Explainable AI for Financial Markets

Avinash K. Pandey
avinash.kumar.pandey@emory.edu
Finance PhD, Emory University
Atlanta, Georgia, USA

Swati Rajwal
swati.rajwal@emory.edu
CSI PhD, Emory University
Atlanta, Georgia, USA

Vishva Patel
vishva.patel2@emory.edu
CSI MS, Emory University
Atlanta, Georgia, USA

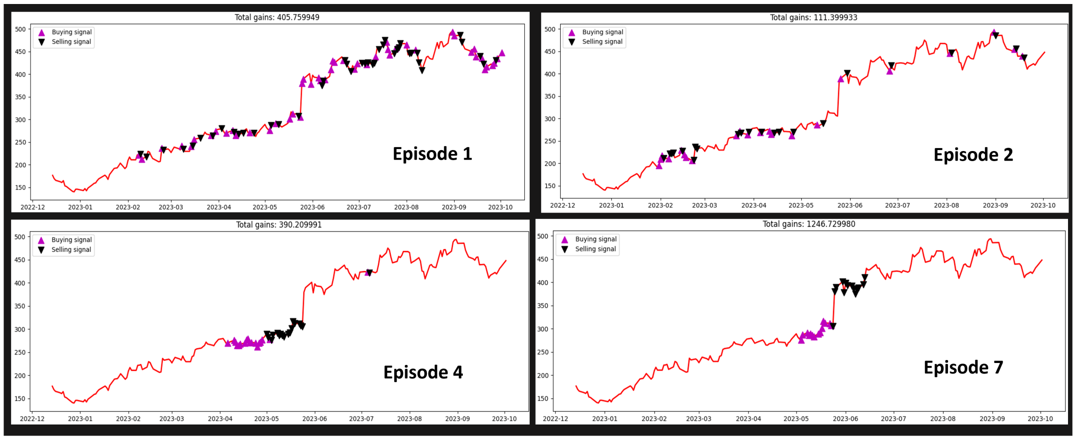

Figure 1: Reinforcement Learning Agent, Episodes v/s Total Gain

ABSTRACT

Investing strategies for mutual and pension funds are crucial for safeguarding the life savings of the general public. Traditional factor models have worked well but their limitations are becoming apparent with the rise of complex deep neural nets (DNN) and reinforcement learning (RL) agents. In this study, we aim to address the general problem of opacity in advanced financial models while providing higher returns. Here we show that by integrating Explainable Artificial Intelligence (XAI) algorithms with complex models, we can achieve higher returns and also gain valuable insights into the models' decision-making processes. Our experimental results show that DNN outperforms traditional linear models. This improvement is crucial, especially in the context of previous beliefs that higher returns necessitate a compromise on transparency. Such integration is beneficial for the fund management industry to ensure that complex, high-return models remain accessible to stakeholders which enhances trust in automated investment strategies. Furthermore, we also study higher frequency RL trading agent that incorporates multiple technical signals and demonstrates a successful convergence towards higher profits.

1 INTRODUCTION

1.1 Motivation

The choice of investment strategies for mutual and pension fund investments involves critical decision-making since the life savings of the general public are involved. These funds collectively manage money worth $23 Trillion USD which is far greater than the consolidated GDP of many nations. The majority of large institutional funds (Mutual Funds and Pension funds) in the United States use traditional linear regression-based factor models. With the advent of artificial intelligence, researchers have shown that models such as Random Forests and DNN. However, these higher returns come at the cost of employing 'black-box' models which are difficult to comprehend and trust in the high stakes situation. In this research project, we aim to combine the superior returns of complex models with transparency offered by Explainable Artificial Intelligence algorithms (XAI). By integrating these cutting-edge tools, we aim to contribute to the literature that ensures the public's savings are managed both effectively and responsibly.

1.2 Background & Related Work

The empirical literature on factor models has evolved from linear regression models on financial characteristics or 'factors' (CAPM by Sharpe [8], three-factor model of Fama and French [2], five-factor model of Fama and French [3], Stambaugh and Yuan [9] four-factor, and Hou et al. [5] Four Factor model and recently to Barillas and Shanken [1] six-factor model). However, recently there has been a growing interest in asset pricing models with a large number of financial factors that incorporate non-linear effects and interaction terms. Kozak et al. [6] used regularization-based regression models such as LASSO regression, Ridge regression, Elastic-net regression, and Principal Component models to achieve a superior linear factor model. Gu et al. [4] tackled the problem of high dimensionality by employing machine learning methods such as random forest, Principal Least Squares (PLS), Principal Component Models, etc. They also show that investors could generate higher sharper ratios by constructing portfolios using Neural Networks and Random Forest with relatively mild drawdowns compared to the widely accepted linear regression-based models. In conclusion, one of the common themes from these studies is that the machine learning and deep learning models that incorporate the interaction terms and non-linearity in the financial factors are able to outperform classical linear factor models. However, there is still a literature gap in understanding what drives the performance of these "black-box" deep learning models.We aim to address this research question using recent advances in explainable AI- methods such as SHAP [7] and PDP plots.

1.3 Problem Statment

In this work, we aim to design and develop an explainable AI-based pipeline that outperforms traditional linear fund strategies and is also explainable to stakeholders. We employ XAI methodology to understand the decision-making processes of complex algorithms such as Random Forest and DNN using SHAP (Shapely Additive Explanation Values) and PDP (Partial Dependency Plots). Furthermore, we design a reinforcement learning (RL) trading agent that utilizes price and volume-based technical indicators in conjunction with DNN to study higher-frequency AI models.

2 METHODOLOGY

2.1 Return Prediction Models

We trained predictive models (Random Forest, Linear Regression, and Deep Neural Networks (DNN)) on a rolling window basis. These models are trained over a period of 60 months to ensure that the models capture temporal dynamics and are robust to different market conditions. The training process used square root features and a maximum tree depth of 5 for the Random Forest algorithm, to prevent overfitting and to facilitate model interpretability. We also employed SHAP to extract the relative importance of features. Additionally, we generate distributional scatter impact plots and dependence plots. These tools enable us to visualize and comprehend the contribution of individual variables to the model's predictions which is crucial for the validation of the investment strategies.

2.2 Reinforcement Learning Trading Agent

For studying AI models that make trading decisions on a higher frequency, we constructed a reinforcement learning (RL) trading agent that utilizes technical indicators and a deep neural network to make informed trading decisions, specifically focusing on buying low and selling high. The model is a deep neural network constructed using Keras. It comprises an input layer that accepts five technical indicator outputs (MACD, CCI, RSI, Stochastic Oscillator, VWAP) and current closing price, three hidden layers with 64,32 and 8 neurons respectively with ReLU activation function, and an output layer with three units representing 'buy', 'sell' or 'hold' decisions. The network is compiled with a mean squared error loss function and an Adam optimizer. The trading agent uses a Deep Q-learning approach with an epsilon-greedy policy for action selection. The agent's memory, implemented as a deque with a maximum length of 1000, stores experiences for replay. At each time step, the agent observes the current state (derived from the technical indicators and the closing price) and decides whether to buy, sell, or hold. The agent receives a reward based on the profit or loss from each action. The model is trained iteratively over multiple episodes, with the aim of maximizing cumulative rewards (total gains).

2.3 SHapley Additive exPlanations or SHAP

SHAP [7] is a game theory-based approach used to explain the output of machine learning models. This approach enhances interpretability by consistently quantifying feature importance which offers an intuitive understanding of model behavior. The foundation of SHAP is the Shapley value from cooperative game theory, which distributes the 'payout' fairly among the 'players'. Mathematically, for a prediction model f and a particular instance x, the SHAP value ϕ_j for feature j is defined as:

$$\phi_j = \sum_{S \subseteq N \setminus \{j\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} \left[f_x(S \cup \{j\}) - f_x(S) \right] \quad (1)$$

Here, N is the set of all features, S is a subset of features excluding j, $|S|$ is the cardinality of set S, $f_x(S)$ is the prediction for the feature subset S, and the term $f_x(S \cup \{j\}) - f_x(S)$ represents the marginal contribution of feature j when added to the subset S.

3 EXPERIMENTAL SETUP

3.1 Dataset

We considered firms listed in the NYSE, AMEX, and NASDAQ. The firm's monthly returns data were collected from the CRSP database from March 1975 till December 2022. A total of 30,000 firms were used in the analysis, with a total of 1.5 million observations. We also constructed 94 stock-level characteristics using the COMPUSTAT database. These financial features have been shown to have predictive power on stock returns in the finance academic literature. Our dataset is divided into three disjoint time series data on a rolling basis with a training window size of 60 months, rolling validation sample (following 6 months), and out-of-sample data.

For the RL trading agent, data is imported using yfinance library, which provides historical stock prices. The study focuses on NVIDIA Corp. (ticker: NVDA), with data spanning the most recent 365 days. The dataset includes opening , closing , highest and lowest price of the day, and trading volume. For features to the RL trading agent, five technical indicators namely: MACD, CCI, RSI, Stochastic Oscillator, and VWAP are calculated as input in addition to the current day's closing price. Please refer to the appendix for more information on the technical indicators.

3.2 Baseline & Evaluation Metrics

We used linear regression as a baseline since it has been predominantly used both in academic literature and fund-management industries. We compare the results from the linear regression model with that of random forest and deep neural network. For the reinforcement learning agent, the baseline is the model performance on episode zero with a predominantly large number of random trades. We then compare the performance with the trading agent which uses price and volume-based technical indicators at different stages of training (refer Figure 1).

For evaluation of model performance (Linear, RF, DNN), out-of-sample mean returns, standard deviations, and the t-statistics values of various portfolio ranks are reported. For the evaluation of the RL trading agent, total gains for various episodes are plotted. However, the quality of trades (e.g. is the model able to identify low-price regions consistently for buying and vice versa for selling)

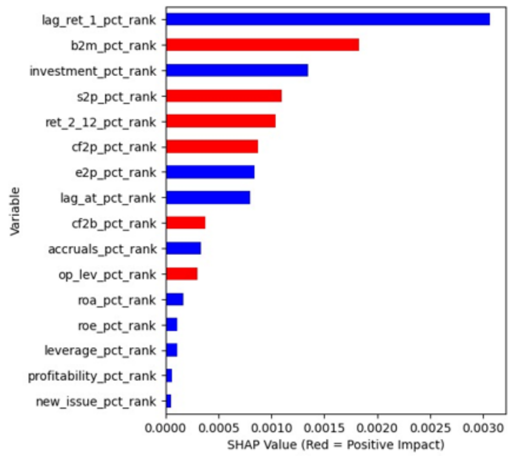

Figure 2: Feature importance RF

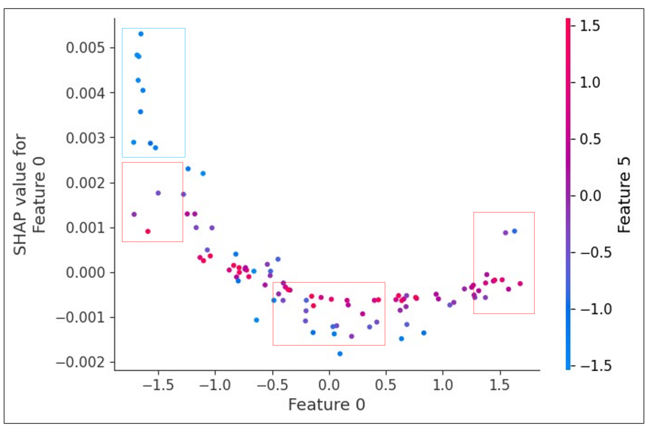

Figure 3: Partial Dependence Plot DNN

seems to be a more important evaluation metric.Finally, we use SHAP summary plots and PDP plots to evaluate and understand the underlying decision-making criteria of these models.

4 RESULTS AND DISCUSSION

Predictive Models: Table 1 shows the results of our analysis in which we focus primarily on Linear Regression (benchmark), Random Forest, and DNN. We trained the model on a rolling 60-month period and predicted the future monthly returns of the stocks. Based on the predicted returns, we sort the stocks into ten deciles or portfolio ranks. Results show that both Random Forest and DNN outperform the Linear Regression model. Interestingly, the DNN model is superior in identifying the worst-performing stocks as evidenced by the significant negative in the first two deciles. The "Diff" row (Table 1) is the strategy where the fund buys the winning stocks and sells (or shorts) the losing stocks. We believe that DNN outperforms the linear and RF models potentially due to its ability to capture non-linearity and interaction effects.

SHAP Analysis: Figure 4 provides an aggregate measure and direction of each feature's importance in a more granular fashion.

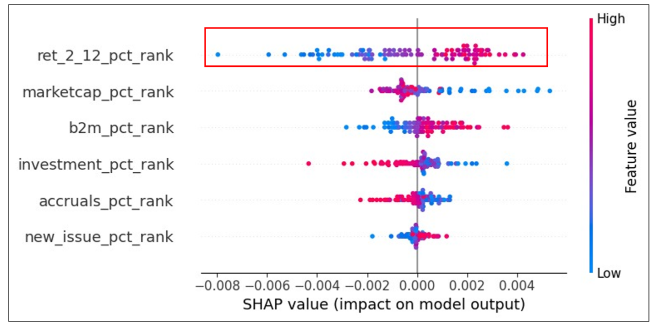

Figure 4: SHAP Summary Plot for Deep Neural network

The most prominent features influencing monthly returns were 'ret_2_12_pct_rank' (momentum factor), 'marketcap_pct_rank' (size rank), and 'b2m_pct_rank' (Book-to-market ratio). Our results suggest that DNN prioritizes small stocks with high momentum and book-to-market values with low accruals and low issuance of recent shares. The results in our analysis provide insight into DNN decision-making and are in line with academic literature.

Interaction Term Visualization: We will briefly discuss one of the Partial Dependence Plots (PDP) regarding the interaction effect of market size (Feature 0) and momentum (Feature 5) with monthly returns. In Figure 3, we note that in the top left region, small stocks with lower momentum values correspond to higher SHAP values. This indicates that smaller companies with less recent price movements are considered valuable investments by the model. In the middle and right regions, we observe positive returns for the stocks with high momentum values suggesting that momentum is more important for mid-cap and large-cap stocks. These findings suggest a complex interaction between the size of a company in the market and its recent performance trend with implications for investment strategies.

RL Trading Strategy: Figure 1 shows RL agent's evolution in trading strategy across different episodes. In episode 1, the RL agent explores the action space extensively, indicated by frequent buys and sells and a gain of 405 USD. This behavior is consistent with the early stages of training, where exploration is predominant due to a higher epsilon value (leading to more random decisions). The agent's trading frequency then significantly reduces, suggesting a shift from exploration to exploitation. In later episodes, the agent's actions become more consistent, with longer periods between buying and selling. However, the profits from these trades are not maximized, indicating that while the agent is learning, it has not yet fully optimized its strategy. Finally, the RL agent appears to buy consistently just before significant price increases and sell just after the price peaks, resulting in higher total gains of 1246 USD. Over time, as the epsilon value decays, the agent transitions from a phase of high exploration to one of high exploitation, refining its strategy to increase the total gains from trades.

4.1 Error Analysis

For error analysis, we plot SHAP summary values on a rolling basis on different small sub-sample periods. The SHAP summary plots on two different training periods during the 1980s and 1990s

	Linear Regression			Random Forest			Deep Neural Network		
Portfolio Rank	**Mean Ret.**	**Std**	**t-Stat**	**Mean Ret.**	**Std**	**t-Stat**	**Mean Ret.**	**Std**	**t-Stat**
0	0.704192	8.172514	1.614317	0.289818	8.516631	0.637546	-2.885486	19.297503	-2.104023
1	0.850636	6.950077	2.293018	0.787706	6.972445	2.116570	-1.806788	16.580387	-1.533363
2	0.866521	6.303476	2.575446	0.907640	6.446111	2.637969	-0.010345	12.476946	-0.011667
3	0.991166	5.998124	3.095882	1.080264	6.011265	3.366802	-0.512313	10.594428	-0.680441
4	1.103684	5.737527	3.603907	1.149094	5.700220	3.776744	0.046836	9.991868	0.065958
5	1.146138	5.772467	3.719881	1.204434	5.451576	4.139180	0.598480	7.580705	1.110893
6	1.311835	5.682504	4.325069	1.243976	5.363142	4.345564	1.246805	6.167876	2.844433
7	1.470676	5.758554	4.784725	1.337808	5.823150	4.304170	2.248423	14.023947	2.256007
8	1.607909	5.994418	5.025370	1.489129	6.057587	4.605599	5.274351	18.877229	3.931546
9	**2.175578**	7.212994	**5.650837**	**2.795485**	8.046111	**6.509158**	**9.799533**	37.205317	**3.706235**
Diff	1.471387	6.472405	4.259069	2.536452	5.582490	8.512404	12.685019	35.960093	4.963670

Table 1: Performance Results for Predictive Models

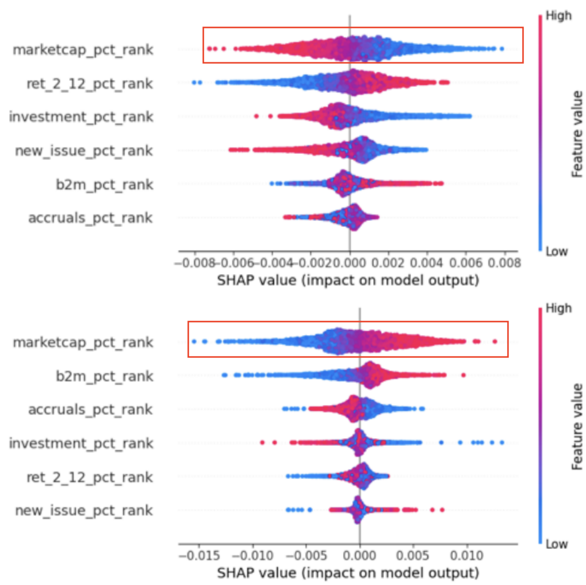

Figure 5: Error Analysis

in Figure 5 reveal the time-varying impact of factors such as the market size of the stocks on monthly returns. In the first SHAP summary plot (top plot), we observe a negative impact of market capitalization on the model's output corresponding to a period where small growth stocks are preferred to the model. Conversely, during the second time (bottom plot), we observe a positive impact of firm size corresponding to a period where large stable stocks are preferred. Also, it is necessary to train the model on large enough data to capture time-varying trends and avoid any small sample bias. Therefore, investment strategies based on static factor models trained on small training periods may potentially lead to sub-optimal investment decisions.

4.2 Ablation Study

The analysis of the ablated features and their corresponding mean values at portfolio rank 9 provides insightful information regarding the importance of these features in predictive models (Appendix 2). A higher mean value at rank 9 suggests that the absence of the feature does not significantly impair the model's performance, indicating its lower importance. Conversely, a lower mean value at rank 9 implies greater importance of the feature, as its absence markedly degrades the model's performance. We observe that features such as 'lag_at_pct_rank' (assets) and 'op_lev_pct_rank' (operational leverage), with lower rank 9 mean values of 10.781 and 10.114 respectively, seem to be more significant which is in line with our SHAP analysis. The lower mean values indicate a noticeable drop in the model's performance when these features are not included, highlighting their importance. In summary, this analysis aids in identifying which features are more or less crucial for the accuracy and effectiveness of the predictive model.

5 CONCLUSION

Our experimental analysis confirms the effectiveness of using ML techniques over traditional factor models for financial market investments. Our trained models, especially the Deep Neural Network, not only resulted in higher returns but also provided insights into the decision-making process through Explainable AI methods like SHAP and PDP. The DNN's ability to identify interactions such as the outperformance of high-momentum small stocks and its sensitivity to the book-to-market values proved critical in outperforming standard models. Furthermore, our reinforcement learning trading agent demonstrated a successful transition from exploration to exploitation, enhancing trading profits over time. We also highlight the importance of features such as size, momentum, investments, and book-market ratio through our SHAP analysis. This study not only bridges the gap between complex predictive models and their interpretability but also paves the way for more responsible and effective management of public savings in the investment sector. Through continuous adaptation and learning, the models showed potential for capturing time-varying market trends which is crucial for long-term investment strategy development.

CODE AVAILABILITY

Python 3 was used for all the coding scripts which can be located in this repository: https://github.com/cosinesimilarity1/CS-557.

ACKNOWLEDGEMENTS

We would like to thank Prof. Eugene Agichtein for teaching the coursework CS 557. This work has been inspired by the concepts learned during the lectures.

REFERENCES

[1] Francisco Barillas and Jay Shanken. 2018. Comparing Asset Pricing Models. *Journal of Finance* 73 (2018). Issue 2. https://doi.org/10.1111/jofi.12607
[2] Eugene F. Fama and Kenneth R. French. 1993. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics* 33 (1993). Issue 1. https://doi.org/10.1016/0304-405X(93)90023-5
[3] Eugene F. Fama and Kenneth R. French. 2015. A five-factor asset pricing model. *Journal of Financial Economics* 116 (2015). Issue 1. https://doi.org/10.1016/j.jfineco.2014.10.010
[4] Shihao Gu, Bryan Kelly, and Dacheng Xiu. 2020. Empirical Asset Pricing via Machine Learning. *Review of Financial Studies* 33 (2020). Issue 5. https://doi.org/10.1093/rfs/hhaa009
[5] Kewei Hou, Chen Xue, and Lu Zhang. 2015. Digesting anomalies: An investment approach. *Review of Financial Studies* 28 (2015). Issue 3. https://doi.org/10.1093/rfs/hhu068
[6] Serhiy Kozak, Stefan Nagel, and Shrihari Santosh. 2020. Shrinking the cross-section. *Journal of Financial Economics* 135 (2020). Issue 2. https://doi.org/10.1016/j.jfineco.2019.06.008
[7] Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems* 30 (2017).
[8] William F. Sharpe. 1964. CAPITAL ASSET PRICES: A THEORY OF MARKET EQUILIBRIUM UNDER CONDITIONS OF RISK. *The Journal of Finance* 19 (1964). Issue 3. https://doi.org/10.1111/j.1540-6261.1964.tb02865.x
[9] Robert F. Stambaugh and Yu Yuan. 2017. Mispricing factors. *Review of Financial Studies* 30 (2017). Issue 4. https://doi.org/10.1093/rfs/hhw107

APPENDICES

A ADDITIONAL RESULTS

Ablated Feature	Mean Returns of Best Portfolio
Full Model	18.571142
op_lev_pct_rank	10.114654
lag_at_pct_rank	10.781856
lag_ret_1_pct_rank	15.070012
s2p_pct_rank	14.295363
accruals_pct_rank	17.475780
b2m_pct_rank	17.975681
roa_pct_rank	18.258998
roe_pct_rank	18.665869
ret_2_12_pct_rank	19.119816
e2p_pct_rank	20.763561
cf2p_pct_rank	22.520341
profitability_pct_rank	23.730783
new_issue_pct_rank	25.156532
cf2b_pct_rank	26.011038
leverage_pct_rank	26.742425
investment_pct_rank	27.071372

Table 2: Portfolio Rank 9 Mean Values for Ablated Features

TECHNICAL INDICATORS

A. Moving Average Convergence Divergence (MACD)

- **Default Values:** period_short=12, period_long=26, signal=9
- **Description:** The MACD is a trend-following momentum indicator that shows the relationship between two moving averages of prices. It is calculated by subtracting the 26-period Exponential Moving Average (EMA) from the 12-period EMA, with a signal line to indicate buy or sell points.

B. Commodity Channel Index (CCI)

- **Default Values:** n=20
- **Description:** The CCI is used to identify new trends or extreme conditions of a security. It measures the difference between the current Typical Price (mean of high, low, and close) and its average over a set period, normalized by the mean deviation.

C. Relative Strength Index (RSI)

- **Default Values:** period=14
- **Description:** RSI is a momentum oscillator that measures the speed and change of price movements. It oscillates between zero and 100 and is typically used to identify overbought or oversold conditions in trading assets.

D. Stochastic Oscillator

- **Default Values:** k_period=14, d_period=3
- **Description:** This momentum indicator compares a security's closing price to its price range over a given time period. The sensitivity of the oscillator to market movements is reducible by adjusting the time period or by taking a moving average of the result.

E. Volume Weighted Average Price (VWAP)

- **Default Values:** Not Applicable
- **Description:** VWAP is used by traders to gauge the market direction and filter trade execution. It represents the average price a security has traded at throughout the day, based on both volume and price. It is often used as a trading benchmark.

Received 15 December 2023