JPX Tokyo Stock Exchange Prediction

Sriya Gorrepati
Computer Science
Utah State University
Logan, USA
a02370648@usu.edu

Yingying Deng
Computer Science
Utah State University
Logan, USA
A02394960@usu.edu

***Abstract*—The stock market is very unpredictable, any geopolitical change can impact the share trend of stocks in the share market, recently we have seen how covid-19 has impacted the stock prices, which is why on financial data doing a reliable trend analysis is very difficult. Numerous quantitative trading approaches are currently employed to analyze financial markets and develop investment strategies. Our project focuses on utilizing financial data specific to the Japanese market, with the aim of empowering retail investors to conduct comprehensive market analysis. The dataset contains the information from 2017 to 2021. Our objective is to rank 2000 stocks in descending order and the build portfolios from the predicted 2000 stocks using various models such as Linear Regression, Lasso Regression, LSTM, Adaboost, and LightGBM. To effectively handle the large dataset and maximize exploration, we have divided the project into two approaches. After preprocessing, we performed data analysis and visualization, followed by evaluating the performance of the models using appropriate evaluation metrics and the Sharpe Ratio of the daily spread return.**

***Index Terms*—Stock Market, Machine Learning, Linear Regression, Lasso Regression, LSTM, LightGBM, Evaluation Metrics, Sharpe Ratio**

I. Introduction

Japan's JPX Exchange is one of the world's largest stock exchanges. Predicting JPX stock movements is difficult, like any stock market. There are several ways to study historical data and patterns to predict stock values.

Fundamental, technical, and machine learning techniques predict JPX stock. Fundamental analysis uses financial and economic data to estimate a company's value and prospects. Technical analysis uses past market trends and patterns to find trading opportunities.

Machine learning algorithms analyze massive datasets using artificial intelligence and statistical models to find patterns and trends for prediction. These algorithms can be trained on past JPX market data to forecast market behavior.

Predicting JPX stock prices is difficult, but investors and traders can maximize results by using many analysis approaches and tools. JPX, also known as Nippon Torihikijo, is a Japanese financial instruments exchange holding company that follows the Financial Instruments and Exchange Act and Financial Services Agency regulations. TSE, OSE, and TOCOM are JPX's licensed financial instruments exchange corporations. On January 1, 2013, TSE and OSE merged to form JPX's primary securities exchange and largest derivatives exchange. JPX bought TOCOM in 2019 to increase its commodity derivatives trading.

Financial professionals evaluate markets and build investment strategies using quantitative trading methodologies. Buy low stocks, sell overvalued ones. Retail investors, who may have trouble getting historical and real-time data, must know how and when to buy and sell stocks. JPX runs one of the world's largest stock exchanges. In a JPX-hosted Kaggle competition, data scientists and retail investors can examine Tokyo Stock Exchange data for the Japanese market. Machine learning allows these individuals to investigate quantitative trading and make informed selections based on model predictions.

II. Related Work

Academics and industry experts have long studied stock market forecasting. Predicting stock values has been attempted using everything from econometric methods to machine learning. This thorough page discusses Brown and Kane's (1978) studies on value stock risk and return, Campbell, Lo, and MacKinlay's (1997) book "The Econometrics of Financial Markets," and others on stock prediction. Stock market studies by Fama (1965), Granger (1986) on cointegrated economic variables, Huang, Shen, and Zhou (2018) on recurrent neural networks (RNNs) for prediction, Lai, Wang, Huang, and Liu (2015) on machine learning algorithms for stock market forecasting, Lo and MacKinlay (1999) "A Non-Random Walk Down Wall Street," Zhang, Patuwo, and Hu (1998) reviewed forecasting with artificial neural networks, and Tsantekidis, Passalis, Tefas, and Kanniainen (2017) used limit order book data to predict stock prices using CNNs.

Brown and Kane (1978) examined how risk affects value stock returns. Value shares beat growth businesses over the long term, despite stock prices being considered to reflect all essential information. Fama (1965) found that stock prices behave like a random walk, with little useful information from prior price fluctuations. Campbell, Lo, and MacKinlay's "The Econometrics of Financial Markets" (1997) covers a variety of econometric methods for modeling and forecasting financial time series data, including the autoregressive integrated moving average (ARIMA) model, the generalized autoregressive conditional heteroskedasticity (GARCH) model, and the vector autoregression (VAR) model. Granger (1986) found that many

non-stationary time series can be integrated into a stationary connection using cointegration, laying the framework for more advanced stock prediction models.

As machine learning has advanced, researchers have explored stock prediction algorithms. Huang, Shen, and Zhou (2018) suggested using recurrent neural networks (RNNs) to predict stock prices since they can incorporate temporal dependencies. LSTM and GRU RNN variants performed well in prediction accuracy. Lai, Wang, Huang, and Liu (2015) used SVM, RF, and KNN to predict stock prices. SVM has the best prediction accuracy. Tsantekidis, Passalis, Tefas, and Kanniainen (2017) presented CNNs to anticipate stock prices using the limit order book. In recent decades, interest in stock market forecasting using machine learning and other AI has grown. This comprehensive essay will review various research papers that have used machine learning algorithms and statistical methods to predict the stock market.

"The Behavior of Stock-Market Prices," published in 1965 in the Journal of Business, was one of Fama's first studies in this subject. Fama's claim that stock prices follow a random walk is that future price fluctuations are unpredictable and unaffected by past price movements. This raised doubt on stock market prediction using conventional statistical methods.

Recent research has questioned Fama's random walk theory and examined whether machine learning can predict stock prices. Zhang et al. published "Forecasting with Artificial Neural Networks: The State of the Art" in the International Journal of Forecasting in 1998. The authors suggested ANNs for stock market forecasting. Artificial neural networks (ANNs) model the human brain in form and function. They can spot patterns in vast datasets and draw conclusions. Zhang et al. (1998) reviewed the state of the art in stock market prediction using ANNs and assessed their merits and cons.

Stock market forecasting has also used other machine learning approaches besides ANNs. Lai et al. presented "Stock Market Forecasting Using Machine Learning Algorithms" at the 2015 IEEE/ACIS 16th International Conference on Software Engineering, AI, Networking, and Parallel/Distributed Computing (SNPD). Support vector machines, decision trees, and k-nearest neighbors were examined for stock market forecasting. Support vector machines are particularly good at stock price prediction.

Stock market predictions have been made using RNNs. Huang et al. published "Stock Market Prediction with Recurrent Neural Network" in IEEE Transactions on Industrial Informatics in 2018. Technical indicators including moving averages and the relative strength index were incorporated in an RNN-based stock market forecasting method. Their method predicted better than normal statistical approaches.

Deep learning, which uses multi-layered neural networks, has become popular for stock market prediction. Mallqui and Larriva-Novo wrote "Deep Learning for Stock Market Prediction: A Comparative Study" for Expert Systems with Applications in 2019. The authors compared the stock market predictions of numerous deep learning models, including CNNs and LSTM networks. LSTM networks, a deep learning model, predicted stock values well.

Machine learning and statistics have predicted stock market outcomes. Granger's 1986 Oxford Bulletin of Economics and Statistics article "Developments in the Study of Cointegrated Economic Variables" provides an example. Granger defined cointegration as a stable linear combination between non-stationary variables in long-term equilibrium. Time-series analysis uses cointegration to replicate the relationship between stock prices and economic factors like interest rates and inflation for stock market predictions.

Engle's 1982 Econometrica article, "Autoregressive Conditional Heteroskedasticity with Estimates of the Variance of United Kingdom Inflation," is another major stock market forecasting study. Engle calls time-varying financial volatility autoregressive conditional heteroskedasticity (ARCH). Stock market volatility is useful for risk management and portfolio optimization, hence ARCH and GARCH models are widely used to model and predict it.

Stock market forecasting is also an example of ensemble approaches, which combine multiple models to get a conclusion. Zhang et al. (2019) published "Stock Market Prediction with Ensemble Methods" in the Journal of Applied Sciences. Random forests, SVMs, and DNNs were proposed for stock market forecasting. Their ensemble strategy predicted more accurately and consistently than any individual model.

Stock market forecasting evolves constantly. Market conditions, data quality, and prediction time can also affect these methods. Thus, these methods must be thoroughly tested before being used in real-world trading decisions.

In conclusion, machine learning algorithms, deep learning methods, statistical approaches, and ensemble methods have greatly improved stock market prediction. These approaches have predicted stock prices, market trends, and volatility well. Due to the markets' complexity and volatility, stock market predictions are notoriously difficult. Stock market prediction models should be used cautiously due to the financial markets' inherent risks and unknowns.

III. Methodology

A. Data Collection and Preprocessing

The data set is on the JPX Tokyo stock exchange prediction and is chosen from the Kaggle. The goal is building portfolios from the predicted stocks (around 2,000 stocks). The size of the dataset is 1.33GB. The dataset contains of multiple files such as stock_list, supplemental files, train files which includes stock_prices.csv, secondary_stock prices, stock_list, finances, trades etc. Our main focus was on two core data files, namely the train file and the supplemental file, which include the stock_prices file consisting of 2,332,531 rows and 8 columns, and the stock list file containing 4,417 rows and 2 columns. The stock prices are calculated on a daily basis and provide information for each company. It's important to understand the terms "Target" which refers to the average return, and "Volume" which represents the number of shares traded.

Since it's a huge data set with multiple data files we want to explore it to the fullest. For this reason we divided

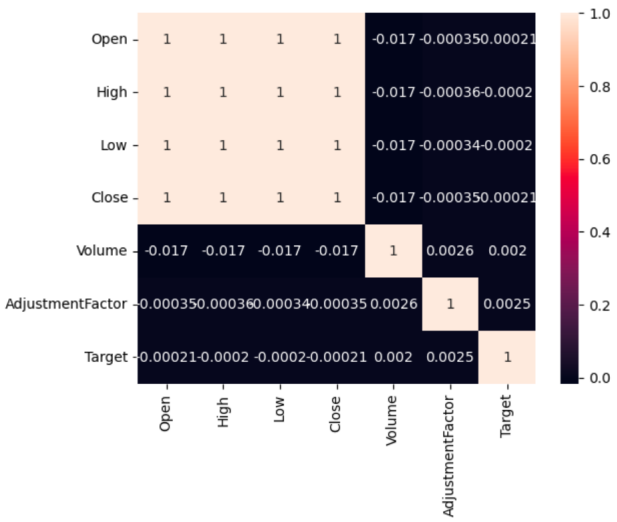


Figure 1. Data Correlation.

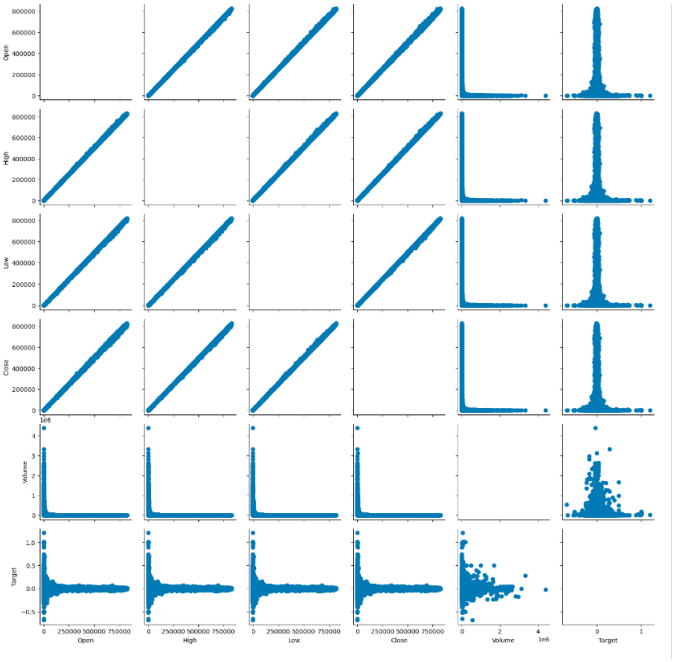


Figure 2. Scatter graphs

this project into 2 parts, approach 1 and approach2. In each approach we either dropped the missing data or set them to '0.' We dropped some of the columns such as 'RowId','ExpectedDividend','AdjustmentFactor','SupervisionFlag' which are not used for our analyzation.

1) Approach 1: In this approach, firstly, Feature selection is performed using two methods: heatmapping the correlation coefficients and plotting scatter graphs between pairs of variables. From the stock_prices file, The interested variables are Open, Close, High, Low, Volume and Target. Figure 1 shows Data correlation coefficients and Figure 2 is scatter graph. We observed that there is a strong correlation between Open,Close,High,Low prices. The trading volume decreases as the stock prices rise. Generally the high price stocks have low trading volume. The Target presents a normal distribution with other variables. Therefore, we picked up Open,Close,High,Low,Volume as input features, Target as output.

Secondly, in order to minimize the effect of data values,we used Z-scores to scale our data.In investing and trading, Z-scores are measures of an instrument's variability and can be used by traders to help determine volatility. Finally,we found that stock_prices and secondary_stock_prices files contain stock transaction information for different stock at the same times, thence we wanted to increase the size of sample data by merging these two files. The train data obtained by merging the two files from train file is 4617191 rows and 8 columns from 2017-01-04 to 2021-12-03. while the test data obtained by merging them from the supplemental file is 542908 rows and 8 columns from 2021-12-06 to 2022-06-24. The following Figure 3 and Figure 4 are train data and test data.

2) Approach 2: In approach 2, the focus is on predicting the 'Close' target variable. To ensure data integrity, all missing values were removed from the stock_list and stock_prices train files. Relevant attributes such as 'Date',

	Date	SecuritiesCode	Open	High	Low	Close	Volume	Target
0	2017-01-04	1301	-0.089131	-0.089355	-0.088372	-0.088874	-0.129920	0.000730
1	2017-01-04	1332	-0.150102	-0.150187	-0.149885	-0.149988	0.686797	0.012324
2	2017-01-04	1333	-0.076652	-0.076734	-0.075700	-0.059260	0.006154	
3	2017-01-04	1376	-0.123585	-0.122995	-0.123003	-0.122429	-0.135852	0.011053
4	2017-01-04	1377	-0.074043	-0.072744	-0.073044	-0.072322	-0.094679	0.003026

Figure 3. Train data.

'SecuritiesCode', 'Open', 'High', 'Low', 'Close', 'Volume', and 'Target' were extracted from stock_prices, while 'SecuritiesCode' and 'Name' were taken from stock_list. These two datafiles were merged into a single dataset called 'stocks' based on the common 'SecuritiesCode' attribute. After dropping any redundant rows and merging, the resulting dataset contains 2324923 rows and 12 columns. To prepare the data for modeling, it was split into a 70:30 ratio for training and testing respectively, and normalized using the min_max scaler. The following Figure.5 shows the newly formed merged stocks file.

	Date	SecuritiesCode	Open	High	Low	Close	Volume	Target
0	2021-12-06	1301	-0.084491	-0.085320	-0.084006	-0.084729	-0.150852	-0.003263
1	2021-12-06	1332	-0.149112	-0.149169	-0.148923	-0.149164	0.330741	-0.008993
2	2021-12-06	1333	-0.101092	-0.101235	-0.100529	-0.100797	-0.109173	-0.009963
3	2021-12-06	1375	-0.131861	-0.132021	-0.131554	-0.131987	-0.125132	-0.015032
4	2021-12-06	1376	-0.128914	-0.128458	-0.128413	-0.128551	-0.151814	0.002867

Figure 4. Test data

	Date	SecuritiesCode	Open	High	Low	Close	Volume	Target	Name
0	2017-01-04	1301	2734.0	2755.0	2730.0	2742.0	31400	0.000730	KYOKUYO CO.,LTD.
1	2017-01-05	1301	2743.0	2747.0	2735.0	2738.0	17900	0.002920	KYOKUYO CO.,LTD.
2	2017-01-06	1301	2734.0	2744.0	2720.0	2740.0	19900	-0.001092	KYOKUYO CO.,LTD.
3	2017-01-10	1301	2745.0	2754.0	2735.0	2748.0	24200	-0.005100	KYOKUYO CO.,LTD.
4	2017-01-11	1301	2748.0	2752.0	2737.0	2745.0	9300	-0.003295	KYOKUYO CO.,LTD.
...	...	...	...	...	...	...	...	...	...
2332526	2021-11-29	4169	6970.0	7350.0	6970.0	6970.0	772500	0.009972	ENECHANGE Ltd.
2332527	2021-11-30	4169	6770.0	7240.0	6410.0	7020.0	887400	0.060649	ENECHANGE Ltd.
2332528	2021-12-01	4169	7190.0	7380.0	6670.0	7090.0	496800	-0.039894	ENECHANGE Ltd.
2332529	2021-12-02	4169	7160.0	7870.0	7110.0	7520.0	783000	-0.127424	ENECHANGE Ltd.
2332530	2021-12-03	4169	7410.0	7680.0	7150.0	7220.0	316800	-0.036508	ENECHANGE Ltd.

2332531 rows × 9 columns

Figure 5. Merged stocks table.

B. Exploratory Data Analysis

During the exploration of the data set, our goal was to thoroughly analyze the data and conduct data analysis in order to observe and understand the relationships within the data.

To gain insights into the stock market trends from 2017 to 2021, we generated graphical representations of key attributes such as Close, Volume, Target, and High. These graphs allowed us to gather statistical information and better understand the dynamics of the stocks during this period.

The following Figure.6 shows the statistics of Target, Close, Volume, High attributes from 2017 to 2021 years and Figure.7 shows the range for the year 2021. In the top of the graph we can select the year range that we want and hover through the graph to get the particular year and value.

Second, We wanted to see the relationship between Close price and the year attributes as in Figure.8.

We can see that the year 2021 is having the highest average closing price with up to 3000 range where as the year 2017 has the lowest closing price with up to 2500 range. This stats are useful for the stock analysts while analysing the stock closing prices.

As part of our analysis, we identified the top companies in Japan based on the highest volume of stocks traded. To determine this, we calculated the average volume for the top 200 stocks, allowing us to identify which companies were trading the highest number of stocks. This information provides valuable insights into the market activity and helps us identify the companies with the highest share trading volume in Japan.

From the Figure.5, we can see that the Toshiba Corporation is having the highest shares traded.

C. Models

in this competition, the several different models are considered to find out which approach is more commonly used.These subsection will briefly explain the working concepts of the prediction model that were used to build/test the training and testing sets.

1) Linear Regression: Linear regression is a quiet and the simplest statistical regression method used for predictive analysis in machine learning[**b4**]. Linear regression analysis can be applied to various areas in business and academic study

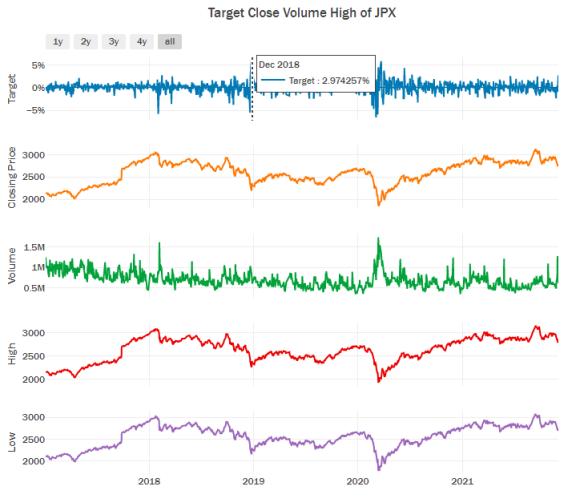


Figure 6. Target, Close, High, Volume of JPX for all years.

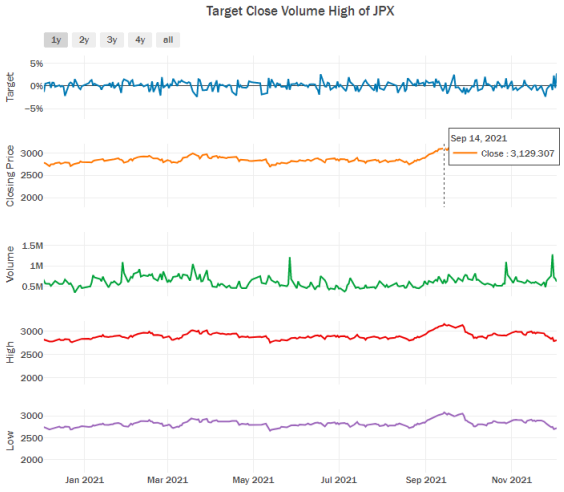


Figure 7. Target, Close, High, Volume of JPX for year 2021.

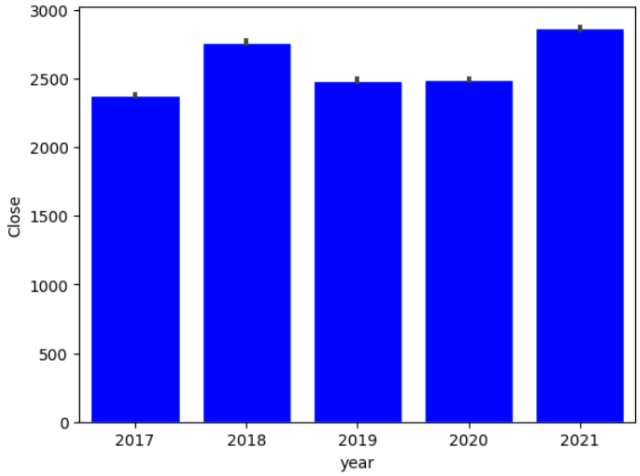


Figure 8. Close Price vs Year.

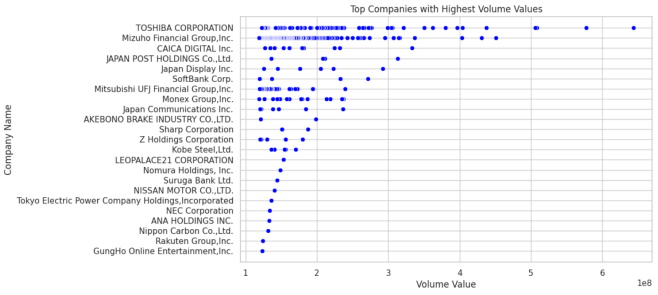


Figure 9. Companies having highest volume.

to predict the value of a variable based on the value of another variable. One of the main reasons for us choosing this model is that it can be trained very quickly and reliably predict the future.

2) Lasso Regressor: Lasso regression is a regularization technique employed in regression methods to enhance prediction accuracy. This approach applies shrinkage, which involves pulling data values towards a central point, typically the mean. The lasso procedure promotes simpler and sparser models by reducing the number of parameters. This type of regression is particularly effective for models with high levels of multicollinearity, or when automating aspects of model selection such as variable selection or parameter elimination.

3) LSTM: LSTM (long short-term memory networks) is used in the field of Deep Learning. It is a variety of recurrent neural networks (RNNs) that are capable of learning long-term dependencies, especially in sequence prediction problems[**b5**].Due to its capability of storing past information, LSTM is very useful in predicting stock prices. This is because the prediction of a future stock price is dependent on the previous prices[**b6**]. our LSTM network architecture is to define a Sequential model which consists of a linear stack of layers, then add a LSTM layer by giving it 50 network units and set the return_sequence to true so that the output of the layer will be another sequence of the same length. We also add a dropout layer and a dense layer with 1 unit as output.In training step, we adopt "adam" optimizer and set the mean square error as loss function.When Training the model by fitting it with the training set. We can try with batch_size of 128 and run the training for 10 epochs.

4) Adaboost: The AdaBoost algorithm, also known as Adaptive Boosting, is an Ensemble Method used in Machine Learning to reduce bias and variance in supervised learning. It assigns higher weights to incorrectly classified instances, hence the term "Adaptive" in its name. AdaBoost grows learners sequentially, where subsequent learners are built upon previously grown learners, converting weak learners into strong ones. This algorithm follows the same principle as boosting, with a slight variation, which we will discuss in detail.

5) LightGBM: LightGBM (Light Gradient Boosting Machine) is a gradient boosting framework based on decision trees to increases the efficiency of the model and reduces memory usage.As the size of data is increasing day by day and it is becoming difficult for traditional data science algorithms to give faster results,Light GBM can handle the large size of data and takes lower memory to run[**b7**]. The reason We choose this algorithm is that our datasets is very large, using this model can save a lot of running time. In this model, the difficulty is to tune the parameters. We chosen the parameters as follow: boosting_type:gbdt, objective: regression, metric: rmse, num_leaves:31, learning_rate:0.05, feature_fraction: 0.9,force_col_wise: True. In training model, we also add paramers: num_boost_round=3000,early_stopping_round=1000 , log_evaluation(period=100).

IV. Evaluations

A. Evaluation metric

Evaluation metrics usually provide a measure of how well the observed outputs are being generated by the models in machine learning. in this competition, we used Root Mean Squared Error(RMSE) to measure the performance of stock exchange prediction in approach 1.

B. Competition Score

In the competition,submissions are evaluated on the Sharpe Ratio of the daily spread returns which we called competition score.In order to calculate the competition score, we have three hypothesis: (1)The returns for a single day treat the 200 highest (e.g. 0 to 199) ranked stocks as purchased and the lowest (e.g. 1999 to 1800) ranked 200 stocks as shorted. (2)The stocks are weighted based on their ranks. (3)The stocks were purchased the next day and sold the day after that. we refered to a python implementation of the metric[**b8**] to calculate competition score. The rate of change of close price $r_{(k,t)}$: it is the rate of change of close price for stock k at business day t.

$$r_{(k,t)} = \frac{C_{(k,t+2)} - C_{(k,t+1)}}{C_{(k,t+1)}}$$

Where $C_{(k,t+1)}$ is the close price of stock k at the following day$t+1$. we used it to rank the stocks and found that Target and rate match by calculating according to the above formula, therefore Target is used to rank the stocks. S_{up}:it is the sum of multiply by their respective rate of change with linear weights of 2-1 for the top 200 stocks predicted.

$$S_{up} = \sum \frac{r_{(k,t)} * linearfunction(2,1)}{Average(linearfunction(2,1))}$$

S_{down}:it is the sum of multiply by their respective rate of change with linear weights of 2-1 for the bottom 200 stocks predicted.

$$S_{down} = \sum \frac{r_{(k,t)} * linearfunction(2,1)}{Average(linearfunction(2,1))}$$

R_{day}: it is the daily spread return.

$$R_{day} = S_{up} - S_{down}$$

Competition Score: The mean/standard deviation of the time series of daily spread returns is used as the score. Score calculation formula (x is the business day of public/private period)

$$Score = \frac{Mean(R_{day_1-day_x})}{STAD(R_{day_1-day_x})}$$

C. MSE

Mean Squared Error (MSE) is a commonly used and straightforward metric, closely related to Mean Absolute Error (MAE) but with a slight modification. MSE involves calculating the squared difference between the actual and predicted values, while MAE involves calculating the absolute difference. In other words, MSE quantifies the squared distance between the actual and predicted values. Squaring the differences is done to prevent the cancellation of negative terms, which is an advantage of MSE over other metrics. We used MSE evaluation metric in approach.2 to measure the model performance.

D. R2

The R2 score is a performance metric that provides an indication of how well a model is performing, but it does not quantify the absolute loss or error of the model. In contrast, metrics such as Mean Absolute Error (MAE) and Mean Squared Error (MSE) depend on the context of the problem, whereas the R2 score is context-independent. The R2 score serves as a baseline for comparing the performance of a model, which is not provided by other metrics. In classification problems, a fixed threshold of 0.5 is often used, similar to how R2 calculates the improvement of a regression line over a mean line. Hence, R2 score is also referred to as the Coefficient of Determination or the Goodness of Fit. We used R2 in approach.2.

V. Experimental Results

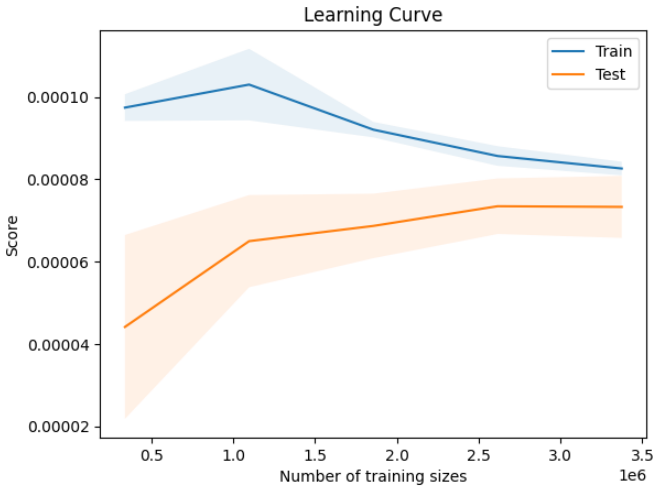
A. Approach.1

According to the methodology that we preciously defined, we used approach 1 to implement machine learning algorithms(Linear Regression, LSTM, LightGBM) on train and test dataset. we obtained different results as shown in the following table:

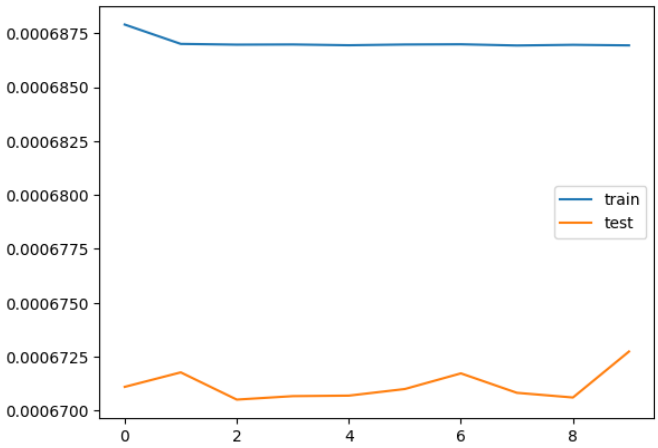
TABLE I
RESULTS OF EVALUATION MEASURES BY USING THREE MODELS IN APPROACH 1

Models	RMSE	Competition Score
Linear Regression	0.026096	0.08333
LSTM	0.025937	0.27103
LightGBM	0.025912	0.13008

The difference in RMSE between these models is very small as shown in the above Table.1. LSTM has the highest score(0.271).When we submitted our models to the kaggle competition, our submissions were estimated and we got very low public/private score while the first place winner received 0.381 score.Maybe this is because this competition is time series. we also used loss function to analyze our models' performance. The following graphs we plotted are shown.

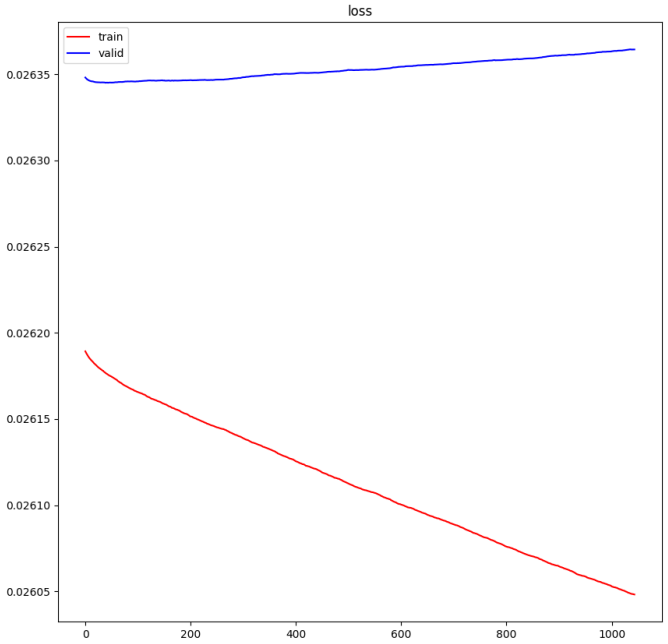


model1. Linear Regression



model2. LSTM

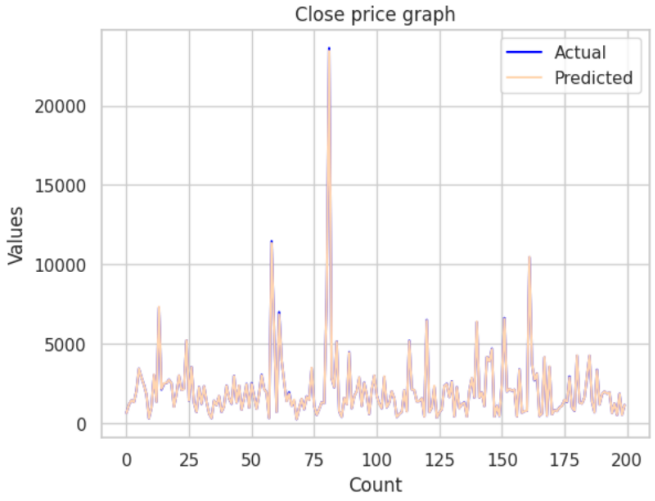
In linear Regression model, we used learning cure to compares the performance of a model on training and testing data over a varying number of training sizes. From the graph, we can see that Training score (green line) decreases and plateau and indicates underfitting High bias, while test score (red line) increases over time and plateau. The Smaller the gap between train score and test score, the better our model generalizes. In LSTM and LightGBM models, we plotted MSE and RMSE loss respectively to examine the performance of our models. The loss in training decline until still and the loss in testing is fluctuation in a very small range in LSTM model,the difference of loss between train and test is very little, just only 0.001%. While in LightGBM model, the loss of training rises and the loss of testing decreases, although it has the shortest running time. Therefore, we think the LSTM model is better than other models.



model3. LightGBM

B. Approach.2

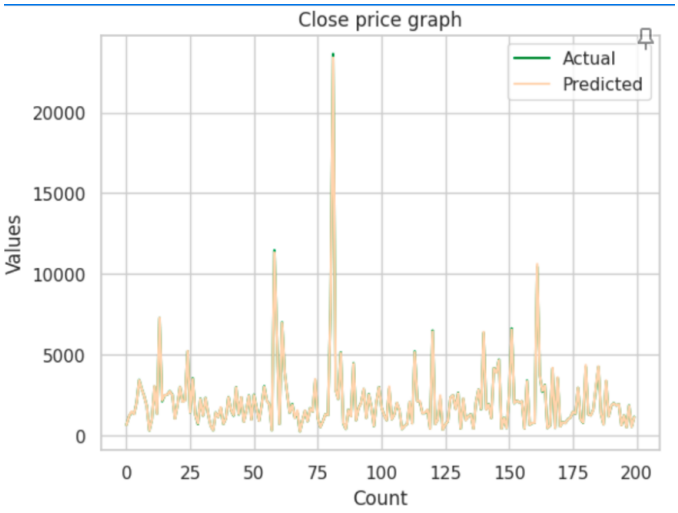
As discussed in the methodology, after the data normalization on the stocks data frame, we used the linear regressor model on our stocks data. We plotted a graph for actual and predicted values of the test data for the model as shown in the model.4. We have plotted this for 200 samples so that we can get a clear view of picture.



model 4. Linear Regressor for actual and predicted values

From the model 4 we can see that the actual and predicted values are equal and are aligning with each other. From this we can conclude that the linear regression is working better for the stocks data frame.

Next, we used Lasso regressor model as seen in the model5.



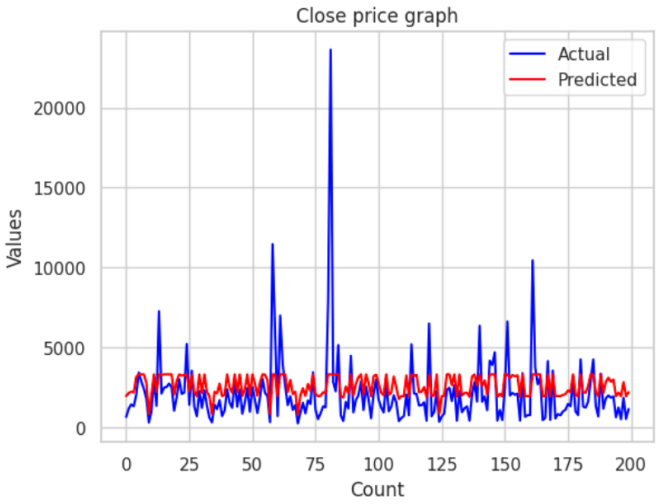
model 5. Lasso Regressor for actual and predicted values

From model 5, we can see that both actual and predicted graph lines are aligning with each other

Third, we used adaboost model and found out that the values are close to each other from model 6.

model 6. Adaboost model for actual and predicted values

Fourth, we used the LSTM model to check if the values are aligning and close.



model 7. LSTM model for actual and predicted values

From the model 7 graph, we can see that the actual and predicted values are not aligning with each other. From this we can infer that the LSTM model is not working for the stocks dataset.

TABLE II
RESULTS OF EVALUATION MEASURES BY USING FOUR MODELS IN APPROACH 2

Models	MSE	R2
Linear Regression	1239.16	0.99
Lasso Regression	2692.27	0.99
LSTM	11233.34	0.13
Adaboost	4784.55	0.99

VI. CONCLUSION AND FUTURE WORK

Our project uses Japanese financial data to educate average investors. 2017–2021 data is included. We use Linear Regression, Lasso Regression, LSTM, Adaboost, and LightGBM to rank 2000 stocks in descending order and build portfolios from them. To manage the enormous dataset and increase investigation, we split the research into two directions. We preprocessed, examined, and displayed the data before computing the daily spread return Sharpe Ratio to evaluate the models. Results and evaluation criteria indicate that models perform differently on traintest data. Because we trained and evaluated LSTM on various data sets in approach 1, it performs better and has lower rmse and high competition score, as seen in the table below.The stocks data set performs better using Method 2's regression models and adaboost (R2 = 0.99).The huge dataset and over 23,000 items cause the astronomical MSE. Our future work will include Our goal is to analyze and research individual company stock moves in respect to their sector IDs. Furthermore, we intend to thoroughly investigate the dataset by importing all available files, including financial, trade, and other pertinent data. Furthermore, we intend to test various regression models in order to improve our analysis.

our team members contributed equally to this project: Sriya Gorrepati worked in data pre-processing and models in approach 2 in this project; Yingying Deng contributed in data featuring and models created by using approach 1 of the project.

REFERENCES

[1] Everything you need to Know about Linear Regression. (n.d.). Retrieved April 20, 2023, from https://web.stanford.edu/class/cs245/readings/aurora.pdf

[2] What is LSTM? Introduction to Long Short Term Memory. (n.d.). Retrieved April 20, 2023, from https://web.stanford.edu/class/cs245/readings/aurora.pdf

[3] Stock Prices Prediction Using Long Short-Term Memory (LSTM) Model in Python. (n.d.). Retrieved April 20, 2023, from https://web.stanford.edu/class/cs245/readings/aurora.pdf

[4] What is LightGBM, How to implement it? How to fine tune the parameters? (n.d.). Retrieved April 20, 2023, from https://web.stanford.edu/class/cs245/readings/aurora.pdf

[5] JPX Competition Metric Definition. (n.d.). Retrieved April 20, 2023, from https://web.stanford.edu/class/cs245/readings/aurora.pdf

[6] Brown, S.J., Kane, E.J. (1978). Risk and return: An equilibrium model. *The Journal of Finance*, 33(1), 1-9.

[7] Campbell, J.Y., Lo, A.W., MacKinlay, A.C. (1997). The econometrics of financial markets. Princeton University Press.

[8] Fama, E.F. (1965). The behavior of stock-market prices. *The Journal of Business*, 38(1), 34-105.

[9] Granger, C.W. (1986). Developments in the study of cointegrated economic variables. *Oxford Bulletin of Economics and Statistics*, 48(3), 213-228.

[10] Huang, X., Shen, S., Zhou, J. (2018). Stock market prediction with recurrent neural network. *IEEE Transactions on Industrial Informatics*, 14(8), 3529-3537.

[11] Lai, K.K., Wang, Y., Huang, J., Liu, J. (2015). Stock market forecasting using machine learning algorithms. In *2015 IEEE/ACIS 16th International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD)* (pp. 154-159). IEEE.

[12] Lo, A.W., MacKinlay, A.C. (1999). A non-random walk down Wall Street. Princeton University Press.

[13] Mallqui, O.P., Larriva-Novo, O.A. (2019). Deep learning for stock market prediction: A comparative study. *Expert Systems with Applications*, 116, 243-257.

[14] Tsantekidis, A., Passalis, N., Tefas, A., Kanniainen, J. (2017). Using deep learning to detect price change indications in financial markets. *Journal of Financial Engineering*, 4(2), 1750013.

[15] Zhang, G.P., Patuwo, B.E., Hu, M.Y. (1998). Forecasting with artificial neural networks: The state of the art. *International Journal of Forecasting*, 14(1), 35-62.