Deep stock representation learning: from candlestick charts to investment decisions

Hu, G., Hu, Y., Yang, K., Yu, Z., Sung, F., Zhang, Z., Xie, F., Liu, J., Robertson, N., Hospedales, T., & Miemie, Q. (Accepted/In press). *Deep stock representation learning: from candlestick charts to investment decisions*. Paper presented at IEEE International Conference on Acoustics, Speech and Signal Processing, Calgary, Canada.

Document Version:
Peer reviewed version

Queen's University Belfast - Research Portal:
Link to publication record in Queen's University Belfast Research Portal

Publisher rights
© 2018 IEEE.
This work is made available online in accordance with the publisher's policies. Please refer to any applicable terms of use of the publisher.

General rights
Copyright for the publications made accessible via the Queen's University Belfast Research Portal is retained by the author(s) and / or other copyright owners and it is a condition of accessing these publications that users recognise and abide by the legal requirements associated with these rights.

Take down policy
The Research Portal is Queen's institutional repository that provides access to Queen's research output. Every effort has been made to ensure that content in the Research Portal does not infringe any person's rights, or applicable UK laws. If you discover content in the Research Portal that you believe breaches copyright or violates any law, please contact openaccess@qub.ac.uk.

Download date:26. Aug. 2022

DEEP STOCK REPRESENTATION LEARNING: FROM CANDLESTICK CHARTS TO INVESTMENT DECISIONS

Guosheng Hu[5,*] *Yuxin Hu*[1,*] *Kai Yang*[2] *Zehao Yu*[3] *Flood Sung*[4] *Zhihong Zhang*[3] *Fei Xie*[1,◇]
Jianguo Liu[1] *Neil Robertson*[5] *Timothy Hospedales*[6,♡] *Qiangwei Miemie*[6,7,8,♡]

[1]Shanghai University of Finance and Economics [2]University of Shanghai for Science and Technology [3]Xiamen University [4]Independent Researcher
[5]Queen's University Belfast [6]The University of Edinburgh [7]Yang's Accounting Consultancy Ltd [8]ArrayStream Technologies Ltd

ABSTRACT

We propose a novel investment decision strategy (IDS) based on deep learning. The performance of many IDSs is affected by stock similarity. Most existing stock similarity measurements have the problems: (a) The linear nature of many measurements cannot capture nonlinear stock dynamics; (b) The estimation of many similarity metrics (e.g. covariance) needs very long period historic data (e.g. 3K days) which cannot represent current market effectively; (c) They cannot capture translation-invariance. To solve these problems, we apply Convolutional AutoEncoder to learn a stock representation, based on which we propose a novel portfolio construction strategy by: (i) using the deeply learned representation and modularity optimisation to cluster stocks and identify diverse sectors, (ii) picking stocks within each cluster according to their Sharpe ratio (Sharpe 1994). Overall this strategy provides low-risk high-return portfolios. We use the Financial Times Stock Exchange 100 Index (FTSE 100) data for evaluation. Results show our portfolio outperforms FTSE 100 index and many well known funds in terms of total return in 2000 trading days.

1 Introduction

Investment decision making is a classic research area in quantitative and behavioural finance. One of the most important decision problems is portfolio construction and optimisation [1, 2], which addresses selection and weighting of assets to be held in a portfolio. Financial institutions try to construct and optimise portfolios in order to maximise investor returns while minimising investor risk.

Stock similarity is important for many investment decision strategies [3]. For example, the classical investment strategy, mean-variance theory [1], measures stock similarity using variance. Most similarity measurements have the following problems: (a) Usually, the time series (linear signal) is fed to linear metric (e.g. covariance, Pearson) to obtain similarity. The linear nature of most similarities cannot capture the nonlinear dynamics of the stocks. (b) In [4], it is claimed that n (the number of stocks in one market, e.g. 2,033 tradable stocks in London Stock Exchange) historic days/weeks data is needed to estimate an accurate covariance. However, the past n days/weeks data cannot represent the current market effectively. (c) Most similarity measurements do not consider translation (time)-invariance, which is important for stock similarity. For example, the price of Apple stock increases at one particular day, however, the stock prices of Apple suppliers might increase after 3 days.

To solve the aforementioned problems, we propose to use deep learning (DL) features for stock similarity measurement instead of raw time series. Convolutional DL approaches such as Convolutional AutoEncoder (CAE, unsupervised) [5] and Convolutional Neural Network (CNN, supervised) [6], have achieved very impressive performance for analysing visual imagery. This has motivated researchers to convert raw input signals from other modalities into images to be processed by CNNs or CAEs. In this way, good results have been achieved for diverse applications. For example, traditional speech recognition methods used the 1-D signal vector, e.g. the raw input waveform [7, 8]. In contrast an alternative approach is to convert the 1-D signal to a spectrogram, i.e. an image, in order to leverage the strength of CNNs to achieve promising recognition performance [9]. As another well known example, AlphaGo [10] represents the board position as a 19×19 image, which is fed into a CNN for feature learning. Besides, computer vision techniques have also been applied to judge the quality of paper [11] and calculate the rank of matrix [12] from its appearance only. With similar motivation, we explore to convert a 4-channel stock time-series (lowest, highest, opening and closing price for the day) to candlestick charts by synthesis technique to present price history as images. To avoid expensive annotation, we choose the unsupervised CAE for stock feature learning using the synthetic candlestick images.

Hence, the first novelty of this study is exploiting deep learning (i.e. CAE) to encode stock time series. Compared with raw time series, deeply learned features can effectively capture (i) nonlinear stock dynamics and semantics; (ii) the translation-invariance. The similarity measurements based on deep features can overcome the aforementioned weaknesses of most existing measurements. In addition, we contribute a new valuable signal, deep feature, to the investment decision society, in which new effective signal is important for risk hedging. Though some deep learning models, such as LSTM

* These authors contributed equally to this work
◇ Corresponding author: xiefei@mail.shufe.edu.cn
♡ Email: t.hospedales@ed.ac.uk, yongxin.yang@arraystream.com

[13] and RNN [14], have been applied to optimise portfolio, they use raw time series rather than charts as input.

Second, motivated by momentum effect [15], we construct a novel portfolio generation pipeline including: (1) deep feature learning by visual interpretation price history, (2) clustering the stocks based on the similarity computed on deep features to provide a data-driven segmentation of the market, (3) actual portfolio construction. For *visual representation* learning, we generate millions of training images (synthetic candlestick charts) which are fed to a deep CAE for feature learning. In the next *clustering* step, we aim to segment the market into diverse sectors in a data-driven way. This is important to provide risk reduction by selecting a well diversified portfolio [16, 17]. The similarity embedded in clustering method is computed using deep features. Popular clustering methods such as K-means are not suitable here because they are non-deterministic and/or require a pre-defined numbers of clusters to find. In particular non-deterministic methods are not acceptable to real financial users. To address this we adapt the modularity optimization method [18] – originally designed for network community structure – to stock clustering. Finally, we perform *portfolio construction* by the simple yet effective approach of choosing the best stock within each cluster according to their Sharpe ratio [19]. As we will see in our evaluation, this portfolio selection strategy combines high returns with low risk.

2 Methodology

Our overall investment decision pipeline includes three main modules: deep feature learning, clustering, and portfolio construction. For *deep feature learning*, raw 4-channel time series data describing stock price history are converted to standard candlestick charts. These charts are fed into deep CAEs for visual feature learning. These learned features provide a vector embedding of a historical time-series that captures key quantitative and semantic information. Next we *cluster* the features in order to provide a data-driven segmentation of the market to underpin subsequent selection of a diverse portfolio. Many common clustering methods are not suitable here because they are non-deterministic or require predefinition of the number of clusters. Thus we adapt modularity optimisation for this purpose. Note that stock similarity embedded in our clustering method is computed using the nonlinear deep features. Finally, we perform *portfolio construction* by choosing stocks with the best performance measured by Sharpe ratio [19] from each cluster. The overall pipeline is summarised schematically in Fig. 1. Each component is discussed in more detail in the following sections.

2.1 Deep Feature Learning with CAEs

Chart Encoding To realise an algorithmic portfolio construction method based on visual interpretation of stock charts, we need to convert raw price history data to an image representation. Our raw data for each stock is a 4-channel

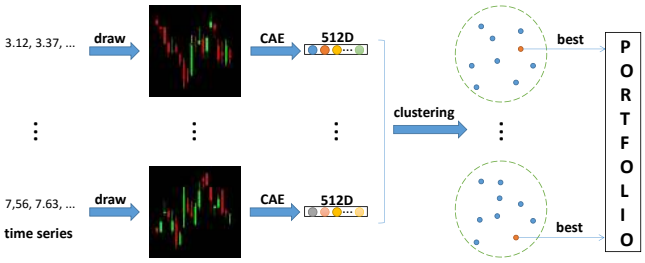

Fig. 1. Schematic illustration of our investment decision pipeline. The architecture of CAE is detailed in Fig. 2.

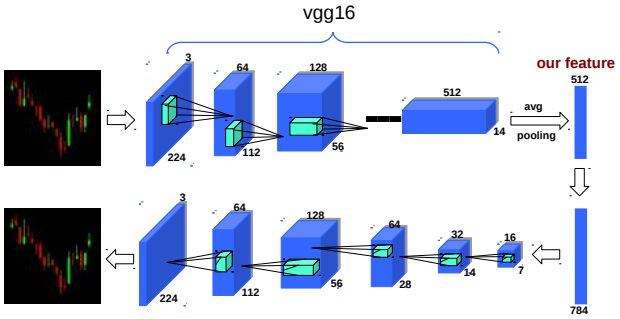

Fig. 2. CAE overview. An encoder (top) - decoder (bottom) framework. The 512D feature following average pooling provides our representation for clustering and portfolio construction.

time series (the lowest, the highest, open, and closing price for the day) in a 20-day time sequence. We use computer graphics techniques to convert these to a candlestick chart represented as a RGB image as shown in Fig. 1 and 2. The whisker plots describe the four raw channels, with colour coding describing whether the stock closed higher (green) or lower (red) than opening. An encoded candlestick chart image provides the visual representation of one stock over a 20-day window for subsequent visual interpretation by our deep learning method.

Convolutional Autoencoder Our CAE architecture is summarised in Fig. 2. It is based on the landmark VGG network [20], specifically VGG16. The VGG network is a highly successful architecture initially proposed for visual recognition. To adapt it for use as a CAE encoder, we remove the final 4096D FC layers from VGG-16 and replace them by an average pooling layer to generate one 512D feature. The decoder is a 7-layer deconvolutional network that starts with a 784D layer that is fully connected with the 512D embedding layer. Following 6 up-sampling deconvolution layers eventually reconstruct the input based on our 512D feature. When trained with a reconstruction objective, the CAE network learns to compress input images to the 512D bottleneck in a manner that preserves as much information as possible in order to be able to reconstruct the input. Thus this single 512D vector encodes the 20-day 4-channel price history of the stock, and will provide the representation for further processing (clustering and portfolio construction).

2.2 Clustering

We next aim to provide a clustering method for diversified – and hence low risk – portfolio selection. As discussed, many existing clustering methods are non-deterministic or require pre-specification of the number of clusters, which make them unsuited for our application. To solve these problems, we introduce the network modularity method [18] to find the cluster structure of the stocks, where each stock is set as one node and the link between each pair of stocks is set as the cosine similarity calculated by our learned CAE features. Modularity is introduced as the fraction of the links that fall within the given group minus the expected fraction if links are distributed at random. Modularity optimisation [18], originally used for detecting community structure in networks, can end with generating clusters. Specifically, optimisation operates on a graph (one 20-day history of the entire market in our case), and updates the graph to group stocks so as to eventually achieve maximum modularity before terminating. Thus it does not need a specified number of clusters and is not affected by initial node selection.

2.3 Portfolio Construction and Backtesting

Given the learned stock clustering (market segmentation) , we construct a complete portfolio by picking diverse yet high-return stocks, and evaluate the result.

Stock Performance Return (profit on an investment) is defined as $r_t = (V_f - V_i)/V_i$, where V_f and V_i are the final and initial values, respectively. For example, to compute daily stock return, V_f and V_i are closing prices of today and yesterday, respectively. We measure the performance of one particular stock over a period using the Sharpe ratio [19] $s = \overline{r}/\sigma_r$, where $\overline{r}$ is the mean return, σ_r is the standard deviation over that period. Thus the Sharpe ratio s encodes a trade-off of return and stability. Maximum Drawdown (MDD) is the measure of decline from peak during a specific period of investment: MDD $= (V_t - V_p)/V_p$, where V_t and V_p mean the trough and peak values, respectively.

Training and Testing For every 20 trading days, we cluster all the stocks. To actually construct a portfolio we then choose the stock with the highest Sharpe ratio [19] within each cluster. We then hold the selected portfolio for 10 days. Over these following 10 days, we evaluate the portfolio by computing our 'compound return' for each selected stock. The overall return of one portfolio is the average compound return of all the selected stocks. We use a stride of 10. The process of portfolio selection and return computation are analogous to training and testing process in machine learning, respectively.

Fund Allocation Since our clustering method discovers the number of stocks in a data driven way, in different trading periods we may have different number of clusters. Assume that we obtain K_1 clusters in one period and will select K_2 stocks to construct one portfolio. Then, letting Q and R indicate quotient and remainder respectively in $[Q, R] = K_2/K_1$: K_2 stocks are picked by taking (i) Q stocks from each of the K_1 clusters and (ii) the remaining R best performing stocks across all K_1 clusters. Then, we allocate *equally* $1/K_2$ of the fund to each of the chosen stocks.

3 Experiments

We first introduce our dataset and experiment settings. We analyse the outputs of feature learning. Finally, we compare our whole investment strategy (feature extraction, clustering, portfolio optimisation) with alternatives.

3.1 Dataset and Settings

For evaluation, we use the stock data of Financial Times Stock Exchange 100 Index (FTSE 100), which is a share index of the 100 companies listed on the London Stock Exchange with the highest market capitalisation. We use all the stocks in FTSE 100 from 4th Jan 2000 to 14th May 2017. The stock price is adjusted accounting for stock splits, dividends and distributions. Every 20-day 4-channel time series generates a standard candlestick chart. We generate 400K FTSE100 charts in all. The training images for our CAE are candlestick charts rendered as 224×224 images to suit our VGG16 architecture [20]. During training, the batch size is 64, learning rate is set to 0.001, and the learning rate decreases with a factor of 0.1 once the network converges.

3.2 Qualitative Results

Visualising and Understanding Deep Features The features of one year (2012) for a given stock are concatenated to form a new feature. These features of all the stocks are visualised in Fig. 3 using the t-Distributed Stochastic Neighbour Embedding (t-SNE) [21] method. One colour indicates one industrial sector defined by Bloomberg From Fig. 3, we can see the stocks with similar semantics (industrial sector) are represented close to each other in the learned feature space. For example, Materials related stocks are clustered. This illustrates the efficacy of our CAE and learned feature for capturing semantic information about stocks.

3.3 Quantitative Results

For quantitative evaluations, we apply 7 measures for evaluation: Total return, daily Sharpe ratio, max drawdown, daily/monthly/yearly mean return, and win year. Win year indicates the percent of the winning years. The other measures are defined in the section of 'Portfolio Construction and Backtesting'. We choose $K_2 = 5$ stocks to construct all the portfolios compared.

Comparison with FTSE 100 Index We perform backtesting to compare our full portfolio optimisation strategy against the market benchmark (FTSE 100 Index). In Fig. 4, we compare with FTSE 100 index. Fig. 4 (a) shows the comparison over a long-term trading period (4K trading days, 31/01/2000 - 06/10/2016) showing the overall effectiveness of our strategy. Note that it is very difficult for funds to consistently outperform the market index over an extended period of time due

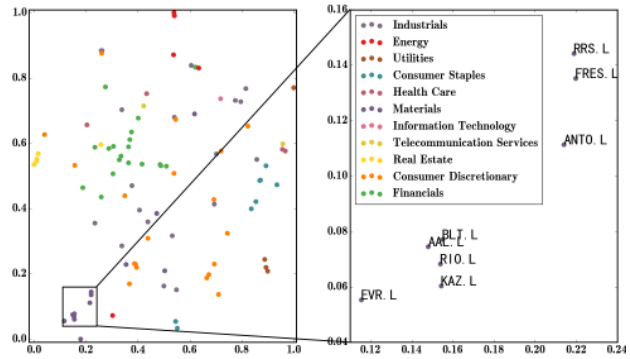

Fig. 3. t-SNE visualisation of FTSE 100 CAE features. One colour indicates one industrial sector. The stocks on the right are all from Sector Materials: RRS.L (Randgold Resources Limited), FRES.L (Fresnillo PLC), ANTO.L (Antofagasta PLC), BLT.L (BHP Billiton Ltd), AAL.L (Anglo American PLC), RIO.L (Rio Tinto Group), KAZ.L (KAZ Minerals), EVR.L (EVRAZ PLC)

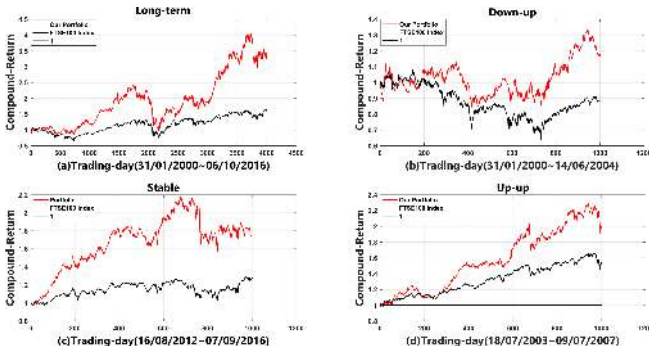

Fig. 4. Our portfolio vs FTSE 100 Index

to the complexity and diversity of market variations. In Fig. 4 (b)-(c) we show specific shorter term periods where the market is behaving very differently including down-up (b), flat (c) and bullish (d). The overall dynamic trends of our strategy reflect the conditions of the market (meaning that the stocks selected by our strategy are representative of the market), yet we outperform the market even across a diverse range of conditions (b-c), and over a long time-period (a).

Feature and Clustering We evaluate features and clustering methods over a long term period (4K trading days). From Table 1, the total return of our method (D-M, deep feature + modularity-based clustering) is higher than R-M (R-M, Raw time series + modularity-based clustering), 283.5% vs 208.8%. It means the deeply learned feature capture richer information, which is more effective for portfolio optimisation than raw time series. Similar conclusions can be drawn based on other measures. In terms of clustering method, our modularity optimization method works better than D-K (deep feature + k-means) in terms of returns and daily Sharpe, showing the effectiveness of modularity-based clustering. As explained in the Introduction, k-means cannot be used for portfolio construction in practice. Specifically, the results of k-means cannot be repeated because of the randomness of the initial seed. Non-deterministic investment strategies are not

Table 1. Comparison of Features and Clustering Methods.

	R-M	**D-M (Ours)**	D-K
Total Ret. (↑)	208.8%	**283.5%**	272.6%
Daily Sharpe (↑)	0.44	**0.50**	0.49
Max Drawdown (↑)	**-55.6%**	-60.5%	-59.0 %
Daily Mean Ret. (↑)	9.7 %	**11.1%**	10.9%
Monthly Mean Ret. (↑)	8.7%	**10.0** %	9.9%
Yearly Mean Ret. (↑)	9.6%	10.0 %	**11.19%**
Win Years (↑)	64.71%	**69.52%**	66.31%

acceptable to financial users in practice as they add another source of uncertainty (risk) that is hard to quantify.

Comparison with Funds To further analyse the effectiveness of our strategy, we compare our strategy with well known public funds in stock market in Table 2. Specifically, we select 2 big funds (CCA and VXX) and the top 3 best performed funds (IEO, PXE, PXI) recommended by YAHOO (`https://finance.yahoo.com/etfs`). Note that the ranking of funds change over time. The fund data is obtained from Yahoo Finance. Because VXX starts from 20/01/2009, this evaluation is computed over 2K trading days (20/01/2009-09/01/2017). From Table 2, our portfolio achieved the highest returns: Total (215.4%), daily (16.7%), monthly (16.6%), yearly (11.8%) in 2000 trading days, showing the strong profitability of our strategy. We also achieved the highest daily Sharpe ratio (0.8), meaning that we effectively balance the profitability and variance. We achieve the 2nd lowest max drawdown, meaning that our method can effectively manage the investment risk. In most years (62.5%), our portfolio makes a profit. It is only slightly worse than PXI in terms of 75.0% of profitable years. This shows the stability of our strategy.

Table 2. Comparison with Well-known Funds

	CCA	VXX	IEO	PXE	PXI	**Ours**
Total Ret.	117.0%	-99.9%	89.9%	101.6%	152.2%	**215.4%**
Daily Sharpe	0.7	-1.1	0.4	0.4	0.6	**0.8**
Max Drawdown	**-22.2%**	-99.9%	-56.8%	-57.6%	-59.3%	-30.9%
Daily Mean Ret.	10.9%	-67.7%	12.7%	13.4%	15.9%	**16.7%**
Monthly Mean Ret.	10.7%	-66.4%	11.5%	12.4%	14.9%	**16.6%**
Yearly Mean Ret	6.5%	-44.6%	5.0%	8.2%	9.4%	**11.8%**
Win Years	62.5%	0.0%	62.5%	50.0%	**75.0%**	62.5%

4 Conclusions

We propose a deep learned-based investment strategy, which includes: (1) novel stock representation learning by deep CAE encoding of candlestick charts, (2) diversification through modularity optimisation based clustering and (3) portfolio construction by selecting the best Sharpe ratio stock in each cluster. Experimental results show: (a) our learned stock feature captures semantic information and (b) our portfolio outperforms the FTSE 100 index and many well-known funds in terms of total return.

Acknowledgements This work was supported by EPSRC (EP/R026173/1), the European Union's Horizon 2020 research and innovation program under grant agreement No 640891, and National Natural Science Foundation of China No. 61773248.

5 References

[1] Harry Markowitz, "Portfolio selection," *The journal of finance*, vol. 7, no. 1, pp. 77–91, 1952.

[2] John L Kelly, "A new interpretation of information rate," *Bell Labs Technical Journal*, 1956.

[3] Olivier Ledoit and Michael Wolf, "Improved estimation of the covariance matrix of stock returns with an application to portfolio selection," *Journal of empirical finance*, vol. 10, no. 5, pp. 603–621, 2003.

[4] Gary Chamberlain and Michael Rothschild, "Arbitrage, factor structure, and mean-variance analysis on large asset markets," 1982.

[5] Jonathan Masci, Ueli Meier, Dan Cireşan, and Jürgen Schmidhuber, "Stacked convolutional auto-encoders for hierarchical feature extraction," *Artificial Neural Networks and Machine Learning–ICANN 2011*, pp. 52–59, 2011.

[6] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, 1998.

[7] Lawrence Rabiner and Biing-Hwang Juang, *Fundamentals of Speech Recognition*, Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1993.

[8] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, et al., "The kaldi speech recognition toolkit," in *IEEE 2011 workshop on automatic speech recognition and understanding*. IEEE Signal Processing Society, 2011.

[9] Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, et al., "Deep speech 2: End-to-end speech recognition in english and mandarin," in *ICML*, 2016, pp. 173–182.

[10] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al., "Mastering the game of go with deep neural networks and tree search," *Nature*, 2016.

[11] Carven von Bearnensquash, "Paper gestalt," in *In Secret Proceedings of Computer Vision and Pattern Recognition (CVPR)*, 2010.

[12] David F. Fouhey, Daniel Maturana, and Rufus von Woofles, "Visually identifying rank," in *ACH Special Interest Group on Harry Quetzcoatl Bovik (SIGBOVIK)*, 2015.

[13] Thomas Fischer and Christopher Krauss, "Deep learning with long short-term memory networks for financial market predictions," *European Journal of Operational Research*, 2017.

[14] Ritika Singh and Shashi Srivastava, "Stock prediction using deep learning," *Multimedia Tools and Applications*, vol. 76, no. 18, pp. 18569–18584, 2017.

[15] Narasimhan Jegadeesh and Sheridan Titman, "Returns to buying winners and selling losers: Implications for stock market efficiency," *The Journal of finance*, vol. 48, no. 1, pp. 65–91, 1993.

[16] SR Nanda, Biswajit Mahanty, and MK Tiwari, "Clustering indian stock market data for portfolio management," *Expert Systems with Applications*, 2010.

[17] Vincenzo Tola, Fabrizio Lillo, Mauro Gallegati, and Rosario N Mantegna, "Cluster analysis for portfolio optimization," *Journal of Economic Dynamics and Control*, 2008.

[18] Mark EJ Newman, "Modularity and community structure in networks," *Proceedings of the national academy of sciences*, 2006.

[19] William F Sharpe, "The sharpe ratio," *The journal of portfolio management*, 1994.

[20] Karen Simonyan and Andrew Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[21] Laurens van der Maaten and Geoffrey Hinton, "Visualizing data using t-sne," *Journal of Machine Learning Research*, pp. 2579–2605, 2008.