Faculty Advisor: Dr. Ying Xie

GC-382

Artificial Intelligence for Quantitative Trading

Winifred Akpan, Emmanuel Ayo, Malek Browning, Jason Kennedy

Abstract

This project involves the use of Artificial Intelligence (AI) and Machine Learning (ML) techniques in quantitative trading and stock market analysis for educational purposes. The goal of this project is to predict stock market movement to help investors mitigate risks associated with trading and to provide higher returns.

This involves the implementation and refinement of python frameworks: sklearn library (pandas and numpy), in addition to the application of a labeling strategy to predict future daily stock trends - with the Supervised Learning AI model (Random Forest). The techniques developed in this project will be used to analyze market trends, predict market movements, and optimize investment strategies. This project will provide knowledge and practical skills in AI, ML, and quantitative trading.

Introduction

AI has become an essential tool in quantitative trading, as it is used to develop and implement profitable trading strategies. For this project, we explore this application using the stock data of NVIDIA Corporation (NVDA), a leading manufacturer of graphics processing units (GPUs). This research will contribute to the growing body of research on AI applications in quant trading and will also provide practical educational insight for traders and investors interested in utilizing AI to develop and implement trading strategies.

Materials

Data Processing: The raw stock market data for Nvidia (Ticker: NVDA) was utilized ('.csv' file).
Analysis: use of python framework, with web-based application (jupyter notebook) and the sklearn library, including pandas and numpy to preprocess and visualize the data and to apply appropriate indicators. Also utilized Random Forest Prediction Model and Random Forest Feature Selection technology to train and test the data, and to make the final prediction.
Evaluation Metrics: Accuracy and Recall score, confusion matrix, and feature selection technology.

Methods

Labeling Strategy: based on labeling a day as '1' indicating a good day to take a buying position if the next 5 days price (Close_Pred) was greater than today's (Close). And if the next 5 day's price is lower than today's price, the label for that day should be '0' which indicates taking a selling position.
Supervised Learning Model: The Supervised Learning model, Random Forest, was used to analyze market trends, and predict potential market movements. The model uses training and testing of data based on the labels to make market predictions for buying or selling shares or stocks.
Optimization: Utilize the indicators and Volume as the training features, and test this on the prediction of the 'Close' price going up or down to indicate buying/selling.

Results

- Accuracy Score of 61.70%, Recall rate of 69.43%. Model accurately predicted 114 days as good days to sell (0), and 184 days as good days to buy (1).
- The preliminary findings suggest that the combination of these machine learning techniques can be effective in predicting the future of NVDA stock.
- Through enrichment of the raw stock data with relevant stock indicators and market indicators such as trends, momentum, volatility, and volume, patterns were identified in the data that could inform trading strategies.
- Analysis showed that the random forest model was able to predict future trends/market movements with reasonable accuracy.
- While our preliminary findings suggest that the combination of our previously mentioned methods can be effective in predicting the future of NVDA stock, further analysis and testing is necessary to confirm the effectiveness of these techniques.

Labeling the Data

```
In [11]:  #defining predicted column 'Close_Pred' which is derived from the
          nvda["Close_Pred"] = nvda["Close"].shift(-5)

In [12]:  #Defining the labeling critieria, which is, if the new future valu
          #However, if 'Close_Pred' future values are lesser than 'Close' hi
          nvda['Label'] = (nvda['Close_Pred'] > nvda['Close']).astype(int)
```

Fig.1: **The labeling strategy** was based on labeling a day as '1' indicating a good day to take a buying position if the next 5 days price (Close_Pred) was greater than today's (Close); and labeling as '0' to indicate taking a selling position if the next 5 day's close price was lower than today's Close price.

Feature Selection

```
In [21]:  from sklearn.feature_selection import SelectFromModel
          clf = SelectFromModel(RandomForestClassifier(n_estimators=100))
          clf.fit(X_train, y_train)

Out[21]:  SelectFromModel(estimator=RandomForestClassifier())

In [22]:  clf.get_support()

Out[22]:  array([False, False, False, False, False, False, False,  True, False,
                 False,  True,  True,  True,  True,  True,  True,  True,  True])

In [23]:  selected_feat= X_train.columns[(clf.get_support())]
          len(selected_feat)

Out[23]:  9

In [24]:  print(selected_feat)

          Index(['SMA200', 'EMA26', 'MACD', 'RSI_21', 'BB_M', 'BB_H', 'BB_L', 'OBV',
                 'ATR'],
                dtype='object')
```

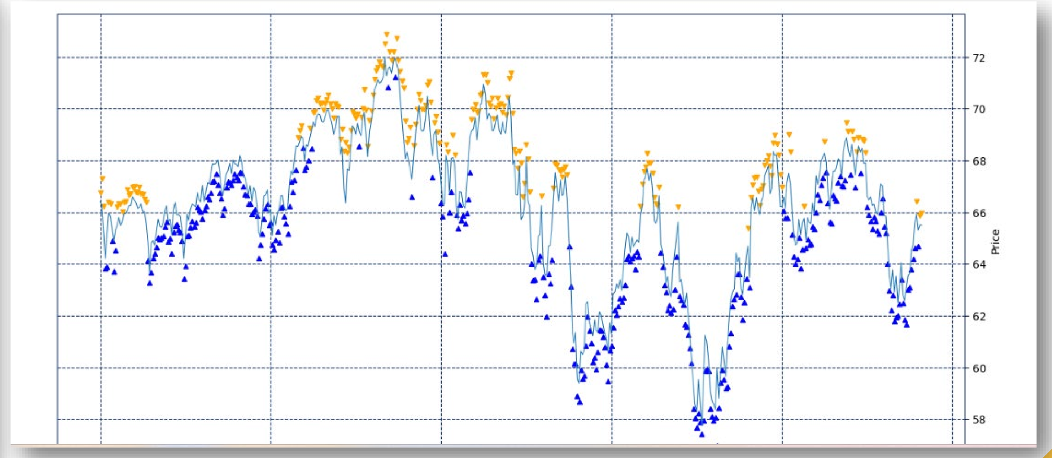
Fig. 3

- SMA200: 200-day simple moving average
- EMA26: 26-day exponential moving average
- MACD: Moving Average Convergence Divergence
- RSI_21: Relative Strength Index
- BB_M: Bollinger Bands Middle Indicator
- BB_H: Bollinger Bands Higher Indicator
- BB_L: Bollinger Bands Lower Indicator
- OBV: On Balance Volume
- ATR: Average True Range

Confusion Matrix

```
y_pred = clf.predict(X_test)

#Evaluate model's performance
accuracy = accuracy_score(y_test, y_pred)
print("Accuracy: {:.2f}%".format(accuracy * 100))

Accuracy: 61.70%

In [16]:  #To display confusion matrix values along with predicted and actual values
          cm = confusion_matrix(y_pred, y_test)
          print(cm)
          result = pd.DataFrame({'Pred': y_pred, 'True': y_test})
          pd.set_option('display.max_rows', None)
          print(result)

[[114  81]
 [104 184]]
                           Pred  True
Date
2021-05-07 00:00:00-04:00   0.0   0.0
2021-05-10 00:00:00-04:00   0.0   0.0
2021-05-11 00:00:00-04:00   0.0   0.0
2021-05-12 00:00:00-04:00   1.0   1.0
2021-05-13 00:00:00-04:00   1.0   1.0
2021-05-14 00:00:00-04:00   0.0   0.0
2021-05-17 00:00:00-04:00   0.0   0.0
```

Fig. 2:
True Negative (TN): 114– this shows the number of records that model predicted as 0 and that are real 0s. That is, the model predicted that we take a selling position for these days, and it was right.
False Positive (FP): 104 - this shows the number of records that model predicted as 1 and that are real 0s. That is, the model predicted that we take a buying position for these days, but instead, we should really take a selling position.
False Negative (FN): 81 - this shows the number of records that model predicted as 0 and that are real 1s. That is, the model predicted that we take a selling position for these days, but instead, we should really take a buying position.
True Positive (TP): 184 - this shows the number of records that model predicted as 1 and that are real 1s. That is, the model predicted that we take a buying position for these days, and it was right.



Conclusions

The conclusion of the testing is that our model is good for prediction with its high recall rate (69.43%), which signifies the number of times it predicts a day as good for buying and was right. Even though the accuracy score of 61.70% was not very high, the confusion matrix shows that the model did excellently well in predicting which days to sell and buy. It accurately predicted 114 days as selling (0) days and 184 days as buying (1) days.

Our research and testing also shows that the features that were very important and contributed to our model's performance in predicting positions to take are: SMA200, EMA26, MACD, RSI_21, BB_M, BB_H, BB_L, OBV and ATR.

Our findings suggest that AI-based techniques can be a valuable tool for traders and investors in identifying profitable opportunities in financial markets. Additional analysis, training and testing is strongly recommended to confirm the effectiveness of these techniques and to explore their potential limitations and shortcomings.

Acknowledgments

- Dr. Ying Xie – Faculty Advisor
- Kennesaw State University College of Computing and Software Engineering (CCSE)
- Computing Showcase Day (C-Day) sponsors, judges and audience.
- Online Research Resources
- IT7993 – AI Quant Trading Team 2

Contact Information

- **Emmanuel Ayo**: eayo@students.kennesaw.edu
- **Jason Kennedy**: jkenne93@students.kennesaw.edu
- **Malek Browning**: mbrow542@students.kennesaw.edu
- **Winifred Akpan**: wakpan@students.kennesaw.edu

Project Website:
https://sites.google.com/view/team-2-ai-quant-capstone/home

References

Cohen, G. (2022). Algorithmic Trading and Financial Forecasting Using Advanced Artificial Intelligence Methodologies. *Mathematics*, *10*(18), 3302. MDPI AG. Retrieved February 18, 2023, from http://dx.doi.org/10.3390/math10183302
Li, K. (Y. (2023, January 25). *Predicting stock prices using machine learning*. neptune.ai. Retrieved February 18, 2023, from https://neptune.ai/blog/predicting-stock-prices-using-machine-learning
Lim, J. (2021, July 21). *How AI algorithms are changing trading forever*. Entrepreneur. Retrieved February 18, 2023, from https://www.entrepreneur.com/en-in/technology/how-ai-algorithms-are-changing-trading-forever/377985
Wadhwa, M. (2023, January 23). *AI in stock market: Predicting the ups and downs*. DataToBiz. Retrieved February 18, 2023, from https://www.datatobiz.com/blog/ai-in-stock-market/

KENNESAW STATE UNIVERSITY
COLLEGE OF COMPUTING AND SOFTWARE ENGINEERING