Classification and Clustering of Stocks, using Genetic Algorithms and Fundamental Analysis

David Moura, MSc Student, IST

Abstract—Since the last two decades the ease of access to information has grown exponentially, making it easier to analyze and use this data in every field of science, including computational finance. This work presents an architecture made from scratch of a trading system that classifies stocks using two techniques, in order to conclude which one is superior: one with user input parameters, and an unsupervised one using a genetic algorithm to optimize clustering position, with a constant number of clusters. A genetic algorithm is also applied to optimize fundamental indicators to give buy and sell signals in each of the groups obtained, in order to conclude if stocks in the same group behave in similar fashion. Results have shown that the group with best results obtained with user input parameters is superior to the group with best returns obtained with the clustering algorithm. However the clustering algorithm classified stocks better, having increased performance over the user input method when used with the genetic algorithm using fundamental indicators. The proposed system was implemented from scratch, and was contains optimization modules, processing modules and a trading simulator.

Index Terms—Genetic Algorithms; Fundamental Analysis; Fundamental Indicators; Classification; Clustering; Stock Market

1 INTRODUCTION

This section is an introduction to the subject of computational techniques applied to companies’ financial data as a way to find trading rules and patterns in the stock market.

1.1 Motivation and Context

Several techniques are used to predict stock market quotes, being the most popular ones Artificial Intelligence (AI) methods to optimize financial indicators’ parameters. There are two types of financial indicators: fundamental and technical indicators. Other methods include the evaluation of each stock of a certain type, being this type defined by its sector or any other rule.

1.2 Problem Statement

The problem here is to construct a system that evaluates the stock market and accurately groups stocks that show similar behaviors.

1.3 Proposed Solution

The implemented solution is a system that classifies companies into groups based on their financial statements, in two ways: a supervised way with parameters defined by the user and an unsupervised way, applying a genetic algorithm to optimize clustering. After both classifications, the application uses a genetic algorithm once more to optimize indicators to buy and sell stocks, comparing all the results to conclude which method classifies better. The proposed system was implemented in C++ with about 9000 lines of code and it was made to be used by third parties that wish to improve its features.
1.4 Document Structure
In this section the document structure is presented. Section 2 shows theoretical concepts required to develop this project and an overview of the results obtained by related works. Section 3 documents the proposed solution. Section 4 presents a validation of the system. Section 5 summarizes this work, concluding the achievements, limitations and proposing future improvements.

2 STATE OF ART

2.1 Financial Indicators
Financial indicators analyze statistics and present that information in the form of ratios. These indicators will give better results if used alongside the right strategy [1].

There are several types of financial indicators, divided into Fundamental Indicators (FI) or Technical Indicators, which come from Fundamental Analysis (FA) and Technical Analysis (TA).

2.1.1 Fundamental Indicators
FA is used to check the intrinsic value of a certain market, industry or company, being FA applied to the latter the most used one. FA uses financial statements to know if a company is under or overvalued. FI are constructed using FA, and these type of indicators do not take into account market trends, only its intrinsic value, or in other words, the real raw value of the company [2].

FA has been made famous by value investors like Benjamin Graham and Warren Buffet (for their value investment strategies [3], [4]).

Using information given in financial statements, one can calculate some fundamental ratios used to compare companies, to decide which ones are the best to invest in.

2.1.2 Technical Indicators
TA and FA use different approaches towards investment, since TA uses movement of stock prices [5] and volume of transactions [6] as the main information to predict stock markets. TIs look for patterns in past data and use those patterns to forecast market tendencies [7]. TA generates trading rules by analyzing previous patterns of technical indicators [2].

2.2 Computational Intelligence (CI) Algorithms
This work uses Genetic Algorithms (GA). GAs are the most used kind of Evolutionary Algorithms (EA), an CI technique.

2.2.1 Genetic Algorithms
GAs are stochastic algorithms based on the evolutionary process of species. A simple GA pseudocode is given in algorithm 1.

```
t ← 0;
P(t) ← random;
Evaluation P(t);
while not Endcondition do
  Pp(t) ← Selection of parents from P(t);
Pc(t) ← Crossover from Pp(t);
Pm(t) ← Mutation of Pc(t);
Evaluation Pm(t);
P(t + 1) ← New Generation Creation from (P(t), Pm(t));
t ← t + 1;
end
```

Algorithm 1: Simple GA

The basic operators of a GA are selection, crossover and mutation. There are also other operators or techniques that can be applied to the GA in order to improve its results, such as elitism (propagating a percentage of the best individuals in a population into the next generation) or random immigrants (substituting the worst individuals in the population by random individuals).

2.3 Clustering
Clustering is an unsupervised technique, since it does not use preclassified data. Instead the algorithm discovers similarities (in the requested attributes) between objects of the set, grouping them in the same cluster. In [8] a GA is used to optimize clustering, using the Calinski-Harabasz index as fitness function, obtaining better results than classical clustering methods as K-means and Fuzzy C-Means (FCM) (for more information see [9] and [10]).
2.4 Related Work Results

In this section, some previous work results and data used are analyzed and compared between them in table 2.4.

<table>
<thead>
<tr>
<th>Work</th>
<th>Date</th>
<th>Approach</th>
<th>Data</th>
<th>Application</th>
<th>Benchmark</th>
<th>Validation Period</th>
<th>Returns</th>
</tr>
</thead>
<tbody>
<tr>
<td>[14]</td>
<td>2014</td>
<td>GA</td>
<td>TI</td>
<td>Portfolio Composition</td>
<td>S&amp;P Database</td>
<td>N/A</td>
<td>Better than B1/4 and TI</td>
</tr>
<tr>
<td>[16]</td>
<td>2011</td>
<td>GA</td>
<td>TI</td>
<td>Portfolio Composition</td>
<td>S&amp;P Database</td>
<td>N/A</td>
<td>Better than B1/4 and TI</td>
</tr>
<tr>
<td>[17]</td>
<td>2011</td>
<td>GA</td>
<td>TI</td>
<td>Portfolio Composition</td>
<td>S&amp;P Database</td>
<td>N/A</td>
<td>Better than B1/4 and TI</td>
</tr>
<tr>
<td>[18]</td>
<td>2011</td>
<td>GA</td>
<td>TI</td>
<td>Portfolio Composition</td>
<td>S&amp;P Database</td>
<td>N/A</td>
<td>Better than B1/4 and TI</td>
</tr>
</tbody>
</table>

3 ARCHITECTURE

This Chapter will give a description of the system architecture. First it will be given an overview of the proposed solution, afterwards a module style description of the most important modules and lastly, an explanation of other implementation details. The architecture of the proposed solution was made from scratch in C++. The implementation has 22 C++ classes, and a size of about 9000 lines of code.

3.1 Module View of the Global Architecture

The overview of the module architecture of the proposed solution is presented in this section (see figure 1).

There are nine main modules in the system, each one with a specific functionality.

3.2 Modules

This section explains in more detail the most important modules of this work. The modules are connected between them as showed in figure 2.

3.2.1 Fundamental Analysis (FA)

This module is a data analysis module, that will compute growth analysis and FIs from the financial statements of the companies (this data was obtained with [22] algorithm). The indicators used in this work will always have a maximization objective. They are the following:
\[
\text{Debt}_{\text{indicator}} = 1 - \frac{\text{Total Debt}}{\text{Total Assets}} 
\]

\[
\text{Payout Ratio} = \frac{\text{DPS}}{\text{EPS}} 
\]

\[
\text{ReturnOnEquity} = \frac{\text{Net Income (NI)}}{\text{Total Equity}} 
\]

\[
\text{ProfitMargin} = \frac{\text{NI}}{\text{Revenue}} 
\]

\[
\text{RevenueGrowth} = \frac{\text{Rev}_{\text{Actual}} - \text{Rev}_{\text{Last Year}}}{\text{Rev}_{\text{Last Year}}} 
\]

\[
\text{NetIncomeGrowth} = \frac{\text{NI}_{\text{Actual}} - \text{NI}_{\text{Last Year}}}{\text{NI}_{\text{Last Year}}} 
\]

\[
\Delta RG = \frac{\text{RG}_{\text{Actual}} - \text{RG}_{\text{Last Year}}}{\text{RG}_{\text{Last Year}}} 
\]

\[
\Delta NIG = \frac{\text{NIG}_{\text{Actual}} - \text{NIG}_{\text{Last Year}}}{\text{NIG}_{\text{Last Year}}} 
\]

\[
\Delta CFOA = \frac{\text{CFOA}_{\text{Actual}} - \text{CFOA}_{\text{Last Year}}}{\text{CFOA}_{\text{Last Year}}} 
\]

3.2.2 Classifier

This module does the classification of stocks for each quarter, based on the approach used by Peter Lynch in [23] (see figure 3 for better understanding of the structure).

The classification given has only into account the size and the growth of the company (see figure 3), and not the financial health, that was implemented for possible human analysis.

An example of how size classification is computed is given (health is similar). For growth classification the method is the same but with nine instead of three thresholds:

- Classification $= 1 \rightarrow \text{Value} < TH_{\text{Low}}$
• Classification = 2 → $TH_{Low} \leq Value < TH_{High}$
• Classification = 3 → $TH_{High} \leq Value$

Afterwards the classification given is accordingly:

• Small → Total Classification ∈ [0, 1.5]
• Medium → Total Classification ∈ [1.5, 2.5]
• Big → Total Classification ∈ [2.5, 3]

In this work the only Size indicator used is the last year’s Total Assets average, and the thresholds are: $TH_{Low} = 10B$ and $TH_{High} = 25B$.

The only growth indicator used is the revenue yearly growth, and the thresholds are: $TH_1 = -0.2$, $TH_2 = -0.1$, $TH_3 = -0.05$, $TH_4 = -0.02$, $TH_5 = 0$, $TH_6 = 0.02$, $TH_7 = 0.05$, $TH_8 = 0.1$, $TH_9 = 0.2$.

And the only financial Health indicator used is the last year’s $\frac{Total Debt}{Total Assets}$ indicator average, and the thresholds are: $TH_{Low} = 0.3$ and $TH_{High} = 0.7$

3.2.3 Clustering

This module creates five clusters each quarter, and associates each stock to the nearest cluster in the Growth × Size space. It uses the GA module to optimize the clusters’ positions, given at least one year training.

Values are scaled to give more appropriate weights to both growth and size. These scaled units are given in equation 10.

$$Size = \frac{Total\ Assets[Million\$]}{1000} \quad (10a)$$

$$Growth = \Delta\ Revenue \times 100 \quad (10b)$$

The growth measure is the annual revenue growth, and the size measure is the average of the Total Assets over the last year.

There will be a fixed amount of 5 clusters (the same amount as the types in the Classifier module), enumerated from A to E (the GA will recompute the best locations for clusters each quarter passed). To maintain consistency, the first cluster (cluster A) will always be the one nearest to the origin of the plane (Origin = Coordinates (0, 0)), B the second nearest, and so on.

3.2.4 GA

This module has two functionalities. One is to optimize weights given to fundamental indicators, and other is to optimize the location of the clusters in the plane. The way each works is described as follows:
3.2.4.1 Fundamental Analysis GA (FA GA): Since the Portfolios use FI, and FI requires a more long term analysis than TI, each time unit is considered a quarter. A generation is an iteration over the last 4 quarters of the GA. See figure 5 to see a structure of a chromosome.

Fig. 5. FI Chromosome Representation

At the beginning of the algorithm the population is generated randomly. Each FI weight is initialized as in equation 11a, the buy signal value is initiated as in equation 11b, and the sell signal value initiated as in equation 11c.

\[ r \in [0, 1] \quad (11a) \]
\[ b = \sum_{i=1}^{N} r_i a_i \quad (11b) \]
\[ s = \frac{b}{2} \quad (11c) \]

In equations 11 \( N \) is the number of indicators used, \( r \) and \( a \) are different random numbers, \( b \) is the buy signal value and \( s \) is the sell signal value. The \( s \) value is calculated as in equation 11c to guarantee that it is smaller and has a substantial percentage difference from the \( b \) value.

Buy and sell signals are given as in equation 12, where \( v \) is the value of indicator \( i \), and \( w \) is the weight of indicator \( i \).

\[ \text{Signal} = \begin{cases} \text{BUY} & \rightarrow \sum_{i=1}^{N} v_i \times w_i > b \\ \text{SELL} & \rightarrow \sum_{i=1}^{N} v_i \times w_i < s \end{cases} \quad (12) \]

The fitness function will be the ROI of each individual, when simulating investment.

\[ \text{ROI} = \frac{\text{Return} - \text{Initial Investment}}{\text{Initial Investment}} \quad (13) \]

3.2.4.2 Clustering GA: This is where the training and validation of the algorithm to optimize clusters’ positions occur, inspired in the ACGA algorithm from [8]. The GA module will run the GA to find the best locations for clusters centroids, being the output the centroids locations in the \( \text{Size} \times \text{Growth} \) plane, using Calinski-Harabasz index as fitness function. See figure 6 to see the structure of a chromosome.

The chromosomes are constructed with cluster points. Activation values are auxiliary structures that will determine if a certain cluster point is going to be used or not. The number of cluster points and activation values is the possible number of cluster positions given by the user. Cluster points and activation values are such that:

- Cluster Point is a tuple \((\text{size}, \text{growth})\), where \( \text{size} \in \mathbb{R}^+ \) and \( \text{growth} \in \mathbb{R} \)
- Activation signal is a number \( n \in [0, 1] \)

Fig. 6. Clustering Chromosome Representation

The minimum training period of the algorithm is 1 year. Chromosomes are randomly initiated, by assigning random values to Size and Growth, in order to create the \( N \) different possible points. This is done as in equation 14.

\[ \text{Size}_{\text{random}} = \frac{\max_{\text{size}}}{2} \times r_1 \quad (14a) \]
\[ \text{Growth}_{\text{random}} = \frac{\max_{\text{growth}}}{2} \times r_2 \quad (14b) \]

In equation 14 \( r_1 \) and \( r_2 \) are distinct random numbers and \( \max_{\text{size}}, \max_{\text{growth}} \) are the maximum size and growth measured in the first year.

The activation values are obtained as in equation 15.

\[ \text{Activation}_i = \frac{\# \text{ Stocks} \in C_i}{\# \text{ Stocks}} \quad (15) \]

In equation 15 \( i \) is the index of the solution, and \( C_i \) is the cluster of index \( i \).

Within the number of solutions decided by the user, the clusters with bigger activation values’ are chosen as solutions of that chromosome. The fitness function used is the Calinski-Harabasz metric (sometimes called variance ratio criterion), as described in equations 16. Although this metric is usually used to optimize the number of clusters created, in this work it will be used only as a
fitness function to the GA that optimizes clusters’ positions.

\[ CH = \frac{SS_B}{SS_W} \times \frac{N - K}{K - 1} \quad (16a) \]

\[ SS_B = K \sum_{j=1}^{K} n_j \| C_j - C \| \quad (16b) \]

\[ SS_W = \sum_{j=1}^{K} \sum_{i \in I_j} \| X_{ij} - C_j \| \quad (16c) \]

\[ C = \left( \frac{\sum_{\text{Stocks size}}}{N}, \frac{\sum_{\text{Stocks growth}}}{N} \right) \quad (16d) \]

In equations 16, \( SS_B \) is the between-cluster variance, \( SS_W \) is the within-cluster variance, \( N \) is the total number of stocks, \( K \) is the number of clusters (the number of solution clusters), \( n_j \) is the number of data points belonging to cluster index \( j \), \( C \) is the centroid of the dataset, \( C_j \) is cluster of index \( j \), \( I_j \) is the set of data points belonging to cluster \( j \) and \( X_{ij} \) is data point index \( i \) belonging to cluster index \( j \).

3.2.4.3 Operators: Two types of selection are implemented. A roulette wheel and a ranking selection. For this it will be used the fitness values shifted by the fitness of the worst individual, to avoid negative fitnesses.

The type of crossover done is a single point crossover.

The mutation implemented uses a \( \psi \) value to determine the maximum value of the perturbation. This \( \psi \) is a percentage \( \delta \) of the value that will be mutated. After finding this value, the mutation value \( \alpha \) will be computed as being a random number between \([0, \psi]\).

The mutation can be mathematically written:

\[ \psi = \delta \times v \quad (17) \]
\[ \alpha = \text{random} \in [0, \psi] \quad (18) \]
\[ \text{value} = \text{value} \pm \alpha \quad (19) \]

In equation 17 \( \text{value} \) is the value receiving the mutation, \( \alpha \) is the perturbation applied to the value, \( \psi \) is the maximum value of the perturbation and \( \delta \) is the percentage of perturbation chosen.

In this work values of \( \delta = 0.5 \) in indicators weights, and \( \delta = 0.2 \) in cluster positions are used.

3.3 Implementation Details

3.3.1 Configuration

The population size is 100, number of generations is 50 for FA GA and 200 for clustering optimization, mutation rate is 5%, elitism rate is 40%, random immigration rate is 20% and chromosome size is 11 for the FA GA (equals the number of indicators) and 75 for the clustering optimization (number of possible cluster points). These values were chosen taking into account related works and the execution time of the algorithm, although they did not take into account neither the size of the problem or the features optimized in this work. Transaction costs of 0.3% of the stock value are used in this work.

4 RESULTS

Every solution is compared with the S&P500 index and the B&H of the dataset in the specific time period. In the GAs used, the roulette wheel selection got better results, and these are the ones presented. These results are not compared with the existing related works because I was unable to find works that traded in a similar time period, and because this work uses quarterly trading only, while most works use daily and weekly trading.

In order to test the performance of the proposed solution, the metrics used in this work were the following:

- ROI - The Return on Investment will be used to measure the amount of return, given in percentage, that a certain investment had.
- Drawdown - This metric evaluates the biggest peak-to-trough decline during a specific period of investment.
- Sharpe Ratio - Used to measure the risk associated with the return of a portfolio.
- Success rate of trades
- Average time in the market
- Average return per trade
- Rate of positive quarters
- Average return per quarter
4.1 Case Studies

All the results were computed without dividends (stocks’ closing quotes were used), and with transaction costs of 0.3% in every buy or sell action.

There is no comparison with related studies because when this work was proposed I was unable to find neither works that used the same validation period, neither works that traded only quarterly, and so this work uses the B&H and the S&P500 index (specially the latter) as comparison with the obtained results.

The execution time of the clusters’ positions optimization algorithm is of about 5 hours and it takes 7 hours to create all of the portfolios once, with an i7 microprocessor.

4.1.1 Case Study I - User Input Classification

This case study presents the results of classifying stocks using user input data.

The strategy is to do a portfolio containing only a type of stock. This portfolio will buy stocks of a certain type the quarter that stock is classified as belonging to that type, and sell it in the quarter it stops belonging to that type. Since this classification is fixed, the values presented are the result of a single run.

We can see in figure 7 the graph of the accumulated returns, and in figure 8 the results of the several metrics used.

One can see that every type got worst results than the S&P500 index and the B&H of the dataset. The results of the Turnarounds type (the type with worst results) was expected, since companies with worst revenue growths are expected to have more trouble growing.

4.1.2 Case Study II - Clustering

This case study presents the results of clustering stocks by their size (total assets) and revenue growth, in the plane $Size \times Growth$, with the algorithm detailed in section 3. The algorithm has run 10 times, and the solution that achieved the median value of the averages of all quarters’ fitnesses was used, as shown in table 1.

Just as in Case Study I, the portfolios will buy stocks of a certain cluster the quarter that stock is classified as belonging to that cluster, and sold in the quarter it stops belonging to that cluster.

We can see in figure 9 the accumulated returns of the portfolios, and in figure 10 the metrics evaluating each portfolio.

In figure 11 is shown how companies are
TABLE 1
Average fitness per quarter of the clustering algorithm

<table>
<thead>
<tr>
<th>Clustering Method</th>
<th>Best</th>
<th>Worst</th>
<th>Median</th>
</tr>
</thead>
<tbody>
<tr>
<td>Calinski-Harabasz</td>
<td>806.4</td>
<td>665.37</td>
<td>767.8</td>
</tr>
</tbody>
</table>

Fig. 9. Clustering classification portfolios. The key is the following: A - Cluster A; B - Cluster B; C - Cluster C; D - Cluster D; E - Cluster E.

It is remarkable how data is so sparse in the Size axis. Cluster A, is the biggest cluster, containing most of the companies, while cluster E has only 4 companies in the fourth quarter of 2012.

One can conclude that the biggest companies of the dataset got huge returns on bull markets, and huge losses on bear markets, having small Sharpe ratios, and huge drawdowns. Big companies, outside this small set of huge companies, have the worst returns of all (the ones from cluster D). It can also be concluded that revenue growth can be chosen as an indicator to pick stocks with relatively good Sharpe ratios.

With the scale and number of clusters used there was a better grouping in size than in growth. Due to the amount and sparsity of data, to get a better clustering in the growth axis (especially in high values of size) it would be necessary more cluster points.

Fig. 10. Clustering portfolios metrics table. The key is the following: A - Cluster A; B - Cluster B; C - Cluster C; D - Cluster D; E - Cluster E.

4.1.3 Case Study III - Using GAs to optimize FI

This case study studies the results of applying GAs to give buy and sell signals on each of the groups of case studies 1 and 2, and the whole dataset.

The algorithm ran 10 times for each model, and in the models that used the clustering al-
algorithm, the clustering algorithm has run twice, so for each portfolio that uses clustering, 5 runs used the first run of the clustering algorithm, and the other 5 used the other run of the clustering algorithm.

One portfolio using user input classifications (the Slow Growers type) and two portfolios using clustering showed improvements (cluster D and E). The results present how the GA improved a cluster type portfolio (cluster E) and a classification type portfolio (Slow Growers), from the original portfolios of case study 1 and 2.

One can notice in figure 12 that in the portfolio E-GA there was a 27.34% increase in performance, most of which in the year 2015. The GA limited the losses of cluster E in 2015, and consequently, increased the Sharpe ratio.

In figure 13 the metrics of these portfolios are shown. It reduced substantially the number of trades done, when compared with portfolio SG (78.03% - from 132 trades in portfolio SG, to 29 trades in portfolio SG-GA).

One can conclude from this case study that Fundamental Indicators optimization has better results when applied to small sets of companies (as there were cluster E and D).

Also, the GA used in this work greatly reduced the number of trades done in all portfolios, reducing potential losses, but also potential gains. In the case of the E-GA portfolio, the major difference from the cluster E portfolio presented in case study 2 was the reduction of losses in 2015.

5 Conclusion

This work describes the implementation of a system that classifies stocks in the plane Size × Growth with two methods, using in both the same metrics of Fundamental Analysis, and that uses Fundamental Indicators for trading simulation.

The classification methods used were successful, having both achieved great returns in the expected groups. The genetic algorithm for weight optimization of fundamental indicators improved two cluster based portfolios, and one classification based portfolio. It also showed great success when applied to the cluster with bigger returns, improving the returns by 27.34%. To have better results in this optimization it would be necessary a bigger number of clusters so the algorithm could optimize Fundamental Indicators weights in smaller sets of stocks. These results allowed
to conclude that automatic classification using genetic algorithms is a better way of classifying stocks than using human input, since it is not prone to human error and does a more careful analysis of the data, grouping companies with more similar behaviors.

5.1 Achievements
The main achievements in this work were the following:

- The implementation from scratch of an architecture of a system containing data processing, genetic algorithms and a trading simulator.
- A classification method using Fundamental Analysis and Genetic Algorithm to optimize clusters' positions, with a fixed number of clusters.
- The comparison between an user input classification method and an automatic classification method.
- A long term investment strategy based on the implemented classification methods.
- The exclusive use of Fundamental Analysis and Fundamental Indicators with Genetic Algorithms.

5.2 Future Work

- Try different number of clusters, and optimize the number of clusters created to maximize the Calinski-Harabasz index.
- Use Technical Analysis to give the buy and sell signals of a portfolio.
- Tackle the Markwoitz Portfolio composition problem.

References


