Optimizing the Pair-Trading Strategy Using Deep Reinforcement Learning with Stop Loss and Take Profit Boundaries

【Teammate】吳泓緯、陳鴻杰
【Professor】黃俊龍、黃思皓

OUR PROCESS

01 PROBLEM DESCRIPTION

02 PROPOSED SOLUTION

03 SYSTEM DESIGN & IMPLEMENT

04 RESULT ANALYSIS

05 CONCLUSION & CONTRIBUTION

PROBLEM DESCRIPTION

01

- Pair Trading Introduction
- Conventional Method

01 Pair Trading Introduction

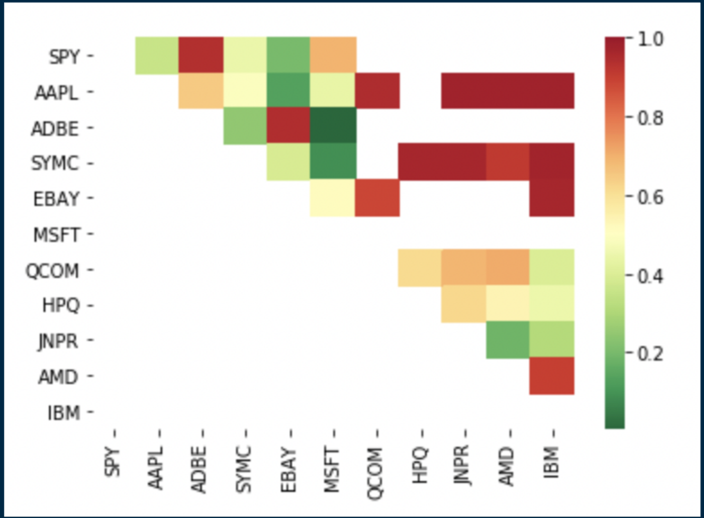
Statistical Arbitrage

+

Market Neutral Strategy

01 Conventional Method

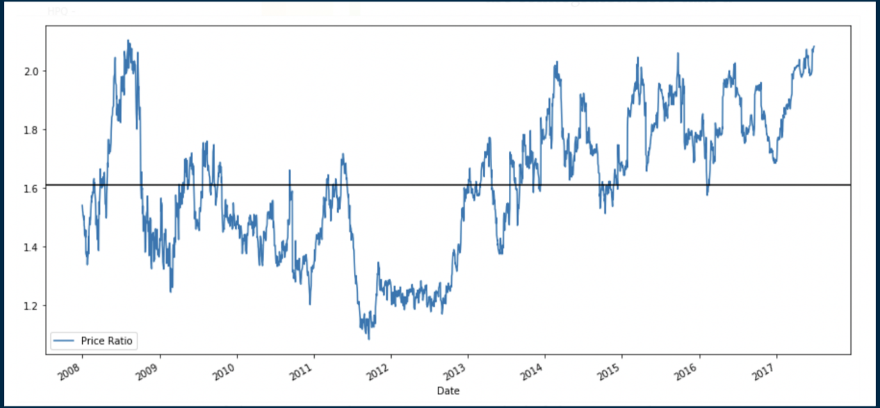
- ➢ select two stocks using cointegration test
- ➢ calculate price ratio of two stocks
- ➢ calculate z-score of price ratio
- ➢ buy and sell two stocks when price ratio diverts from mean



01

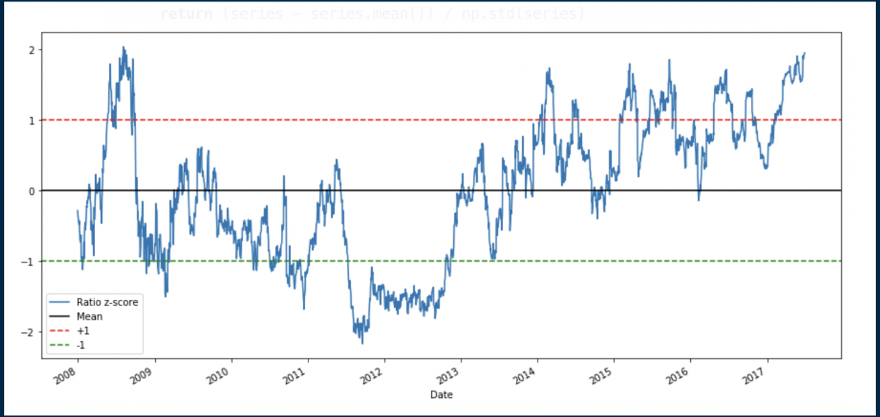
Conventional Method

- select two stocks using cointegration test
- calculate price ratio of two stocks
- calculate z-score of price ratio
- buy and sell two stocks when price ratio diverts from mean



01 Conventional Method

- select two stocks using cointegration test
- calculate price ratio of two stocks
- calculate z-score of price ratio
- buy and sell two stocks when price ratio diverts from mean



01 Conventional Method

- select two stocks using cointegration test
- calculate price ratio of two stocks
- calculate z-score of price ratio
- buy and sell two stocks when price ratio diverts from mean

Drawback

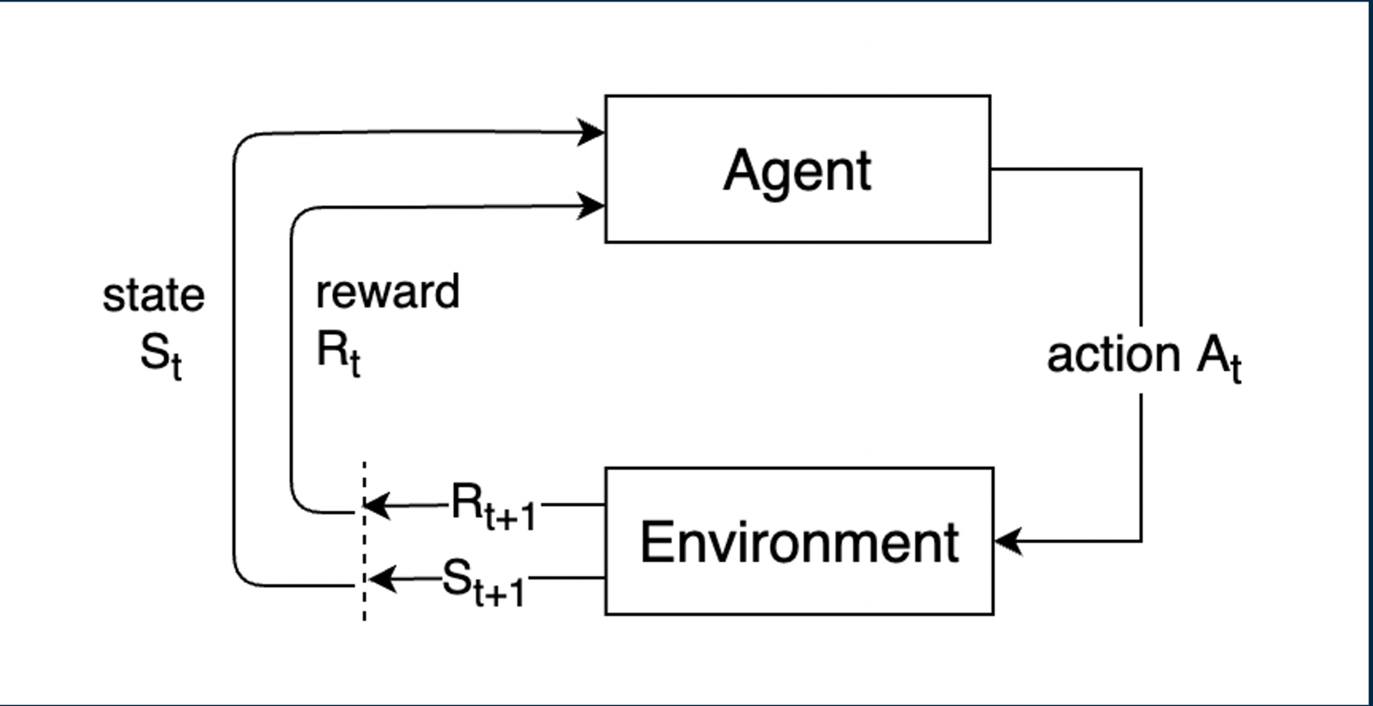
- Future cointegration hypothesis based on historical stock price may not be accurate
- In the conventional method, we assume the stock price ratio has a normal distribution. That may not be the case in the real world

PROPOSED SOLUTION

02

- Reinforcement Learning Introduction
- Double Deep Q-Learning Introduction
- Proposed Method

02 Reinforcement Learning Introduction



02

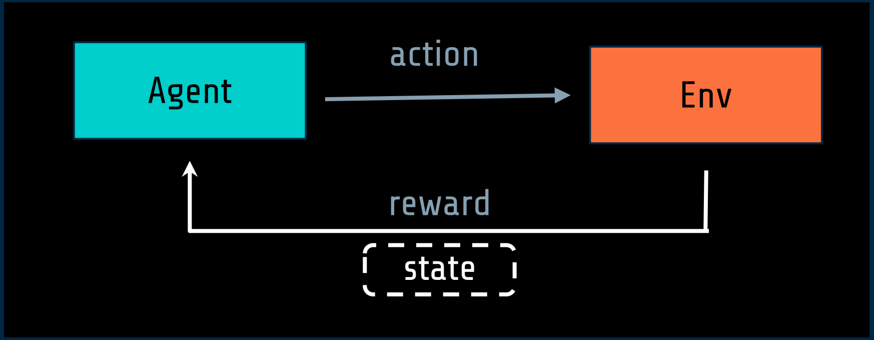
Double Deep Q-Learning Learning Introduction

Algorithm 1 Double Q-Learning

```
Initialize a primary network Q_θ, target network Q_θ', replay buffer D
for each iteration do
    for each environment step do
        Observe state s_t and select a_t ~ π(a_t, s_t)
        Execute a_t and observe next state s_{t+1} and reward r_t = R(S_t, a_t)
        Store (s_t, a_t, r_t, s_{t+1}) in replay buffer D
    end for
end for
for each update step do
    Sample e_t = (s_t, a_t, r_t, s_{t+1}) ~ D
    Compute target Q value:
        Q*(s_t, a_t) ≈ r_t + γQ_θ(s_{t+1}, argmax_a' Q_θ'(s_{t+1}, a')))
    Perform gradient descent step on (Q*(s_t, a_t) − Q_θ(s_t, a_t))^2
end for
Update target network parameters every few episodes:
    θ' ← θ
```

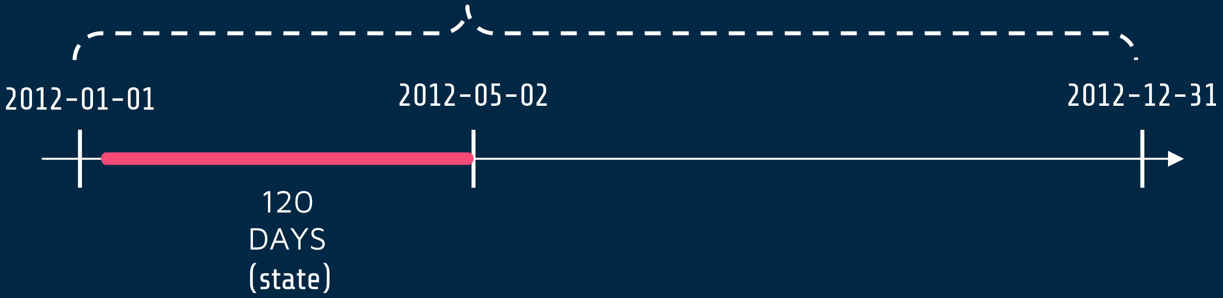
02 Proposed Method

RL MODEL



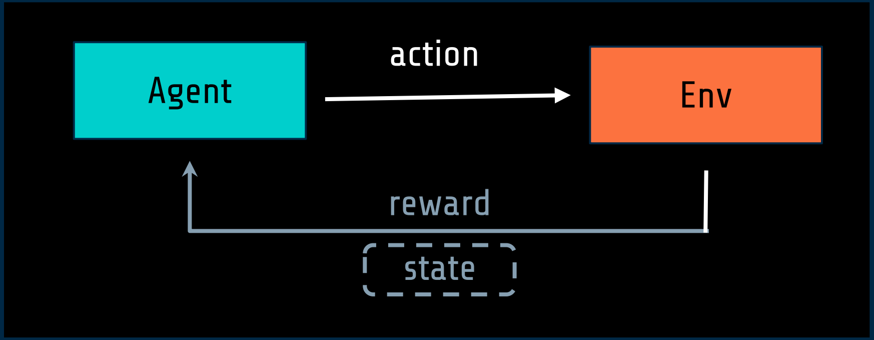
➢ State

- Close price of two stocks in the past 120 days



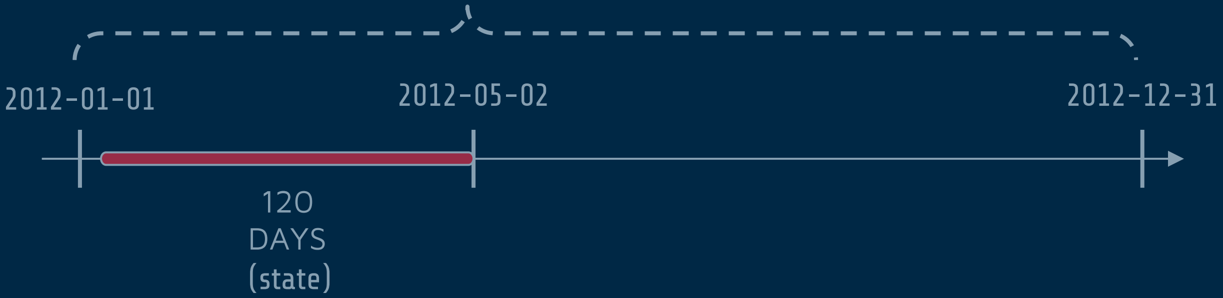
02 Proposed Method

RL MODEL



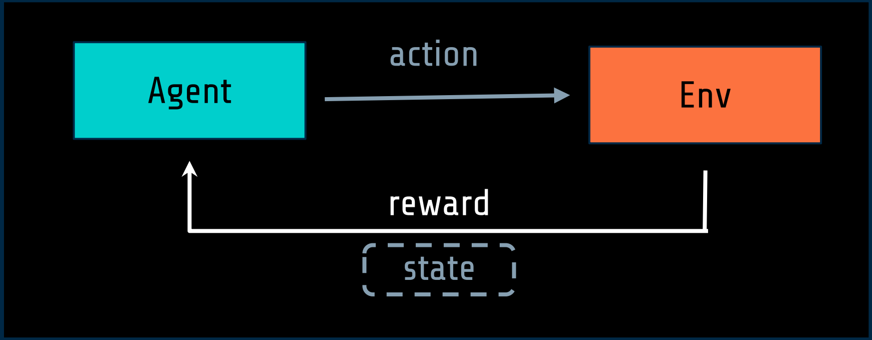
➢ Action

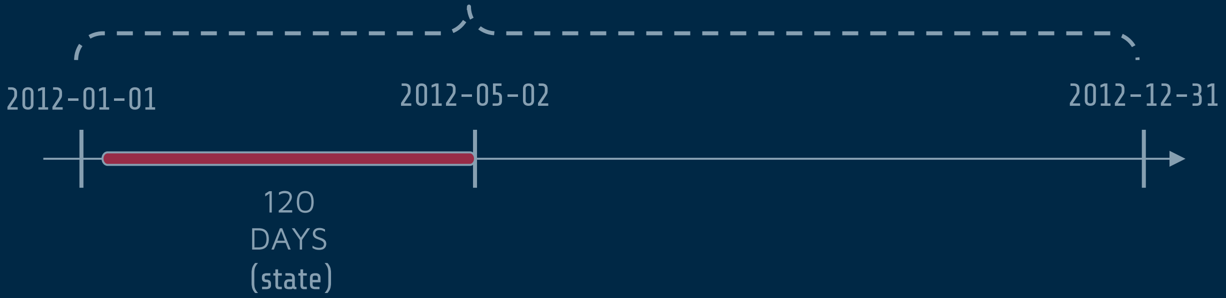
- ❑ Buy 1 unit of high-priced stock & sell n units of low-priced stock
- ❑ Sell 1 unit of high-priced stock & buy n units of low-priced stock
- ❑ No operation



02 Proposed Method

RL MODEL





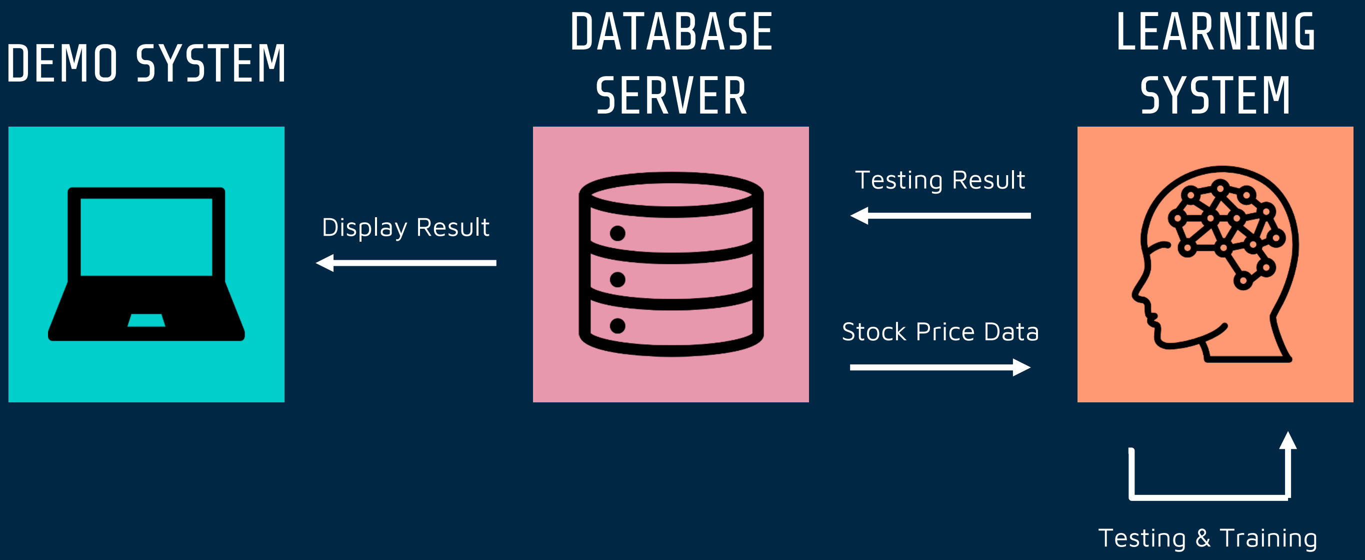
- **Reward**
  - The difference between future and current portfolio value cause by current action

SYSTEM DESIGN & IMPLEMENT

- Overview
- Learning System
- Demo System

03

03 Overview


DEMO SYSTEM
DATABASE SERVER
LEARNING SYSTEM
Display Result
Testing Result
Stock Price Data
Testing & Training

03 Learning System

01 Prepare Date

02 Define Network

03 Pretrain Network

04 Train Network

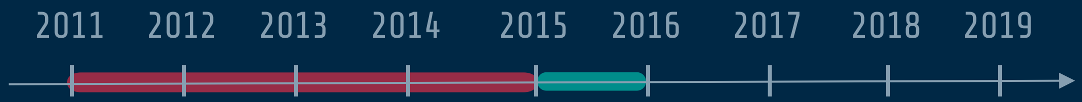
05 Evaluate Network

03 Learning System – Prepare Data

- collect close price of stock from yahoo finance
- select candidate pairs using Engle-Granger cointegration test
- final pairs of stock for pair trading :

Pair	Period
ADBE & MSFT	2011 ~ 2019
GOOG & AMZN	2011 ~ 2019

- 5 years for train, 1 years for test (rolling window)

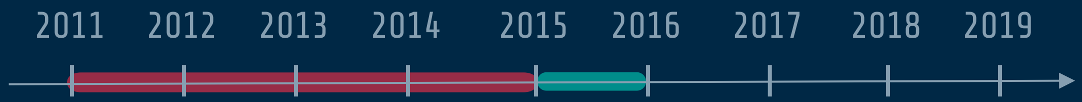


03 Learning System - Prepare Data

- collect close price of stock from yahoo finance
- select candidate pairs using Engle-Granger cointegration test
- final pairs of stock for pair trading :

Pair	Period
ADBE & MSFT	2011 ~ 2019
GOOG & AMZN	2011 ~ 2019

- 5 years for train, 1 years for test (rolling window)

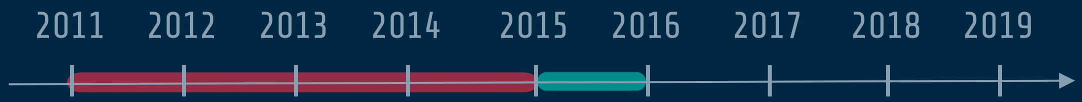


03 Learning System – Prepare Data

- ➢ collect close price of stock from yahoo finance
- ➢ select candidate pairs using Engle-Granger cointegration test
- ➢ final pairs of stock for pair trading :

Pair	Period
ADBE & MSFT	2011 ~ 2019
GOOG & AMZN	2011 ~ 2019

- ➢ 5 years for train, 1 years for test (rolling window)

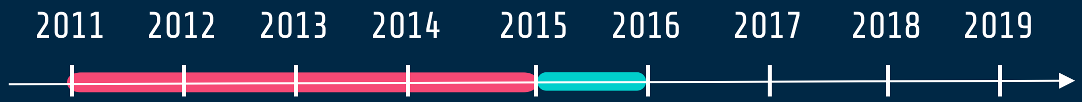


03 Learning System - Prepare Data

- ➢ collect close price of stock from yahoo finance
- ➢ select candidate pairs using Engle-Granger cointegration test
- ➢ final pairs of stock for pair trading :

Pair	Period
ADBE & MSFT	2011 ~ 2019
GOOG & AMZN	2011 ~ 2019

- ➢ 5 years for train, 1 years for test (rolling window)



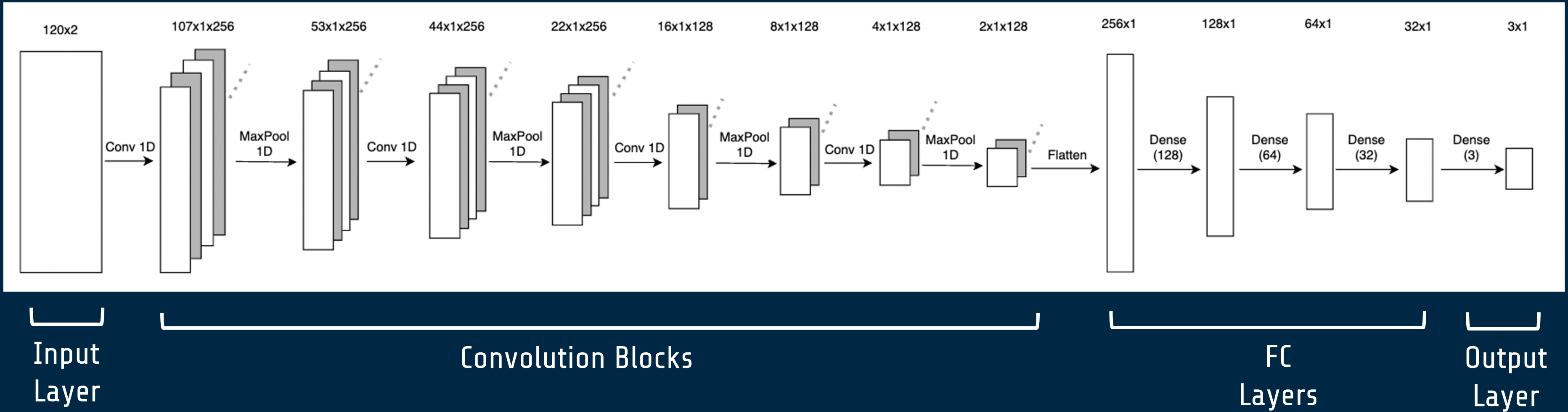
03 Learning System – Prepare Data

- ➢ collect close price of stock from yahoo finance
- ➢ select candidate pairs using Engle-Granger cointegration test
- ➢ final pairs of stock for pair trading :

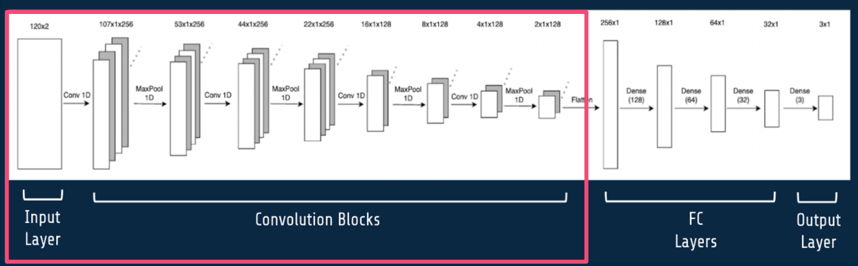
Pair	Period
ADBE & MSFT	2011 ~ 2019
GOOG & AMZN	2011 ~ 2019

- ➢ 5 years for train, 1 years for test (rolling window)

03 Learning System – Define Network

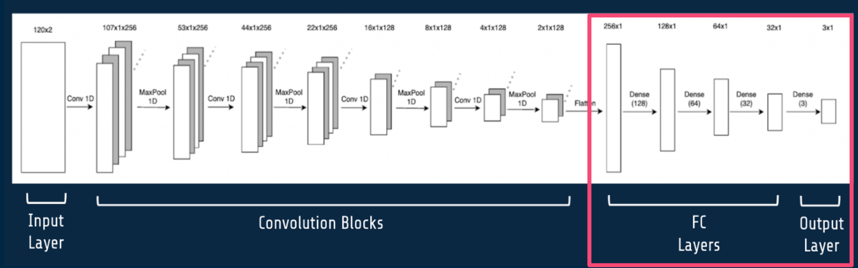


03 Learning System – Define Network



- ➢ input layer
  - ❑ input state (close price of two stocks in the past 120 days)
  - ❑ shape: (120, 2)
- ➢ convolution blocks
  - ❑ composed of Conv1D and MaxPooling1D operation
  - ❑ feature extraction on time series data

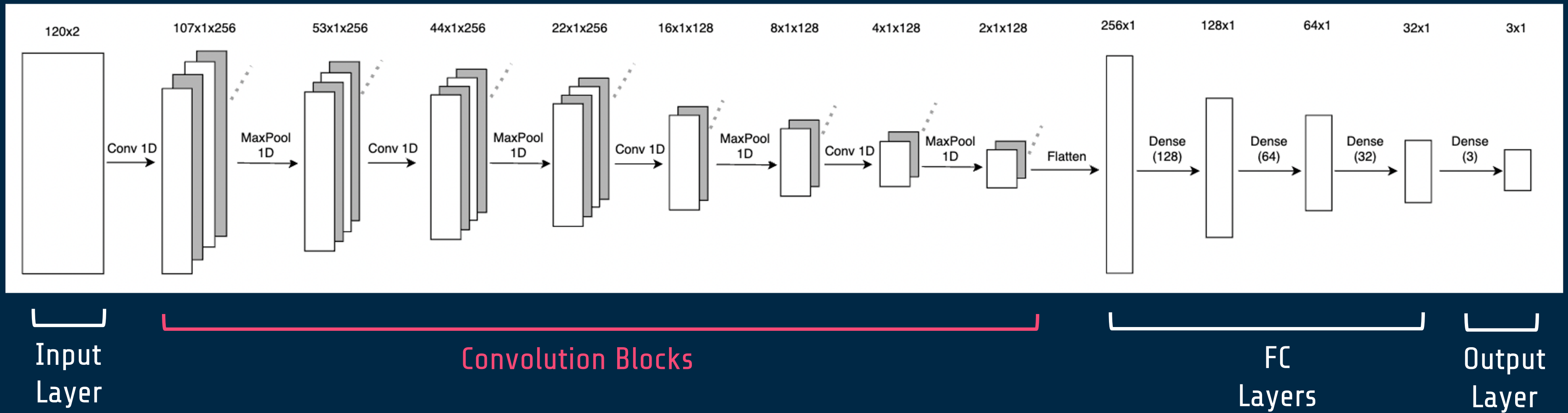
03 Learning System – Define Network



- ➢ fully-connected layers
  - ❑ processing on feature map

- ➢ output layer
  - ❑ output q-value of three actions (long, short and no operation)
  - ❑ shape: (3)

03 Learning System – Pretrain Network

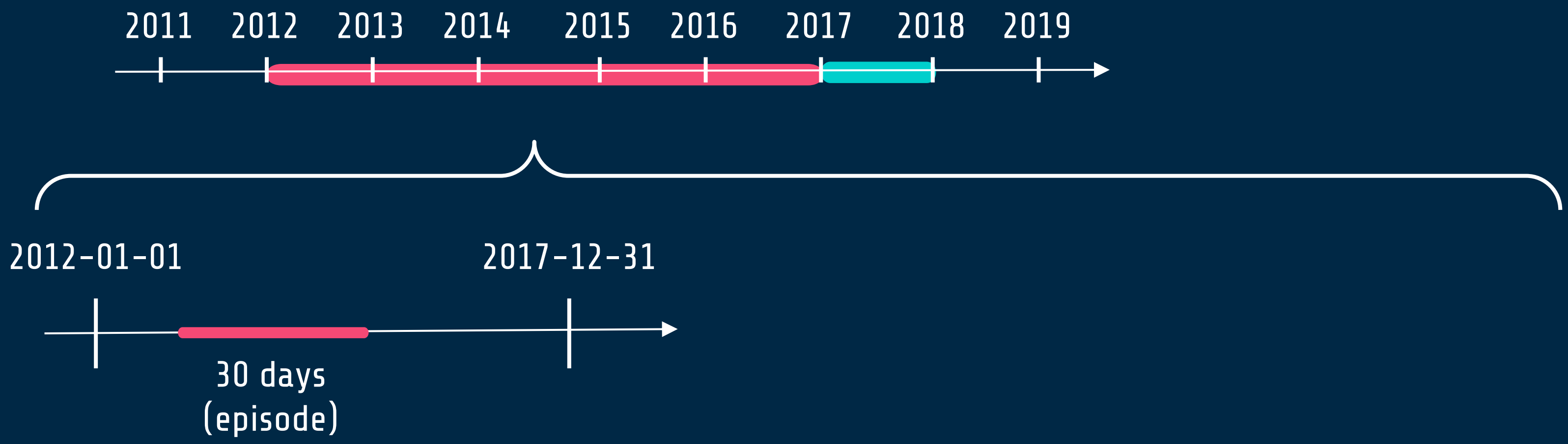


03 Learning System – Pretrain Network

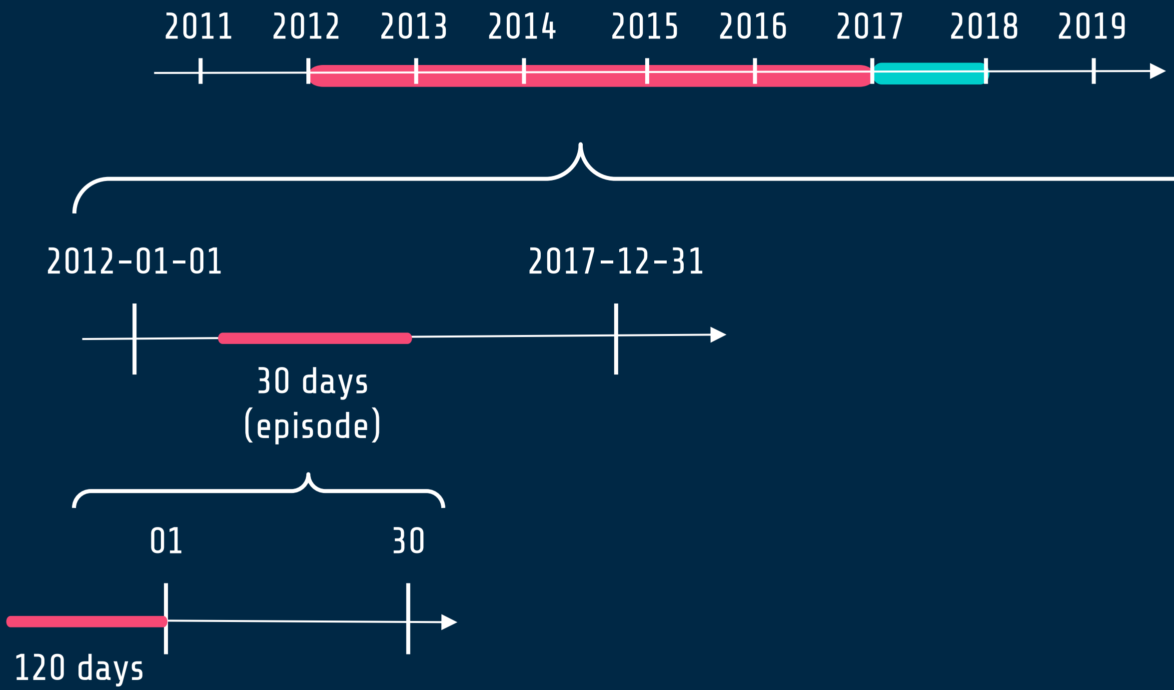
- ➢ pretrain convolution blocks to extract better feature map from input state
- ➢ generate labels based on today's stock price change
  - ❑ x : close price of two stocks in the past 120
  - ❑ y : today's price change more than 5% ⇨ Class A
  - ❑ y : today's price change 0~5% ⇨ Class B
  - ❑ y : today's price change -5~0% ⇨ Class C
  - ❑ y : today's price change more than -5% ⇨ Class D

03 Learning System – Train Network

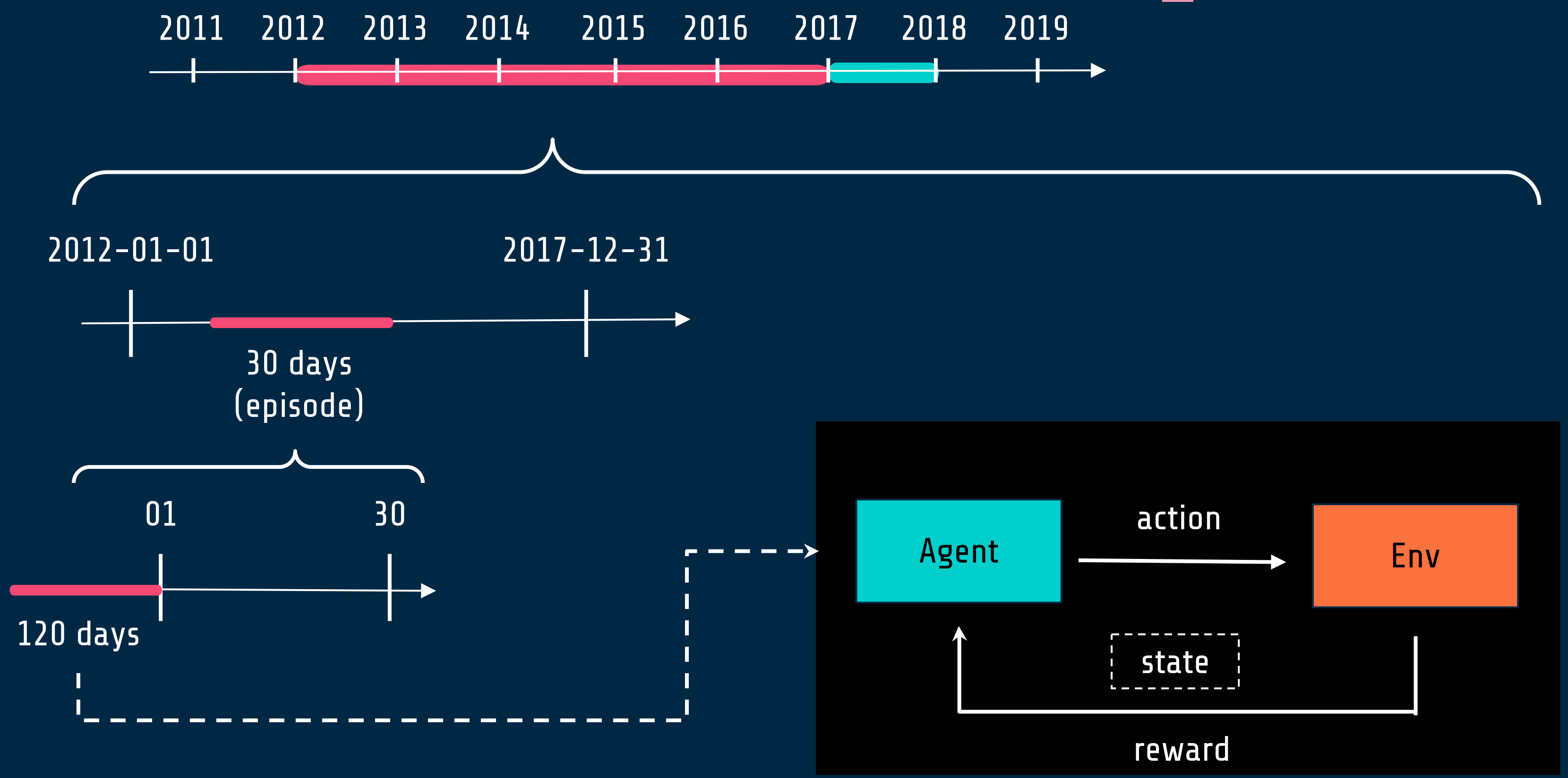
03 Learning System – Train Network



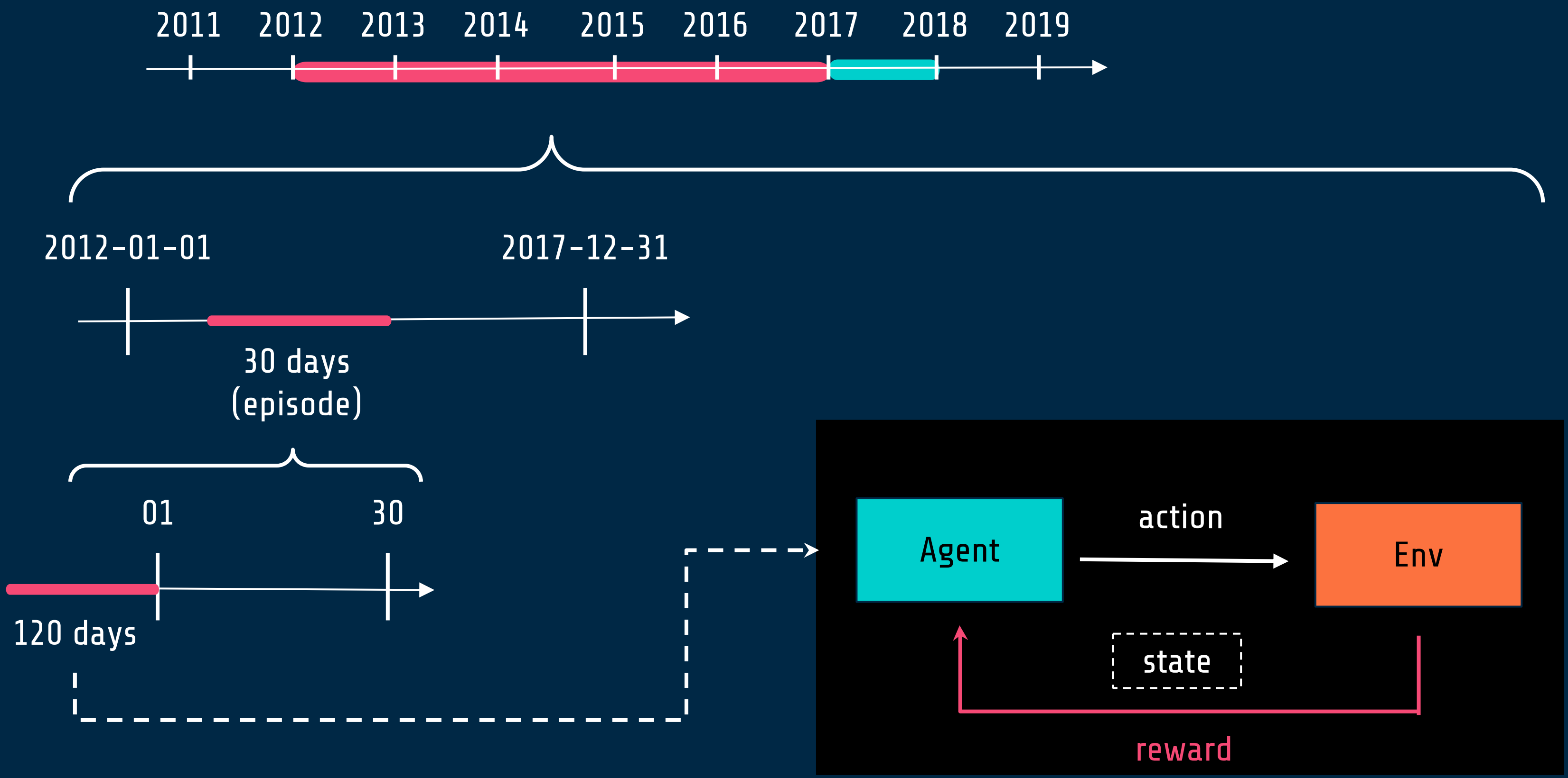
03 Learning System – Train Network



03 Learning System - Train Network



03 Learning System - Train Network

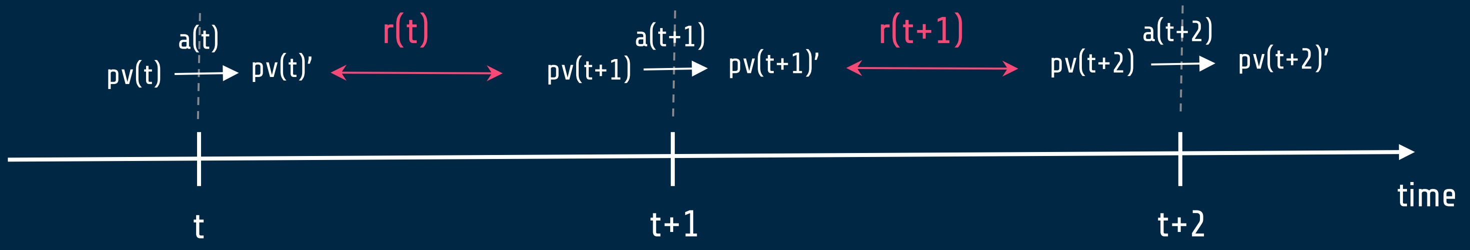


03 Learning System – Train Network

➢ reward function

- ❑ r(a, s) = tomorrow's PV (before action) – current PV (after action)
- ❑ close a position before new opposite position

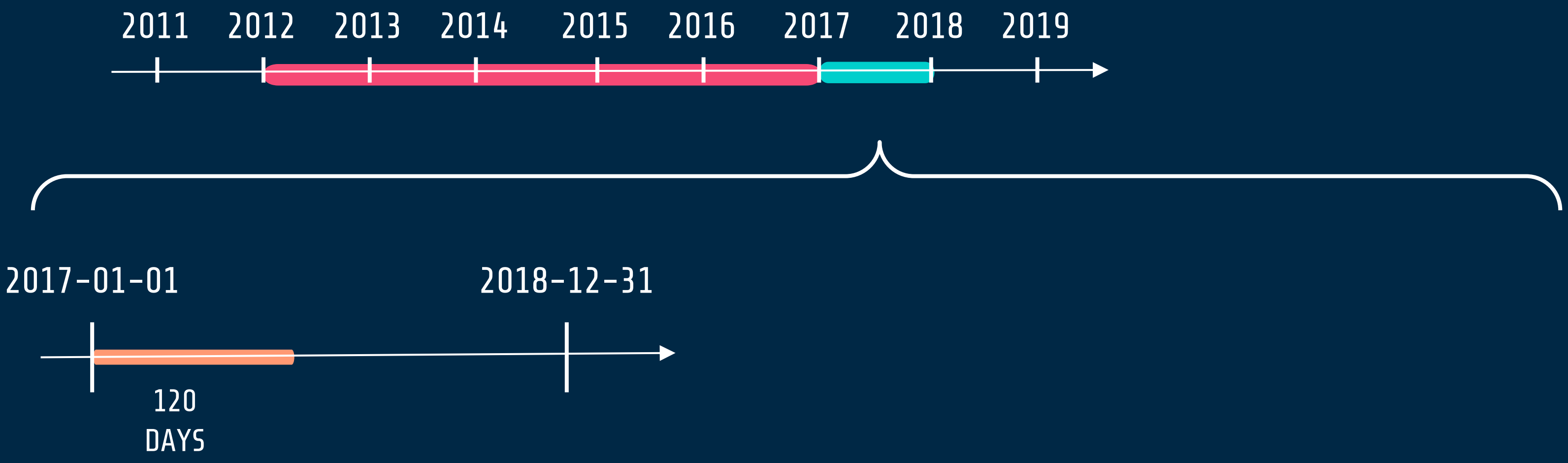
t : time
pv ( ) : portfolio value
a ( ) : action



03

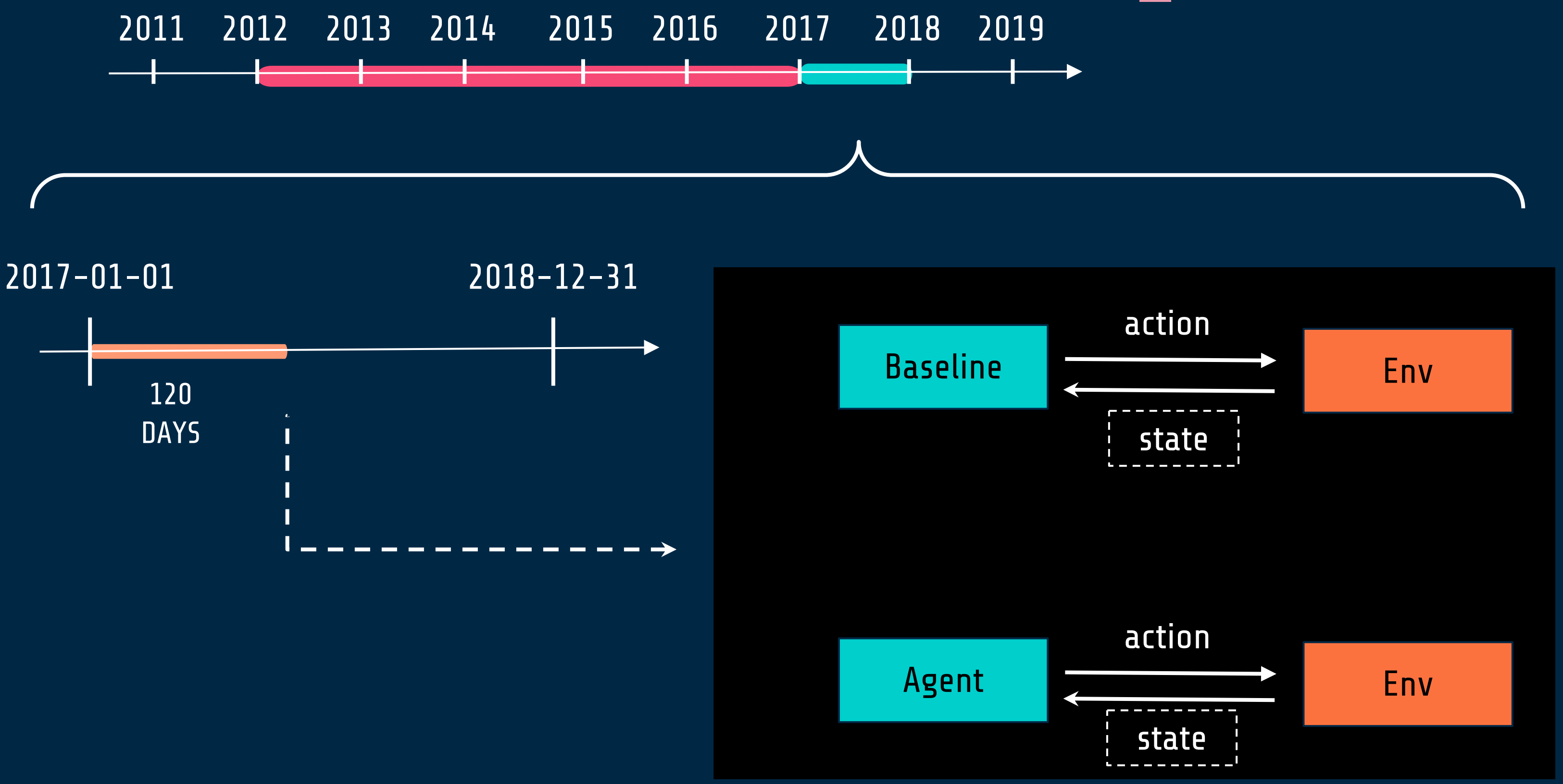
Learning System – Evaluate Network

03 Learning System – Evaluate Network



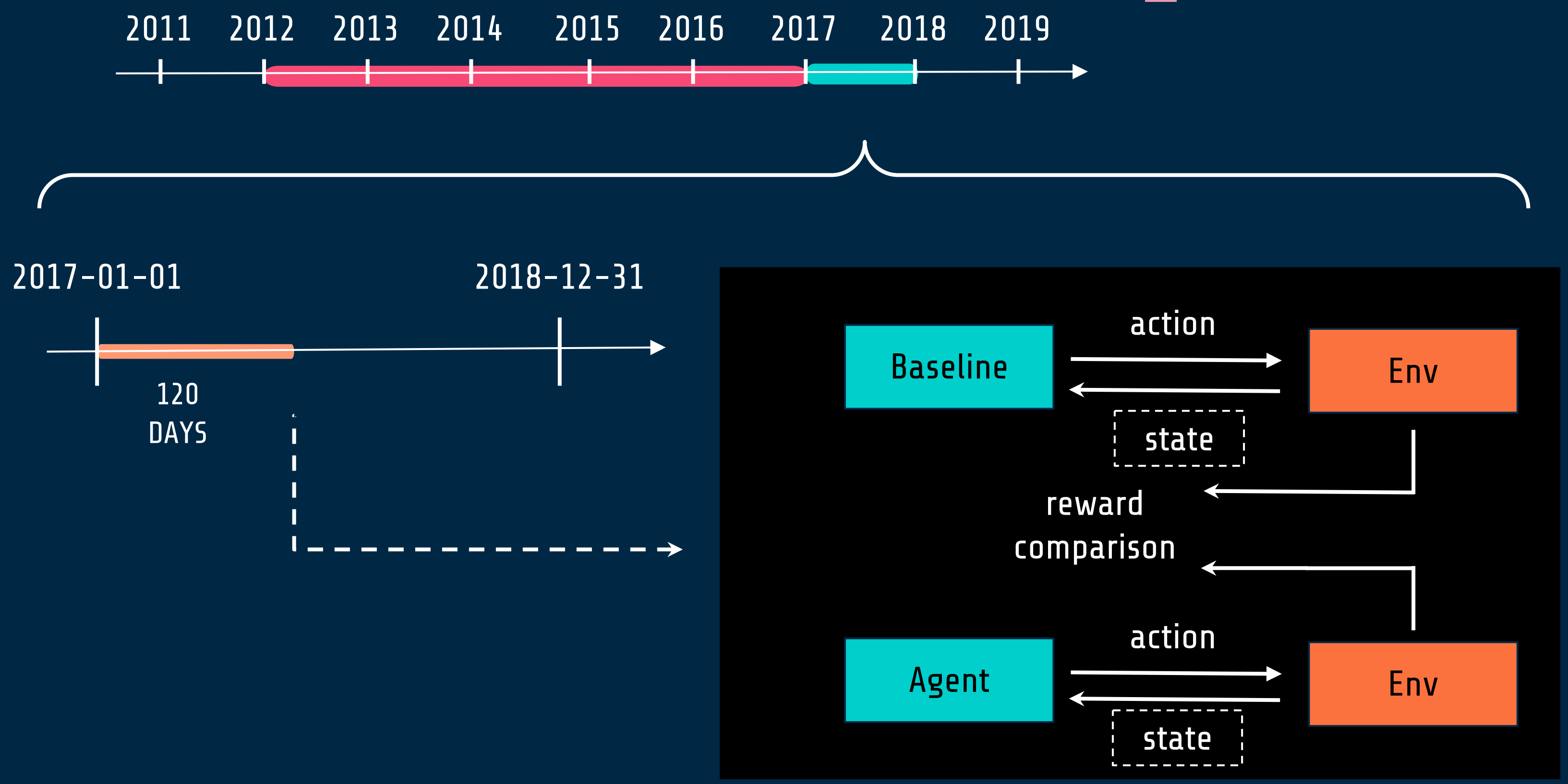
03

Learning System – Evaluate Network

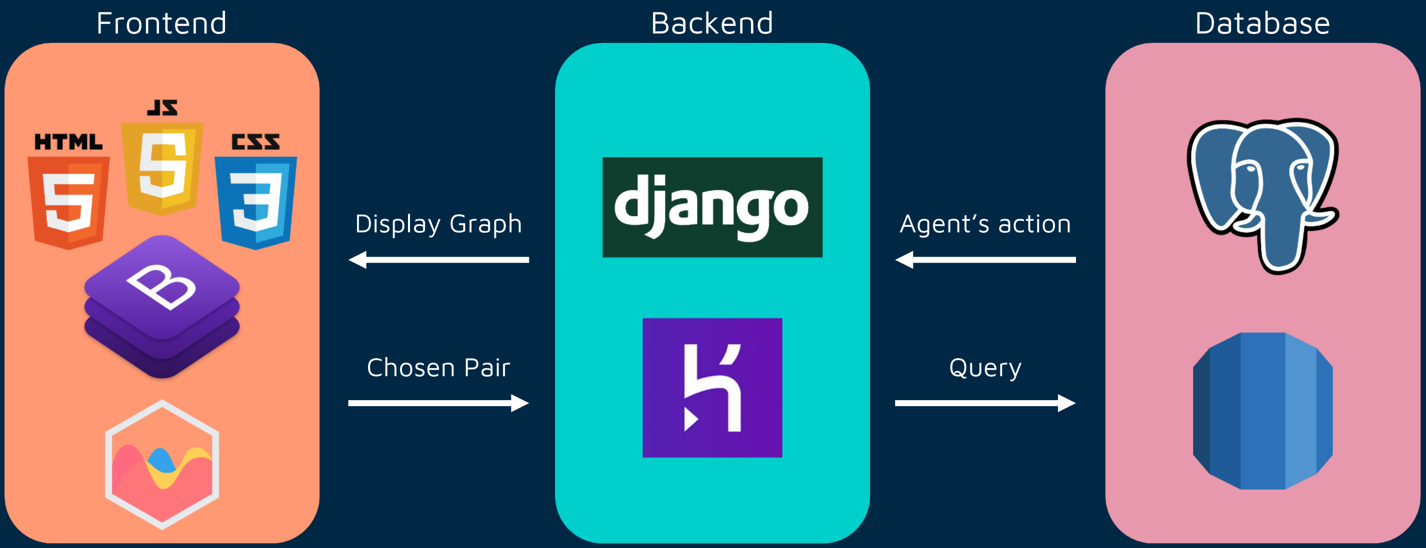


03

Learning System - Evaluate Network



03 Demo System



PTRL WEBAPP

RESULT ANALYSIS

- Overview
- Description

04

04 Overview

	year	p-value	Return				Sharpe Ratio				MDD			
			Normal	Stop Loss (10%)	Take Profit (10%)	Baseline	Normal	Stop Loss (10%)	Take Profit (10%)	Baseline	Normal	Stop Loss (10%)	Take Profit (10%)	Baseline
ADBE & MSFT	2016	0.67	0.21	0.08	0.11	-0.09	1.29	1.39	1.47	-1.57	0.07	0.02	0.02	0.11
	2017	0.16	0.33	0.21	0.18	0.63	2.18	2.03	1.42	1.42	0.01	0.06	0.03	0.15
	2018	0.38	0.21	0.28	0.30	-0.16	0.78	1.03	1.07	0.66	0.04	0.05	0.03	0.72
	2019	0.43	-0.01	-0.05	-0.09	0.56	0.09	-0.02	0.05	-1.01	0.09	0.06	0.35	1.46
	AVG		0.19	0.13	0.13	0.24	1.09	1.11	1.00	-0.13	0.05	0.05	0.11	0.61
GOOG & AMZN	2016	0.33	1.81	1.51	1.69	1.70	2.40	2.16	2.25	1.36	0.04	0.21	0.03	0.92
	2017	0.26	2.64	4.65	3.42	1.53	1.07	1.12	1.40	1.13	0.60	0.81	0.26	0.64
	2018	0.14	0.81	1.23	0.57	4.38	-0.94	1.19	1.63	1.38	1.00	0.39	0.93	2.33
	2019	0.81	0.61	0.58	10.97	2.88	0.76	0.80	1.59	0.22	0.29	0.28	0.27	1.51
	AVG		1.47	1.99	4.16	2.62	0.82	1.32	1.72	1.02	0.48	0.42	0.37	1.35

04 Overview

	year	p-value	Return				Sharpe Ratio				MDD			
			Normal	Stop Loss (10%)	Take Profit (10%)	Baseline	Normal	Stop Loss (10%)	Take Profit (10%)	Baseline	Normal	Stop Loss (10%)	Take Profit (10%)	Baseline
ADBE & MSFT	2016	0.67	0.21	0.08	0.11	-0.09	1.29	1.39	1.47	-1.57	0.07	0.02	0.02	0.11
	2017	0.16	0.33	0.21	0.18	0.63	2.18	2.03	1.42	1.42	0.01	0.06	0.03	0.15
	2018	0.38	0.21	0.28	0.30	-0.16	0.78	1.03	1.07	0.66	0.04	0.05	0.03	0.72
	2019	0.43	-0.01	-0.05	-0.09	0.56	0.09	-0.02	0.05	-1.01	0.09	0.06	0.35	1.46
	AVG		0.19	0.13	0.13	0.24	1.09	1.11	1.00	-0.13	0.05	0.05	0.11	0.61
GOOG & AMZN	2016	0.33	1.81	1.51	1.69	1.70	2.40	2.16	2.25	1.36	0.04	0.21	0.03	0.92
	2017	0.26	2.64	4.65	3.42	1.53	1.07	1.12	1.40	1.13	0.60	0.81	0.26	0.64
	2018	0.14	0.81	1.23	0.57	4.38	-0.94	1.19	1.63	1.38	1.00	0.39	0.93	2.33
	2019	0.81	0.61	0.58	10.97	2.88	0.76	0.80	1.59	0.22	0.29	0.28	0.27	1.51
	AVG		1.47	1.99	4.16	2.62	0.82	1.32	1.72	1.02	0.48	0.42	0.37	1.35

- ➢ A high cointegration relationship p-value will result in the agent getting a higher return

04 Overview

	year	p-value	Return				Sharpe Ratio				MDD			
			Normal	Stop Loss (10%)	Take Profit (10%)	Baseline	Normal	Stop Loss (10%)	Take Profit (10%)	Baseline	Normal	Stop Loss (10%)	Take Profit (10%)	Baseline
ADBE & MSFT	2016	0.67	0.21	0.08	0.11	-0.09	1.29	1.39	1.47	-1.57	0.07	0.02	0.02	0.11
	2017	0.16	0.33	0.21	0.18	0.63	2.18	2.03	1.42	1.42	0.01	0.06	0.03	0.15
	2018	0.38	0.21	0.28	0.30	-0.16	0.78	1.03	1.07	0.66	0.04	0.05	0.03	0.72
	2019	0.43	-0.01	-0.05	-0.09	0.56	0.09	-0.02	0.05	-1.01	0.09	0.06	0.35	1.46
	AVG		0.19	0.13	0.13	0.24	1.09	1.11	1.00	-0.13	0.05	0.05	0.11	0.61
GOOG & AMZN	2016	0.33	1.81	1.51	1.69	1.70	2.40	2.16	2.25	1.36	0.04	0.21	0.03	0.92
	2017	0.26	2.64	4.65	3.42	1.53	1.07	1.12	1.40	1.13	0.60	0.81	0.26	0.64
	2018	0.14	0.81	1.23	0.57	4.38	-0.94	1.19	1.63	1.38	1.00	0.39	0.93	2.33
	2019	0.81	0.61	0.58	10.97	2.88	0.76	0.80	1.59	0.22	0.29	0.28	0.27	1.51
	AVG		1.47	1.99	4.16	2.62	0.82	1.32	1.72	1.02	0.48	0.42	0.37	1.35

- The agent will often perform better in terms of SR and MDD no matter if the cointegration relationship has a high or low p-value

04 Overview

	year	p-value	Return				Sharpe Ratio				MDD			
			Normal	Stop Loss (10%)	Take Profit (10%)	Baseline	Normal	Stop Loss (10%)	Take Profit (10%)	Baseline	Normal	Stop Loss (10%)	Take Profit (10%)	Baseline
ADBE & MSFT	2016	0.67	0.21	0.08	0.11	-0.09	1.29	1.39	1.47	-1.57	0.07	0.02	0.02	0.11
	2017	0.16	0.33	0.21	0.18	0.63	2.18	2.03	1.42	1.42	0.01	0.06	0.03	0.15
	2018	0.38	0.21	0.28	0.30	-0.16	0.78	1.03	1.07	0.66	0.04	0.05	0.03	0.72
	2019	0.43	-0.01	-0.05	-0.09	0.56	0.09	-0.02	0.05	-1.01	0.09	0.06	0.35	1.46
	AVG		0.19	0.13	0.13	0.24	1.09	1.11	1.00	-0.13	0.05	0.05	0.11	0.61
GOOG & AMZN	2016	0.33	1.81	1.51	1.69	1.70	2.40	2.16	2.25	1.36	0.04	0.21	0.03	0.92
	2017	0.26	2.64	4.65	3.42	1.53	1.07	1.12	1.40	1.13	0.60	0.81	0.26	0.64
	2018	0.14	0.81	1.23	0.57	4.38	-0.94	1.19	1.63	1.38	1.00	0.39	0.93	2.33
	2019	0.81	0.61	0.58	10.97	2.88	0.76	0.80	1.59	0.22	0.29	0.28	0.27	1.51
	AVG		1.47	1.99	4.16	2.62	0.82	1.32	1.72	1.02	0.48	0.42	0.37	1.35

➢ Fluctuation during testing period decreases when stop loss and take profit mechanism were added during the training period.

CONCLUSION & CONTRIBUTION

05

- Conclusion & Contribution
- Future Work

05 Conclusion & Contribution

- Reward function designed to make profit with Double DQN algorithm
- Agent will perform higher SR and lower MDD than the baseline even if cointegration between two stocks is low
- Agent's performance on SR and MDD is enhanced when stop loss and take profit boundaries is added in training process
- Conservativeness of agent will largely impacted by the amount of trading quantity

05 Future Work

- ➢ More realistic trading environment
  - ❑ Transaction cost & Slippage effect
  - ❑ News information & sentiment
  - ❑ Market Index & Indicator

- ➢ Other reinforcement learning algorithm
  - ❑ Advantage-Actor-Critic (A2C)

Reference

- Gatev, E., Goetzmann, W. N., & Rouwenhorst, K. G. (2006). Pairs trading: Performance of a relative-value arbitrage rule. The Review of Financial Studies, 19(3), 797-827
- Van Hasselt, H., Guez, A., & Silver, D. (2016, March). Deep reinforcement learning with double q-learning. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 30, No. 1)
- Brim, A. (2020, January). Deep Reinforcement Learning Pairs Trading with a Double Deep Q-Network. In 2020 10th Annual Computing and Communication Workshop and Conference (CCWC) (pp. 0222-0227). IEEE
- Kim, T., & Kim, H. Y. (2019). Optimizing the pairs-trading strategy using deep reinforcement learning with trading and stop-loss boundaries. Complexity, 2019.
- Brim, A. (2019). Deep Reinforcement Learning Pairs Trading
- Fallahpour, Saeid, et al. "Pairs trading strategy optimization using the reinforcement learning method: a cointegration approach." Soft Computing 20.12 (2016): 5051-5066

THANK YOU FOR LISTENING