

Strongly-typed genetic programming with an active trading operator for algorithmic trading using fundamental, sentiment, and technical analysis

Eva Christodoulaki^{a,*}, Michael Kampouridis^a

^a*School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester, CO4 3SQ, United Kingdom*

Abstract

Algorithmic trading has advanced significantly in recent years, with investors typically relying on either fundamental analysis (FA) or technical analysis (TA) indicators. Sentiment analysis (SA), however, has only gained substantial attention in the last decade. Consequently, algorithms have become essential tools in creating pre-programmed trading strategies, yielding notable financial gains. Despite the extensive individual study of these three types of analyses, their combined potential remains underexplored. Given the success of FA, TA, and SA in identifying profitable trading strategies, this paper aims to examine whether merging these indicators can further enhance profitability. To this end, we introduce a novel strongly-typed Genetic Programming (GP) algorithm within a soft computing framework, designed to enhance robustness in trading strategy development. Each branch in the algorithm's tree structure is dedicated to one analysis type, meaning that

*Corresponding author.

Email addresses: christodoulakis.eva@gmail.com (Eva Christodoulaki),
mkampo@essex.ac.uk (Michael Kampouridis)

one branch includes only fundamental indicators, another solely sentiment indicators, and the final branch exclusively technical indicators. This structured approach allows for better exploration and exploitation of the indicator search space. Additionally, the algorithm incorporates a unique fitness function, which evaluates not just overall trading performance but also the contributions of the individual FA, TA, and SA branches. Furthermore, we propose an innovative GP operator that promotes active trading by injecting trees into the GP population capable of executing numerous trades while maintaining high profitability and low risk. The algorithm’s effectiveness is tested using the stocks of 42 international companies, where it is compared against eleven alternative algorithms across three key financial metrics: Sharpe ratio, rate of return, and risk. The findings show that the proposed GP algorithm, leveraging evolutionary computing principles, demonstrates practical financial benefits. The algorithm enhances the profitability and risk-adjusted performance of trading strategies, achieving statistically significant improvements over competing models and active trading potential. This work contributes to the advancement of decision support systems and financial analysis applications.

Keywords: Fundamental Analysis, Sentiment Analysis, Technical Analysis, Genetic Programming, Algorithmic Trading

1. Introduction

The use of algorithmic trading has become increasingly widespread in the financial industry, leveraging pre-programmed strategies to achieve profit. As with many fields, algorithms play a pivotal role in finance, where researchers

5 apply Machine Learning (ML) techniques to analyse historical market data.
6 These techniques help detect patterns that can be used to generate signals
7 forecasting future trends, thereby optimising potential returns.

8 Algorithmic trading commonly relies on three main methods: fundamen-
9 tal analysis (FA), which evaluates a company’s financial and economic fun-
10 damentals; sentiment analysis (SA), which interprets and classifies the sen-
11 timents, attitudes, or opinions found in textual data; and technical analysis
12 (TA), which scrutinises price patterns and trends to discover trading opportu-
13 nities. While earlier studies primarily focused on fundamental and technical
14 analysis indicators, such as earnings per share (EPS), volatility, and moving
15 averages, sentiment analysis indicators like sentiment polarity have recently
16 gained prominence in the field.

17 Each of these methods has demonstrated the ability to produce profitable,
18 low-risk trading strategies, yet their combined use remains relatively rare,
19 largely due to the differing assumptions and methodologies underlying each
20 approach. Given the success of these individual methods, there is growing
21 interest in combining them to enhance trading strategy performance.

22 Building on this foundation, we propose a novel strongly-typed genetic
23 programming (GP) algorithm that, for the first time, fully integrates all
24 three types of analysis: FA, SA, and TA. Each GP tree is designed with
25 distinct branches dedicated to FA, SA, and TA indicators. This structural
26 constraint enhances the GP’s ability to explore and exploit the search space
27 of each type of indicator more effectively. Furthermore, type constraints are
28 applied during crossover and mutation operations to ensure that the offspring
29 generated remain valid.

30 Our fitness function incorporates a form of structural sampling by eval-
31 uating not only the overall performance of each evolved tree but also the
32 performance of its constituent FA, SA, and TA branches. This design en-
33 courages the discovery of models that perform robustly across different mar-
34 ket signal types. Conceptually, this aligns with sampling-based evaluation
35 strategies commonly used in machine learning, where model robustness is
36 assessed across multiple data partitions or objective components to avoid
37 overfitting and improve generalisation. By integrating performance feedback
38 from multiple subtrees into a unified fitness score, our approach effectively
39 “samples” the quality of different decision paths within the model, guiding
40 the evolutionary process toward more stable and diverse solutions.

41 Lastly, we aim to encourage active trading strategies that regularly assess
42 market conditions and capitalise on short-term opportunities. Such strategies
43 benefit from the ability to respond to price movements within shorter time-
44 frames, potentially enhancing profitability. To facilitate this, we introduce
45 a new GP operator, which complements traditional crossover and mutation.
46 This operator selects FA, SA, and TA subtrees that are executing highly ac-
47 tive, profitable, low-risk trading strategies and combines them into new trees
48 for the subsequent generation.

49 The goal of this study is to demonstrate the proposed GP algorithm’s
50 ability to create innovative and profitable trading strategies by integrating
51 insights from fundamental, sentiment, and technical analysis. We evalu-
52 ate its performance against a set of eleven benchmarks, which include sev-
53 eral GP variants, a financial benchmark, and four algorithmic benchmarks:
54 multi-layer perceptron, support vector machine, extreme gradient boosting

55 (XGBoost), and long short-term memory (LSTM) neural network. The ex-
56 periments are conducted using stock data from 42 international companies,
57 with performance assessed across three key financial metrics: Sharpe ratio,
58 rate of return, and risk. This way, we validate the practical utility of the
59 proposed GP algorithm and demonstrate the algorithm’s adaptability and
60 practical impact.

61 In summary, besides continuing to analyse our work in detail, we ex-
62 tend our research compared to previous papers in the following ways: (i)
63 we present a novel strongly-typed genetic programming (GP) algorithm that
64 integrates three analysis types, instead of two as in the past (ii) the fitness
65 function of the proposed GP model considers the overall performance of the
66 tree and the individual contributions of the three branches, (iii) we intro-
67 duce an active trading operator that combines parts of the most profitable
68 and low-risk trees, and (iv) we compare our algorithm with a total of 11
69 benchmarks.

70 The paper is structured as follows: Section 2 provides a review of past
71 research on the application of fundamental analysis (FA), sentiment analysis
72 (SA), and technical analysis (TA) indicators in algorithmic trading. Section
73 3 outlines the methodology used in this study, starting with an explanation of
74 the features (indicators) employed, followed by a detailed description of the
75 proposed GP algorithm. The experimental setup, which includes the dataset,
76 performance metrics, and benchmarks, is discussed in Section 4. The results
77 and analysis are presented in Section 5, where we compare the performance
78 of the GP algorithm and assess the effectiveness of combining FA, SA, and
79 TA indicators. Finally, Section 6 summarises the key findings and suggests

80 potential directions for future research.

81 **2. Literature Review**

82 This section reviews prior research on financial forecasting and algorithmic trading, with a specific focus on studies that utilise indicators from fundamental, technical, and/or sentiment analysis.

85 Fundamental analysis (FA) is a method used to assess and evaluate securities—such as stocks, bonds, or currencies—based on intrinsic factors that influence their value. This analysis considers both qualitative and quantitative factors related to the asset, including financial statements, industry trends, competitive positioning, management, economic indicators, and market conditions. The primary goal of FA is to estimate the intrinsic value¹ of an asset and determine whether it is overvalued or undervalued by the market. Investors commonly use FA to make long-term decisions and identify investment opportunities based on an asset’s fundamentals. Previous studies have integrated FA indicators with deep neural networks, as demonstrated by [1] and [2], and hybrid models combining linear regression and XGBoost have also been employed for financial forecasting, as seen in [3]. Further examples include works that focus on financial models and trading strategies, such as those by [4] and [5].

99 Market fluctuations often result from events and decisions influenced by
100 online information, causing stock prices to rise and fall. Consequently, there

¹The intrinsic value of a stock is an estimate of the stock’s true worth, calculated from fundamental factors such as earnings, assets, and growth prospects, independent of its current market price.

101 has been a growing interest in incorporating sentiment analysis (SA) into fi-
102 nancial forecasting and algorithmic trading. SA involves extracting sentiment
103 from online texts, such as articles or comments, and using this information
104 to improve the accuracy of stock predictions and enhance the profitability of
105 trading strategies. Several studies have explored SA’s role in this context.
106 For example, [6] applied neural networks, and [7] proposed an event-driven
107 stock model. [8] used support vector machines (SVM) with tree kernels and
108 semantic frame parsing to generalise from sentences to scenarios. Additional
109 research includes [9], which evaluated the impact of news quality on stock
110 movements, and [10], which analysed sentiments in tweets. Furthermore,
111 [11] examined 8-scale emotions. SA has also been used in various studies,
112 including [12], [13], [14], and [15], for financial forecasting and trading. A few
113 studies, such as [16], [17], and [18], have incorporated SA indicators into GP
114 algorithms for algorithmic trading, demonstrating that sentiment analysis
115 can lead to profitable outcomes.

116 Technical analysis (TA) is another widely used tool in financial forecast-
117 ing and algorithmic trading, focusing on analysing financial metrics to create
118 indicators that help identify market trends and trading opportunities. TA
119 leverages financial data, such as stock prices and trading volumes, to gain
120 insights into market dynamics and trader behaviour. By studying historical
121 price movements and other technical factors, traders can make informed buy
122 or sell decisions. For instance, [19] incorporated TA indicators with linear
123 models, while [20] used a long short-term memory (LSTM) model to fore-
124 cast stock prices. Moreover, a hybrid deep learning model integrating TA
125 for financial forecasting was proposed by [21], and [22] showed that GP algo-

126 rithms could surpass traditional TA methods. Similar results were reported
127 in studies such as [23, 24, 25].

128 Compared to the studies utilising the algorithms individually, research
129 combining two of the three analysis types has been relatively limited. Some
130 studies, such as [26] and [27], combined fundamental analysis (FA) with
131 technical analysis (TA), finding that this approach increased explanatory
132 power in trading strategies. Similarly, in the Forex market, [28] used fused the
133 two by first processing fundamental data with BERT1, combining them with
134 technical indicators via an autoencoder, and refining using BERT2. Other
135 studies, including [29], examined FA in combination with sentiment analysis
136 (SA), focusing in particular on sentiment extracted from 10-K and 10-Q
137 filings.² It is an essential document for investors making informed decisions.

138 Combining TA and SA indicators has received more attention in the liter-
139 ature. For example, [30] employed a hybrid model combining recurrent and
140 convolutional neural networks, while [31] used reinforcement learning with
141 stock price data and sentiment from news headlines. Additionally, [32] and
142 [33] applied genetic programming (GP) to combine TA and SA, achieving
143 strong results. However, few studies have explored the combination of all
144 three analysis types, with notable exceptions being [34], [35], and [36]. More
145 recent work further supports the value of multimodal integration, as [37] com-

²A 10-K filing is a comprehensive annual report submitted by publicly traded companies to the SEC, providing detailed information about their financial performance, operations, and risk factors. A 10-Q filing is the corresponding quarterly report, submitted for each of the three fiscal quarters not covered by the 10-K, and provides an unaudited but otherwise comparable update of financial performance for that quarter.

146 bined LSTM models with social media sentiment and technical indicators,
147 showing improved predictive accuracy, while [38] used genetic algorithms to
148 select sentiment and technical features for an LSTM forecaster. Additional
149 studies, such as [39] and [40], demonstrate that deep-learning approaches can
150 effectively exploit multiple indicator types to enhance trading performance.

151 In summary, while fundamental, sentiment, and technical analyses have
152 all been used in algorithmic trading, studies combining these approaches
153 remain limited. Notably, our prior work [32, 33, 41] has shown that combin-
154 ing SA and TA using GP outperforms using either type individually, while
155 [42] and [43], have combined the three analysis types successfully. The for-
156 mer using non strongly typed GP and the latter using indicators from the
157 three analysis types in the Forex market. Similarly, [44, 45] supported mul-
158 timodal integration in financial prediction. The former demonstrates that
159 fusing heterogeneous data sources (e.g., technical indicators, textual senti-
160 ment, and fundamental signals) yields superior accuracy compared to uni-
161 modal statistical and deep-learning baselines, while the latter shows that
162 incorporating fundamental, technical, and sentiment variables within a uni-
163 fied neural-network framework improves forecasting performance compared
164 to using any single analysis type alone.

165 In this paper, we introduce a novel GP algorithm that incorporates fun-
166 damental analysis (FA) in addition to SA and TA, aiming to further enhance
167 the profitability and risk management of trading strategies.

168 3. Methodology

169 The research methodology is divided into two main sections. Section 3.1
170 outlines the three types of analysis along with the indicators selected for
171 each. Following this, Section 3.2 details the GP methodology, highlighting
172 the strongly-typed framework of the algorithm, as well as the introduction
173 of a new fitness function and operator.

174 3.1. Financial analysis processes

175 In this section, we examine fundamental, sentiment, and technical anal-
176 ysis as distinct processes, with each one discussed in its own subsection.

177 3.1.1. Fundamental analysis

178 In our research, we use twelve widely recognised and frequently applied in-
179 dicators for fundamental analysis (FA) [46], which include: Net Profit Ratio,
180 Return on Equity, Quick Ratio, Debt to Equity Ratio, Price-Earnings Ratio,
181 Price to Book Ratio, Price-Sales Ratio, Total Revenues, Levered Free Cash
182 Flow, Diluted Earnings Per Share (EPS), Earnings Before Interest, Taxes,
183 Depreciation, and Amortization (EBITDA), and Research and Development
184 (R&D) expenses. A complete summary of these indicators can be found in
185 Table 1.

186 The data for this study was sourced from 10-K and 10-Q filings, accessed
187 via the financial content platform "Seeking Alpha," which links to each com-
188 pany's primary filing documents. Fundamental indicators were extracted
189 directly from these primary filing documents, so that the values used reflect

Table 1 Fundamental Analysis indicators

Fundamental analysis indicators	
Indicators from 10-K/10-Q filings	Net Profit ratio Return on Equity, Quick ratio, Price-Earnings ratio, Price to Book ratio, Price-Sales ratio, Debt to Equity, Total Revenues, Levered Free Cash Flow, Diluted EPS, EBITDA, R&D Expenses

the figures as originally reported by the company.³ Of the twelve indicators previously mentioned, Diluted EPS, Total Revenues, EBITDA, Levered Free Cash Flow, and R&D Expenses are directly available from the 10-K and 10-Q reports. The remaining seven indicators were calculated using the following equations, where y refers to the financial year, $price$ denotes the stock value, and j represents a specific day within the five-year period.

The Net Profit ratio (NetProf) is a profitability metric calculated by dividing the Net Income by the Revenue.

$$\text{NetProf}(y) = \frac{\text{NetIncome}_y}{\text{Revenue}_y} \quad (1)$$

Return on Equity (RoE) is a profitability metric that measures the return

³<https://seekingalpha.com>. Last accessed: August 2023.

199 generated on shareholders' investments. It is usually calculated by dividing
200 Net Income by Shareholders' Equity.

$$\text{RoE}(y) = \frac{\text{NetIncome}_y}{\text{ShareholderEq}_y} \quad (2)$$

201 The Quick Ratio (QuickR) is a liquidity metric used to evaluate a com-
202 pany's ability to cover its short-term liabilities with its most liquid assets. It
203 is determined by subtracting Inventory from Current Assets and then divid-
204 ing the result by Current Liabilities.

$$\text{QuickR}(y) = \frac{\text{CurrentAssets}_y - \text{Inventory}_y}{\text{CurrentLiabilities}_y} \quad (3)$$

205 The Debt to Equity ratio (DebtEq) measures the proportion of a com-
206 pany's assets that are financed through debt versus shareholders' equity. It
207 is calculated by dividing Total Debt by Shareholders' Equity.

$$\text{DebtEq}(y) = \frac{\text{TotalDebt}_y}{\text{ShareholderEq}_y} \quad (4)$$

208 The Price-Earnings ratio (P/E) is a valuation metric used to assess whether
209 a company's stock is overvalued or undervalued. It is calculated by dividing
210 the Share Price (p) by the Earnings per Share (EPS).

$$\text{P/E}(j) = \frac{\text{price}_j}{\text{EPS}_y} \quad (5)$$

211 The Price to Book ratio (P/B) is a valuation metric used to assess po-
212 tential investment opportunities. It is calculated by dividing the Share Price
213 (p) by the Book Value per Share (BVPS).

$$\text{P/B}(j) = \frac{\text{price}_j}{\text{BVPS}_y} \quad (6)$$

214 The Price-Sales ratio (P/S) is a valuation metric that reflects how much
 215 the financial markets value each dollar of a company’s revenue. It is calcu-
 216 lated by dividing the Share Price (p) by the Revenue per Share.

$$P/S(j) = \frac{price_j}{Revenue_y} \quad (7)$$

217 For each company, the accounting variables extracted from a given filing
 218 are treated as becoming available from the first trading day after the public
 219 release of that filing (based on the filing’s SEC filing date) and are held
 220 constant until the next filing. This ensures that the GP strategies never
 221 use financial statement information that would not yet have been observable
 222 on the decision date, while remaining consistent with our daily, non-event-
 223 driven design. All FA indicators were normalised between $[-1, 1]$.

224 3.1.2. *Sentiment analysis*

225 Before implementing the sentiment analysis indicators, it was essential
 226 to gather the relevant data. In this study, we used the GoogleNews Python
 227 library, an interface to Google’s News search results, together with newspa-
 228 per3k for downloading and parsing full article text, to collect articles related
 229 to the selected companies. For each company and each day in the study
 230 period, we queried the library using the company’s name as the keyword, re-
 231 stricted to that date, and retrieved the full paginated output returned by the

232 library, concatenating results across pages before filtering.⁴ These articles
233 were collected for the same time period as the technical analysis indicators.
234 Additionally, we obtained each article’s title and summary, which enabled
235 the creation of polarity, subjectivity, and sentiment indicators for both the
236 full text and its title and summary. The choice of using Google Search was
237 driven by their broad coverage and ease of access for multiple companies
238 across time. Alternative data sources were considered during preliminary
239 stages, including Twitter. However, Twitter data was found to be excessively
240 noisy for some companies included in this study, like Apple, particularly due
241 to unrelated mentions and advertisements related to investment tools. Ad-
242 ditionally, for several companies in our sample, the volume of Twitter data
243 was insufficient to construct a reliable and consistent sentiment time series.
244 For these reasons, we opted to rely on structured web-based article data,
245 while acknowledging that future research may benefit from combining mul-
246 tiple sentiment sources, such as financial APIs, curated news feeds, or social
247 media, to enhance coverage and robustness.

248 To reduce noise and ensure the relevance of the sentiment data, we ap-
249 plied strict filtering rules to the collected articles. Specifically, only articles
250 with a minimum length of 500 characters were retained, which helped exclude
251 brief or contentless entries such as headlines without context. In addition,

⁴This retrieval reflects Google’s own ranking of News search results for the given key-
word and date. We do not apply any further selection by rank or sentiment beyond the
length and relevance filters described above (minimum 500 characters, company name and
ticker both present in the text). All qualifying articles within the retrieved pages are
included in the daily aggregation.

252 each article was required to contain both the company’s full name and its
253 stock ticker symbol, ensuring the content was specifically related to the tar-
254 get company and reducing the risk of irrelevant or misattributed sentiment.
255 To properly align sentiment with market data, we synchronised each article’s
256 publication date with stock prices. For each trading day, the sentiment feature
257 is computed as the average sentiment across all calendar days since the pre-
258 vious trading day’s close, up to and including the current day. On ordinary
259 weekdays this reduces to the sentiment of that single day. For a trading day
260 immediately following a weekend (typically Monday), the feature aggregates
261 sentiment from the preceding Friday together with the intervening Saturday
262 and Sunday, since all of this information accumulates before the market re-
263 opens and only becomes actionable at that next session. This roll forward
264 convention ensures that no sentiment feature is ever constructed from infor-
265 mation dated later than the decision it informs, as seen in [47, 48]

266 This weekend-mapping convention is standard in daily-horizon sentiment
267 studies and avoids assigning information to future dates, thereby preventing
268 look-ahead bias in feature construction.

269 When multiple articles were published on the same day for a company,
270 we averaged the sentiment values of the relevant articles. For days with
271 no published articles, a neutral sentiment value of zero (0) was assigned,
272 ensuring the continuity of our data.

273 Next, we present the sentiment analysis indicators. Two widely used indi-
274 cators are sentiment polarity and subjectivity. Sentiment polarity indicates
275 the overall sentiment direction, categorising the text as positive, negative,
276 or neutral. Subjectivity, meanwhile, measures how much of the text re-

277 flects personal opinion rather than objective information. In our analysis, we
278 utilise indicators based on these concepts, with different calculation methods
279 explained below.

280 *TextBlob* is a Python library offering a simple API for sentiment polarity
281 and subjectivity analysis. For each article, title, and summary, we extracted
282 polarity (ranging from -1 for negative to 1 for positive) and subjectivity
283 (ranging from 0 for completely objective to 1 for completely subjective). *Sen-*
284 *tiWordNet* 3.0 is a lexical resource built on the *WordNet* taxonomy, designed
285 for sentiment classification and opinion mining. In our implementation, we
286 first POS-tagged the text using the NLTK library and then computed senti-
287 ment scores by averaging the weighted positive and negative synsets. *AFINN*
288 *sentiment*, developed by Finn Årup Nielsen, is a sentiment lexicon that as-
289 signs integer polarity scores to over 3,300 English words. All methods were
290 implemented using open-source Python libraries to ensure transparency and
291 reproducibility.

292 To ensure a diverse representation of sentiment, we employed 12 sentiment
293 analysis (SA) indicators derived from three widely used sentiment analysis
294 tools: TextBlob, SentiWordNet, and AFINN. Each tool was applied to three
295 different textual components associated with each news article: the full article
296 text, the title, and the summary, all retrieved via the Google Search API.

- 297 • TextBlob provides two measures: polarity (ranging from negative to
298 positive sentiment) and subjectivity (ranging from objective to subjec-
299 tive content). Applied to the three textual components, this yields 6
300 indicators ($2 \text{ metrics} \times 3 \text{ components}$).
- 301 • SentiWordNet provides a single polarity score, applied to each of the

302 three components, resulting in 3 indicators.

- 303 • AFINN also produces a single polarity score for each component, giving
304 another 3 indicators.

305 Altogether, this results in 12 sentiment indicators. This design captures both
306 lexical and semantic sentiment features across multiple content layers (title,
307 summary, and full text), allowing the model to reflect sentiment as it may be
308 differently expressed or emphasised in different parts of a document. All SA
309 indicators were normalised between $[-1, 1]$. A summary of these indicators
310 is presented in Table 2.

Table 2 Sentiment Analysis Indicators. TEXT corresponds to a complete article, TITLE to the title of that article, and SUMM to the summary of that article, as provided by the Google Search results.

Analysis type	Indicator
Sentiment Analysis (TextBlob)	TEXTpol, TEXTsub TITLEpol, TITLESsub SUMMpol, SUMMsub
Sentiment Analysis (SentiWord-Net)	TEXTsenti, TITLESenti, SUMMsenti
Sentiment Analysis (AFINN)	TEXTafinn, TITLEafinn, SUM- Mafinn ERC

311 *3.1.3. Technical analysis*

312 In this study, we use six well-established technical analysis (TA) indica-
 313 tors: Moving Average, Momentum, Rate of Change (ROC), Williams %R,
 314 Midprice, and Volatility (Equations (8) - (13)). They are calculated using
 315 historical data from the selected companies, including the (adjusted) closing
 316 prices, as well as the daily highest and lowest prices, sourced from Yahoo!
 317 Finance⁵. Each indicator is evaluated using two different lookup windows of
 318 $n = 5$ and $n = 10$ days, producing a total of 12 TA indicators, summarised
 319 in Table 3. Further details are provided in Section 4.1.

Table 3 Technical Analysis indicators. Each indicator is considered for two different lookup windows (n).

lookup windows $n = 5$ and $n = 10$	
Technical analysis indi- cators	Moving Average
	Momentum
	ROC
	Williams %R
	Volatility
	Midprice

320 The Moving Average is a technical analysis tool used to smooth stock
 321 price data, helping to identify trends by filtering out short-term fluctuations

⁵<https://finance.yahoo.com>. Last accessed: August 2023

322 or noise. In this context, p_j denotes the adjusted closing price of the stock
 323 on the j -th day.

$$\text{Moving Average}(n, j) = \frac{\sum_{i=j-n}^j p_i}{n}, \text{ for } j \geq n. \quad (8)$$

324 The Momentum indicator measures the difference between the current
 325 adjusted closing price and the adjusted closing price from n days earlier, as
 326 shown below.

$$\text{Momentum}(n, j) = p_j - p_{j-n}, \quad (9)$$

327 while the Rate of Change (ROC) normalises the momentum.

$$\text{ROC}(n, j) = \left(\frac{p_j}{p_{j-n}} - 1 \right) \cdot 100 \quad (10)$$

328 Volatility is a statistical metric that quantifies the variation in returns
 329 over a given period. The relevant indicator is calculated as follows.

$$\text{Volatility}(n, j) = \sqrt{\text{Var} \left(\left\{ \frac{p_{j-i}}{p_{j-n}} - 1 \right\}_{i \in \{0, \dots, n-1\}} \right)}, \quad (11)$$

330 where Var defines the sample variance over a dataset.

331 The Williams %R indicator, defined in Equation (12), compares the most
 332 recent closing price cl_j (on day j) to the highest price $hh_{n,j}$ recorded within
 333 the lookup window ending on day j , while also taking into account the lowest
 334 price $ll_{n,j}$ observed within the same period.

$$\text{Williams \%R}(n, j) = -100 \cdot \frac{hh_{n,j} - cl_j}{hh_{n,j} - ll_{n,j}} \quad (12)$$

335 The *Midprice*, as defined in Equation (13), represents the midpoint be-
 336 tween the highest price, $hh_{n,j}$, and the lowest price, $ll_{n,j}$, over all days within
 337 the lookup window ending on day j .

$$\text{Midprice}(n, j) = \frac{hh_{n,j} + ll_{n,j}}{2} \quad (13)$$

338 All TA indicators were normalised between $[-1, 1]$.

339 *3.2. Proposed ActTrade GP algorithm*

340 GP algorithms have been very popular algorithms for many financial
341 tasks, including algorithmic trading [49]. In this work, we propose a novel
342 strongly-typed Genetic Programming algorithm that has three main novel
343 components: (i) a strongly-typed nature, (ii) a novel fitness function, and
344 (iii) a novel genetic operator.

345 For the rest of this section, we will first present the model representation
346 and the strongly-typed nature of the algorithm (Section 3.2.1), and then
347 explain how trading takes place (Section 3.2.2). Then, in Section 3.2.3 we
348 will discuss the novel fitness function of the proposed GP algorithm, as well
349 as the metrics used for the evaluation of the GP trees. Next, Section 3.2.4
350 will present the GP operators used in our experiments, along with our novel
351 operator that encourages active trading. Lastly, Section 3.2.5 will discuss the
352 selection process of individuals to go through the GP operators.

353 *3.2.1. Model representation*

354 The function nodes are based on a 3-input **AND** function with a boolean
355 output (we call it **3-AND**), as well as binary **AND**, **OR**, Greater than (**GT**) and
356 Less than (**LT**) functions, with different variants allowing for each indicator
357 type. In particular, our algorithm uses AND_{FA} , OR_{FA} , GT_{FA} , LT_{FA} function nodes
358 in the FA branch, AND_{SA} , OR_{SA} , GT_{SA} , LT_{SA} function nodes in the SA branch
359 and it uses AND_{TA} , OR_{TA} , GT_{TA} , LT_{TA} function nodes in the TA branch.

360 The **3-AND** function takes 3 children. The first is of type FA, the second of
361 type SA, and the third of type TA. Thus, the first branch can take any of the

362 relevant FA-type functions, i.e. AND_{FA} , OR_{FA} , GT_{FA} , or LT_{FA} . The same principle
363 applies to the second (SA) and third (TA) branches. All three branches of
364 the 3-AND function need to evaluate to TRUE in order for the function to
365 yield a TRUE outcome.

366 The function set used in our algorithm is summarised in Table 4. This
367 carefully designed function set ensures that the algorithm generates models
368 that fully utilise all indicator types (FA, SA and TA), while at the same
369 time enforcing type consistency. In doing so, we can prevent type errors and
370 enhance the exploration of the search space.

Table 4 Function set for the GP algorithm.

Explanation	Function nodes
Root node	3-AND
FA, SA and TA type for AND	AND_{FA} , AND_{SA} , AND_{TA}
FA, SA and TA type for OR	OR_{FA} , OR_{SA} , OR_{TA}
FA, SA and TA type for Greater Than	GT_{FA} , GT_{SA} , GT_{TA}
FA, SA and TA type for Less Than	LT_{FA} , LT_{SA} , LT_{TA}

371 Regarding the terminal sets, different sets are allowed at different branches
372 of the corresponding trees. In the FA branch (Table 1), the terminal set con-
373 sists of specific indicators that are permitted to be used within that branch.
374 Similarly, the SA branch (Table 2) and the TA branch (Table 3) have their
375 own terminal sets of allowed indicators. Additionally, all terminal sets in-

376 clude an Ephemeral Random Constant (ERC) variable, which serves as a
377 threshold value for the indicators and consists of random values between -1
378 and 1 .

379 Compared to a non-strongly typed GP, our proposed algorithm has the
380 advantage of fully utilising the search space of each individual indicator type.
381 The three branches corresponding to the FA, SA and TA indicators are con-
382 nected using the 3-AND function at the root of the tree. This integration cre-
383 ates a foundation for better exploration and exploitation of the search space.
384 As a result, the model can generate more diverse, effective, and adaptable
385 trading strategies.

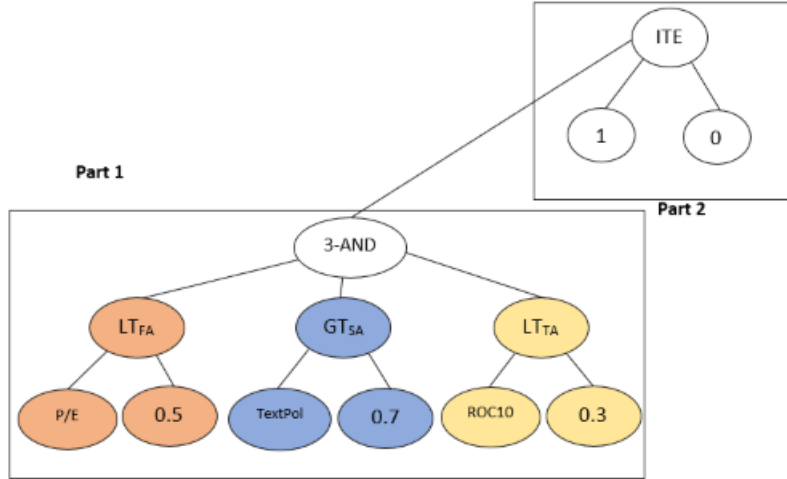
386 *3.2.2. Trading signals and trading strategy*

387 Trading signals are generated by the GP trees, where the root 3-AND
388 function, which outputs a TRUE/FALSE value, is used to decide whether to
389 issue a buy recommendation or hold the stock. This decision is implemented
390 using an If-Then-Else (ITE) node at the root of another tree structure. The
391 second branch represents the buy action (1), while the third branch signifies
392 the hold action (0), as shown in the example GP tree in Figure 1. It is
393 important to note that only Part 1 of Figure 1 undergoes evolution via GP
394 operations.

395 In Part 1 of Figure 1, the evolved tree begins with the root node 3-AND.
396 The fundamental analysis (FA) branch starts with the LT_{FA} node (depicted
397 by orange-coloured nodes), the sentiment analysis (SA) branch begins with
398 the GT_{SA} node (also in orange), and the technical analysis (TA) branch is
399 represented by the LT_{TA} node (shown in yellow).

400 If the GP tree (Part 1) returns a TRUE value, it activates the second

Figure 1 GP sample tree. The first child of the 3-AND function is enforced to be FA-related, the second child is SA-related and the third is enforced to be TA-related. This sample tree checks if the P/E indicator is less than the ERC of 0.5, and if the TEXTpol indicator is greater than the ERC 0.7, and if the ROC10 indicator is less than the ERC 0.3. If all three of them are TRUE, the recommendation will be to buy (1), otherwise it will be to hold (0).



401 branch (1) of the ITE tree, signaling a buy action. In this scenario, the GP
402 algorithm purchases the stock. To determine when to sell, the algorithm
403 follows this rule: given a predefined increase rate r and a time period of d
404 days, if the stock's return within the next d days exceeds r , the stock is sold
405 on that day. If the return does not surpass r within the d -day window, the
406 stock is sold at the end of that period. For instance, assuming $d = 30$ and
407 $r = 0.05$, the rule becomes: "If the stock price increases by more than 5%
408 of the purchase price within the next 30 days, sell the stock on that day.
409 Otherwise, sell it at the end of the 30-day period." On the other hand, if the

410 ITE tree signal is 0, indicating a hold decision, no trades are executed.

411 The parameters d and r are optimised during the validation phase and
412 remain constant across all GP algorithms considered, though they may differ
413 for different companies (see Section 4.3).

414 3.2.3. *Fitness function and Metrics*

415 Our analysis evaluates the metrics of return, risk, and the Sharpe ratio,
416 which are defined as follows.

417 The *return*, R , of a trade reflects the profit made as a percentage of the
418 invested amount. The profit calculation incorporates transaction costs of
419 0.025% of the selling price (c_t). Specifically, return is computed as detailed
420 in Equation (14), where V_f represents the final value, or the price at which
421 the stock was sold, and V_i represents the initial value, or the price at which
422 the stock was purchased

$$R = \frac{(1 - c_t)V_f - V_i}{V_i} \quad (14)$$

423 This simplified, uniform transaction cost model is commonly used in
424 daily-frequency trading studies to ensure comparability across strategies and
425 to avoid introducing additional assumptions about intraday liquidity. It cap-
426 tures the principal cost of executing a trade, while keeping the focus on the
427 relative performance differences between the proposed GP approach and the
428 ML baselines. Future work could include more sophisticated costs, such as
429 round-trip costs (buy and sell), bid-ask spreads, and slippage effects.

430 The *rate of return* (RoR) is the average return of all trades over a given
431 period. It is calculated by averaging the individual returns from each trade.

432 *Risk* is quantified as the standard deviation of the returns, expressed as
 433 $\sqrt{var[R]}$.

434 The *Sharpe ratio*, SR , is defined as the ratio of the expected excess return,
 435 relative to the risk-free rate, R_f , to the level of risk. Formally,

$$SR = \frac{E[R - R_f]}{\sqrt{var[R]}}, \quad (15)$$

436 where R_f represents the risk-free return, set at a value of 0.022%, an average
 437 figure reported by Fernandez et al. [50] and referenced by Long et al. [51].

438 The proposed *fitness function*, denoted as f_{sum} , evaluates both global and
 439 local performance within the GP tree structure. It is defined as the weighted
 440 sum of four components: the Sharpe ratio (SR^C) of the entire tree, which
 441 integrates FA, SA, and TA indicators, the Sharpe ratio (SR^{FA}) of the subtree
 442 containing only FA indicators, the Sharpe ratio (SR^{SA}) of the subtree with
 443 SA indicators, and the Sharpe ratio (SR^{TA}) of the subtree focusing on TA
 444 indicators. Each of these components is weighted by w_c , w_{FA} , w_{SA} , and
 445 w_{TA} , respectively, to balance the influence of the global and the branch
 446 performances. Formally,

$$f_{sum} = w_c \cdot SR^C + w_{FA} \cdot SR^{FA} + w_{SA} \cdot SR^{SA} + w_{TA} \cdot SR^{TA} \quad (16)$$

447 For instance, in Figure 1, SR^C represents the subtree with the root node
 448 3-AND (i.e., Part 1), while SR^{FA} refers to the subtree with the root node
 449 LT_{FA} (represented by orange-colored nodes), SR^{SA} corresponds to the subtree
 450 rooted at GT_{SA} (blue-colored nodes), and SR^{TA} refers to the subtree with the
 451 root node LT_{TA} (yellow-colored nodes).

452 The benefit of this fitness function is that it enables the evolution of trees
 453 that optimise all four Sharpe ratios, thereby allowing the GP to guide the
 454 search toward trading strategies with strong performance across all compo-
 455 nents. This is crucial because, if the fitness function were based solely on the
 456 Sharpe ratio of the entire tree (as is common in the literature), the GP could
 457 identify well-performing trees, but it might not fully leverage the strongly-
 458 typed structure that ensures the inclusion of FA, SA, and TA indicators.⁶⁷

459 3.2.4. GP operators

460 This section is divided into two parts. First, we introduce what we refer
 461 to as "traditional GP operators", including elitism, subtree crossover, and
 462 point mutation. Then, we present our proposed operator designed to promote

⁶In fact, early experiments we conducted confirmed this issue: when the fitness function was limited to the Sharpe ratio of the entire tree, the best-performing trees often had a large, well-performing subtree focused on just one analysis type, such as SA, while the FA and TA subtrees were small and underperforming. Moreover, the overall tree's performance did not surpass that of a non-strongly typed GP that allowed FA, SA, and TA indicators to be mixed, rendering the strongly-typed approach unnecessary. Our proposed fitness function resolves this issue.

⁷It is also worth noting that during the development of ActTrade, we experimented with incorporating parsimony pressure in the GP fitness function (specifically, lexicographic parsimony pressure to penalise larger trees). However, we observed that imposing a strong size penalty often led to a deterioration in trading performance: the GP would converge prematurely to overly simplistic models that under-utilized the rich set of indicators. As a result, we opted not to include an explicit parsimony term in the final runs. Instead, we controlled tree complexity by enforcing a maximum tree depth of 6 in our experiments (this constraint ensures trees do not grow unreasonably deep). More detail about the maximum tree depth and the other GP parameter values can be found in Section 4.

463 active trading.

464 *Traditional GP operators.* In elitism, the individual with the highest fitness
465 value in each generation is directly carried over to the next generation. For
466 subtree crossover, we apply it independently to each analysis type branch.
467 It is crucial to ensure that any operation is valid, meaning that it adheres to
468 the three distinct types (FA, SA, TA) within the GP trees. Consequently,
469 exchanged nodes must match in both type (e.g., terminal node with terminal
470 node) and data type (e.g., FA branch with FA branch) to preserve the integrity
471 of the tree. To ensure the validity of this exchange, we first swap the FA
472 branches between the two selected parents, followed by the SA branches, and
473 finally, the TA branches.

474 In a strongly-typed setting, point mutation also comes with specific con-
475 straints. For instance, a function node such as OR_{SA} can only be substituted
476 with AND_{SA} , and a function node like GT_{TA} can only be replaced by LT_{TA} (and
477 similarly for other function nodes). Additionally, an ERC can only be re-
478 placed with another ERC, and a terminal variable must be swapped with
479 another variable of the same indicator type. The algorithm ensures that mu-
480 tated nodes are replaced with valid data types. Point mutation occurs in one
481 of the three branches of the tree.

482 The parents selected for these operations are chosen through tournament
483 selection. An individual may undergo crossover with a probability of p and
484 mutation with a probability of $1 - p$.

485 *A novel GP operator for active trading.* As outlined in Section 1, one of
486 our key objectives is to develop an algorithm that implements highly active
487 trading strategies, which can lead to greater profit potential. To achieve

488 this, we concentrated on producing offspring that aim to maximise both the
489 number of trades and the Sharpe ratio.

490 To accomplish this, we follow the process outlined below: for each genera-
491 tion, we iterate through the population and separate the three subtrees (FA,
492 SA, TA) of each individual into three respective lists—one for each analysis
493 type. We then calculate the number of trades and the Sharpe ratio for each
494 subtree. After processing all trees in the current generation, we identify the
495 FA subtree with the highest number of trades and the one with the highest
496 Sharpe ratio; the same steps are repeated for the SA and TA subtrees. Addi-
497 tionally, to encourage simpler, more interpretable trees, if subtrees have the
498 same number of trades, we select the smaller one.

499 Next, we generate three offspring by combining subtrees that optimise
500 both the Sharpe ratio and the number of trades. Specifically, the first off-
501 spring is formed by combining the FA subtree with the highest Sharpe ratio,
502 the SA subtree with the most trades, and the TA subtree with the most
503 trades, placing them under the 3-AND function. This structure is then car-
504 ried over into the next generation. Similarly, the second offspring combines
505 the SA subtree with the highest Sharpe ratio and the FA and TA subtrees
506 with the most trades. Finally, the third offspring consists of the TA subtree
507 with the highest Sharpe ratio and the FA and SA subtrees with the most
508 trades.

509 It is important to note that this operator is applied at every generation
510 during training, meaning these three offspring are copied into the next gen-
511 eration before crossover and mutation take place. This process is outlined in
512 Algorithm 1.

513 By using this approach, the GP operator instills two key characteristics
514 in the population’s individuals: a high number of trades and a high Sharpe
515 ratio.⁸ This strategy enhances the quality of the evolving trees in subsequent
516 generations, favouring individuals that exhibit both traits to achieve more
517 effective and profitable trading outcomes.

⁸During early experimentation, we explored alternative strategies, such as constructing trees where all three subtrees were optimised either for maximising the number of trades or the Sharpe ratio. However, in those scenarios, only a small number of cases evaluated to TRUE, leading to very few trading actions due to the root function being 3-AND. This restrictive condition made the GP overly passive, often resulting in no trades. Other alternatives we considered were (i) using an OR function as the root, which didn’t ensure simultaneous consideration of all three indicator types during trading, and (ii) employing a non-strongly-typed GP approach. Neither of these alternatives produced strong financial results. Further investigation of these alternatives has been left for future work.

Algorithm 1 Active trading genetic operator procedure

Require: Population of individuals, Analysis types: FA, SA, TA

```
1: Initialise empty lists for FA subtrees, SA subtrees, and TA subtrees.
2:
3: for each individual in the population do
4:   Extract FA, SA, and TA subtrees from the individual.
5:   Append these subtrees to the respective lists for the analysis type.
6: end for
7:
8: for each subtree in the corresponding analysis type (FA, SA, TA) list do
9:   Calculate number of trades and Sharpe ratio.
10: end for
11:
12: for each list (FA subtrees, SA subtrees, TA subtrees) do
13:   Identify subtree with highest number of trades (max_trades_subtree) and subtree with highest
      Sharpe ratio (max_sharpe_subtree).
14: end for
15:
16: Create three offspring (named Offspring1, Offspring2, Offspring3):
17: for each offspring do
18:   Select corresponding subtrees:
19:   if Offspring1 then
20:     FA subtree: max_sharpe_subtree from FA subtrees
21:     SA subtree: max_trades_subtree from SA subtrees
22:     TA subtree: max_trades_subtree from TA subtrees
23:   else if Offspring2 then
24:     FA subtree: max_trades_subtree from FA subtrees
25:     SA subtree: max_sharpe_subtree from SA subtrees
26:     TA subtree: max_trades_subtree from TA subtrees
27:   else
28:     # Offspring3
29:     FA subtree: max_trades_subtree from FA subtrees
30:     SA subtree: max_trades_subtree from SA subtrees
31:     TA subtree: max_sharpe_subtree from TA subtrees
32:   end if
33:
34:   Combine the three subtrees (FA, SA, TA) under the ternary 3-AND function and enter them into
      the population of the next generation.
35: end for
```

518 3.2.5. Selection

519 In the previous section, we explained how our proposed genetic operator
520 creates three offspring, which are then placed into the population of the
521 next generation. We call them *Offspring1*, *Offspring2*, and *Offspring3*. As
522 these three individuals might not always make large impact in the search, we
523 decided to also probabilistically allow them to act as parents during crossover
524 and mutation. In this way, we allow their genetic material to be copied into
525 more trees in the population.

526 To achieve the above, we assign four probabilities p_1 , p_2 , p_3 , and p_4 ,⁹
527 to select the first parent for crossover, and the single parent for mutation.
528 More specifically, with probability p_1 , we use tournament selection, where k
529 individuals (trees) are randomly selected from the population and then the
530 individual with the highest fitness value in the tournament is selected as a
531 parent to undergo crossover/mutation. In addition, we assign a probability
532 p_2 , p_3 , and p_4 , to each one of Offspring1, Offspring2, and Offspring3, respec-
533 tively. In this way the proposed genetic operator can also affect the creation
534 of new offspring by carrying part of its genetic material to the new offspring.

535 For the second parent, we always use tournament selection i.e. $p_1 = 1$.
536 Algorithm 2 summarises the selection process.

537 To conclude, the proposed GP algorithm is a strongly typed GP that
538 aims to maximise the Sharpe ratio. The fitness function is determined by the
539 weighted sum of the FA, SA, TA subtree, and the complete tree comprising
540 of the three subtrees. The strongly typed structure of the GP enables more

⁹The sum of $p_1 + p_2 + p_3 + p_4$ is always equal to 1.

Algorithm 2 Parents selection procedure

Require: Probabilities p_1, p_2, p_3, p_4 , Offspring1, Offspring2, Offspring3

- 1: **if** Selecting first parent for crossover or parent for mutation **then**
 - 2: With probability p_1 : Perform tournament selection to select parent
 - 3: With probability p_2 : Select Offspring1 as parent
 - 4: With probability p_3 : Select Offspring2 as parent
 - 5: With probability p_4 : Select Offspring3 as parent
 - 6: **else**
 - 7: Perform tournament selection to select second parent for crossover
 - 8: **end if**
-

541 effective exploration and utilisation of all three indicator types, while the
542 fitness function emphasises the contribution of each indicator in enhancing
543 the trading performance of the algorithm. Lastly, the proposed operator
544 allows for offspring to be created that focus both on maximising the Sharpe
545 ratio, as well as maximising the number of trades. The relevant Python code
546 for these experiments is available on GitHub.¹⁰

547 4. Experimental Setup

548 4.1. Data

549 In this study, we gathered data from 42 international companies, from
550 different stock exchanges. The research period covered five years, from Jan-
551 uary 1st, 2015, to January 31st, 2020, intentionally excluding the COVID-19

¹⁰<https://github.com/avechris/ActTrade>

552 pandemic period. This exclusion was necessary to maintain consistency in
553 the training and validation sets, ensuring reliable parameter tuning.

554 The fundamental analysis data was sourced from the 10-K and 10-Q filings
555 available on the financial content platform "Seeking Alpha". For sentiment
556 analysis, we employed the GoogleNews Python library, together with news-
557 paper3k, to collect articles, titles, and summaries. The technical analysis
558 was performed using daily closing price data obtained from Yahoo! Finance.
559 After gathering all the data, we generated the 12 fundamental analysis (FA)
560 indicators, 12 sentiment analysis (SA) indicators, and 12 technical analysis
561 (TA) indicators as outlined in Section 3.1. It is important to note that our
562 study focuses exclusively on individual company stocks rather than index-
563 level assets such as the VIX or NASDAQ100, so we can isolate and evaluate
564 company specific signals derived from fundamental, sentiment, and technical
565 indicators. We therefore restrict our evaluation to 42 international compa-
566 nies across diverse geographies and industries, which provides heterogeneity
567 while keeping the analysis aligned with our objective of measuring the ben-
568 efits of integrating FA, SA, and TA at the stock level. The dataset was split
569 chronologically into training, validation, and testing sets, ensuring that all
570 training data precedes validation and all validation data precedes testing.
571 We split the datasets sequentially into three portions: 60% for training, 20%
572 for validation, and 20% for testing. This design prevents future information
573 leakage across phases, which is a critical issue in financial time series. It
574 provides a clear measure of its ability to generalise to new, out-of-sample
575 market data. More sophisticated validation schemes, such as walk-forward
576 or rolling-window evaluation, are not as feasible with daily data over a five-

577 year horizon per stock (roughly 1,200–1,300 trading days), as GP requires
578 reasonably long contiguous periods for evolution to converge.

579 It is important to note that our study is not a high-frequency trading
580 (HFT) framework. The algorithm is designed for daily decision making,
581 reflecting the resolution of the available data, where sentiment indicators are
582 aggregated daily and fundamental indicators are released quarterly. While
583 technical indicators can be computed intraday, excluding FA and SA at those
584 horizons would undermine the purpose of integrating the three analysis types.
585 Hence, our contributions are positioned within the context of daily trading
586 strategies, consistent with prior work in quantitative finance, as in [52] and
587 [53].

588 Lastly, it is worth noting that the datasets used in this work have varied
589 daily trading volume and return volatility. When looking at the one year test
590 set, we find a range of market conditions, where companies like Apple and
591 Tesla have very high trading volumes (above 10^8 shares on average), Amazon
592 and Walmart (above 10^7 shares on average), while firms such as Suzuki,
593 Fujitsu, and Kodak operate at volumes three to four orders of magnitude
594 smaller, evidencing low volume days. In parallel, daily return volatility varies
595 substantially across companies, with technology and automotive stocks in
596 particular showing elevated volatility levels compared to common consumer
597 companies. More specifically, we computed the standard deviation of daily
598 prices across the test set, which is a year. The results show companies in
599 all volatility regimes, for example Ford ($\sigma = 0.54$) and IBM ($\sigma = 4.05$) in
600 the low-volatility group (≤ 10), Apple ($\sigma = 10.3$) and Adidas ($\sigma = 30.7$) in
601 the moderate group (10–50), Amazon ($\sigma = 93$) in the high-volatility group

602 (50–100), and extremely volatile stocks such as Suzuki ($\sigma = 458.3$). This
 603 demonstrates that the one year test period includes both stable and highly
 604 volatile markets, thereby ensuring that the reported annualised performance
 605 metrics inherently account for different volatility regimes.

606 These descriptive statistics confirm that our evaluation period indeed
 607 spans both low volume and high volatility conditions, and that the reported
 608 performance metrics already integrate the model’s behaviour in such condi-
 609 tion, as we will highlight in more detail in Section 5.

610 *4.2. Benchmarks*

611 We refer to our proposed GP algorithm as ActTrade, named for its active
 612 trading operator. We compare its performance against six GP variants, four
 613 machine learning algorithms, and one financial-based trading strategy. These
 614 benchmarks are introduced in the following section.

- 615 • **GP-FA** is a GP algorithm that exclusively uses fundamental analysis
 616 indicators in its terminal set. The purpose of this benchmark is to assess
 617 the trading performance of a non-strongly-typed GP relying solely on
 618 fundamental analysis indicators.
- 619 • **GP-SA** is a GP algorithm that includes only sentiment analysis indica-
 620 tors in its terminal set. Similar to GP-FA, this benchmark is designed
 621 to evaluate the performance of a GP algorithm that operates exclusively
 622 with sentiment analysis indicators.
- 623 • **GP-TA** is a GP algorithm that solely utilises technical analysis indi-
 624 cators in its terminal set. The objective, as with GP-FA and GP-SA,

625 is to examine the trading performance of a GP that operates only with
626 technical analysis indicators.

627 • **GP-FASATA** is a (non-strongly-typed) GP algorithm that combines
628 fundamental, sentiment, and technical analysis indicators. The goal
629 of this benchmark is to assess how well the GP performs when it in-
630 tegrates all three types of indicators, without any typing constraints.
631 Since there are no defined branches, the GP is free to select any combi-
632 nation of indicators, regardless of whether they come from fundamental,
633 sentiment, or technical analysis.

634 • **STGP-FASATA** is a strongly-typed GP algorithm that incorporates
635 fundamental, sentiment, and technical analysis indicators. This al-
636 gorithm does not use the novel fitness function or the active trading
637 operator introduced in ActTrade, making it a valuable benchmark to
638 understand the impact of these innovations on the GP’s performance.

639 • **STGP-FASATA-S** is similar to STGP-FASATA but includes the same
640 fitness function used in ActTrade (the summation of Sharpe ratio fit-
641 ness functions). The only distinction between STGP-FASATA-S and
642 ActTrade is the absence of the active trading operator, serving as a
643 benchmark to highlight the added value of the active trading mecha-
644 nism.

645 Furthermore, the study also included two additional algorithmic bench-
646 marks, namely:

647 • **Multilayer perceptron (MLP)**

- 648 • **Support vector machine (SVM)**
- 649 • **eXtreme Gradient Boosting (XGBoost)**
- 650 • **Long short-term memory (LSTM)**

651 All the algorithms mentioned above were used to tackle a binary clas-
 652 sification problem: predicting whether a stock’s price will rise by a certain
 653 percentage ($r\%$) within the following d days. A Class 1 prediction signals a
 654 buy action, while a Class 0 prediction indicates a hold action. As discussed
 655 in Section 3.2.2, the sell action is executed as part of the trading strategy.
 656 and the same trading rules, transaction-cost assumptions, and (d,r) param-
 657 eters are applied to both GP-based and ML-based methods. In this sense,
 658 the ML algorithms serve as predictive reference models whose signals are fed
 659 into the identical downstream trading protocol as ActTrade, rather than as
 660 an alternative GP strategy with evolution mechanisms. Our aim is therefore
 661 to compare the quality of the induced trading performance under a common
 662 evaluation pipeline, not to claim that the GP and ML benchmarks operate
 663 in exactly the same way.

664 Furthermore, the proposed ActTrade is evaluated against two financial
 665 benchmarks:

- 666 • **Buy and Hold (BnH)**: The Buy and Hold strategy is a widely used
 667 investment approach and serves as a financial benchmark. It relies on
 668 historical price data, where one unit of stock is purchased at the start
 669 of the trading period and sold at the end. The strategy is based on the
 670 idea that, over time, the value of investments generally increases.

671 • **Trading-Strategy _{d,r} (TS _{d,r}):** In this strategy, we purchase stock at
672 the start of each trading period and sell it when the price increases by
673 more than a specified rate r , or after d days have elapsed.

674 The TS _{d,r} benchmark serves as a baseline trading strategy without a
675 learning component, as it follows the same trading rules employed by the GP
676 algorithms. This allows us to isolate and assess the added value of the GP
677 algorithm, independent of the underlying trading strategy.

678 4.3. Parameter Tuning

679 To mitigate the risk of overfitting and ensure the robustness of our results,
680 we employed a structured experimental design. Initially, we split the data
681 into three subsets: training, validation, and testing. The validation set was
682 utilised exclusively to tune and select appropriate parameters (such as GP
683 population size, crossover/mutation rates, and tree depth). After identifying
684 optimal parameters via the validation set, we combined the training and
685 validation sets to retrain the final GP model, maximising the available data
686 for model learning. Importantly, model performance was assessed on a test
687 set that remained entirely unseen during training and validation, providing a
688 clear measure of its ability to generalise to new, out-of-sample market data.

689 Table 5 provides the GP parameters used in the experiments conducted
690 for this study. A two-step grid search was performed on the validation set
691 to identify the optimal GP parameters. In the first step, key parameters
692 such as population size, crossover probability (p_c)¹¹, number of generations,

¹¹Since the mutation probability equals $1 - p_c$, it was excluded from the parameter tuning process.

693 tournament size, and maximum tree depth were tuned. Additionally, Ta-
694 ble 5 lists the probabilities p_1 , p_2 , p_3 , and p_4 for selecting the first parent
695 during crossover or mutation (p_1 : tournament selection; p_2 , p_3 , and p_4 are
696 for the three offspring generated by the active trading operator). The trad-
697 ing parameters d and r were fixed at 30 and 0.05, respectively, to expedite
698 the tuning process and ensure a fair comparison by identifying parameter
699 combinations that performed well across all GP variants.

Table 5 GP Parameters for the eight GP algorithms.

GP Parameters	
Population size	1000
Crossover probability	0.95
Mutation probability	0.05
Generations	50
Tournament size	4
Maximum tree depth	6
Tournament probability (p_1)	0.4
Active trading offspring probability (p_2 , p_3 , p_4)	(0.2, 0.2, 0.2)

700 In the second phase of parameter tuning, we concentrated on optimising
701 the trading strategy parameters, d and r . This tuning was conducted in-
702 dividually for each company across the algorithms, enabling improved trad-
703 ing performance by tailoring the parameters to the specific characteristics
704 of each company. The optimal values for the two trading parameters were

determined by selecting the combination that produced the best overall performance across all GP algorithms on the validation set.

The parameter tuning for MLP, SVM, XGBoost, and LSTM was performed separately using a binary classification approach, where one class represents a certain percentage increase in the stock price for the following day, and the other class represents a different price change (see Section 4.2). The goal was to identify the optimal parameter settings that maximise the models’ predictive accuracy on the validation set. After tuning, the model with the best predictive performance on the validation set was selected. The predicted class from this model was then used as a trading signal for the testing dataset, which was applied to the trading strategy. The trading parameters, d (days) and r (percentage increase), were set to the same values used in the GP variants to ensure consistency in evaluating the performance of the different algorithms.

After the parameter-tuning stage, we run 50 independent GP runs for each company, using the fixed parameter values identified during tuning and are trained only on the training and validation sets. After evolution, all strategies are evaluated on the unseen test set, and all performance statistics reported in Section 5 are computed across these these 50 evaluations. This procedure ensures that the reported results do not rely on a single run and that model selection remains robust.

5. Results and Analysis

In this section, we present the results of our experiments, comparing the proposed ActTrade algorithm with the benchmarks outlined in Section 4.2.

729 It is important to note that our analysis is conducted per company, since
730 strategies are evolved and evaluated individually for each stock rather than
731 at the portfolio level. For each GP algorithm, we conducted 50 independent
732 runs across the 42 companies. Each run generated a unique trading strategy,
733 which was subsequently evaluated on the test set.

734 All experiments were conducted on the CERES cluster of the University
735 of Essex. CERES is a computational cluster built using the Rocks Clustering
736 Solution with CentOS Linux. For computational purposes, the cluster has
737 1632 processing cores (3232 with hyperthreading) provided by servers with
738 a mix of Intel E5-2698, Intel Gold 5115, 6152 and 6238L processors, and
739 between 500Gb and 6Tb RAM each.

740 To assess the statistical significance of our findings, we applied the two-
741 sample Kolmogorov-Smirnov (KS) test to the performance results of all runs
742 across companies for each algorithm. This non-parametric test evaluates
743 whether the distributions of performance metrics differ significantly between
744 ActTrade and each benchmark.

745 Given the multiple comparisons involved, (algorithms $\times$ metrics), we con-
746 trol the false discovery rate (FDR) using the Benjamini–Hochberg (BH) pro-
747 cedure with $q = 0.05$. This choice balances the Type I error control with
748 statistical power across a large set of related tests.

749 5.1. Sharpe ratio

750 Table 6 shows the average Sharpe ratio values across 50 runs for the 42
751 companies. A value of zero (0) for a company indicates that the respective
752 algorithm chose not to trade, likely to avoid potential losses. As observed,
753 GP-SA has the highest occurrence of best Sharpe ratio values, leading in 11

754 companies, followed by STGP-FASATA with 8 companies, GP-TA with 7,
755 and ActTrade with 6. GP-FA achieves the highest values in 5 companies,
756 while STGP-FASATA-S and GP-FASATA lead in 4 and 2 companies, respec-
757 tively. It is also worth noting that both STGP-FASATA-S and ActTrade
758 display several 0 Sharpe ratio values, suggesting that these algorithms are
759 highly conservative. As mentioned earlier, this conservatism allows them to
760 avoid trades in unfavourable situations, potentially improving their average
761 and median performance, as will be discussed next.

762 Table 7 presents summary statistics for the 50 GP runs conducted for
763 each company across the seven GP algorithms. Runs with a Sharpe ratio
764 of 0, where no trades were executed, were excluded from the calculation of
765 these statistics. When interpreting the Sharpe ratio results, it is important
766 to remember that this metric can be sensitive to the number of trades. It
767 can yield disproportionately large values (positive or negative) when only a
768 few trades are made. This occurs because the risk, which is computed as
769 the standard deviation of returns over a given period, can become extremely
770 small if an algorithm performs only a few trades (e.g., 2-4 trades over the
771 entire test set). Since risk is the denominator in the Sharpe ratio calculation,
772 a very small risk value can result in an unusually high Sharpe ratio.

773 As we can observe from Table 7, the proposed ActTrade algorithm has
774 the highest mean and median values, indicating a strong risk-adjusted per-
775 formance compared to the other algorithms and highlighting its robustness
776 and consistency in delivering favourable risk-adjusted returns. Generally, all
777 strongly-typed GP algorithms experience higher mean values, which indicates
778 the advantage of the architecture in comparison to non strongly-typed GPs.

Table 6 Average values for Sharpe ratio per company. Boldface is used to denote the best value for the particular dataset.

<i>Company</i>	<i>GP-FA</i>	<i>GP-SA</i>	<i>GP-TA</i>	<i>GP-FASATA</i>	<i>STGP-FASATA</i>	<i>STGP-FASATA-S</i>	<i>ActTrade</i>
AAPL	5.5915	2.9952	6.2435	6.0608	2.6744	8.4403	7.2507
ADIDAS	0.0455	-0.8810	0.1087	0.1709	0.6694	0	0
ALIBABA	1.3241	12.8724	2.7733	4.4762	2.7338	2.3999	11.8146
AMAZON	1.1791	1.7928	14.7694	5.5078	3.7653	0	0
ASUS	-1.3757	3.4320	0.5216	0.8423	-0.2138	-0.5439	6.3323
ATVI	0.5347	2.5614	3.5067	0.7391	3.0072	0.3750	2.6987
BLACKB	-1.2496	-0.0329	2.8176	1.3741	1.6736	1.0647	1.0846
COKE	8.7271	-0.5819	0.6859	1.2089	1.8917	2.6953	1.1766
EBAY	0.6157	6.3670	0	-0.5972	-0.0702	0.4302	0.1362
ESTEE	0.2138	2.0982	1.2340	0.7858	1.8651	8.7272	1.6824
FORD	-0.1341	8.3010	-0.5957	0.1256	5.2550	0	3.2462
FUJIFILM	1.0442	0.5517	-1.5904	-0.5409	1.0232	-1.7277	0.1871
FUJITSU	0.7533	3.7077	13.7997	15.5743	9.9302	24.6731	15.8708
GM	1.0112	4.0144	-20.9106	-0.2385	-0.6598	0	0
GOOGL	0.0520	2.4776	2.3801	3.7570	1.1243	-0.1268	14.1991
HITACHI	-0.1452	0.2454	0	0.0971	-1.2785	0	0
HONDA	1.2830	6.6325	2.5088	1.6852	2.0304	2.4846	0
HSBC	2.8015	2.9204	0.1729	0.0370	1.9430	0	0.0372
HYUNDAI	2.4618	0.7194	-0.6452	31.4885	10.8667	1.1241	2.3572
IBM	0.4644	0.9346	10.5784	1.8172	6.8871	3.2115	1.7280
INTC	0.8131	1.9789	1.4645	0.9523	2.2122	0	-0.0318
KERING	0.5836	0.7536	0.9268	0.7333	1.1841	0.3960	0
KODAK	1.1804	1.4972	1.6641	0.4966	1.1869	1.0455	1.2845
MCDON	-0.0502	0.9143	-4.3046	0	-0.0126	0	-1.2986
NESTLE	-0.0649	2.3210	-0.0132	2.6653	9.8132	0.1730	5.5454
NFLX	8.0428	17.8119	0.9170	6.6686	3.1125	5.0024	9.3962
NINTENDO	0.1078	1.1302	-0.1277	0.4118	3.8835	0.7198	0.6481
NYT	0.3141	3.2256	-0.7486	0.1152	-0.0254	-2.3197	0.1232
PANASONIC	0.1002	2.5060	3.9634	2.3647	1.3352	3.6128	7.4402
SAINSBURY	14.9238	2.2750	2.0274	5.3955	1.0712	15.2586	2.3479
SHISEIDO	-0.6942	4.9143	0.5602	-2.4178	-1.0828	1.8925	-0.6254
SUBARU	0.6109	-0.1824	7.7342	0	-0.5199	0.0901	2.0504
SUZUKI	5.5550	-1.0981	2.6771	0.6488	0	6.3275	9.1027
TENCENT	0.3319	0	-0.5544	15.8820	4.7378	0.3920	3.7368
TESCO	0.1186	0.9316	2.2420	-0.2571	15.6728	0	-14.5138
TESLA	0.7089	3.5859	2.1825	2.0695	0.9962	1.6463	2.7183
TOYOTA	3.0496	1.1699	0	-0.3441	3.0060	0	0
UBISOFT	1.0275	0	1.5272	4.0745	5.7834	-0.4325	0
WALMART	2.8774	1.1366	1.9536	0.2454	1.5732	0.1634	1.5442
XEROX	6.4406	0.1473	-0.8290	2.3401	1.4751	0	0.3188
YAMAHACO	0.5634	1.3774	0.4464	0.7569	1.7156	2.1979	2.2089
YAMAHAMO	1.6966	0	0.6509	0	6.6631	-0.1101	0.3586

Table 7 Summary statistics of Sharpe ratio across all 50 GP runs and all 42 companies. Boldface is used to denote the best value for each statistic.

<i>Algorithm</i>	<i>Average</i>	<i>Median</i>	<i>StDev</i>	<i>Maximum</i>	<i>Minimum</i>	<i>Trades</i>
GP-FA	2.18	0.67	10.37	83.59	-52.54	100
GP-SA	2.65	0.57	10.75	173.13	-14.16	5
GP-TA	2.32	0.70	5.62	31.47	-20.9	15
GP-FASATA	2.99	0.58	12.1	206.25	-9.21	27
STGP-FASATA	3	0.64	9.9	144.32	-8.81	21
STGP-FASATA-S	3.78	1.07	11.57	83.59	-27.80	17
ActTrade	3.9	1.318	8.22	64.35	-14.51	18

ActTrade’s standard deviation is also third best (GP-TA actually has the lowest values), indicating a higher concentration around the mean. The last column of Table 7 also presents the average number of trades per algorithm. On average, the ActTrade model executed 18 trades per company, compared to 17 for STGP-FASATA-S, 21 for STGP-FASATA, and 25 for GP-FASATA, all of which performed competitively among the benchmark algorithms. We observe a clear trend in which the number of trades decreases as the algorithm becomes more structured and selective — from 100 trades for the simplest GP-FA model down to 18 for ActTrade. At the same time, performance, as measured by average Sharpe ratio, improves consistently, rising from 2.18 to 3.9. Importantly, the more selective algorithms are not merely under-trading, but rather executing higher-quality trades. This points to increased efficiency and better risk-adjusted decision-making as algorithmic

792 sophistication increases.

793 Since each trade incurs the fee once, a uniform per-trade fee reduces
794 a strategy’s cumulative return roughly in proportion to how frequently it
795 trades, so strategies with different turnover are not affected equally by a
796 single fee assumption. We therefore report average trade counts alongside
797 performance metrics (Table 7), allowing readers to judge how sensitive the
798 reported rankings are to this cost assumption.

799 The algorithms fall into two broad groups: a low-turnover cluster (Act-
800 Trade: 18, STGP-FASATA-S: 17, STGP-FASATA: 21) and a higher-turnover
801 cluster (GP-FASATA: 27, GP-TA: 15, GP-FA: 100). Because ActTrade’s
802 closest competitors in Sharpe ratio and rate of return fall within the low-
803 turnover cluster, its advantage over them is relatively insensitive to the exact
804 fee level, since both incur similar cost penalties as fees rise. Its advantage
805 over markedly higher-turnover strategies such as GP-FA is more sensitive to
806 the cost assumption, as GP-FA’s much larger trade count means its returns
807 erode faster under higher fees.

808 To better interpret the results, we conducted pairwise Kolmogorov–Smirnov
809 (KS) tests comparing the ActTrade algorithm (chosen as the control due to
810 its superior average and median performance) against each of the six GP
811 benchmarks. The KS test evaluates whether the distributions of performance
812 metrics (Sharpe ratio, rate of return, and risk), in this case only showing the
813 Sharpe ratio results, for two algorithms originate from the same population.
814 A low p-value indicates a statistically significant difference in their empiri-
815 cal distributions. To control for the multiple pairwise comparisons across al-
816 gorithms and metrics, we applied the Benjamini–Hochberg False Discovery

817 Rate (FDR) correction with $q = 0.05$. This approach balances statistical
818 power across the 18 simultaneous tests. Furthermore, for the sensitivity test,
819 we conducted two analyses, one using only valid runs (non-zero-return
820 strategies) and the other using all runs, including zero-return cases, to assess
821 any potential bias.

822 The results in Table 8 show that ActTrade exhibits statistically signifi-
823 cant differences from nearly all other algorithms across Sharpe ratio under
824 the FDR-adjusted significance level. When including zero-return runs (sen-
825 sitivity test), the results remain robust, with most comparisons remaining
826 significant, with only a few (notably versus GP-FA and STGP-FASATA)
827 losing significance due to the inclusion of inactive runs.

Table 8 Kolmogorov–Smirnov (KS) test p-values comparing ActTrade against GP benchmarks under Benjamini-Hochberg (FDR) correction at $q = 0.05$ for Sharpe ratio. Bold values denote statistically significant differences. Results are shown for valid runs and for all runs (sensitivity analysis).

Dataset	Comparison	p-value	FDR-adjusted p
not incl. zero-return runs	ActTrade vs GP-FA	1.41E-03	2.45E-03
	ActTrade vs GP-SA	2.49E-05	6.40E-05
	ActTrade vs GP-TA	9.78E-101	8.80E-100
	ActTrade vs GP-FASATA	8.87E-06	2.66E-05
	ActTrade vs STGP-FASATA	2.21E-03	3.32E-03
	ActTrade vs STGP-FASATA-S	2.33E-01	2.33E-01
incl. zero-return runs	ActTrade vs GP-FA	1.58E-01	1.67E-01
	ActTrade vs GP-SA	3.26E-34	2.93E-33
	ActTrade vs GP-TA	1.21E-02	1.67E-02
	ActTrade vs GP-FASATA	5.88E-10	2.60E-09
	ActTrade vs STGP-FASATA	4.66E-07	1.40E-06
	ActTrade vs STGP-FASATA-S	6.41E-02	8.24E-02

828 *5.2. Rate of Return*

829 Table 9 presents the average rate of return values for all runs of the 42
830 companies. The algorithm that has the most highest values is ActTrade
831 (9 companies), followed by GP-SA and GP-FA with 8 and 7 companies,
832 respectively. GP-TA, STGP-FASATA and STGP-FASATA-S all have the
833 highest value in 6 companies, while GP-FASATA in none of the companies.

834 Table 10 summarises the results, by presenting the mean, median, stan-

Table 9 Average values for rate of return per company. Boldface is used to denote the best value for the particular dataset.

<i>Company</i>	<i>GP-FA</i>	<i>GP-SA</i>	<i>GP-TA</i>	<i>GP-FASATA</i>	<i>STGP-FASATA</i>	<i>STGP-FASATA-S</i>	<i>ActTrade</i>
AAPL	0.0116	0.0135	0.0237	0.0243	0.0227	0.0277	0.0242
ADIDAS	0.0014	0.0014	0.0050	0.0023	0.0082	0	0
ALIBABA	0.0200	0.0304	0.0146	0.0229	0.0210	0.0244	0.0326
AMAZON	0.0295	-0.0013	0.0205	0.0223	0.0187	0	0
ASUS	-0.0339	0.0009	0.0191	-0.0107	-0.0022	0.0009	0.0119
ATVI	0.0103	0.0099	0.0157	0.0080	0.0124	0.0079	0.0259
BLACKB	0.0062	-0.0089	0.0344	0.0146	0.0281	0.0144	0.0022
COKE	0.0163	-0.0403	0.0085	0.0095	0.0131	0.0111	0.0119
EBAY	0.0094	0.0016	0	-0.0295	-0.0059	0.0102	0.0026
ESTEE	0.0028	0.0095	0.0091	0.0054	0.0094	0.0092	0.0053
FORD	-0.0038	0.0140	-0.0285	0.0006	0.0102	0	0.0182
FUJIFILM	0.0317	0.0308	-0.0257	0.0005	0.0131	-0.0321	0.0071
FUJITSU	0.0355	0.0646	0.0932	0.0894	0.0689	0.1016	0.0833
GM	-0.0007	0.0237	-0.0894	-0.0125	-0.0354	0	0
GOOGL	0.0015	0.0060	-0.0071	0.0019	0.0099	-0.0033	0.0103
HITACHI	-0.0027	0.0125	0	0.0054	-0.0184	0	0
HONDA	0.0111	0.0085	0.0012	0.0128	0.0141	0.0139	0
HSBC	-0.0026	0.0099	0.0023	0.0095	0.0052	0	0.0005
HYUNDAI	0.0130	0.0124	-0.0205	0.0268	0.0191	0.0344	0.0356
IBM	0.0150	0.0100	0.0315	0.0041	0.0134	0.0467	0.0250
INTC	0.0251	-0.0093	0.0016	-0.0092	0.0185	0	-0.0026
KERING	0.0217	0.0179	0.0132	0.0056	0.0273	0.0192	0
KODAK	0.0115	0.0535	0.0293	0.0113	0.0158	0.0228	0.0390
MCDON	-0.0019	0.0123	-0.0003	0	-0.0126	0	-0.0563
NESTLE	-0.0025	0.0082	-0.0024	0.0004	0.0052	-0.0021	0.0039
NFLX	0.0608	0.0033	-0.0105	0.0598	0.0349	0.0470	0.0525
NINTENDO	0.0036	0.0143	-0.0124	0.0147	0.0061	0.0069	0.0168
NYT	0.0127	0.0326	-0.0015	0.0059	0.0053	0.0113	0.0087
PANASONIC	0.0011	0.0260	0.0160	0.0147	0.0068	0.0061	0.0262
SAINSBURY	0.0348	0.0167	0.0154	0.0257	0.0118	0.0270	0.0217
SHISEIDO	-0.0480	-0.0106	0.0208	-0.0399	-0.0292	0.0283	-0.0468
SUBARU	-0.0057	-0.0155	0.0338	0	-0.0256	0.0036	0.0292
SUZUKI	0.0355	-0.0365	0.0007	0.0140	0	0.0646	0.0403
TENCENT	0.0069	0	-0.0543	0.0163	-0.0013	0.0099	0.0187
TESCO	0.0039	0.0164	0.0088	-0.0097	0.0419	0	-0.0302
TESLA	0.0505	0.0550	0.0751	0.0433	0.0413	0.0596	0.0601
TOYOTA	0.0172	0.0087	0	-0.0029	0.0333	0	0
UBISOFT	-0.0044	0	0.0281	0.0267	-0.0118	-0.0443	0
WALMART	0.0034	-0.0028	0.0047	-0.0004	0.0038	0.0029	0.0026
XEROX	0.0315	0.0058	-0.0175	0.0223	0.0405	0	0.0174
YAMAHACO	-0.0001	0.0049	-0.0069	0.0024	0.0076	0.0140	0.0113
YAMAHAMO	0.0160	0	0.0063	0	-0.0091	-0.0025	0.0042

835 dard deviation, maximum and minimum values across all 50 runs for each
836 company. As we can observe, the proposed ActTrade has again the best mean
837 value, and also has a median value very close to the best observed (0.0145
838 vs 0.015). Its standard deviation (0.031) is comparable to the lowest value
839 of 0.028, showing again that it has relatively low volatility. The maximum
840 rate of return value is the same for all algorithms but four algorithms and
841 ActTrade has the lowest minimum value for rate of return.

Table 10 Summary statistics of rate of return across all 50 GP runs and all 42 companies. Boldface is used to denote the best value for each statistic.

<i>Algorithm</i>	<i>Average</i>	<i>Median</i>	<i>StDev</i>	<i>Maximum</i>	<i>Minimum</i>
GP-FA	0.014	0.015	0.038	0.069	-0.30
GP-SA	0.009	0.01	0.028	0.11	-0.12
GP-TA	0.013	0.014	0.033	0.10	-0.089
GP-FASATA	0.015	0.014	0.029	0.10	-0.10
STGP-FASATA	0.0135	0.0125	0.028	0.11	-0.18
STGP-FASATA-S	0.021	0.015	0.035	0.11	-0.087
ActTrade	0.0225	0.0145	0.031	0.11	-0.06

842 Kolmogorov–Smirnov (KS) test p-values comparing ActTrade against GP
843 benchmarks under Benjamini-Hochberg (FDR) correction at $q = 0.05$ for rate
844 of return. Bold values denote statistically significant differences. Results are
845 shown for valid runs and for all runs (sensitivity analysis).

Table 11 Kolmogorov–Smirnov (KS) test p-values comparing ActTrade against GP benchmarks on the Rate of Return. Values are shown for both valid runs and all runs (sensitivity analysis). Reported p-values are raw and Benjamini–Hochberg FDR-adjusted ($q = 0.05$); significant differences are shown in bold.

Dataset	Comparison	p-value	FDR-adjusted p
not incl. zero-return runs	ActTrade vs GP-FA	1.50E-03	2.45E-03
	ActTrade vs GP-SA	3.78E-06	1.36E-05
	ActTrade vs GP-TA	6.85E-90	4.11E-89
	ActTrade vs GP-FASATA	3.29E-03	4.23E-03
	ActTrade vs STGP-FASATA	2.73E-03	3.77E-03
	ActTrade vs STGP-FASATA-S	1.98E-01	2.09E-01
incl. zero-return runs	ActTrade vs GP-FA	9.17E-01	9.17E-01
	ActTrade vs GP-SA	4.61E-35	8.30E-34
	ActTrade vs GP-TA	7.73E-03	1.16E-02
	ActTrade vs GP-FASATA	4.79E-10	2.60E-09
	ActTrade vs STGP-FASATA	1.50E-06	3.39E-06
	ActTrade vs STGP-FASATA-S	1.02E-01	1.23E-01

846 *5.3. Risk*

847 Table 12 presents the average risk values for all runs of the 42 companies.

848 ActTrade has again the highest number of best (lowest) values, along with

849 GP-TA (10 each)¹². GP-SA and STGP-FASATA-S have 8 and 7 companies
850 with the lowest risk values, respectively. Lastly, the algorithms with the least
851 companies are STGP-FASATA (4 companies), GP-FASATA (2 companies)
852 and GP-FA (1 company).

¹²Zero (0) risk values, which indicate that no trading was performed by an algorithm, are not taken into account when comparing values to identify the best performing algorithm.

Table 12 Average values for risk per company. Boldface is used to denote the best value for the particular dataset.

<i>Company</i>	<i>GP-FA</i>	<i>GP-SA</i>	<i>GP-TA</i>	<i>GP-FASATA</i>	<i>STGP-FASATA</i>	<i>STGP-FASATA-S</i>	<i>ActTrade</i>
AAPL	0.0076	0.0236	0.0082	0.0089	0.0174	0.0059	0.0110
ADIDAS	0.0270	0.0148	0.0264	0.0276	0.0261	0	0
ALIBABA	0.0280	0.0040	0.0308	0.0239	0.0264	0.0256	0.0027
AMAZON	0.0248	0.0311	0.0022	0.0179	0.0196	0	0
ASUS	0.0248	0.0219	0.0362	0.0137	0.0131	0.0162	0.0087
ATVI	0.0212	0.0043	0.0083	0.0239	0.0168	0.0201	0.0094
BLACKB	0.0429	0.0869	0.0280	0.0387	0.0352	0.0411	0.0593
COKE	0.0070	0.0701	0.0125	0.0084	0.0091	0.0060	0.0100
EBAY	0.0185	0.0288	0	0.0523	0.0455	0.0234	0.0176
ESTEE	0.0230	0.0044	0.0168	0.0211	0.0139	0.0028	0.0151
FORD	0.0295	0.0057	0.0482	0.0334	0.0147	0	0.0055
FUJIFILM	0.0302	0.0554	0.0291	0.0235	0.0237	0.0193	0.0296
FUJITSU	0.0515	0.0173	0.0170	0.0147	0.0324	0.0079	0.0241
GM	0.0418	0.0166	0.0043	0.0532	0.0530	0	0
GOOGL	0.0253	0.0146	0.0466	0.0198	0.0203	0.0279	0.0104
HITACHI	0.0488	0.0565	0	0.0534	0.0145	0	0
HONDA	0.0100	0.0080	0.0082	0.0175	0.0176	0.0127	0
HSBC	0.0179	0.0054	0.0088	0.0064	0.0078	0	0.0066
HYUNDAI	0.0231	0.0216	0.0446	0.0163	0.0320	0.0304	0.0150
IBM	0.0326	0.0674	0.0038	0.0488	0.0483	0.0145	0.0144
INTC	0.0300	0.0446	0.0317	0.0704	0.0083	0	0.0902
KERING	0.0378	0.0258	0.0343	0.0411	0.0294	0.0478	0
KODAK	0.0463	0.0356	0.0361	0.0753	0.0615	0.0347	0.0345
MCDON	0.0432	0.0397	0.0299	0	0	0	0.0435
NESTLE	0.0237	0.0034	0.0173	0.0139	0.0101	0.0160	0.0102
NFLX	0.0075	0.0622	0.0898	0.0113	0.0271	0.0310	0.0123
NINTENDO	0.0310	0.0357	0.0491	0.0353	0.0206	0.0505	0.0314
NYT	0.0413	0.0190	0.0406	0.0399	0.0405	0.0297	0.0385
PANASONIC	0.0426	0.0109	0.0204	0.0259	0.0262	0.0302	0.0263
SAINSBURY	0.0196	0.0272	0.0229	0.0304	0.0344	0.0198	0.0115
SHISEIDO	0.0699	0.0864	0.0585	0.0417	0.0487	0.0625	0.0758
SUBARU	0.0349	0.0559	0.0170	0	0.0515	0.0380	0.0141
SUZUKI	0.0324	0.0686	0.0561	0.0455	0	0.0102	0.0314
TENCENT	0.0260	0	0.0987	0.0088	0.0227	0.0246	0.0049
TESCO	0.0361	0.0189	0.0224	0.0415	0.0027	0	0.0021
TESLA	0.0407	0.0474	0.0346	0.0550	0.0665	0.0590	0.0426
TOYOTA	0.0127	0.0184	0	0.0276	0.0110	0	0
UBISOFT	0.0466	0	0.0277	0.0297	0.0460	0.0817	0
WALMART	0.0116	0.0176	0.0074	0.0244	0.0153	0.0225	0.0127
XEROX	0.0300	0.0612	0.0213	0.0340	0.0272	0	0.0539
YAMAHACO	0.0314	0.0242	0.0374	0.0263	0.0241	0.0142	0.0156
YAMAHAMO	0.0243	0	0.0175	0	0.0206	0.0229	0.0276

853 Similarly to Section 5.1 and 5.2, Table 13 displays the summary statistics
854 for the values of risk of all 50 runs for each company across the eight GP
855 algorithms. As we can observe, we get a very similar picture to what we
856 have seen so far: the proposed ActTrade algorithm has the lowest mean and
857 median values, as well as the lowest standard deviation value. The lowest
858 maximum risk value is observed by STGP-FASATA-S, while the lowest min-
859 imum risk by STGP-FASATA. The KS statistical test in Table ?? confirms
860 the above results, as the proposed ActTrade algorithm is able to statistically
861 and significantly outperform all other GP algorithms at the 5% level.

Table 13 Summary statistics of risk across all 50 GP runs and all 42 companies. Boldface is used to denote the best value for each statistic.

<i>Algorithm</i>	<i>Average</i>	<i>Median</i>	<i>StDev</i>	<i>Maximum</i>	<i>Minimum</i>
GP-FA	0.029	0.026	0.024	0.21	0.0007
GP-SA	0.034	0.022	0.04	0.25	0.0003
GP-TA	0.027	0.018	0.025	0.18	0.0006
GP-FASATA	0.030	0.023	0.027	0.17	0.00007
STGP-FASATA	0.028	0.022	0.026	0.19	0.000067
STGP-FASATA-S	0.026	0.0177	0.026	0.126	0.00046
ActTrade	0.024	0.016	0.023	0.18	0.0004

862 Table 14 reports the pairwise Kolmogorov–Smirnov (KS) test results for
863 risk, comparing ActTrade against the GP benchmarks. Following the Ben-
864 jamini–Hochberg FDR correction, ActTrade remains statistically distinct
865 from nearly all benchmarks in both valid-run and all-run datasets. These

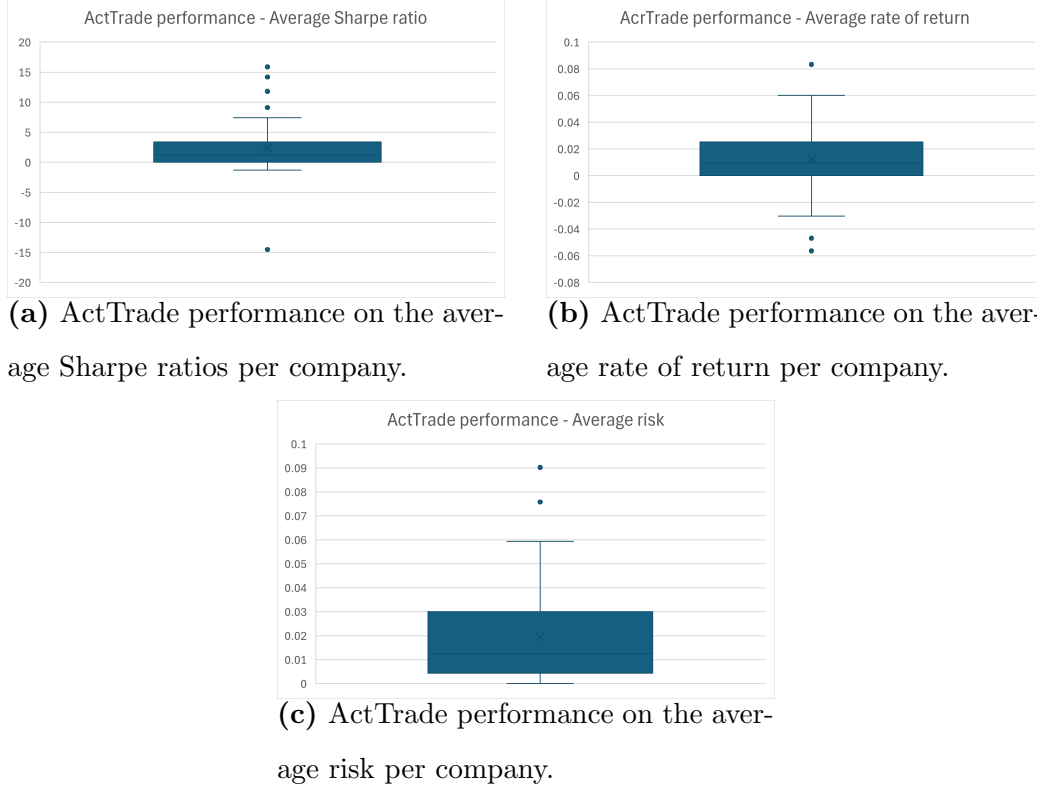
866 outcomes confirm that ActTrade’s superior performance is not sensitive to
867 data filtering and is robust across evaluation metrics.

Table 14 Kolmogorov–Smirnov (KS) test p-values comparing ActTrade against GP benchmarks under Benjamini-Hochberg (FDR) correction at $q = 0.05$ for risk. Bold values denote statistically significant differences. Results are shown for valid runs and for all runs (sensitivity analysis).

Dataset	Comparison	p-value	FDR-adjusted p
not incl. zero-return runs	ActTrade vs GP-FA	2.05E-06	9.22E-06
	ActTrade vs GP-SA	6.52E-05	1.47E-04
	ActTrade vs GP-TA	5.24E-152	9.43E-151
	ActTrade vs GP-FASATA	6.61E-04	1.32E-03
	ActTrade vs STGP-FASATA	5.94E-03	7.13E-03
	ActTrade vs STGP-FASATA-S	1.85E-01	2.09E-01
incl. zero-return runs	ActTrade vs GP-FA	6.34E-06	1.27E-05
	ActTrade vs GP-SA	1.48E-03	2.43E-03
	ActTrade vs GP-TA	4.42E-05	7.95E-05
	ActTrade vs GP-FASATA	7.21E-10	2.60E-09
	ActTrade vs STGP-FASATA	9.16E-07	2.35E-06
	ActTrade vs STGP-FASATA-S	1.37E-01	1.54E-01

868 For completeness, we also present Figure 2, which shows box plots of the
869 ActTrade algorithm for the three metrics of Sharpe ratio, rate of return, and
870 risk.

Figure 2 ActTrade performance in Sharpe ratio, rate of return, and risk.



871 *5.4. The role of the active trading operator*

872 Given the above positive results for the ActTrade algorithm, we are in-
873 terested in understanding the effects of the proposed genetic operator during
874 the learning (training) phase of the models. To do this, we track the fitness
875 value (Sharpe ratio) of the population for each generation.

876 As previously mentioned in Sections 3.2.4 and 3.2.5, the novel genetic op-
877 erator yields offspring through two distinct mechanisms. First, it generates
878 offspring directly, by incorporating the three newly created individuals into
879 the next generation. These individuals are identified as follows: Offspring-

880 FA (where the FA subtree maximises the Sharpe ratio, while the SA and TA
881 subtrees maximise the number of trades), Offspring-SA (with the SA sub-
882 tree maximising the Sharpe ratio, and the FA and TA subtrees maximising
883 trades), and Offspring-TA (where the TA subtree maximises the Sharpe ra-
884 tio, while the FA and SA subtrees maximise trades). Second, the operator
885 indirectly influences offspring generation by using Offspring-FA, Offspring-
886 SA, and Offspring-TA as parents for crossover and mutation. Consequently,
887 at each generation, we encounter not only Offspring-FA, Offspring-SA, and
888 Offspring-TA but also several individuals generated by having these offspring
889 as parents. Furthermore, traditional tournament selection also contributes
890 to the creation of individuals in each generation.¹³

891 To organise our analysis effectively, we categorise the offspring in each
892 generation in the following categories:

- 893 • Offspring-FA
- 894 • Offspring-SA
- 895 • Offspring-TA
- 896 • Tournament Selection Group (Group-Tourn): Individuals in this group
897 are the result of conventional tournament selection.

¹³There is also an elitist individual that exists at each generation, but for the purpose of this section’s analysis, we focus solely on understanding the performance of individuals derived through the proposed genetic operator and how they compare to those generated by tournament selection.

- 898 • FA Group (Group-FA): This group comprises offspring whose first par-
899 ent is Offspring-FA. Given that this is a selection method, several off-
900 spring are generated in this way, hence the word ‘group’ in the naming.
- 901 • SA Group (Group-SA): The SA group includes offspring whose first
902 parent is Offspring-SA.
- 903 • TA Group (Group-TA): Offspring in this group have parents whose first
904 parent is Offspring-TA.

905 Table 15 presents the average Sharpe ratio values for each offspring group,
906 across all generations, over the 50 GP runs. Our primary objective is to rigor-
907 ously compare the performance of offspring generated through our proposed
908 operator with those generated via traditional tournament selection. Note
909 that we exclude duplicate models (i.e. models/trees that exist more than
910 once in a given group). As we can observe, all categories of offspring show
911 high average Sharpe ratio values. More importantly, there are many cases
912 that offspring generated through tournament selection (Group-Tourn) have
913 on average lower fitness values than the other categories. This is an impor-
914 tant finding, because it indicates that offspring generated by the proposed
915 genetic operator (either directly or indirectly through acting as a parent for
916 crossover and mutation) has the ability to create competitive models that
917 often outperform models derived from tournament selection.

Table 15 Average fitness function (Sharpe ratio) values of the different offspring categories, across all generations, over 50 GP runs. Boldface is used to denote the best value for the particular company.

<i>Company</i>	<i>Offspring-FA</i>	<i>Offspring-SA</i>	<i>Offspring-TA</i>	<i>Group-Tourn</i>	<i>Group-FA</i>	<i>Group-SA</i>	<i>Group-TA</i>
AAPL	5.33	39.28	11.63	15.20	15.67	16.85	11.63
ADIDAS	19.43	2.73	6.06	13.62	15.89	13.86	15.29
ALIBABA	14.22	13.08	16.29	11.39	13.26	12.39	15.81
AMAZON	5.32	32.30	42.95	24.27	25.43	27.13	26.56
ASUS	12.13	5.60	23.25	11.83	12.63	12.35	12.07
ATVI	6.95	10.54	13.31	8.54	8.50	8.63	8.37
BLACKB	5.68	16.89	28.64	14.96	15.33	16.98	16.83
COKE	8.61	9.48	21.29	10.62	11.13	10.65	13.24
EBAY	8.68	20.11	34.07	16.18	16.81	17.33	18.46
ESTEE	8.52	7.74	6.80	6.49	7.78	7.67	6.99
FORD	6.60	10.07	18.90	9.54	10.98	11.05	8.92
FUJIFILM	11.03	13.54	15.00	12.00	14.53	14.52	10.32
FUJITSU	17.42	10.06	50.89	20.25	24.53	24.21	19.32
GM	27.42	4.34	63.08	28.01	31.12	28.40	33.35
GOOGL	10.20	39.83	11.62	24.43	25.25	27.69	25.90
HITACHI	206.79	3.81	72.04	141.38	173.94	159.22	169.06
HONDA	14.99	16.94	20.11	16.33	18.42	18.98	17.17
HSBC	3.18	4.28	11.64	5.55	6.14	6.39	6.23
HYUNDAI	5.23	6.31	19.49	11.54	12.28	12.34	12.62
IBM	13.74	106.57	26.19	46.14	41.85	50.43	26.60
INTC	14.62	45.51	7.78	19.78	21.86	23.82	17.26
KERING	11.65	21.62	17.77	12.16	13.69	14.58	15.18
KODAK	4.41	10.41	6.39	5.16	5.93	6.38	5.21
MCDON	19.96	4.43	51.78	31.28	33.16	31.55	35.05
NESTLE	5.44	12.37	23.92	11.30	12.55	13.05	10.94
NFLX	17.24	12.92	24.38	16.66	17.17	16.11	19.79
NINTENDO	5.04	4.72	6.98	4.39	4.86	4.79	4.24
NYT	57.03	8.04	597.05	225.10	241.75	233.01	246.84
PANASONIC	8.07	8.03	15.21	9.73	9.93	10.31	9.34
SAINSBURY	25.82	29.24	35.56	18.64	21.78	22.03	17.83
SHISEIDO	11.54	10.35	28.96	16.75	18.76	17.69	17.14
SUBARU	12.11	5.39	13.89	10.63	11.39	10.29	9.91
SUZUKI	10.66	17.85	19.50	15.66	15.36	15.74	14.34
TENCENT	5.69	12.82	12.11	9.30	7.11	8.42	9.47
TESCO	64.87	16.83	13.56	36.99	45.71	41.62	50.50
TESLA	13.41	19.12	8.58	11.73	13.05	13.78	12.26
TOYOTA	3.61	12.95	279.70	161.56	98.91	103.87	258.83
UBISOFT	15.82	10.47	11.49	15.51	17.07	17.02	18.72
WALMART	3.26	11.28	7.49	6.51	6.23	7.10	6.73
XEROX	19.79	7.84	49.99	27.40	29.48	27.81	34.08
YAMAHACO	3.47	5.31	11.01	6.18	6.96	6.70	6.30
YAMAHAMO	12.98	8.41	15.83	11.16	12.63	12.31	13.53

918 *5.5. Ablation study integrating the three branches in the ActTrade algorithm*

919 For each company, we first exclude runs with $\text{Sharpe}_{\text{Total}} = 0$ (to avoid
 920 degenerate cases), then compute per-company averages over all remaining
 921 runs. We report the mean of these per-company averages across the 42
 922 companies.

923 We evaluate interpretability and robustness of the proposed ActTrade
 924 based trading strategies by performing an internal ablation analysis of the
 925 individual branches in the algorithm. For each company and each branch
 926 ($x \in \{\text{FA}, \text{SA}, \text{TA}, \text{Total}\}$), we compute the following metrics:

Delta Sharpe Ratio Contribution (ΔSRC).

$$\Delta\text{SRC}_{\text{Max}} = \text{Sharpe}_{\text{Total}} - \max(\text{Sharpe}_{\text{FA}}, \text{Sharpe}_{\text{SA}}, \text{Sharpe}_{\text{TA}}) \quad (17)$$

$$\Delta\text{SRC}_{\text{Avg}} = \text{Sharpe}_{\text{Total}} - \frac{\text{Sharpe}_{\text{FA}} + \text{Sharpe}_{\text{SA}} + \text{Sharpe}_{\text{TA}}}{3} \quad (18)$$

Trade share.

$$\text{TradeShare}_x = \frac{\text{Trades}_x}{\text{Trades}_{\text{Total}}}, \quad x \in \{\text{FA}, \text{SA}, \text{TA}\} \quad (19)$$

927 Table 16 reports the results averaged across 42 companies. On average,
 928 the integrated strategy attains a higher Sharpe ratio (3.9) than any indi-
 929 vidual branch (FA 0.84, SA 0.87, TA 1.3), and a positive $\Delta\text{SRC}_{\text{Avg}}$ (2.2),
 930 indicating that the full tree improves over the average performance of the
 931 three separate branches. However, the relatively small $\Delta\text{SRC}_{\text{Max}}$ (0.3) and
 932 the non-significant Wilcoxon result show that the integrated tree does not
 933 systematically outperform the best single branch for a given company. Large

934 standard deviations reflect substantial cross-company heterogeneity, consis-
 935 tent with the market-dependent effectiveness of FA, SA, and TA signals.

Table 16 Internal ablation results (average across companies).

Metric	Mean
Sharpe _{FA}	0.84
Sharpe _{SA}	0.87
Sharpe _{TA}	1.3
Sharpe _{Total}	3.9
$\Delta\text{SRC}_{\text{Max}}$	0.3
$\Delta\text{SRC}_{\text{Avg}}$	2.2
TradeShare _{FA}	71.71%
TradeShare _{SA}	44.74%
TradeShare _{TA}	44.57%

936 Overall, these patterns suggest that ActTrade gains from integrating all
 937 three branches in terms of typical performance and stability, but the effect
 938 is incremental rather than a strong synergy. The integrated tree delivers
 939 Sharpe ratios that are comparable to, and in many cases higher than, the
 940 best individual FA, SA, or TA branch, while avoiding strong dependence
 941 on any single signal. This suggests that ActTrade’s strength lies in providing
 942 robust performance rather than systematically outperforming the optimal
 943 branch.

944 To isolate the contribution of each of the three novel components intro-
 945 duced in this study, we map the evidence above explicitly onto each compo-
 946 nent in turn.

947 For the active trading operator, Section 5.4 and Table 15 show that off-
948 spring generated directly or indirectly by this operator achieve, on average,
949 higher fitness (Sharpe ratio) than offspring generated by standard tourna-
950 ment selection across the majority of companies, indicating that this opera-
951 tor’s contribution to performance is isolated from the effect of the strongly-
952 typed architecture itself.

953 For the third analysis branch, the ΔSRC_{Max} and ΔSRC_{Avg} results re-
954 ported above indicate that the integrated tree improves over average single-
955 branch performance, though it does not systematically exceed the single best-
956 performing branch for every company. This suggests the benefit of integration
957 is incremental and stability-driven rather than a uniform synergy across all
958 cases.

959 For the fitness function, the comparison between STGP-FASATA and
960 STGP-FASATA-S (Tables 7, 10, 13) isolates its contribution, where both
961 share the same strongly-typed architecture and lack the active-trading op-
962 erator, differing only in the fitness function used. STGP-FASATA-S’s im-
963 provement over STGP-FASATA across Sharpe ratio, rate of return, and risk
964 indicates a positive, isolated contribution from the proposed fitness function,
965 independent of the active-trading operator.

966 5.6. Results of each market

967 Although all 42 companies were observed over the same time period, their
968 price trajectories vary significantly, with some exhibiting an overall upward
969 trend, while others trend downward. We therefore consider it important to
970 examine the performance of the GP algorithms across these different market
971 profiles to understand their adaptability under varying conditions.

972 To do this, we looked at the first and last price of the test set for each
973 company, and calculated the return value. We then created three groups:

- 974 • Group 1, which includes those companies whose price experienced a
975 long-term increase of at least 20%.
- 976 • Group 2, which includes those companies whose price experienced a
977 long-term increase between 0% and 19.99%.
- 978 • Group 3, which includes those companies whose price experienced a
979 long-term decrease, i.e. had a negative return.

980 After defining the above groups, 19 companies were placed in Group 1,
981 14 in Group 2, and 9 in Group 3.

982 We then report the average value of each metric (Sharpe ratio, rate of
983 return, and risk) for each GP algorithm, across the datasets of each group.
984 As we can observe from Table 17, the proposed ActTrade performs strongly
985 in terms of the aggregate metric of Sharpe ratio for Group 1, and it has a
986 high value for Group 3, while in Groups 2 and 3 the winner is GP-FASATA.
987 With regards to rate of return, ActTrade has the best value for Group 1 and
988 Group 3, while in Group 2 the best performance is that of STGP-FASATA-
989 S. Lastly, in terms of risk ActTrade performs way lower than the rest of the
990 algorithms for Group 2 and Group3, while STGP-FASATA has the lowest
991 risk for Group 1.

992 In summary, ActTrade achieves the best rate of return in two of the groups
993 and consistently delivers low risk values across all groups. Given the variabil-
994 ity of results across groups and metrics, the Sharpe ratio is particularly use-
995 ful as an aggregate measure of both return and risk. As discussed, ActTrade

demonstrates strong performance in datasets with significant positive price movements (Group 1) and performs best in two metrics for datasets with negative price movements (Group 3). It also performs notably well, particularly in terms of risk, for datasets with average price movements (Group 2). These results suggest that our proposed algorithm excels in both strongly uptrend and downtrend markets—an important finding that highlights its versatility across different market conditions. Furthermore, the solid performance of the strongly-typed GP algorithms in terms of the Sharpe ratio emphasises the value of the strongly-typed GP architecture in effectively combining different types of analysis indicators.

These three groups of companies also provide an indirect view of the model’s behaviour under varying market conditions — with Group 3 (declining prices) and Group 2 (moderate price changes) likely encompassing periods of higher volatility or lower trading volume. The model’s ability to maintain favourable Sharpe ratios and minimise risk in these groups demonstrates its robustness under such challenging conditions.

Table 17 Separated average results per metric per trend group.

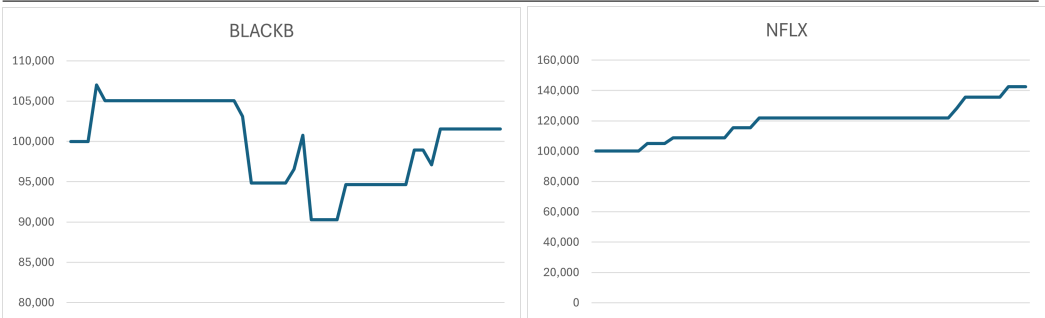
<i>Market</i>	<i>Algorithm</i>	<i>Sharpe ratio</i>	<i>Rate of return</i>	<i>Risk</i>
Group 1 (>20%)	GP-FA	1.63	0.016	0.028
	GP-SA	2.2	0.013	0.029
	GP-TA	2.7	0.012	0.029
	GP-FASATA	2.49	0.014	0.029
	STGP-FASATA	2.55	0.017	0.023
	STGP-FASATA-S	3.3	0.017	0.024
	ActTrade	4.17	0.020	0.027
Group 2 (0% - 19.99%)	GP-FA	1.09	0.003	0.031
	GP-SA	3.8	0.01	0.034
	GP-TA	1.89	0.0035	0.039
	GP-FASATA	4.86	0.006	0.030
	STGP-FASATA	3.7	0.008	0.029
	STGP-FASATA-S	1.8	0.02	0.029
	ActTrade	1.8	0.011	0.021
Group 3 (<0%)	GP-FA	2.99	0.010	0.030
	GP-SA	2.26	0.001	0.036
	GP-TA	-1.32	-0.001	0.025
	GP-FASATA	1.43	0.006	0.035
	STGP-FASATA	2.26	-0.001	0.033
	STGP-FASATA-S	3.5	0.011	0.030
	ActTrade	2.18	0.012	0.024

1012 Lastly, Figure 3 presents the average equity curves over 50 GP runs for
1013 BlackBerry and Netflix during the test period from February 2019 to January
1014 2020. These two companies exhibited volatile price trajectories: BlackBerry's

1015 share price declined by nearly 25%, while Netflix saw a more modest aver-
1016 age daily volatility of around 2%, a maximum drawdown of almost 34%,
1017 and ended the period with a 1.3% increase. We include these equity curves
1018 to demonstrate the algorithm’s ability to adapt to challenging and uncer-
1019 tain market conditions—preserving capital during sharp downturns (as with
1020 BlackBerry) and capturing gains even in mildly declining markets (as seen
1021 with Netflix).

1022 As shown, ActTrade effectively managed risk in the face of a steep de-
1023 cline in BlackBerry’s price, limiting drawdowns to the low 90k range and ul-
1024 timately recovering to finish slightly above its starting equity. This contrasts
1025 sharply with the substantial losses that would have occurred under a passive
1026 buy-and-hold strategy. For Netflix, the strategy steadily compounded gains,
1027 rising from 100k to over 140k despite the stock’s overall volatility. These re-
1028 sults highlight ActTrade’s dual strengths in volatile conditions: strong draw-
1029 down control and the ability to capitalise on trading signals to generate
1030 consistent returns, even when market direction is unfavourable.

Figure 3 ActTrade equity curves for BlackBerry (BLACKB) and Netflix (NFLX).



1031 5.7. *Best tree results*

1032 Although the average performance metrics discussed in previous sections
1033 provide a general overview of how each algorithm performs across multiple
1034 runs, this section focuses on the best-performing outcome achieved by each
1035 genetic programming (GP) algorithm. Specifically, for each of the 50 inde-
1036 pendent runs conducted per company, we identify the tree with the highest
1037 fitness score on the training set. This top-performing tree is then evaluated
1038 on the corresponding unseen test set to assess its generalisation ability. Thus,
1039 the results in this section represent the single best model produced across
1040 the multiple runs, per dataset.

1041 This type of analysis holds particular relevance in financial applications.
1042 In practice, investors and practitioners who use GP-based trading strategies
1043 often execute multiple runs and select the most promising model, typically
1044 the one with the highest fitness or best financial performance, for deployment
1045 in live trading. Therefore, an algorithm’s capacity to generate high-quality
1046 individual solutions, even if only occasionally, is a key strength in real-world
1047 financial scenarios.

1048 Table 18 reports the average test-set performance of these best-performing
1049 trees across all 42 datasets for each GP algorithm. To ensure completeness,
1050 this table includes all GP variants introduced earlier, even those not previ-
1051 ously discussed in detail with respect to best-tree performance, in order to
1052 avoid redundancy in earlier sections.

1053 For consistency with prior analyses, any company where all runs produced
1054 a performance value of 0 has been excluded from this summary. Lastly, note
1055 that results are presented separately for each financial performance metric

1056 (Sharpe ratio, rate of return, and risk) to facilitate a clearer comparison of
1057 the algorithms' strengths.

Table 18 Best trees average performance across the 42 datasets

<i>Algorithm</i>	<i>Sharpe ratio</i>	<i>Rate of return</i>	<i>Risk</i>
GP-FA	-3.92	-0.003	0.025
GP-SA	1.77	0.018	0.024
GP-TA	0.22	0.003	0.05
GP-FASATA	0.95	0.020	0.025
STGP-FASATA	3.23	-0.004	0.027
STGP-FASATA-S	1.76	0.020	0.04
ActTrade	3.51	0.039	0.021

1058 As we can observe, the proposed ActTrade algorithm has the highest
1059 Sharpe ratio and rate of return values while achieving the lowest risk. STGP-
1060 FASATA-S, STGP-FASATA and GP-SA algorithms have the highest Sharpe
1061 ratio values after ActTrade, although STGP-FASATA does have the lowest
1062 rate of return. As the Sharpe ratio is an aggregate metric that considers both
1063 return and risk, the fact that the best tree of ActTrade has the best value
1064 makes it a very positive result. It is also worth noting that practitioners pay
1065 particular attention to such aggregate metrics [54], thus ActTrade best tree's
1066 performance is of particular importance.

1067 To further contextualise the best tree results, we compared them against
1068 the average performance across all 50 runs. For the ActTrade algorithm, the
1069 best tree achieved a Sharpe ratio of 3.51, compared to an average of 3.9; a

1070 rate of return of 0.039, versus an average of 0.0225; and a risk value of 0.021,
1071 versus an average of 0.023. These relatively small differences indicate that the
1072 best-performing model is not an extreme outlier, but rather representative
1073 of the upper end of the distribution. This reinforces the relevance of the
1074 best tree analysis in practical financial settings, where users may reasonably
1075 select and deploy the strongest model from a batch of independent runs.

1076 *5.8. Algorithmic Complexity*

1077 The computational complexity of the proposed ActTrade GP algorithm is
1078 driven by (i) the cost of fitness evaluation, (ii) the cost of GP operators under
1079 the strongly-typed constraints, and (iii) the overhead of the active-trading
1080 operator. Let N denote the population size, G the number of generations, T
1081 the number of trading days, and D the maximum tree depth.

1082 For each individual, the fitness function evaluates four Sharpe ratios, the
1083 global strategy and the three FA, SA, and TA subtrees. Each Sharpe com-
1084 putation requires iterating through the entire price series, executing trading
1085 logic, and recording the return distribution. Since all three subtrees reuse the
1086 same daily data and trading simulation, the asymptotic cost remains linear
1087 in T . Thus, the per-individual fitness cost is $O(T)$.

1088 The crossover operator exchanges compatible subtrees between parents
1089 and the strong typing constrains the exchange to subtrees of the same analysis
1090 type (FA, SA, TA). Since tree depth is limited to D , crossover requires time
1091 $O(D)$.

1092 Point mutation involves replacing a function or terminal node with an-
1093 other node of the same type. This also requires at most a subtree regeneration
1094 step bounded by depth D , resulting in $O(D)$ for mutation.

1095 Across the entire population, the cumulative operator cost per genera-
 1096 tion is therefore $O(ND)$, which is dominated by the $O(NT)$ fitness term for
 1097 realistic values where $T \gg D$.

1098 For the active trading operator, we identify the high-activity, risk-adjusted
 1099 individuals and inject modified variants of their subtrees into the popula-
 1100 tion. Thus, we have two main computations, candidate selection and tree
 1101 construction. The first ranks individuals by activity and Sharpe ratio uses
 1102 statistics already computed during fitness evaluation and therefore requires
 1103 only $O(N)$. The second recombines FA, SA, and TA subtrees which produce
 1104 new individuals with a time bounded by $O(D)$.

1105 Thus, the overall overhead introduced by the active trading operator per
 1106 generation is $O(N + D)$.

1107 In addition to these operators, elitism selects the best individuals from the
 1108 current population and copies them into the next generation. This requires
 1109 ranking or partially sorting individuals by fitness. Even if we assume that
 1110 the algorithm is sorting the entire population, the computational cost is at
 1111 most $O(N \log N)$ per generation. In practice, since the elitist set is small,
 1112 the effective cost is closer to $O(N)$, but in either case it is dominated by the
 1113 fitness-evaluation cost when $T \gg \log N$. Thus, elitism does not change the
 1114 overall complexity of the algorithm.

1115 For the overall complexity, we aggregate the contributions of all compo-
 1116 nents, yielding a computational cost of $O(NGT) + O(NGD) + O(NG)$.

1117 Since $T \gg D$ in typical financial datasets and T dominates all constant
 1118 or depth-bounded terms, the complexity simplifies to the standard GP form
 1119 $O(NGT)$, representing linear scaling in the number of trading days, popula-

tion size, and generations.

5.9. Indicators' usage and model interpretability

In the indicator usage analysis (see Table 19), we focused on the best-performing tree from each algorithm for each of the 42 companies. To provide a comprehensive overview, we computed the total number of times each indicator was used across all datasets. This analysis reveals a well-distributed utilisation of indicators across the GP algorithms.

Notably, the GP-SA and STGP-FASATA-S algorithms tend to produce trees with higher complexity. For GP-SA, this is due to the consistently high use of indicators across companies, while for STGP-FASATA-S, it stems from outlier cases in certain datasets. In contrast, the GP-FASATA algorithm shows less indicator overuse. Among the indicator groups, technical analysis (TA) features are more frequently selected by GP-FASATA, while the strongly-typed GP variants show balanced usage across all three analysis types. As expected, STGP-FASATA-S –guided by its fitness function– consistently selects a broad range of features from FA, SA, and TA. Meanwhile, ActTrade demonstrated the ability to select from a wide range of indicators while relying on a smaller subset, leading to more compact trees and improved predictive performance.

This ability to generate smaller trees marks an important step towards model interpretability. As shown in Table 20, the average number of nodes in the final generation for the ActTrade algorithm is 24, while the average number for best-performing trees is 22. Many of these trees are simple and easy to interpret, as illustrated in Figure 4, which shows a representative example from the Apple dataset. The tree consists of 14 nodes and encodes a

1145 straightforward trading strategy. The root node is a 3-AND, combining funda-
 1146 mental analysis (LT_{FA}), sentiment analysis (AND_{SA}) and technical analysis
 1147 (LT_{TA}) sub-branches, of them two using one indicator and associated thresh-
 1148 olds and the third going further, using two indicators and the corresponding
 1149 thresholds.

1150 Beyond indicator frequencies, the interpretability of the evolved mod-
 1151 els can be assessed by examining the composition of representative trees.
 1152 Figure 4 shows such an example for Apple (AAPL), where the ActTrade al-
 1153 gorithm produces a compact 14-node tree. The tree structure demonstrates
 1154 three important interpretability characteristics:

- 1155 1. **Balanced feature integration:** The root node is a 3-AND oper-
 1156 ator, requiring agreement between one FA indicator (**Quick Ratio**),
 1157 one SA composite node (**SUMsubj**, **SUMsen**), and one TA indicator
 1158 (**Midprice-5**). This reflects the algorithm’s design to encourage the
 1159 combination of difference indicator types.
- 1160 2. **Threshold-based transparency:** The tree encodes explicit thresh-
 1161 olds (e.g., **Quick Ratio** < 0.98 , **Midprice-5** < 0.45), making the decision
 1162 rules interpretable and actionable for human analysts. Unlike black
 1163 box models, this allows the user to understand the economic meaning
 1164 behind each branch.
- 1165 3. **Selective complexity:** Although ActTrade is capable of producing
 1166 large trees (see Table 20), many of the best-performing individuals are
 1167 compact.

1168 When we combine the above, we observe a synergy between frequency
 1169 analysis and structural interpretability. While some indicators (e.g., Price-to-

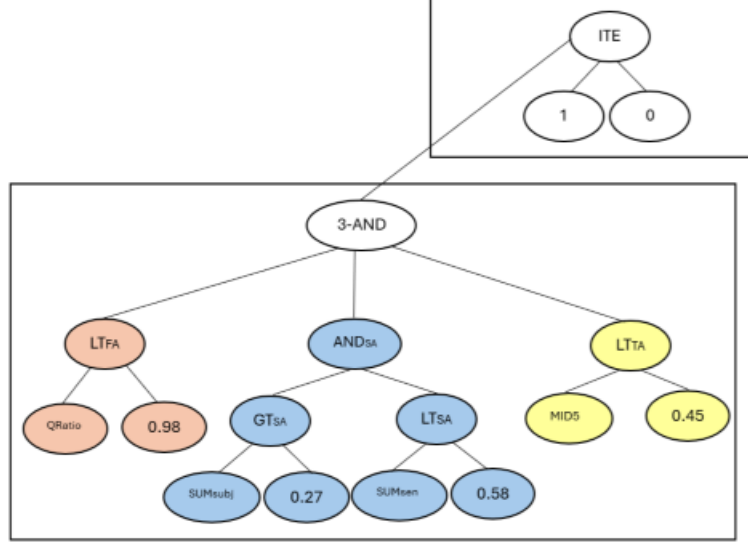
Table 19 Indicators from best models of 42 companies for strongly-typed GP algorithms.

<i>Indicator</i>	<i>GP-FA</i>	<i>GP-SA</i>	<i>GP-TA</i>	<i>GP-FASATA</i>	<i>STGP-FASATA</i>	<i>STGP-FASATA-S</i>	<i>ActTrade</i>
Net Profit Ratio	47	-	-	22	7	35	1
Return on Equity	27	-	-	17	6	16	4
Quick Ratio	42	-	-	24	8	43	2
Price-Earnings ratio	26	-	-	10	10	33	7
Price to Book ratio	55	-	-	29	4	73	12
Price-Sales ratio	89	-	-	40	7	88	13
Debt to Equity	104	-	-	15	10	62	13
Total Revenues	40	-	-	7	6	30	1
Levered free Cash Flow	50	-	-	17	3	12	4
Diluted EPS	30	-	-	13	7	11	2
EBITDA	39	-	-	7	5	8	3
R&D Expenses	37	-	-	28	0	48	6
TEXTpol	-	90	-	15	6	21	11
TEXTsubj	-	52	-	15	9	20	3
TTITLEpol	-	88	-	53	6	36	4
TTITLEsubj	-	91	-	28	10	67	6
SUMMpol	-	69	-	14	8	29	6
SUMMsubj	-	63	-	30	7	72	2
TEXTsenti	-	41	-	17	1	26	6
TITLEsenti	-	44	-	14	2	23	4
SUMMsenti	-	62	-	26	11	17	8
TEXTafinn	-	121	-	20	4	93	9
TITLEafinn	-	94	-	23	11	43	5
SUMMafinn	-	169	-	14	11	165	10
Moving Average - 5 days	-	-	40	34	8	102	11
Moving Aeverage - 10 days	-	-	62	64	12	89	9
Momentum - 5 days	-	-	46	17	9	18	6
Momentum - 10 days	-	-	38	18	4	21	10
ROC - 5 days	-	-	60	17	9	24	5
ROC - 10 days	-	-	40	12	6	18	13
William %R - 5 days	-	-	45	14	9	52	16
Williams %R - 10 days	-	-	39	38	14	38	9
Midprice - 5 days	-	-	53	24	11	39	7
Midprice - 10 days	-	-	45	51	7	33	6
Volatility - 5 days	-	-	53	19	9	59	8
Volatility - 10 days	-	-	51	64	7	50	15

Table 20 Final and Best Tree Total Nodes per Company

Company	Final Tree Nodes	Best Tree Nodes
AAPL	14	14
ADIDAS	14	14
ALIBABA	26	30
AMAZON	10	10
ASUS	26	14
ATVI	22	14
BLACKB	10	22
COKE	18	14
EBAY	18	14
ESTEE	10	14
FORD	18	10
FUJIFILM	26	14
FUJITSU	22	18
GM	18	18
GOOGL	10	30
HITACHI	18	18
HONDA	14	30
HSBC	14	10
HYUNDAI	42	22
IBM	10	22
INTC	26	14
KERING	18	14
KODAK	30	18
MCDON	14	26
NESTLE	22	14
NFLX	26	30
NINTENDO	62	14
NYT	18	42
PANASONIC	46	42
SAINSBURY	10	14
SHISEIDO	18	22
SUBARU	10	22
SUZUKI	22	66
TENCENT	46	30
TESCO	30	10
TESLA	30	42
TOYOTA	46	46
UBISOFT	26	22
WALMART	42	30
XEROX	18	18
YAMAHACO	22	14
YAMAHAMO	38	10

Figure 4 Tree of ActTrade for AAPL.



1170 Sales ratio, Debt-to-Equity, SUMMafinn) appear frequently across datasets,
 1171 their actual role within final trees varies, since they may be used in simple
 1172 one-node tests or in deeper architectures. This suggests that indicator fre-
 1173 quency alone does not dictate model complexity; rather, the combination
 1174 of the FA, SA, and TA indicators contributes to robust and generalisable
 1175 strategies.

1176 5.10. Non-GP Benchmarks

1177 5.10.1. ActTrade compared to the Machine Learning benchmarks

1178 For further algorithmic benchmarks, aside of the GP variants, the built-
 1179 in models of scikit-learn library are used in Python for the machine learning
 1180 algorithms.

1181 As shown in Table 21, the average values for MLP across the 42 companies
 1182 were 0.30 for the Sharpe ratio, 0.009 for the rate of return, and 0.041 for risk.

1183 These results contrast with the significantly higher values achieved by the
1184 GP algorithms, particularly in Sharpe ratio and risk. The KS statistical test
1185 revealed that MLP was statistically different from ActTrade, with p-values
1186 of $7.45e - 10$ for the Sharpe ratio, 0.004 for the rate of return, and 0.0018 for
1187 risk, all well below the 5% significance level.

1188 Similar trends were observed with the SVM algorithm, which had an
1189 average Sharpe ratio of 0.37, a rate of return of 0.01, and a risk value of
1190 0.038. Again, significant differences were noted in the Sharpe ratio and risk.
1191 The KS tests indicated that SVM was statistically different from ActTrade,
1192 with p-values of $7.45e - 10$ for the Sharpe ratio, 0.0018 for the rate of return,
1193 and 0.018 for risk, all falling below the 5% significance threshold.

1194 For the XGBoost algorithm, the average values were 0.36 for the Sharpe
1195 ratio, 0.010 for the rate of return, and 0.041 for risk. ActTrade outperformed
1196 XGBoost in the financial metrics, and the KS tests confirmed that the novel
1197 GP algorithm was statistically different from XGBoost. The p-values were
1198 $3.89e - 05$ for the Sharpe ratio, 0.0001 for the rate of return, and 0.041 for
1199 risk.

1200 Finally, the LSTM algorithm had average values of 0.23 for the Sharpe
1201 ratio, 0.006 for the rate of return, and 0.043 for risk. ActTrade performed
1202 better across all three financial metrics, and the KS tests showed statistical
1203 differences between the novel GP algorithm and LSTM, with p-values of
1204 $3e - 05$ for the Sharpe ratio, $1.13e - 06$ for the rate of return, and 0.0040 for
1205 risk.

1206 These results highlight the limitations of these four algorithms compared
1207 to GP algorithms in algorithmic trading, particularly when creating strate-

gies that account for both returns and risk.

Table 21 Average values for ActTrade, MLP, SVM, XGBoost, and LSTM

Algorithm	Sharpe ratio	Rate of return	Risk
MLP	0.30	0.009	0.041
SVM	0.37	0.010	0.038
XGBoost	0.36	0.010	0.041
LSTM	0.23	0.006	0.043
ActTrade	3.9	0.0225	0.024

5.10.2. ActTrade compared to financial benchmarks

In this section, we compare the ActTrade algorithm to the Buy and Hold (BnH) strategy. To ensure a fair comparison, we run ActTrade using maximisation of the cumulative returns as its fitness function, which measures the total profit or loss over a given period. We focus on cumulative returns rather than the Sharpe ratio because BnH involves a single trade—buying one unit of stock on the first day and selling it on the last day, without considering the rate of return and risk metrics that factor into the Sharpe ratio calculation.

When comparing ActTrade and BnH, the GP algorithm achieves an average cumulative return of 0.75 with a median of 0.91, whereas BnH records an average of 0.16 and a median of 0.169. The Kolmogorov-Smirnov test confirms that these results are statistically different at the 5% level, with a p-value of $3.93e - 06$.

Furthermore, we compare ActTrade to the same trading strategy ActTrade uses when training. As noted in Section 3.2.2, the $TS_{d,r}$ strategy buys

one unit of stock on the first day of trading and sells it when the price increases by $r\%$ within d days or at the end of the d -day period.

When comparing ActTrade and TSd, r , we first observe that TSd, r performs significantly more trades on average (240) compared to ActTrade, which averages only 19 trades. This is expected, as ActTrade focuses on the most profitable and low-risk opportunities, choosing not to trade in less favourable conditions. As a result, ActTrade shows superior performance across all three financial metrics, as seen in Table 22. The KS test confirms that the differences in Sharpe ratio and risk are statistically different at the 5% level, with p-values of $4.3e - 13$ for the Sharpe ratio, $1.13e - 6$ for the rate of return, and 0.00011 for risk.

Table 22 Average values of ActTrade and TSd, r .

<i>Algorithm/Metric</i>	<i>Sharpe ratio</i>	<i>Return</i>	<i>Risk</i>
TSd, r	0.10	0.006	0.067
ActTrade	3.9	0.0225	0.024

5.11. Real-world deployment and scalability

Real-time scalability and practical deployment considerations are crucial for the applicability of any algorithmic trading system. In this section, we discuss the computational resources and time requirements for both training and live execution of the proposed ActTrade GP algorithm.

The training phase of ActTrade is computationally intensive, involving the evolutionary process across multiple generations and numerous candidate solutions (GP individuals). This process typically takes several hours to

complete on standard computational hardware, depending on dataset size and complexity.

Table 23 presents the average training time per run, per company, in minutes for all algorithms tested in this paper. Note that the GP-based models are trained over 50 generations in one run to evolve full trading strategies, while our Machine Learning benchmarks (MLP, SVM, XGBoost, LSTM) are trained to perform binary classification. As such, the reported times reflect different computational objectives and are not directly comparable in terms of task complexity. Nevertheless, they provide a rough estimate of training cost across methods. Also, it needs to be noted that the computational times per algorithm were very close across all companies.

Table 23 Computational times per algorithm

Algorithm	Value (in minutes)
GP-FA	1.51
GP-SA	1.55
GP-TA	1.40
GP-FASATA	1.53
STGP-FASATA	1.36
STGP-FASATA-S	7.10
ActTrade	7.48
MLP	0.05
SVM	0.05
XGBoost	0.08
LSTM	0.07

1254 Although GP-based models require longer training times compared to
1255 traditional machine learning models, they tackle a fundamentally more com-
1256 plex task: directly evolving complete trading strategies rather than classify-
1257 ing short-term stock movements. This increased complexity naturally results
1258 in longer training durations. However, in algorithmic trading practice, this
1259 training process typically occurs offline, while only the final evolved mod-
1260 els are deployed in real-time. Once trained, the evolved GP trading rules
1261 consist of simple mathematical and logical expressions that can be executed
1262 extremely efficiently, requiring just milliseconds per decision. Therefore, de-
1263 spite their offline computational demands, GP-based models like ActTrade
1264 remain highly suitable for practical real-time trading scenarios.

1265 Evaluating the GP strategies is computationally negligible, but efficient
1266 storage and streaming of real-time market and sentiment data are necessary
1267 to ensure timely decisions. The modular design of ActTrade makes it compat-
1268 ible with existing parallel data pipelines, allowing it to scale from a handful of
1269 assets to large portfolios in practical trading environments. We recognise the
1270 need for periodic retraining of the GP models to maintain high performance
1271 as market conditions evolve. In a live deployment, retraining could be con-
1272 ducted at intervals aligned with data availability and market dynamics, such
1273 as monthly or quarterly, to incorporate recent market behaviour and ensure
1274 continued adaptability and performance. This a common practice, aligning
1275 with previous research [55, 56], where it is found that constant retraining is
1276 not necessary and periodic retraining maintains accuracy while cutting costs
1277 and energy usage.

1278 Furthermore, integrating ActTrade into a live trading pipeline would in-

1279 involve continuously updating the input features (fundamental, sentiment, and
1280 technical indicators) as new data becomes available. For instance, fundamen-
1281 tal and sentiment indicators can be refreshed daily or whenever new financial
1282 reports or news articles are published. Technical indicators can be updated at
1283 even higher frequencies if intraday data is utilised. Given the low latency of
1284 evaluating the GP-generated rules, trading decisions can be generated almost
1285 instantaneously upon receipt of updated input data.

1286 From an engineering perspective, scaling ActTrade for real-time operation
1287 could be effectively managed using parallel computing. The inherent paral-
1288 lelism in GP evaluation (where multiple GP individuals or trading rules can
1289 be evaluated simultaneously) can be leveraged to ensure scalability across
1290 large datasets or multiple trading assets (e.g., see [57]).

1291 In addition to computational scalability, practical deployment also re-
1292 quires consideration of execution latency, transaction costs, and market con-
1293 straints. The latency requirements of ActTrade depend on the trading style
1294 as mentioned above. Since the evolved rules consist of lightweight mathemat-
1295 ical expressions, ActTrade can satisfy these low-latency constraints with ease,
1296 since its usage is to already create trading strategies and does not require
1297 daily training.

1298 Overall, ActTrade is designed to accommodate real-world trading require-
1299 ments efficiently. With proper engineering and system design, our approach
1300 is well-suited for practical deployment in real-time algorithmic trading envi-
1301 ronments.

5.12. Summary of findings

To sum up, as seen in Tables 7 - 22 and focusing on the findings from ActTrade, the results have been summarised in 2 categories: first on GP variants' results and, second, on the results of other benchmarks.

When comparing the GP variants with each other, it is evident that:

- The proposed ActTrade algorithm demonstrates strong performance across all three financial metrics—Sharpe ratio, rate of return, and risk—while also statistically and significantly outperforming other GP variants.
- Developing trading strategies that incorporate information from all three indicator types (as in the strongly-typed GP variants) results in more profitable and lower-risk strategies compared to those that rely on a single indicator type (e.g., GP-FA, GP-SA, GP-TA).
- The strongly-typed architecture of the proposed algorithm enables the GP to more effectively explore the space of FA, SA, and TA indicators, compared to a GP that includes all indicator types in its terminal set without the strongly-typed structure (GP-FASATA). This enables a more structured and focused exploration of the indicator space, balancing exploration with exploitation.
- The proposed genetic operator has the ability to create strong offspring (in terms of Sharpe ratio), which often outperform offspring derived by tournament selection. This novel operator supports optimisation by actively generating high-quality solutions that maintain a favourable risk-return balance.

1326 Regarding the comparison of ActTrade to the non-GP benchmarks, we
1327 observe the following:

- 1328 • ActTrade outperforms the four algorithmic benchmarks, with statisti-
1329 cally significant results. The higher risk associated with the machine
1330 learning benchmarks is reflected in their low Sharpe ratio values, all
1331 below 0.4, while the novel GP algorithm achieves a rate of return that
1332 is twice that of the four algorithmic benchmarks.
- 1333 • The ActTrade algorithm is more financially advantageous compared
1334 to both the Buy and Hold strategy and the TSd, r trading strategy.
1335 It outperforms the Buy and Hold strategy in cumulative returns and
1336 surpasses TSd, r across all financial metrics. These results underscore
1337 the robustness of the ActTrade algorithm, particularly in its ability to
1338 manage uncertainty and adapt to the dynamic conditions of financial
1339 markets.

1340 6. Conclusion

1341 In conclusion, this article introduced a novel strongly-typed GP algorithm
1342 that integrates fundamental, sentiment, and technical analysis indicators.
1343 The algorithm also featured an innovative fitness function that rewards both
1344 individual subtrees and the overall tree. Additionally, a new genetic operator
1345 was proposed to promote active trading. This novel approach aligns with key
1346 soft computing principles, offering a flexible, adaptive solution to financial
1347 trading challenges by managing uncertainty and exploiting the search space
1348 more effectively.

1349 The importance of combining the three analysis indicator types, which
1350 is not usually occurring in the previous studies, is evident from our find-
1351 ings. Combining the indicators enhances the models’ knowledge and create
1352 financially more advantageous trading strategies. Moreover, based on our
1353 analysis, it is not profitable enough to simply combine the different types of
1354 indicators, as GP-FASATA does, and a strongly-typed architecture is essen-
1355 tial towards achieving an improved performance. Lastly, the use of the novel
1356 fitness function and the novel genetic operator leads to significantly improved
1357 results across all three metrics of Sharpe ratio, rate of return, and risk.

1358 While our findings are promising, it is important to acknowledge the lim-
1359 itations of this study. Our experiments were conducted on historical data
1360 and a fixed set of international stocks, and as such, real-world performance
1361 may vary under different market conditions or asset classes. Additionally, the
1362 model’s flexibility and customisability come with added structural complex-
1363 ity, which can pose challenges for interpretability, although we have taken
1364 steps to address this through explicit design choices and indicator usage anal-
1365 ysis. As a result, there are several research avenues for future work. For exam-
1366 ple, one possible direction is the incorporation of more advanced sentiment
1367 analysis techniques. While we employed established and interpretable tools
1368 in this study, more sophisticated approaches – including neuromorphic mod-
1369 els (e.g., spiking neural networks) and state-of-the-art pre-trained language
1370 models such as BERT [58] or VADER – may offer richer and more human-
1371 like sentiment interpretation. Integrating these methods into our framework
1372 could potentially improve the expressiveness and nuance of the sentiment
1373 signals used in trading decisions. Furthermore, we wish to explore the use of

1374 alternative sentiment data sources, such as specialist news sentiment APIs.
1375 While our current approach relies on Google News for their broad cover-
1376 age and consistency across companies, incorporating more domain-specific
1377 or real-time sentiment sources may further enhance the model’s effectiveness
1378 and robustness, as well as adjusting our current daily trading framework to
1379 high-frequency intraday time frames. Moreover, incorporating explicit mod-
1380 elling of market microstructure effects, including liquidity constraints and
1381 dynamic adaptation of strategies over time, could further validate robust-
1382 ness in realistic deployment scenarios. As the current study is deliberately
1383 designed at the single-stock level to isolate and evaluate the effect of inte-
1384 grating FA, SA, and TA on individual strategy performance we plan to ex-
1385 tend this to a portfolio-level analysis. To further strengthen model robustness,
1386 future work could explore the use of ensemble techniques, such as combin-
1387 ing multiple top-performing GP individuals from different runs. This may
1388 help reduce variance in outcomes, albeit at the cost of model simplicity. An-
1389 other potential enhancement is the application of time-based cross-validation
1390 strategies, such as rolling window evaluation, to assess generalisation across
1391 different market regimes. While these approaches were beyond the scope of
1392 the present work due to computational complexity, they represent promising
1393 directions to improve stability and reliability in real-world trading scenarios.

1394 Another limitation of our work is that our current methodology relies on
1395 a single, chronologically ordered train/validation/test split. This ordering
1396 prevents the model from training on future information, but it leaves two
1397 distinct issues unaddressed. First, a single split yields one performance es-
1398 timate rather than a distribution, so it lacks the statistical guarantees that

1399 purged/embargoed cross-validation would provide against overfitting to a
1400 particular split. Second, while we apply Benjamini–Hochberg FDR correc-
1401 tion to control for multiplicity across our pairwise KS tests, this does not
1402 correct for the prior selection of ActTrade as the control on the basis of its
1403 superior average/median performance — a form of selection bias that a fixed
1404 set of pairwise comparisons cannot address, and one that grows more severe
1405 as more strategies are compared. Addressing the first would require combi-
1406 natorial purged cross-validation with train-only preprocessing pipelines; ad-
1407 dressing the second would require selection-aware significance testing such as
1408 Model Confidence Set or Deflated Sharpe Ratio, which are designed specifi-
1409 cally to correct for the bias introduced when a “best” strategy is chosen from
1410 a larger candidate set. We treat both as necessary next steps before claiming
1411 statistically robust outperformance.

1412 Lastly, it is worth noting our proposed algorithm is not limited to equity-
1413 specific features. Hence, it could be applied to other equity types, such as
1414 commodities and crypto currencies, provided that appropriate indicators are
1415 available or defined.

1416 References

- 1417 [1] M. H. Eng, Y. Li, Q.-G. Wang, T. H. Lee, Forecast forex with arti-
1418 ficial neural networks using fundamental data, in: 2008 International
1419 Conference on Information Management, Innovation Management and
1420 Industrial Engineering, Vol. 1, IEEE, 2008, pp. 279–282.
- 1421 [2] M. Abe, H. Nakayama, Deep learning for forecasting stock returns in

- 1422 the cross-section, in: Pacific-Asia Conference on Knowledge Discovery
1423 and Data Mining, Springer, 2018, pp. 273–284.
- 1424 [3] Q. He, Z. Cai, W. Liu, Prediction of listed companies’ revenue based on
1425 model-fused, in: 2019 International Conference on Machine Learning,
1426 Big Data and Business Intelligence (MLBDBI), IEEE, 2019, pp. 257–
1427 260.
- 1428 [4] A. Herawati, A. S. Putra, et al., The influence of fundamental analysis
1429 on stock prices: The case of food and beverage industries, European
1430 Research Studies Journal 21 (3) (2018) 316–326.
- 1431 [5] C. Ajekwe, A. Ibiamke, Financial statement analysis to predict stock
1432 returns of listed consumer goods firms in nigeria, Available at SSRN
1433 3247805 (2018).
- 1434 [6] K. Kohara, T. Ishikawa, Y. Fukuhara, Y. Nakamura, Stock price pre-
1435 diction using prior knowledge and neural networks, Intelligent Systems
1436 in Accounting, Finance & Management 6 (1) (1997) 11–22.
- 1437 [7] X. Ding, Y. Zhang, T. Liu, J. Duan, Deep learning for event-driven stock
1438 prediction, in: Twenty-fourth international joint conference on artificial
1439 intelligence, 2015.
- 1440 [8] B. Xie, R. Passonneau, L. Wu, G. G. Creamer, Semantic frames to pre-
1441 dict stock price movement, in: Proceedings of the 51st annual meeting
1442 of the association for computational linguistics, 2013, pp. 873–883.
- 1443 [9] M.-Y. Day, C.-C. Lee, Deep learning for financial sentiment analysis

- 1444 on finance news providers, in: 2016 IEEE/ACM International Confer-
 1445 ence on Advances in Social Networks Analysis and Mining (ASONAM),
 1446 IEEE, 2016, pp. 1127–1134.
- 1447 [10] R. Gupta, M. Chen, Sentiment analysis for stock price prediction, in:
 1448 2020 IEEE conference on multimedia information processing and re-
 1449 trieval (MIPR), IEEE, 2020, pp. 213–218.
- 1450 [11] C. Qian, N. Mathur, N. H. Zakaria, R. Arora, V. Gupta, M. Ali, Under-
 1451 standing public opinions on social media for financial sentiment analysis
 1452 using ai-based techniques, *Information Processing & Management* 59 (6)
 1453 (2022) 103098.
- 1454 [12] H. Yun, G. Sim, J. Seok, Stock prices prediction using the title of news-
 1455 paper articles with korean natural language processing, in: 2019 inter-
 1456 national conference on artificial intelligence in information and commu-
 1457 nication (ICAIIC), IEEE, 2019, pp. 019–021.
- 1458 [13] T. Marty, B. Vanstone, T. Hahn, News media analytics in finance: a
 1459 survey, *Accounting & Finance* 60 (2) (2020) 1385–1434.
- 1460 [14] B. Hasselgren, C. Chrysoulas, N. Pitropakis, W. J. Buchanan, Using
 1461 social media & sentiment analysis to make investment decisions, *Future*
 1462 *Internet* 15 (1) (2023) 5.
- 1463 [15] M. Costola, O. Hinz, M. Nofer, L. Pelizzon, Machine learning senti-
 1464 ment analysis, covid-19 news and stock market reactions, *Research in*
 1465 *International Business and Finance* (2023) 101881.

- 1466 [16] R. Hochreiter, Computing trading strategies based on financial senti-
1467 ment data using evolutionary optimization, in: Mendel 2015: Recent
1468 Advances in Soft Computing, Springer, 2015, pp. 181–191.
- 1469 [17] S. Y. Yang, S. Y. K. Mo, A. Liu, A. A. Kirilenko, Genetic programming
1470 optimization for a sentiment feedback strength based trading strategy,
1471 Neurocomputing 264 (2017) 29–41.
- 1472 [18] E. Christodoulaki, M. Kampouridis, P. Kanellopoulos, Technical and
1473 sentiment analysis in financial forecasting with genetic programming,
1474 in: IEEE Symposium on Computational Intelligence for Financial En-
1475 gineering & Economics (CIFEr), 2022.
- 1476 [19] M. M. Mostafa, Forecasting stock exchange movements using neural
1477 networks: Empirical evidence from kuwait, Expert Systems with Appli-
1478 cations 37 (9) (2010) 6302–6309.
- 1479 [20] D. M. Nelson, A. C. Pereira, R. A. de Oliveira, Stock market’s price
1480 movement prediction with long short-term memory neural networks, in:
1481 2017 International joint conference on neural networks (IJCNN), IEEE,
1482 2017, pp. 1419–1426.
- 1483 [21] A. F. Kamara, E. Chen, Z. Pan, An ensemble of a boosted hybrid of
1484 deep learning models and technical analysis for forecasting stock prices,
1485 Information Sciences 594 (2022) 1–19.
- 1486 [22] J. Li, E. P. Tsang, Improving technical analysis predictions: An applica-
1487 tion of genetic programming., in: flairs Conference, 1999, pp. 108–112.

- 1488 [23] M. Kampouridis, E. Tsang, Investment opportunities forecasting: Ex-
1489 tending the grammar of a genetic programming based tool, *International*
1490 *Journal of Computational Intelligence Systems* 5 (3) (2012) 530–541.
- 1491 [24] M. Kampouridis, A. Alsheddy, E. Tsang, On the investigation of hyper-
1492 heuristics on a financial forecasting problem, *Annals of Mathematics*
1493 *and Artificial Intelligence* 68 (2013) 225–246.
- 1494 [25] M. Kampouridis, F. E. Otero, Heuristic procedures for improving the
1495 predictability of a genetic programming financial forecasting algorithm,
1496 *Soft Computing* 21 (2) (2017) 295–310.
- 1497 [26] J. L. Bettman, S. J. Sault, E. L. Schultz, Fundamental and technical
1498 analysis: substitutes or complements?, *Accounting & Finance* 49 (1)
1499 (2009) 21–36.
- 1500 [27] E. Beyaz, F. Tekiner, X.-j. Zeng, J. Keane, Comparing technical
1501 and fundamental indicators in stock price forecasting, in: 2018 IEEE
1502 20th International Conference on High Performance Computing and
1503 Communications; IEEE 16th International Conference on Smart City;
1504 IEEE 4th International Conference on Data Science and Systems
1505 (HPCC/SmartCity/DSS), IEEE, 2018, pp. 1607–1613.
- 1506 [28] A. Pornwattanaichai, S. Maneeroj, S. Boonsiri, Bertforex: Cascading
1507 model for forex market forecasting using fundamental and technical in-
1508 dicator data based on bert, *IEEE Access* 10 (2022) 23425–23437.
- 1509 [29] M. Azimi, A. Agrawal, Is positive sentiment in corporate annual reports

- 1510 informative? evidence from deep learning, Evidence from Deep Learning
1511 (July 2019) (2019).
- 1512 [30] M. R. Vargas, B. S. De Lima, A. G. Evsukoff, Deep learning for stock
1513 market prediction from financial news articles, in: 2017 IEEE Interna-
1514 tional Conference on Computational Intelligence and Virtual Environ-
1515 ments for Measurement Systems and Applications (CIVEMSA), IEEE,
1516 2017, pp. 60–65.
- 1517 [31] A. Nan, A. Perumal, O. R. Zaiane, Sentiment and knowledge based
1518 algorithmic trading with deep reinforcement learning, arXiv preprint
1519 arXiv:2001.09403 (2020).
- 1520 [32] E. Christodoulaki, M. Kampouridis, Combining technical and sentiment
1521 analysis under a genetic programming algorithm, in: UK Workshop of
1522 Computational Intelligence (UKCI), 2022.
- 1523 [33] E. Christodoulaki, M. J. Kampouridis, Using strongly typed genetic pro-
1524 gramming to combine technical and sentiment analysis for algorithmic
1525 trading, in: 2022 IEEE Congress on Evolutionary Computation (CEC),
1526 2022, pp. 1–8.
- 1527 [34] B. Weng, M. A. Ahmed, F. M. Megahed, Stock market one-day ahead
1528 movement prediction using disparate data sources, Expert Systems with
1529 Applications 79 (2017) 153–163.
- 1530 [35] A. Picasso, S. Merello, Y. Ma, L. Oneto, E. Cambria, Technical analysis
1531 and sentiment embeddings for market trend prediction, Expert Systems
1532 with Applications 135 (2019) 60–70.

- 1533 [36] J. Min, Financial market trend forecasting and performance analysis
1534 using long short-term memory, arXiv preprint arXiv:2004.01502 (2020).
- 1535 [37] G. Mu, N. Gao, Y. Wang, L. Dai, A stock price prediction model based
1536 on investor sentiment and optimized deep learning, Ieee Access 11 (2023)
1537 51353–51367.
- 1538 [38] J.-Y. Huang, C.-L. Tung, W.-Z. Lin, Using social network sentiment
1539 analysis and genetic algorithm to improve the stock prediction accuracy
1540 of the deep learning-based approach, International Journal of Compu-
1541 tational Intelligence Systems 16 (1) (2023) 93.
- 1542 [39] A. L. Awad, S. M. Elkaffas, M. W. Fakh, Stock market prediction using
1543 deep reinforcement learning, Applied System Innovation 6 (6) (2023)
1544 106.
- 1545 [40] G.-M. Chatziloizos, D. Gunopulos, K. Konstantinou, Deep learning for
1546 stock market prediction using sentiment and technical analysis, SN Com-
1547 puter Science 5 (5) (2024) 446.
- 1548 [41] E. Christodoulaki, M. Kampouridis, M. Kyropoulou, A novel strongly-
1549 typed genetic programming algorithm for combining sentiment and tech-
1550 nical analysis for algorithmic trading, Knowledge-Based Systems (2025)
1551 113054.
- 1552 [42] E. Christodoulaki, M. Kampouridis, Fundamental, technical and sen-
1553 timent analysis for algorithmic trading with genetic programming, in:
1554 2023 IEEE Symposium Series on Computational Intelligence (SSCI),
1555 IEEE, 2023, pp. 83–89.

- 1556 [43] F. Dakalbab, A. Kumar, M. A. Talib, Q. Nasir, Advancing forex predic-
1557 tion through multimodal text-driven model and attention mechanisms,
1558 Intelligent Systems with Applications 26 (2025) 200518.
- 1559 [44] P. Gangwani, V. Panthi, Leveraging multimodal data and deep learning
1560 for enhanced stock market prediction, AI-Based Advanced Optimization
1561 Techniques for Edge Computing (2025) 93–127.
- 1562 [45] M. A. Arauco Ballesteros, E. A. Martínez Miranda, Stock market fore-
1563 casting using a neural network through fundamental indicators, techni-
1564 cal indicators and market sentiment analysis, Computational Economics
1565 66 (2) (2025) 1715–1745.
- 1566 [46] A. Damodaran, The little book of valuation: how to value a company,
1567 pick a stock and profit, Vol. 34, John Wiley & Sons, 2011.
- 1568 [47] J. Deveikyte, H. Geman, C. Piccari, A. Provetti, A sentiment analysis
1569 approach to the prediction of market volatility, Frontiers in Artificial
1570 Intelligence 5 (2022) 836809.
- 1571 [48] M. S. Azuero Melo, Topic modeling and sentiment analysis for predict-
1572 ing the direction of movement of the s&p 500 index based on financial
1573 news/author miguel santiago azuero melo, bsc (2024).
- 1574 [49] A. Brabazon, M. Kampouridis, M. O’Neill, Applications of genetic pro-
1575 gramming to finance and economics: past, present, future, Genetic Pro-
1576 gramming and Evolvable Machines 21 (1) (2020) 33–53.
- 1577 [50] P. Fernandez, E. de Apellániz, J. F Acín, Survey: Market risk premium
1578 and risk-free rate used for 81 countries in 2020 (2020).

- 1579 [51] X. Long, M. Kampouridis, P. Kanellopoulos, Genetic programming for
1580 combining directional changes indicators in international stock markets,
1581 in: International Conference on Parallel Problem Solving from Nature,
1582 Springer, 2022, pp. 33–47.
- 1583 [52] S. Abolmakarem, F. Abdi, K. Khalili-Damghani, H. Didehkhani, A
1584 multi-stage machine learning approach for stock price prediction: Engi-
1585 neered and derivative indices, *Intelligent Systems with Applications* 24
1586 (2024) 200449.
- 1587 [53] R. Menoita, S. Silva, Evolving financial trading strategies with vectorial
1588 genetic programming, in: *Proceedings of the Genetic and Evolutionary*
1589 *Computation Conference Companion*, 2025, pp. 875–878.
- 1590 [54] W. F. Sharpe, The Sharpe ratio, *Streetwise—the Best of the Journal of*
1591 *Portfolio Management* 3 (1998) 169–185.
- 1592 [55] B. Bieganski, R. Ślepaczuk, Supervised autoencoder mlp for financial
1593 time series forecasting, *Journal of Big Data* 12 (1) (2025) 207.
- 1594 [56] M. Zanotti, Do global forecasting models require frequent retraining?,
1595 arXiv preprint arXiv:2505.00356 (2025).
- 1596 [57] J. Brookhouse, F. E. Otero, M. Kampouridis, Working with opencl to
1597 speed up a genetic programming financial forecasting algorithm: Initial
1598 results., in: *Proceedings of the Companion Publication of the 2014 An-*
1599 *annual Conference on Genetic and Evolutionary Computation*, 2014, pp.
1600 1117–1124.

[58] J. Devlin, M.-W. Chang, K. Lee, K. Toutanova, Bert: Pre-training of deep bidirectional transformers for language understanding, arXiv preprint arXiv:1810.04805 (2018).

Appendix A. Hyperparameter search spaces

Table A.24 reports the search space explored during the two-step grid search described in Section 4.3 for the GP-based algorithms. Table A.25 reports the corresponding search space for the machine learning benchmarks (MLP, SVM, XGBoost, LSTM). In both cases, the final selected values used in the reported experiments are given in Table 5 (GP) and Section 4.3 (ML benchmarks); the ranges below indicate the full space considered during tuning, not only the selected configuration.

Table A.24 Hyperparameter search space for the GP algorithms.

Parameter	Search space
Population size	[100, 300, 500, 1000, 2000]
Crossover probability (p_c)	[0.7, 0.9, 0.95]
Number of generations	[20, 30, 50]
Tournament size	[2 – 5]
Maximum tree depth	[4 – 7]
Tournament probability (p_1)	0.3, 0.4, 0.5
Active-trading offspring probabilities (p_2, p_3, p_4)	(0.10, 0.15, 0.20)

Table A.25 Hyperparameter search space for the machine learning benchmarks.

Algorithm	Parameter	Search space
MLP	Hidden layer size(s)	[1–4 layers]
	Learning rate	[0.0001–0.01]
	Activation function	[ReLU, tanh, logistic]
SVM	Kernel	[linear, RBF, polynomial]
	C (regularisation)	[0.01–100]
	γ (RBF kernel)	[0.0001–1, ‘auto’]
XGBoost	Max tree depth	[3–10]
	Learning rate	[0.01–0.3]
	Number of estimators	[50–500]
LSTM	Layers / units	[1–4 layers]
	Learning rate	[0.0001–0.01]
	Batch size	[32–128]