Comparison of Regressions and Multi-feature LSTM in Stock Price Prediction for Top Car Companies

Jiale Li[1,a,*]

[1]Department of Economic and Management Department, Beijing Institute of Technology, Beijing, China
a. alisaskyle@mail.sdufe.edu.cn
**corresponding author*

Abstract: Machine learning (ML) has gradually permeated in quantitative finance, among which Regressions and Deep Learning are two well-known ML domains. However, as traditional prediction methods, regressions have shown their limitation in more complicated datasets with temporal sequence and non-linear relationships between features and responses. In order to compare the capability of widely-used deep learning methods SVM and LSTM with that of the methods in regressions, this study acquired historical data of 10 car companies' stocks from Kaggle.com to simulate the share price in the next period based on four different regression methods and two Deep Learning methods, respectively. Moreover, the author uses RMSE and MAPE as evaluation metrics to compare those model's training and testing sets fitness. According to the result, LSTM has better fitness than other regression models in predicting time series data. Although share price from Kaggle might not reflect the overall market condition, this study would contribute to mend the research gap of empirical application of different machine learning methods.

Keywords: Regressions, LSTM, SVM, quantitative finance, empirical analysis

1. Introduction

Stock market investment has continuously been seen as an effective way in financial world. Before investing a stock, there are two types of analysis need to be carried out: fundamental analysis and technical analysis [1]. The former requires investors to take a comprehensive view of the target companies from reports and evaluate the actual value of stocks, including comparisons of the intrinsic value, using multiples and metrics as indicators such as P/E ratio and price [2], and the current market to evaluate whether the price is over or under-valued. Moreover, qualitative factors such as business model, corporate governance and industrial competitive advantage should be considered as well. Technical analysis, on the other hand, takes a further step in analyzing the historical data by detecting trends or patterns. Stock prediction is one of the technical analysis that is seen as a critical method in analyzing current market trend and target companies' development. Moreover, it reflects national current economic situation. For institutional and retail investors, stock prediction is also an important basis for profit earning and risk avoidance [3].

Nevertheless, uncertainties and fluctuations in stock price caused by numerous factors hinders the decision-making for investors. In primary phase, simple algorithms such as classical regression are used in prediction, along with linear algorithms such as Autoregressive Model (AR) and Movingaverage Model (MA) [4]. However, linear regression could not fully take all the information at each time steps into consideration, and the high noises and uncertainty reduce the accuracy of results, which makes it difficult to predict stock price over longer period [5]. Enhanced technical analysis methods have been developed such as Lasso, Regression and Polynomial Regression to enhance model's interpretability by improving fitting effect and increasing variables [6]. Furthermore, compared with classical linear regression, algorithms based on machine learning could extract effective information without setting assumptions in advance. Besides, stock price in reality often possess non-linear and non-stationary characteristics [7], which is too complex for linear regression to capture for enhancing model fitness. However, machine learning algorithms such as decision trees, SVM and Naive Bayes have higher learning ability and shows advantages in processing non-linear data, thus is used widely in research fields these days [8]. Among them, the SVM model is often used for classification to take a research approach to determine the direction of the stock price trend [1], and decision trees and Naive Bayes could both used for quantitative and qualitative research.

In addition, recent deep learning revolution improve the ability of machine learning in time series prediction. Neural Network (NN) has been popular in multi-time step prediction, and different neural network architectures have different strengths and weaknesses [9]. For models used in prediction, Artificial Neural Network (ANN) as a feed-forward NN is one of the most widely used models which possess a group of multiple neurons at each layers. Recurrent Neural Networks (RNNs) such as LSTM are well known for modelling temporal sequences compared to ANN [10], which save the learning process in memory cells and feed the output back to the model for optimization. The iteration learning steps through backpropagation help the machine learn to predict the outcome of each layers, which is more complex but more accurate in time series prediction.

In practice, when choosing features for prediction, price interactions between industrial stocks are amplified by increased competitiveness and numbers of listed companies. It is evident that predicting the stock price solely based on patterns of historical data is limited. Thus, the author uses different car companies' price patterns to predict that of the target company. Moreover, comparisons between four regression models (Linear Regression, Lasso Regression, Ridge Regression and Polynomial Regression) and deep learning model (LSTM) are processed. The article first briefly introduces those six machine learning techniques used in time series prediction and relevant evaluation metrics, and then uses data from Kaggle.com to carry out feature engineering, process the models and display comparisons using Python. Explanations and limitations are furtherly discussed.

2. Data and Method

2.1. Data

The data selected from this article is from Kaggle.com, which includes 10 top international car companies' stock: Audi, BMW, Honda, Lucid Motors, NIO, Nissian, Rolls Royces, Tata, Tesla, and Volkswagen. The sample size is over 30,000 and the dataset is picked from August 23, 2019 to August 20, 2021. The author uses TimeSeriesSplit to get the index of training and testing sets, which include 318 and 158 columns respectively.

2.2. Models and Parameters

Linear Regression is a commonly-used model for exploring linear relationship between features and dependent variable. After data preprocessing, the variables with most correlation is selected and the parameters as well as assumed linear function are set:

$$\omega = \{\omega_1, \omega_2, \omega_3, \cdots, \omega_n\} \tag{1}$$

$$\mathrm{h}(\omega,\mathrm{x}) \;=\; \omega_1 \cdot \mathrm{x}_1 + \omega_2 \cdot \mathrm{x}_2 + \omega_3 \cdot \mathrm{x}_3 + \cdots + \omega_\mathrm{n} \cdot \mathrm{x}_\mathrm{n} \tag{2}$$

Linear Regression could be used for primary prediction by introduce Least squared Method. Further improvements in preventing over-fitting effect could be realised by means of Ridge Regression, Lasso Regression and Polynomial Regression. Using Least Squared Method, the cost function is determined for each x_i in sample $(\mathrm{x_i}, \mathrm{y_i})$:

$$\mathrm{J}(\omega_1, \omega_2, \cdots, \omega_\mathrm{n}) = \frac{1}{2\mathrm{m}} \sum_{\mathrm{i}=1}^{\mathrm{m}} (\mathrm{h}_{\omega_\mathrm{n}}(\mathrm{x})^{(\mathrm{i})} - \mathrm{y}^{(\mathrm{i})})^2 \tag{3}$$

The gradient descent is then made by calculating the derivatives based on each parameter ω of the Bell-shaped cost function $\mathrm{J}(\omega_1, \omega_2, \cdots \omega_\mathrm{n})$, and return the value of each ω when all the derivatives are zero. Thus, the basic linear regression function is determined. Lasso and Ridge Regression both add penalty λ to the features used for independent variables for improvement of over-fitting problem of linear regression in the training set [11]. This process is called regularization, and the effect becomes more significant when the λ gets bigger. Both these Regression could be used in the regularization process of Linear, Logistic and Polynomial Regression. In Lasso Regression, the penalty function is:

$$\sum_{\mathrm{j}=1}^{\mathrm{n}} \lambda_\mathrm{j} \cdot |\omega_\mathrm{j}| \tag{4}$$

The L_2 penalty term requires the multiplied parameter after the penalty in a form of absolute value [12]. In Ridge Regression, the penalty function is:

$$\sum_{j=1}^{n} \lambda_j \cdot {\omega_j}^2 \tag{5}$$

Ridge Regression adds an L_1 penalty term to the parameter, which means that the penalty term is in a form of quadratic term, thus the slope ω_j's effect would be small when λ_j is large. However, in Ridge Regression, L_1 penalty could only asymptotically reduce the weight of the parameter to zero, but in Lasso Regression, the L_2 penalty term could directly reduce the slope to zero, which is more suitable than Ridge when there are more parameters in the equation. Thus, Lasso Regression could exclude unnecessary variables better in practice.

Polynomial Regression is often used to reflect the non-linear relationship between features and the response. The model replaces standard linear model

$$y_i = \beta_0 + \beta_1 x_i \tag{6}$$

with a polynomial function

$$y_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \beta_3 x_i^3 + \cdots + \beta_j x_i^j \tag{7}$$

For large degree of j, the polynomial function could produce an extremely powerful non-linear curve. For better enhancement, methods such as Piecewise Polynomials could be used to separate the range of X based on the index c:

$$y_i = \begin{cases} \beta_{01} + \beta_{11} x_i + \beta_{21} x_i^2 + \beta_{31} x_i^3 + \cdots + \beta_{j1} x_i^j, if\ x_i < c_k \\ \beta_{02} + \beta_{12} x_i + \beta_{22} x_i^2 + \beta_{32} x_i^3 + \cdots + \beta_{j2} x_i^j, if\ x_i \geq c_k. \end{cases} \tag{8}$$

More pieces of range could improve the fitting effect, and the spline of the regression would need further smoothing as the knots c_k that separate each range would have low consistency.

SVM is a generalization of a simple and intuitive classifier called the Maximal Margin Classifier [13], which is an optimal hyperplane used for linearly separating two classes. The support vectors are points that support the selection of the optimal hyperplane, which could be presented in the formula

$$\omega^{\mathrm{T}} \cdot x + b = 0 \tag{9}$$

To solve a linearly inseparable problem, SVM could allow the least number of points be misclassified, which is similar to the Least Squared Method. As for non-linear classification problem, if the attributes of features couldn't satisfy the classification requirements, then there must be a high-dimensional space that makes the sample linearly separable. Thus, using kernel function, SVM maps two-dimensional points to three-dimensional space, to find a plane to handle problems which could not be solved in two-dimensional space. Commonly-used kernel functions are linear kernel, Polynomial kernel, Radial Basis Function Kernel, Gaussian Kernel, and sigmoid kernel. In multi-classification problem with $\mathrm{k} > 2$ responses, SVM would first map input data into a dataset with $\mathrm{k} - 1$ responses and one randomly selected response. Several binary classifications would be made to find out the multi optimal classifiers. SVM could also use for regression problem. By inputting new data into the SVM model, the data would be classified based on the hyperplane learned during the training phase [14].

Supervised Long Short-Term Memory (LSTM) RNNs is a widely-used deep learning method in predicting time-series data [10]. In Deep Learning, although CNN could optimize the model by increasing the number of hidden layers, it does not consider the temporal sequential change of a single hidden layer. RNN proposed by Jordan focuses on the time sequence in all neurons, and is a structure that iteratively learns the sequence data. To solve the gradient explosion and elimination in RNN, LSTM is introduced by adding input gate, forgot gate, output gate and inner memory units to control the learning process of features' attributes. However, in real practice, LSTM has high complexity and overfitting problem. Moreover, researchers need to adjust its hyperparameters iteratively to acquire optimal results. Recently, experts have effectively improved the LSTM based on stock prediction. Zeng and Nie proposed a bidirectional LSTM stock prediction model, which effectively integrates Dropout to further optimize the fitness of the model [15]. The experimental results show that this model can greatly reduce errors. Shi et al. proposed the DMD-LSTM model to predict stock prices [16]. Using the stock features extracted by DMD algorithm as input to LSTM could more accurately describe the changes in stock prices. Li has further proposed a deep neural network DP-LSTM based on LSTM technology through relevant exploration [17], which includes using news articles as hidden messages, and then utilizing differential privacy mechanism to effectively aggregate and utilize various news sources to predict the S&P 500 stock price. Moreover, Sun, Zhou and Pan have established single feature and multi feature LSTM models to predict stock prices separately, and combine XGBoost tool to achieve further prediction [18]. Compared with NN models, LSTM is the second-highest approach to be used [1]. However, LSTM's use only started in 2015 as the use of SVM is diminishing. Thus, the comparison between two methods in practice is required.

2.3. Evaluation Metrics

The evaluation metric used for comparison between different models is RMSE (Root Mean Squared Error) and MAPE (Mean Absolute Percentage Error). The MAE (Mean Absolute Error) is the simplest regression error metric whose formula is

$$\mathrm{MAE} = \textstyle\sum_{\mathrm{i}=1}^{\mathrm{n}} |\mathrm{residual}_{\mathrm{i}}| \tag{10}$$

The residual is the absolute difference between sample's responses' true value and the predicted value. As MAE calculates the sum of the residuals' absolute value, it is robust to the outliers. In MSE (Mean Squared Error), the sum of the error grows quadratically:

$$\mathrm{MSE} = \sum_{i=1}^{n}(\mathrm{residual}_i)^2 \tag{11}$$

RMSE is the root of the MSE:

$$\mathrm{RMSE} = \sqrt{\sum_{i=1}^{n}(\mathrm{residual}_i)^2} \tag{12}$$

Those two models are commonly used in judging the model's effectiveness. The higher the MSE, the lower the fitness of the model in training and testing sets. In this empirical analysis, MSE is larger than MAE, and the difference between each models' fitness could be enlarged for better analysis. Furthermore, TSLA's highest stock price is 880.00, while the lowest is 35.79, and the fluctuation from 2020 to 2021 is high, thus MSE with quadratically growing error would not be appropriate for metric. As the root of the MSE, RMSE is smaller and more reliable in comparing the error in stock price. However, as those two metrics is not in the form of absolute difference, they are not robust to the outliers. Moreover, the metric MAPE is the percentage equivalent of MAE. The equation looks just like that of MAE but with adjustments to convert everything into percentages, which measures how far the model's predictions are off from their corresponding outputs on average. The lower the value for MAPE, the better the machine learning model is at predicting values. In addition, MAPE is robust to the outliers, which could compensate for the outliers' effects on RMSE.

3. Results and Discussion

3.1. Feature Engineering

The six top car companies' stock price information are firstly extracted into a combined dataframe based on their source. A further re-structure process is made to the data include cleaning the data with missing values and disposing source columns "Lucid" and "NIO" which have only 234 and 742 non-null features respectively. The processed data structure is (13, 1231). The visualized close price data from 2016 to 2021 of those six companies is shown in Fig. 1. Afterwards, the logdifference of other six car companies except Tesla after 2019 is calculated. As logdifference is one of the standardised tools for the data, the logdifferences of each company approximately follow Gaussian Distribution. For linear regression model, the author first tests the correlation coefficient of each feature (seen in Fig. 2). The heatmap above depicts that the correlation between logdifference of Volkswagen and BMW is high, thus for variable selection of linear regression, it is necessary to remove at least one of them.

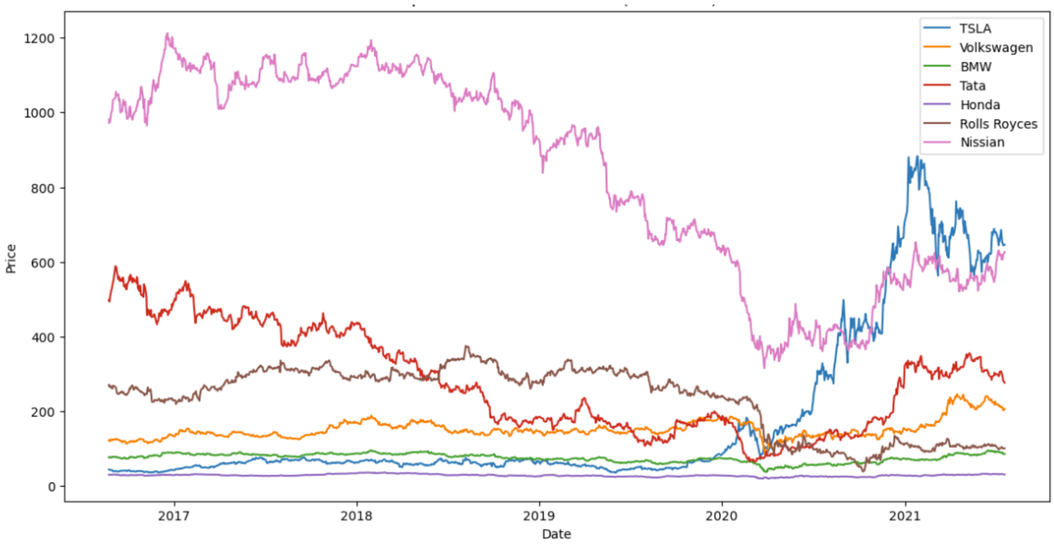
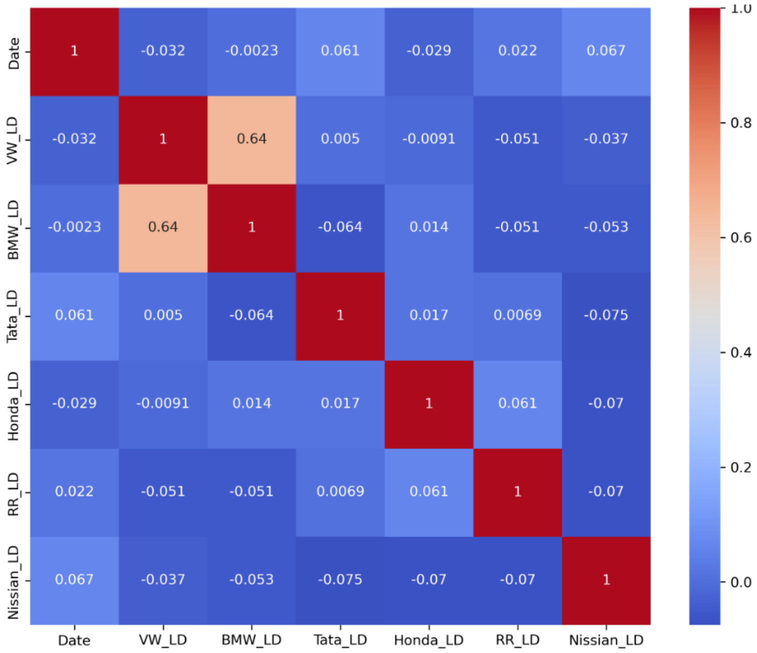


Figure 1: Top 7 Car Companies Stock Price from 2019 to 2021 (Photo/Picture credit: Original).

The heatmap above depicts that the correlation between logdifference of Volkswagen and BMW is high, thus for variable selection of linear regression, it is necessary to remove at least one of them. For linear regression,the author further carries out hypothesis tests. The hypothesis test input BMW_LD and VW_LD separately as those two variables have high correlation coefficient. According to the analysis, BMW_LD, VW_LD, Tata_LD and Nissian_LD's P-value are high ($P_{BMW\_LD} = 0.62, P_{VW\_LD} = 0.31, P_{Tata\_LD} = 0.62, P_{Nissian\_LD} = 0.48$), thus the predictors used in Linear Regression, Lasso Regression, Ridge Regression and Polynomial Regression are Honda_LD and RR_lD. For variable selection in SVM and LSTM Neural Network, the author uses SelectKBest in sklearn library to select features for SVM, and shap library is used to perform variable importance check for LSTM (seen from Fig. 3). The labelled numbers above corresponds to each feature's column numbers in the processed dataframe after iterative training of these two models. Feature 11, 17, 5, 2, 8 and 23 represents RR and BMW, which is further used as variables in LSTM.



	Date	VW_LD	BMW_LD	Tata_LD	Honda_LD	RR_LD	Nissian_LD
Date	1	-0.032	-0.0023	0.061	-0.029	0.022	0.067
VW_LD	-0.032	1	0.64	0.005	-0.0091	-0.051	-0.037
BMW_LD	-0.0023	0.64	1	-0.064	0.014	-0.051	-0.053
Tata_LD	0.061	0.005	-0.064	1	0.017	0.0069	-0.075
Honda_LD	-0.029	-0.0091	0.014	0.017	1	0.061	-0.07
RR_LD	0.022	-0.051	-0.051	0.0069	0.061	1	-0.07
Nissian_LD	0.067	-0.037	-0.053	-0.075	-0.07	-0.07	1

Figure 2: Heatmap of Features (Photo/Picture credit: Original).

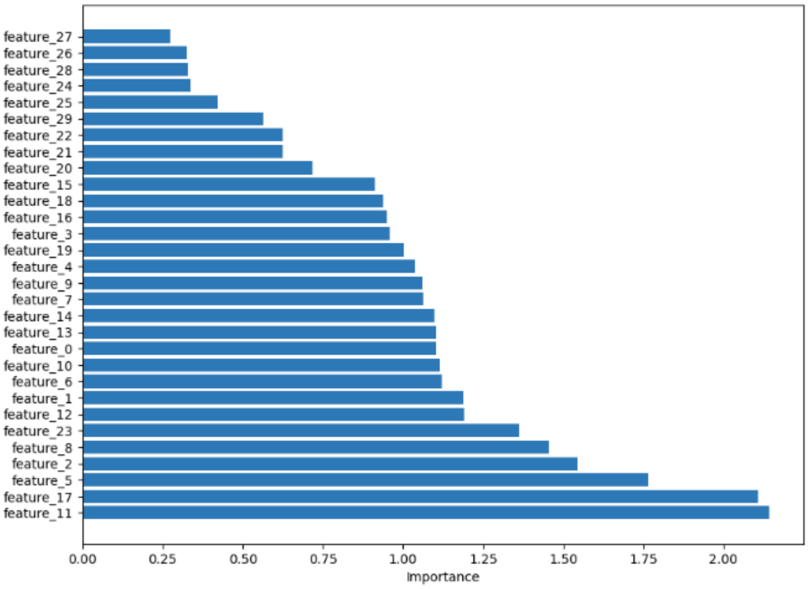


Figure 3: Feature importance (Photo/Picture credit: Original).

3.2. Linear Model

The author uses TimeSeriesSplit in sklearn library to split the data into training and testing set. The training set 318 columns and the testing set have 158 columns. After the three-fold cross validation, the training and validation loss are plotted (as shown in Fig. 4). The fluctuation of validation loss is much more significant than that of the training loss, indicating that the fitting effect might need to be further improved. The outcome of training and testing set compared with actual price are plotted in Fig. 5. The high fitness for training set and low interpretation of testing set imply that the linear regression is over-fitted. Consequently, Lasso and Ridge Regression is further used for comparison.

Lasso and Ridge Regression's results are similar, and in order to compare their consequences with traditional Linear Regression Model, the author put them into a same figure for training and testing sets comparison in Fig. 6. The results shows that the regularization effect of Lasso Regression is small. Thus, there might be underlying non-linear connections between the figure and responses, which might require further analysis.

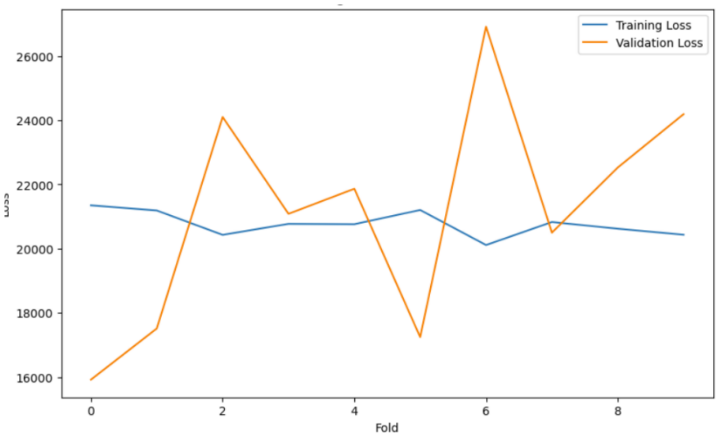


Figure 4: Training and Validation Loss for Linear Regression (Photo/Picture credit: Original).

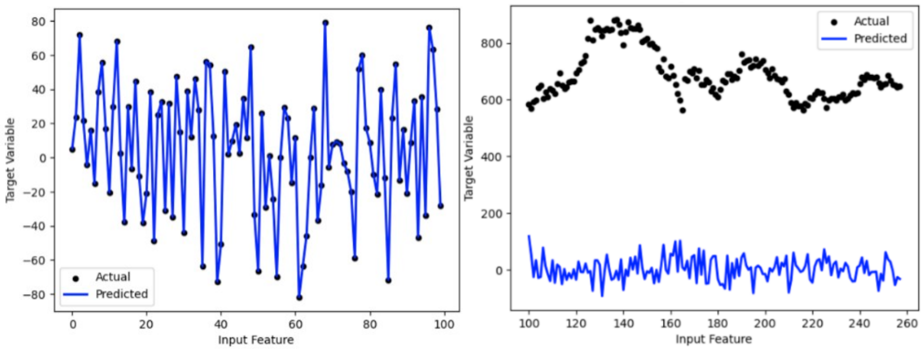


Figure 5: Training and testing fitness of Linear Regression (Photo/Picture credit: Original).

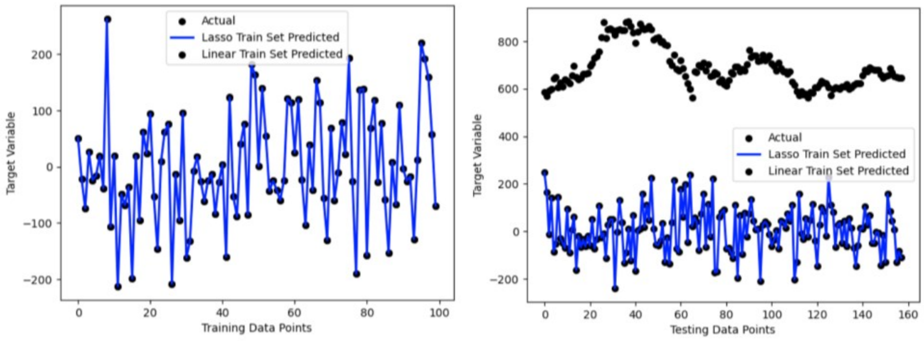


Figure 6: Training (left) and testing (right) fitness of Linear and Lasso Regression (Photo/Picture credit: Original).

3.3. Polynomial Regression

Allowing degree to be 2 and the interaction term between those two variables to exist, the training and validation loss for Polynomial Regression is given in Fig. 7. The fitness of all Linear Regression Models are lower, thus there are non-linear relationship between the predictors and responses that is hard for linear regression to capture.

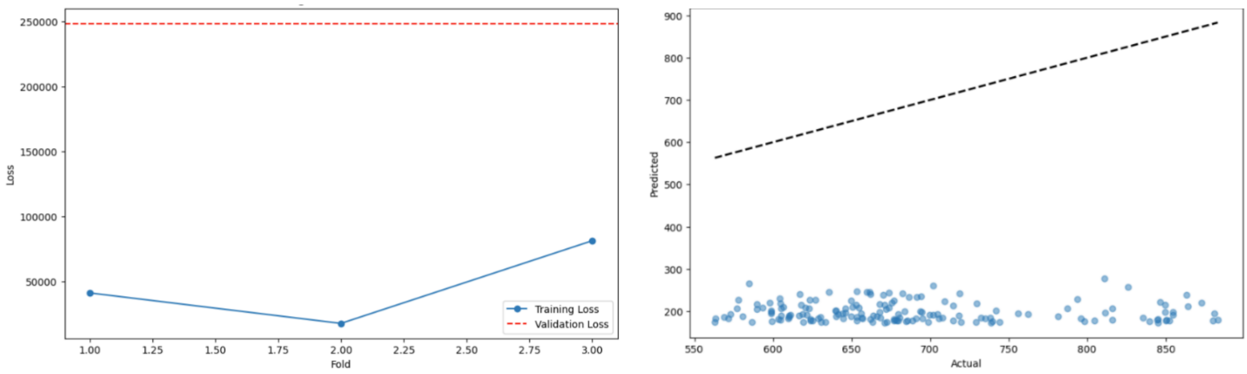


Figure 7: Training (left) and testing (right) and Validation Loss of Polynomial Regression (Photo/Picture credit: Original).

3.4. SVM Model

After the iterative test for parameters, the hyperparameter C is 100, and the eplison is 0.001, and non-linear radial basis function kernel is used for classification. The fitness effect for SVM is not good, because of the small sample size and numbers of features. Moreover, SVM could not iteratively learning time-series data as LSTM Model (seen from Fig. 8 and Fig. 9).

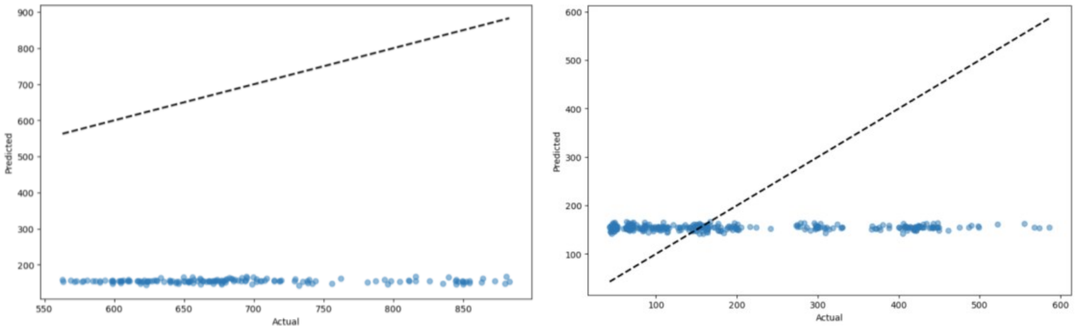


Figure 8: Training and Testing Fitness for SVM (Photo/Picture credit: Original).

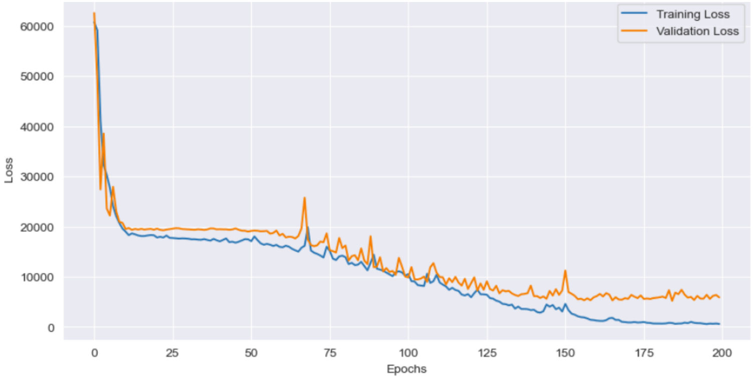


Figure 9: Training and Validation Loss for LSTM (Photo/Picture credit: Original).

3.5. LSTM Model

After iterative testing for hyperparameters, the LSTM Model reaches its optimal fitness with 5 timesteps, 200 epochs and learning rate at 0.005 in the start (Fig. 10 and Fig. 11). The explanatory effect of predictions is not ideal, but it could be seen from the figures that LSTM Neural Network's fitness is much better than other machine learning methods.

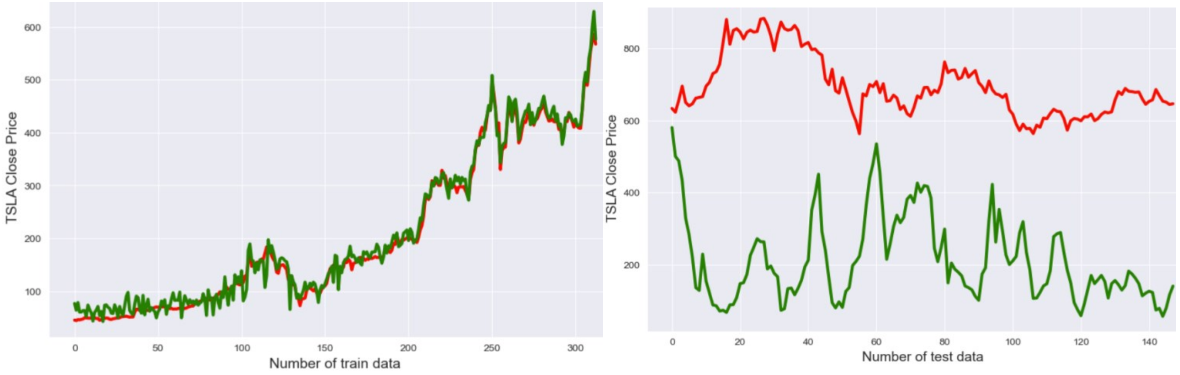


Figure 10: Training and Testing Fitness for LSTM (Photo/Picture credit: Original).

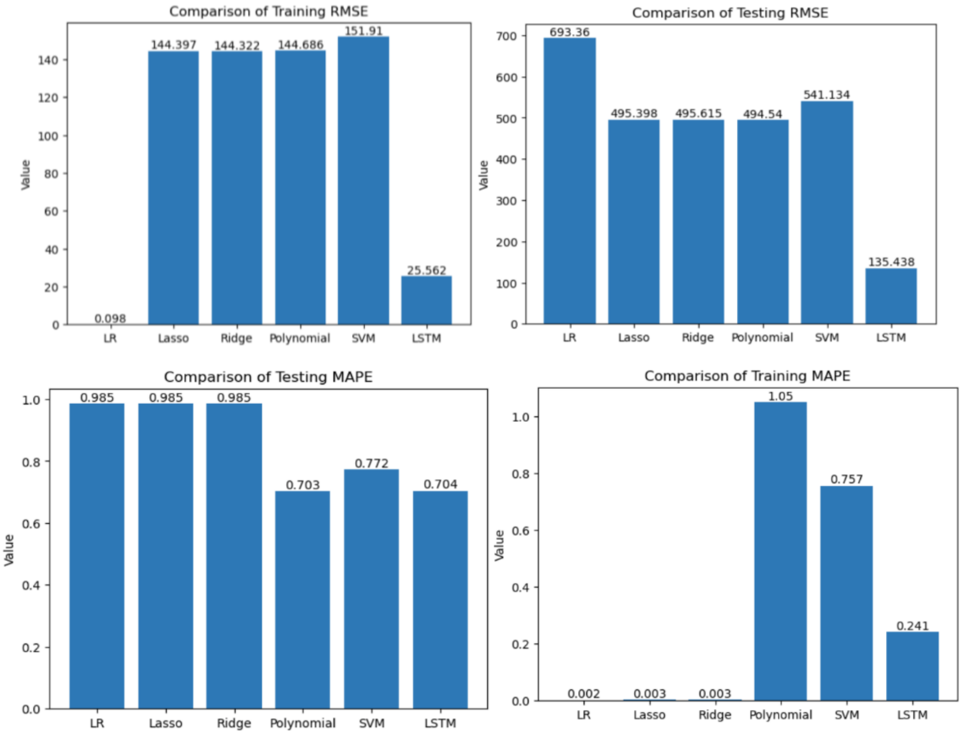


Figure 11: Comparison of Training and Testing RMSE and MAPE (Photo/Picture credit: Original).

3.6. Comparison and Explanation

This study uses RMSE (Root Mean Squared Error) and MAPE (Mean Absolute Percentage Error) to compare the effectiveness of different models in predicting TSLA's price (seen from Fig. 11). The results shows that there is a tendency for regression models to have lower RMSE and MAPE in training set, but higher RMSE and MAPE in testing set. Moreover, the model with lowest fitness is Polynomial Regression, as the training RMSE is high. LSTM performs well as both training and testing RMSE is low compared to other models. Small testing sets, outliers and short time for learning might cause its high testing MAPE.

4. Limitations

The study aims to utilize the linkage between different stocks' daily yield to compare six different machine learning techniques. The results of those models' practices present certain academic significance, however, limitations also exist in sampling and variable selection phase. The selection of the features in LSTM model does not consider the amount of impact between them. Moreover, the lack of fitness in Polynomial and SVM might cause by overly simplistic variable selection. In further stock price prediction research, it is necessary to include more influentials, such as US tax, domestic inflation rate, time value of money e^{-rt}, customer price index (CPI).

5. Conclusion

To sum up, this study investigates the fitness of different machine learning models in predicting time series data. The results reveal that LSTM Neural Network as a Deep Learning Method, could capture non-linear relationship between features and learn the temporal sequential data over time, which could predict much more precisely than SVM and other regression models. However, LSTM is more complex and might cause overfitting problem, the hyperparameter and time series data also influence the effectiveness of the model. Thus, in practice, the decision-making based on machine learning should consider different models' capabilities.

References

[1] Mintaryaa, L.N., Halima, J.N.M., Angiea, C., Achmada, S., Kurniawan, A. (2023) Machine Learning Approaches in Stock Market Prediction: A Systematic Literature Review. Procedia Computer Science, 216, 96-102.

[2] Patel, J., Shah, S., Thakkar, P., Kotecha, K. (2015) Predicting Stock and Stock Price Index Movement Using Trend Deterministic Data Preparation and Machine Learning Techniques. Expert Systems with Applications, 42(1), 259-68.

[3] Zhang, Q., Lin, T., Qi, X., Zhao, X. (2020) A Review of Stock Prediction Research Based on Machine Learning. Journal of Hebei Academy of Sciences, 37(04), 15-21.

[4] Han, J., Lu, S., Sun, N., Shang, J., Zhang Y., Bao Z. (2023) Application Research of Deep Learning in Stock Prediction Information. Technology and Informatization, 9, 190-193.

[5] Ren, C., Song, C. (2022) Research on Financial Data Prediction Based on Bidirectional Recurrent Neural Networks Network Security Technology and Applications, 4, 53-55.

[6] Yang, G. (2022). Comparative Analysis of Linear Regression Methods for Predicting Five Types of Stocks Based on Stock Correlation. Modern Business, 29, 42-45.

[7] Tang, P., Tang, C., and Wang, K.R. (2024) Stock Movement Prediction: A Multi-input LSTM Approach. Journal of Forecasting, 10, 3071.

[8] Xie, Q., Cheng, G., Xu, X. (2019) Research on Stock Prediction Models Based on Neural Network Ensemble Learning. Computer Engineering and Applications, 55(08), 238-243.

[9] Chandra, R., Goyal, S., Gupta, R. (2021) Evaluation of Deep Learning Models for Multi-Step Ahead Time Series Prediction. IEEE Access, 9, 83105-83123.

[10] Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. Neural Network, 61, 85-117.

[11] Peña, D., Tsay, R. (2021) Statistical Learning for Big Dependent Data. John Wiley & Sons, Hoboken, NJ.

[12] Pontines, V., Rummel, O. (2023) LIBOR Meets Machine Learning: A Lasso Regression Approach to Detecting Data Irregularities. Finance Research Letters, 55, 103852.

[13] James, G., Witten, D., Hastie, T., Tibshirani, R., Taylor, J. (2013) An Introduction to Statistical Learning: with Applications in Python. Springer Texts in Statistics. New York, NY: Springer.

[14] Liagkouras, K., Metaxiotis, K. (2020) Stock Market Forecasting by Using Support Vector Machines. In: Tsihrintzis, G., Jain, L. (eds) Machine Learning Paradigms. Learning and Analytics in Intelligent Systems, 18, Springer, Cham.

[15] Zeng, A., Nie, W. (2019) Stock Recommendation System Based on Deep Bidirectional LSTM. Computer Science, 10, 84-89.

[16] Shi, J., Zou, J., Zhang, J., Wang, C., Wei, Z. (2020) Research on Stock Price Time Series Prediction Based on DMD-LSTM Model. Computer Application Research (03), 662-666.

[17] Li, X., Li, Y., Yang, H., Yang, H., Liu, X. (2019) DP-LSTM: Differential Privacy-inspired LSTM for Stock Prediction Using Financial News. Retrieved from https://arxiv.org/abs/1912.10806.

[18] Sun, N., Zhou, S., Pan, Z. (2023) Research on Multi Feature Stock Price Prediction Based on XGBoost LSTM Model Mathematical Modeling and Its Applications, 4, 32-39.