

<참고자료>

#### 베이스라인 코드 : 퀀티랩 Github  
<https://github.com/quantylab/rltrader>

<In-Out Data 정리>

※ Input

1. 종목코드.csv

(1) 차트 데이터 (v2.1)

column = ['date', 'open', 'high', 'low', 'close', 'volume']

(2) 차트 데이터 + 투자지표 (v2.2)

column = ['date', 'open', 'high', 'low', 'close', 'volume', 'value', 'stock\_volume',  
'stock\_value', 'eps', 'per', 'bps', 'pbr', 'dividend\_per\_stock', 'dividend\_yield\_ratio',  
'volume\_inst\_buy', 'volume\_inst\_sell', 'volume\_fore\_buy', 'volume\_fore\_sell',  
'value\_inst\_buy', 'value\_inst\_sell', 'value\_fore\_buy', 'value\_fore\_sell', 'kospi\_close',  
'kospi\_open', 'kospi\_high', 'kospi\_low']

(재무지표 추가여부)

2. argparse 통해 입력하는 파라미터

--stock\_code: 강화학습의 환경이 될 주식의 종목 코드입니다. A3C의 경우 여러 개의 종목  
코드를 입력('+')합니다.

--ver: RLTrader의 버전을 명시합니다. (default='v2')

--rl\_method: 강화학습 방식을 설정합니다. (default='a2c')

--net: 가치 신경망과 정책 신경망에서 사용할 신경망 유형을 선택합니다. (default='lstm')

--num\_steps: lstm과 cnn에서 사용할 Step 크기를 정합니다. 이 크기만큼 자질 벡터의 크  
기가 확장됩니다. (default='5')

--lr: 학습 속도를 정합니다. (default='0.001')

--discount\_factor: 할인율을 정합니다. (default='0.9')

--start\_epsilon: 시작 탐험률을 정합니다. 에포크가 수행되면서 탐험률은 감소합니다.  
(default='1')

--balance: 주식투자 시뮬레이션을 위한 초기 자본금을 설정합니다. (default='10000000')

--num\_epochs: 수행할 에포크 수를 지정합니다. (default='100')

--delayed\_reward\_threshold: 지연 보상의 임계값을 정합니다. (default='0.05')

--backend: Keras의 백엔드로 사용할 프레임워크를 설정합니다. (default='tensorflow')

--output\_name: 로그, 가시화 파일, 신경망 모델 등의 출력 파일을 저장할 폴더의 이름  
(default=utils.get\_time\_str())

--value\_network\_name: 신경망 모델 재사용 시, 가치 신경망 모델 파일명 입력

--policy\_network\_name: 신경망 모델 재사용 시, 정책 신경망 모델 파일명 입력

--reuse\_models: 신경망 모델 재사용 유무

--learning: 강화학습 유무  
--start\_date: 차트 데이터 및 학습 데이터 시작 날짜  
--end\_date: 차트 데이터 및 학습 데이터 끝 날짜

### 3. 코드에서 직접 수정해야 하는 파라미터

#### (1) 피쳐 관련

- 전처리 여부 결정 후에 csv 파일 또는 코드에 피쳐 추가/삭제

#### (2) 신경망 모듈

- 신경망 레이어 차원, activation, dropout 확률, optimizer, loss 등

#### (3) 강화학습 모듈

- 보상 : get\_batch() 안에서 r 계산 공식 변경

- 미니 배치 학습 여부

#### (4) 에이전트 모듈

- 수수료, 세금 : TRADING\_CHARGE, TRADING\_TAX

- 최대/최소 단일 거래 단위 : max\_trading\_unit/min\_trading\_unit

- 행동 결정 로직 : decide\_action() 안에서 변경

※ cf) 학습 데이터 (신경망 input)

전처리 통해 생성 (차트 데이터, 투자지표, 보조지표) + 에이전트 상태 (주식 보유 비율 (ratio\_hold), 포트폴리오 가치 비율(ratio\_portfolio\_value))

cf) 추가로 고려해볼만한 상태 : 연속 매수/매도한 횟수, 20 거래일 동안의 매수/매도 비율 등

```
['value', 'stock_value', 'stock_volume', 'eps', 'per', 'bps', 'pbr', 'dividend_per_stock',  
'dividend_yield_ratio', 'volume_inst_buy', 'volume_inst_sell', 'volume_fore_buy',  
'volume_fore_sell', 'value_inst_buy', 'value_inst_sell', 'value_fore_buy',  
'value_fore_sell', 'kospi_close', 'kospi_open', 'kospi_high', 'kospi_low', 'ma_5',  
'ma_10', 'ma_15', 'ma_20', 'ma_30', 'ma_60', 'ma_120', 'ema_5', 'ema_10', 'ema_15',  
'ema_20', 'ema_30', 'ema_60', 'ema_120', 'wma_5', 'wma_10', 'wma_15', 'wma_20',  
'wma_30', 'wma_60', 'wma_120', 'ma_v5', 'ma_v10', 'ma_v20', 'ma_v60', 'ma_v120',  
'volume_inst_sub', 'volume_fore_sub', 'ma_inst_v5', 'ma_inst_v10', 'ma_inst_v20',  
'ma_inst_v60', 'ma_inst_v120', 'ma_fore_v5', 'ma_fore_v10', 'ma_fore_v20',  
'ma_fore_v60', 'ma_fore_v120', 'ubb', 'mbb', 'lbb', 'macd', 'macdsignal9',  
'macdhist', 'rsi', 'slowk', 'slowd', 'fastk', 'fastd', 'fastk_rsi', 'fastd_rsi', 'cci', 'willR',  
'sar', 'adx', 'plus_di', 'plus_dm', 'atr', 'obv', 'var', 'ratio_hold',  
'ratio_portfolio_value']
```

83

※ cf) 신경망 output

정책신경망 출력 : 매수/매도 시 PV를 높일 확률

가치신경망 출력 : 행동에 대한 예측 가치(손익률)

### Output

#### (1) 파라미터 기록

- 입력받은 파라미터(인자)를 params.json 파일로 저장
- 파이썬 json 모듈의 loads(), dumps() 함수로 json 문자열에서 딕셔너리로, 딕셔너리에서 json 문자열로 변환 가능

## (2) 로그 기록

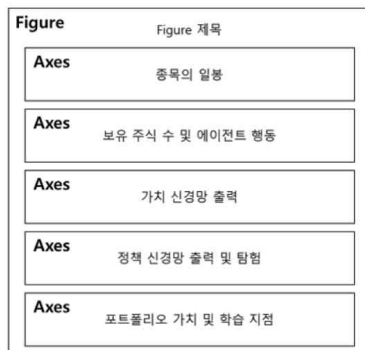
<stock\_code\_name>.log

## (3) 신경망 가중치 학습

학습한 가치 신경망 <value\_network\_name>.h5

학습한 정책 신경망 <policy\_network\_name>.h5

## (4) 시각화



## ※ 실행 예시

```
train 인자 : --stock_code 035420 --rl_method a2c --net lstm --learning
--num_steps 5 --output_name c_035420 --num_epochs 100 --lr 0.001
--start_epsilon 1 --discount_factor 0.9 --start_date 20180101 --end_date 20200910
```

```
test 인자 : --stock_code 035420 --rl_method a2c --net lstm --num_steps 5
--output_name c_035420 --num_epochs 1 --lr 0.001 --start_epsilon 0
--discount_factor 0.9 --value_network_name a2c_lstm_value_c_035420
--policy_network_name a2c_lstm_policy_c_035420 --reuse_models --start_date
20200911 --end_date 20200914
```