COS30018 – Intelligent System

Option B: Stock Price Prediction

Report v0.6 extension B.7

Name Phong Tran
Student Id:104334842

Class: 1-5
Tutor: Tuan Dung Lai

Table of Contents

Research on potential approaches for predicting companies' stock prices/trends

I. Introduction

This study demonstrates briefly on how I attempted the research on the original data for stock price to predict based on the best method after conducting different literature reviews. This will shift from the traditional methods into the new method integrated with broader data and Machine Learning models. From the literature review, we will focus on the technologies which improve the prediction accuracy and solve the limitations in the time-series forecasting. Furthermore, there will be more insights into a diverse data and model for the analysis of stock trends.

II. Objective

This will mainly focus on the work of applying techniques on exploring and exploiting the stock data in order to improve the accuracy and the reliability of the stock without relying on the traditional methods. This will cooperate with machine learning techniques and diverse data sources, with more information and reliability based on different integrations.

III. Literature Review

3.1 Sentiment analysis in stock prediction

With the analysis, the data from news will be used to extract and quantify sentiment in order to reflect reality in the market. Latent Dirichlet Allocation (LDA) helps to find the topic at the news, which reflects on type news and its relevance on stock trends (Hao et al. 2021). For example, the connection between the news on positive or negative sentiment with greater or lesser impact on stock performance.

3.2 Textual modelling and textual representation

It has been widely used in applying Latent Dirichlet Allocation (LDA) technique for topic modelling in stock prediction with the features of basic topic in news articles, enhancing the model to distinguish among the context of articles (Hao et al. 2021). For instance, the applications of LDA attempts on the topics in news articles, with relations with stock trends, improve the insights of predictions with the key things in the news context. The topics can influence significantly due to the news topic. For example, even the positive sentiment may lead to stock trends drop because of several things, such as hidden costs or controversy surrounding the topic, therefore, highlighting the relevance of the topic with the sentiment scores.

3.3 Fuzzy Set Theory and twin SVM for dealing with uncertain data

This theory improves the model by giving the different categories of uncertain data, such as vague news or volatility price movements (Hao et al. 2021). Therefore, this will allow it to address outliers effectively, improving its performance to the best at the stock market data.

3.4 Particle Swarm Optimisation in Feature Selection to search hidden topics could affect stock prices rise or fall

In the text prediction, it is critical at feature selection where there is a massive amount of unrelated or redundant data. With the PSO (Hao et al. 2021), it helps reduce the high-dimensional features, like text mining for stock prediction. Therefore, this will reduce overfitting by selecting on the most impact features. The high-dimensional features like historial price, sentiment score, and topic vectors, PSO impacts on balancing the work of exploring and exploiting, which helps the model to find on the best features only concentrate on capturing the trends of the stock price.

3.5 Moving Average

This technique is used to perform the statistics applied in time-series to free the fluctuation obstructions in short-term and demonstrate the underlying stock trend (Yoon, Lee & Kim 2000). It is used to calculate the average data points in the specific time-window, which makes the new available data.

3.6 Wavelet Transform

Discrete Wavelet Transform (DWT) using the approach to analyse the signal at multiresolution analysis (MRA) techniques, therefore, good or bad time resolution will get their opposite status frequency resolution (Langi, Pitara & Kyspriyanto 2012). Using the filters high-pass and low-pass filtering into detail and approximation signal components, this will reveal underlying noise or trends within the movements of stock.

IV. Limitations

- Depend on the quality and quality of external dataset available for sentiment analysis, which might affect due to problems or crisis of other categories which might impact on stock market.
- Not account on other sentiment sources, such as social media, which reduce the diversity of information that could impact on stock price.

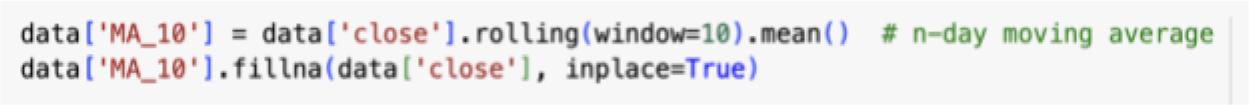
- Large dataset at stock price with requirements on high processing power may make the wavelet analysis difficult in stock trend display.
- Unpredictability in major events and news reports could increase the volatility and noise signals, which can lead to overfitting and reducing stable models.
- Depending on the hyperparameters, this may impact on the model's performance. Like lack or improper tuning may decrease the performance model, which leads to poor prediction.

V. Advantage

- Sentiment analysis on text mining at social media, which provides us with insights from time to time and makes better decisions on short-term trading.
- Wavelets transform decompose data into different dimensions to filter out noise, therefore, revealing underlying trends. Moreover, the provision of time and frequency, this ideal on both short-term and long-term trends in stock price markets.
- Particle Swarm Optimisation helps to optimise the feature selection in order to reduce dimensional data, therefore, more display on relevant predictors to improve accuracy and efficiency.

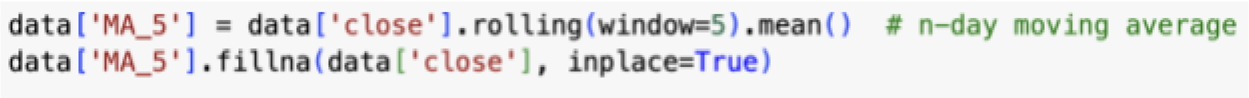
VI. Implementation on the idea and evaluation result

Moving Average



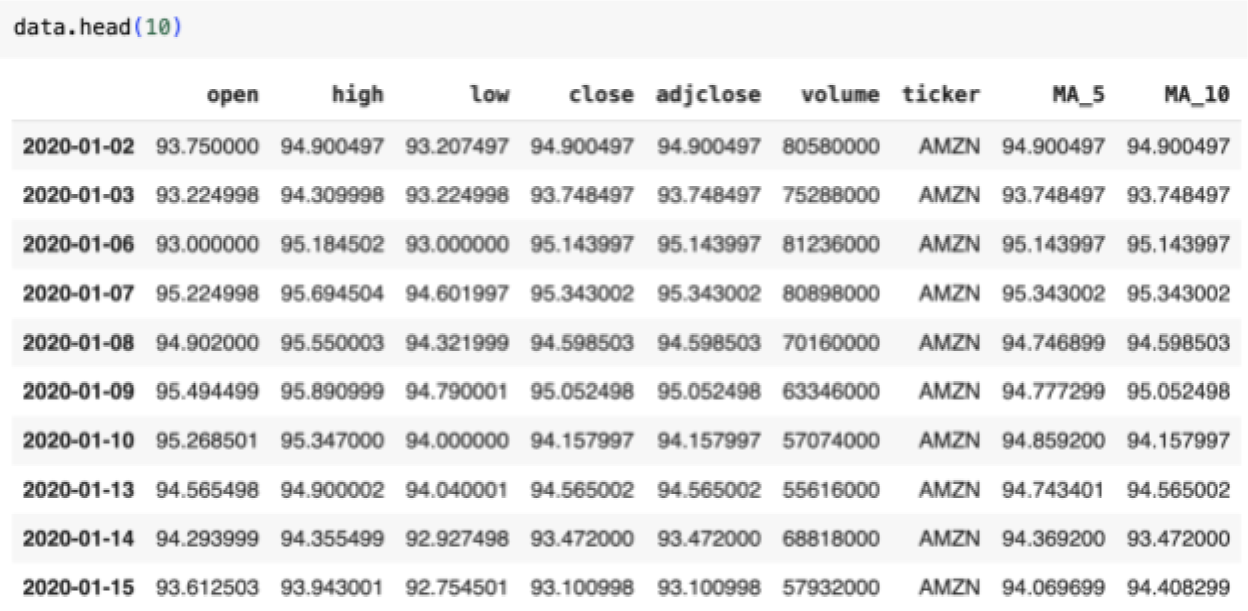
```
data['MA_10'] = data['close'].rolling(window=10).mean()  # n-day moving average
data['MA_10'].fillna(data['close'], inplace=True)
```

MA_10 means each day's row will be checked the previous 10 days, including this one. Mean of total closing price on that 10 days before filling into the new moving average 10 days column. However, if the range of the date specified in the data is not 10 days enough, the moving average on that day's cell is the same as the closing cell data.



```
data['MA_5'] = data['close'].rolling(window=5).mean()  # n-day moving average
data['MA_5'].fillna(data['close'], inplace=True)
```

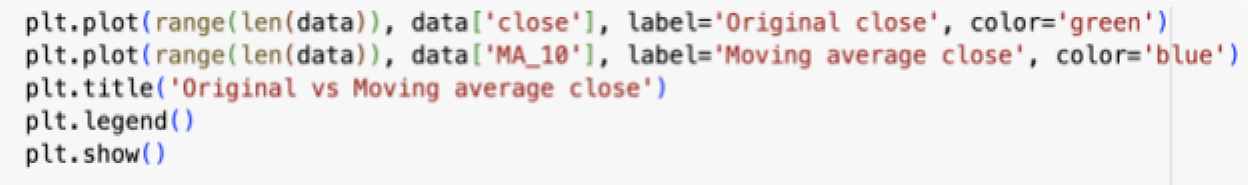
Same approach as MA_5, which is 5 days moving average calculated in total, before filling into the cell of the moving average 5 days column.



```
data.head(10)
```

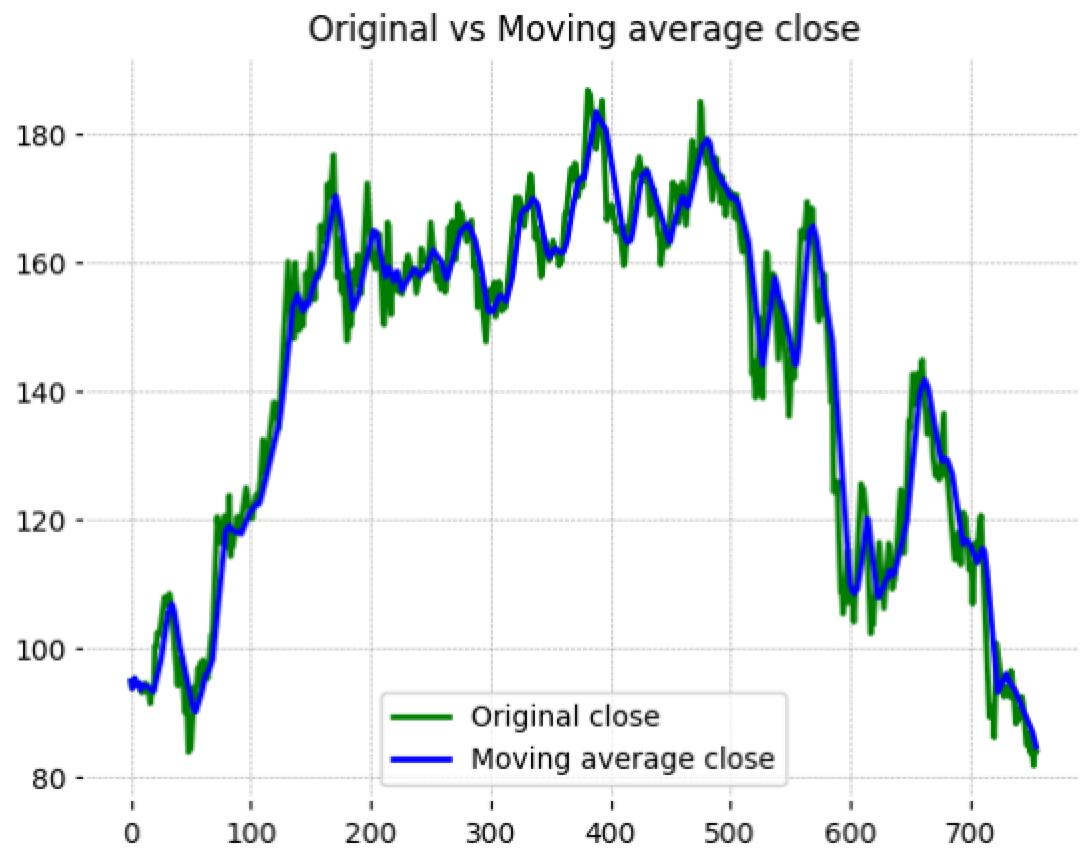
	open	high	low	close	adjclose	volume	ticker	MA_5	MA_10
2020-01-02	93.750000	94.900497	93.207497	94.900497	94.900497	80580000	AMZN	94.900497	94.900497
2020-01-03	93.224998	94.309998	93.224998	93.748497	93.748497	75288000	AMZN	93.748497	93.748497
2020-01-06	93.000000	95.184502	93.000000	95.143997	95.143997	81236000	AMZN	95.143997	95.143997
2020-01-07	95.224998	95.694504	94.601997	95.343002	95.343002	80898000	AMZN	95.343002	95.343002
2020-01-08	94.902000	95.550003	94.321999	94.598503	94.598503	70160000	AMZN	94.746899	94.598503
2020-01-09	95.494499	95.890999	94.790001	95.052498	95.052498	63346000	AMZN	94.777299	95.052498
2020-01-10	95.268501	95.347000	94.000000	94.157997	94.157997	57074000	AMZN	94.859200	94.157997
2020-01-13	94.565498	94.900002	94.040001	94.565002	94.565002	55616000	AMZN	94.743401	94.565002
2020-01-14	94.293999	94.355499	92.927498	93.472000	93.472000	68818000	AMZN	94.369200	93.472000
2020-01-15	93.612503	93.943001	92.754501	93.100998	93.100998	57932000	AMZN	94.069699	94.408299

Display the first 10 rows, and it displays the new column 5 days moving average and 10 days moving average. In MA_5, the fifth row is calculated in moving average, then the 10th row at MA_10 is calculated, other data below than 5 days and 10 days at MA_5 and MA_10 is same as close data.

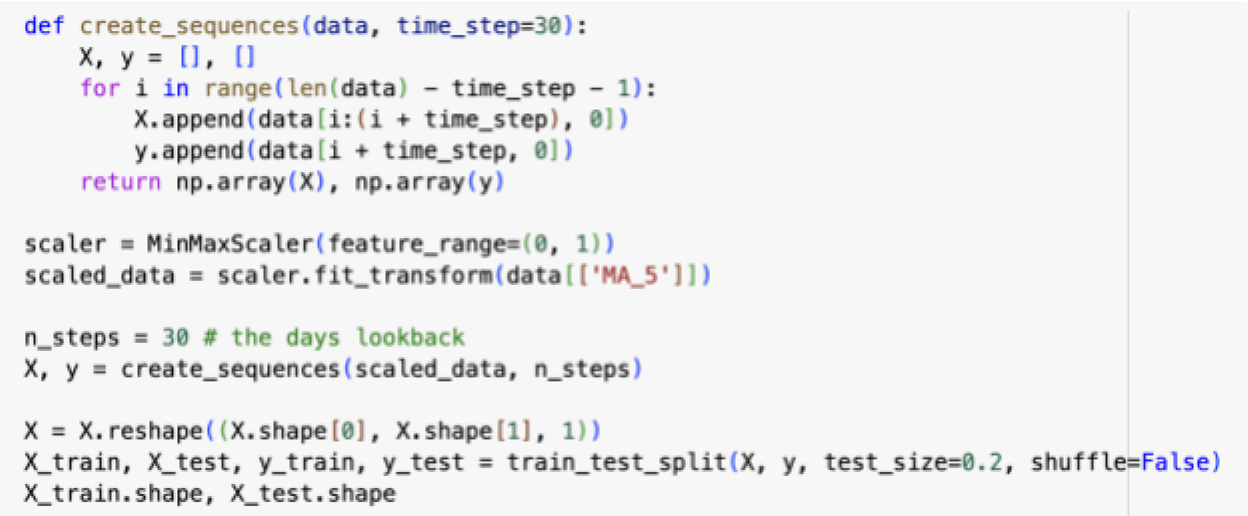


```
plt.plot(range(len(data)), data['close'], label='Original close', color='green')
plt.plot(range(len(data)), data['MA_10'], label='Moving average close', color='blue')
plt.title('Original vs Moving average close')
plt.legend()
plt.show()
```

Display the plot with original closing data and 10 days moving average closing data. It looks at the graph at below

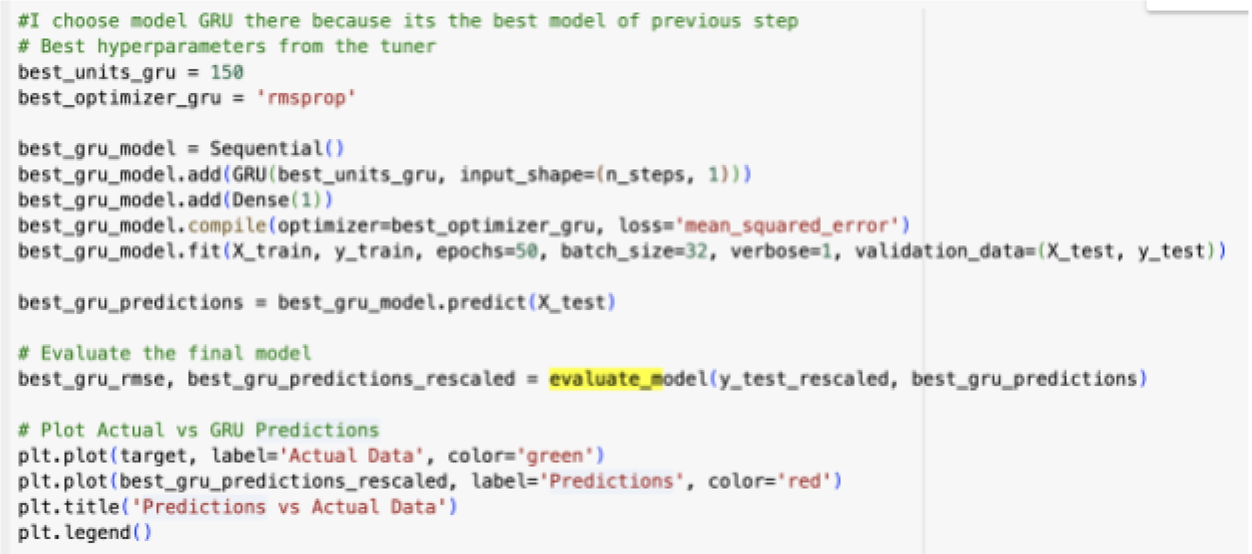


With the moving average 10 days data, its line becomes consistent without showing the outlier contrast with original close data.



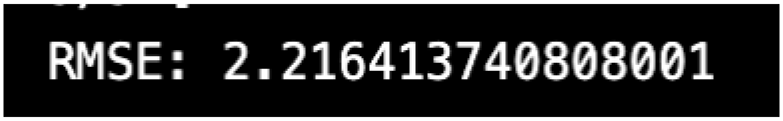
```
def create_sequences(data, time_step=30):
    X, y = [], []
    for i in range(len(data) - time_step - 1):
        X.append(data[i:(i + time_step), 0])
        y.append(data[i + time_step, 0])
    return np.array(X), np.array(y)

scaler = MinMaxScaler(feature_range=(0, 1))
scaled_data = scaler.fit_transform(data[['MA_5']])

n_steps = 30 # the days lookback
X, y = create_sequences(scaled_data, n_steps)

X = X.reshape((X.shape[0], X.shape[1], 1))
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, shuffle=False)
X_train.shape, X_test.shape
```

Creating the method sequences on the historical data of moving average 5 days in 30 days lookback. Then using MinMaxScaler to range the data feature between 0 and 1, and applied into the method to create sequences before reshape the X data to split train and test randomly.



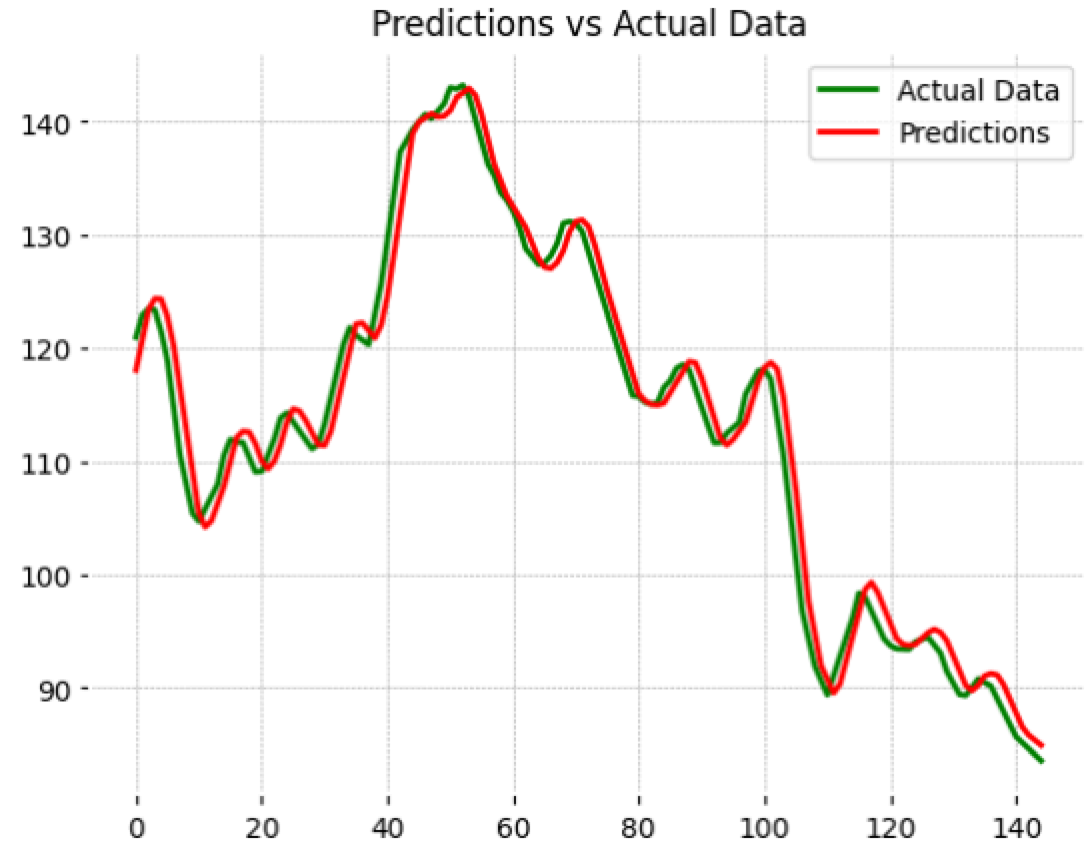
```
#I choose model GRU there because its the best model of previous step
# Best hyperparameters from the tuner
best_units_gru = 150
best_optimizer_gru = 'rmsprop'

best_gru_model = Sequential()
best_gru_model.add(GRU(best_units_gru, input_shape=(n_steps, 1)))
best_gru_model.add(Dense(1))
best_gru_model.compile(optimizer=best_optimizer_gru, loss='mean_squared_error')
best_gru_model.fit(X_train, y_train, epochs=50, batch_size=32, verbose=1, validation_data=(X_test, y_test))

best_gru_predictions = best_gru_model.predict(X_test)

# Evaluate the final model
best_gru_rmse, best_gru_predictions_rescaled = evaluate_model(y_test_rescaled, best_gru_predictions)

# Plot Actual vs GRU Predictions
plt.plot(target, label='Actual Data', color='green')
plt.plot(best_gru_predictions_rescaled, label='Predictions', color='red')
plt.title('Predictions vs Actual Data')
plt.legend()
```

Compile the model GRU with its hyperparameters because it is the best one previously, then predict the X test to evaluate the final model. After that, plotting to show between the target data (close) and the resaled_data after passing to inverse transform the data.



```
RMSE: 2.216413740808001
```

The root means square display when run at the evaludate_model, which is the result of around 2.2. It is quite impressive as compared to the previous model run like ARIMA, SARIMA, RF, or LSTM…



The plot displays the consistencies between the actual data and predictions as training with GRU model.

REFERENCES

[1]. Hao, PY, Kung, CF, Chang, CY, & Ou, JB 2021, 'Predicting stock price trends based on financial news articles and using a novel twin support vector machine with fuzzy hyperplane', *Applied Soft Computing Journal*, vol. 98.

[2]. Langi, AZR, Pitara, SW, & Kyspriyanto 2012, 'Stock prices trends analysis using wavelet transform', *International Conference on Cloud Computing and Social Networking (ICCCSN),* pp. 1-4, DOI: 10.1109/ICCCSN.2012.6215753.

[3]. Yoon, JP, Lee, J, & Kim, S 2000, ‘Trend similarity and prediction in time-series databases’,, Data *Mining and Knowledge Discovery: Theory, Tools, and Technology II*, DOI: https://doi.org/10.1117/12.381734