Fundamental filters and technical genetic algorithms for stock investing

Marco Montez, Rui Neves
marco.montez@ist.utl.pt

Instituto Superior Técnico, Lisboa, Portugal

October 2019

Abstract

The system developed invested in S&P 500 securities employing an hybrid strategy consisting of fundamental and technical analysis. Volatility and fundamental factor filtering were implemented to choose the portfolio stocks along with a technical strategy optimized with a genetic algorithm. Fundamental growth filters with spending and earnings factors showed good performance. This was further improved by choosing only the middle volatility quintiles of companies. Overall the fundamental strategy revealed good potential returns (216% total ROI, or 13,6% annually, with a sharpe ratio of 2.64) in the period from 2011 to 2018, adding the GA resulted in a slightly lower performance of (164% total ROI, or 11,4% annually, sharpe ratio of 2.48), both with substantially higher returns than the S&P 500 index (95% total ROI, 7,7% annually, sharpe ratio of 1.81).
Keywords: Genetic Algorithm, Stock Market, Portfolio, Fundamental Analysis, Technical Analysis

1. Introduction

To invest in the market different portfolios are constructed based on each investor's needs and risk tolerance. Generally lower risk, more stable portfolios have lower returns and higher risk more volatile ones have higher returns (or losses).

Historically this job required analysing technical indicators and trends[23],[28], as well as reading countless financial statements and analyzing large quantities of information performed by human investors [12],[14],[24],[25],[8],[4]. However with the advent of computers increasingly more powerful capabilities and the ever-growing amount of data available, computer models started being used to understand and predict financial trends. To the point that, nowadays, several institutions and funds rely profoundly on these models to construe their portfolios [6],[18],[26],[11]. Solving this portfolio optimization problem with deterministic techniques is practically impossible, which is why researchers developed meta heuristics like Simulated Annealing (SA), Particle Swarm Optimization (PSO), Ant Colony Optimization (ACO), and Genetic Algorithms (GA) to solve them. This class of algorithms, with special merit to genetic algorithms, is at the forefront of machine learning and soft computing areas due to their ability of handling large complex systems and are hence worth exploring [21],[13],[20].

The problem therefore is how to create a profitable strategy to obtain a constant positive return on the stock market while ensuring it is robust enough to handle different market conditions and therefore minimizes risk.

The main goals of this work are to achieve a 5% increase in yearly returns when compared to the S&P 500 index, while maintaining equal or lower risk. And the study of the relations between future companies returns and volatility/fundamental factors when filtered by growth and value filters, in search of profitable and robust predictive strength.

The chosen approach was to implement a hybrid genetic algorithm that combines fundamental analysis of the financial statements, as well as volatility filtering, to identify the most promising companies and technical analysis to avoid buying in downtrends and optimize mid to long-term returns.

The **main contributions** of this work are:

- Creating a portfolio of stocks based on volatility and fundamental growth filters such as filtering the companies that have consistently grown their R&D expense and their cash flow in the previous three years.
- Dynamically adjusted periods for technical indicators and stop losses to deal with changing market conditions.
- Employing a hyper mutation mechanism when the population diversity drops low in order to

motivate exploration of other local maximum and prevent premature convergence.

The structure of this document is described below: Chapter 2 explores the literature review and state of the art approaches on the subject of stock investing and portfolio optimization, including an historical overview on this problem and an analysis of the existing algorithms and current paradigms. Chapter 3 describes the system architecture of the designed algorithmic solution along with its constraints and validation metrics. Chapter 4 presents the obtained results and accompanying case studies. And chapter 5 summarizes the main conclusions, the current limitations and the proposed future work.

2. Background

The state of the art solutions for portfolio optimization problems are often found in the field of soft computation. To give evidence as to why [22] zadeh says that *the point of departure in soft computing is the thesis that precision and certainty carry a cost and that computation, reasoning, and decision making should exploit, wherever possible, the tolerance for imprecision and uncertainty.* Meaning that soft computation is expected to be relevant in areas where exact solution are too expensive, not practical or do not yet exist.

2.1. Existing solutions

Such is the area of portfolio optimization with potentially hundreds to thousands of stocks and respective indicators to analyze, resulting in a very vast search landscape where soft computing algorithms like GA, PSO and simulated annealing among others excel.

Genetic algorithms can be used to chose the weights used in the portfolio simulator and trading indicators. GA algorithms can find robust solutions to the efficient frontier portfolio problem with several types of risk as mentioned in [5] that uses mean/semi variance and variance with skewness as risk measures.

Fu in 2013 [7] tackles this problem and resorts to traditional and hierarchical GA's, he finds that the hierarchical GA finds a less risky investment strategy than the Buy-and-hold strategy but with less returns, particularly in bullish market. Aleen in 1999 [1] used genetic algorithms on trees of trading rules on the testing period starting in 1977 until 1995. Their algorithm had a slightly inferior return than the buy-and-hold strategy, claiming that transaction cost have a significant effect on the returns.

Baúto [3] employs GA with SAX pattern matching and parallel GPU optimization, reaching a speed up of 30 to 180 times and ROI average returns of 70% to 100% over the 12 years. Lin [16] uses GA in portfolio optimization and attest its efficiency. They show its possible to achieve the same efficient frontier with less assets, reducing assets, from a number of 200 to 40, thus reducing computing time.

MOEA are able to create efficient portfolios by optimizing simultaneously risk and return. Silva [27] utilized a multi-objective evolutionary algorithm (MOEA) with an hybrid approach for portfolio investing, using both fundamental indicators and technical ratios. The resulting efficient frontier is slightly above the benchmark. Anagnostopoulos in 2010 [2] proposed a triple objective algorithm, optimizing for return, risk and number of stocks. Several GA variants such as NSGA-II, PESA and SPEA2 were used, the last one yielding the best results.

Almeida [10] explored forex investing. A SVM is used to classify the market trend into either up,down or sideways and trains a GA for each situation. The algorithm then uses the investing algorithm concurrent with the current classified trend. They get returns in the order of 40% annually.

Meghwani [19] also employed a tri-objective portfolio optimization with risk, return and transaction costs as the objetives. They also explore numerous pratical constraints like cardinality, self-financing, quantity, pre-assignment and other cost related constraints. Three known risk measures are used, variance, Value-at-Risk(VaR) and Conditional-Value-at-Risk (CVaR). Several algorithms are employed namely NSGA-II, GWASFGA and MOEA/D. The NSGA-II and GWASFGA outperformed the MOEA/D and had similar results among them, with the GWASFGA having lower overall transaction costs.

Particle swarm optimization is a meta heuristic with some similarities to genetic algorithms, where each particle has a position and velocity and is influenced by both its best position and the swarms best position. Dallagnol [9] compared the performance between PSO and GAs, finding that both consistently converge to the optimal solution in 95% of the tested cases. PSO has a faster convergence than GA, both in terms of iterations and running time. However the PSO is more easily trapped in local minima and is highly affected by the initial particles position. Loraschi [17] used a distributed GA, where small segments of populations migrate to neighbouring populations, allowing for a bigger genetic diversity, that might explain for the increased accuracy and speed of the distributed GA. Also it enables the use of parallel computing and potential large parallel clusters since this class of meta heuristic problems is usually computationally intensive.

Huang [15] used an hybrid approach, combining GA for feature selection and parameter optimization and SVR to simulate future stock returns and pick the most promising ones. Feature selection had a substantially bigger impact on the results than parameter optimization. The benchmark had total returns of 170% over 14 years, while the 30/20/10 stocks portfolio had returns of 1300, 2000 and 3700 respectively, corresponding to an average annualized return of 67%, 72% and 80% versus the 44% in the benchmark.

3. Solution Architecture

The goal of this system is to create an evolving investment strategy capable of managing a portfolio with several stocks and achieving a good ROI. It employs a computational module that decides which company to invest based on a fundamental stock screener and a technical genetic algorithm.

3.1. System layout

The system first receives the user and financial data, such as financial statements and pricing information. The data enters the pre-processing module where it is cleaned and normalized and the fundamental ratios are calculated.

Next, in the computation module the stock screener creates a portfolio of stocks based on their volatility and fundamental attributes, this portfolio is sent to the GA that implements the trading strategy and informs when to enter and exit the market.

The GA begins by generating a random population where each individual is then classified by the portfolio simulator that uses the technical ratios and the individual chromosomes to rank and buy the most promising companies, using indicators such as aroon and EMA slope. The best ranked individuals (higher ROI) are selected, then recombined using crossover and mutation, replacing the old population. This cycle continues until the end of the simulation. The resulting chromosomes are then tested in the testing data set using a sliding window, three years for training and one year for testing, the results are sent to the system validation module that employs several metrics to perform an in-depth analysis of the individual's strategies.

This system is divided in four high level modules: The first module is the **input module** that receives data and user input. The data input consists of fundamental information such as earnings per share, revenue margin and long-term debt, along others and pricing information such as price and volume traded. User input controls certain decisions such as the initial budget used, the ratios chosen, the fitness function metric, the model constraints, etc.

The second module is the **data pre-processing module** whose job is to perform operations to make the data usable. It cleans the dataset by handling missing or wrong data and afterwards calculates the fundamental and technical ratios that will be used in the computational module.

Then enters the **computation module**, which contains the stock screener, technical GA, portfolio simulator and technical ratio manager . Initially the stock screener chooses the stocks to constitute the portfolio and then the GA trains the trading strategy using the portfolio simulator ROI as its fitness function, that simulates buying and selling stocks and evaluates the performance of each individual's investing strategy. The portfolio simulator implements the trading strategy of a given chromosome and the technical ratio manager supplies it with the needed technical indicators.

The fourth module is the **system validation module** that includes two sub-modules: The testing manager that runs the individuals strategies in the training set, taking special attention not to mix the testing and training sets. And the results presenter that shows the end metrics and graphs in a graphical user interface.

3.2. Stock Screener

The stock screener goal is to build a portfolio with the companies displaying the best fundamentals. It first selects the companies that were in the SP 500 that year in order to avoid survivorship bias, where only companies that survived until today are analyzed, resulting in a skewed sample. Then the companies are filtered by volatility and assigned to 5 quintiles, each one has 20% of the companies, the first quantile has the first 20% and the least volatile companies, the second the following 20% companies, the fifth quintile has last 20% and thus the more volatile companies.

After filtering by volatility we filter the companies by fundamental performance, such as only accepting companies that have grown their revenue for the past 3 years (revenue growth filter). Figure 1 shows the stock screener flow diagram.

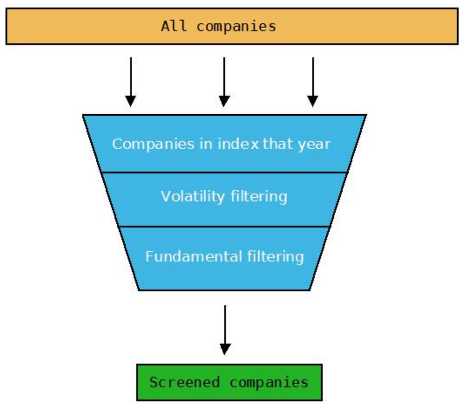

Figure 1: Stock Screener flow diagram

The chosen filters for the stock screener have periods between 1 to 2 years, since this range gave the best results. Filters with more years are more strict and select fewer companies. Longer filters With 3 or more years of data have a very small number of companies that pass the filtering. This results in portfolios with few companies which brings higher risk. And, even the good companies generally have a bad year occasionally. These longer filters would be too strict.

Two kinds of filters were used: value and growth filters. Value filters accept companies that had a positive value for that ratio in the previous 3 years, such as positive net income. Growth filter accept companies that had a positive constant growth the previous 3 years. If the company in one or more of the three years had negative growth it would be rejected by the growth filter and analogously for the value filter.

The advantage of this filtering system is that several filters can be used simultaneously to search for certain company profiles and define specific strategies for each. For example to create a portfolio of growth companies revenue, cashflow and ebitda growth filters can be used so that only companies with growing earning potential are seletect. Or if instead a value portfolio is of interested, then maybe an approach focusing on a decreasing book-value ratio and low-debt could be used.

These single filters were combined into 4 fundamental strategies: Increased Earnings, Increased Spending, Spending and cash, Piotroski F-score.

Which were then combined into one composite strategy, that chose what strategy to utilize that year based on the strategy with the best performance the previous year, Figure 2 shows this process. This is implemented and not a fixed strategy, so that if market conditions change the implemented strategy changes with it.

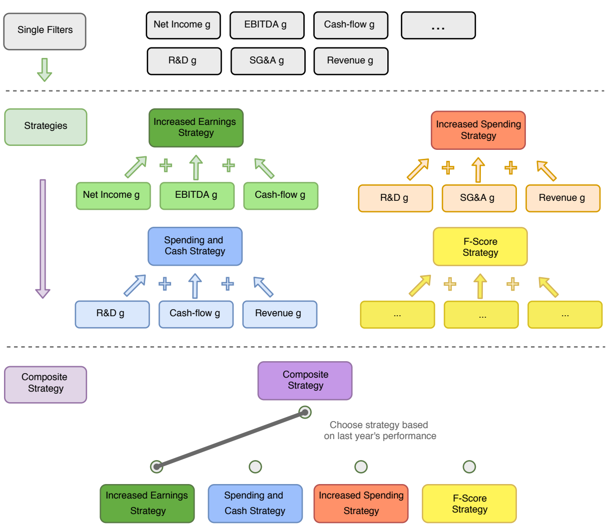

Figure 2: Filtering strategies flow diagram

3.3. Genetical Algorithm

The genetic simulator (Figure 3) is the main computation module. It runs the genetic algorithm for N epochs and saves the best individuals. The process starts with the creation of a random chosen population so as to have initial diversity in its gene pool. This population becomes the current population of the algorithm. It is then filtered through the portfolio simulator, that acts has a fitness function and picks the fittest individuals, that is, the ones with highest ROI. The best individuals are combined and create offspring through crossover that transfers a random combination of the parents genes to the children, and mutation, that slightly changes the value of some genes. The best individuals and the now created offspring constitute the new population that replaces the old current population, completing the evolution cycle.

This cycle is repeated for N epochs, or iterations of the algorithm. At each epoch the population adapts and in average keeps increasing its fitness. After N epochs the algorithm completes its execution and returns the best individual for that run.

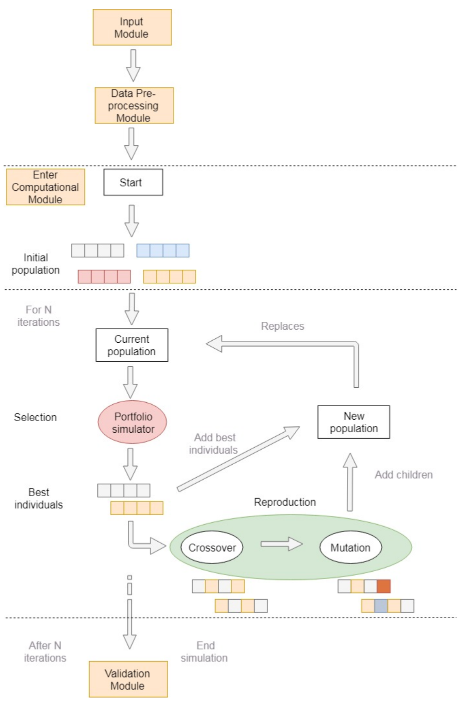

Figure 3: Genetic simulator flowchart

This class of algorithms can be seen as exploring a fitness landscape trying to find the highest peaks. So in essence this is a search optimization problem with the goal of finding the global maximum. Due to the nature of the algorithm it doesn't guarantee to find a global maximum and it generally finds only the local maximum.

Therefore, in order for the algorithm to work efficiently and find a "high" local maximum in a practical time-frame, its important to correctly set its parameters and implementation, with some of key ones being: the population size, number of total epochs, fitness function metrics and mutation rate. A high population size results in high initial and ongoing gene diversity, a good thing, but it slows down the execution since at each epoch there are more individuals to analyze with the fitness function. The chosen population size was of 50 elements, since higher numbers did not speed the convergence or led to better solutions. Similarly a superior number of iterations results in a longer execution time but a higher population fitness level, since the fitness of the population generally keeps increasing per iteration. However, after a sufficiently large number of iterations the algorithm reaches the peak of the local maximum and further iterations will not significantly increase the results, for this experience a number of 25 epochs was found to be optimal, after which the algorithm converged.

The choice of the fitness function is critical since it is what defines the goal to maximize. In this case, maximizing the return on investment. The mutation rate should be high enough to add variety to the gene pool, but not so much that it randomizes the chromosomes and the genetic inheritance from the parents is lost. The chosen mutation rate was 5% and the type is a Gaussian mutation with a sigma or average value of 0,5. So that in average a gene changes 50% of its value either a positive or negative change. Also a minimum step for mutations of 0,05 was used so that negligible changes that had no real effect were substituted by a considerable effect. The crossover was implemented as a uniform crossover in order to give equal weight to each gene and distribute them among the offspring without bias. The probability for crossover is 90%. The selection mechanism uses a round robin tournament of size 2, where 2 individuals are selected randomly from the population and the one with highest fitness score is included in the parents group for the next generation. The survivor selection is generational in that the old population is replaced by the new population that consists of 80% offspring and 20% elites from the previous population. The elites are the best 20% individuals from the last population, and are used so that the most profitable strategies are always remembered. Each individual gene was encoded as floating point number, with the exception of indicator periods that were encoded as integers.

Also some experiments were made with hyper mutation to hopefully reduce premature convergence and bring more exploration. When a hyper mutation occurred the probability for mutation was 100% and the sigma equal to 2. The hyper mutation was triggered when the average population fitness was very similar to the best individual fitness, this meant that the algorithm had converged to a single peak and wasn't making significant progress, so in order to motivate the algorithm to explore different areas a high degree of mutation was employed for a single epoch to add diversity to the gene pool. Whenever the average population fitness was equal or greater than 90% of the best individual fitness hyper mutation was triggered.

Table 1 shows a synopsis of the particular implementations used for the GA operations and Table 2 displays the parameters used.

Table 1: Operations used in GA

Operation	**Implementation**
Mutation	Gaussian
Crossover	Uniform
Selection	Tournament and Elitism
Survivor Selection	Generational
Representation	Floating Point and Integers
Fitness Function	ROI

3.3.1 Technical Indicators

The strategy module uses two types of indicators: Entry and money management indicators, further discriminated below:

Entry:

- Arron up and down
- CMF - Chaickin money flow
- CMO - Change momentum oscillator
- MFI - Money flow index
- SSL - Gan hilo activator
- EMA Slope

Money management:

- ATR - Average true range

The entry indicators decides if this is a good time to enter the market, they check several market conditions such as trade volume, price volatility and previous highs/lows to confirm if this is indeed a favorable timing. The confirmation indicators are

Table 2: GA parameters

Type	Parameter(s)
Population	50
Epochs	25
Mutation	$\sigma = 0.5$, p = 5%
Crossover	p = 90%
Selection	Tournament Rounds of 2
Survivor Selection	80% offspring and 20% elites
Hyper Mutation	$\sigma = 2$, p = 100%
Hyper Mutation Threshold	avg fitness >= 90% best fitness
Minimum Step	0.05

a collection of six weighted indicators normalized so that their sum equals 1. It is considered a buy signal when their weighted sum is higher than 0,7.

The money management indicators, consisting of the stop loss and trailing stop loss are there to limit trading risk, by limiting potential losses and secure winnings respectively. They decide when the stock is sold.

3.3.2 Chromosome structure

The chromosome encodes three types of genes: Genes that encode the periods of some indicators such as the aroon indicator and the SSL indicator. Genes that encode the parameters of the stop loss and trailing stop loss tools. Genes that encode the weights of the confirmation indicators.

Not all indicators have an adjustable period since that would increase the complexity of the variables to optimize in the GA, so only the ones with greater variability due to changes in the period were chosen to be dynamically adjusted.

4. Results

This work utilized a combination of fundamental and technical strategies to invest in the market. The fundamental strategy main goal was to build a portfolio each year with the most promising companies. This was achieved by looking at volatility and fundamental factors, that were then combined into multi-factor strategies and aggregated into one composite strategy. The fundamental factors were calculated in two ways, value factors and growth factors. Value factors look to see if a certain metric such as net income has been consistently above 0 for the past n years, while growth factors check if that metric has been rising or growing for every year the past n years. These single factors were then combined into multi-factor strategies that attempt to capture certain market profiles from the filters used. These strategies were aggregated into one Composite Strategy, that decides what strategy to use the current year, based on last year's performance.

4.1. Volatility filtering

The first step of the fundamental stock screener is, for each year, to choose the companies that where in the S&P index that year (on January 1st) in order to avoid survivorship bias, where only companies that have survived until today are chosen, thereby inflating the results.

The second step is to divide the companies, yearly, into 5 quintiles in order of their volatility, the first quintile (Q1) has 20% ($\frac{1}{5}$) of the companies with the least volatility, while the last quintile has the 20% of companies with the most volatility. This volatility measure is done based on the past year's volatility starting in January and ending in December. The returns are measured as the total ROI of the company on that year.

The volatility filtering was combined with fundamental factors filtering. Figure 4 shows the volatility quintiles ROI after applying a 2 year turnover growth filter.

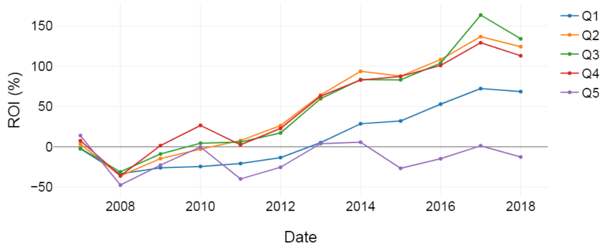

Figure 4: Influence of turnover growth filter on ROI, per volatility quintile

It was observed that in average the three middle quintiles (Q2,Q3,Q4) had superior performance than the extreme quintiles (Q1 and Q5). Therefore in further analysis only the middle quintiles were considered.

4.2. Fundamental filtering

The influence of these factors on the portfolio returns were analyzed and are summarized in Table 3. It applies all the fundamental factors in the periods of 1 to 3 years to see their influence on returns. The growth factors in the factor table end with a 'g' and the value factors end with a 'v' in order to provide a faster distinction. Higher returns are depicted with a darker shade of green. Nan values were given to filters that return a portfolio with less than 10 companies on all years.

Table 3: Fundamental factor influence on ROI

	1	2	3
roa_v	91.9638	94.431	93.2038
roa_g	113.306	149.155	nan
cfroa_v	91.9638	94.431	93.2038
quality_earnings_v	91.9638	94.431	93.2038
gearing_g	69.5947	64.388	nan
working_g	108.256	127.061	nan
shares_issued_gn	103.79	116.464	96.7862
gross_margin_g	119.418	129.183	132.083
turnover_g	116.879	143.42	205.792
book_value_g	86.3744	95.3991	87.4961
revenue_g	116.514	123.799	141.597
net_income_v	91.9638	94.431	93.2038
net_income_g	126.505	140.953	174.43
long_debt_gn	92.6938	118.654	97.3691
total_liabilities_gn	59.9014	nan	nan
cashflow_g	125.615	155.052	159.317
capex_g	107.833	108.37	124.536
rnd_g	188.517	204.223	190.812
sga_g	129.65	141.166	137.338
ebit_g	120.954	135.163	146.715
ebitda_g	127.599	134.956	135.144

4.3. Multi factor strategies

While combining these single filters, 4 main strategies were chosen to represent both the value and growth approaches. For the value approach, the modified Pietrosky F-Score was chosen. For the growth approach the original strategies of Increased Earnings (henceforth IEarnings), Increased Spending (henceforth ISpending) and R&D cash Strategy (henceforth R&DC) were chosen. Table 4 shows the filters used for each strategy and figure 5 depicts the cumulative returns on these four strategies.

When building these multi-filter strategy a selection criteria was applied of a minimum of 10 companies for each year. If a filter returned less than 10 companies it would be discarded, since that would be too specific and bring higher risk into the portfolio. This is the reason why filters of three years were not chosen. Despite showing, in isolation, better performance in some areas such as turnover and net income growth, when they were combined with other filters they failed to satisfy the portfolio minimum of 10 companies.

When analysing the total returns on the strategies, it can be seen that the F-score provides the lowest return (≈170%), the increased earnings provides a reasonable return (≈190%) , the R&D cash and increased spending both provide similar and high returns (≈ 250,270%). The yearly returns are also displayed (Figure 6) since they allow better understanding as to the performance of the strategy each year. It can be seen that all strategies are highly correlated and positive for most years, except of course for the market crash of 2008.

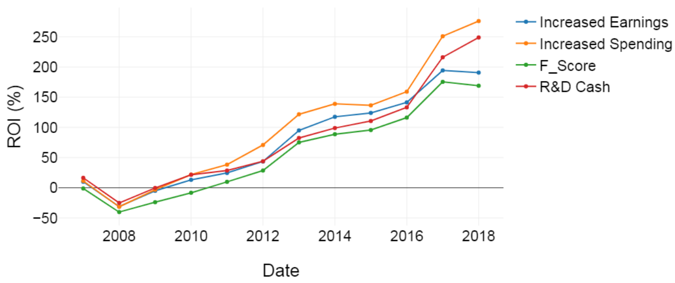

Figure 5: Cumulative returns of the fundamental strategies.

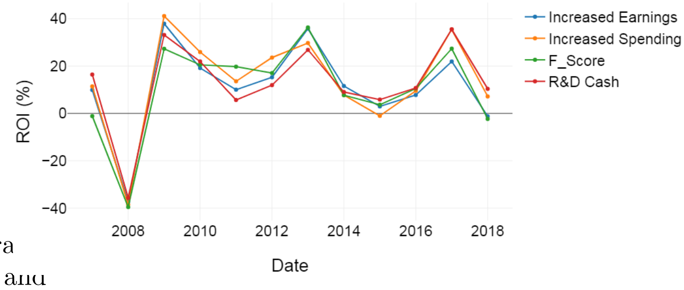

Figure 6: Yearly returns of the fundamental strategies.

4.4. Composite Strategy

In order to decide what strategy to implement in the current year, the previous year ROI is analyzed for each strategy and the one with the highest ROI is chosen to be implemented the following year. This composition of strategies was called the composite strategy.

Figure 9 shows the composite strategy cumulative and yearly returns and Table 5 shows the strategy chosen for each year in the composite strategy, along with their cumulative and yearly ROI. Figure 10 displays the total returns of the Composite, GA and Index. The composite strategy generally

Table 4: Filters composing each strategy. The letters G and V stand for growth and value. The value between parenthesis is the number of trailing years used to calculate the filter.

IEarnings	ISpending	R&DC	F-Score
Net Income G(2)	R&D G(2)	R&D G(2)	ROA V(1)
EBITDA G(2)	Revenue G(2)	Revenue G(2)	ROA G(1)
Cashflow G(2)	SG&A G(2)	Cashflow G(2)	Turnover G(1)
			Working Capital G(1)
			Gross Margin G(1)

provides annual returns higher than the index. In 2008,2011 and 2016 the returns were very similar but only in 2014 did it offer lower profits.

4.5. Hyper Mutation on GA

When developing the GA, it was noted that it tended to get stuck at local optima rather quickly. In order to prevent this and motivate the algorithm to diversify its population and explore the surrounding areas an hypermutation mechanism was implemented.

Figure 7 plots the best individual and average population fitness of each iteration for both the normal GA and the hypermutation GA.

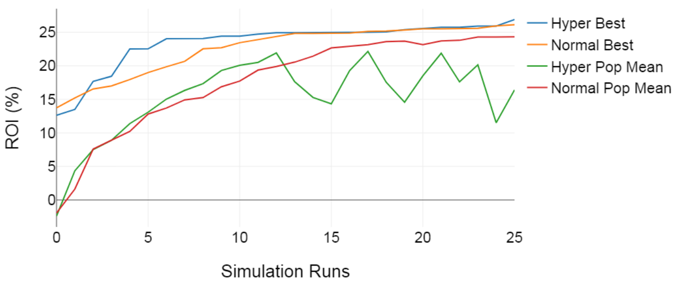

Figure 7: Yearly returns of the fundamental strategies.

The full simulation returns for both approaches were calculated and are presented in Figure 8. The best individual curve is very similar for both GAs and therefore it was concluded that this optimization did not provide a substantial increase in performance and hence was not used.

A possible explanation is that while the hypermutation brought diversity to population it didn't affect the elites which constitute 20% of the population. This created a high selection pressure that essentially rapidly replaced the new diverse chromosomes with the old elites and just after a few iterations population converged to the same local optima.

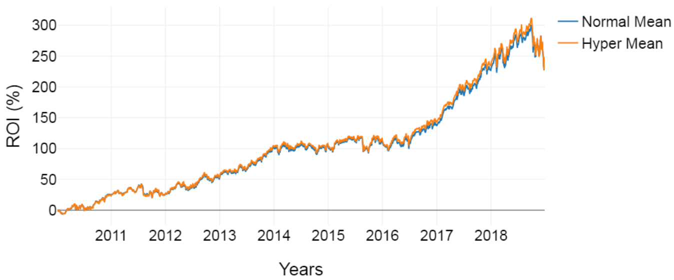

Figure 8: Yearly returns of the fundamental strategies.

Table 5: Strategies chosen each year for the composite strategy along with their yearly and cumulative ROI

Year	Chosen Strategy	Yearly ROI (%)	Cumulative ROI (%)
2008	R&DC	-35.7	-35.7
2009	R&DC	33.3	-14.4
2010	ISpending	25.9	7.8
2011	ISpending	13.5	22.5
2012	F-Score	17.0	43.3
2013	ISpending	29.7	85.9
2014	F-Score	7.7	100.3
2015	IEarnings	3.0	106.2
2016	R&DC	10.7	128.3
2017	R&DC	35.5	209.5
2018	R&DC	10.4	241.6

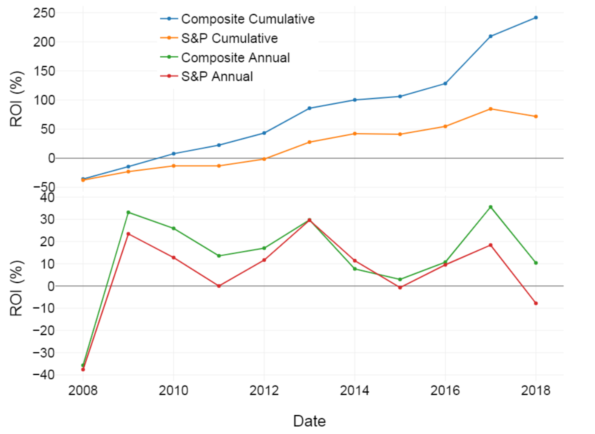

Figure 9: Cumulative and yearly returns of Composite Strategy

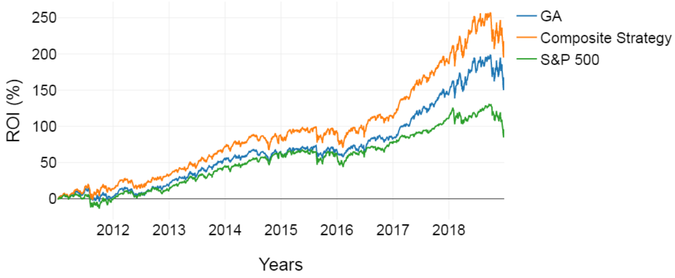

Figure 10: Cumulative Returns of Composite, Average GA and S&P Index

5. Conclusions

The results above show that volatility and certain fundamental metrics, such as earnings and spending metrics have an influence on stock's returns. The fundamental metrics that measure consistent growth were more significant than significant positive value. The most profitable rations were: increasing spending in R&D and increasing cash-flow, net income and turnover growth. The optimum look back period was of 2 years when combining several filters. Smaller lookback period of 1 year didn't perform as well as the 2 year period, probably because they don't have enough information to properly access the company quality. For example, good results could be attributed to ephemerally good market conditions or one good management decision that in that specific time increased profits but does not necessarily mean a steady future growth. Higher lookback filters of 3 plus years end up overfiltering the companies resulting in too few companies to build a portfolio. This problem is further evident in the recession years, where filters of 3 plus years would result in a portfolio of 1-5 companies, which is not diversification enough to make a robust strategy.

When looking just at volatility, little correlation to price changes was seen, however when fundamental factors were combined there appeared the clear pattern of the middle quintiles (Q2,Q3,Q4) having consistently better results than the tail quintiles (Q1 and Q5). The proposed explanation for these effects is that the least volatile quintile (Q1) contains the companies with reduced price movement, meaning it is unlikely a substantial rise will occur, explaining why their profits (and losses) are generally lower. They have less downsize as well as upside. Companies enter high volatility periods (Q5) generally due to increased uncertainty about the future, caused by bad news, unresolved issues (such as lawsuits), or because of excessive speculation. Future uncertainty is normally not favorable to future returns, likewise excessive speculation tends to disturb the normal price discovery mechanism where companies converge to their fundamental value and might result in overvalued companies, which also hurts future returns, since these over bid prices tend to deflate over time. From the period between 2011 and 2018, the Composite strategy provided a substantial return of 216%, when a GA was applied the returns decreased to 164%. Still they both were above the S&P 500 return for that period (95%). The sharpe ratios of the Composite and the GA were 2.64 and 2.48 respectively, also beating the Index which had 1.81 for that period. The GA did not provide better returns nor risk adjusted returns than the Composite, a possible cause is the existence of overfitting to the training set since test scores do not improve as the training progresses. Also it was observed that the GA tended to rapidly converge to local optima, in order to avoid this and promote exploration after a maximum was found, an hyper mutation mechanism was employed . However the hyper mutation was not successful in promoting more exploration for the algorithm, essentially due to 20% of the population being elites that were not affected by the mutation and exerted a high selection pressure on the population after the hyper mutation, thus demotivating and shortening the exploration phase, to the point no significant increases where achieved with the hyper mutation vs the normal one.

5.1. Future Work

This section describes the present limitations of this work, followed by the proposed future improvements.

5.2. Current Limitations

- Overfitting GA parameters to the training set.
- Premature convergence on GA, a higher degree of exploration is needed.

- Use of daily pricing data results in some "slippage" when a stop loss is hit.
- Limited historical data, only 15 years daily data from S&P 500's companies were used.

5.3. Proposed Improvements

- Use shorter testing periods such as 3 month and 6 month windows, since markets change rapidly.
- Reduce the train overfitting by finding other indicators or factors that have higher correlation across different periods
- Try alternative mechanisms to increase GA exploration, such as multi-population approachs
- Implement a GA to optimize the weights of the portfolio's fundamental factors.
- Divide fundamental factors in quintiles and further explore their influence on returns
- Using smaller granularity data such as hourly pricing data.
- Simulate returns distributions based on past data and run Monte Carlo simulations to find better statistically optimized parameters.
- Apply other fundamental factors and metrics
- Explore other markets such as the Euronext to see if the fundamental factors remain significant
- Multi objective optimization with ROI and risk

References

[1] F. Allen and R. Karjalainen. Using genetic algorithms to find technical trading rules. *Journal of Financial Economics*, 1999.

[2] K. Anagnostopoulos and G. Mamanis. A portfolio optimization model with three objectives and discrete variables. *Computers & Operations Research*, 2010.

[3] J. Baúto, A. Canelas, R. Neves, and N. Horta. Parallel sax / ga for financial pattern matching using nvidia ' s gpu. *Expert Systems With Applications*, 2018.

[4] M. Buffett. *Warren Buffett and the Interpretation of Financial Statements: The Search for the Company with a Durable Competitive Advantage*. Scribner, 1^{nd} edition, 2008. ISBN:978-1416573180.

[5] T.-J. Chang, S.-C. Yang, and K.-J. Chang. Expert systems with applications portfolio optimization problems in different risk measures using genetic algorithm. *Expert Systems with Applications*, 2009.

[6] H. Y. Chen, C.-F. Lee, and W. K. Shih. Technical, fundamental, and combined information for separating winners from losers. *Pacific Basin Finance Journal*, 2016.

[7] T. chung Fu, C. pang Chung, and F. lai Chung. Adopting genetic algorithms for technical analysis and portfolio management. *Computers and Mathematics with Applications*, 2003.

[8] L. A. Cunnningham. *The Essays of Warren Buffet: Lessons for Corporate America.* The Cunningham Group & Carolina Academic Press, 4^{nd} edition, 2015. ISBN:978-1611637588.

[9] V. A. F. Dallagnol, J. van den Berg, and L. Mous. Portfolio management using value at risk: A comparison between genetic algorithms and particle swarm optimization. *International Journal of Intelligent Systems*, 2009.

[10] B. J. de Almeida, R. F. Neves, and N. Horta. Combining support vector machine with genetic algorithms to optimize investments in forex markets with high leverage. *Applied Soft Computing*, 2018.

[11] E. F. Fama. Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, 1970.

[12] P. A. Fisher. *Common Stocks and Uncommon Profits and Other Writings*. Wiley, 2^{nd} edition, 2003. ISBN:978-0471445500.

[13] D. E. Goldberg. *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison-Wesley Professional, 1^{nd} edition, 1989. ISBN:978-0201157673.

[14] B. Graham and D. L. Dodd. *Security Analysis*. McGraw-Hill, 3^{nd} edition, 1934. ISBN:0-07-144820-9.

[15] C.-F. Huang. A hybrid stock selection model using genetic algorithms and support vector regression. *Applied Soft Computing*, 2012.

[16] C.-C. Lin and Y.-T. Liu. Genetic algorithms for portfolio selection problems with minimum transaction lots. *European Journal of Operational Research*, 2008.

[17] A. Loraschi, M. Tomassini, A. Tettamanzi, and P. Verda. Distributed genetic algorithms with an application to portfolio selection problems. *Artificial Neural Nets and Genetic Algorithms*, 1995.

[18] H. Markowitz. Portfolio selection harry. *The Journal of Finance*, 1952.

[19] S. S. Meghwani and M. Thakur. Multi-objective heuristic algorithms for practical portfolio optimization and rebalancing with transaction cost. *Applied Soft Computing*, 2018.

[20] M. Mitchell. *An Introduction to Genetic Algorithms (Complex Adaptive Systems)*. MIT Press, 1^{nd} edition, 1998. ISBN:978-0262631853.

[21] M. Mitchell. *Complexity: A Guided Tour*. Oxford University Press, 1^{nd} edition, 2011. ISBN:978-0199798100.

[22] A. Mochón, D. Quintana, Y. Sáez, and P. Isasi. Soft computing techniques applied to finance. *Springer Science*, 2007.

[23] J. J. Murphy. *Technical Analysis of the Financial Markets: A Comprehensive Guide to Trading Methods and Applications*. New York Institute of Finance, 1^{nd} edition, 1999. ISBN:978-0735200661.

[24] J. A. Ou and S. H. Penman. Financial statement analysis and the rediction of stock returns. *Journal of Accounting and Economics*, 1989.

[25] S. H. Penman. Accounting for intangible assets : There is also an income statement. *A Journal of Accounting, Finance and Business Studies*, 2009.

[26] W. F. Sharpe. Capital asset prices: a theory of market equilibrium under conditions of risk. *The Journal of Finance*, 1964.

[27] A. Silva, R. Neves, and N. Horta. A hybrid approach to portfolio composition based on fundamental and technical indicators. *Expert Systems with Applications journal*, 2015.

[28] J. W. Wilder. *New Concepts in Technical Trading Systems*. Trend Research, 1^{nd} edition, 1978. ISBN:978-0894590276.