Using Supervised Machine Learning Techniques to trade intraday Index Futures

Michael Huber[1]

[1] Harrisburg University of Science and Technology

Author Note

The authors made the following contributions. Michael Huber: Conceptualization, Data Aggregation, Coding - Original Draft Perperation, Coding - Review & Editing, Writing - Original Draft Preparation, Writing - Review & Editing.

Correspondence concerning this article should be addressed to Michael Huber, Harrisburg University Address. E-mail: mhuber@my.harrisburgu.edu

Using Supervised Machine Learning Techniques to trade intraday Index Futures

Research and implementation of algorithmic trading in financial markets is consistently evolving. The complexity of the financial markets creates a novel, solvable puzzle that changes every day at market open. How to apply and develop systems to either support human-decision making or create an autonomous trading system, is one of the most important challenges in the financial market domain (Fama (1970), Golub et al. (2018)). Researchers have tried to apply mathematical and statistical concepts to develop a successful trading edge for centuries, such as Fibonacci in 1202 and Bernoulli in the late 16th Century. Both mathematicians have homonymous technical indicators in contemporary Quant-trading and Hedge-Fund analysis ((Akyıldırım and Soner (2014), Mandelbrot and Taylor (1967)).

Motivation

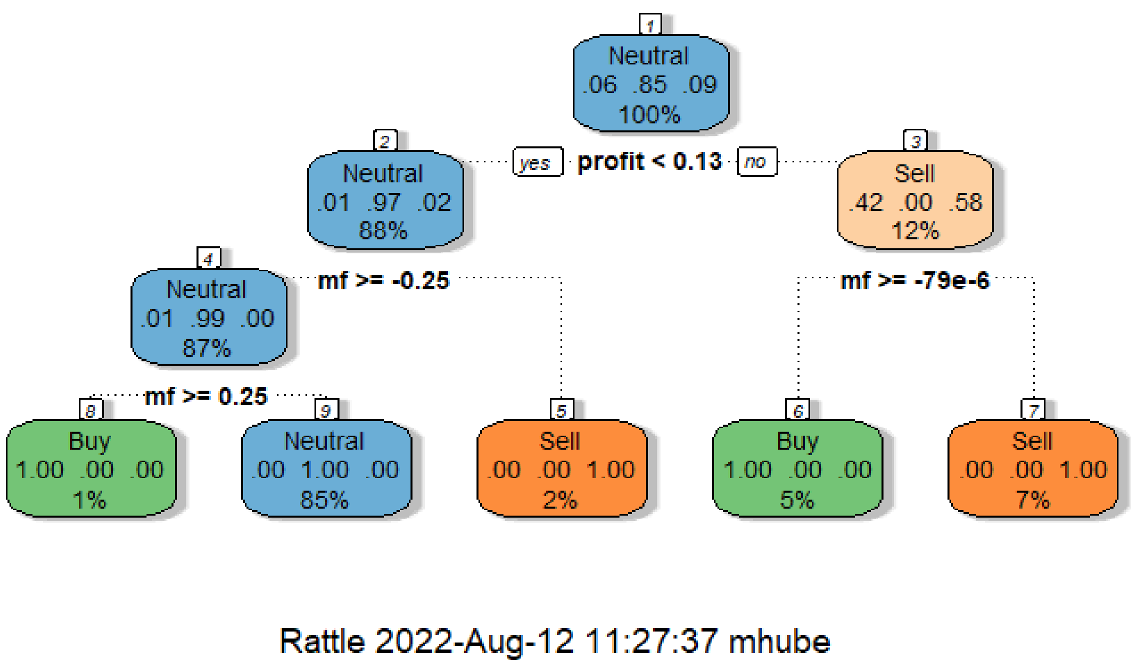
Especially in Equity trading, the shortcomings of human decision-making become obvious. Technical support, such as combining multiple market indicators to create a well-rounded stock forecasting model is necessary for a profitable shot as an Equity Trader (Li et al. (2020)). Technical Analysis is the main forecasting method for traders analyzing previous market data of equity. Those indicators usually revolve around price and volume (Kirkpatrick and Dahlquist (2006)). Technical Analysis has become a widely accepted method and useful analytical method to predict future price movement. Trading volume has been an effective predictor of stock price returns, and according to Chen et al. (2001) is directly related to return volatility. In this data-set, I will focus on Chalkin Money-Flow or MF and apply supervised Machine Learning techniques to trade future price-action of Index Futures intraday. The goal is to clarify if intraday Futures trading can be profitable with supervised Machine Learning algorithms applying the Chalking Money Flow technical indicator in addition to Volume, Open, High, Low, and Close price.

Methodology & Data

Thomsett (2010) showed how Money Flow can be a successful technical trading indicator. MF focuses on change in Price and Volume and produces a numerical value from -1 to +1. Based on these values a position can be taken. Generally, a higher number indicates a Buy position and a lower/negative number indicates a Short/Sell position. Kannan et al. (2010) also applied MF successfully in data mining techniques, proving the importance of this technical indicator. For the data used in this research paper, I have applied 1-minute Open, High, Low, Close, and Volume data for the Futures Index ES, which represents the S&P 500. The 1-minute data is particularly important, as intraday trading is often executed on low minute time-frames. Trades usually do not exceed a 5-minutes. The data provided ranges from March 1st, 2022 to March 14th, 2022. This data then underwent standard data-cleaning, data-processing, and data-exploration techniques. When analyzing correlation we can see that especially price of the underlying security can be highly correlated. I then added MF to the Data set, using n=20. We then further excluded the first 19 observations. This reduced the data to 13359 observations of 6 variables. Furthermore, I coded the trading features, which would take positions based on the numerical value provided by the MF technical indicator. If the data-set then applied a Buy or Sell rating, I initialized a position and calculated the potential profit in a separate column. I then separated the data into training (80%) and testing (20%) data sets. For the testing data set, the MF indicator would have initialized 314 trades for a total profit of $717.75. The ES, which stands for E-mini S&P 500 is a futures contract that provides delivery of the underlying security, S&P 500, at maturity. Keep in mind this is not perfectly accurate, as execution costs (the price per Futures contract is usually $5) have to be incorporated. Additionally, this also implies immediate execution and no ask-bid disparity, which are hurdles that can also impact profitability. Nevertheless, given that Futures contracts are a leveraged product, the $717.75 profit implied is only the nominal difference of the underlying security. $0.25 difference of the ES-mini materializes as a profit of $12.50 per Futures contract traded. Meaning my applied strategy would have netted the investor $35,887.5 minus execution costs.

Decision Tree

The Decision learning technique focused on two different metrics. For the first Decision Tree, I trained to model to take appropriate action - Buy or Sell - given the information provided from the training set. We used rpart2 for the predictive analysis. Given randomized data and actual data, we achieved a 100% accuracy on either model. Meaning the DT model took the same trade as the training set implied 100% of the time. Excluding High-Low-and Profit Variables did indeed not alter the results.



Random Forest

For the Random Forest technique, a altered the variables and tried to instead predict the correct action (Buy, Hold or Sell), to predict profitability. The predicted profitability then had a total accuracy of 29.91% but it reached an overall sum of profits of $974.75 on 353 Buy or Sell signals. The max return per trade was $44.50 and the max loss was $3.50.

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
-3.500	0.500	1.250	2.716	2.750	44.500

Testing the same model with the Action variable, we achieved a 99.96% accuracy with 386 Buy or Sell signals.

prof_pred	Buy	Neutral	Sell
Buy	165	0	0
Neutral	1	2287	0
Sell	0	0	221

Support Vector Machines

Support Vector Machines usually classify data for learning algorithms. Given the Data provided, we did not expect a great outcome in the algorithm learning curve. In particular predicting Profit, which is inherently flawed given that is the difference between the Open and Close price, the profit predictive methodology does not apply well in this circumstance. Furthermore, even when predicting the profitability of the testing set, we did only achieve $276.75 given 331 actionable buy and sell ratings. Classifying the data for Decision tree might be helpful, but for trading/financial data that has high correlation and is already grouped, I see no evidence of being useful in intraday trading algorithms [1] 276.75 w 331 Actions

Naive-Bayes

Naive Bayes, which is often used for text classification was applied last to the data set. We tried to classify the action variable. This led to a high accuracy while grouping the testing set and applying the NB model. The model achieved an overall accuracy of 97.34%. Most importantly the algorithm did not suggest a buy rating for a sell rating and vice versa. The applied probability for the Chalkin Money Flow on a Gaussian metric was relatively low, as we can see clear dominance of the Neutral or hold action. The mean of either action was fairly similar for both Buy and Sell ratings which clarifies the quality of the data set. The standard deviation was largest for Neutral, as this has the highest MF allocated.

```
          Buy Neutral Sell
  Buy     129       1    0
  Neutral  37    2281   28
  Sell      0       5  193
[1] 97.3448
```

```
 A priori probabilities:

      Buy    Neutral       Sell
0.05970429 0.85775781 0.08253790

---------------------------------------------

 Tables:

---------------------------------------------
 ::: mf (Gaussian)
---------------------------------------------

mf             Buy      Neutral         Sell
  mean 0.339291954 -0.003478046 -0.332713066
  sd   0.083399484  0.124591687  0.071229887
```

Conclusion

Intraday ES-mini Futures trading according to the Chalkin Money Flow had certain predictive capabilities and seems to function well in the provided data set. Supervised grouping and classifier machine learning algorithms do not serve the greatest purpose in classifying the data. This would be helpful when comparing multiple technical indicators and observing the strength of either profitability or the Sharpe ratio. The Decision Tree and Random Forest Algorithm did well in terms of classification and predictability. Analyzing the test set either algorithm had a 100% accuracy in assigning a buy or hold rating given the price, volume, and MF provided. Trading analogous of Money-Flow indicator has to be tested with a wider data set to analyze the future profitability. Including execution costs, such as round trips, are also advisable. The trained Random-Forest or Decision-Tree model can be applied to new data to make more poignant predictions. In this case, our trading model netted $35,887.5 minus execution costs in 10 trading days, trading one Futures contract at a time.

We used R (Version 4.2.1; R Core Team, 2022) and the R-packages }*ggplot2*

[@}R-ggplot2], *papaja* (Version 0.1.0.9999; Aust & Barth, 2020), *plotly* (Version 4.10.0; Sievert, 2020), and *tinylabels* (Version 0.2.3; Barth, 2022) for all our analyses.

References

Akyıldırım, E., & Soner, H. M. (2014). A brief history of mathematics in finance. *Borsa Istanbul Review*, *14*(1), 57–63.

Aust, F., & Barth, M. (2020). *papaja: Prepare reproducible APA journal articles with R Markdown*. https://github.com/crsh/papaja

Barth, M. (2022). *tinylabels: Lightweight variable labels*. https://cran.r-project.org/package=tinylabels

Chen, J., Hong, H., & Stein, J. C. (2001). Forecasting crashes: Trading volume, past returns, and conditional skewness in stock prices. *Journal of Financial Economics*, *61*(3), 345–381.

Fama, E. F. (1970). Efficient capital markets: A review of theory and empirical work. *The Journal of Finance*, *25*(2), 383–417.

Golub, A., Glattfelder, J. B., & Olsen, R. B. (2018). The alpha engine: Designing an automated trading algorithm. In *High-performance computing in finance* (pp. 49–76). Chapman; Hall/CRC.

Kannan, K. S., Sekar, P. S., Sathik, M. M., & Arumugam, P. (2010). Financial stock market forecast using data mining techniques. *Proceedings of the International Multiconference of Engineers and Computer Scientists*, *1*, 4.

Kirkpatrick, C. D., & Dahlquist, J. (2006). *Technical analysis: The complete resource or financial market technicians. New jersey: FT press.*

Li, Y., Ni, P., & Chang, V. (2020). Application of deep reinforcement learning in stock trading strategies and stock forecasting. *Computing*, *102*(6), 1305–1322.

Mandelbrot, B., & Taylor, H. M. (1967). On the distribution of stock price differences. *Operations Research*, *15*(6), 1057–1062.

R Core Team. (2022). *R: A language and environment for statistical computing*. R Foundation for Statistical Computing. https://www.R-project.org/

Sievert, C. (2020). *Interactive web-based data visualization with r, plotly, and shiny.*

Chapman; Hall/CRC. https://plotly-r.com

Thomsett, M. C. (2010). *CMF–chaikin money flow: Changes anticipating price reversal*. FT Press.