive accuracy. On the other hand, the KS test results show that the RMSE values tend to be similar for machine learning models using TAIs and those not including those additional features in the case of one-day-ahead predictions. Although the inclusion of TAIs does not always lead to an enhancement in the predictive performance of ML (especially in the case of short-term predictions), the KS tests performed highlight the significant improvement in the ML predictive capability when including TAIs for long-term predictions.

5.2 Portfolio optimisation

This section contains the results of the Genetic Algorithm (GA) applied to portfolio allocation, which takes into account a transaction cost of 0.02%. The GA was used to generate 100 optimised portfolios per algorithm considered. For each generated portfolio, the optimised weights were used to calculate the expected return, expected risk, and expected Sharpe Ratio for the portfolio. These were then pooled over all generated portfolios to create and analyse the distributions of expected returns (Sect. 5.2.1), expected risks (Sect. 5.2.2), and expected Sharpe Ratios (Sect. 5.2.3) respectively. The following subsections provide a summary of key statistics for these metrics, namely the mean and standard deviation. We compare the performance of our proposed approaches, i.e. ML models that utilise TAIs as additional features, to benchmarks, which consist of portfolios built using ML models, as well as HLTM, TBATS, and ARIMA.

5.2.1 Expected portfolio returns

Table 11 shows descriptive statistics for expected return distributions obtained from the GA portfolio optimisation for a 30-, 60-, 90-, 120-, and 150-day holding period. For a 30-day prediction period, we observe an increase in the average expected return obtained from out-of-sample predictions resulting from TAI models. On the other hand, the standard deviation values are not always lower for the proposed approaches. For HLTM, TBATS, and ARIMA models, the average expected return values appear to be lower compared to the proposed approaches. In the case of the one-day-ahead predictions, the average and standard deviation of the expected return distributions also improve when introducing TAIs. For instance, the best result is observed for the KNN algorithm (3.78×10^{-3}), which shows an improvement of almost 175% when adding TAIs. The HLTM, TBATS, and ARIMA algorithms show lower expected return values compared to the ML algorithms that use TAIs. We can observe similar improvements brought by the use of the TAIs for the remaining periods (60, 90, 120, and 150 days) across both out-of-sample and one-day-ahead methods. Standard deviation results are more mixed, with the best performance alternating between the setups that use TAIs and those that do not.

To compare the expected return distributions obtained via TAI models versus non-TAI models from ML algorithms that use TAIs as additional features and those obtained from algorithms that use lagged values only, we conducted a Kolmogorov–Smirnov (KS) test at the 5% significance level. Here again, the null hypothesis assumes that the compared return distributions arise from the same continuous distribution. We performed five comparisons (one for each prediction period, i.e. 30, 60, 90, 120, and 150 days), and to account

for multiple comparisons, we again applied Bonferroni's correction by adjusting the alpha value to $0.05/5 = 0.01$. For out-of-sample predictions, the KS test produced p-values of 7.17×10^{-28}, 3.59×10^{-27}, 8.24×10^{-24}, 1.12×10^{-20}, and 3.27×10^{-17} for 30-, 60-, 90-, 120-, and 150-day periods, respectively, which are much lower than the adjusted significance level of 0.01, suggesting that the use of TAIs leads to a significant improvement in the expected return distributions. Similarly, for one-day-ahead predictions, the KS test produced p-values of 5.11×10^{-33}, 3.71×10^{-36}, 3.97×10^{-25}, 3.59×10^{-27}, and 5.11×10^{-33} for 30-, 60-, 90-, 120-, and 150-day periods, respectively, leading to the same conclusion as for the out-of-sample scenario.

In conclusion, the results obtained from the Genetic Algorithm confirm that the portfolios obtained using TAI-models can lead to improvements of up to 150% in the case of out-of-sample predictions, and of up to 230% in the case of one-day-ahead predictions. According to the KS test results, there is a statistically significant difference in the expected return distributions for the machine learning algorithms when introducing TAIs as additional features. The HLTM, TBATS, and ARIMA methods generally show lower expected return average values compared to the proposed approaches. Our results show the importance of (i) using TAIs as additional features vs using lagged prices only; and (ii) using ML models vs traditional financial benchmarks.

5.2.2 Expected portfolio risks

Table 12 shows descriptive statistics for the expected portfolio risks for a 30-, 60-, 90-, 120-, and 150-day testing period in the case of out-of-sample and one-day-ahead predictions. In both cases, the average expected risk tends to increase when including TAIs in the regression problem for all periods. As we can observe, in the case of out-of-sample prediction, there is an average increase of around 187% when adding TAIs for a 30-day prediction period (with a decrease in the case of LSTM of around 20%), which drops to around 113% for a 60-day prediction period, to around 70% for a 90-day prediction period, to around 44% for a 120-day prediction period, and rises to 72% for a 150-day prediction period. The standard deviation values are lower for algorithms that use TAIs in most cases, indicating a higher concentration of values around the mean. For instance, the average standard deviation is 5.18×10^{-4} when not using TAIs and 1.44×10^{-4} when using TAIs for a 30-day period. We observe similar values for the other periods. Lastly, it is worth noting that for the first time in our study, the HLTM, TBATS, and ARIMA algorithms outperform the ML algorithms, as they show relatively low average and standard deviation values. For example, the average expected risk is 2.49×10^{-3} obtained from the TBATS algorithm for a 150-day period, which is the lowest observed for the considered period. On the other hand, the lowest volatility of expected risk values is observed for HLTM at 6.94×10^{-18}.

In the case of one-day-ahead predictions, the average expected portfolio risk again tends to be lower when not using TAIs as features, while the standard deviation appears to be lower when using TAIs. In other words, the predictions obtained from algorithms that include TAIs lead to higher expected portfolio risk, but at the same time, the risk values appear to be more concentrated around the mean. This might indicate a lower presence of outliers. For instance, the addition of TAIs as features results in an average expected risk increase from 3.54×10^{-3} to 1.11×10^{-2} for the KNN algorithm and 60-day prediction period, while its standard deviation appears to be reduced from 7.27×10^{-4} to 1.06×10^{-4}

Table 11 Expected portfolio return summary statistics

	Out-of-sample				One-day-ahead			
	Without TA		With TA		Without TA		With TA	
30 Days	Mean	SD	Mean	SD	Mean	SD	Mean	SD
LR	1.12×10^{-3}	$\mathbf{4.75 \times 10^{-6}}$	3.07×10^{-3}	2.16×10^{-4}	1.31×10^{-3}	3.52×10^{-4}	$\mathbf{3.51 \times 10^{-3}}$	$\mathbf{1.93 \times 10^{-4}}$
SVR	1.44×10^{-3}	3.95×10^{-4}	3.40×10^{-3}	$\mathbf{2.02 \times 10^{-4}}$	1.44×10^{-3}	4.92×10^{-4}	$\mathbf{3.55 \times 10^{-3}}$	$\mathbf{1.67 \times 10^{-4}}$
KNN	1.43×10^{-3}	$\mathbf{1.25 \times 10^{-5}}$	3.46×10^{-3}	2.05×10^{-4}	1.38×10^{-3}	2.95×10^{-4}	$\mathbf{3.78 \times 10^{-3}}$	$\mathbf{1.69 \times 10^{-4}}$
XGBoost	1.23×10^{-3}	5.04×10^{-4}	3.45×10^{-3}	$\mathbf{2.06 \times 10^{-4}}$	1.47×10^{-3}	$\mathbf{1.50 \times 10^{-4}}$	3.49×10^{-3}	1.99×10^{-4}
LSTM	1.44×10^{-3}	$\mathbf{1.69 \times 10^{-4}}$	3.36×10^{-3}	1.86×10^{-4}	2.72×10^{-3}	2.37×10^{-4}	$\mathbf{3.70 \times 10^{-3}}$	$\mathbf{1.57 \times 10^{-4}}$
HLTM	9.06×10^{-4}	1.78×10^{-6}			9.62×10^{-4}	1.79×10^{-4}		
TBATS	1.93×10^{-4}	7.73×10^{-5}			9.02×10^{-4}	3.79×10^{-4}		
ARIMA	6.73×10^{-4}	2.85×10^{-5}			1.25×10^{-3}	4.35×10^{-4}		
60 days								
LR	8.40×10^{-4}	3.49×10^{-4}	2.98×10^{-3}	$\mathbf{2.13 \times 10^{-4}}$	1.86×10^{-3}	1.97×10^{-4}	3.40×10^{-3}	$\mathbf{1.45 \times 10^{-4}}$
SVR	1.52×10^{-3}	6.36×10^{-4}	2.51×10^{-3}	$\mathbf{2.34 \times 10^{-4}}$	1.48×10^{-3}	$\mathbf{1.90 \times 10^{-4}}$	3.13×10^{-3}	2.16×10^{-4}
KNN	1.02×10^{-3}	$\mathbf{9.18 \times 10^{-5}}$	2.90×10^{-3}	1.66×10^{-4}	1.75×10^{-3}	2.17×10^{-4}	3.49×10^{-3}	$\mathbf{1.72 \times 10^{-4}}$
XGBoost	1.58×10^{-3}	6.06×10^{-4}	3.46×10^{-3}	$\mathbf{1.86 \times 10^{-4}}$	2.07×10^{-3}	2.28×10^{-4}	3.66×10^{-3}	$\mathbf{1.76 \times 10^{-4}}$
LSTM	1.26×10^{-3}	$\mathbf{4.09 \times 10^{-5}}$	2.40×10^{-3}	2.45×10^{-4}	1.45×10^{-3}	4.50×10^{-4}	3.62×10^{-3}	$\mathbf{1.80 \times 10^{-4}}$
HLTM	3.81×10^{-4}	2.33×10^{-6}			6.90×10^{-4}	1.55×10^{-4}		
TBATS	2.40×10^{-4}	2.78×10^{-5}			2.84×10^{-4}	1.34×10^{-4}		
ARIMA	6.72×10^{-4}	7.48×10^{-5}			2.12×10^{-3}	2.16×10^{-4}		
90 days								
LR	8.21×10^{-4}	$\mathbf{2.08 \times 10^{-4}}$	2.43×10^{-3}	2.35×10^{-4}	1.74×10^{-3}	$\mathbf{2.06 \times 10^{-4}}$	2.79×10^{-3}	2.37×10^{-4}
SVR	1.35×10^{-3}	4.26×10^{-4}	2.64×10^{-3}	$\mathbf{2.58 \times 10^{-4}}$	1.91×10^{-3}	$\mathbf{1.77 \times 10^{-4}}$	3.48×10^{-3}	1.85×10^{-4}
KNN	1.70×10^{-3}	$\mathbf{2.38 \times 10^{-4}}$	2.44×10^{-3}	2.60×10^{-4}	1.85×10^{-3}	2.68×10^{-4}	3.03×10^{-3}	$\mathbf{2.07 \times 10^{-4}}$
XGBoost	1.42×10^{-3}	2.89×10^{-4}	2.85×10^{-3}	$\mathbf{2.31 \times 10^{-4}}$	1.71×10^{-3}	$\mathbf{1.91 \times 10^{-4}}$	3.00×10^{-3}	1.99×10^{-4}
LSTM	1.40×10^{-3}	4.99×10^{-4}	2.32×10^{-3}	$\mathbf{2.90 \times 10^{-4}}$	1.73×10^{-3}	$\mathbf{1.46 \times 10^{-4}}$	3.06×10^{-3}	2.03×10^{-4}

Table 11 (continued)

	Out-of-sample				One-day-ahead			
	Without TA		With TA		Without TA		With TA	
30 Days	Mean	SD	Mean	SD	Mean	SD	Mean	SD
HLTM	6.49×10^{-4}	4.14×10^{-6}			9.84×10^{-4}	1.73×10^{-4}		
TBATS	1.70×10^{-4}	1.39×10^{-5}			9.62×10^{-4}	1.48×10^{-4}		
ARIMA	3.92×10^{-4}	6.81×10^{-5}			1.91×10^{-3}	1.73×10^{-4}		
120 days								
LR	1.14×10^{-3}	4.15×10^{-4}	$\mathbf{2.21 \times 10^{-3}}$	$\mathbf{2.73 \times 10^{-4}}$	1.49×10^{-3}	$\mathbf{1.76 \times 10^{-4}}$	$\mathbf{2.72 \times 10^{-3}}$	2.33×10^{-4}
SVR	1.14×10^{-3}	2.26×10^{-4}	2.08×10^{-3}	2.47×10^{-4}	1.42×10^{-3}	6.14×10^{-4}	$\mathbf{3.75 \times 10^{-3}}$	$\mathbf{1.99 \times 10^{-4}}$
KNN	1.12×10^{-3}	3.50×10^{-4}	$\mathbf{1.91 \times 10^{-3}}$	1.97×10^{-4}	1.32×10^{-3}	$\mathbf{1.15 \times 10^{-4}}$	$\mathbf{2.92 \times 10^{-3}}$	1.77×10^{-4}
XGBoost	1.11×10^{-3}	2.79×10^{-4}	2.13×10^{-3}	2.55×10^{-4}	1.22×10^{-3}	$\mathbf{1.93 \times 10^{-4}}$	2.57×10^{-3}	1.97×10^{-4}
LSTM	1.15×10^{-3}	2.72×10^{-4}	$\mathbf{2.09 \times 10^{-3}}$	$\mathbf{1.98 \times 10^{-4}}$	1.43×10^{-3}	$\mathbf{1.77 \times 10^{-4}}$	$\mathbf{2.76 \times 10^{-3}}$	2.26×10^{-4}
HLTM	5.19×10^{-4}	1.05×10^{-18}			5.48×10^{-4}	9.70×10^{-5}		
TBATS	1.85×10^{-4}	1.04×10^{-5}			3.75×10^{-4}	1.39×10^{-4}		
ARIMA	3.21×10^{-4}	2.92×10^{-5}			8.56×10^{-4}	8.18×10^{-5}		
150 days								
LR	9.53×10^{-4}	$\mathbf{5.63 \times 10^{-5}}$	1.99×10^{-3}	2.43×10^{-4}	1.51×10^{-3}	$\mathbf{9.19 \times 10^{-5}}$	$\mathbf{3.01 \times 10^{-3}}$	1.87×10^{-4}
SVR	1.17×10^{-3}	3.15×10^{-4}	2.22×10^{-3}	$\mathbf{2.09 \times 10^{-4}}$	1.75×10^{-3}	$\mathbf{1.77 \times 10^{-4}}$	2.61×10^{-3}	2.29×10^{-4}
KNN	9.31×10^{-4}	$\mathbf{4.53 \times 10^{-5}}$	2.26×10^{-3}	2.15×10^{-4}	1.76×10^{-3}	1.34×10^{-4}	2.98×10^{-3}	2.20×10^{-4}
XGBoost	1.27×10^{-3}	$\mathbf{2.31 \times 10^{-4}}$	2.45×10^{-3}	2.37×10^{-4}	1.76×10^{-3}	1.14×10^{-4}	3.01×10^{-3}	2.39×10^{-4}
LSTM	1.03×10^{-3}	3.38×10^{-4}	2.41×10^{-3}	$\mathbf{1.94 \times 10^{-4}}$	1.78×10^{-3}	1.26×10^{-4}	$\mathbf{2.73 \times 10^{-3}}$	1.72×10^{-4}
HLTM	1.13×10^{-4}	1.06×10^{-4}			1.40×10^{-3}	8.89×10^{-5}		
TBATS	1.11×10^{-4}	1.05×10^{-4}			1.38×10^{-3}	1.15×10^{-4}		
ARIMA	6.94×10^{-4}	1.09×10^{-4}			1.65×10^{-3}	1.33×10^{-4}		

Values in bold represent the best results for each row

Table 12 Expected portfolio risk summary statistics

	Out-of-sample				One-day-ahead			
	Without TA		With TA		Without TA		With TA	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
30 Days								
LR	1.86×10^{-3}	3.44×10^{-5}	1.07×10^{-2}	1.85×10^{-4}	1.80×10^{-3}	6.41×10^{-4}	1.12×10^{-2}	1.31×10^{-4}
SVR	6.16×10^{-3}	1.70×10^{-3}	1.11×10^{-2}	1.40×10^{-4}	1.79×10^{-3}	9.90×10^{-4}	1.11×10^{-2}	1.27×10^{-4}
KNN	3.13×10^{-3}	7.79×10^{-5}	1.12×10^{-2}	1.17×10^{-4}	1.90×10^{-3}	1.12×10^{-3}	1.13×10^{-2}	1.28×10^{-4}
XGBoost	4.61×10^{-3}	9.17×10^{-4}	1.12×10^{-2}	1.31×10^{-4}	1.57×10^{-3}	8.05×10^{-4}	1.12×10^{-2}	1.32×10^{-4}
LSTM	3.39×10^{-3}	1.06×10^{-3}	1.09×10^{-2}	1.48×10^{-4}	1.85×10^{-3}	8.39×10^{-4}	1.13×10^{-2}	1.13×10^{-4}
HLTM	8.24×10^{-3}	3.41×10^{-5}			2.66×10^{-3}	6.95×10^{-4}		
TBATS	1.21×10^{-3}	2.75×10^{-4}			2.76×10^{-3}	1.26×10^{-3}		
ARIMA	3.91×10^{-3}	4.27×10^{-5}			8.29×10^{-3}	5.99×10^{-4}		
60 days								
LR	4.03×10^{-3}	3.89×10^{-3}	1.07×10^{-2}	1.65×10^{-4}	4.04×10^{-3}	1.38×10^{-3}	1.09×10^{-2}	1.18×10^{-4}
SVR	6.77×10^{-3}	2.05×10^{-3}	1.04×10^{-2}	1.73×10^{-4}	3.10×10^{-3}	7.58×10^{-4}	1.08×10^{-2}	1.64×10^{-4}
KNN	3.84×10^{-3}	4.49×10^{-4}	1.04×10^{-2}	1.47×10^{-4}	3.54×10^{-3}	7.27×10^{-4}	1.11×10^{-2}	1.06×10^{-4}
XGBoost	6.00×10^{-3}	1.95×10^{-3}	1.10×10^{-2}	1.46×10^{-4}	4.13×10^{-3}	8.27×10^{-4}	1.12×10^{-2}	1.30×10^{-4}
LSTM	5.37×10^{-3}	2.24×10^{-4}	1.02×10^{-2}	1.93×10^{-4}	2.86×10^{-3}	9.96×10^{-4}	1.12×10^{-2}	1.30×10^{-4}
HLTM	4.51×10^{-3}	6.24×10^{-6}			4.96×10^{-3}	5.93×10^{-4}		
TBATS	3.89×10^{-3}	7.39×10^{-5}			5.17×10^{-3}	3.52×10^{-4}		
ARIMA	5.05×10^{-3}	2.37×10^{-4}			1.38×10^{-2}	1.17×10^{-3}		
90 days								
LR	3.71×10^{-3}	3.81×10^{-4}	1.04×10^{-2}	1.54×10^{-4}	4.29×10^{-3}	7.32×10^{-4}	1.06×10^{-2}	1.72×10^{-4}
SVR	5.78×10^{-3}	1.15×10^{-3}	1.07×10^{-2}	1.81×10^{-4}	4.57×10^{-3}	5.03×10^{-4}	1.11×10^{-2}	1.35×10^{-4}
KNN	1.00×10^{-2}	1.55×10^{-3}	1.07×10^{-2}	1.61×10^{-4}	4.82×10^{-3}	1.09×10^{-3}	1.08×10^{-2}	1.42×10^{-4}
XGBoost	8.02×10^{-3}	1.97×10^{-3}	1.05×10^{-2}	2.14×10^{-4}	4.22×10^{-3}	1.37×10^{-3}	1.07×10^{-2}	1.50×10^{-4}
LSTM	6.98×10^{-3}	1.55×10^{-3}	1.07×10^{-2}	1.71×10^{-4}	4.20×10^{-3}	5.59×10^{-4}	1.07×10^{-2}	1.49×10^{-4}

Table 12 (continued)

| | Out-of-sample | | | | One-day-ahead | | | |
| | Without TA | | With TA | | Without TA | | With TA | |
30 Days	Mean	SD	Mean	SD	Mean	SD	Mean	SD
HLTM	5.89×10^{-3}	2.41×10^{-5}			5.92×10^{-3}	1.09×10^{-3}		
TBATS	2.73×10^{-3}	4.62×10^{-5}			5.75×10^{-3}	5.45×10^{-4}		
ARIMA	5.73×10^{-3}	2.92×10^{-4}			1.94×10^{-2}	1.65×10^{-3}		
120 days								
LR	$\mathbf{7.68 \times 10^{-3}}$	1.39×10^{-3}	1.04×10^{-2}	$\mathbf{1.94 \times 10^{-4}}$	$\mathbf{3.89 \times 10^{-3}}$	6.04×10^{-4}	1.05×10^{-2}	$\mathbf{1.74 \times 10^{-4}}$
SVR	$\mathbf{7.32 \times 10^{-3}}$	1.81×10^{-3}	1.02×10^{-2}	$\mathbf{1.65 \times 10^{-4}}$	$\mathbf{2.99 \times 10^{-3}}$	9.70×10^{-4}	1.00×10^{-2}	$\mathbf{1.55 \times 10^{-4}}$
KNN	$\mathbf{7.05 \times 10^{-3}}$	4.35×10^{-4}	9.65×10^{-3}	$\mathbf{1.57 \times 10^{-4}}$	$\mathbf{3.02 \times 10^{-3}}$	1.33×10^{-3}	1.05×10^{-2}	$\mathbf{1.44 \times 10^{-4}}$
XGBoost	$\mathbf{6.52 \times 10^{-3}}$	9.75×10^{-4}	1.00×10^{-2}	$\mathbf{2.19 \times 10^{-4}}$	$\mathbf{2.86 \times 10^{-3}}$	9.66×10^{-4}	1.02×10^{-2}	$\mathbf{1.76 \times 10^{-4}}$
LSTM	$\mathbf{6.32 \times 10^{-3}}$	9.92×10^{-4}	9.91×10^{-3}	$\mathbf{1.41 \times 10^{-4}}$	$\mathbf{3.75 \times 10^{-3}}$	6.15×10^{-4}	1.06×10^{-2}	$\mathbf{1.70 \times 10^{-4}}$
HLTM	3.93×10^{-3}	6.94×10^{-18}			4.96×10^{-3}	8.69×10^{-4}		
TBATS	2.92×10^{-3}	5.79×10^{-5}			7.05×10^{-3}	4.32×10^{-4}		
ARIMA	5.42×10^{-3}	1.59×10^{-4}			2.03×10^{-2}	1.75×10^{-3}		
150 days								
LR	$\mathbf{5.49 \times 10^{-3}}$	2.68×10^{-4}	9.92×10^{-3}	$\mathbf{1.97 \times 10^{-4}}$	$\mathbf{5.02 \times 10^{-3}}$	6.39×10^{-4}	1.06×10^{-2}	$\mathbf{1.44 \times 10^{-4}}$
SVR	$\mathbf{6.81 \times 10^{-3}}$	1.20×10^{-3}	9.97×10^{-3}	$\mathbf{1.67 \times 10^{-4}}$	$\mathbf{5.72 \times 10^{-3}}$	5.65×10^{-4}	1.04×10^{-2}	$\mathbf{1.79 \times 10^{-4}}$
KNN	$\mathbf{5.04 \times 10^{-3}}$	2.06×10^{-4}	9.99×10^{-3}	$\mathbf{1.78 \times 10^{-4}}$	$\mathbf{5.66 \times 10^{-3}}$	6.71×10^{-4}	1.06×10^{-2}	$\mathbf{1.80 \times 10^{-4}}$
XGBoost	$\mathbf{7.75 \times 10^{-3}}$	$\mathbf{1.34 \times 10^{-3}}$	1.03×10^{-2}	1.81×10^{-4}	$\mathbf{5.66 \times 10^{-3}}$	4.83×10^{-4}	1.08×10^{-2}	$\mathbf{1.73 \times 10^{-4}}$
LSTM	$\mathbf{5.07 \times 10^{-3}}$	1.29×10^{-3}	1.01×10^{-2}	$\mathbf{1.56 \times 10^{-4}}$	$\mathbf{5.89 \times 10^{-3}}$	2.16×10^{-3}	1.04×10^{-2}	$\mathbf{1.11 \times 10^{-4}}$
HLTM	3.93×10^{-3}	6.94×10^{-18}			9.34×10^{-3}	8.19×10^{-4}		
TBATS	2.49×10^{-3}	2.15×10^{-4}			9.26×10^{-3}	6.46×10^{-4}		
ARIMA	4.67×10^{-3}	1.88×10^{-4}			3.07×10^{-2}	3.39×10^{-3}		

Values in bold represent the best results for each row

Table 13 Expected portfolio Sharpe Ratio summary statistics

	Out-of-sample				One-day-ahead			
	Without TA		With TA		Without TA		With TA	
30 Days	Mean	SD	Mean	SD	Mean	SD	Mean	SD
LR	1.88×10^{-2}	1.86×10^{-4}	2.78×10^{-2}	2.15×10^{-3}	3.12×10^{-2}	7.08×10^{-3}	3.21×10^{-2}	1.86×10^{-3}
SVR	2.55×10^{-2}	2.69×10^{-3}	3.10×10^{-2}	1.95×10^{-3}	3.43×10^{-2}	6.00×10^{-3}	3.28×10^{-2}	1.58×10^{-3}
KNN	2.49×10^{-2}	1.06×10^{-4}	3.15×10^{-2}	1.96×10^{-3}	3.29×10^{-2}	6.30×10^{-3}	3.45×10^{-2}	1.66×10^{-3}
XGBOOST	2.23×10^{-2}	9.52×10^{-3}	3.14×10^{-2}	1.99×10^{-3}	3.17×10^{-2}	4.36×10^{-3}	3.19×10^{-2}	1.91×10^{-3}
LSTM	2.29×10^{-2}	1.72×10^{-3}	3.08×10^{-2}	1.78×10^{-3}	3.42×10^{-2}	5.65×10^{-3}	3.40×10^{-2}	1.50×10^{-3}
HLTM	1.04×10^{-2}	3.60×10^{-6}			1.75×10^{-2}	3.05×10^{-3}		
TBATS	4.91×10^{-3}	1.12×10^{-3}			1.72×10^{-2}	4.72×10^{-3}		
ARIMA	1.05×10^{-2}	4.22×10^{-4}			1.35×10^{-2}	4.67×10^{-3}		
60 days								
LR	1.35×10^{-2}	3.37×10^{-3}	2.72×10^{-2}	2.09×10^{-3}	2.96×10^{-2}	3.66×10^{-3}	3.18×10^{-2}	1.41×10^{-3}
SVR	1.81×10^{-2}	5.12×10^{-3}	2.29×10^{-2}	2.30×10^{-3}	2.66×10^{-2}	3.68×10^{-3}	2.84×10^{-2}	2.13×10^{-3}
KNN	1.62×10^{-2}	5.44×10^{-4}	2.72×10^{-2}	1.64×10^{-3}	2.96×10^{-2}	4.07×10^{-3}	3.23×10^{-2}	1.67×10^{-3}
XGBOOST	2.01×10^{-2}	5.06×10^{-3}	3.17×10^{-2}	1.79×10^{-3}	3.23×10^{-2}	4.21×10^{-3}	3.34×10^{-2}	1.67×10^{-3}
LSTM	1.69×10^{-2}	4.21×10^{-4}	2.16×10^{-2}	2.43×10^{-3}	2.69×10^{-2}	4.96×10^{-3}	3.31×10^{-2}	1.74×10^{-3}
HLTM	5.11×10^{-3}	3.84×10^{-5}			9.99×10^{-3}	1.93×10^{-3}		
TBATS	3.54×10^{-3}	4.58×10^{-4}			3.67×10^{-3}	1.65×10^{-3}		
ARIMA	9.17×10^{-3}	8.21×10^{-4}			1.79×10^{-2}	1.54×10^{-3}		
90 days								
LR	1.31×10^{-2}	2.66×10^{-3}	2.23×10^{-2}	2.29×10^{-3}	2.64×10^{-2}	3.10×10^{-3}	2.53×10^{-2}	2.33×10^{-3}
SVR	1.74×10^{-2}	4.71×10^{-3}	2.40×10^{-2}	2.46×10^{-3}	2.80×10^{-2}	2.64×10^{-3}	3.20×10^{-2}	1.75×10^{-3}
KNN	1.67×10^{-2}	1.17×10^{-3}	2.21×10^{-2}	2.48×10^{-3}	2.69×10^{-2}	4.65×10^{-3}	2.79×10^{-2}	2.01×10^{-3}
XGBOOST	1.57×10^{-2}	1.98×10^{-3}	2.56×10^{-2}	2.27×10^{-3}	2.63×10^{-2}	1.80×10^{-3}	2.77×10^{-2}	1.92×10^{-2}
LSTM	1.62×10^{-2}	5.58×10^{-3}	2.06×10^{-2}	2.82×10^{-3}	2.65×10^{-2}	2.40×10^{-3}	2.82×10^{-2}	1.98×10^{-3}

Table 13 (continued)

| | Out-of-sample | | | | One-day-ahead | | | |
| | Without TA | | With TA | | Without TA | | With TA | |
30 Days	Mean	SD	Mean	SD	Mean	SD	Mean	SD
HLTM	8.20×10^{-3}	3.81×10^{-3}			1.26×10^{-2}	2.05×10^{-3}		
TBATS	2.89×10^{-3}	2.54×10^{-4}			1.25×10^{-2}	1.85×10^{-3}		
ARIMA	4.91×10^{-3}	7.47×10^{-4}			1.36×10^{-2}	1.12×10^{-3}		
120 days								
LR	1.26×10^{-2}	3.75×10^{-3}	$\mathbf{1.97 \times 10^{-2}}$	$\mathbf{2.29 \times 10^{-3}}$	$\mathbf{2.36 \times 10^{-2}}$	$\mathbf{1.95 \times 10^{-3}}$	2.47×10^{-2}	2.31×10^{-3}
SVR	1.60×10^{-2}	5.80×10^{-3}	$\mathbf{1.89 \times 10^{-2}}$	$\mathbf{2.46 \times 10^{-3}}$	2.07×10^{-2}	2.45×10^{-3}	$\mathbf{2.17 \times 10^{-2}}$	$\mathbf{2.01 \times 10^{-3}}$
KNN	1.55×10^{-2}	$\mathbf{1.12 \times 10^{-3}}$	1.79×10^{-2}	2.48×10^{-3}	2.02×10^{-2}	3.86×10^{-3}	2.74×10^{-2}	$\mathbf{1.73 \times 10^{-3}}$
XGBOOST	1.49×10^{-2}	$\mathbf{1.88 \times 10^{-3}}$	1.90×10^{-2}	2.27×10^{-3}	2.04×10^{-2}	2.67×10^{-3}	2.39×10^{-2}	$\mathbf{1.95 \times 10^{-3}}$
LSTM	1.67×10^{-2}	$\mathbf{2.40 \times 10^{-3}}$	1.98×10^{-2}	2.82×10^{-3}	2.31×10^{-2}	$\mathbf{2.01 \times 10^{-3}}$	2.52×10^{-2}	2.17×10^{-3}
HLTM	7.55×10^{-3}	1.42×10^{-17}			7.98×10^{-3}	1.26×10^{-3}		
TBATS	3.07×10^{-3}	2.14×10^{-4}			4.23×10^{-3}	1.50×10^{-3}		
ARIMA	4.10×10^{-3}	3.37×10^{-4}			5.90×10^{-3}	7.67×10^{-4}		
150 days								
LR	1.26×10^{-2}	$\mathbf{4.93 \times 10^{-4}}$	1.78×10^{-2}	2.39×10^{-3}	2.12×10^{-2}	$\mathbf{1.39 \times 10^{-3}}$	2.80×10^{-2}	1.84×10^{-3}
SVR	1.38×10^{-2}	2.79×10^{-3}	2.22×10^{-2}	$\mathbf{2.09 \times 10^{-4}}$	2.30×10^{-2}	2.52×10^{-3}	2.38×10^{-2}	$\mathbf{2.22 \times 10^{-3}}$
KNN	1.28×10^{-2}	$\mathbf{3.84 \times 10^{-4}}$	2.05×10^{-2}	2.05×10^{-3}	2.32×10^{-2}	$\mathbf{1.65 \times 10^{-3}}$	2.71×10^{-2}	2.15×10^{-3}
XGBOOST	1.42×10^{-2}	$\mathbf{1.70 \times 10^{-3}}$	2.23×10^{-2}	2.36×10^{-3}	2.32×10^{-2}	$\mathbf{1.73 \times 10^{-3}}$	2.72×10^{-2}	2.33×10^{-3}
LSTM	1.40×10^{-2}	3.04×10^{-3}	2.25×10^{-2}	$\mathbf{1.92 \times 10^{-3}}$	2.32×10^{-2}	$\mathbf{1.63 \times 10^{-3}}$	2.58×10^{-2}	1.73×10^{-3}
HLTM	7.76×10^{-3}	1.42×10^{-17}			1.43×10^{-2}	7.98×10^{-4}		
TBATS	3.38×10^{-3}	1.97×10^{-4}			1.41×10^{-2}	8.73×10^{-4}		
ARIMA	3.88×10^{-3}	1.03×10^{-3}			9.32×10^{-3}	5.83×10^{-4}		

Values in bold represent the best results for each row

. In the case of a 30-day prediction period, the average expected risk appears to increase at an average rate of 532%, which drops to 219% for a 60-day period, to 145% for a 90-day period, increases to 218% for a 120-day period, and decreases to 90% for a 150-day period. Similarly to what is observed in the case of out-of-sample predictions, the average expected risk values tend to be lower for HLTM, TBATS, and ARIMA than for the algorithms that use TAIs for all periods.

We compared the expected risk value distributions using a KS test at the 5% significance level, similar to our comparison of the expected return distributions. Similarly to what we discussed for the expected return distributions, the null hypothesis was that the compared risk distributions came from the same continuous distribution. We again conducted five comparisons, one for each period, and thus, accounted for multiple comparisons by adjusting the alpha value to 0.01 using Bonferroni's correction. For the out-of-sample predictions, the KS test produced p-values of 7.17×10^{-28}, 1.32×10^{-38}, 7.19×10^{-43}, 3.97×10^{-43}, and 3.88×10^{-41} for 30-, 60-, 90-, 120-, and 150-day periods, respectively. These p-values were all below the adjusted significance level of 0.01, indicating that using TAIs resulted in a significant increase in the expected risk distributions. Similarly, for one-day-ahead predictions, the KS test produced p-values of 1.23×10^{-44}, 3.88×10^{-41}, 3.64×10^{-41}, 2.76×10^{-40}, and 3.88×10^{-41} respectively, also indicating a statistically significant difference in the expected risk distributions.

To conclude, the Genetic Algorithm results show that the introduction of the TAIs in the feature set has led to statistically significant increases in mean risk across all algorithms and periods. This is, of course, a non-favourable result, but we should keep in mind that risk is just one of the metrics used to evaluate a portfolio's performance. In fact, it should not be considered in isolation, but in conjunction with returns, which as we have already seen are significantly higher when using TAIs. It is thus important to study an aggregate metric, such as the Sharpe Ratio, which adjusts returns for the level of risk taken. By incorporating the standard deviation of returns, it provides a measure of how much return an investment generates per unit of risk. This enables a fair comparison of different investments or portfolios, considering their risk profiles. We present the Sharpe Ratio results next.

5.2.3 Expected portfolio sharpe ratios

In Table 13, we report the results for the expected portfolio Sharpe Ratio distributions over 30-, 60-, 90-, 120-, and 150-day testing periods. We observe that the proposed algorithms tend to outperform the benchmarks for all periods in the case of out-of-sample predictions. For instance, the highest average Sharpe Ratio value is observed for KNN (3.15×10^{-2}) over a 30-day period, for XGBoost (3.17×10^{-2}) over a 60-day period, for XGBoost again (2.56×10^{-2}) over a 90-day period, for LSTM (1.98×10^{-2}) over a 120-day period, and for LSTM again (2.24×10^{-2}) over a 150-day period. The standard deviation values appear to be lower for algorithms that use TAIs in some cases. For instance, the volatility of Sharpe Ratio distributions decreases from 2.69×10^{-3} to 1.95×10^{-3} for SVR over a 30-day period. In the case of one-day-ahead predictions, we observe that the Sharpe Ratio values obtained from the proposed approaches tend to be closer on average compared to the benchmark approaches. For instance, the average Sharpe Ratio value tends to range between 2.10×10^{-2} and 2.30×10^{-2} for benchmarks, and between 2.38×10^{-2} and 2.80×10^{-2} for the proposed approaches.

Similarly to what we have done for the expected return and risk distributions, we conducted a KS test to compare the expected distributions of Sharpe Ratio values. Since we are making multiple comparisons, we again adjusted the significance level according to Bonferroni's correction. For out-of-sample predictions, the KS test generated p-values of 7.17×10^{-28}, 3.97×10^{-25}, 2.77×10^{-21}, 1.12×10^{-20}, and 3.96×10^{-16} for 30-, 60-, 90-, 120-, and 150-day periods, respectively. All of these p-values were below the adjusted significance level of 0.01, indicating that using TAIs resulted in a significant improvement in the expected Sharpe Ratio distributions. For one-day-ahead predictions, the KS test produced p-values of 3.59×10^{-1}, 3.64×10^{-1}, 8.24×10^{-1}, 1.83×10^{-1}, and 1.68×10^{-1} respectively; in this case, the p-values are above the adjusted significance level, indicating that there is no statistically significant difference in the Sharpe Ratio distributions.

In summary, the above results confirm that using TAIs in ML can lead to an improvement in the risk-adjusted portfolio performance, with room for improvement of up to 66.10% in the case of out-of-sample predictions, and up to 20.07% in the case of one-day-ahead predictions. Moreover, the use of TAIs as additional features for machine learning algorithms causes a statistically significant difference in the expected Sharpe Ratio distributions in the case of out-of-sample predictions, as shown by the KS test. Even though we did not observe a statistically significant difference in the Sharpe Ratio distributions when using TAIs in the case of one-day-ahead predictions, we observe improvements across all periods. These results confirm the importance of using TAIs as additional features.

5.3 Shapley values

In the previous section, we observed that incorporating TAIs as additional features in our regression problem can significantly reduce the error rate and improve portfolio performance. The experimental results described in the previous section outlined that the inclusion of TAIs in our regression problem led to a notable reduction in error rates and an improvement in portfolio performance. In this section, we will explore the relative importance of those features using the SHAP (Lundberg and Lee 2017) and SAGE (Covert et al. 2020) algorithms, which produce metrics describing different aspects of feature quality, and are thus widely used for model explainability in a variety of machine learning contexts (Sundararajan and Najmi 2020; Fryer et al. 2021).

Both SHAP and SAGE build on the concept of Shapley values (Roth 1988); in traditional co-operative game theory, Shapley values reflect a partitioning of the overall output of a group (or 'grand-coalition'), which expresses this output as the sum of the individual contributions of its members, obtained by quantifying the average marginal contribution of each member across all possible member combinations (i.e. 'sub-coalitions'). In the context of assessing feature quality in machine learning algorithms, a Shapley value treatment of an algorithm's features provides an assessment of how much each feature contributes to a measure of interest in relation to the model. However, calculating true marginal contributions for obtaining classical Shapley values can be a computationally prohibitive step, and therefore algorithms like SHAP and SAGE rely on computationally efficient variants, which involve approximating marginal contributions as deviations of conditional distributions from practical prior baselines.

In the literature, SHAP primarily tends to be used in 'explainability' contexts; given a prediction, it measures the extent to which each feature has contributed to the prediction.
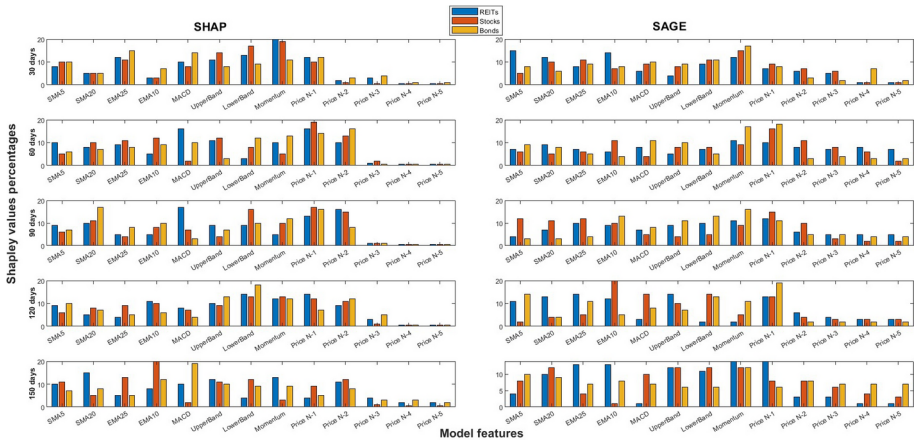
Fig. 3 Shapley average value for each asset class and feature classified by period considered

However, under the assumption that important features will be given larger weights in the final models following training, and that therefore the average influence of a feature over all predictions reflects its weighting in the model to a large extent, this can then be interpreted as a proxy measure for evaluating feature importance. Conversely, SAGE measures feature quality more directly; instead of making assumptions about the model's internals, it measures the influence of each feature on the evaluation metric directly.[8]

In order to have a clear view of the marginal contribution of each feature in each case, we present them here as percentages. To achieve this, we divided the average SHAP (or SAGE) value of each feature by the sum of SHAP (or SAGE) values for all features. Figure 3 presents the percentage SHAP (on the left side) and SAGE values (on the right side) calculated on the testing set for each feature, across all TAI-based algorithms using the out-of-sample method, displayed for each asset class and considered period. Regarding the SHAP values, we can observe that the relevance of prices lagged by two or more days tends to be lower compared to the other features. For REITs, TAIs combined account for 82% for a 30-day testing period, 72% for a 60-day period, 69% for a 90-day period, 73% for a 120-day period, and 77% for a 150-day period; while $N_1 + N_2$ account for a further 14% for a 30-day period; 26% for a 60-day period, 29% for a 90-day period, 23% for a 120-day period, and 15% for a 150-day period; and then the remaining lags only account for 4% for a 30-day period, 2% for a 60-day period, 2% for a 90-day period, 4% for a 120-day period, and 8% for a 150-day period. Similarly for stocks and bonds, TAIs account for 83% for a 30-day period, 66.5% for a 60-day period, 70% for a 90-day period, 75% for a 120-day period, 78% for a 150-day period; $N_1 + N_2$ account for 13% for a 30-day period, 31% for a 60-day period, 28% for a 90-day period, 21% for a 120-day period, 17% for a 150-day period; and the remaining lags only account for 3.75% for a 30-day period, 2.25% for a 60-day period, 2% for a 90-day period, 4% for a 120-day period, and 5% for a 150-day period.

Regarding the SAGE values, for REITs, we can observe that the combined contribution for TAIs tends to be 80% for a 30-day period, 60% for a 60-day period, 67% for a 90-day

---

[8] Note that, when relying on the RMSE for model evaluation, SAGE actually uses the negative RMSE internally instead, such that Shapley values denoting important features end up positive (with negative values denoting harmful features respectively)

period, 71% for a 120-day period, and 78% for a 150-day period; while the combined contribution for $N_1 + N_2$ is 13% for a 30-day period, 18% for a 60-day period, 18% for a 90-day period, 19% for a 120-day period, and 17% for a 150-day period; and the contribution of the remaining lags is 7% for a 30-day period, 22% for a 60-day period, 15% for a 90-day period, 10% for a 120-day period, and 5% for a 150-day period. Regarding stocks and bonds, the combined contribution for TAIs tends to be 77% for a 30-day period, 63% for a 60-day period, 69.5% for a 90-day period, 73.5% for a 120-day period, and 68% for a 150-day period; while the combined contribution for $N_1 + N_2$ is 13.5% for a 30-day period, 24% for a 60-day period, 20.5% for a 90-day period, 19% for a 120-day period, and 15% for a 150-day period; and the contribution of the remaining lags is 9.5% for a 30-day period, 13% for a 60-day period, 10% for a 90-day period, 7.5% for a 120-day period, and 17% for a 150-day period.

The combined SHAP and SAGE findings above may explain the substantial improvement in terms of RMSE, achieved by employing ML algorithms making use of TAIs in their feature-set (see Sect. 5.1). It is worth noting that, in the current literature, commonly employed approaches for financial forecasting currently tend to rely on lagged observations exclusively (Mehtab and Sen 2020; Sen and Mehtab 2021).

5.4 Computational times

The computational times for most of the algorithms are quite similar. On average, the execution time of HLTM, TBATS, and ARIMA was around 0.168 min, while LR, SVR, and KNN had slightly longer execution times, falling in the range of 0.2 to 0.3 min. LSTM had the longest computational time, averaging around 1.818 min. It is important to note that the difference in runtime, although varying across algorithms, is generally not considered significant. This is because these algorithms are typically run offline during the model-building phase, and only the resulting models are used in real-time applications where computational efficiency is crucial.

With regards to the Genetic Algorithm, we found that on average it was taking around 10.92 s per run. However, it is also worth noting that Genetic Algorithms are highly parallelisable, and thus their computational times can be further reduced through parallelisation processes (Brookhouse et al. 2014).

5.5 Discussion

Our experiments aimed to investigate the extent to which the inclusion of Technical Analysis Indicators (TAIs) as additional features could significantly reduce the error rate in predicting the time-series of REITs, stocks, and bonds, and improve the performance of a portfolio made of those asset classes. We outline below the main advantages observed from including TAIs, as per our study findings:

5.6 Reduction in error rate

The inclusion of TAIs caused a general reduction in the average RMSE for both out-of-sample and one-day-ahead predictions, particularly for REITs and stocks. On the other hand, there were less evident improvements in the average RMSE for bond time-series given the

already low RMSE values and low volatility of bond returns. The KS test results confirm that there is a statistically significant reduction in the RMSE for REITs and stocks following the inclusion of TAIs in the regression algorithms.

5.7 More consistent predictions

For bonds and stocks, we observed a lower standard deviation in the RMSE distributions across the holding periods considered when including TAIs. This indicates a higher concentration of RMSE values around their average value, suggesting more consistent predictions when TAIs are included.

5.8 Effects on portfolio performance

The second aim of our experiments was to demonstrate how the inclusion of TAIs affects the performance of a multi-asset portfolio. We observed significant improvements in the risk-adjusted performance of a portfolio made of REITs, stocks, and bonds across the considered holding periods, when optimising asset-allocation on the basis of price predictions obtained from TAI-driven ML models. According to our KS test results, there is a statistically significant difference in the portfolio performance when using out-of-sample predictions, while the difference is less significant (given a p-value lower than 0.05) for one-day-ahead predictions.

5.9 Trade-off between return and risk

The increase in the expected portfolio return caused by the use of TAIs is associated with an increase in the expected portfolio risk, thus creating a trade-off. However, we also observed an increase in the average expected Sharpe ratio, an aggregate metric, which is often considered more attractive than the isolated metrics of return or risk.

5.10 Feature importance analysis

We also analysed the importance of each feature with respect to the final prediction and its contribution to the overall RMSE for each asset class and prediction period. According to our findings, TAIs tend to show more relevance with respect to lagged prices as features, indicating that the information contained in the TAIs is critical in determining and predicting future prices.

In conclusion, our experiments demonstrated that the incorporation of TAIs into machine learning models for the prediction of REIT, stock, and bond time-series confers several benefits, including lower prediction error rate, increased consistency in prediction, and improved risk-adjusted portfolio performance. We also demonstrated that TAIs tend to contribute more highly to the reduction in RMSE compared to lagged prices.

6 Conclusion

This study focused on the task of predicting out-of-sample and one-day-ahead prices for REITs, stocks, and bonds. We employed five different machine learning algorithms in combination with Technical Analysis Indicators (TAIs) for five prediction periods (30-, 60-, 90-, 120-, and 150-day). Our experimental results indicate that: the use of TAIs generates a reduction in the average and volatility of RMSE distributions for the asset classes considered, which in turn leads to an improvement in the risk-adjusted performance of a portfolio made of those asset classes. Furthermore, we observed that TAIs tend to show greater relevance compared to the lagged prices, as demonstrated through the SHAP and SAGE average values.

The main limitation of the study relates to the unavoidably limited number of datasets, i.e. 10 companies for each market and each asset class. Other than repeating such analyses at a time where more data becomes available, other ways to address this limitation in future work could involve exploring further asset classes (e.g. risk-free securities or commodities) or other markets (e.g. Europe or emerging markets). Another limitation is the exclusive focus on TAIs as the additional features; future work could explore features beyond TAIs, such as financial statements, economic data, or industry trends, to assess whether these further improve the predictive performance of the models considered, and thus lead to improvements in risk-adjusted portfolio performance of multi-asset portfolios. Additionally, since our study only considers a limited number of algorithms, another area for future research could involve considering a larger number of algorithms for real estate price prediction. Finally, it would be useful to investigate longer prediction periods (for example, up to 2 or more years) to address the needs of an institutional investor aiming to hold the investment for the long-term.

References

Adegboye A, Kampouridis M, Otero F (2023) Algorithmic trading with directional changes. Artif Intell Rev 56(6):5619–5644

Agrawal M, Khan AU, Shukla PK (2019) Stock indices price prediction based on technical indicators using deep learning model. Int J Emerg Technol 10(2):186–194

Aguirre AAA, Medina RAR, Méndez NDD (2020) Machine learning applied in the stock market through the moving average convergence divergence (MACD) indicator. Invest Manage Financ Innov 17(4):44

Axelsson B, Song H-S (2023) Univariate forecasting for reit with deep learning: a comparative analysis with an arima model

Bollinger J (1992) Using bollinger bands. Stocks Commod 10(2):47–51

Block RL (2011) Investing in REITs: real estate investment trusts. Wiley, Hoboken

Bhuyan R, Kuhle J, Ikromov N, Chiemeke C (2014) Optimal portfolio allocation among REITs, stocks, and long-term bonds: an empirical analysis of us financial markets. J Math Financ. https://doi.org/10.4236/jmf.2014.42010

Brabazon A, Kampouridis M, O'Neill M (2020) Applications of genetic programming to finance and economics: past present future. Genet Program Evolvable Mach 21:33–53

Brookhouse J, Otero FE, Kampouridis M (2014) Working with OpenCL to speed up a genetic programming financial forecasting algorithm: initial results. In: Proceedings of the companion publication of the 2014 annual conference on genetic and evolutionary computation, pp. 1117–1124

Bulkowski TN (2012) Visual guide to chart patterns. Wiley, Hoboken

Chen W, Zhang H, Mehlawat MK, Jia L (2021) Mean-variance portfolio optimization using machine learning-based stock price prediction. Appl Soft Comput 100:106943

Christodoulaki E, Kampouridis M, Kyropoulou M (2023) Enhanced strongly typed genetic programming for algorithmic trading. In: Proceedings of the genetic and evolutionary computation conference (GECCO). ACM, Lisbon, Portugal

Covert I, Lundberg SM, Lee S-I (2020) Understanding global feature contributions with additive importance measures. Adv Neural Inf Process Syst 33:17212–17223

De Livera AM, Hyndman RJ, Snyder RD (2011) Forecasting time series with complex seasonal patterns using exponential smoothing. J Am Stat Assoc 106(496):1513–1527

Deb K (2011) Multi-objective optimisation using evolutionary algorithms: an introduction. Springer, New York

Delfim J-C, Hoesli M (2019) Real estate in mixed-asset portfolios for various investment horizons. J Portfolio Manage 45(7):141–158

Delfim J-C, Hoesli M (2020) Practical applications of real estate in mixed-asset portfolios for various investment horizons. Pract Appl 8(1):1–7

Dinesh S, Rao N, Anusha S, Samhitha R (2021) Prediction of trends in stock market using moving averages and machine learning. In: 2021 6th International conference for convergence in technology (I2CT), pp. 1–5. IEEE

Dutta P, Ramanathan R (2019) Forecasting electricity demand using arima models: a case study of the southern region of india. Energy Rep 5:1507–1515

Elton EJ, Gruber MJ, Brown SJ, Goetzmann WN (2009) Modern portfolio theory and investment analysis. Wiley, Hoboken

Fama EF, French KR (1992) The cross-section of expected stock returns. J Financ 47(2):427–465

Fryer D, Strümke I, Nguyen H (2021) Shapley values for feature selection: the good, the bad, and the axioms. IEEE Access 9:144352–144360

Gatzlaff DH, Sirmans GS (1991) Appreciation and depreciation of single-family homes. J Urban Econ 29(1):129–147

Gatzlaff DH, Geltner DM (1991) Portfolio diversification effects of reits. J Real Estate Financ Econ 4(2):157–173

Geiger P, Cajias M, Fuerst F (2016) A class of its own: the role of sustainable real estate in a multi-asset portfolio. J Sustain Real Estate 8(1):190–218

Geltner D, Miller NG, Clayton J, Eichholtz P (2016) Commercial real estate analysis and investments. Cengage Learning, Boston

Goldberg DE (1989) Genetic algorithms in search, optimization, and machine learning. addison. Reading

Habbab FZ, Kampouridis M (2022) Optimizing mixed-asset portfolios with real estate: why price predictions? In: 2022 IEEE World Congress on Computational Intelligence (WCCI), pp. 1–8. IEEE

Habbab FZ, Kampouridis M (2022) Machine learning for real estate time series prediction. In: 2022 UK Workshop on Computational Intelligence (UKCI). IEEE, Sheffield, UK

Habbab FZ, Kampouridis M (2023) Optimizing a prediction-based, mixed-asset portfolio including reits. In: 2023 IEEE Symposium Series on Computational Intelligence (SSCI). IEEE, pp. 1–4

Habbab FZ, Kampouridis M (2024) An in-depth investigation of five machine learning algorithms for optimizing mixed-asset portfolios including reits. Expert Syst Appl 235:121102

Habbab FZ, Kampouridis M, Voudouris AA (2022) Optimizing mixed-asset portfolios involving REITs. In: 2022 IEEE Symposium on Computational Intelligence for Financial Engineering and Economics (CIFEr), pp. 1–8. IEEE

Habbab FZ, Kampouridis M, Papastylianou T (2023) Improving REITs time series prediction using ML and technical analysis indicators. In: 2023 International Joint Conference on Neural Networks (IJCNN). IEEE, Gold Coast, Queensland, Australia

Hocht S, Ng KH, Wolf J, Zagst R (2008) Optimal portfolio allocation with asian hedge funds and asian REITs. Int J Serv Sci 1(1):36–68

Hung NH (2016) Various moving average convergence divergence trading strategies: a comparison. Investment management and financial innovations (13, Issue 2 (contin. 2)), 363–369

Hyndman RJ, Athanasopoulos G (2018) Forecasting: principles and practice. OTexts, Melbourne

Jalil RA, Ali HM, Razali N, Yim JLM (2015) Optimal portfolio allocation of malaysian real estate investment trusts during economic downturn. Int J Real Estate Stud 9(2):1–15

Jiang H, Wu C, Wu X (2019) Forecasting stock prices using ARIMA model and social media sentiment analysis. IEEE Access 7:107935–107944

Jones CA, Trevillion E (2022) Portfolio theory and property in a multi-asset portfolio. In: Real estate investment, pp. 129–155. Springer, NY, USA

Kavitha S, Varuna S, Ramya R (2016) A comparative analysis on linear regression and support vector regression. In: 2016 Online international conference on green engineering and technologies (IC-GET), pp. 1–5. IEEE

Khairi TW, Zaki RM, Mahmood WA (2019) Stock price prediction using technical, fundamental and news based approach. In: 2019 2Nd Scientific conference of computer sciences (SCCS), pp. 177–181 . IEEE

Kohli S, Godwin GT, Urolagin S (2020) Sales prediction using linear and knn regression. In: Advances in machine learning and computational intelligence: proceedings of ICMLCI 2019, pp. 321–329. Springer, New York, NY, USA

Lawal IA, Ibrahim RB, Chika U (2018) Seasonal arima model for weather variables forecasting in Nigeria. Int J Sci Technol Res 7(7):30–37

Lento C, Gradojevic N, Wright CS (2007) Investment information content in bollinger bands? Appl Financ Econ Lett 3(4):263–267

Levy RA (1966) Conceptual foundations of technical analysis. Financ Anal J 22(4):83–89

Li X, Liang J, Li Y, Liu X (2015) A survey of multi-objective portfolio optimization using evolutionary algorithms. J Comput Sci 9:135–146

Li J, Chen M, Li H, Tang Y (2022) Machine learning prediction for chinese reits market. Available at SSRN 4219420

Liow KH (2016) Handbook of Asian finance: REITs, trading, and fund performance. Academic Press, San Diego

Liu H, Liu D, Xin J (2002) Real-time recognition of road traffic sign in motion image based on genetic algorithm. In: Proceedings. International conference on machine learning and cybernetics, vol. 1, pp. 83–86 . IEEE

Long X, Kampouridis M, Kanellopoulos P (2023) Multi-objective optimisation and genetic programming for trading by combining directional changes and technical indicators. In: Proceedings of the IEEE congress on evolutionary computation (CEC). IEEE, Chicago, USA

Lundberg SM, Lee S-I (2017) A unified approach to interpreting model predictions. In: Guyon I, Luxburg UV, Bengio S, Wallach H, Fergus R, Vishwanathan S, Garnett R (eds) Advances in neural information processing systems. Curran Associates Inc, Red Hook, pp 4765–4774

Ma Y, Han R, Wang W (2020) Prediction-based portfolio optimization models using deep neural networks. Ieee Access 8:115393–115405

Ma Y, Han R, Wang W (2021) Portfolio optimization with return prediction using deep learning and machine learning. Expert Syst Appl 165:113973

Maricar MA (2019) Analisa perbandingan nilai akurasi moving average dan exponential smoothing untuk sistem peramalan pendapatan pada perusahaan xyz. Jurnal Sistem dan Informatika (JSI) 13(2):36–45

Markowitz H (1952) Portfolio selection. J Financ 7(1):77–91

Martins PNV (2017) Technical analysis in the foreign exchange market: the case of the macd (moving average convergence divergence) indicator. Master's thesis, School of Economics and Management. University of Porto

Mehtab S, Sen J (2020) Stock price prediction using cnn and lstm-based deep learning models. In: 2020 International conference on decision aid sciences and application (DASA), pp. 447–453. IEEE

Michaud RO (1989) The markowitz optimization enigma: is 'optimized' optimal? Financ Anal J 45(1):31–42

Miles D (2004) Real estate investment: a strategic asset allocation solution. J Portf Manag 30(3):119–129

Murphy JJ (1999) Technical analysis of the financial markets. New York Institute of Finance, New York

Mussumeci E, Coelho FC (2020) Machine-learning forecasting for dengue epidemics-comparing lstm, random forest and lasso regression. Medrxiv, 2020–01

Oncharoen P, Vateekul P (2018) Deep learning for stock market prediction using event embedding and technical indicators. In: 2018 5th International conference on advanced informatics: concept theory and applications (ICAICTA), pp. 19–24. IEEE

Oriani FB, Coelho GP (2016) Evaluating the impact of technical indicators on stock forecasting. In: 2016 IEEE Symposium Series on Computational Intelligence (SSCI), pp. 1–8. IEEE

Rehring C (2012) Real estate in a mixed-asset portfolio: the role of the investment horizon. Real Estate Econ 40(1):65–95

Rosillo R, Fuente D, Brugos JAL (2013) Technical analysis and the spanish stock exchange: testing the rsi, macd, momentum and stochastic rules using spanish market companies. Appl Econ 45(12):1541–1550

Roth AE (1988) The shapley value: essays in honor of lloyd shapley. Cambridge University Press, Cambridge

Santos AAP, Torrent HS (2022) Markowitz meets technical analysis: building optimal portfolios by exploiting information in trend-following signals. Financ Res Lett 49:103063. https://doi.org/10.1016/j.frl.2022.103063

Sen J, Mehtab S (2021) Accurate stock price forecasting using robust and optimized deep learning models. In: 2021 International conference on intelligent technologies (CONIT), pp. 1–9. IEEE

Sharifzadeh M, Sikinioti-Lock A, Shah N (2019) Machine-learning methods for integrated renewable power generation: a comparative study of artificial neural networks, support vector regression, and gaussian process regression. Renew Sustain Energy Rev 108:513–538

Sharpe WF (1964) Capital asset prices: a theory of market equilibrium under conditions of risk. J Financ 19(3):425–442

Shehadeh A, Alshboul O, Al Mamlook RE, Hamedat O (2021) Machine learning models for predicting the residual value of heavy construction equipment: an evaluation of modified decision tree, lightgbm, and xgboost regression. Autom Constr 129:103827

Stephen L, Simon S (2005) The case for REITs in the mixed-asset portfolio in the short and long run. J Real Estate Portf Manage 11(1):55–80

Sundararajan M, Najmi A (2020) The many shapley values for model explanation. In: International conference on machine learning, pp. 9269–9278. PMLR

Thakkar A, Chaudhari K (2021) A comprehensive survey on portfolio optimization, stock price and trend prediction using particle swarm optimization. Arch Comput Methods Eng 28:2133–2164

Vrieze SI (2012) Model selection and psychological theory: a discussion of the differences between the akaike information criterion (aic) and the bayesian information criterion (bic). Psychol Methods 17(2):228

Whitley D (1994) A genetic algorithm tutorial. Stat Comput 4:65–85

Yamaoka K, Nakagawa T, Uno T (1978) Application of akaike's information criterion (aic) in the evaluation of linear pharmacokinetic equations. J Pharmacokinet Biopharm 6(2):165–175

Zhu Y, Zhou G (2009) Technical analysis: An asset allocation perspective on the use of moving averages. J Financ Econ 92(3):519–544

Authors and Affiliations

Fatim Z. Habbab[1] · Michael Kampouridis[1] · Tasos Papastylianou[2]

✉ Fatim Z. Habbab
   fh20175@essex.ac.uk

   Michael Kampouridis
   mkampo@essex.ac.uk

   Tasos Papastylianou
   tasos.papastylianou@essex.ac.uk

[1] Centre for Computational Finance and Economics Agents, School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, UK

[2] Institute of Public Health and Wellbeing, University of Essex, Wivenhoe Park, Colchester, UK