The machine learning algorithms to the market return

Ran You

The machine learning algorithms to the market return

Ran You[1][0000−0002−5911−5721]

Central University of Finance and Economics, Haidian District Beijing 100098, China
2017310812@email.cufe.edu.cn

Abstract. The financial market and stock market have experienced great changes during the past decades, which lead to lots of excellent new methods about how to calculate and predict the return of the market and stocks.
Currently, a common practice of stock investors is to implement the Capital Asset Pricing Model (CAPM) and calculate excess returns with the difference between the real value and the theoretical value of stocks. Among Capital Asset Pricing Models, the Fama-French factors model is used frequently.
In this paper, we applied several machine learning algorithms in the stock market to find out whether they are useful in predicting the stock price and what are the possible reasons behind the results. With proper data, the result can also predict the return of the market with excess return in Fama-French three factors model.
The machine learning algorithms were used to predict the price of S&P500, using data of S&P500 constituents. It is chosen because S&P500 data is a standard of the return of the market, and the data of its constituents are highly related to the excess return α. Then the machine learning algorithms were proposed to identify the relationship between the excess return of the last day and the return of the market of the next day.
This research showed that some machine learning algorithms can do the prediction well with proper models and parameters. Besides, the return of the market can be predicted with proper data.

Keywords: Machine learning, Stock market, Return of market, S&P500

1 Introduction

In recent years, with the development of computer science and technology, deep learning becomes a popular area for researchers. Comparing with traditional approaches, it has characteristic of preciser prediction and forecast, therefore, many researchers in traditional areas begin to turn to the method of deep learning. Combining traditional models and deep learning, researchers usually can get precise and explainable results.

In this paper, multiple methods in deep learning are used to get preciser prediction in the stock market, such as deep neural network (DNN), convolutional

neural network (CNN) and recurrent neural network (RNN). As a classical data set, Standard and Poor's 500 (S&P500) data and part of its constituents are used for analysis. In order to get a more accurate and explainable conclusion, the data across twenty years (2000.01.03-2020.12.31) is chosen. And the result has shown, deep learning methods perform very well in the field of econometrics, although there are differences in results among each method.

Besides, compared to the trend, most investors pay more attention to whether the price goes up or down. So this paper also do prediction on classification problem with the deep learning method.

Meanwhile, Capital Asset Pricing Model (CAPM) is a bunch of models that cannot be avoided when study stocks and return. Among CAPMs, Fama-French three-factor model is the most convincing and concise one. The model divides the change of stocks' price into several parts, and then calculate the value of each part under given assumptions. It also performs well in many circumstances. In the paper, we try to find out how deep learning methods work. So Fama-French three-factor model is applied to fit the value of each parameter, and then release the relationship among parameters.

Note that in the Fama-French model, different coefficients are supposed to be independent. But it is not always true in reality, and deep learning methods are able to extract these conditions and apply them to prediction procedures. On the other hand, in traditional prediction procedures, researchers might treat these conditions as random errors, so that left alone important hints. With the help of validation methods like cross-validation, deep learning methods are able to detect trends and abandon random errors efficiently.

In the meantime, as a classical method, ordinary least squares(OLS) has been proven to be a useful method in many fields as well as econometrics. So in the process of applying the Fama-French three-factor model, OLS is used to explore the relationship between coefficients and help to predict the return of the market in the Fama-French three-factor model.

2 Literature review

The term machine learning is introduced by Arthur Samuel [1]. Before that, machine learning had other names include cybernetics and connectionism[2]. It is usually referred to statistical methods through a computer program.

The neural network was firstly introduced by Mcculloch [3] in 1943. The idea was inspired by the structure of the nervous system in human brains. However, it was not taken seriously by scientists until the invention of backpropagation [4]. With the development of computer technology, more researchers came up with better and powerful algorithms, and apply their algorithms to real problems. In 1988, a convolutional neural network was introduced by Zhang [5]. Weng, Ahuja and Huang introduced max-pooling method to help recognize 3D objects [6] in 1992. A recurrent neural network was invented based on the research of Rumelhart [4] to deal with sequential data input. Long short term memory was

introduced by Hochreiter and Schmidhuber [7]. After LSTM was invented, it began to change the area of speech recognition.

Researchers started to use statistical methods as tools to solve and describe problems in economics a long time ago. In 1962, Okun proposed Okun's law to describe the relationship between unemployment and losses in a country's production with the statistical model of linear regression [8]. But until 1984, machine learning algorithms were applied to economics by Cohen et al [9]. In 1988, the neural network was used on predicting the stock return of IBM by White [10]. From then on, machine learning algorithms started to be widely used in the field of economics. For example, in 2019, Storm, Baylis and Heckelei explored the application of machine learning in econometrics [11].

Machine learning algorithms are also applied in different areas including engineering, biology, etc. In 1999, Adeli and Yeh designed a perceptron learning model to solve problems in the field of engineering [12]. Reich and Barai evaluate the performance and shortcomings of artificial neural networks in engineering in 1999 [13]. Tarca [14] introduced several machine learning methods and applied them to a biological databases. Sommer [15] elaborated methods of applying machine learning methods in microscopy assays and optimizing experimental workflow.

When it comes to the CAPM model, it was independently introduced by Treynor, Sharpe [16], Lintner [17] and Mossin [18]. The well-known Fama-French three-factor model was introduced by Fama and French in 1993 [19], and extended to five factors in 2015 [20]. In 2020, Chen, Pelger and Zhu [21] combined deep learning with traditional asset pricing theory, and show accurate and explainable results.

3 Methodology

3.1 Deep neural network

The deep neural network (DNN) is a kind of ordinary neural network with at least three layers. The deep neural network is one of the basic kinds of neural network, since it contains the basic elements of the neural network, such as neurons, biases and weights. At the begining, DNN is feedforward. When trying to minimize the loss function, DNN will be tuning weights between neurons. In the paper, DNN is chosen as a fundamental method. A model with only one neuron is found as a linear model, as well as a classical DNN model with three layers. With the performance of DNN, the performance of other deep learning models can be measured.

3.2 Recurrent neural network(RNN)

The recurrent neural network is a kind of neural network, basing on Rumelhart's work in 1986 and Hopfield's work in 1982. RNN is designed for better processing of temporal information. It introduces state variables to store past information and uses them together with the current input to determine the current output. RNN is often used to process sequential data, such as a piece of text or sound, the order of shopping or watching a movie, or even a row or column of pixels in an image. Therefore, cyclic neural networks have a wide range of practical applications, such as language modeling, text classification, machine translation, speech recognition, image analysis, handwriting recognition and recommendation systems.

Generally speaking, when getting input, the recurrent neural network would treat the input as a sequence, and then train the parameters in the model with one input at a time. In this case, the parameters in RNN would be trained several times before output parameters. This is also the reason why it is called recurrent.

In RNN, the result of the training process would also be a sequence. In normal RNN models, only the last output value is useful since researchers usually pay less attention to its training process. The paper also uses simple RNN to build models for time series forecasting.

Long short term memory(LSTM) In RNN, there are different kinds of neurons to deal with data from the past properly. This is the reason why LSTM and GRU is introduced. Long short term memory is proposed by Hochreiter and Schmidhuber in 1997. It is designed to deal with gradient explosion and gradient vanish when training long sequences.

The basic elements of LSTM include a cell, an input gate, an output gate and a forget gate. The forget gate would decide how to handle data from the past, including whether to forget data on one day or not and whether data is very important or of little importance. In this case, the LSTM is better than a simple RNN. The paper uses LSTM and combine it with CNN.

3.3 Convolutional neural network(CNN)

The convolutional neural network, also calls as shift invariant or space invariant artificial neural networks (SIANN), is a type of deep neural network. It is often used in computer vision and image classification.

In a deep neural network, each layer of the network is fully connected to the adjacent layer. However, this does not take into account the spatial distribution of the pixels in the image. It does not make any difference whether the two pixels are very close or very far away from each other, which is obviously unreasonable. CNN is created, so that the neural network can detect patterns near each neuron and response to those patterns.

Since neurons in this layer do not have to connect with every neuron in the next layer, the number of parameters of the CNN model is much lower than the normal deep neural network. So that it usually costs less time to train compared to a fully connected DNN.

3.4 Ordinary least squares(OLS)

Ordinary least squares (OLS) is a kind of classical method to estimate unknown coefficient under fixed forms of formulas. Generally speaking, OLS aims at minimizing the loss of the true values Y and the estimations $\bar{Y}$.

The OLS method is used to find out the relationship of price (including α) and the return of the stock R_m. It is introduced as a basic and commonly used model in the paper. And a simple check of other models can be done with OLS model (the result of other model should be better than OLS model).

Then for each stock, Pearson correlation coefficients are computed between excess return α on the last day and the return of market R_m on the next day. So that it can be known that whether the relation between α and R_m is linear or more complex.

3.5 Some other set of models

In this paper, methods including OLS, DNN, RNN, CNN are used to find the best machine learning algorithms in the forecast of stock price. And since the time series data of S&P500 is not stationary, this paper uses an ordinary method in time series problems of taking its first order difference. And it becomes rather stationary from the figure (at least the average of time series becomes 0).

Metrics In those models, we use mean absolute error (MAE) as metric. The formula of MAE is:

$$MAE = \frac{\sum_{i=1}^{n} (y_i - x_i)}{n} = \frac{\sum_{i=1}^{n} e_i}{n} \tag{1}$$

This is because mean absolute percentage error (MAPE) is more suitable for the time series problems, but it cannot handle data that is near 0 properly. The formula of MAPE is as below:

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\frac{A_t - F_t}{A_t} \tag{2}$$

in which A_t is the actual value and F_t is the forecast value. We can know from the formula, if A_t is near 0, and even if A_t and F_t are very close, the value of MAPE still can be significantly large. On the other hand, although it is not appropriate to use MAE in time series forecasting, but it can handle data near 0 properly. Since the data is transformed as stationary time series data, MAE is chosen as metric.

Model selection in Keras In deep learning, it is possible that the model can show the overfitting problem. This problem shows up when the metric of the model goes up on the training set but goes down on the validation set and test set. Since one of the main functions of the validation set is to restrict overfitting problems, the training of the model can be early stopped if the metric of the model keep going down on validation set. In this paper, the patience is set as 50. In other words, if the metric of the model keep going down on 50 epochs, the training process stops and returns the best model before 50 epochs.

About data and variables This paper uses the data of S&P500 from 2000-01-03 to 2020-12-31 for data analysis obtained from Wharton Research Data Services (WRDS) and Yahoo Finance. This is because the Standard & Poor's Index is a stock price index compiled by Standard & Poor's Company, the largest securities research institution in the United States. More than 90% of its more than 500 constituent stocks are listed on the New York Stock Exchange. As a result, it is more representative than the Dow Jones Average, so it can more accurately reflect the changes in stock prices. Besides, S&P500 data is able to reflect the return of the market R_m properly.

The paper used 206 constituent stocks of S&P500 as a predictive variables. This is because these 206 stocks are constituent stocks for 20 years, so that consistency, as well as interpretability, can be enhanced. Besides, constituent stocks of S&P500 are able to show the excess return of stocks since they are typical and are chosen by Standard & Poor's Company.

Along with these constituent stocks, 7 indicators are used to describe each stock. For detailed information on indicators, please check the appendix below.

Train-test split and standardization The paper split data as a training set, validation set and test set. The ratio of each set is 0.7, 0.2, 0.1. It means in 5284 trading days during 20 years, first 70% data are used to train models, 20% of data in the middle are used to validate the parameters when training each model. And this paper used the last 10% of data to test the performance of each model.

For each of the seven indicators, standardization is implemented. Since only the information from the training set can be used, the mean of the training set

of each indicator is set as 0 and the variance is set as 1. The mean and variance of the training set are used to standardize data invalidation set and test set.

Besides, compared to standardization, the training process on data that is standardized is faster than on data that is not standardized. It can fast about 30% to 40%.

3.6 Fama-French three factor model

Fama-French three-factor model is a CAPM model. It uses some of the data in the stock market to calculate the values of three factors each day. The formula of the Fama-French three-factor model is as below:

$$r = R_f + \beta * (R_m - R_f) + b_s * SMB + b_v * HML + \alpha \tag{3}$$

where SMB stands for Small market capitalization Minus Big and HML stand for High book-to-market ratio Minus Low. α stands for exceed return of a stock. These two factors along with R_f are released on the website of Fama-French every day. Since the form of the model is a linear regression with intercept, the paper use the OLS method to compute α (the intercept) and the relationship between α and R_m. This is because, from the assumption of OLS, each parameter should be independent of other parameters. However, in reality, this assumption usually cannot hold. The paper wants to use the method to show machine learning methods can process time series forecasting on stock data better than the OLS model.

4 Result

In this part, several machine learning models are built for the purpose of time series forecasting on S&P500 stock price.

4.1 Linear model

Table 1 shows the structure of the linear model.

Layer (type)	Output Shape	Param #
$dense_1(Dense)$	$(None, 1)$	43471

Table 1. Structure of linear model

Since we want to find a good learning rate to avoid the problem caused by the learning rate, the learning rate is chosen as 10^{-5} here with proper methods.

Figure 1 and Figure 2 shows the performance of the model on the test set and validation set.

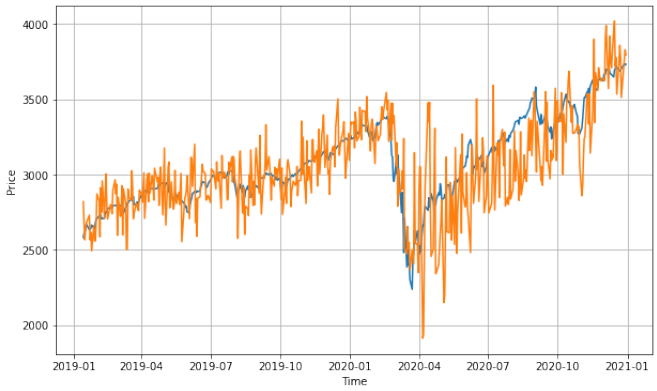

Fig. 1. performance of the linear model on the test set

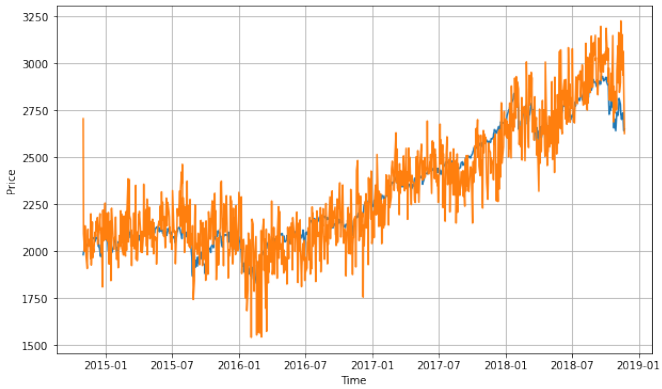

Fig. 2. performance of the linear model on the validation set

In these two plot, the blue line is the original data and the orange line is the prediction of linear model. As we can see, the linear is very unstable when dealing with time series forecasting. And it is not a big problem since linear model is just a test and a beginning.

4.2 Deep neural network

Then, the DNN model is used. Table 2 shows the structure of DNN in the paper.

Figure 3 and Figure 4 shows the performance of DNN on validation set and test set.

In these two plots, the blue line stands for original data and the orange line is the prediction of the DNN model. As it shows, the DNN model is much more stable and accurate than the linear model above.

Layer (type)	Output Shape	Param #
$dense(Dense)$	$(None, 256)$	11128576
$dense_1(Dense)$	$(None, 128)$	32896
$dense_2(Dense)$	$(None, 1)$	129

Table 2. Structure of DNN

Fig. 3. performance of the DNN model on test set

4.3 Recurrent neural network

Since the RNN model is widely used in time series problems, it is then used on the same data set. The structure of the RNN model is shown in Table 3.

Layer (type)	Output Shape	Param #
$simpleRnn(SimpleRNN)$	$(None, 30, 256)$	436736
$simpleRnn_1(SimpleRNN)$	$(None, 128)$	49280
$dense(Dense)$	$(None, 1)$	129

Table 3. Structure of RNN

Notice that the data structure of input is a little different from the linear model and DNN, since RNN needs different input data structure. This is also the reason that the number of parameters RNN trained is much smaller than DNN. The performance of RNN on the test set and validation set is shown in Figure 5 and Figure 6.

in which the blue line is original time series data and the orange line is the prediction of RNN model. As we can see, the RNN model fit the time series data very well.

Fig. 4. performance of the DNN model on validation set

Fig. 5. performance of the RNN on test set

4.4 Convolutional neural network

Although the CNN model is not often used in time series forecasting, the paper still tried CNN. The structure of the CNN model is shown in Table 4.

CNN is used to extract features in time series data and pass features to the LSTM layer. LSTM layers then do the job of forecasting. The performance of CNN is shown in Figure 7 and Figure 8.

in which the blue line is the original time series data and the orange line is the forecasting of the CNN model. As we can see, the performances of DNN, RNN and CNN are very similar. In order to show the difference, MAE of each result is calculated.

Fig. 6. performance of the RNN on validation set

Layer (type)	Output Shape	Param #
$conv1d(Conv1D)$	$(None, 30, 32)$	231872
$lstm(LSTM)$	$(None, 30, 256)$	295936
$lstm_1(LSTM)$	$(None, 128)$	197120
$dense(Dense)$	$(None, 1)$	129

Table 4. Structure of DNN

5 Conclusion and discussion

The result of each model is shown as Table 5.

	DNN	RNN	CNN
test set	326.746	323.739	321.171
val set	338.608	331.752	333.380

Table 5. MAE of models

As we can see, the linear model performs unstable and inaccurate. The DNN model performs more stable than the linear model. The result of the RNN and CNN model is very close, since the MAE of the result on validation set and test set are very similar. And two plots also demonstrate the statement.

However, in the real world, investors usually care about whether they could earn money more than the trend of the stock market in a few days. In other words, they want to know the price, especially close price, compared to the last day. So the paper also builds and compares several classification models based on RNN and CNN to show which method is more accurate. As the result, all models seem to be well, 58% is the best accuracy they can get. If additional

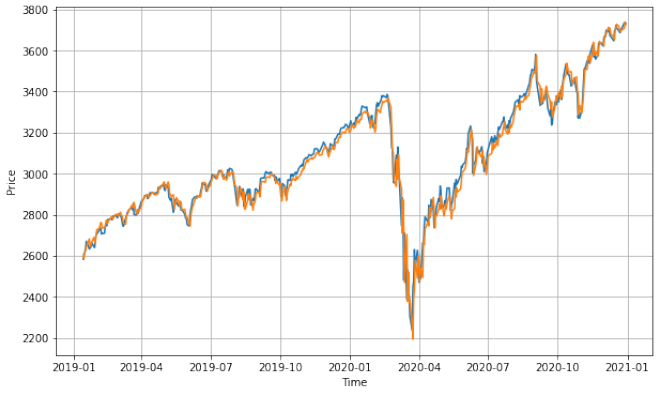

Fig. 7. performance of the CNN on test set

Fig. 8. performance of the CNN on validation set

information is added, such as the macro data, models are able to get higher accuracy.

The relation between the return of market Rm and the excess return α of Fama-French three-factor model is demonstrate. It means if someone can acquire information of of all stocks each day, he can also predict the circumstance of the stock market tomorrow. Although it is a betray of the statistical assumption of OLS in Fama-French three-factor model, the result is not that hard to understand. More and more investors tend to focus on α to get an excess return, and if one α changes a lot today, investors are more likely to buy in or sell out this stock, and this will lead to the change of the market.

However, R_m and α in Fama-French data has always been very little, which means if only investors want to get a return in the stock market, they need to undertake risks. And this feature might make the relationship between α and R_m vulnerable to changes. Besides, if some reason other than the stock market itself

influence the market a lot, such as an epidemic, α would not influence R_m very much. This kind of systematic influence is very hard to predict and calculate.

On the other hand, the result is based on the dataset of 20 years (2000-2020) on 206 stocks. If a shorter time period relationship between α and R_m is considered, the result can be unconvincing, since other effects can influence the stock market more than itself.

From the result, the relation between R_m and α can be calculated, and this might be the reason why machine learning algorithms works when predicting the SP500 index. However, the mechanism needs further research. But if we use the OLS method to detect linear correlation between R_m and α, we will find that there is no linear correlation between these two coefficients (about -0.006). In this case, the relationship between R_m and α must be more complex, and it still needs further study.

Appendix

Description of SP500 data

The description of 7 indicators are as below:

"permno": It is a unique integer number for each stocks. Most of them are larger than 1e4 and less than 1e5. In order to difference S&P500 data and constituent stocks, the paper set the permno of S&P500 as 0. It is standardized before training in models.

"bidlo": It stands for "Bid or Low Price". Bid price is the highest price that a buyer (i.e., bidder) is willing to pay for a goods. So if there is no trade in a trading day, the low price will be filled by bid price. The paper use the low price of S&P500 when filling S&P500 into the structure of data.

"askhi": It stands for "Ask or High Price". Ask price is the price a seller states they will accept. So if there is no trade in a trading day, the high price will be filled by ask price. The paper use the high price of S&P500 when filling S&P500 into the structure of data.

"prc": It stands for "Price or Bid/Ask Average". It is the average of bid price and ask price. The paper use the close price of S&P500 when filling S&P500 into the structure of data.

"vol": It stands for "Volume Traded". It is the volume traded in the trading day. A large volume means that lots of people buy and sell that stock in the trading day. The paper use the volume of S&P500 when filling S&P500 into the structure of data.

"ret": It stands for "Holding Period Total Return". It is the return of the stock. In the paper, it is the return in a trading day. As for return of S&P500,

the paper use the average of its constituent stocks in each day to calculate it.

”shrout”: It stands for ”Shares Outstanding”. Shares outstanding refer to a company’s stock currently held by all its shareholders. As for shares outstanding of S&P500, the paper use the average of its constituent stocks in each day to calculate it.

References

[1] Arthur L Samuel. Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, 3(3):210–229, 1959.

[2] Ian Goodfellow, Yoshua Bengio, Aaron Courville, and Yoshua Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.

[3] Warren S McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5 (4):115–133, 1943.

[4] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. Learning representations by back-propagating errors. *Nature*, 323(6088):533–536, 1986.

[5] Wei Zhang et al. Shift-invariant pattern recognition neural network and its optical architecture. In *Proceedings of annual conference of the Japan Society of Applied Physics*, 1988.

[6] Juyang Weng, Narendra Ahuja, and Thomas S. Huang. Matching two perspective views. *IEEE Computer Architecture Letters*, 14(08):806–825, 1992.

[7] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[8] Arthur M Okun. *Potential GNP: its measurement and significance*. Cowles Foundation for Research in Economics at Yale University, 1963.

[9] Michael D Cohen and Robert Axelrod. Coping with complexity: The adaptive value of changing utility. *The American Economic Review*, 74(1):30–42, 1984.

[10] Halbert White. Economic prediction using neural networks: The case of ibm daily stock returns. In *ICNN*, volume 2, pages 451–458, 1988.

[11] Hugo Storm, Kathy Baylis, and Thomas Heckelei. Machine learning in agricultural and applied economics. *European Review of Agricultural Economics*, 47(3):849–892, 2020.

[12] H Adeli and C Yeh. Perceptron learning in engineering design. *Computer-Aided Civil and Infrastructure Engineering*, 4(4):247–256, 1989.

[13] Yoram Reich and SV Barai. Evaluating machine learning models for engineering problems. *Artificial Intelligence in Engineering*, 13(3):257–272, 1999.

[14] Adi L Tarca, Vincent J Carey, Xue-wen Chen, Roberto Romero, and Sorin Drăghici. Machine learning and its applications to biology. *PLoS Comput Biol*, 3(6):e116, 2007.

[15] Christoph Sommer and Daniel W Gerlich. Machine learning in cell biology–teaching computers to recognize phenotypes. *Journal of cell science*, 126 (24):5529–5539, 2013.

[16] William F Sharpe. Capital asset prices: A theory of market equilibrium under conditions of risk. *The journal of finance*, 19(3):425–442, 1964.

[17] John Lintner. The valuation of risk assets and the selection of risky investments in stock portfolios and capital budgets. In *Stochastic optimization models in finance*, pages 131–155. Elsevier, 1975.
[18] Jan Mossin. Equilibrium in a capital asset market. *Econometrica: Journal of the econometric society*, pages 768–783, 1966.
[19] Eugene F Fama and Kenneth R French. Common risk factors in the returns on stocks and bonds. *Journal of Financial Economics*, 33:3–56, 1993.
[20] Eugene F Fama and Kenneth R French. A five-factor asset pricing model. *Journal of financial economics*, 116(1):1–22, 2015.
[21] Luyang Chen, Markus Pelger, and Jason Zhu. Deep learning in asset pricing. *Available at SSRN 3350138*, 2020.