XAI models applied to investing

Alban Sarfati

CentraleSupélec

Abstract. Machine learning techniques have recently become the norm for detecting patterns in financial markets. However, relying solely on machine learning algorithms to make decisions can have negative consequences, especially in such a critical area as investing. Investing involves making decisions about buying or selling securities based on various factors. Investors are compensated for the risk of holding securities, which may decline in value between the purchase and the sale. Can an investor/trader explain clearly the intuition of his position on securities ? This project proposes a machine learning approach powered by eXplainable Artificial Intelligence techniques integrated into a trading pipeline. Specifically, I propose a investing strategy that include feature selection methods providing accuracy improvements and is less noisy than the one embedded with the whole feature set.

Keywords: Trading Strategy · Random Forest · Lime · Shap.

1 Introduction

The financial sector is well suited to the application of machine learning, as large amounts of data are available due to the revolution of electronic trading and more accessible computing power.

In a machine learning context, a basis investing task it to use models to analyze financial data and make prediction about future behavior of financial markets such as the direction of securities movements (upward or downward).

Financial data series have unique characteristics that make prediction tasks difficult, such as their non-linear and non-stationary nature. To try to reveal the entire spectrum of information, powerful AI systems have been engineered with a single goal of maximizing predictive performance.

The goal in this work is to use these "black boxes" model to balance accuracy and eXplainability. I first define a feature selection framework tailored for stock returns direction forecasting that increases the accuracy of the employed ML model and then I use the SHAP (SHapley Additive exPlanation) values to evaluate the contributions of each individual feature to the overall upward and downward movement logit probability.

2 Data

2.1 Overview

I construct the dataset by using daily historical data from yahoo finance, the data has been requested for technological stocks - Apple and Google - over the last twenty years. Each instance is indexed by a timestamp and characterized by the open price, close price, highest price, lowest price and volume. Based on this information, for each stock, I create the features and the labels that will allow to form the training, validation and testing dataset.

2.2 Features

The first kind of feature is the one that uses only the closing price of each day, they are the returns (i), the direction (ii) and the volatility (iii). (i) The returns of the closing price (ROC) is calculated as the percentage change between the price at t and price at $t-1$. (ii) The direction (upward or downward) of the price corresponds to the sign of the difference between the price at t and the price at $t-d$ where $d \in \{1, 2, 3, 4, 5, 21\}$. These lagged features are intended to provide ML algorithms with information in two different time horizons: the recent and medium past. (iii) The volatility of returns which measures the level of risk associated with investing in a particular security.

The second kind of features used are technical indicators, they represent statistical tools and are extensively use to make investment decisions by generating signals. I developed eight of them to identify trends, regime switches, momentum and potential reversal points in the stock market. They can be can be divided into four categories:

Trend followers identify the recent main movements of stock prices in recent past, using indicators like Exponential Moving Average (EMA) with a period of 7, 14 and 21 days.

Divergence identifiers aim to detect potential regime switches, where the current trend ends and the prices start moving in the opposite direction, using indicators like the Moving Average Convergence Divergence (MACD).

Momentum indicators measure the strength and direction of a stock's price movement, such as the Relative Strength Index (RSI).

Oscillators track the price variation of stocks to identify possible reversal points using indicators like Williams %R and STOCH.

By combining these features with different time windows, I obtain a total of 17 features.

2.3 Label

The label is a binary variable given by the price direction. It is therefore used to predict upward or downward movements in the market.

2.4 Transformation

The features are scaled within the range $(0, 1)$ by applying a MinMaxScaler fit on the training set. The rescaling is a necessary step as it un-biased learning and ensure that each feature contributes equally to the decision-making process. The transformation is learnt on the training set only in order to ensure that no data-leakage happens.

2.5 Environment

The training, validation and testing set are constructed based on 2.1, 2.2, 2.3 and 2.4 with a training split corresponding to the first 70% time series instances, while validation and testing are 15% (2017 to 2019 and 2020 to 2022). The validation part is used for the feature selection, the testing part evaluate if this selection has improved the performance of the trading strategy and accuracy. To take this into account, the construction of the environments is based on specified features (full set or subset).

3 Model

3.1 Overview

In the forecasting process, I have considered Random Forests which is a decision-tree-based model and therefore is intrinsically explainable. Moreover, by being composed of deep de-correlated decision trees, as a consequence of bagging and random feature selection, random forest can handle noisy and outliers instances, which can be particularly useful when working with stock price data.

3.2 Tuning

Optimizing the hyper-parameters of the RandomForestClassifier is an important step in achieving optimal performance and accuracy for the trading strategy. For each stock, the number of trees in the forest, the maximum depth of each tree, the maximum number of features considered at each split have been tuned by using : (i) cross validation approach (TimeSeriesSplit) on only the training set, (ii) full set of features, (iii) maximizing accuracy score.

3.3 Training

Based on the optimal hyper-parameters, I train each RandomForestClassifier using either the full set of features or only filtered ones. The model associated with the full set of features is the baseline and will be the benchmark for performance comparisons.

3.4 Metrics

In the experiments I provide the performance of these models according to different metrics: accuracy, f1-score, AUC, brier-score. However, the good performance of these metrics is not necessarily correlated to a good financial performance. Indeed, when a large number of transactions with a modest impact is correctly matched, while transactions with a very high impact is missed, the outcome would be a good statistical performance but a potentially very bad financial performance. Hence, it is important to correctly predict large market movements and not just the majority of market movements.

Trading Strategy The direction's predictions of the model are transformed into positions, i.e. if at time t the model predict 0 (upward) then $position_t = 1$ else $position_t = -1$. The positions are backtested in the following steps:

*At each time step t, the return of the strategy is $position_{t-1} * stockreturn_t$.

*Deduction of transaction costs when a trade has taken place.

*Calculation of the net asset value of the strategy and benchmark.

*Calculation of the absolute and the out performance of the strategy.

The performance (absolute or out) provide a complete picture of the model's financial performance.

4 Feature selection

4.1 Overview

The second block of the forecasting phase aims at assigning importance scores to features in order to identify features that are uninformative for the prediction task and to propose them for deletion.

4.2 Methods

Four methods were used to calculate the importance of each feature within the full set of features. Their common properties are : (i) it is model agnostic, i.e. it can be applied to any tree-based model, (ii) it computes the feature importance on validation set, which makes it possible to highlight which features contribute the most to the generalization power of the inspected model, (iii) it has no tuning parameters.

Impurity importance

Permutation importance

Lime importance

SHAP importance

4.3 Removal strategy

I compute the feature importance value (on validation set) with each of the proposed method by using the baseline classifier and I store the features having the lowest importance. For the permutation importance this is all the features with a negative feature importance scores, otherwise it is the 5 features with lowest importance scores.

This set of 5 features (or more) constitutes the space of removable features and for k $\in (1, 2, 3, 4, 5)$ a new set of features is formed as $\binom{5}{k}$. For each of these combinations, a RandomForestClassfier is fitted without these features and I compare its validation accuracy with the one of the baseline.

Hence, for each of the proposed method, I am looking for the combination of removable features that leads to the largest increase in accuracy over the validation set. Once found, the respective RandomForestClassfier, i.e the one fitted with the best features, is evaluated on the test set and the metrics are saved. Note that there may not have been an increase of validation accuracy between the baseline and the respective RandomForestClassfier of a feature selection method.

Finally, I compare these final test metrics for each proposed feature importance method and select the one leading to the largest increase in accuracy.

5 Feature interpretation

5.1 Overview

Drawing from the research conducted by Lundberg and Lee in 2017, the study employs Shapley values as a means to depict the impact of individual features on the likelihood of an upward movement occurring. Providing a global interpretation of features' impacts.

5.2 Method

We can rank the Shapley values in order of importance, defined as the average absolute Shapley value over the testing set. By doing so, the testing prediction of the classification model can be examined.

I use the famous SHAP summary plot which represent the joint behavior of Shapley and features values at each point in time to better grasp their non-linear dependencies. The y-axis reports the Shapley values, i.e. the feature contributions to the model output in log-odds while the color of the dot represents the value of that feature at each point in time.

For each stock, I analyse the testing predictions of the model built with the feature importance method leading to the highest testing accuracy.

6 Results

6.1 Feature selection

From Table 1, one can see that only the lime and permutation feature importance method have increased the Apple validation accuracy, compared to the baseline. However, all of the methods have higher f1-score, which can be considered as a better metric than accuracy in the context of predicting stock returns because accuracy alone can be misleading when the dataset is imbalanced, which is often the case in financial market (bull market vs bear market). However, only the permutation method allowed to have a positive outperformance of the trading strategy of 73% compared to -115% for the baseline.

From Table 2, we can observe that each feature selection method allowed the Apple accuracy and the trading outperformance to be increased on the testing set. Hence, the best RandomForestClassifier of each feature selection method has generalized well, theses methods introduce clearly improvements both in terms of predictive and financial performance.

Method	accuracy	f1-score	auc	brier	outperformance	removed features
baseline	0.46	0.4	0.5	0.26	-1.15	None
impurity	0.46	0.42	0.51	0.25	-1.29	[Direction3, Direction21, Direction1]
permutation	0.56	0.49	0.5	0.25	0.73	[Direction21, EWMA7, EWMA14, EWMA21, Volatility14]
lime	0.48	0.47	0.5	0.25	-1.21	[EWMA7, Direction3, Direction21, Direction2, Direction1]
shap	0.46	0.42	0.51	0.25	-1.29	[Direction3, Direction1, Direction21]

Table 1. Apple validation metrics for feature selection methods.

Method	accuracy	f1-score	auc	brier	outperformance	removed features
baseline	0.49	0.45	0.52	0.25	-1.09	None
impurity	0.51	0.48	0.54	0.25	-0.47	[Direction3, Direction21, Direction1]
permutation	0.53	0.44	0.54	0.25	1.65	[Direction21, EWMA7, EWMA14, EWMA21, Volatility14]
lime	0.51	0.51	0.53	0.25	-0.81	[EWMA7, Direction3, Direction21, Direction2, Direction1]
shap	0.51	0.48	0.54	0.25	-0.47	[Direction3, Direction1, Direction21]

Table 2. Apple testing metrics for feature selection methods.

From Table 3 and 4, one can see that each feature selection method allowed the Google accuracy and the trading outperformance to be increased on the validation and testing set.

Overall, i.e. for Apple and Google, the permutation method is the one working the best in terms of accuracy and trading outperformance. I plotted in Table 5 and 6, the gross performance of the trading strategy vs the passive benchmark.

Method	accuracy	f1-score	auc	brier	outperformance	removed features
baseline	0.51	0.45	0.48	0.25	0.05	None
impurity	0.53	0.44	0.5	0.25	0.08	[Direction2, Direction3, Direction5, Direction21]
permutation	0.57	0.56	0.56	0.25	2.16	[OBV, EWMA7, Direction3, RSI, Direction1, ..]
lime	0.54	0.4	0.51	0.25	0.15	[EWMA7, Direction5, Direction21]
shap	0.53	0.44	0.5	0.25	0.08	[Direction1, Direction2, Direction3, Direction21]

Table 3. Google validation metrics for feature selection methods.

Method	accuracy	f1-score	auc	brier	outperformance	removed features
baseline	0.52	0.43	0.53	0.25	-0.25	None
impurity	0.53	0.42	0.5	0.25	-0.01	[Direction2, Direction3, Direction5, Direction21]
permutation	0.53	0.53	0.51	0.25	1.01	[OBV, EWMA7, Direction3, RSI, Direction1, ..]
lime	0.53	0.38	0.52	0.25	0.01	[EWMA7, Direction5, Direction21]
shap	0.53	0.42	0.5	0.25	-0.01	[Direction1, Direction2, Direction3, Direction21]

Table 4. Google testing metrics for feature selection methods.

For Apple, the number of trades is 127 for a total of 730 ticks, the absolute performance of the strategy with fees is 247% and the outperformance of the strategy with tc is 106%. For Google, the number of trades is 218 for a total of 674 ticks, the absolute performance of the strategy with fees is 112% and the outperformance of the strategy with tc is 101%.

Table 5. Apple testing trading strategy with permutation method.

Hence, the trading strategies that include such feature selection methods improve the financial performances by providing long-short predictive signals whose information content suffices and is less noisy than the one embedded in the whole feature set.

Table 6. Google testing trading strategy with permutation method.

6.2 Feature interpretation

Concerning Table 7 and 8, they represent the SHAP summary plot of Apple and Google on testing predictions of the model built with the permutation method. By analysing the results, we can infer the contribution of each feature to the upward or downward movement, note that as the models have only a maximum of 0.53 accuracy on testing, then some Shapley value don't reflect the true contribution.

Overall, the direction features seem to be the least significant, which can be attributed to the fact that, on the one hand, they are the only categorical features, and, on the other hand, they may not provide a significant signal since the Random walk theory suggests that changes in asset prices are random. However, a positive $Direction_5$, i.e. an increase of price since the previous week, has a positive impact on the upward movement probability at time t.

Overall, the most significant feature to a downward movement is ATR which measures market volatility by decomposing the entire range of an asset price for that period. In general, high volatility is an indicator of a bear market, however here I can see that low or high ATR values induce negative SHAP value, making ATR a powerful contrarian indicator.

For Apple, we can observe that the $MACD$ has a linear trend to the upward probability which seems to be related to the literature. Indeed, when the $MACD$ value is high, it generally indicates that the bullish momentum is strong and the price of the financial instrument is increasing, which is interpreted as a buy signal. OBV is also a significant feature to an upward movement, most of OBV values are high and the related SHAP values are positive. From the literature, a high OBV value suggests that the buying volume of a financial instrument is increasing and that there is more demand for it, which may be an indication of a bullish trend.

For Google, one can note that the higher the values of $Williams\%R$, $STOCK$ and $ChaikinMoneyFlow$ the higher they contribute to a downard probability.

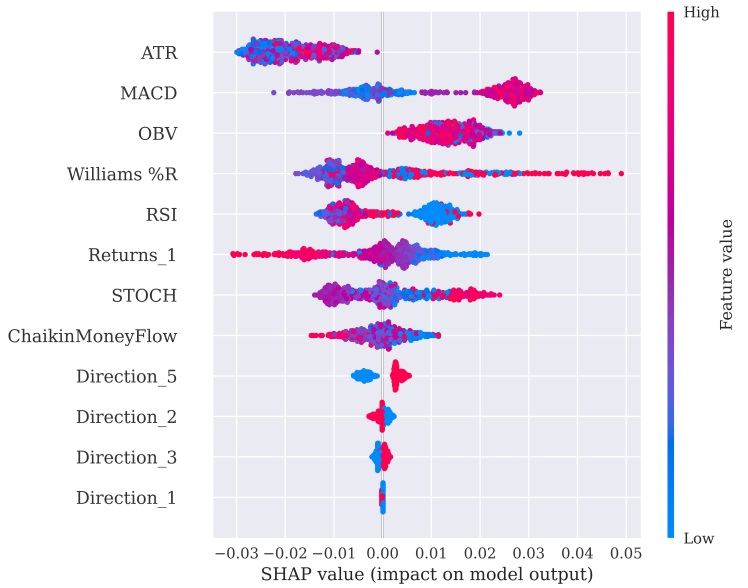

Table 7. Apple Summary plot on testing.

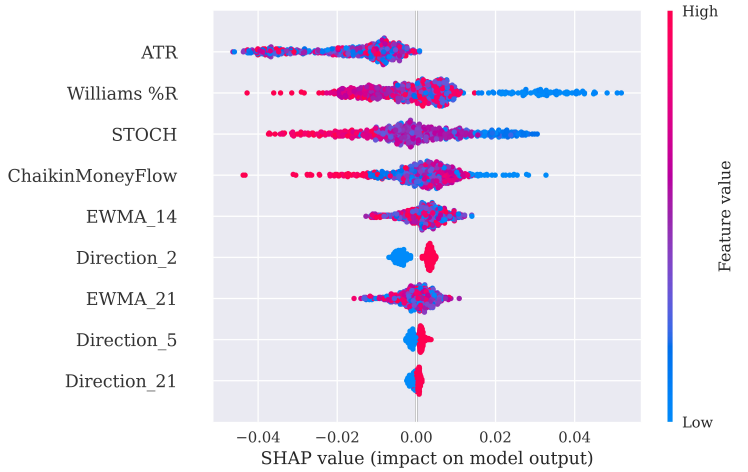

Table 8. Google Summary plot on testing.

In the literature, having high values these indicators indicate an overbought condition in a financial instrument, which may be an indication of a bear trend. For example, an high *ChaikinMoneyFlow* value suggests that there is a lot

of buying pressure in the market, which may be unsustainable and lead to a downward correction or reversal.

Hence, Shapley values allow for a global understanding of the model behavior which has traits in common with the trading decisions of an investor relying on the literature of technical indicators and market factors.

7 Conclusion

The increasing popularity of algorithmic decision-making systems in finance raises concerns about the potential negative consequences for those affected by their decisions. This study shows that automatic feature selection is a possible solution to promote greater reliability and robustness in explainable artificial intelligence. The paper also emphasizes the importance of understanding the global predictions of AI models through feature's contribution to the upward movement probability at each observation date. However, one must be careful not becoming overconfident about the forecasting ability of the model as it is generating false positive and false negative signals. Finally, as financial markets exhibit highly non-stationary behaviour, they are subject to large out-of-sample prediction errors if new patterns emerge.