prInvestor: Pattern Recognition based Financial Time Series Investment System

Dymitr Ruta[1]

BT Exact Technologies, Adastral Park, Orion Building, 1st floor - pp12,
Martlesham Heath, Ipswich IP5 3RE, UK, dymitr.ruta@bt.com

ABSTRACT
Predictability of financial time series (FTS) is a well-known dilemma. A typical approach to this problem is to apply a regression model, built on the historical data and then further extend it into the future. If however the goal is to support or even make investment decisions, regression-based FTS predictions are inappropriate as on top of being uncertain and unnecessarily complex, they require further analysis to make an investment decision. Rather than precise FTS prediction, a busy investor may prefer a simple decision on the current day transaction: buy, wait, sell, that would maximise his return on investment. Based on such assumptions a classification model is proposed that learns the transaction patterns from optimally labelled historical data and accordingly gives the profit-driven decision for the current day transaction. Exploiting a stochastic nature of an investment cycle the model is locally reduced to a 2-class classification problem and is built on many features extracted from the share price and transaction volume time series. Simulation of the model over 20 years of NYSE:CSC share price history showed substantial improvement of the profit compared to a passive long-term investment.

KEY WORDS
Financial Time Series, Regression, Classification, Decision Support

1 Introduction

Prediction of the financial time series represents a very challenging signal processing problem. Many scientist consider FTS as very noisy, non-stationary and non-linear signal but believe that it is at least to a certain degree predictable [3], [6]. Other analysis suggest that a financial market is self-guarded against predictability as whenever it shows some signs of apparent predictability, investors immediately attempt to exploit the trading opportunities, thereby affecting the series and turning it unpredictable [2]. Stable forecasting of FTS seems therefore unlikely to persist for longer periods of time and will self-destruct when discovered by a large number of investors. The only prediction model that could be successful and sustainable seems to be the one that exploits the supportive evidence either hidden to other investors or the evidence that is available but highly dispersed among many sources and therefore considered irrelevant, too difficult or too costly to be incorporated in the prediction model.

Irrespective of the above a number of techniques is being developed in an attempt to predict what seems unpredictable: tomorrow's share price based on historical data. Starting from simple linear Autoregressive Moving Average models (ARMA) [3] through conditional heteroscedastic models like ARCH or GARCH [3] up to the complex non-linear models [3], [4], the idea is similar: establish the regression-based description of the future samples based on the historical data series. More recently a number of machine learning techniques started to be applied to a financial forecasting and on a number of occasions showed considerable improvement compared to a traditional regression models [5], [6], [7]. Neural networks are shown to be particularly good at capturing complex non-linear characteristics of FTS [5], [6]. Support vector machines represent another powerful regression technique that immediately found applications in financial forecasting [8], [7].

While there is already extensive knowledge available in pattern recognition domain, it has been rarely used for FTS prediction. The major problem lies in the fact that classification model learns to categorise patterns into crisp classes, rather than numerical values of the series. A temporal classification models would have to provide a specific definition of classes or obtain it from the series by discretisation. Although some work has already been done in this field [10], [9], [11], [13] there is still lack of pattern recognition based models that would offer immediate investment applications surpassing in functionality and performance the traditional regression based models.

The proposed prInvestor model is a step forward towards a fully automated pattern recognition based investment system. Rather than predicting future series values it uses a classification model that learns from expandable historical evidence how to automatically categorise the future series into investment actions: buy, wait or sell. The prototype of prInvestor is tested on the 20-years of daily share price series and the results are analysed to give recommendations towards prospective fully automated platform development.

The remainder of the paper is organised as follows. Next section provides a detailed analysis of the proposed temporal classification with prInvestor, specifying the investment cycle, an algorithm for optimal labelling of training data, feature extraction process and the classification model used. Section 3 presents the results of extensive experiments with a real 20-year share price data series evaluating the performance of the prInvestor system. The concluding remarks and some suggestions for model refinement are shown in the closing Section 4.

2 Temporal classification with prInvestor

Classification represents a supervised learning technique that tries to correctly label the patterns based on multidimensional set of features [1]. The model is fully built in the training process carried out on the labelled dataset with a discriminative set of features. Based on the knowledge gained from the training process, a classifier assigns the label to a new previously unseen pattern.

Adapting the pattern recognition methodology, prInvestor would have to generate the action label: buy, sell or wait to the current day feature values based on the knowledge learnt from historical data. For that to be possible the training data have to be optimally labelled such that the investments corresponding to the sequence of labels generate the maximum return possible. Furthermore, to maximise the discrimination among classes, prInvestor should exploit the scalability of pattern recognition models and use as many relevant features as possible, far beyond just the historical share price series. All the properties mentioned above along with some mechanisms controlling model flexibility and adaptability are addressed in the presented prInvestor system.

2.1 Investment cycle

In the simplified investment cycle, considered in this paper, the investor is using all assets during each transaction which means he buys shares using all the available cash and sells all shares at once. Assuming this simplification, there are four different states an investor can fall into during the investment cycle. He enters the cycle at the state ”wait with money” (WM), where the investor is assumed to possess financial assets and is holding on in the preparation for a ”good buy”. From there he can either wait at the same state WM or purchase the shares at the actual price of the day thereby entering the ”buy” state (B). From state B investor can progress to two different states. Either he enters ”wait with shares” state (WS) preparing for the good selling moment, or he sells immediately (the day after the purchase) all the shares transferring them back to money at the ”Sell” (S) state. If the investor chooses to wait with shares (WS), he can stay at this state or may progress only to the state S. The sell state has to be followed by either the starting ”wait with money” state WM or directly buy state (B) that both launch the new investment cycle. The complete investment cycle is summarised by the conceptual diagram and accompanied directed cyclic graph both shown in Figure 1.

Immediate consequence of the investment cycle is that the labelling sequence is highly constrained by the allowed transaction paths i.e. obeying the following sequentiality rules:

- WM may follow only with WM or B
- B may follow only with WS or S
- WS may follow only with WS or S
- S may follow only with WM or B

The above rules imply that for the current day sample the system has to pick only one out of two labels depending on the label from the previous sample. That way the 4-class problem is locally simplified to the 2-class classification problem. In a real-time scenario it means that to classify a sample, the model has to be trained on the dynamically filtered training data from only two valid classes at each sample. If the computational complexity is of concern, retraining at each sample can be replaced by 4 fixed models that can be built on the training data subsets corresponding to to 4 combinations of valid pairs of classes as stated in the above sequentiality rules.

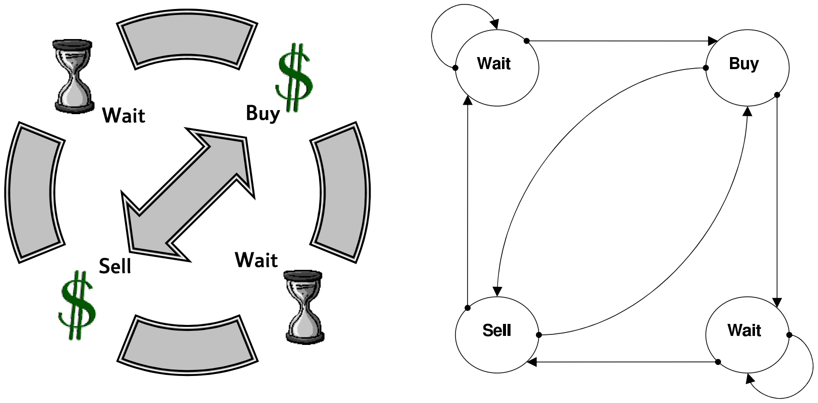


Fig. 1. Visualisation of the investment cycle applied in prInvestor. The wait state has been separated into 2 states as they have different preceding and following states.

2.2 Training set labelling

Classification, representing supervised learning model, requires labelled training data for model building and classifies incoming data using available class labels. In our share investment model, the training data initially represent unlabelled daily time series that has to be labelled using four available states WM, B, S, WS, subject to sequentiality rules. A sequence of labels generated that way determines the transaction history and allows for a calculation of the key performance measure - the profit.

In a realistic scenario each transaction is subjected to a commission charge, typically set as a fixed fraction of the transaction value. Let x_t for $t \in 0, 1, .., N$ be the original share price series and c stand for the transaction commission rate. Assuming that b and $s = b + p$, where $p \in 1, .., N - b$, denote the buying and selling indices, the relative capital after a single buy-sell investment cycle is defined by:

$$C_b^s = \frac{x_s}{x_b} \frac{1-c}{1+c} \qquad (1)$$

Note that the same equation would hold if the relative capital was calculated in numbers of shares resulting from a sell-buy transaction. Assuming that there are T cycles in the series let $b(j)$ and $s(j)$ for $j = 1,..,T$ denote indices of buying and selling at j^{th} cycle such that $x_{b(j)}$ and $x_{s(j)}$ stand for buy and sell prices in j^{th} cycle. Then the relative capital after k cycles $(0 < k \leq T)$ can be easily calculated by:

$$C_{b(1)}^{s(k)} = \prod_{j=1}^{k} C_{b(j)}^{s(j)} \tag{2}$$

The overall performance measure related to the whole series would be then the closing relative capital, which means the relative capital after T transactions:

$$C_T = C_{b(1)}^{s(T)} \tag{3}$$

Given C_T the absolute value of the closing profit can be calculated by:

$$P = C_0(C_T - 1) \tag{4}$$

where C_0 represents the absolute value of the starting capital. Finally, to be consistent with the investment terminology one can devise the return on investment performance measure, which is an annual average profit related to initial investment capital:

$$R = \frac{\overline{P}_{ANN}}{C_0} = \left[C_{b(1)}^{s(T)}\right]^{\frac{t_{ANN}}{s(T)-b(1)}} - 1 \tag{5}$$

where t_{ANN} stands for the average number of samples in 12 months.

The objective of the model is to deliver an optimal sequence of labels, which means the labels that through the corresponding investment cycles generate the highest possible profit. Such optimal labelling is only possible if at the actual sample to be labelled, the knowledge of the future samples (prices) is available. Although reasoning from the future events is forbidden in the realistic scenario, there is no harm of applying it to the training data series. The classification model would then have to try to learn the optimal labelling from historical data and use this knowledge for classification of new previously unseen samples.

An original optimal labelling algorithm is here proposed. The algorithm is scanning the sequence of prices and subsequently finds the best buy and sell indices labelling the corresponding samples with B and S labels respectively. Then all the samples in between of B and S labels are labelled with WS label and all the samples in between of S and B labels are labelled with WM label as required by the sequentiality rules. The following rules are used to determine whether the scanned sample is identified as optimal buy or sell label:

- Sample x_b is classified with the label B (optimal buy) if for all samples x_t $(b < t < s)$ between sample x_b and the nearest future sample x_s $b, s \in$

$1, .., N \cap b < s$, at which the shares would be sold with a profit ($C_b^s > 1$), the capital $C_t^s < C_b^s$.

- Sample x_s is classified with the label S (optimal sell) if for all samples x_t ($s < t < b$) between sample x_s and the nearest future sample x_b $b, s \in 1, .., N \cap s < b$, at which the shares would be bought increasing the original number of shares ($C_b^s > 1$), the capital $C_b^t < C_b^s$.

2.3 Feature extraction

The data in its original form represents only a share price time series. Extensive research dedicated to time series prediction [3] proves that building the model solely on the basis of historical data of the FTS is very uncertain as it exhibits considerable proportion of a random crawl. At the same time an attractive property of classification systems is that in most of the cases they are scalable, which means they can process large number of features in a non-conflicting complementary learning process [1]. Making use of these attributes, prInvestor takes the original share price series, the average transaction volume series as well as the label series as the basis for the feature generation process. Details of the family of features used in prInvestor model are listed in Table 1. Apart from a typical moving average, defferencing features, there are new features (plf, ppf) that exploit the labels of past samples in their definition. The use of labels as features might draw some controversies as it imposes that current model outputs depend on its previous outputs. This is however truly a reflection of the fact that investment actions strongly depend on the previous actions as for example if a *good buy* moment was missed the following *good sell* point could no longer be *good*. Incorporation of the dependency on previous system outputs (labels) injects also a needed element of the flexibility to the model such that after a wrong decision, it could quickly recover rather than make further losses. Another consequence of using past labels as features is the high non-linearity and indeterminism of the model and hence its limited predictability.

It is important to note that the features proposed in prototype of the prInvestor model are just a proposition of simple, cheap and available features which by no means form the optimal set of features. In fact as the series is time related, countless number of features starting from the company's P/E ratio or economy strength indicators up to type of weather outside or the investment mood could be incorporated. The problem of generation and selection of the most efficient features related to the prInvestor model is by far open and will be considered in more detail in the later version of prInvestor.

2.4 Classification model

Given the set of features, labelled training set and the performance measure, the model needs only a relevant classifier that could learn to label the samples based on optimal labels and the corresponding historical features available in

Name	Description
prc	Average daily share price
vol	Daily transactions volume
$mva_i(x)$	Moving average - mean from i last samples of the series x
$atd_i(x)$	Average difference between the current value of x and $mva_i(x)$
$dif_i(x)$	Series x differenced at i^{th} order
plf_i	The *past label* of the sample taken i steps before the current sample.
ppf	Difference between the current price and the price at B or S labels

Table 1. A list of features used in the prInvestor model.

the training set. Before the decision on the classifier is made, it is reasonable to consider the complexity and adaptability issues related to prInvestor working in a real-time mode. Depending on the choice of training data there are three different modes prInvestor can operate on. In the most complex *complete mode* the model is always trained on all available data to date. At each new day the previous day would have to be added to the training set and the model retrained on typically immense dataset covering all available historical evidence. In the *fixed mode* the model is trained only once and then used in such fixed form day by day without retraining that could incorporate the new data. Finally in the *window mode* each day the model is retrained on the same number of past samples. Undoubtedly the model is fastest in *fixed mode*, which could be a good short-term solution particularly if complexity is of concern. *Complete mode* offers the most comprehensive training, however at huge computational costs and poor adaptability capabilities. The most suitable seems to be the *window mode* in which the model is fully adaptable and its complexity can be controlled by the window width.

Given relatively large datasets and the necessity of retraining, it seems reasonable to select a rather simple easily scalable classifier that would not be severely affected by the increase in sizes of both feature and data sets. The simple quadratic discriminant analysis (QDA) classifier seems to be a good choice that accommodates the above properties while being still capable of capturing some non-linearities. Details of quadratic discriminant classifier can be found in [1]. It is important to note that given a day lag of the series, there is plenty of time for retraining even using large datasets. The model is therefore open for more complex classifiers or even the mixture of classifiers that could potentially improve its performance. The QDA classifier used in this work, due to its simplicity is particularly useful in this early prototyping stage where the experimentation comprehensiveness is the top priority rather than maximum possible performance. Moreover, simple classifiers are also preferred for shorter-lag series where the retraining might be necessary every hour or minute.

3 Experiments

Extensive experimentation work has been carried out to evaluate prInvestor. Specifically prInvestor was assessed in terms of the relative closing capital compared to the relative closing capital of the passive investment strategy of buying the shares at the beginning and selling at the end of the experimental series. Rather than a large number of various datasets only one dataset has been used but covering almost 20 years of daily average price and volume information. The dataset represents Computer Sciences Corporation average daily share price and volume series since 1964 to 1984, available at the corporation website (www.csc.com).

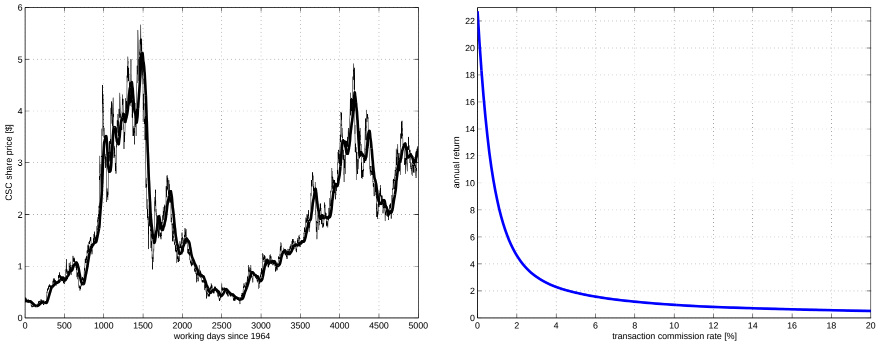
Initially the dataset has been optimally labelled for many different commission rates, just to investigate what is the level of maximum possible oracle-type return from the share market investment. The experimental results shown in Figure 2(b) reveal surprisingly large double-figure return for small commission charges ($< 1\%$) falling sharply to around 0.5 (50%) for high commission charges ($> 10\%$). Relating this information to the plot of the original series shown in Figure 2(a) it is clear that to generate such a huge return the algorithm has to exploit all possible price rises that exceed the profitability threshold determined by the commission rate. It also indicates that the most of the profit is generated on small but frequent price variations ($\pm 2\%$), which in real-life could be considered as noise and may not be possible to predict from the historical data. The maximum possible annual return or the profile presented in Figure 2(b) can be also considered as a measure describing potential speculative investment attractiveness of the corresponding company.

Due to many varieties of prInvestor setup and a lack of presentation space in this paper the experiments have been carried out in the moving-window mode as a balanced option featuring reasonable flexibility and adaptability mechanisms. A large pool of around 40 features have been generated from the family of features presented in Table 1. Then, for each of the window width fixed at: 6 months, 1, 2 and 3 years, prInvestor has been run 100 times using QDC classifier and a random subset of features. The performances obtained in a form of closing capital related to the capital of passive strategy *rcp* indicated that the width of 2 years (around 500 days) is optimal.

Having decided on the 2 years moving window mode prInvestor was further tuned by selection of the most relevant features. A simple evaluative search based on probability based incremental learning [12] has been used to find the best possible subset of features. As a result 21 features listed in Table 2 have been selected. The model was then trained on the first 500 days (2 years window) that have been optimally labelled. In the next step the investment simulation was launched, in which at each next day the model generated the transaction label based on the training on the preceding 500 optimally labelled samples. The resulted sequence of transaction labels represents complete output of the model. Figure 3(a) illustrates the transactions generated by the prInvestor model while Table 3 shows the performance results. Important point is that most of investment cycles generated by prInvestor were profitable. Moreover the occasional loss cycles occur mostly during *bessa* and are relatively short in duration. The model is quite eager to invest during *hossa*, which seemed to be at least partially picked from the historical data. Numerical evaluation of the prInvestor depicted in Figure 3(b), shows more than 5 times higher closing capital than for the case of passive investor who buys at the beginning and sells at the end of 20 years period. Such remarkable results correspond on average to almost 20% annual return from investment and give a good prospect for the development of the complete investment platform with a number of carefully developed features and the data incoming automatically to the system for daily processing resulting in a final investment decision.

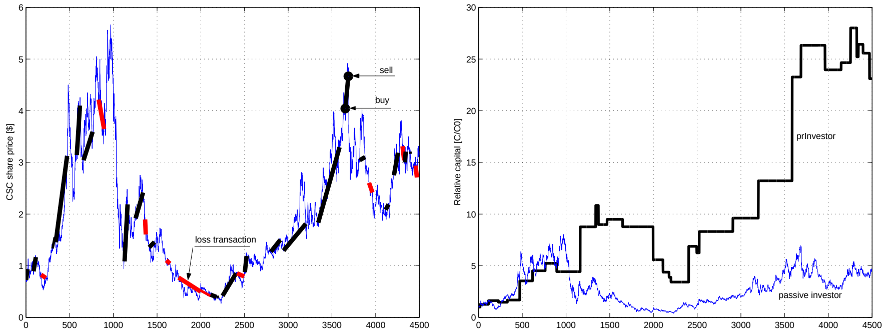
prc	vol	$mva_i(x)$	$atd_i(x)$	$dif_i(x)$	plf_i	ppf
prc	vol	$mva_1(vol)$	$atd_1(prc)$	$dif_1(p)$	plf_1	ppf
		$mva_5(prc)$	$atd_1(vol)$	$dif_1[mva_{20}(prc)]$		
		$mva_{20}(vol)$	$atd_5(prc)$	$dif_1[mva_{20}(vol)]$		
		$mva_{50}(prc)$	$atd_5(vol)$	$dif_2(prc)$		
		$mva_{50}(vol)$	$atd_{20}(prc)$	$dif_2(vol)$		
			$atd_{20}(vol)$	$dif_2[mva_{20}(prc)]$		
				$dif_2[mva_{20}(vol)]$		

Table 2. Features selected by the probability based incremental learning algorithm [12] from the pool of around 40 featured generated from the family of features introduced in Table 1. The selected features are grouped by their families.



(a) CSC daily share price since 1964 (b) Annual return as a function of transaction commission rate

Fig. 2. Visualisation of the optimal labelling algorithm capability.



(a) prInvestor transactions (b) Relative capital - comparison

Fig. 3. prInvestor in action: the transactions returned as a result of learning to invest with profit from historical data. 3(a) Visualisation of the transactions generated by prInvestor. 3(b) Comparison of the relative capital evolutions of the prInvestor and passive investor always keeping shares.

Investor	Annual return [%]	Relative closing capital (rcc) [%]
Passive investor	8.9	463.8
prInvestor	19.1	2310.7

Table 3. Performance comparison between passive investment strategy and prInvestor obtained over investment simulation on 20 years of CSC share price history.

4 Conclusions

prInvestor is a proposition of the intelligent share investment system. Based on extensive historical evidence it uses pattern recognition mechanisms to generate a transaction decision for the current day: *buy*, *sell* or *wait*. The advantage of this model over the existing techniques is that it is capable of incorporating in a complementary, non-conflicting manner various types of evidence beyond just the historical share price data. The proposed system benefits further from optimal labelling algorithm developed to assigns the transaction labels to the training series such that the maximum possible return on investment is achieved. The model features basic flexibility and adaptability mechanisms such that it can quickly recover from bad decisions and adapt to the novel trend behaviour that may suddenly start to appear. The robustness of the prInvestor is demonstrated on just a the few simple features generated upon the share price and volume series. With such a simple setup the model hugely outperformed the passive investment strategy of buying at the beginning and selling at the end of the series, resulting on average in 20% annual return from investment. Despite tremendous average results the model is not always consistent and occasionally generates losses. Full understanding of this phenomenon requires deeper analysis of the role of each individual feature in the decision process. In addition there is plenty of unknowns relating to the choice of features, classifier and the real-time classification mode. All these problems and doubts will be the subject of further investigation towards fully automated and robust investment system that could potentially find some commercial applications. For that to happen the system would have to meet very restrictive reliability requirements confirmed by the extensive testing across different company shares, markets, and time.

References

1. Duda RO, Hart PE, Stork DG (2001) Pattern classification. John Wiley & Sons. New York
2. Timmermann A, Granger CWJ (2004) Efficient market hypothesis and forecasting. International Journal of Forecasting 20(1): 15-27(13)
3. Tsai RS (2002) Analysis of financial time series. John Wiley & Sons
4. Casdagli M (1989) Nonlinear prediction of chaotic time series. Physica 35: 335-356
5. Vojinovic Z, Kecman V, Seidel R (2001) A data mining approach to financial time series modelling and forecasting. International Journal of Intelligent Systems in Accounting, Finance & Management 10(4): 225-239(15)
6. Giles CL, Lawrence S, Tsoi AC.(2001) Noisy time series prediction using recurrent neural networks and grammatical inference. Machine Learning 44(1/2): 161-183(23)
7. Tay FEH, Cao LJ (2002) Modified support vector machines in financial time series forecasting. Neurocomputing 48(1): 847-861(15)
8. Mukherjee S, Osuna E, Girosi F (1997). Nonlinear prediction of chaotic time series using support vector machines. In Proceedings of the 7^{th} IEEE Workshop on Neural Networks for Signal Proc., IEEE Press
9. Singh S (2000) Noisy time-series prediction using pattern recognition techniques. Computational Intelligence 16(1): 114-133(20)
10. Morik K, Rueping S (2002) A multistrategy approach to the classification of phases in business cycles. In Taprio E, Mannila H, Toivonen H (eds) Lecture Notes in Artificial Intelligence (Machine Learning ECML 2002), pp. 307-318, Springer-Verlag
11. Wu B, Hung S-L (1999) A fuzzy identification procedure for nonlinear time series: With example on ARCH and bilinear models. Fuzzy Sets and Systems 108(3): 275-287(13)
12. Ruta D, Gabrys B. Application of the evolutionary algorithms for classifier selection in multiple classifier systems with majority voting. In Proceedings of the 2^{nd} International Workshop on Multiple Classifier Systems, LNCS 2096, pp 399-408 Springer Verlag
13. Siekmann S, Neuneier R, Zimmermann H-G, Kruse R (1999) Neuro fuzzy systems for data analysis. In Computing with Words in Zadeh LA, Kacprzyk J (eds) Information/Intelligent Systems Applications (Studies in Fuzziness & Soft Computing, Springer-Verlag