CONNECTION INSTRUCTIONS

- Wi-Fi SSID: GTC_Hands_On
  Password: HandsOnGpu
- 登入 nvlabs.qwiklab.com
  尋找classroom:
  Deep Learning for Finance Trading Strategy
- 請先不要點選任何課程
- 需要任何協助，請詢問助教
- 講義下載點: https://goo.gl/zKuHbr

Algorithmic Trading using Deep Autoencoder based Statistical Arbitrage

Andrew Liu, Ph.D.

Certified Instructor, NVIDIA Deep Learning Institute
Solution Architect
NVIDIA Corporation

DEEP LEARNING INSTITUTE

DLI Mission

Helping people solve challenging problems using AI and deep learning.

- Developers, data scientists and engineers
- Self-driving cars, healthcare and finance
- Training, optimizing, and deploying deep neural networks

TOPICS

- Lab Overview
- Financial Terminology
- A Simple Mean-Reversion Based Algorithmic Trading Example
- Lab
  - Deep Autoencoder
  - Stat-Arb
  - Backtesting
  - Performance Evaluations
  - Next Steps

LAB OVERVIEW

WHAT THIS LAB IS

- An introduction to:
  - Financial Terminology
  - Financial Time Series Data
  - Keras with TensorFlow backend
  - Algorithmic Trading
- Hands-on exercises use Pandas Dataframe to process stock data
- Hands-on exercises using Keras with TensorFlow backend for algorithmic trading
- Complete trading strategy that generates profit and loss curve (P&L)

WHAT THIS LAB IS NOT

- Intro to machine learning from first principles
- Rigorous mathematical formalism of neural networks
- Survey of all the features and options of TensorFlow, Pandas or other tools

ASSUMPTIONS

- You are familiar with Autoencoders
- Helpful to have:
  - Keras experience
  - Python experience

TAKE AWAYS

- Understanding the methods for algorithmic trading
- Ability to setup backtesting and train a autoencoder network
- Enough info to start using Keras with TensorFlow backend to learn from your own data

Financial Terminologies

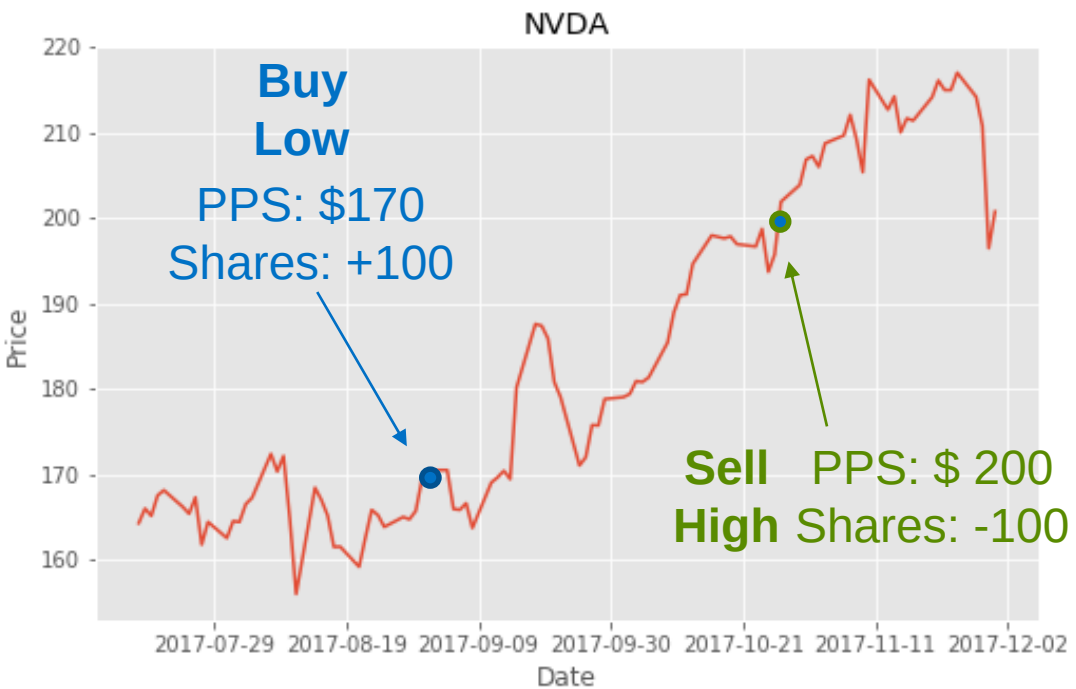
- Algorithmic Trading
- Stock
- Share
- Portfolio
- Long
- Short

Positions

Buy Low & Sell High

Long

Own and Hold -> Sell for profit



Profit: 200*100 – 170*100 ~= 3000

Short

Borrow and Sell -> Buy for Profit

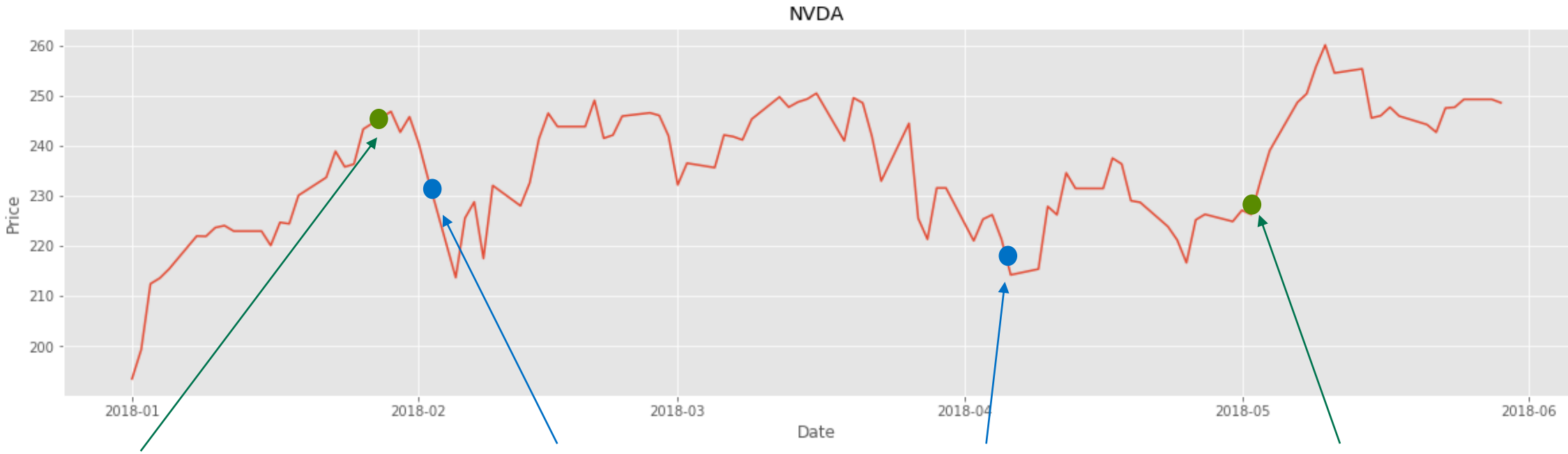
Profit: 29*100 – 26*100 ~= 300

Financial Terminologies

- Return
- Sharpe Ratio
- Mean Reversion
- Paris Trading
- Stat-Arb
  - Ratio of STMA and LTMA
  - PCA
  - Auto Encoder

Mean reversion

220 - 230 - 240



higher enough than average
Open a short position

Close enough to average
Close the short position

Short

Borrow and Sell -> Buy for Profit

lower enough than average
Open a long position

Close enough to average,
Close the long position

Long

Own and Hold -> Sell for profit

Get the data

- !pip install pandas_datareader
- Read in stock data
- Locate through stock symbol index / timestamps / features

		Close	High	Low	Open	Volume
Symbol	**Date**					
NVDA	**2010-01-01**	18.680	18.9500	18.660	18.79	0
	2010-01-04	18.490	18.6200	18.110	18.51	20008817
	2010-01-05	18.760	18.9600	18.420	18.50	18217439
	2010-01-06	18.880	18.9200	18.570	18.68	16229167
	2010-01-07	18.510	18.8600	18.370	18.78	13694746
	2010-01-08	18.550	18.6821	18.250	18.43	11954399

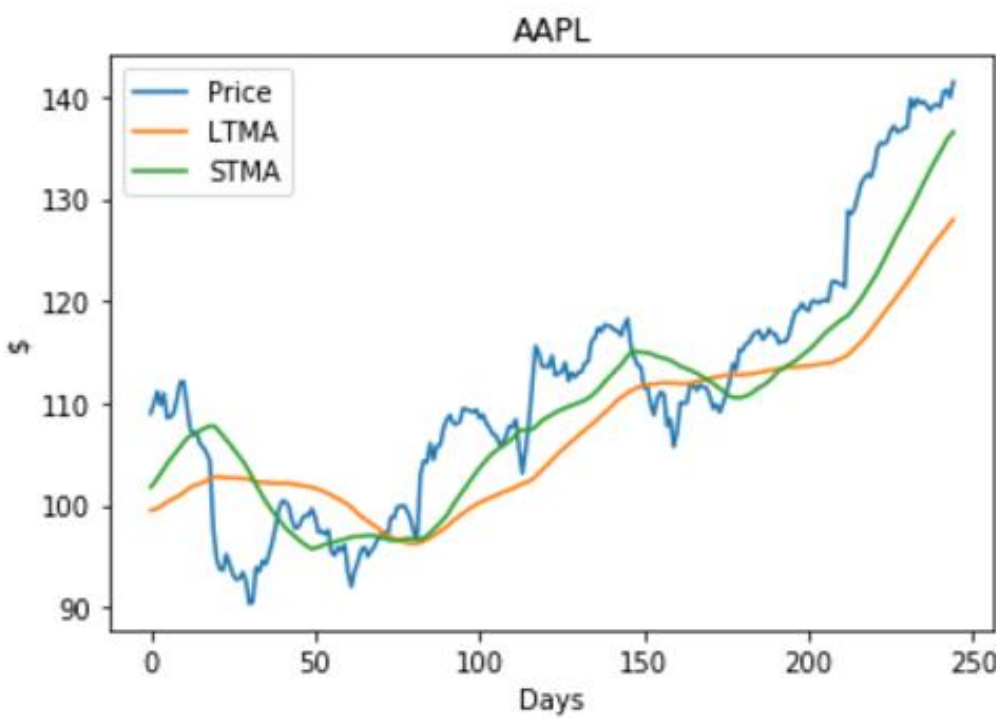
Some datafram operation

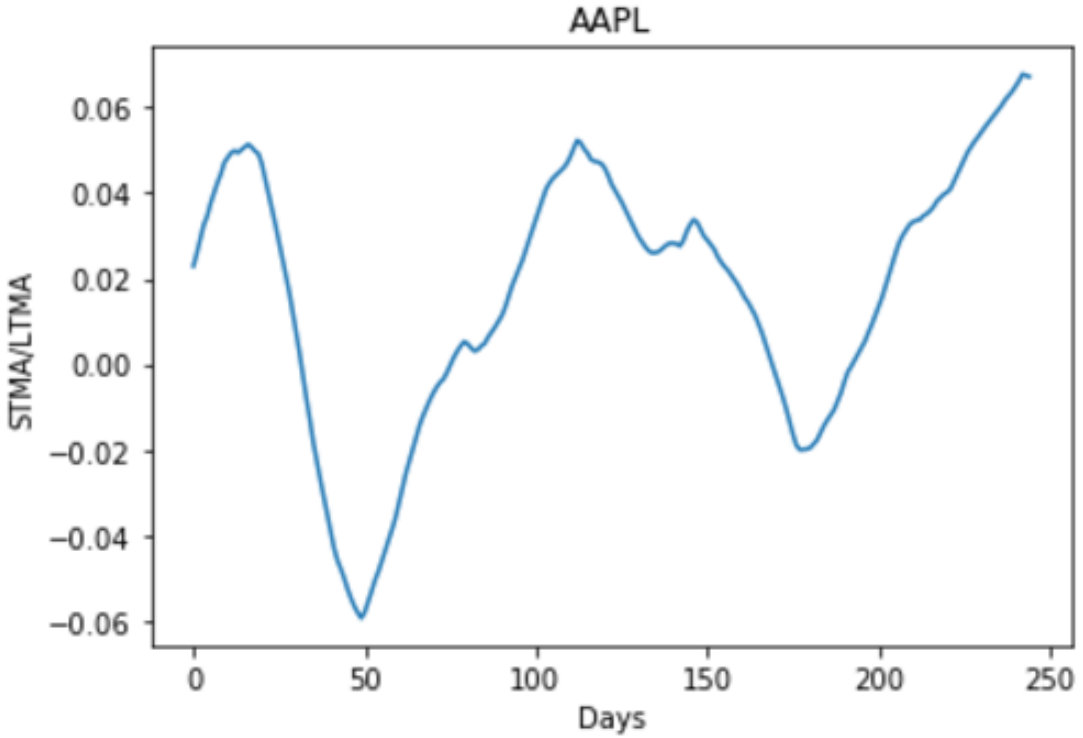
- Plot the stock price dataframe with matplotlib
- Mean Average
- Stock Return

Moving Average – Mean Reversion

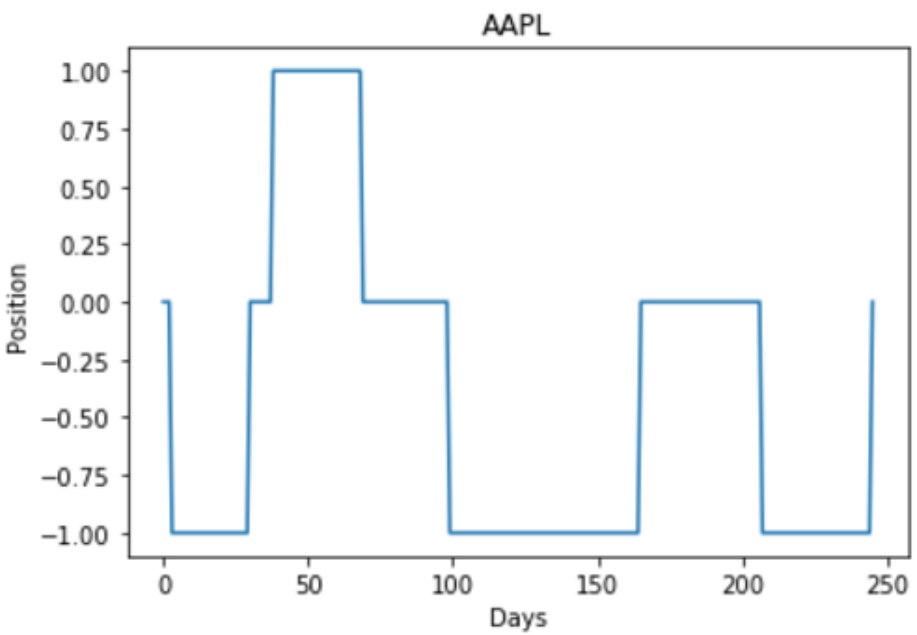
- Apple Inc.’s Stock Return
- Long Term Moving Average
- Short Term Moving Average
- Mean Reversion Signal
- Thresholds for opening and closing positions

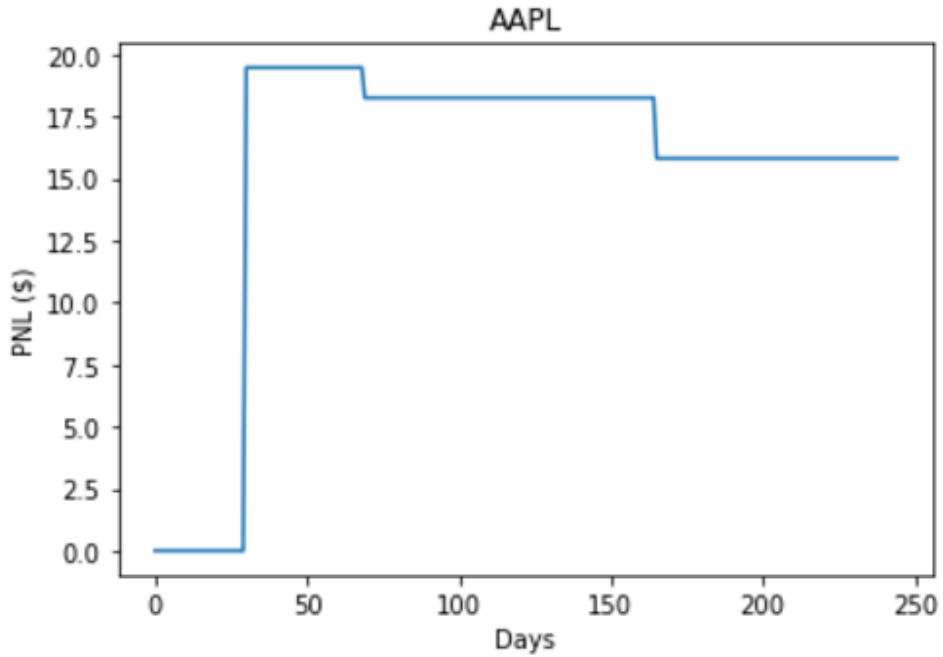
Moving Average - Mean Reversion





Moving Average - Mean Reversion



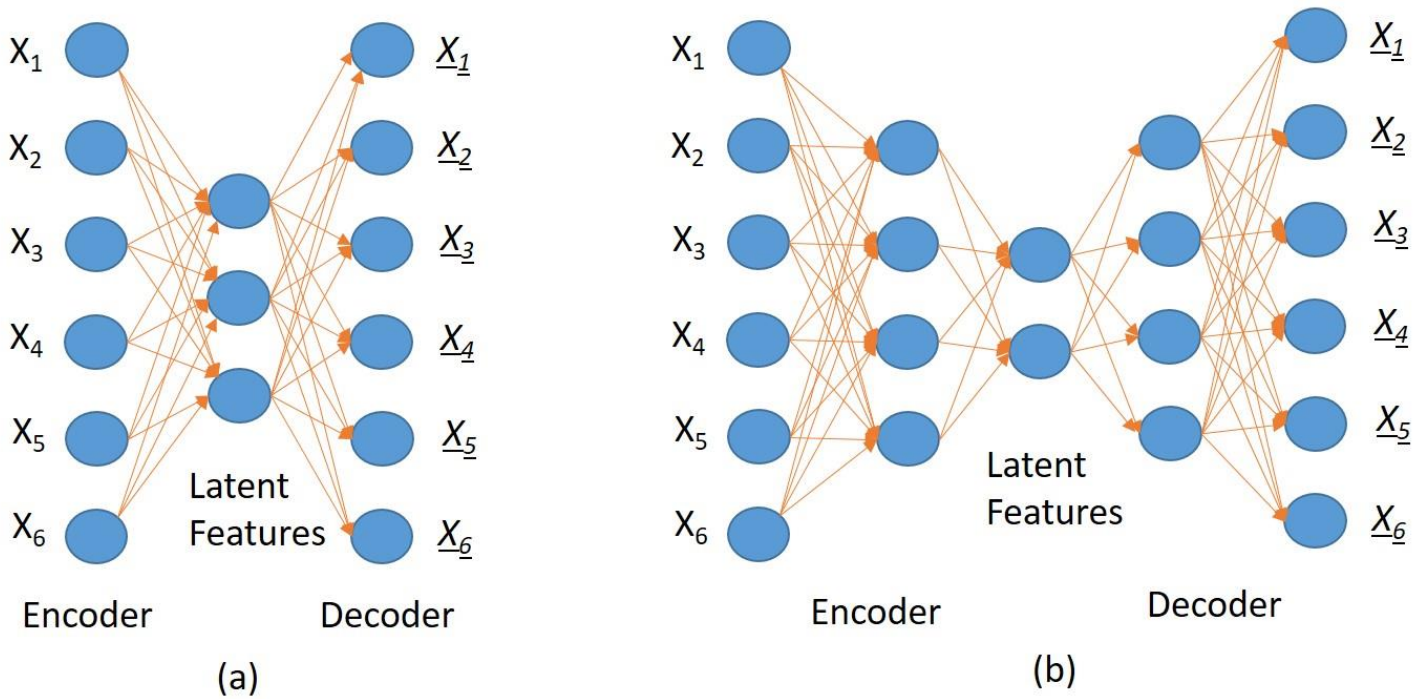


LAB

Deep Autoencoder based Stat-Arb

- Autoencoder
- Backtesting
- Data
- Performance Evaluation
- Hyperparameters

Autoencoder



Dataset

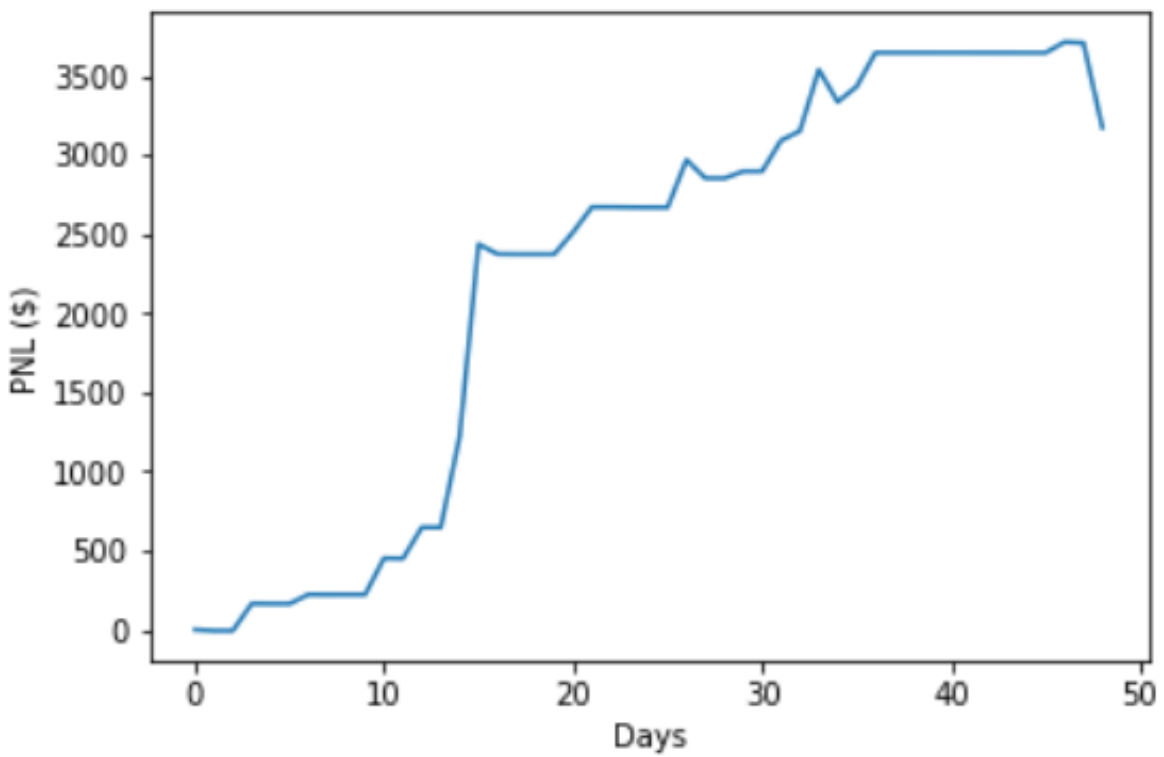
LAB EVALUATION

Hyperparameters

- Set Ts['T1'] to 1.2 in the three above cell. Then re-run everything using Kernel->Restart & Run All.
- If you have enough time, you can try different hyperparameters to see if the Sharpe ratio is improving.
- Is the Sharpe ration you are getting above 3.0?

P&L

- Sharpe Ratio: 4.58



Next Steps

After the lab, we recommend you doing the following options;

- Build a HPC system with multiple GPUs for overnight or real-time hyperparameter search.
- Develop the multi-GPU version of the code.
- Perform hyperparameter search periodically and use the best performing hyperparameters for a period (a day, a week, etc)
- Try deeper autoencoders and see the performance.

深度學習實作坊問券

請掃描右方QR code填寫問券

保留填寫完成畫面於離場
前換取精美禮物

Instructor: Andrew Liu, Ph.D.

www.nvidia.com/dli