```
(pid=24563)         'agent_1': np.ndarray((2,), dtype=int64, min=5.0, max=5.0,
mean=5.0)},
(pid=24563)    2: { 'agent_0': np.ndarray((2,), dtype=int64, min=12.0, max=12.0,
mean=12.0),
(pid=24563)         'agent_1': np.ndarray((2,), dtype=int64, min=12.0, max=12.0,
mean=12.0)},
(pid=24563)    3: { 'agent_0': np.ndarray((2,), dtype=int64, min=2.0, max=2.0,
mean=2.0),
(pid=24563)         'agent_1': np.ndarray((2,), dtype=int64, min=2.0, max=2.0,
mean=2.0)},
(pid=24563)    4: { 'agent_0': np.ndarray((2,), dtype=int64, min=1.0, max=1.0,
mean=1.0),
(pid=24563)         'agent_1': np.ndarray((2,), dtype=int64, min=1.0, max=1.0,
mean=1.0)},
(pid=24563)    5: { 'agent_0': np.ndarray((2,), dtype=int64, min=1.0, max=1.0,
mean=1.0),
(pid=24563)         'agent_1': np.ndarray((2,), dtype=int64, min=1.0, max=1.0,
mean=1.0)},
(pid=24563)    6: { 'agent_0': np.ndarray((2,), dtype=int64, min=13.0, max=13.0,
mean=13.0),
(pid=24563)         'agent_1': np.ndarray((2,), dtype=int64, min=13.0, max=13.0,
mean=13.0)},
(pid=24563)    7: { 'agent_0': np.ndarray((2,), dtype=int64, min=13.0, max=13.0,
mean=13.0),
(pid=24563)         'agent_1': np.ndarray((2,), dtype=int64, min=13.0, max=13.0,
mean=13.0)}}
(pid=24563) 2019-10-05 13:25:04,162        INFO sampler.py:305 -- Info return from
env: { 0: {'agent_0': {}, 'agent_1': {}},
(pid=24563)    1: {'agent_0': {}, 'agent_1': {}},
(pid=24563)    2: {'agent_0': {}, 'agent_1': {}},
(pid=24563)    3: {'agent_0': {}, 'agent_1': {}},
(pid=24563)    4: {'agent_0': {}, 'agent_1': {}},
(pid=24563)    5: {'agent_0': {}, 'agent_1': {}},
(pid=24563)    6: {'agent_0': {}, 'agent_1': {}},
(pid=24563)    7: {'agent_0': {}, 'agent_1': {}}}
(pid=24563) 2019-10-05 13:25:04,163        INFO sampler.py:403 -- Preprocessed
obs: np.ndarray((2,), dtype=int64, min=11.0, max=11.0, mean=11.0)
(pid=24563) 2019-10-05 13:25:04,163        INFO sampler.py:407 -- Filtered obs:
np.ndarray((2,), dtype=float64, min=0.0, max=0.0, mean=0.0)
(pid=24563) 2019-10-05 13:25:04,169        INFO sampler.py:521 -- Inputs to
compute_actions():
(pid=24563)
(pid=24563) { 'agent_0': [ { 'data': { 'agent_id': 'agent_0',
(pid=24563)                            'env_id': 0,
(pid=24563)                            'info': {},
(pid=24563)                            'obs': np.ndarray((2,), dtype=float64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                            'prev_reward': 0.0,
(pid=24563)                            'rnn_state': []},
(pid=24563)                  'type': 'PolicyEvalData'},
(pid=24563)                { 'data': { 'agent_id': 'agent_0',
(pid=24563)                            'env_id': 1,
(pid=24563)                            'info': {},
(pid=24563)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.707, max=-0.707, mean=-0.707),
(pid=24563)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
```

```
(pid=24563)                                 'prev_reward': 0.0,
(pid=24563)                                 'rnn_state': []},
(pid=24563)                   'type': 'PolicyEvalData'},
(pid=24563)                { 'data': { 'agent_id': 'agent_0',
(pid=24563)                            'env_id': 2,
(pid=24563)                            'info': {},
(pid=24563)                            'obs': np.ndarray((2,), dtype=float64,
min=0.704, max=0.704, mean=0.704),
(pid=24563)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                            'prev_reward': 0.0,
(pid=24563)                            'rnn_state': []},
(pid=24563)                   'type': 'PolicyEvalData'},
(pid=24563)                { 'data': { 'agent_id': 'agent_0',
(pid=24563)                            'env_id': 3,
(pid=24563)                            'info': {},
(pid=24563)                            'obs': np.ndarray((2,), dtype=float64,
min=-1.147, max=-1.147, mean=-1.147),
(pid=24563)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                            'prev_reward': 0.0,
(pid=24563)                            'rnn_state': []},
(pid=24563)                   'type': 'PolicyEvalData'},
(pid=24563)                { 'data': { 'agent_id': 'agent_0',
(pid=24563)                            'env_id': 4,
(pid=24563)                            'info': {},
(pid=24563)                            'obs': np.ndarray((2,), dtype=float64,
min=-1.026, max=-1.026, mean=-1.026),
(pid=24563)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                            'prev_reward': 0.0,
(pid=24563)                            'rnn_state': []},
(pid=24563)                   'type': 'PolicyEvalData'},
(pid=24563)                { 'data': { 'agent_id': 'agent_0',
(pid=24563)                            'env_id': 5,
(pid=24563)                            'info': {},
(pid=24563)                            'obs': np.ndarray((2,), dtype=float64,
min=-0.866, max=-0.866, mean=-0.866),
(pid=24563)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                            'prev_reward': 0.0,
(pid=24563)                            'rnn_state': []},
(pid=24563)                   'type': 'PolicyEvalData'},
(pid=24563)                { 'data': { 'agent_id': 'agent_0',
(pid=24563)                            'env_id': 6,
(pid=24563)                            'info': {},
(pid=24563)                            'obs': np.ndarray((2,), dtype=float64,
min=1.214, max=1.214, mean=1.214),
(pid=24563)                            'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                            'prev_reward': 0.0,
(pid=24563)                            'rnn_state': []},
(pid=24563)                   'type': 'PolicyEvalData'},
(pid=24563)                { 'data': { 'agent_id': 'agent_0',
(pid=24563)                            'env_id': 7,
(pid=24563)                            'info': {},
(pid=24563)                            'obs': np.ndarray((2,), dtype=float64,
min=1.041, max=1.041, mean=1.041),
(pid=24563)                            'prev_action': np.ndarray((), dtype=int64,
```

```
min=0.0, max=0.0, mean=0.0),
(pid=24563)                                 'prev_reward': 0.0,
(pid=24563)                                 'rnn_state': []},
(pid=24563)                       'type': 'PolicyEvalData'}],
(pid=24563)    'agent_1': [ { 'data': { 'agent_id': 'agent_1',
(pid=24563)                                 'env_id': 0,
(pid=24563)                                 'info': {},
(pid=24563)                                 'obs': np.ndarray((2,), dtype=float64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                                 'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                                 'prev_reward': 0.0,
(pid=24563)                                 'rnn_state': []},
(pid=24563)                       'type': 'PolicyEvalData'},
(pid=24563)                     { 'data': { 'agent_id': 'agent_1',
(pid=24563)                                 'env_id': 1,
(pid=24563)                                 'info': {},
(pid=24563)                                 'obs': np.ndarray((2,), dtype=float64,
min=-0.707, max=-0.707, mean=-0.707),
(pid=24563)                                 'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                                 'prev_reward': 0.0,
(pid=24563)                                 'rnn_state': []},
(pid=24563)                       'type': 'PolicyEvalData'},
(pid=24563)                     { 'data': { 'agent_id': 'agent_1',
(pid=24563)                                 'env_id': 2,
(pid=24563)                                 'info': {},
(pid=24563)                                 'obs': np.ndarray((2,), dtype=float64,
min=0.704, max=0.704, mean=0.704),
(pid=24563)                                 'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                                 'prev_reward': 0.0,
(pid=24563)                                 'rnn_state': []},
(pid=24563)                       'type': 'PolicyEvalData'},
(pid=24563)                     { 'data': { 'agent_id': 'agent_1',
(pid=24563)                                 'env_id': 3,
(pid=24563)                                 'info': {},
(pid=24563)                                 'obs': np.ndarray((2,), dtype=float64,
min=-1.147, max=-1.147, mean=-1.147),
(pid=24563)                                 'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                                 'prev_reward': 0.0,
(pid=24563)                                 'rnn_state': []},
(pid=24563)                       'type': 'PolicyEvalData'},
(pid=24563)                     { 'data': { 'agent_id': 'agent_1',
(pid=24563)                                 'env_id': 4,
(pid=24563)                                 'info': {},
(pid=24563)                                 'obs': np.ndarray((2,), dtype=float64,
min=-1.026, max=-1.026, mean=-1.026),
(pid=24563)                                 'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                                 'prev_reward': 0.0,
(pid=24563)                                 'rnn_state': []},
(pid=24563)                       'type': 'PolicyEvalData'},
(pid=24563)                     { 'data': { 'agent_id': 'agent_1',
(pid=24563)                                 'env_id': 5,
(pid=24563)                                 'info': {},
(pid=24563)                                 'obs': np.ndarray((2,), dtype=float64,
min=-0.866, max=-0.866, mean=-0.866),
```

```
(pid=24563)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                               'prev_reward': 0.0,
(pid=24563)                               'rnn_state': []},
(pid=24563)                     'type': 'PolicyEvalData'},
(pid=24563)                   { 'data': { 'agent_id': 'agent_1',
(pid=24563)                               'env_id': 6,
(pid=24563)                               'info': {},
(pid=24563)                               'obs': np.ndarray((2,), dtype=float64,
min=1.214, max=1.214, mean=1.214),
(pid=24563)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                               'prev_reward': 0.0,
(pid=24563)                               'rnn_state': []},
(pid=24563)                     'type': 'PolicyEvalData'},
(pid=24563)                   { 'data': { 'agent_id': 'agent_1',
(pid=24563)                               'env_id': 7,
(pid=24563)                               'info': {},
(pid=24563)                               'obs': np.ndarray((2,), dtype=float64,
min=1.041, max=1.041, mean=1.041),
(pid=24563)                               'prev_action': np.ndarray((), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                               'prev_reward': 0.0,
(pid=24563)                               'rnn_state': []},
(pid=24563)                     'type': 'PolicyEvalData'}]}
(pid=24563)
(pid=24563) 2019-10-05 13:25:04,170        INFO tf_run_builder.py:92 -- Executing
TF run without tracing. To dump TF timeline traces to disk, set the
TF_TIMELINE_DIR environment variable.
(pid=24563) 2019-10-05 13:25:04,297        INFO sampler.py:548 -- Outputs of
compute_actions():
(pid=24563)
(pid=24563) { 'agent_0': ( np.ndarray((8,), dtype=int64, min=2.0, max=13.0,
mean=5.375),
(pid=24563)                []],
(pid=24563)                { 'q_values': np.ndarray((8, 15), dtype=float32,
min=-1.007, max=1.919, mean=0.272)}),
(pid=24563)   'agent_1': ( np.ndarray((8,), dtype=int64, min=1.0, max=13.0,
mean=6.125),
(pid=24563)                []],
(pid=24563)                { 'q_values': np.ndarray((8, 15), dtype=float32,
min=-5.616, max=0.0, mean=-2.832)})}
(pid=24563)
(pid=24563) 2019-10-05 13:25:04,636        INFO sample_batch_builder.py:161 --
Trajectory fragment after postprocess_trajectory():
(pid=24563)
(pid=24563) { 'agent_0': { 'data': { 'actions': np.ndarray((32,), dtype=int64,
min=0.0, max=14.0, mean=6.156),
(pid=24563)                          'agent_index': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                          'dones': np.ndarray((32,), dtype=bool,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                          'eps_id': np.ndarray((32,), dtype=int64,
min=60241224.0, max=60241224.0, mean=60241224.0),
(pid=24563)                          'infos': np.ndarray((32,), dtype=object,
head={'delta': 0.6763565409015408}),
(pid=24563)                          'new_obs': np.ndarray((32, 2), dtype=float32,
min=-1.688, max=1.772, mean=0.028),
(pid=24563)                          'obs': np.ndarray((32, 2), dtype=float32,
```

```
min=-1.688, max=1.772, mean=0.1),
(pid=24563)                             'prev_actions': np.ndarray((32,),
dtype=int64, min=0.0, max=14.0, mean=6.156),
(pid=24563)                             'prev_rewards': np.ndarray((32,),
dtype=float32, min=0.0, max=0.4, mean=0.303),
(pid=24563)                             'q_values': np.ndarray((32, 15),
dtype=float32, min=-1.04, max=2.96, mean=0.428),
(pid=24563)                             'rewards': np.ndarray((32,), dtype=float32,
min=0.285, max=1.13, mean=0.886),
(pid=24563)                             't': np.ndarray((32,), dtype=int64, min=0.0,
max=31.0, mean=15.5),
(pid=24563)                             'unroll_id': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                             'weights': np.ndarray((32,), dtype=float32,
min=2.293, max=2.676, mean=2.406)},
(pid=24563)                  'type': 'SampleBatch'},
(pid=24563)   'agent_1': { 'data': { 'actions': np.ndarray((32,), dtype=int64,
min=0.0, max=14.0, mean=8.438),
(pid=24563)                             'agent_index': np.ndarray((32,), dtype=int64,
min=1.0, max=1.0, mean=1.0),
(pid=24563)                             'dones': np.ndarray((32,), dtype=bool,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                             'eps_id': np.ndarray((32,), dtype=int64,
min=60241224.0, max=60241224.0, mean=60241224.0),
(pid=24563)                             'infos': np.ndarray((32,), dtype=object,
head={'delta': 0.060679271417553256}),
(pid=24563)                             'new_obs': np.ndarray((32, 2), dtype=float32,
min=-1.688, max=1.772, mean=0.028),
(pid=24563)                             'obs': np.ndarray((32, 2), dtype=float32,
min=-1.688, max=1.772, mean=0.1),
(pid=24563)                             'prev_actions': np.ndarray((32,),
dtype=int64, min=0.0, max=14.0, mean=8.25),
(pid=24563)                             'prev_rewards': np.ndarray((32,),
dtype=float32, min=0.0, max=0.387, mean=0.249),
(pid=24563)                             'q_values': np.ndarray((32, 15),
dtype=float32, min=-5.672, max=1.451, mean=-1.837),
(pid=24563)                             'rewards': np.ndarray((32,), dtype=float32,
min=0.177, max=0.969, mean=0.72),
(pid=24563)                             't': np.ndarray((32,), dtype=int64, min=0.0,
max=31.0, mean=15.5),
(pid=24563)                             'unroll_id': np.ndarray((32,), dtype=int64,
min=0.0, max=0.0, mean=0.0),
(pid=24563)                             'weights': np.ndarray((32,), dtype=float32,
min=2.303, max=2.928, mean=2.536)},
(pid=24563)                  'type': 'SampleBatch'}}
(pid=24563)
(pid=24563) 2019-10-05 13:25:04,681        INFO rollout_worker.py:485 -- Completed
sample batch:
(pid=24563)
(pid=24563) { 'count': 256,
(pid=24563)   'policy_batches': { 'agent_0': { 'data': { 'actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.508),
(pid=24563)                                              'agent_index':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24563)                                              'dones':
np.ndarray((256,), dtype=bool, min=0.0, max=0.0, mean=0.0),
(pid=24563)                                              'eps_id':
np.ndarray((256,), dtype=int64, min=60241224.0, max=1819536995.0,
mean=885664162.75),
```

```
(pid=24563)                                                 'infos':
np.ndarray((256,), dtype=object, head={'delta': 0.6763565409015408}),
(pid=24563)                                                 'new_obs':
np.ndarray((256, 2), dtype=float32, min=-1.78, max=1.774, mean=-0.04),
(pid=24563)                                                 'obs': np.ndarray((256,
2), dtype=float32, min=-1.78, max=1.774, mean=-0.012),
(pid=24563)                                                 'prev_actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=6.32),
(pid=24563)                                                 'prev_rewards':
np.ndarray((256,), dtype=float32, min=0.0, max=0.42, mean=0.279),
(pid=24563)                                                 'q_values':
np.ndarray((256, 15), dtype=float32, min=-1.653, max=3.033, mean=0.523),
(pid=24563)                                                 'rewards':
np.ndarray((256,), dtype=float32, min=0.214, max=1.13, mean=0.814),
(pid=24563)                                                 't': np.ndarray((256,),
dtype=int64, min=0.0, max=31.0, mean=15.5),
(pid=24563)                                                 'unroll_id':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24563)                                                 'weights':
np.ndarray((256,), dtype=float32, min=2.267, max=2.752, mean=2.427)},
(pid=24563)                                       'type': 'SampleBatch'},
(pid=24563)                           'agent_1': { 'data': { 'actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=7.262),
(pid=24563)                                                 'agent_index':
np.ndarray((256,), dtype=int64, min=1.0, max=1.0, mean=1.0),
(pid=24563)                                                 'dones':
np.ndarray((256,), dtype=bool, min=0.0, max=0.0, mean=0.0),
(pid=24563)                                                 'eps_id':
np.ndarray((256,), dtype=int64, min=60241224.0, max=1819536995.0,
mean=885664162.75),
(pid=24563)                                                 'infos':
np.ndarray((256,), dtype=object, head={'delta': 0.060679271417553256}),
(pid=24563)                                                 'new_obs':
np.ndarray((256, 2), dtype=float32, min=-1.78, max=1.774, mean=-0.04),
(pid=24563)                                                 'obs': np.ndarray((256,
2), dtype=float32, min=-1.78, max=1.774, mean=-0.012),
(pid=24563)                                                 'prev_actions':
np.ndarray((256,), dtype=int64, min=0.0, max=14.0, mean=7.102),
(pid=24563)                                                 'prev_rewards':
np.ndarray((256,), dtype=float32, min=0.0, max=0.42, mean=0.26),
(pid=24563)                                                 'q_values':
np.ndarray((256, 15), dtype=float32, min=-5.907, max=1.451, mean=-2.062),
(pid=24563)                                                 'rewards':
np.ndarray((256,), dtype=float32, min=0.177, max=1.113, mean=0.763),
(pid=24563)                                                 't': np.ndarray((256,),
dtype=int64, min=0.0, max=31.0, mean=15.5),
(pid=24563)                                                 'unroll_id':
np.ndarray((256,), dtype=int64, min=0.0, max=0.0, mean=0.0),
(pid=24563)                                                 'weights':
np.ndarray((256,), dtype=float32, min=2.293, max=3.324, mean=2.597)},
(pid=24563)                                       'type': 'SampleBatch'}},
(pid=24563)   'type': 'MultiAgentBatch'}
(pid=24563)
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorflow/python/framework/dtypes.py:516: FutureWarning: Passing (type, 1) or
'1type' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_qint8 = np.dtype([("qint8", np.int8, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
```

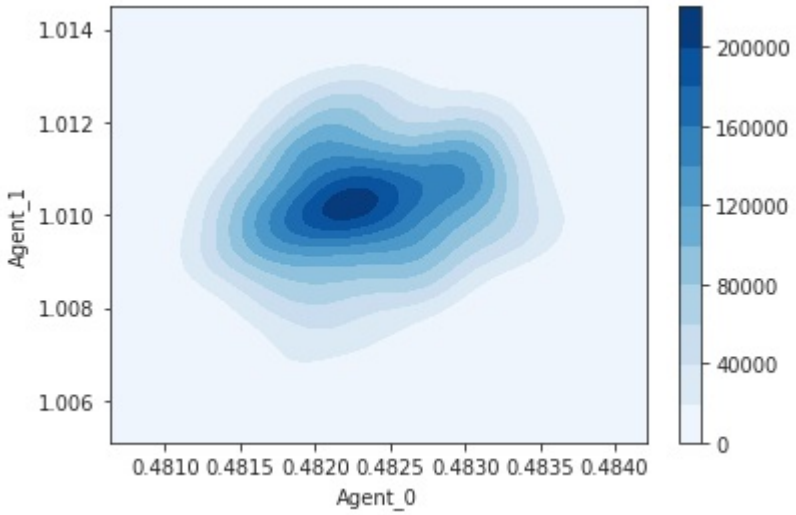
```
tensorflow/python/framework/dtypes.py:517: FutureWarning: Passing (type, 1) or
'1type' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorflow/python/framework/dtypes.py:518: FutureWarning: Passing (type, 1) or
'1type' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_qint16 = np.dtype([("qint16", np.int16, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorflow/python/framework/dtypes.py:519: FutureWarning: Passing (type, 1) or
'1type' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorflow/python/framework/dtypes.py:520: FutureWarning: Passing (type, 1) or
'1type' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_qint32 = np.dtype([("qint32", np.int32, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorflow/python/framework/dtypes.py:525: FutureWarning: Passing (type, 1) or
'1type' as a synonym of type is deprecated; in a future version of numpy, it will
be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   np_resource = np.dtype([("resource", np.ubyte, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorboard/compat/tensorflow_stub/dtypes.py:541: FutureWarning: Passing (type, 1)
or '1type' as a synonym of type is deprecated; in a future version of numpy, it
will be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_qint8 = np.dtype([("qint8", np.int8, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorboard/compat/tensorflow_stub/dtypes.py:542: FutureWarning: Passing (type, 1)
or '1type' as a synonym of type is deprecated; in a future version of numpy, it
will be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_quint8 = np.dtype([("quint8", np.uint8, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorboard/compat/tensorflow_stub/dtypes.py:543: FutureWarning: Passing (type, 1)
or '1type' as a synonym of type is deprecated; in a future version of numpy, it
will be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_qint16 = np.dtype([("qint16", np.int16, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorboard/compat/tensorflow_stub/dtypes.py:544: FutureWarning: Passing (type, 1)
or '1type' as a synonym of type is deprecated; in a future version of numpy, it
will be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_quint16 = np.dtype([("quint16", np.uint16, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorboard/compat/tensorflow_stub/dtypes.py:545: FutureWarning: Passing (type, 1)
or '1type' as a synonym of type is deprecated; in a future version of numpy, it
will be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   _np_qint32 = np.dtype([("qint32", np.int32, 1)])
(pid=24575) /home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
tensorboard/compat/tensorflow_stub/dtypes.py:550: FutureWarning: Passing (type, 1)
or '1type' as a synonym of type is deprecated; in a future version of numpy, it
will be understood as (type, (1,)) / '(1,)type'.
(pid=24575)   np_resource = np.dtype([("resource", np.ubyte, 1)])
(pid=24575) WARNING:tensorflow:From /home/lorenzo/anaconda3/envs/py36/lib/
python3.6/site-packages/tensorflow/python/compat/v2_compat.py:61:
disable_resource_variables (from tensorflow.python.ops.variable_scope) is
deprecated and will be removed in a future version.
(pid=24575) Instructions for updating:
```

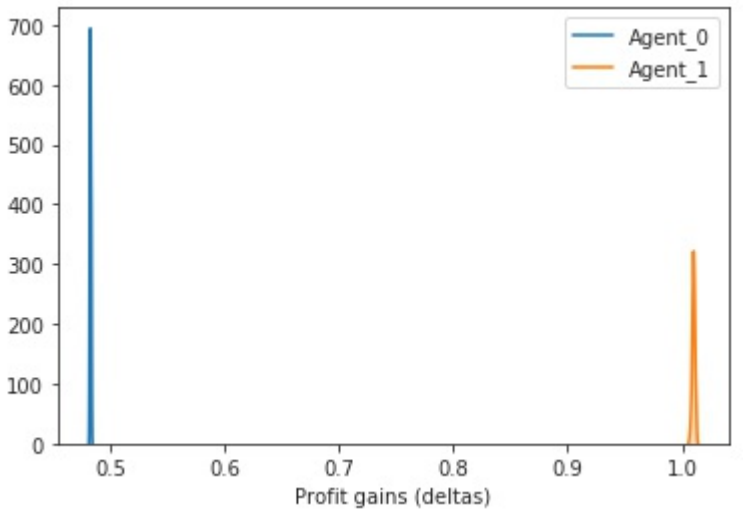
```
(pid=24575) non-resource variables are not supported in the long term
Episode reward 6182.565421237154
Episode 1 of 100
Episode reward 6180.119738393753
Episode 2 of 100
Episode reward 6183.563568267211
Episode 3 of 100
Episode reward 6182.467664389923
Episode 4 of 100
Episode reward 6182.335274299167
Episode 5 of 100
Episode reward 6184.533465302735
Episode 6 of 100
Episode reward 6185.969934606965
Episode 7 of 100
Episode reward 6181.4867746687805
Episode 8 of 100
Episode reward 6182.670452301463
Episode 9 of 100
Episode reward 6183.829958155293
Episode 10 of 100
Episode reward 6185.11063003251
Episode 11 of 100
Episode reward 6182.089106470671
Episode 12 of 100
Episode reward 6185.440612916957
Episode 13 of 100
Episode reward 6183.080900822606
Episode 14 of 100
Episode reward 6184.115634849413
Episode 15 of 100
Episode reward 6185.28615708174
Episode 16 of 100
Episode reward 6184.195550288678
Episode 17 of 100
Episode reward 6184.960147805719
Episode 18 of 100
Episode reward 6183.55842189513
Episode 19 of 100
Episode reward 6182.952106491426
Episode 20 of 100
Episode reward 6183.881293213472
Episode 21 of 100
Episode reward 6183.916924761798
Episode 22 of 100
Episode reward 6183.875378564239
Episode 23 of 100
Episode reward 6180.40049462572
Episode 24 of 100
Episode reward 6181.980333003391
Episode 25 of 100
Episode reward 6181.425973648642
Episode 26 of 100
Episode reward 6183.85709319694
Episode 27 of 100
Episode reward 6182.2180848368735
Episode 28 of 100
Episode reward 6184.3198407699865
Episode 29 of 100
```

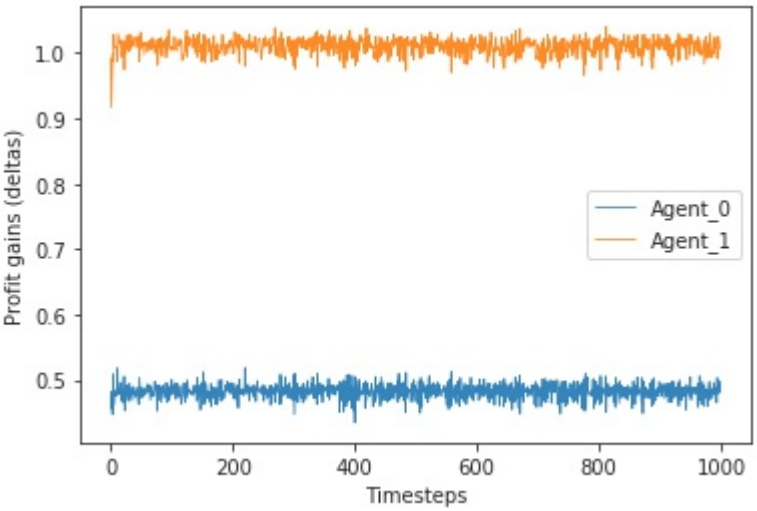
```
Episode reward 6182.984505045675
Episode 30 of 100
Episode reward 6180.002159721872
Episode 31 of 100
Episode reward 6182.569576886728
Episode 32 of 100
Episode reward 6185.857236377929
Episode 33 of 100
Episode reward 6186.055770927456
Episode 34 of 100
Episode reward 6184.337547368531
Episode 35 of 100
Episode reward 6183.144979767602
Episode 36 of 100
Episode reward 6183.324690328708
Episode 37 of 100
Episode reward 6185.55734719023
Episode 38 of 100
Episode reward 6185.166053657725
Episode 39 of 100
Episode reward 6185.18079339632
Episode 40 of 100
Episode reward 6184.066185157365
Episode 41 of 100
Episode reward 6183.774371710554
Episode 42 of 100
Episode reward 6182.157543351078
Episode 43 of 100
Episode reward 6186.141368188629
Episode 44 of 100
Episode reward 6184.965618953217
Episode 45 of 100
Episode reward 6182.130101133922
Episode 46 of 100
Episode reward 6182.935505735654
Episode 47 of 100
Episode reward 6182.35402401719
Episode 48 of 100
Episode reward 6178.815618020387
Episode 49 of 100
Episode reward 6181.1864891676105
Episode 50 of 100
Episode reward 6182.787290663906
Episode 51 of 100
Episode reward 6184.33410356036
Episode 52 of 100
Episode reward 6185.477739085409
Episode 53 of 100
Episode reward 6181.433799922986
Episode 54 of 100
Episode reward 6182.870930460592
Episode 55 of 100
Episode reward 6181.78062171189
Episode 56 of 100
Episode reward 6181.056655641465
Episode 57 of 100
Episode reward 6184.667502455752
Episode 58 of 100
Episode reward 6183.661562221587
```

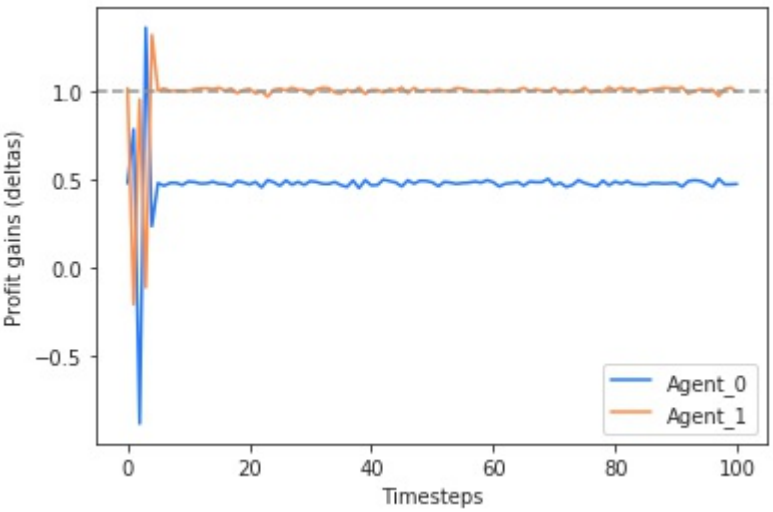
```
Episode 59 of 100
Episode reward 6184.297440265817
Episode 60 of 100
Episode reward 6184.459351574459
Episode 61 of 100
Episode reward 6183.686310658015
Episode 62 of 100
Episode reward 6183.61577065393
Episode 63 of 100
Episode reward 6182.521897763331
Episode 64 of 100
Episode reward 6183.879095834639
Episode 65 of 100
Episode reward 6185.3391359555635
Episode 66 of 100
Episode reward 6182.72005794863
Episode 67 of 100
Episode reward 6183.277598493502
Episode 68 of 100
Episode reward 6183.442583263032
Episode 69 of 100
Episode reward 6184.068760200975
Episode 70 of 100
Episode reward 6185.037808710487
Episode 71 of 100
Episode reward 6183.358833628903
Episode 72 of 100
Episode reward 6181.83105964347
Episode 73 of 100
Episode reward 6182.603078245047
Episode 74 of 100
Episode reward 6182.836279002683
Episode 75 of 100
Episode reward 6185.003047760463
Episode 76 of 100
Episode reward 6182.478996178559
Episode 77 of 100
Episode reward 6180.852984043968
Episode 78 of 100
Episode reward 6181.979602270177
Episode 79 of 100
Episode reward 6182.776990552143
Episode 80 of 100
Episode reward 6184.045228181041
Episode 81 of 100
Episode reward 6180.318131939607
Episode 82 of 100
Episode reward 6180.272640093059
Episode 83 of 100
Episode reward 6183.237592646312
Episode 84 of 100
Episode reward 6184.82619588429
Episode 85 of 100
Episode reward 6181.231828207139
Episode 86 of 100
Episode reward 6184.685530450856
Episode 87 of 100
Episode reward 6183.12223488946
Episode 88 of 100
```

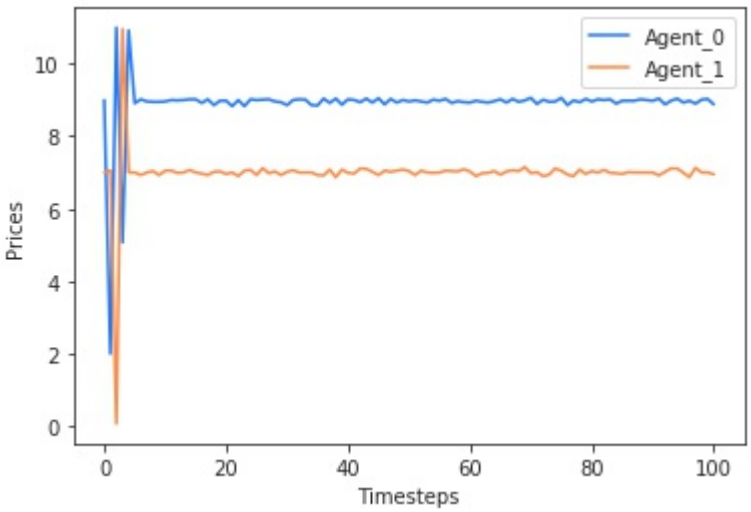
```
Episode reward 6184.618554483392
Episode 89 of 100
Episode reward 6181.797387052291
Episode 90 of 100
Episode reward 6181.31011277267
Episode 91 of 100
Episode reward 6182.581653331954
Episode 92 of 100
Episode reward 6184.083445358256
Episode 93 of 100
Episode reward 6184.008267174556
Episode 94 of 100
Episode reward 6184.3189601941
Episode 95 of 100
Episode reward 6182.759950141808
Episode 96 of 100
Episode reward 6182.014918052456
Episode 97 of 100
Episode reward 6180.911080112808
Episode 98 of 100
Episode reward 6181.512424420543
Episode 99 of 100
Episode reward 6183.462642906648
Overall deltas mean: 0.7463 and std: 0.2640
Agent0 deltas mean: 0.4823 and std: 0.0005
Agent1 deltas mean: 1.0102 and std: 0.0012
/home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/seaborn/
distributions.py:679: UserWarning: Passing a 2D dataset for a bivariate plot is
deprecated in favor of kdeplot(x, y), and it will cause an error in future
versions. Please update your code.
  warnings.warn(warn_msg, UserWarning)
```




Agent_0
Agent_1
700
600
500
400
300
200
100
0
0.5
0.6
0.7
0.8
0.9
1.0
Profit gains (deltas)


Agent_0
Agent_1
1.0
0.9
0.8
0.7
0.6
0.5
Profit gains (deltas)
0
200
400
600
800
1000
Timesteps


Agent_0
Agent_1
1.0
0.5
0.0
−0.5
Profit gains (deltas)
0
20
40
60
80
100
Timesteps




```
Traceback (most recent call last):

  File "<ipython-input-1-9464633410d8>", line 1, in <module>
    runfile('/home/lorenzo/algorithmic-pricing/rollout/rollout.py', args='/home/
lorenzo/Desktop/algorithmic-pricing/train_results/Azure_ApexDQN_Disc/
azure_disc_10_res1/APEX_MultiAgentFirmsPricing_0_2019-09-21_10-29-54_19h51zm/
checkpoint_360/checkpoint-360 --run APEX --env env_disc', wdir='/home/lorenzo/
algorithmic-pricing/rollout')

  File "/home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
spyder_kernels/customize/spydercustomize.py", line 827, in runfile
    execfile(filename, namespace)

  File "/home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/
spyder_kernels/customize/spydercustomize.py", line 110, in execfile
    exec(compile(f.read(), filename, 'exec'), namespace)

  File "/home/lorenzo/algorithmic-pricing/rollout/rollout.py", line 404, in
<module>
    Deltas_df = pd.DataFrame(d_array)

  File "/home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/pandas/core/
frame.py", line 440, in __init__
    mgr = init_ndarray(data, index, columns, dtype=dtype, copy=copy)

  File "/home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/pandas/core/
internals/construction.py", line 171, in init_ndarray
    values = prep_ndarray(values, copy=copy)

  File "/home/lorenzo/anaconda3/envs/py36/lib/python3.6/site-packages/pandas/core/
internals/construction.py", line 295, in prep_ndarray
    raise ValueError("Must pass 2-d input")

ValueError: Must pass 2-d input


In [2]:

In [2]:
```