

## COMP7409 Machine learning in trading and finance - Assignment 1

*Submission: Moodle (Submit your answer by uploading the ipynb files (Jupyter Notebook format). Please note that the ipynb files should already be run on Colab with clear output data/graph. You may also provide explanation of answers by adding comments between the codes)*

*Submission deadline: Nov. 23 th*

### a. Stock Price Prediction (50%)

For this question, you are asked to develop LSTM models to predict stock prices. You should modify the LSTM model described in the lecture, so that it is capable of using the last 10 days of the closing prices of the stock to predict the closing price of the coming 5th day from now (example as below):

To predict day 15, the input data is [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

To predict day 16, the input data is [2, 3, 4, 5, 6, 7, 8, 9, 10, 11]

Regarding the required time series, use the following as the training/testing data sets:

- i. Shanghai Stock Exchange Index (ticker: SSE Composite Index - 000001): 10 years timeline (from the start of 2013 to the end of 2022) for training/testing
- ii. Ethereum price in USD (e.g., ticker: ETH-USD): 5 years timeline (from the start of 2018 to the end of 2022) for training/testing
- iii. An artificial time series with > 1000 data points generated by you (e.g., you may generate a sine curve and for every data point, +/- some random values) for training/testing

For each of (i), (ii), (iii) draw the actual/prediction graph and calculate the accuracy of the prediction.

Your answer should include the following steps:

1. Get the data
2. Prepare the data sets (80% training, and 20% testing)
3. Build and train the LSTM model with the training data set
4. Predict the price for the testing data set
5. Plot the actual/prediction graph and calculate the accuracy of the prediction

Accuracy formula:

$$\Sigma |\text{true value} - \text{predicted value}|$$

b. Reinforcement Learning (50%)

In Lecture 4, you were introduced to a program that plays the OpenAI Gym's Cart Pole game by taking random actions for every step. This assignment challenges you to transfer this program to OpenAI Gym's Pong ('PongDeterministic-v4') game and improve it to win against the AI opponent. In addition to submitting the program, you must also provide **a rendered one-minute video of the trained agent playing the game (accelerate it if needed, mark the timestamp of the highest score in a separate short report file)** and the **rewards evolving plot in each training episode**. You will receive a minimum of 70% if the sample training trajectory shows that the agent can beat the AI opponent at least once, meaning the agent receives a reward of 21. Besides, you will

- i. Earn an additional 20% if your solution's performance (i.e., the total number of training episodes, fewer is better) ranks in the top 25% of all submitted answers
- ii. Earn another 10% if it ranks in the top 10%.