Stock Price Prediction:
Dual-List Stocks in Hong Kong and New York

Alan Hui
Big Data Technology
Hong Kong University of Science and Technology
Clearwater Bay, N.T., Hong Kong
slhui@ust.hk

Edgar Leung
Big Data Technology
Hong Kong University of Science and Technology
Clearwater Bay, N.T., Hong Kong
etwleung@ust.hk

Yuki Wong
Big Data Technology
Hong Kong University of Science and Technology
Clearwater Bay, N.T., Hong Kong
lwwongae@ust.hk

ABSTRACT

We selected a popular stock in Hong Kong stock market, which is HSBC stock (0005.HK). This stock is cross-listed on exchanges in Hong Kong and New York. We studied the patterns of information flow for HSBC stocks, trying to find out the relationship between these dual-listed stocks. Results showing that the offshore (New York) markets has significant impact on the domestic (Hong Kong) markets. We can use offshore closing price to predict the trend of domestic markets of the next day. We used 2 different approaches to fit a prediction model, ARIMA and LSTM models. Results showing that ARIMA model is more reliable than LSTM on this stock price prediction.

Keywords: GARCH, ADR, LSTM, Stock Price Prediction, Neural Network, Machine Learning, Information Transmission

1 INTRODUCTION

HSBC Holding Plc is a British multinational investment bank and financial services holding company. It was the 7th largest bank in the world by 2018, and the largest in Europe, with total assets of US$2.558 trillion (as of December 2018).

It has a dual primary listing on the Hong Kong Stock Exchange and the London Stock Exchange and is a constituent of the Hang Seng Index and the FTSE 100 Index. It also has secondary listings on the New York Stock Exchange, Euronext Paris, and the Bermuda Stock Exchange. [1]

As it is the top-10 popular Hong Kong Stock and the transaction volume is relatively high, it is a stock that is representable of the Hong Kong Stock Market.

HSBC also has a listing on the New York Stock Exchange. This offshore market is effective for ask/bid after Hong Kong Market is closed and Hong Kong Market resumes right after offshore market closes. Due to this time zone different behaviour, it is interesting to study whether there is opportunity to leverage the result of New York Market close price to predict the next day of Hong Kong Market. We believe there is a relationship between these two markets. If this assumption is correct, we can further take advantage of this stock.

We tried 2 different approaches to tackle this problem, using traditional time series models (ARIMA) and deep learning model (LSTM – Long Short Term Memory, a class of recurrent neural networks). In this study, we compare the difference between these 2 models and try to fit a model that can have a good prediction result based on the New York closing price to predict Hong Kong stock price.

2 DATA DESCRIPTION

To start with this study, we extracted 2 datasets, HSBC HK history stock price and HSBC US history stock price from yahoo finance [2], data time range start from Jan-2000 to Nov-2019.

In this data, we have 6 data fields:

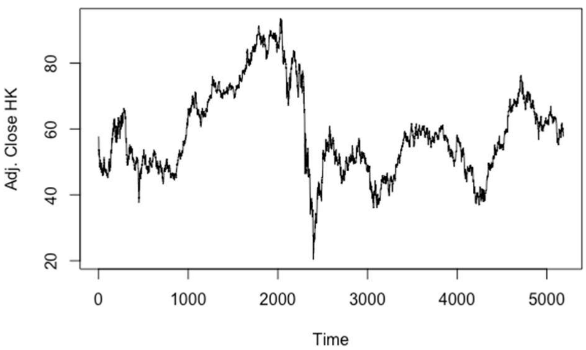
- **Open**: this is the stock price of HSBC when the stock market open for trading
- **High**: this is the highest bid cost of the day
- **Low**: this is the lowest ask cost of the day
- **Close**: this is the stock price of HSBC when the stock market close for trading (with adjustment for splits)
- **Adj. Close**: this is the adjusted close price, adjusted for dividends and splits
- **Volume**: this is the transaction volume

We split our dataset into training data and test data. We set a common cutoff date for both models, which is 1-Oct-2019. All data before this cutoff date is used as training data.

3 ARIMA-GARCH MODEL ON THE TIME SERIES ITSELF

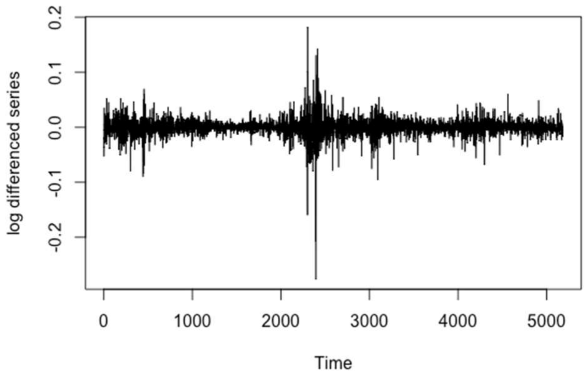
3.1 Stationarity

To perform ARIMA model, we should first make sure if the time series is stationary or not. If we plot the stock price against time, the series is clearly neither stationary in mean nor stationary in variance.



According to the Augmented Dickey-Fuller Test on the adjusted close price in Hong Kong market, we got the p-value 0.24, which is greater than 0.05. We could not reject the null hypothesis that the time series is non-stationary.

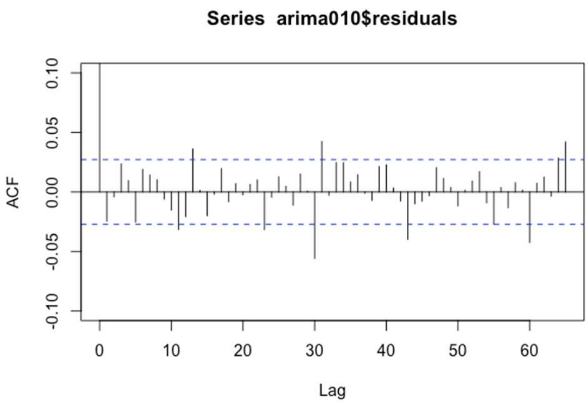
To overcome this, we took log difference on the time series.

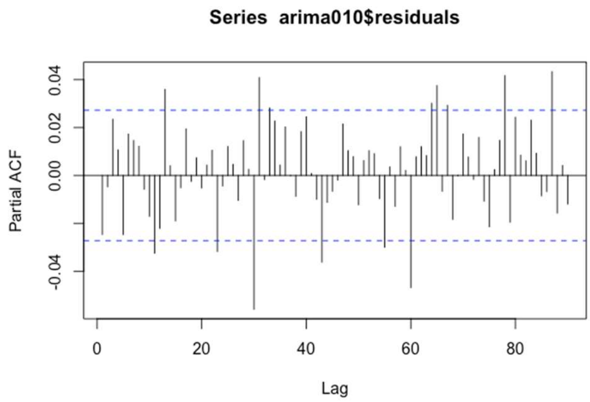


We can see that the series is much more stationary. Augmented Dickey-Fuller Test shows that the p-value is 1e-04. We can conclude that the time series is stationary after log differencing.

3.2 Model 1: ARIMA(0,1,0)

Log differencing on the time series could be interpreted as ARIMA(0,1,0) model on the log transformed time series. To see if ARIMA(0,1,0) is adequate, we could plot the ACF and PACF on the residual of the model.





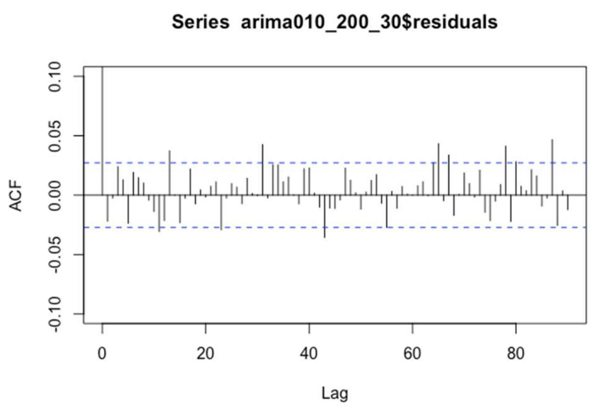
We saw that there are a lot of significant lags in both ACF and PACF.

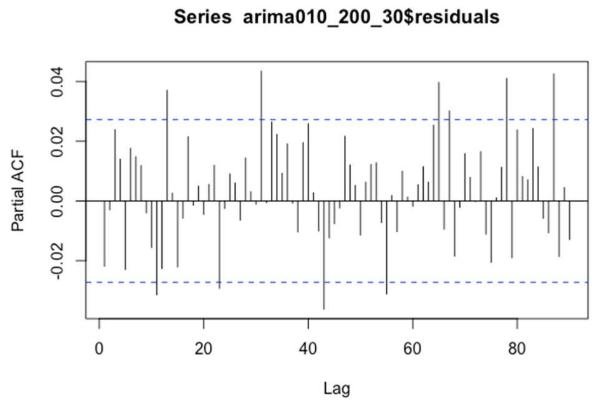
If we plot the Ljung-Box test at different lags, p-value is smaller than 0.05 starting from lag 15. We can reject the null hypothesis that the residual is white noise.

Box-Ljung test	Lag 5	Lag 15	Lag 25	Lag 35
P-value	0.07711	0.008602	0.0251	9.94E-05

3.3 Failed attempt: Seasonal AR2

From the PACF, we clearly significant at lag 30 and lag 60. It seems like there is a seasonal AR2 effect in a 30 days period. This is reasonable to have a monthly seasonality so that same day in last one or two months could be a good indicator on predicting the stock price of the day.





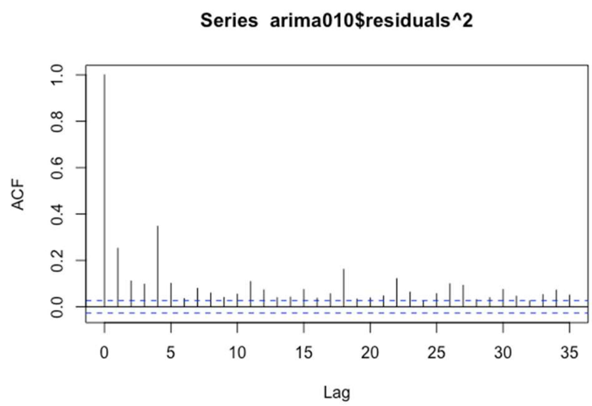
However, if we look at the ACF and PACF of the residual. We can still see a lot of significant lags.

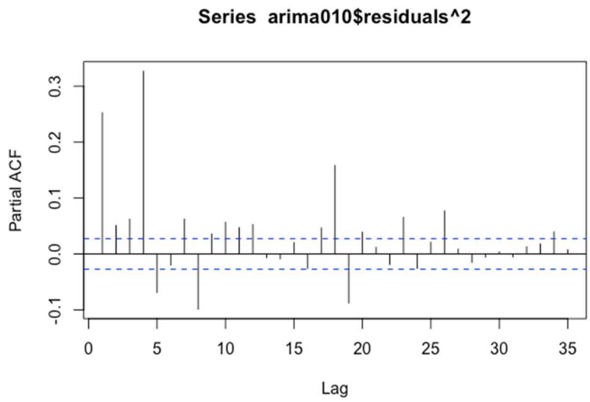
Box-Ljung test	Lag 10	Lag 20	Lag 30
P-value	0.1797	0.02603	0.08018

The Box-Ljung test at lag 20 is still significant. It seems like the seasonal model is a good fit.

3.4 Model 2: Garch(1,1) with mean

Instead of trying seasonal arima model, perhaps we should first test if there is any GARCH effect. The following plots show the ACF and PACF of the square of the residual of ARIMA(0,1,0) model.





We can see that there are many significant lags in both ACF and PACF. The p-value of Box-Ljung test at lag 31 is close to zero. We can see there is a very auto-correlation of the variance against time.

We should try ARIMA(0,1,0)~GARCH(1,1) model.

```
Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
mu     0.0001682   0.0139372    0.012     0.99
omega  0.0110094   0.0023423    4.700  2.6e-06 ***
alpha1 0.0612329   0.0061634    9.935  < 2e-16 ***
beta1  0.9350019   0.0062855  148.756  < 2e-16 ***
```

If we perform t-test on the parameters, we can see that the mean Mu is actually close to zero and not significant (p-value=0.99). This makes sense because we performed differencing on the time series and it should be stationary in mean. It is better to remove the mean from the model.

3.5 Model 3: GARCH(1,1) without mean

```
Error Analysis:
        Estimate  Std. Error  t value Pr(>|t|)
omega   0.011009    0.002342    4.700  2.6e-06 ***
alpha1  0.061231    0.006162    9.937  < 2e-16 ***
beta1   0.935004    0.006284  148.796  < 2e-16 ***
```

After we take out the mean, we saw that all the parameters are significant. However, we saw that alpha1 plus beta1 is very close to 1 (0.061+0.935 ~ 1). This suggest us to try iGARCH model.

3.6 Model 4: iGARCH(1,1) without mean

	GARCH	iGARCH
BIC	3.1785	3.1775

The BIC of iGARCH model is slightly smaller than GARCH model. It seems iGRACH model is a better choice.

Residual	Lag 10	Lag 20	Lag 30
Box-Ljung test P-value	0.1568	0.1901	0.4716

Squared residual	Lag 10	Lag 20	Lag 30
Box-Ljung test P-value	0.3884	0.5739	0.7401

If we perform Box-Ljung test at different lag for both residual and squared residual of the ARIMA(0,1,0)~iGARCH(1,1) model, we can see none of them are significant. It seems like this model is a good fit.

4 ARIMA-GARCH MODEL WITH EXOGENOUS VARIABLE

Although ARIMA(0,1,0)~iGARCH(1,1) is a good fit on the log transformed adjusted closing in HK market. We could further enhance our model by seeing if we can predict better by taking reference on the offshore (New Work) market.

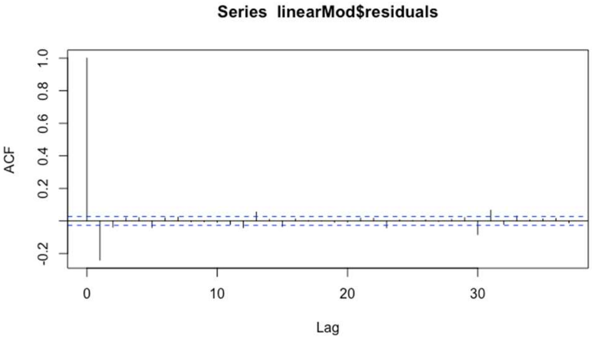
4.1 Model 5: Linear regression

We could use simple linear regression to check if the log differenced New Work stock price one day before is significant in predicting that of stock price in Hong Kong. Here is the model:

$Y \sim b_0 + b_1X$

where

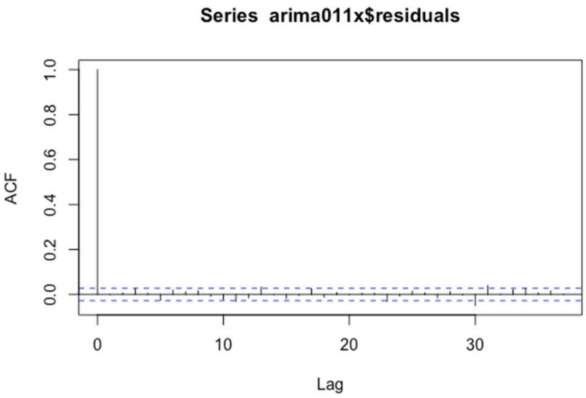
- Y is the log differenced HK adjusted closing;
- X is the log differenced NY adjust closing with one day lag.

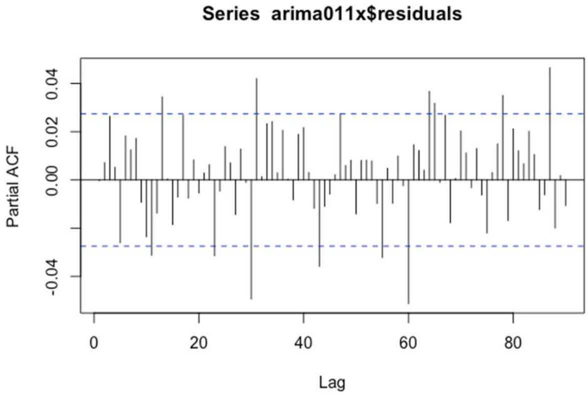


If we plot the ACF of the residual, we observed a significant MA1 effect. It is worth to try MA1 model with exogenous variable of New York stock price.

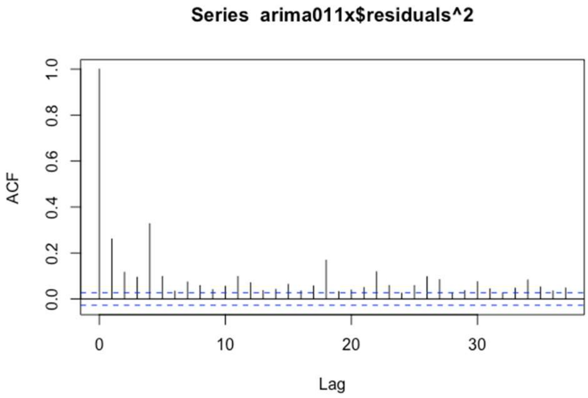
4.2 Model 6: ARIMAX(0,1,1)

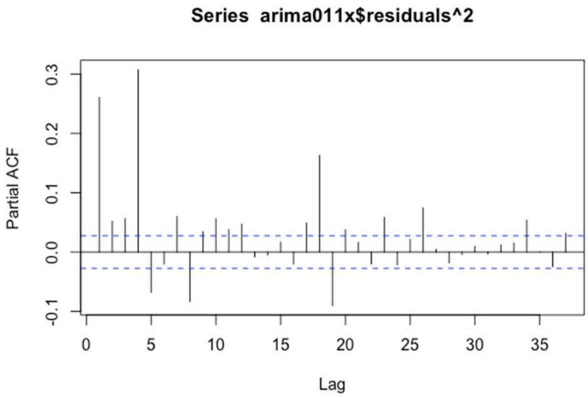
We use ARIMAX(0,1,1) with X represent the exogenous variable[3]. The target time series is log transformed Hong Kong adjusted close. The exogenous variable is the log differenced New York adjusted close with one day lag.





By looking at the ACF and PACF of the residual, we can see that the MA1 effect properly addressed. However, there are still a lot of significant lags in PACF.





In fact, the ACF and PACF of the squared residual showed a lot of significant lag. This means that there is a strong GARCH effect.

We should try GARCH model with exogenous variable.

4.3 Model 7: ARIMAX(0,1,1)-GARCH(1,1)

We tried ARIMAX(0,1,1)-GARCH(1,1) model. Base on the experience we had in modeling GARCH model without exogenous variable, we took out mean Mu from the model as this will never be significant given that we are modeling the log differenced time series. Here is a summary of the estimated parameters.

	Estimate	Std. Error	t value	Pr(>|t|)
ma1	-0.568123	0.045827	-12.3970	0.000000
mxreg1	0.682989	0.037948	17.9980	0.000000
omega	0.013103	0.004876	2.6874	0.007202
alpha1	0.104946	0.019056	5.5073	0.000000
beta1	0.891441	0.019389	45.9769	0.000000

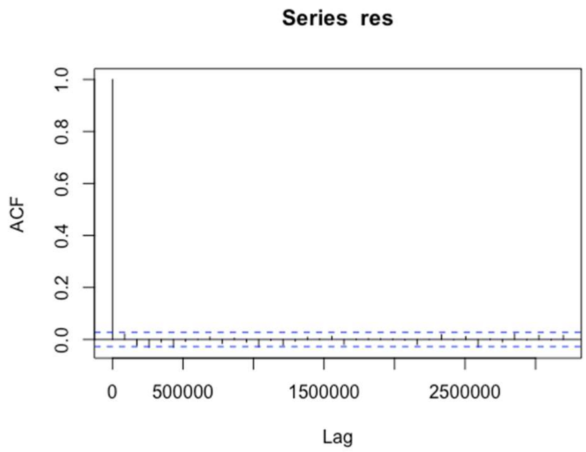
Again, we observed that alpha1 plus beta1 is very close to 1 ($0.105+0.891 \sim 1$). We should try iGARCH model.

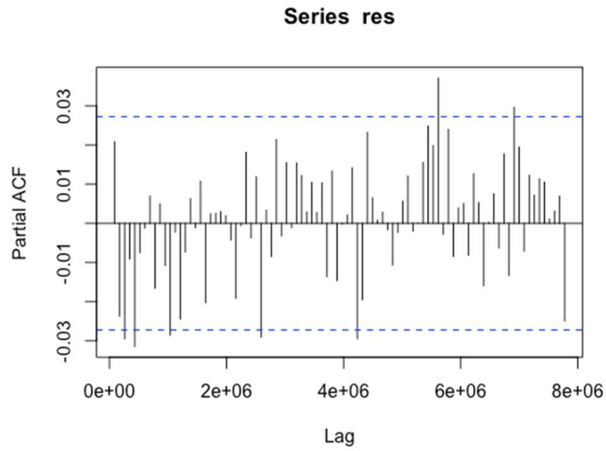
4.4 Final Model: ARIMAX(0,1,1)-iGARCH(1,1)

Here is the summary of the ARIMAX(0,1,1)-iGARCH(1,1) model parameters.

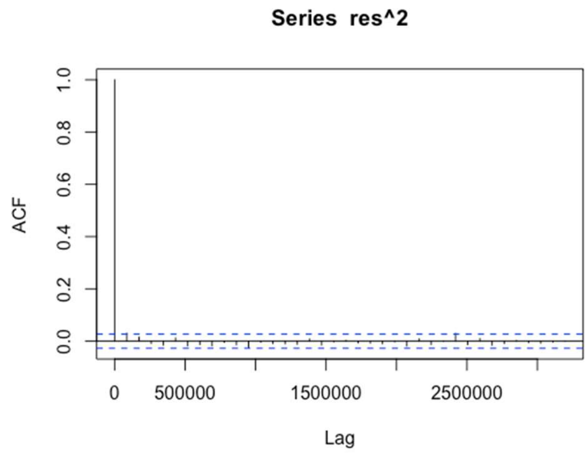
	Estimate	Std. Error	t value	Pr(>|t|)
ma1	-0.569857	0.066255	-8.6009	0.000000
mxreg1	0.684334	0.051149	13.3792	0.000000
omega	0.011668	0.003877	3.0096	0.002616
alpha1	0.108248	0.019303	5.6079	0.000000
beta1	0.891752	NA	NA	NA

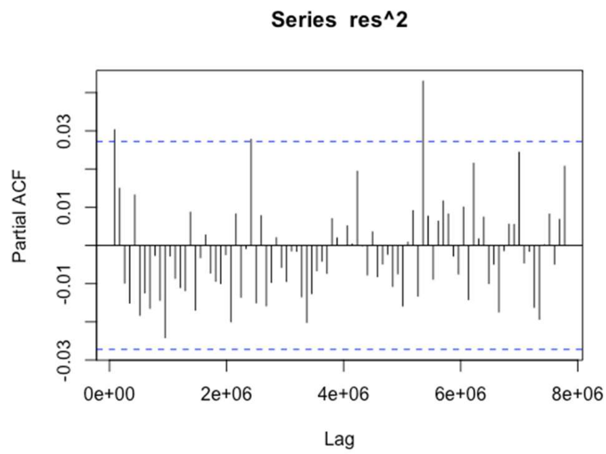
We can see that all the parameters are significant. The BIC of iGARCH model (2.8508) is slightly smaller than that of GARCH model (2.8523).





The ACF and PACF of the model residual shows acceptable result that there are not many significant lags. Same with the ACF and PACF of the squared residual.





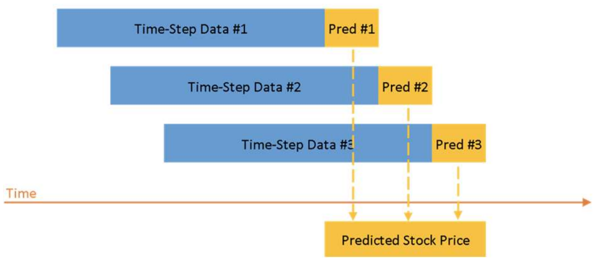
Residual	Lag 10	Lag 20	Lag 30
Box-Ljung test P-value	0.06979	0.1194	0.1943

Squared residual	Lag 10	Lag 20	Lag 30
Box-Ljung test P-value	0.1867	0.4303	0.4538

We performed Box-Ljung test at different lags for both residual and squared residual. None of them are significant. This further conclude that there is no outstanding ARMA effect or GARCH effect.

4.5 Validation on forecasting power

To test how well the model performs, we employ the method of walk forward modelling [4].



As we only have New York stock price one day in advance as the exogenous variable, we could only take this advantage to predict one day ahead.

Starting from 2019 Oct 1st, we treat the data before the cut off date as training set, and predict only one day. Then we move the cut off date one day ahead, and re-train the model with one more day. We keep repeating these steps to get a series of predicted values.

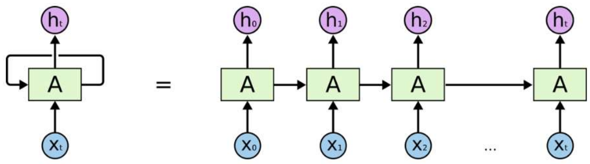
The root mean squared error (R.M.S.E) between the actual and predicted series is 0.49337. Comparing with the best ARIMA-GARCH model without exogenous variable we obtained from section 3, i.e. ARIMA(0,1,0)-iGARCH(1,1) without mean (R.M.S.E. = 0.5755835), the model with the help of US stock market shows a better result.

In this paper, we also compare the same ARIMA model that used not only US adjusted closing, but also all other variables such as Open, High, Low and Volume. Result shows that this model outperform all other models. We will leave the discussion in section 6.

5 Neural Networks

5.1 Recurrent Neural Networks (RNN)

Unlike the standard feed-forward networks, RNN has an additional loop in the architecture [5]. So recurrent networks have two sources of input, the present and the recent past. This connection add memory to the network and allow RNN to learn broader abstractions from the input sequence.



RNN works well in the situation where the gap between the relevant information and the prediction is small. When this gap grows, RNN is no longer able to connect the information from the very past to the current prediction.

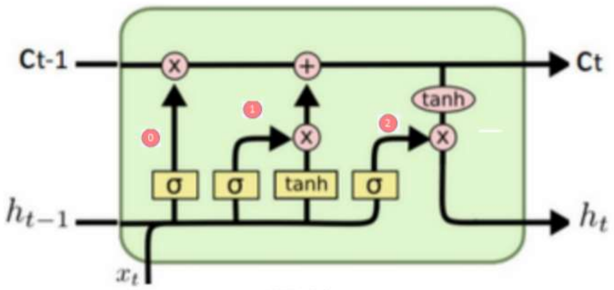
5.2 Long Short Term Memory (LSTM)

The shortcoming of RNN cannot handle long-term dependencies is addressed by LSTM. The key idea is to use memory cells and gates:

$$\mathbf{c}_{(t)} = \mathbf{f}_t \otimes \mathbf{c}_{(t-1)} + \mathbf{i}_t \otimes \mathbf{a}^{(t)}$$

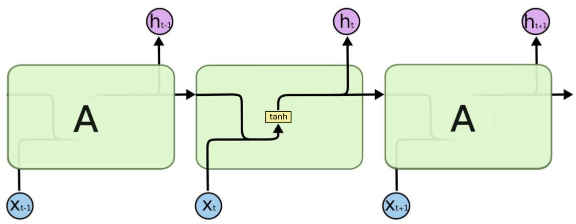
$\mathbf{c}_{(t)}$ is the memory state at t; $\mathbf{a}^{(t)}$ is the new input at t.

The forget gate $\mathbf{f}_t$ (range from 0 to 1) is controlling whether the current memory is kept while the input gate $\mathbf{i}_t$ with the same range is controlling if the current value in the memory cell is replaced by the new input.

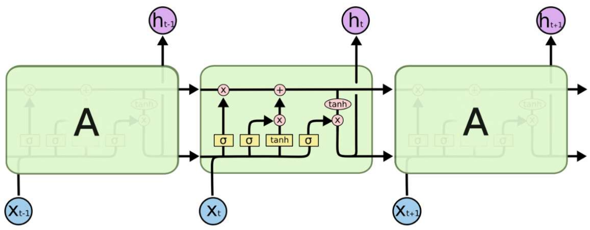


LSTM Cell

Below diagram shows a standard RNN contains a single neural network layer.



While comparing to LSTM, LSTM contains four interacting layers.



The main reason that LSTM can learn longer term dependencies is due to the horizontal line which passes through the top of the diagram. This line runs through the entire chain where useful information can flow through without losing details.

In our paper, LSTM model is chosen in order to capture both long and short term dependencies.

5.3Experiments

LSTM consumes input in 3-dimensional array format [batch_size, time_steps, Features]

These are three important parameters to consider:

- **Batch Size** – how many samples of input does the Neural Network see before updating the weights.
- **Time steps** – how many days of data back in time does the Neural Network see for each prediction
- **Features** – number attributes to include in each time step.

The diagram visualized the case of our data when time step = 3 and features = 12.

Date	Open.hk	High.hk	Low.hk	Close.hk	Adj.Close.hk	Volume.hk	Open.ny.lag	High.ny.lag	Low.ny.lag	Close.ny.lag	Adj.Close.ny.lag	Volume.ny.lag
3/1/2000	111.00	111.50	107.50	108.00	57.75	3,908,164.00	71.44	71.50	70.31	70.69	26.84	59,500.00
4/1/2000	106.00	106.50	105.00	106.00	56.68	6,895,378.00	67.06	68.00	66.00	66.38	25.20	84,500.00
5/1/2000	101.00	102.00	99.50	100.50	53.74	14,657,098.00	64.75	66.31	63.75	66.00	25.06	118,200.00
6/1/2000	101.00	102.00	96.25	97.00	51.87	10,632,084.00	64.00	64.38	63.19	64.13	24.35	105,400.00
7/1/2000	98.50	101.00	97.50	100.50	53.74	8,368,286.00	63.75	65.94	63.38	65.94	25.04	117,700.00
10/1/2000	103.00	103.00	99.50	100.00	53.47	6,201,139.00	63.75	66.00	63.38	66.00	25.06	61,400.00

i.e. All the HK and NY open/high/low/close/adjusted close/volume of 3 consecutive days (big BLUE box) is used to predict the next day HK adjusted close price (small BLUE box)

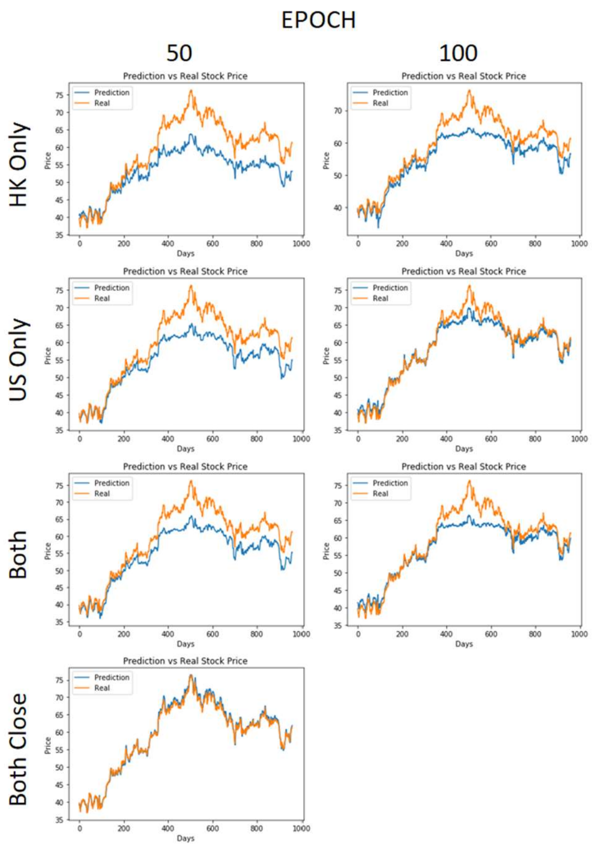
5.3.1 Experiment 1: Feature Extraction

Four feature sets were chosen and experimented:

1. HK open/high/low/close/adjclose/volume
2. US open/high/low/close/adjclose/volume
3. Both open/high/low/close/adjclose/volume
4. Both adj close only

The conclusions of this experiment are:

- US stock data has better predictive power than HK stock data to predict HK stock price
- US adjusted close alone is the best predictor.
- Too much irrelevant data worse off the model. Open/high/low/volume actually have not much correlation to the next day closed price, including them in the model will produce more noises.



5.3.2 Experiment 2: Batch Size

Determining batch size is equal to determining which learning algorithm:

- **Batch Gradient Descent.** Batch Size = Size of Training Set
- **Stochastic Gradient Descent.** Batch Size = 1
- **Mini-Batch Gradient Descent.** 1 < Batch Size < Size of Training Set

We have two considerations on batch size. One is training stability while another one is model validation.

Training stability

Below table shows the first 5 epoch of both training process, we can see that:

- Mini-batch gradient descent converges to a more stable model exemplified by lower variance in accuracy.
- Stochastic gradient descent results rapid learning but a volatile learning process with higher variance in accuracy. E.g. Training error increases during epoch 0, 1 and 2.

	Batch Size = 32 1 Epoch - 7s		Batch Size = 1 1 Epoch - 150s	
Epoch	Training Error	Validation Error	Training Error	Validation Error
0	0.04492	0.02013	0.00148	0.04528
1	0.04352	0.01918	0.00209	0.03182
2	0.03687	0.01997	0.00302	0.03253
3	0.03759	0.01975	0.00248	0.04449
4	0.03499	0.01927	0.00200	0.03877

Model Prediction

In our paper, we used rolling-forecast scenario, also called walk-forward model prediction.

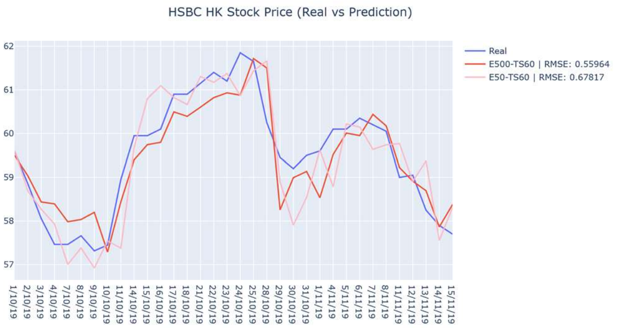
Each time step of the test dataset will be walked one at a time. A model will be used to make a forecast for the time step, then the actual expected value from the test set will be taken and made available to the model for the forecast on the next time step.

Ideally, for our time series stock price prediction problem, it is desirable to use a large batch size when training the network and a batch size of 1 when making predictions in order to predict the next step in the sequence. Therefore we used batch size = 32 during model training while batch size = 1 during prediction.

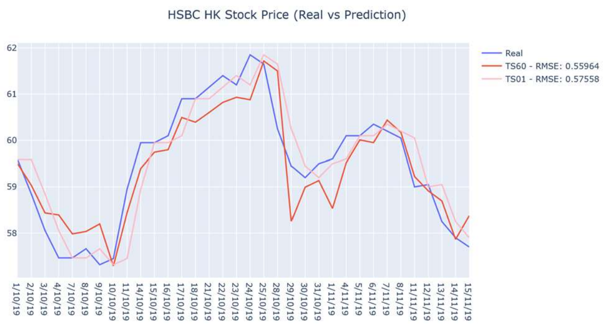
5.3.3 Experiment 3: Time Steps

Unlike ARIMA model, we leave LSTM model to explore the data lag dependencies itself. Therefore we set the time steps = 60 (2 months to predict next day price).

The prediction accuracy from our LSTM model is fair. We observe a much better result can be achieved when using more epoch in training.



In the end, we want to share an interesting finding. If time step is set to 1, the prediction accuracy is also quite good. However, when you look at the chart below carefully, you can see that the TS01 line is same as real but shifted 1 day. i.e. LSTM model copy the previous day HK close price as next day prediction.

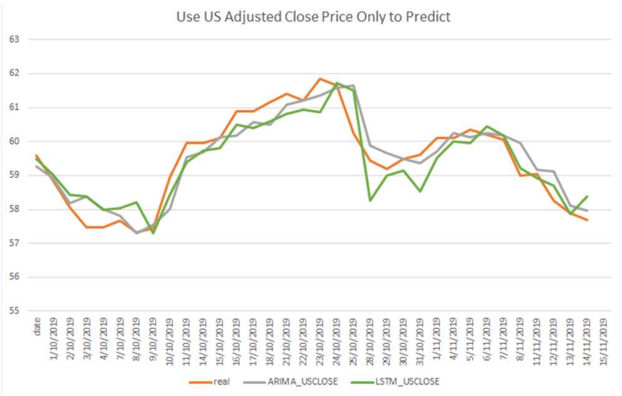


6 Comparison

In this paper, we fitted 2 models (ARIMA and LSTM) for predicting HK HSBC stock price. We also tried to use US adjusted close price only or use all available US stock values to predict. Therefore, we have 4 combinations of prediction results as below.

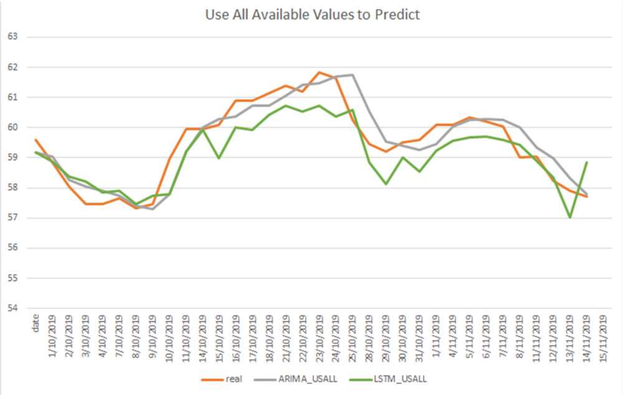
Model	RMSE
ARIMA (US Close Only)	0.49337
LSTM (US Close Only)	0.55964
ARIMA (US All Values)	0.52783
LSTM (US All Values)	0.71868

Below are the prediction results using US Adjusted Close as the only feature. From the graph, we can see that both models have similar up and down pattern. However, the difference between real values and predicted values is smaller when using ARIMA model than using LSTM model.



Below diagram is showing the prediction results while using all available US stock values as the input feature to predict HK stock price. However, we can see that the difference between real values and predicted values are much larger when using US adjusted close price only.

This result tells us that the extra added values are not useful to predict HK close price and might not have a direct relationship that impact the HK stock price.



Back to the results of using US adjusted close price only, we can also find that ARIMA model is out perform than LSTM model. As our scenario is a one-step forecasting on univariate datasets, which in other words is forecasting a short term result. We can conclude that LSTM model is not performing better than ARIMA model for short term prediction. There is also a conclusion that can be found from our study is that under HK HSBC stock, there is not much long-term impact, the stock price is impacted by the US stock market price by a very short period. The impact is not long lasting.

7 CONCLUSION

There were studies [6] on LSTM achieved better results than ARIMA. However, in our case, ARIMA is clearly the better model.

- Simple data series and relationship

  Neural networks works great in identifying the unobservable or stochastic data dependencies. In our case, the data series is rather simple. A direct information transmission from the close price in US stock market to that of HK stock market is also observed. We are able to tailor made configuration on ARIMA model for this specific HSBC stock which yield a more accurate result.

- Short prediction period

  Since it is a daily stock price prediction problem, it has to adopt walk-forward prediction approach to re-train the model and predict the next day price. Since we only predict next day price, ARIMA works better. Some research did find a conclusion that LSTM can have a better prediction accuracy when the prediction period is long.

- Data volume and Training time

  LSTM model requires a large dataset and long training time to train the model up to standard. In our case, we spent around 4~5 hours to train the LSTM model. It is a big advantage of ARIMA model over LSTM model.

We cannot draw a simple conclusion here that ARIMA model is better than LSTM model in all senarios. Experiments needed to be carried out carefully each time in order to choose the best model for your particular data series and use cases.

REFERENCES

[1] https://en.wikipedia.org/wiki/HSBC

[2] https://finance.yahoo.com/quote/0005.HK/history?p=0005.HK

[3] Wiwik, Retno Aulia, Yuni (2015) Performance Comparisons between Arima and Arimax Method in Moslem Kids Clothes Demand Forecasting: Case Study

[4] Chad (2018) Stock Prediction with ML: Walk-forward Modeling. The Alpha Scientist. Retrieved from https://alphascientist.com/walk_forward_model_building.html

[5] http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[6] Forecasting Economics and Financial Time Series: ARIMA vs. LSTM, Sima Siami-Namini, Akbar Siami Namin, 2018