Deep sector rotation swing trading

Joel R. Bock[1] and Akhilesh Maewal[2]

[1]Independent, New Braunfels, Texas, USA
[2]Independent, San Diego, California, USA

January 4, 2023

Abstract

A system for sector rotation swing trading of exchange-traded funds (ETFs) using deep learning is presented. Weekly trades are made on funds representing 11 major sectors of the U.S. economy. The trading system was backtested for the period January 2012 through December 2022. Annualized CAGR returns exceeded the benchmark buy-and-hold strategy by an average 12.63% (median 7.63%). Of particular note is the positive alpha ($\alpha = 28.4\%$) achieved in trading for 2022, a difficult year for stocks in which the S&P 500 index experienced a CAGR loss of 18% .Over the studied period, Sharpe ratios averaged 1.39, and the mean maximum drawdown was 10%. The deep model design is multiple-input, multiple output, and can be easily extended to include other factors that may influence predictability of future price movements. The results presented here are preliminary, and are exclusive of trading costs. Analysis of these costs is prerequisite to deployment as a semi-mechanical swing trading system.

1 Introduction

Sector rotation refers to a portfolio rebalancing strategy wherein money is moved from one economic sector into another, to profit from differential financial performance of different sectors mapped to stages of the business cycle [3]. The cycles follow the sequence: [*trough*, *expansion*, *peak*, *contraction*], before repeating again. The idea is to identify and anticipate the timing of each phase based on macroeconomic indicators, and allocate assets accordingly [10].

For quantitative trading, business cycle timing is problematic. Incorrect prognosis may result in trading out-of-phase, yielding capital losses, or at least underperformance relative to a simple long position in a broad market index. In studies of quarterly switching between index funds of two asset classes (U.S. stocks and bonds) over the period 1993-2017, it was concluded that the optimal timing of rotation between classes was ``indistinguishable from a random sequence" [16]. Even given perfect forecasting of cycles, researchers found at best a 2.3% outperformance relative to the market using a conventional sector rotation strategy [12]. Their study covered multiple economic expansions and contractions over the years 1948–2007.

Industrial momentum strategies for sector rotation eliminate the need to time the market, and may produce outperformance, especially when rebalanced in the short term [17]. In one recent study, industrial momentum strategies with various holding periods were examined on a large historical sample spanning years 1926-2018. U.S. industry portfolios were sorted into quintiles of returns from the previous month. Portfolios comprising industries with the highest previous-month returns were found to produce significantly greater profits than the groups having underperformed in previous month [11].

The present work takes a different strategic approach to short-term sector rotation trading. Deep learning models are developed to inform buy decisions on exchange-traded funds (ETFs) representing the major sectors U.S. economy. For each trading year, a multi-input, multi-output model is trained on recent historical price, volume and auxiliary economic data to make recommendations on which sectors are most likely to increase in value over the next trading week. Selected funds are ranked, curated and bought at Monday market open, and all holdings are liquidated at close of the current trading week. Within-year models are incrementally updated each week. In the short term, ETF price movements exhibit volatility; this can be exploited for tactical asset allocation.

Results of backtesting the current trading strategy are presented for the period January 2012 through December, 2022. Exclusive of trading costs, CAGR outperformance over the benchmark S&P 500 index was found to be meaningful (mean $\alpha = 12.63\%$, median = 7.63%).

Related work

Machine and deep learning techniques as applied to portfolio allocation and trading are abundant in the literature. A selected review of reports relevant to the present work is provided here.

Navon and Keller [18] developed a deep learning-based system to predict price trends of stocks and ETFs. These predictions were used for intraday trading using a ``buy-sell-hold" strategy. Potential open positions were evaluated for likelihood of profitability by thresholding probabilistic neural network outputs. Positions were closed based on forecast changes in price trend. Backtesting over two years, cumulative returns from trading certain large-cap stocks generally outperformed broad market index. The present research was informed by [18] where it was suggested that prediction accuracy could be improved by making dynamic model updates using all available at a point in time. This idea is implemented in the strategy reported here.

In Liew and Mayster [15], ETF price movements were forecast using machine and deep learning models. Prediction horizons of one and three months' duration were found to produce best results in terms of a ``gain criterion" comparing model price forecast accuracy against random inputs. No explicit financial results were reported. One conclusion was that ETF price movements were essentially unpredictable at very short horizons (one to five days) [15]. Findings in the present report contradict this assertion; it will be shown that over a one-week trading horizon, profitability is indeed possible using a deep learning strategy.

Chalkidis and Savani [5] applied several machine learning techniques (including random forests and deep networks) to financial time series prediction. Models were trained to output one of three decisions on price movement--down, up, or slightly in either direction, corresponding to short, long and abstain positions. A trading strategy for commodities futures based on these ``selective classification" models with different feature sets was designed, and subjected to backtesting. By including an ``abstain" position, the selective classification approach outperformed non-selective (binary) classifiers, and resulted in smaller capital losses in trading due to reduced misclassifications.

Pinelis and Ruppert [20] used random forest models for portfolio adjustment on a monthly basis. Their models simultaneously forecast both expected returns and volatility. Results produced improved returns ($\alpha = 3.37\%$) over a buy-and-hold strategy. The authors speculate that further performance gains might be achieved by use of deep learning.

Sauer [21] studied investment strategies, using random forests to identify economic regimes based on monthly GDP data. Portfolio sector weights between cyclical and defensive assets were modified according to the identified regime. It was concluded that for trading, machine learning did not significantly improve results compared to a naïve strategy using equal weights for sectors.

Additional work on deep learning methods for financial applications is surveyed in [19].

Present contribution

The key contributions made in this research are proposed as follows:

- A robust system for sector rotation swing trading using deep learning. A single model architecture, set of training hyper-parameters and trading rules generalizes to all years considered (2012-2022).
- Consistent outperformance over the benchmark buy-and-hold strategy.
- A statistically-based estimate of confidence in each putative trade.
- Coverage includes all major sectors of U.S. economy, with each sector potentially bought in a trading week. Custom models are developed within each year, and updated dynamically using the latest observed price and volume data.
- The deep model design is multiple-input, multiple output and extendable to include other technical indicators or economic factors that may impact future price movements.

2 Methods

The trading system comprises three elements: (1) historical price and volume data for major economic sector ETFs; (2) a deep learning model trained to forecast future price movements using this data; and (3) a trading strategy based on the out-of-sample predictions made by the model.

2.1 Data preparation

The ETFs selected as the basis for development of the trading strategy in this study are listed in Table 1. As a whole, these funds cover the major industrial sectors of the United States economy. Price and volume data for each fund were acquired from Yahoo Finance[†].

In addition to the ETF price and volume data, ancillary economic data (10 year U.S. Treasury yield, USD currency index, crude oil proxy and market volatility indicators) were collected, under the hypothesis they might provide additional informative context to the model. In some experiments, these extra data were as model inputs, and were not actively traded quantities. The final results reported here are based only on model inputs comprising the sector funds of Table 1.

Sector ETF and auxiliary data were sampled at discrete points in time, and filtered to exclude all but ``Friday close" prices. Non-trading days were removed from the data.

Let the cardinality of traded ETFs and auxiliary input data be represented by l and m respectively. Training examples for week t are constructed by assembling matrices containing prices, volumes and other data recorded for the current and previous $N-1$ weeks. Each input data matrix X_t has dimensions $N \times (l+m)$[‡]. Length l binary label vectors y_{t+1} describing the percentage increase in prices in week $t+1$ are assigned to each matrix, resulting in a set of examples $z = (X_t, y_{t+1})$.

All non-target data were normalized to have zero mean and unit variance.

The present swing trading scheme liquidates all holdings upon weekly market close. Training labels y signify price movement for the succeeding week-- an uptick in price signals that the associated ETF should be bought. A threshold increase of 100 basis points was used to assign these labels for the classification task posed to the model during training. By defining a target value in this manner, the trading model was charged with learning to implicitly predict two distinct future values (Monday open and Friday close).

Input data are shown schematically in Figure 1. For clarity, quantities not subject to forecast by the trained model (volumes and optional accessory input data) are not represented in this figure.

Sector name	**Symbol**
Information Technology	XLK
Health Care	XLV
Consumer Discretionary	XLY
Communication Services	VOX
Financials	XLF
Industrials	XLI
Consumer Staples	XLP
Utilities	XLU
Materials	XLB
Real Estate	IYR
Energy	XLE

Table 1: Sectors and their ETF symbols.

2.2 Deep learning model

A deep learning network model was developed to forecast future price movements of ETFs and inform swing trading decisions. The model was designed to process multiple inputs and make predictions for each sector-based asset as described in Section 2.1. After considerable experimentation with different architectures, the final model used here was composed of four fully connected internal layers, each followed by $ReLU$ activation [14] and dropout [24], now standard techniques used to mitigate vanishing gradients and over-fitting on the training data, respectively. The output layer was likewise densely connected, and was terminated with a linear activation function.

Model optimization and dynamic update

Individual models were trained and evaluated for trading years from January 2012 through December 2022 . Each model was fit using historical data from the previous 2 years. After training, the models were used to recommend purchases of sector ETFs once per week in the succeeding trading year. All current holdings were liquidated just

[†] `https://finance.yahoo.com`

[‡] The dimensions are $N \times (2l + m)$ if ETF volumes are used in the analysis.

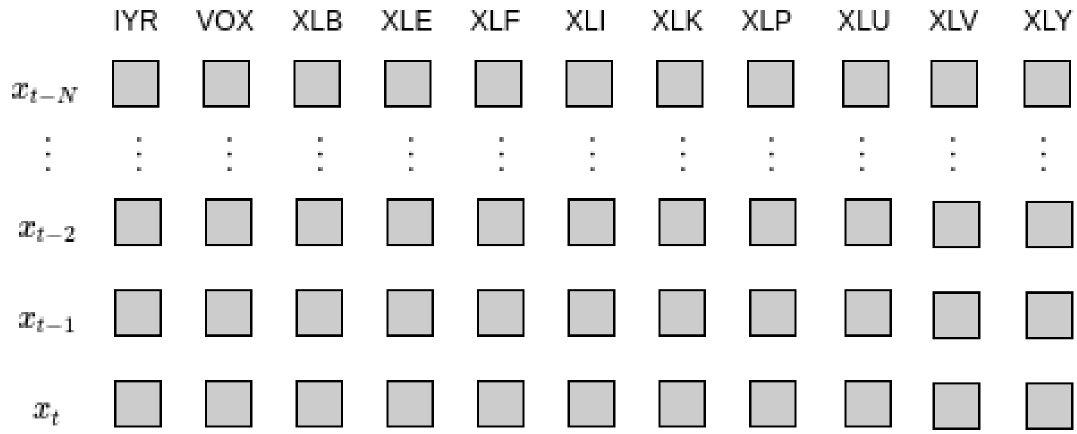

Figure 1: Input data scheme for the trading model. Columns represent traded assets as listed in Table 1. Each box along a row represents a Friday closing price x_t for week t. N is the history depth. In this example, the deep model will learn to suggest trades to be executed during the following week $t+1$. A complete data example comprises one such frame of data and a target label vector containing one number per traded asset. Volumes and optional accessory data (non-forecast quantities) are omitted for clarity.

prior to closing of the trading week, and profits & losses recorded. Once trades were completed, the models were updated incrementally using the observed price data from the week just ended, and used to generate signals for ETF buys in the following week.

To connect dynamic model optimization with financial performance, a custom loss function incorporating recent capital gains and losses was used to steer model weights updates in the direction of profitable trades. The idea of directly using a financial criterion was suggested in [4].

Models were developed used the TensorFlow platform [1] and the Keras deep learning API [6].

Evaluation of trading performance

At trading year-end, overall performance of executed trades was evaluated using financial performance statistics including compound annual growth rate (CAGR), the Sharpe ratio [22], and the maximum drawdown of the portfolio. Present results are compared against the benchmark strategy of full investment in the S&P 500 (SPX) for a given trading year. As defined here, the excess return of the deep models DL is

$$\alpha = CAGR_{DL} - CAGR_{SPX}$$

where $CAGR_{SPX}$ includes reinvestment of dividends[§]. This provided an objective means to assess practical value added of the machine trading approach.

The risk-free return used to estimate Sharpe ratios was based on contemporaneous 90-day U.S. Treasury yields. All trading was performed on out-of-sample data.

2.3 Swing trading

The strategy for weekly sector rotation trading is summarized as follows:

- Immediately prior to Friday closing (week t):
  - **Sell**: Exit all positions from week $t-1$.
- After closing (week t):
  - **Analyze**:
    1. Execute deep model, and select potential ETFs to buy in the next week.

[§]Source: `https://bit.ly/3iavw0u`

2. Assign confidence metric to distribution of predictions for each asset.
3. Apply loss reduction heuristics to refine the selection set.
4. Rank funds in list, and allocate available capital.
5. Update model using the observed week data x_t.

- On Monday open (week $t+1$):
  - **Buy**: Buy sector ETFs as designated in week t.

Generation of ``buy" signal

The trading model produces a vector of real numbers, one for each sector ETF from the collection listed in Table 1. A preliminary buy decision is made by comparing each output value to a threshold value unique to the corresponding asset. Surpassing this value is taken as a ``buy" signal; otherwise, the fund is not bought. The threshold vector is updated dynamically after each trading week, and is estimated by optimizing a cost function based on receiver operating characteristic (ROC) curve analysis [8] of recently observed input data.

Scheduled selling

All positions are liquidated just before market close in the current trading week. While perhaps counter-intuitive, this systematic approach eliminates potential situations where even the professional investor's selling decisions may perform worse than a random selling strategy [2].

Confidence estimation

It has been noted that neural network predictions can be ``...very noisy and unreliable"[4]. This assertion was corroborated in the present research, where point estimates of future price movements were observed to be highly variable, and distributed non-uniformly over successive experimental runs. A confidence measure was implemented by using a technique termed ``Monte Carlo dropout" [9] applied to the output layer in the model. A large number of predictions is made for each asset, for each trading week; each prediction is made with a random group of connection weights nullified. In effect, this produces an ensemble of deep models. For each asset s, the statistical distribution of predictions $\hat{y}_s$ for the ensemble is analyzed. Greater confidence is assigned to forecasts where a significant portion (here, 80%) of the probability mass of the distribution $\hat{P}_s$ is located within one standard deviation of the population median. A dispersed distribution in contrast presents greater uncertainty, and the associated asset is not purchased.

Loss reduction heuristics

Greater psychological discomfort accrues from losses than does pleasure realized from equivalent financial gains [25]. Loss aversion strategies are essential for swing trading to limit short-term losses. Additional filters may be applied to the ETFs marked for purchase in the ensuing week. Filters are parameterized to align with trader's level of loss aversion. These include loss limits at the single trade or complete portfolio level. Actions taken upon exceedance may dictate removal of a symbol from the buy list, or halting of all trading for the current week.

Stop limiting conditions and consequent strategic actions used in the backtesting experiments are summarized in Table 2.

Condition	**Value**	**Time**	**Action**
Recent loss by symbol	5%	week	Remove ETF from buy list
Maximum loss, week-week	\$300	week	Halt trading one week
Portfolio underwater	5%	Q4	Halt trading one week
Maximum loss by symbol	27.5%	Q1-Q4	Remove ETF from buy list
Minimum win rate by symbol	45%	Q4	Remove ETF from buy list

Table 2: Loss mitigation conditions and resulting actions used in backtesting experiments.

Dynamic allocation of capital

Prior to a trading week, available capital is distributed to each sector fund based on total average investment performance to date (win rate), and on momentum of trades in the sector [17]. Define a ``win" as a profitable ETF trade at close of the position. Allocation weights for sector s in the next period are computed by the equation

$w_s = 1.0 + \frac{wins_s}{buys_s} + \frac{streak_s}{wins_s+1}$ where $wins_s$ and $buys_s$ are the total win count and number of weeks fund s was bought, respectively; $streak_s$ is the number of consecutive $buys$ producing a profitable outcome (not necessarily contiguous weeks).

3 Results and discussion

Trading performance

Main results from backtesting the current trading strategy are presented in Table 3. Deep sector rotation produced on average excess returns α of 12.63% per year (median = 7.63%) over the benchmark for 11years studied. The strategy produced positive excess returns each year, the singular exception being 2012, where underperformance was 75 basis points. Of particular note is the positive alpha ($\alpha = 28.4\%$) achieved in trading for 2022, a difficult year for stocks in which the S&P 500 index experienced a CAGR loss of 18% .

Sharpe ratios averaged 1.39, indicating reasonable return-to-risk from trades execution. Overall, $\sim$60% of executed trades were found profitable. Maximum drawdown is observed to be, on average, roughly 10%. The right-most column in Table 3 displays selectivity of the trading system--only 2.25 out of 11 potential sectors are bought per week. This is a result of the procedures used for filtering putative trades as discussed in Section 2.3.

Specific sectors traded for each year in the backtesting period are summarized in Appendix A.

Year	**CAGR**		**α**	**Sharpe**	**MaxDD**	**Wins**	**#Buy/wk.**
	Current	**SPX**					
2012	15.13	15.88	-0.75	1.76	-0.11	60.51	3.02
2013	47.57	32.43	15.14	1.87	-0.09	63.91	3.31
2014	21.44	13.81	7.63	1.09	-0.07	61.90	1.65
2015	2.72	1.31	1.41	1.46	-0.09	50.00	2.78
2016	54.41	11.93	42.48	1.36	-0.07	63.11	1.98
2017	22.48	21.94	0.54	1.61	-0.06	63.08	2.50
2018	-3.18	-4.41	1.23	0.94	-0.15	54.55	1.27
2019	42.46	31.74	10.72	1.91	-0.05	69.33	2.94
2020	25.00	18.38	6.62	0.95	-0.10	61.25	1.57
2021	54.38	28.83	25.55	1.38	-0.07	60.66	2.35
2022	10.29	-18.11	28.40	0.97	-0.21	57.58	1.53
Mean	26.61	13.97	**12.63**	1.39	-0.097	60.62	2.25
Median	22.48	15.88	**7.63**				

Table 3: Backtesting results for the analyzed period January 2012 through December 2022. *CAGR*: compound annual growth rate (%); α: excess return of *Current* vs. *SPX* naïve strategies (%); *Sharpe*: Sharpe ratio; *MaxDD*: maximum drawdown; *Wins*: trade win %; *#Buy/wk*: average number of ETF trades per week.

Discussion

Trading results from the present deep strategy demonstrate non-trivial outperformance for over one decade of backtesting. By using only two years' training data, and incrementally refreshing the models with newly observed data at the end of each trading week, the variance between backtesting and out-of-sample performance can be mitigated. Previous authors [26] suggested that too many backtests increase overfitting, reducing the generalization ability of financial forecasts. That observation was based on a ``fit-once, forecast all" approach, which does not apply to the continuous updating trading scheme followed here.

The densely-connected multiple-input, multiple-output model learns to approximate a function connecting prices and volumes in each major sector of the economy with their future values. Nonlinear interactions between prices across (perhaps linearly uncorrelated) sectors are represented. Results obtained here agree with previous findings of evidence for a universal structure of price formation in high-frequency trading, where deep models trained on all stocks were seen to outperform stock-specific ones [23].

Regarding labeling of the training data, declaring a minimum 1% increase in next week's price as the target positive class was arrived at in order to approximately balance the examples for classification. A more accurate threshold value could be determined with further analysis. The model learns to approximate the target function even with misclassification errors, in part due to the error signal provided by the loss function which includes capital gains or losses from the previous weeks' trading.

Returns reported here are exclusive of ETF trading costs. Commissions (minimal), changes in underlying net asset values, operating expenses, and bid/ask spreads all affect the total cost of ETF ownership. For active trading as considered here, bid/ask spreads are probably the most consequential [7]. Tax liabilities may be incurred from capital gains in taxable trading accounts. Future extensions of this work should a realistic assessment of transaction costs, e.g., following analysis reported in Jensen *et al.* [13].

4 Conclusion

This paper presents a strategy for weekly sector rotation swing trading ETFs using deep learning. Coverage of potential trades each week includes 11 major sectors of the U.S. economy. A single model architecture, set of training hyper-parameters and trading rules was used in backtesting covering the period January 2012 through December 2022. The model design is multiple-input, multiple output and can be easily extended to include other technical indicators or economic factors that may influence predictability of future price movements.

Based on annualized CAGR, returns exceeded the benchmark strategy on average by 12.63% (median 7.63%). Sharpe ratios averaged 1.39, indicating reasonable return-to-risk from trades execution. Mean maximum drawdown of the dynamic portfolio was 10% for the studied period.

The results presented here are preliminary, and not inclusive of real-world trading costs. Analysis of these costs is obviously prerequisite to deployment as a semi-mechanical swing trading system.

Additional extensions to this work might include more exhaustive search for an optimal set of parameter values defining the trading model architecture, or variations in training hyper-parameters and loss reduction heuristics such as those listed in Table 2.

A Appendix

Summaries of trades made by the trading model for years 2012 through 2022. In each table, *#Buy* and *Wins* are counts and win percentages for corresponding ETF trades, respectively.

Symbol	#Buy	Wins
XLV	28	64.29
XLU	22	77.27
IYR	21	52.38
XLF	18	66.67
XLY	17	64.71
XLB	16	56.25
XLK	10	60.0
XLI	9	44.44
VOX	8	50.0
XLP	5	40.0
XLE	3	33.33

Table 4: Trades made for 2012.

Symbol	#Buy	Wins
XLF	36	61.11
XLK	34	64.71
XLB	18	61.11
XLU	18	72.22
XLE	14	64.29
XLI	12	58.33
XLY	12	83.33
XLP	8	100.0
IYR	7	28.57
XLV	7	28.57
VOX	3	66.67

Table 5: Trades made for 2013.

Symbol	#Buy	Wins
XLV	17	76.47
XLU	15	73.33
XLK	13	61.54
XLB	12	66.67
XLI	11	54.55
VOX	5	40.0
XLF	3	33.33
XLP	3	33.33
XLY	3	33.33
IYR	1	100.0
XLE	1	0.0

Table 6: Trades made for 2014.

where DL

Symbol	#Buy	Wins
XLV	24	54.17
XLI	16	56.25
XLY	16	43.75
XLB	15	33.33
XLF	14	42.86
XLK	12	66.67
XLP	12	58.33
VOX	10	60.0
XLE	9	44.44
XLU	9	44.44
IYR	5	40.0

Table 7: Trades made for 2015.

Symbol	#Buy	Wins
XLU	26	57.69
XLF	18	77.78
XLV	10	40.0
XLY	10	40.0
XLE	9	88.89
IYR	8	62.5
VOX	7	71.43
XLP	5	80.0
XLB	4	100.0
XLI	3	33.33
XLK	3	33.33

Table 8: Trades made for 2016.

Symbol	#Buy	Wins
XLV	21	61.9
VOX	16	37.5
XLP	16	87.5
XLB	12	66.67
XLI	12	75.0
XLK	12	66.67
IYR	11	63.64
XLF	11	54.55
XLU	10	80.0
XLY	6	50.0
XLE	3	33.33

Table 9: Trades made for 2017.

Symbol	#Buy	Wins
XLV	11	54.55
XLF	10	50.0
XLU	7	71.43
XLE	6	50.0
XLP	6	33.33
VOX	5	40.0
XLI	5	60.0
XLK	5	100.0
IYR	4	50.0
XLB	4	25.0
XLY	3	66.67

Table 10: Trades made for 2018.

Symbol	#Buy	Wins
VOX	23	73.91
XLP	22	77.27
XLU	21	66.67
IYR	19	78.95
XLY	17	70.59
XLB	12	33.33
XLK	9	100.0
XLV	9	44.44
XLI	8	62.5
XLE	6	83.33
XLF	4	50.0

Table 11: Trades made for 2019.

Symbol	#Buy	Wins
XLF	14	42.86
XLY	13	84.62
XLI	10	80.0
XLV	10	40.0
IYR	9	77.78
XLK	7	71.43
XLP	5	40.0
VOX	3	66.67
XLB	3	66.67
XLE	3	33.33
XLU	3	33.33

Table 12: Trades made for 2020.

Symbol	#Buy	Wins
XLE	21	61.9
XLB	20	55.0
XLK	18	66.67
XLI	14	71.43
XLU	11	27.27
XLY	9	66.67
IYR	8	100.0
XLF	8	37.5
VOX	6	50.0
XLV	4	75.0
XLP	3	100.0

Table 13: Trades made for 2021.

Symbol	#Buy	Wins
XLU	15	66.67
XLE	12	58.33
XLV	9	66.67
XLI	7	57.14
IYR	6	83.33
XLF	6	50.0
XLK	5	20.0
XLP	4	75.0
VOX	3	66.67
XLY	2	50.0
XLB	1	0.00

Table 14: Trades made for 2022.

References

[1] M. Abadi et al. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[2] K. Akepanidtaworn, R. Di Mascio, A. Imas, and L. Schmidt. Selling Fast and Buying Slow: Heuristics and Trading Performance of Institutional Investors. *SSRN Electronic Journal*, pages 1--92, Sept. 2019. `https://ssrn.com/abstract=3301277`.

[3] A. Beber, M. W. Brandt, and K. A. Kavajecz. What does equity sector orderflow tell us about the economy? Working Paper 16534, National Bureau of Economic Research, November 2010.

[4] Y. Bengio. Using a financial training criterion rather than a prediction criterion. *International Journal of Neural Systems*, 8 4:433--43, 1997.

[5] N. Chalkidis and R. Savani. Trading via selective classification. In *Proceedings of the Second ACM International Conference on AI in Finance*, number 14, pages 1--9. ACM, November 2021.

[6] F. Chollet et al. Keras. `https://github.com/fchollet/keras`, 2015.

[7] E. Doak. ETFs: How much do they really cost? `https://bit.ly/3fIRpm6`, September 9 2021.

[8] T. Fawcett. An introduction to ROC analysis. *Pattern Recognition Letters*, 27(8):861--874, 2006.

[9] Y. Gal and Z. Ghahramani. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In M. F. Balcan and K. Q. Weinberger, editors, *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050--1059, New York, New York, USA, 20--22 Jun 2016. PMLR.

[10] T. Greetham and M. Hartnett. The investment clock. Technical report, Merrill Lynch Global Securities Research & Economics Group, 10 November 2004.

[11] K. Grobys and J. Kolari. On industry momentum strategies. *Journal of Financial Research*, 43(1):95--119, 2020.

[12] B. Jacobsen, J. S. Stangl, and N. Visaltanachoti. Sector Rotation Across the Business Cycle. *SSRN Electronic Journal*, pages 1--53, Dec. 2009. `https://ssrn.com/abstract=1467457`.

[13] T. I. Jensen, B. T. Kelly, S. Malamud, and L. H. Pedersen. Machine Learning and the Implementable Efficient Frontier. *SSRN Electronic Journal*, pages 1--70, Aug. 2022. https://ssrn.com/abstract=4187217.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60:84 -- 90, 2012.

[15] J. K.-S. Liew and B. Mayster. Forecasting ETFs with Machine Learning Algorithms. *SSRN Electronic Journal*, Jan. 2017. https://ssrn.com/abstract=2899520.

[16] G. Metcalfe. The mathematics of market timing. *PLOS ONE*, 13(7):1--12, 07 2018.

[17] T. J. Moskowitz and M. Grinblatt. Do industries explain momentum? *The Journal of Finance*, 54(4):1249--1290, August 1999.

[18] A. Navon and Y. Keller. Financial time series prediction using deep learning. Available at arXiv: https://arxiv.org/abs/1711.04174, November 2017.

[19] A. M. Ozbayoglu, M. U. Gudelek, and O. B. Sezer. Deep learning for financial applications : A survey. *Applied Soft Computing*, 93, 2020.

[20] M. Pinelis and D. Ruppert. Machine Learning Portfolio Allocation. *SSRN Electronic Journal*, pages 1--42, Mar. 2020. https://ssrn.com/abstract=3546294.

[21] M. Sauer. Sector rotation through the business cycle: A machine learning regime approach. pages 1--36, September 30 2019. https://ssrn.com/abstract=3473907.

[22] W. F. Sharpe. Mutual fund performance. *Journal of Business*, 39(1):119--138, 1966.

[23] J. Sirignano and R. Cont. Universal Features of Price Formation in Financial Markets: Perspectives From Deep Learning. *SSRN Electronic Journal*, pages 1--20, March 2018. https://ssrn.com/abstract=3141294.

[24] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: A simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(56):1929--1958, 2014.

[25] A. Tversky and D. Kahneman. Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2):263--292, 09 1979.

[26] T. Wiecki, A. Campbell, J. Lent, and J. Stauth. All that glitters is not gold: Comparing backtest and out-of-sample performance on a large cohort of trading algorithms. *The Journal of Investing*, 25(3):69--80, Fall 2016.