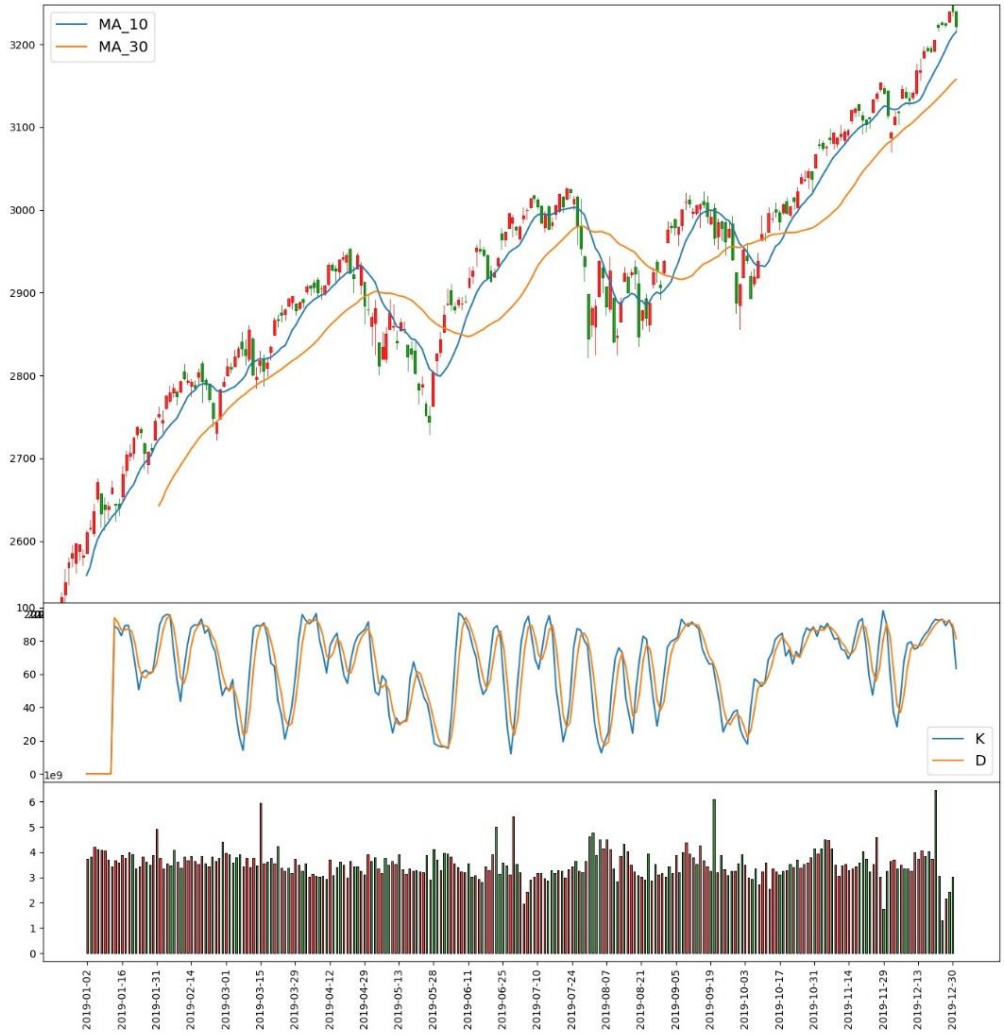
FinTech HW4

徐瑋辰 R08942025

(1) Show your figures from 2019/1/1 to 2019/12/31



(2) Please discuss what did you do for data preprocessing.

The following features are selected for training: ‘High’, ‘Low’, ‘Open’, ‘Close’, ‘Volume’, ’Moving Average 10 days’, ’Moving Average 30 days’ and ’K, D from KD line chart’. Each feature is normalized on a scale of 0 to 1. No other preprocessing is used.

(3) In this exercise, you can choose the batch size on your design. The time step should be 30 because we want to use the last 30 days to predict the ’Close’ value of the next day. And input dimension will be depend on your (ii) design.

I choose 32 as the batch size. I select 9 features (open, high, low, close, volume, K, D Ma_10, Ma_30).

(4) Please construct a RNN model with SimpleRNN cell for predicting the ’Close’ value of the next day according to mean square error. Please explain how do you design your model?

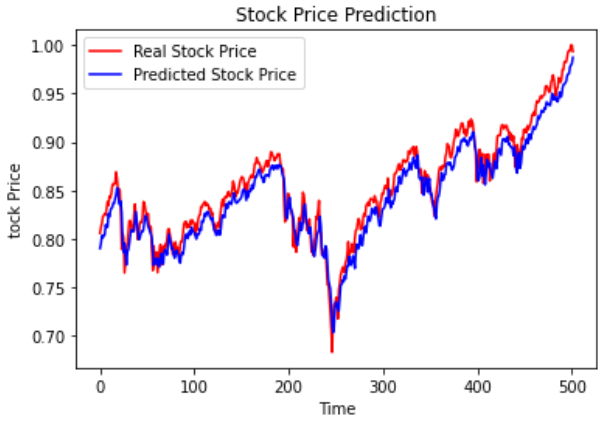
```python
def SimpleRNN_model(shape):

    batch_input_shape = (None, shape[1], shape[2])
    print(batch_input_shape)
    model = Sequential()
    model.add(SimpleRNN(64, batch_input_shape = batch_input_shape, unroll=True))
    # model.add(Dropout(0.2))
    model.add(Dense(1))
    # model.add(Activation('softmax'))
    model.compile(loss = "mse", optimizer = "adam")
    # model.summary()

    return model
```

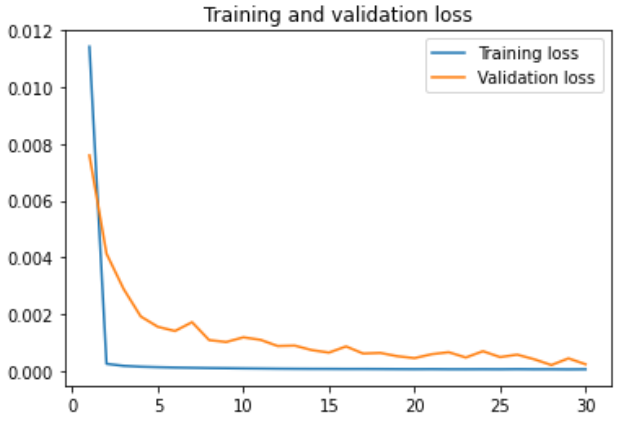
(5) Plot loss curve chart and the prediction of 'Close' value in validation part.

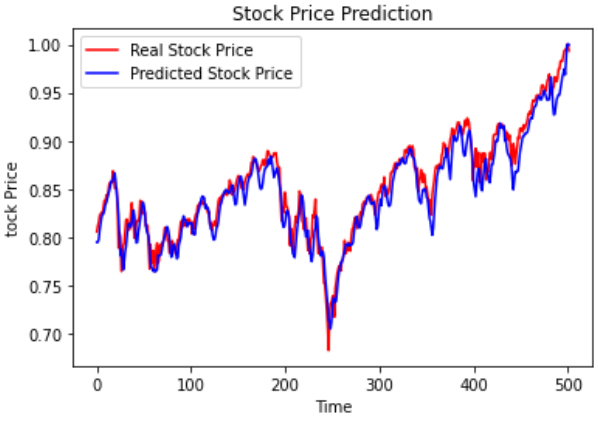
MSE: 8.056619809990817e-05



(6) Substitute LSTM cell for SimpleRNN and repeat (iv), (v).

```
Model: "sequential"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
lstm (LSTM)                  (None, 10)                800
_________________________________________________________________
dense (Dense)                (None, 1)                 11
=================================================================
Total params: 811
Trainable params: 811
Non-trainable params: 0
_________________________________________________________________
```



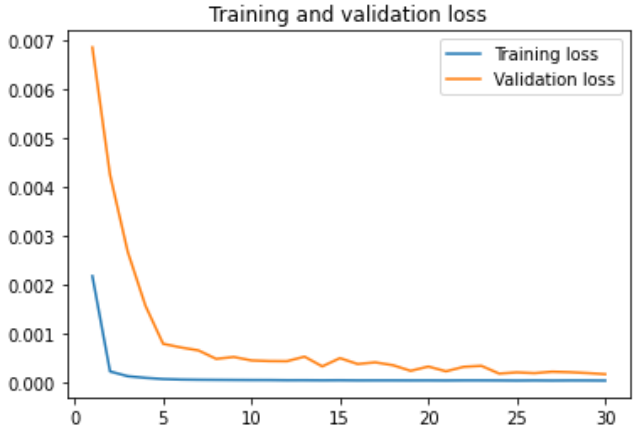


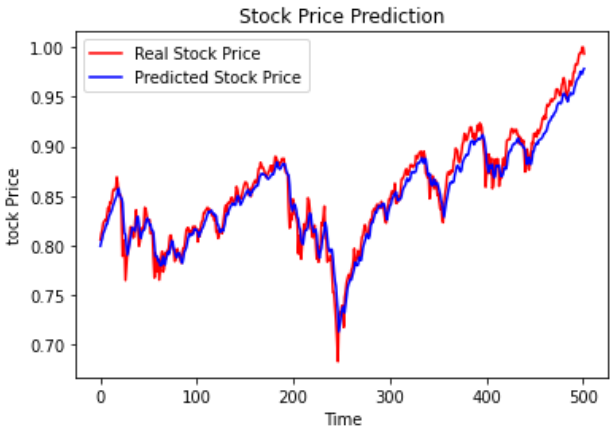
MSE: 0.000225179938623515

(7) Substitute GRU cell for SimpleRNN and repeat (iv), (v).

```
Model: "sequential_2"
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
gru (GRU)                    (None, 10)                630
_________________________________________________________________
dense_2 (Dense)              (None, 1)                 11
=================================================================
Total params: 641
Trainable params: 641
Non-trainable params: 0
_________________________________________________________________
```

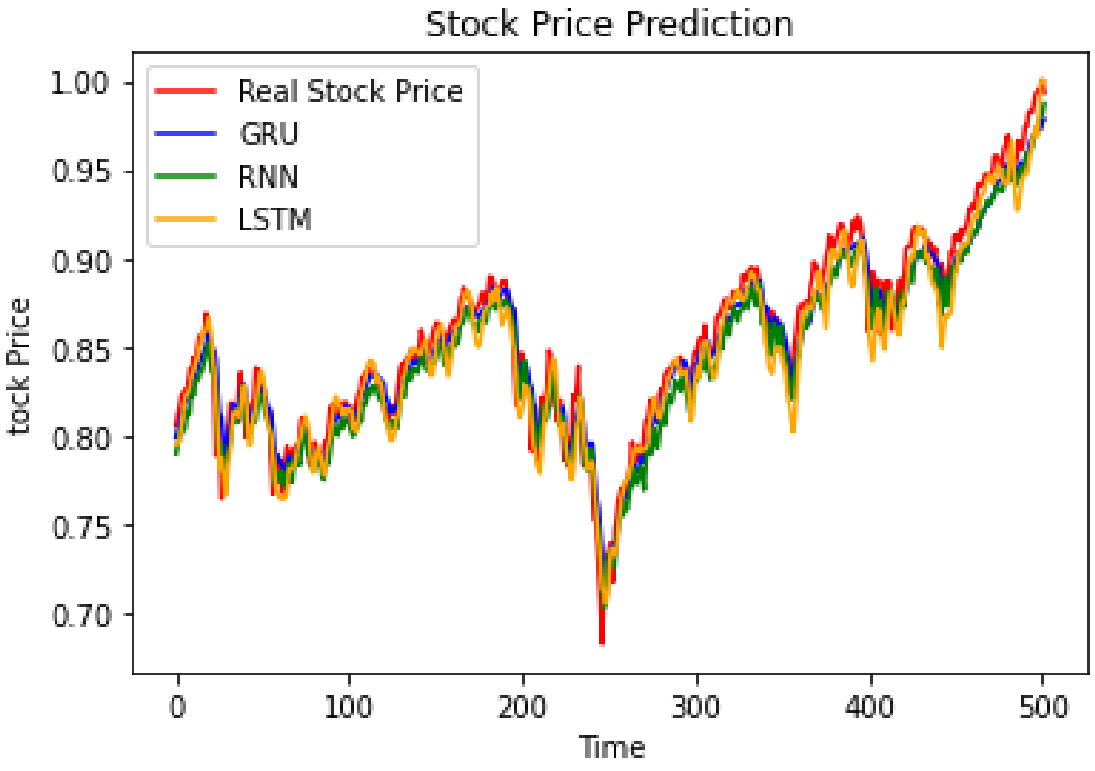
MSE: 8.117422006824418e-05



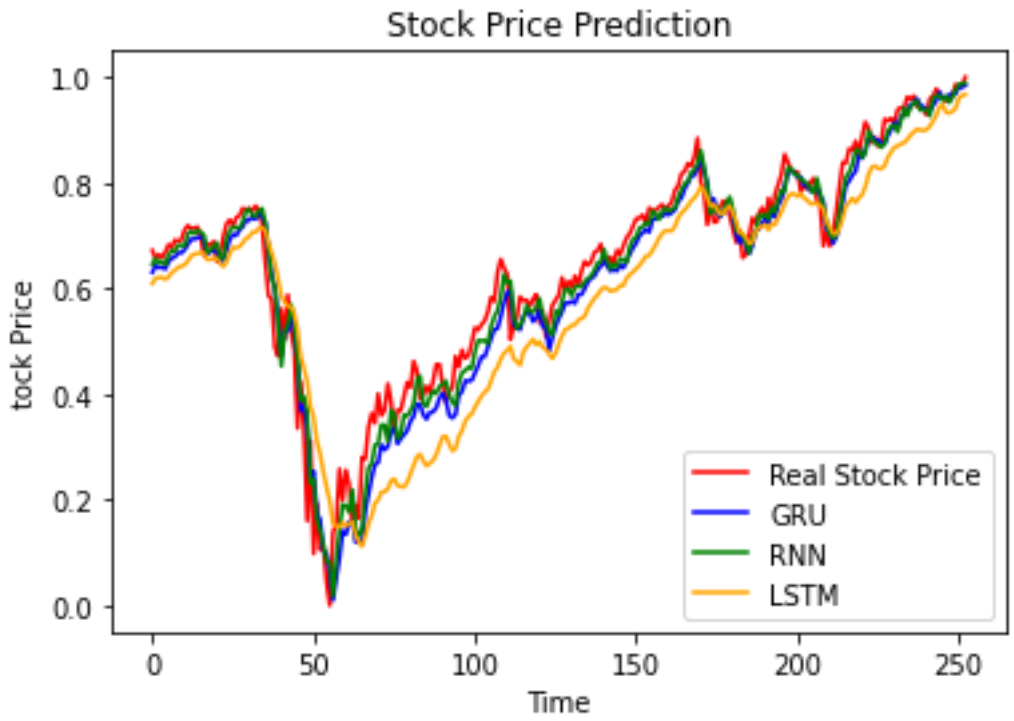


(8) Discuss your findings from (iv) to (vii)?

- MSE: SimpleRNN < GRU < LSTM
- When the real price rises, the predicted price is often smaller than real price.
- When the real price falls, the predicted price is often larger than real price.



(9) Test the model on 2020 data. Does it perform well under COVID-19 conditions? (show results)



- MSE: SimpleRNN < GRU < LSTM
  - 0.0027430117975123214 (GRU)
  - 0.008582557084896431 (LSTM)
  - 0.0018499600307055799 (SimpleRNN)
- Worse than 2019's data

(10) (Bonus 10%) Follow (ix), how to improve the performance of the model? (you can explain your ideas or implement improved models, the latter one get more points.)

- Collect data for 40 days.
- Add an one-hot attribute of weekday
- Add more layers for each model